aws

Entwicklerhandbuch

Amazon CloudFront

Amazon CloudFront: Entwicklerhandbuch

Table of Contents

Was ist Amazon CloudFront?

Amazon CloudFront ist ein Webservice, der die Verteilung Ihrer statischen und dynamischen Webinhalte wie .html-, .css-, .js- und Bilddateien an Ihre Benutzer beschleunigt. CloudFront stellt Ihre Inhalte über ein weltweites Netzwerk von Rechenzentren bereit, die als Edge-Standorte bezeichnet werden. Wenn ein Benutzer Inhalte anfordert CloudFront, mit denen Sie sie bereitstellen, wird die Anfrage an den Edge-Standort weitergeleitet, der die niedrigste Latenz (Zeitverzögerung) bietet, sodass der Inhalt mit der bestmöglichen Leistung bereitgestellt wird.

- Wenn sich der Inhalt bereits am Edge-Standort mit der geringsten Latenz befindet, wird er CloudFront sofort zugestellt.
- Wenn sich der Inhalt nicht an diesem Edge-Standort befindet, CloudFront ruft er ihn von einem von Ihnen definierten Ursprung ab, z. B. einem Amazon S3 S3-Bucket, einem MediaPackage Channel oder einem HTTP-Server (z. B. einem Webserver), den Sie als Quelle für die endgültige Version Ihres Inhalts identifiziert haben.

Nehmen wir an, dass Sie ein Bild über einen traditionellen Webserver und nicht mit CloudFront bereitstellen. Sie können beispielsweise ein Bild, sunsetphoto.png, über die URL bereitstelle `https://example.com/sunsetphoto.png`.

Ihre Benutzer können problemlos zu dieser URL navigieren und das Bild anzeigen. Aber sie wissen wahrscheinlich nicht, dass ihre Anfrage von einem Netzwerk zu einem anderen weitergeleitet wird, bis das Bild schließlich gefunden wurde – durch die komplexe Sammlung miteinander verbundener Netzwerke, aus dem das Internet besteht.

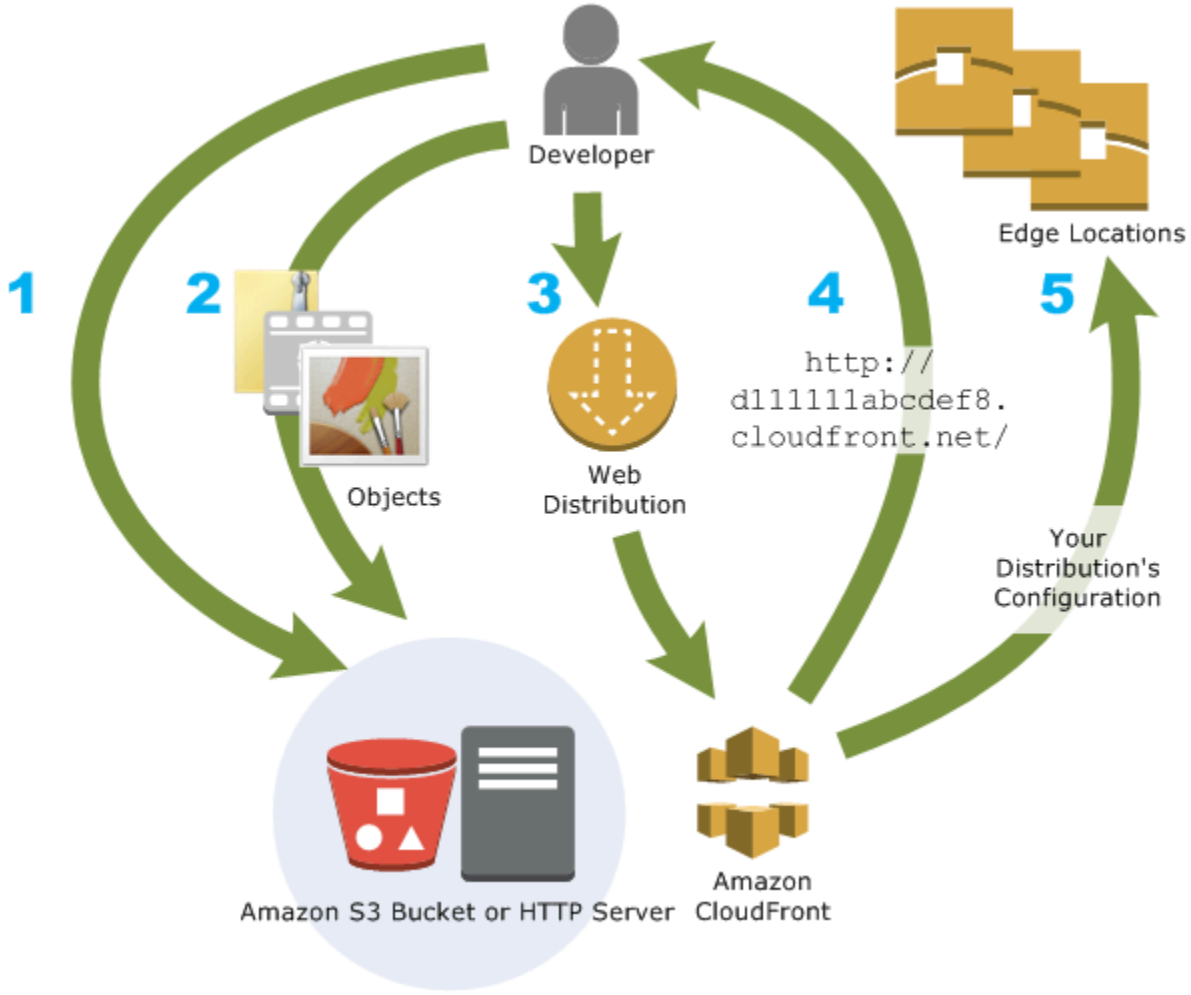
CloudFront beschleunigt die Verteilung Ihrer Inhalte, indem jede Benutzeranfrage über das AWS Backbone-Netzwerk an den Edge-Standort weitergeleitet wird, der Ihre Inhalte am besten bereitstellen kann. In der Regel handelt es sich dabei um einen CloudFront Edge-Server, der dem Betrachter die schnellste Übertragung ermöglicht. Durch die Verwendung des AWS Netzwerks wird die Anzahl der Netzwerke, die die Anfragen Ihrer Benutzer durchlaufen müssen, drastisch reduziert, wodurch die Leistung verbessert wird. Benutzer erfahren eine geringere Latenz – die Zeit bis zum Laden des ersten Bytes der Datei – und erzielen höhere Datenübertragungsraten.

Darüber hinaus erhöht sich die Zuverlässigkeit und die Verfügbarkeit, da Kopien Ihrer Dateien (auch als Objekte bezeichnet) jetzt an mehreren Edge-Standorten auf der ganzen Welt verfügbar (oder dort im Cache gespeichert) sind.

Themen

Wie richten Sie CloudFront die Bereitstellung von Inhalten ein

Sie erstellen eine CloudFront Verteilung, um anzugeben CloudFront , von wo aus die Inhalte geliefert werden sollen, und legen die Einzelheiten zur Nachverfolgung und Verwaltung der Inhaltsbereitstellung fest. Anschließend werden Computer — Edge-Server — CloudFront verwendet, die sich in der Nähe Ihrer Zuschauer befinden, um diese Inhalte schnell bereitzustellen, wenn jemand sie sehen oder verwenden möchte.



Wie konfigurierst du CloudFront , um deine Inhalte bereitzustellen

1. Sie geben Ursprungsserver an, z. B. einen Amazon S3 S3-Bucket oder Ihren eigenen HTTP-Server, von denen CloudFront Ihre Dateien abgerufen werden, die dann von CloudFront Edge-Standorten auf der ganzen Welt verteilt werden.

   Auf einem Ursprungsserver sind die originalen, definitiven Versionen Ihrer Dateien gespeichert. Wenn Sie Inhalte über HTTP bereitstellen, ist Ihr Ursprungs-Server entweder ein Amazon S3-Bucket oder ein HTTP-Server, z. B. ein Web-Server. Ihr HTTP-Server kann auf einer Amazon Elastic Compute Cloud (Amazon EC2) -Instance oder auf einem von Ihnen verwalteten Server ausgeführt werden. Diese Server werden auch als benutzerdefinierte Ursprünge bezeichnet.
2. Sie laden Ihre Dateien auf Ihre Ursprungsserver hoch. Ihre Dateien, auch als Objekte bezeichnet, enthalten normalerweise Webseiten, Bilder und Mediendateien, können jedoch alles sein, was über HTTP bereitgestellt werden kann.

Wenn Sie einen Amazon S3 S3-Bucket als Ursprungsserver verwenden, können Sie die Objekte in Ihrem Bucket öffentlich lesbar machen, sodass jeder, der die CloudFront URLs für Ihre Objekte kennt, darauf zugreifen kann. Sie haben auch die Möglichkeit, private Objekte einzurichten und den Zugriff zu kontrollieren. Siehe Stellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit.

3. Sie erstellen eine CloudFront Distribution, die angibt, von CloudFront welchen Ursprungsservern Ihre Dateien abgerufen werden sollen, wenn Benutzer die Dateien über Ihre Website oder Anwendung anfordern. Gleichzeitig geben Sie Details an, z. B. ob Sie alle Anfragen protokollieren CloudFront möchten und ob die Verteilung aktiviert werden soll, sobald sie erstellt wurde.
4. CloudFront weist Ihrer neuen Distribution einen Domainnamen zu, den Sie in der CloudFront Konsole sehen können oder der in der Antwort auf eine programmatische Anfrage zurückgegeben wird, z. B. eine API-Anfrage. Wenn Sie möchten, können Sie einen alternativen Domänennamen hinzufügen, sodass er stattdessen verwendet wird.
5. CloudFront sendet die Konfiguration Ihrer Distribution (aber nicht Ihre Inhalte) an all ihre Edge-Standorte oder Points of Presence (POPs) — Serversammlungen in geografisch verteilten Rechenzentren, in denen CloudFront Kopien Ihrer Dateien zwischengespeichert werden.

Bei der Entwicklung Ihrer Website oder Anwendung verwenden Sie den Domainnamen, der für Ihre Website oder Anwendung vorgesehen ist. CloudFront URLs Wenn beispielsweise `d111111abcdef8.cloudfront.net` als Domainname für Ihre Distribution CloudFront zurückgegeben wird, lautet die URL für logo.jpg in Ihrem Amazon S3 S3-Bucket (oder im Stammverzeichnis auf einem HTTP-Server)`https://d111111abcdef8.cloudfront.net/logo.jpg`.

Oder Sie können einrichten CloudFront , dass Sie Ihren eigenen Domainnamen für Ihre Distribution verwenden. In diesem Fall könnte die URL folgendermaßen lauten: `https://www.example.com/logo.jpg`.

Optional können Sie Ihren Ursprungsserver so konfigurieren, dass er den Dateien Header hinzufügt, um anzugeben, wie lange die Dateien an CloudFront Edge-Standorten im Cache verbleiben sollen. Standardmäßig wird jede Datei 24 Stunden lang an einem Edge-Standort gespeichert; danach läuft sie ab. Die minimale Ablaufzeit beträgt 0 Sekunden; eine maximale Ablaufzeit existiert nicht. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Wählen Sie zwischen Standardverteilung oder Mehrmandantenverteilung

CloudFront bietet Vertriebsoptionen für einzelne Websites oder Apps sowie für Szenarien mit mehreren Mandanten.

Standardverteilung

Konzipiert für einzigartige Konfigurationen pro Website oder Anwendung. Wählen Sie dies in den folgenden Anwendungsfällen:

- Sie benötigen eine eigenständige CloudFront Distribution
- Jede Site oder Anwendung benötigt ihre eigenen benutzerdefinierten Einstellungen

Die meisten Leute beginnen mit einer Standarddistribution.

Mehrinstanzenverteilungs- und Vertriebsmandanten (CloudFront SaaS Manager)

Speziell für SaaS-Anbieter und Multi-Tenant-Szenarien konzipiert. Wählen Sie dies in den folgenden Anwendungsfällen:

- Sie bauen eine SaaS-Plattform auf, um mehrere Kundenwebsites oder -anwendungen zu bedienen
- Sie müssen mehrere ähnliche Distributionen effizient verwalten
- Sie möchten eine zentrale Kontrolle über gemeinsam genutzte Konfigurationen

Weitere Informationen finden Sie unter Erfahren Sie, wie Distributionen mit mehreren Mandanten funktionieren.

Preisgestaltung

CloudFront Gebühren für Datenübertragungen von den Edge-Standorten aus sowie für HTTP- oder HTTPS-Anfragen. Die Preise variieren je nach Nutzungsart, geografischer Region und Funktionsauswahl.

Die Datenübertragung von Ihrem Ursprung zu CloudFront ist immer kostenlos, wenn Sie AWS Ursprünge wie Amazon Simple Storage Service (Amazon S3), Elastic Load Balancing oder Amazon API Gateway verwenden. Wenn Sie Origins verwenden AWS , wird Ihnen nur die ausgehende Datenübertragung vom CloudFront zum Viewer in Rechnung gestellt.

Weitere Informationen findest du unter CloudFront Preise und im Paket „Abrechnung und Sparen“. FAQs

Verwendungsmöglichkeiten CloudFront

Die Verwendung CloudFront kann Ihnen dabei helfen, eine Vielzahl von Zielen zu erreichen. In diesem Abschnitt werden nur einige aufgelistet, zusammen mit Links zu weiteren Informationen, um Ihnen eine Vorstellung davon zu geben, welche Möglichkeiten Sie haben.

Themen

Beschleunigen der Bereitstellung von Inhalten auf einer statischen Website

CloudFront kann die Bereitstellung Ihrer statischen Inhalte (z. B. Bilder, Stylesheets usw.) für Zuschauer auf der ganzen Welt beschleunigen. JavaScript Durch die Verwendung CloudFront können Sie das AWS Backbone-Netzwerk und die CloudFront Edge-Server nutzen, um Ihren Zuschauern ein schnelles, sicheres und zuverlässiges Erlebnis zu bieten, wenn sie Ihre Website besuchen.

Ein einfacher Ansatz für das Speichern und Bereitstellen von statischen Inhalten ist die Verwendung eines Amazon S3-Buckets. Die Verwendung von S3 zusammen mit CloudFront bietet eine Reihe von Vorteilen, einschließlich der Option, Origin Access Control den Zugriff auf Ihre Amazon S3 S3-Inhalte einfach einzuschränken.

Weitere Informationen zur Verwendung von Amazon S3 zusammen mit CloudFront, einschließlich einer AWS CloudFormation Vorlage, die Ihnen den schnellen Einstieg erleichtert, finden Sie unterBeginnen Sie mit einer sicheren statischen Website.

Bereitstellen von On-Demand- oder Live-Streaming-Video

CloudFront bietet mehrere Optionen für das Streamen Ihrer Medien an Zuschauer weltweit — sowohl vorab aufgezeichnete Dateien als auch Live-Events.

- Für Video-on-Demand-Streaming (VOD) können Sie es verwenden, CloudFront um in gängigen Formaten wie MPEG DASH, Apple HLS, Microsoft Smooth Streaming und CMAF auf jedes Gerät zu streamen.
- Für das Broadcasting eines Live-Streams können Sie Medienfragmente am Edge-Standort zwischenspeichern, sodass mehrere Anforderungen an die Manifestdatei, die die Fragmente in der richtigen Reihenfolge bereitstellt, kombiniert werden können, um den Workload auf Ihrem Ursprungs-Server zu verringern.

Weitere Informationen zur Bereitstellung von Streaming-Inhalten mit finden Sie unter. CloudFront Video-on-Demand und Live-Streaming-Video mit CloudFront

Verschlüsseln von bestimmten Feldern während der Systemverarbeitung

Wenn Sie HTTPS mit konfigurieren CloudFront, verfügen Sie bereits über sichere end-to-end Verbindungen zu den Ursprungsservern. Wenn Sie eine Verschlüsselung auf Feldebene hinzufügen, können Sie neben der HTTPS-Sicherheit bestimmte Daten während der gesamten Systemverarbeitung schützen, sodass nur bestimmte Anwendungen an ihrem Ursprung die Daten sehen können.

Um die Verschlüsselung auf Feldebene einzurichten, fügen Sie einen öffentlichen Schlüssel hinzu CloudFront und geben dann die Gruppe von Feldern an, die mit dem Schlüssel verschlüsselt werden sollen. Weitere Informationen finden Sie unter Vertrauliche Daten durch Verschlüsselung auf Feldebene schützen.

An der Grenze anpassen

Durch das Ausführen von serverlosem Code am Edge-Standort ergeben sich eine Reihe von Möglichkeiten zum Anpassen der Inhalte und Erfahrungen für Betrachter, bei reduzierter Latenz. Sie können zum Beispiel eine benutzerdefinierte Fehlermeldung zurückgeben, wenn Ihr Ursprungs-Server wegen Wartungsarbeiten nicht verfügbar ist, damit Betrachter keine allgemeine HTTP-Fehlermeldung erhalten. Oder Sie können eine Funktion verwenden, mit der Sie Benutzer autorisieren und den Zugriff auf Ihre Inhalte kontrollieren können, bevor Sie eine Anfrage an Ihren CloudFront Absender weiterleiten.

Die Verwendung von Lambda @Edge mit CloudFront ermöglicht eine Vielzahl von Möglichkeiten, den bereitgestellten Inhalt anzupassenCloudFront . Weitere Informationen zu Lambda@Edge und dazu, wie Sie Funktionen mit CloudFront erstellen und bereitstellen können, finden Sie unter

Personalisieren Sie am Rand mit Lambda @Edge. Eine Reihe von Codebeispielen, die Sie für Ihre eigenen Lösungen anpassen können, finden Sie unter Beispielfunktionen für Lambda@Edge.

Bereitstellen von privaten Inhalten mit Hilfe von Lambda@Edge-Anpassungen

Die Verwendung von Lambda @Edge kann Ihnen helfen, Ihre CloudFront Distribution so zu konfigurieren, dass private Inhalte von Ihrem eigenen benutzerdefinierten Ursprung bereitgestellt werden, zusätzlich zur Verwendung signierter URLs oder signierter Cookies.

Um private Inhalte bereitzustellen CloudFront, gehen Sie wie folgt vor:

- Erfordern Sie, dass Ihre Benutzer (Zuschauer) mithilfe signierter URLs oder signierter Cookies auf Inhalte zugreifen.
- Beschränken Sie den Zugriff auf Ihren Ursprungsserver, sodass er nur auf Servern verfügbar ist, die mit dem CloudFront Ursprung verbunden sind. Dazu können Sie einen der folgenden Schritte ausführen:
  - Für einen Amazon-S3-Ursprung können Sie eine Ursprungszugriffssteuerung (OAC) verwenden.
  - Für einen benutzerdefinierten Ursprung können Sie folgendermaßen vorgehen:
    - Wenn der benutzerdefinierte Ursprung durch eine Amazon VPC-Sicherheitsgruppe geschützt ist oder AWS Firewall Manager, können Sie die Liste der CloudFront verwalteten Präfixe verwenden, um eingehenden Datenverkehr zu Ihrem Ursprung nur CloudFront von den ursprünglichen IP-Adressen zuzulassen.
    - Verwenden Sie einen benutzerdefinierten HTTP-Header, um den Zugriff nur auf Anfragen von zu beschränken. CloudFront Weitere Informationen erhalten Sie unter the section called “Beschränken Sie den Zugriff auf Dateien mit benutzerdefinierten Ursprüngen” und the section called “Fügen Sie benutzerdefinierte Header zu ursprünglichen Anfragen hinzu”. Ein Beispiel, das einen benutzerdefinierten Header verwendet, um den Zugriff auf einen Application-Load-Balancer-Ursprung einzuschränken, finden Sie unter the section called “Beschränken Sie den Zugriff auf Application Load Balancers”.
    - Wenn der benutzerdefinierte Ursprung eine benutzerdefinierte Zugriffskontrolllogik erfordert, können Sie Lambda @Edge verwenden, um diese Logik zu implementieren, wie in diesem Blogbeitrag beschrieben: Bereitstellung privater Inhalte mit Amazon CloudFront & Lambda @Edge.

Wie liefert Inhalte CloudFront

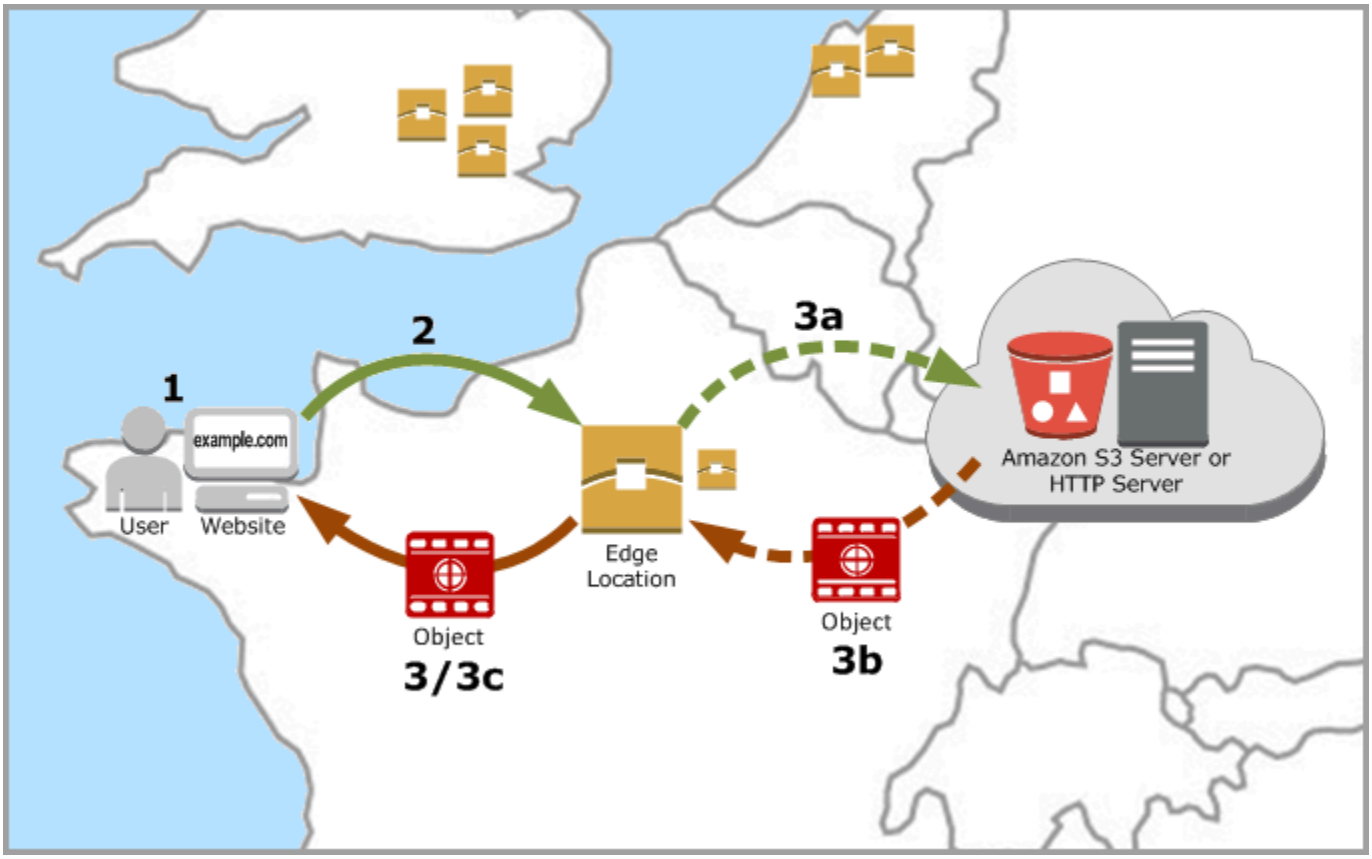
CloudFront Funktioniert nach einiger Ersteinrichtung mit Ihrer Website oder Anwendung zusammen und beschleunigt die Bereitstellung Ihrer Inhalte. In diesem Abschnitt wird erklärt, CloudFront wie Ihre Inhalte bereitgestellt werden, wenn Zuschauer sie anfordern.

Themen

Wie CloudFront werden Inhalte für Ihre Nutzer bereitgestellt

Nachdem Sie CloudFront die Konfiguration für die Bereitstellung Ihrer Inhalte konfiguriert haben, passiert Folgendes, wenn Benutzer Ihre Objekte anfordern:

1. Ein Benutzer greift auf Ihre Website oder Anwendung zu und sendet eine Anforderung für ein Objekt, z. B. eine Bilddatei und eine HTML-Datei.
2. DNS leitet die Anfrage an den CloudFront POP (Edge-Standort) weiter, der die Anfrage am besten bearbeiten kann. In der Regel handelt es sich dabei um den CloudFront POP, der in Bezug auf die Latenz am nächsten ist.
3. CloudFront überprüft seinen Cache auf das angeforderte Objekt. Wenn sich das Objekt im Cache befindet, wird es an den Benutzer CloudFront zurückgegeben. Wenn sich das Objekt nicht im Cache befindet, CloudFront geht Folgendes vor:
   a. CloudFront vergleicht die Anfrage mit den Spezifikationen in Ihrer Distribution und leitet die Anfrage an Ihren Ursprungsserver für das entsprechende Objekt weiter, z. B. an Ihren Amazon S3 S3-Bucket oder Ihren HTTP-Server.
   b. Der Ursprungsserver sendet das Objekt an den Edge-Standort zurück.
   c. Sobald das erste Byte vom Ursprung eingeht, CloudFront beginnt die Weiterleitung des Objekts an den Benutzer. CloudFront fügt das Objekt auch dem Cache hinzu, damit es das nächste Mal von jemandem angefordert wird.



Wie CloudFront funktioniert mit regionalen Edge-Caches

CloudFront Präsenzpunkte (auch bekannt als POPsRandstandorte) stellen sicher, dass beliebte Inhalte Ihren Zuschauern schnell bereitgestellt werden können. CloudFront verfügt außerdem über regionale Edge-Caches, die mehr Ihrer Inhalte Ihren Zuschauern näher bringen, auch wenn die Inhalte nicht beliebt genug sind, um bei einem POP zu bleiben, um die Leistung dieser Inhalte zu verbessern.

Regionale Edge-Caches sind für alle Arten von Inhalten nützlich, insbesondere wenn die Inhalte im Laufe der Zeit immer seltener abgerufen werden. Dazu gehören von Benutzern erstellte Inhalte wie Videos, Fotos oder Bildmaterial, E-Commerce-Elemente wie Produktfotos und -videos sowie Nachrichten und ereignisbezogene Inhalte, die plötzlich sehr stark angefragt werden können.

So funktionieren regionale Caches

Regionale Edge-Caches sind CloudFront Standorte, die weltweit in der Nähe Ihrer Zuschauer bereitgestellt werden. Sie befinden sich zwischen Ihrem Ursprungsserver und den POPs globalen Edge-Standorten, an denen Inhalte direkt für Zuschauer bereitgestellt werden. Wenn Objekte weniger beliebt werden, POPs kann es sein, dass einzelne Personen diese Objekte entfernen, um Platz für

populärere Inhalte zu schaffen. Regionale Edge-Caches verfügen über mehr Cache-Speicherplatz als ein einzelner POP, sodass Ihre Objekte am nächstgelegenen Standort eines regionalen Edge-Cache länger gespeichert werden. Auf diese Weise können Sie Ihren Zuschauern mehr Inhalte näher bringen, sodass Sie nicht mehr CloudFront zu Ihrem Ursprungsserver zurückkehren müssen und die Gesamtleistung für die Zuschauer verbessert wird.

Wenn ein Betrachter eine Anfrage auf Ihrer Website oder über Ihre Anwendung sendet, leitet DNS die Anfrage an den POP weiter, der die Anforderung des Benutzers am besten bedienen kann. Dieser Standort ist in der Regel der nächstgelegene CloudFront Edge-Standort, was die Latenz angeht. Sucht im POP CloudFront im Cache nach dem angeforderten Objekt. Wenn sich das Objekt im Cache befindet, wird es an den Benutzer CloudFront zurückgegeben. Wenn das Objekt nicht im Cache vorhanden ist, wechseln die POPs zum nächstgelegenen regionalen Edge-Cache, um es abzurufen. Weitere Informationen darüber, wann der POP den regionalen Edge-Cache überspringt und direkt zum Ursprung wechselt, finden Sie in der folgenden Anmerkung.

Überprüft den Cache am regionalen Edge-Cache-Standort CloudFront erneut auf das angeforderte Objekt. Wenn sich das Objekt im Cache befindet, CloudFront leitet es an den POP weiter, der es angefordert hat. Sobald das erste Byte vom regionalen Edge-Cache-Standort eintrifft, CloudFront beginnt die Weiterleitung des Objekts an den Benutzer. CloudFront fügt das Objekt außerdem dem Cache im POP hinzu, damit es das nächste Mal von jemandem angefordert wird.

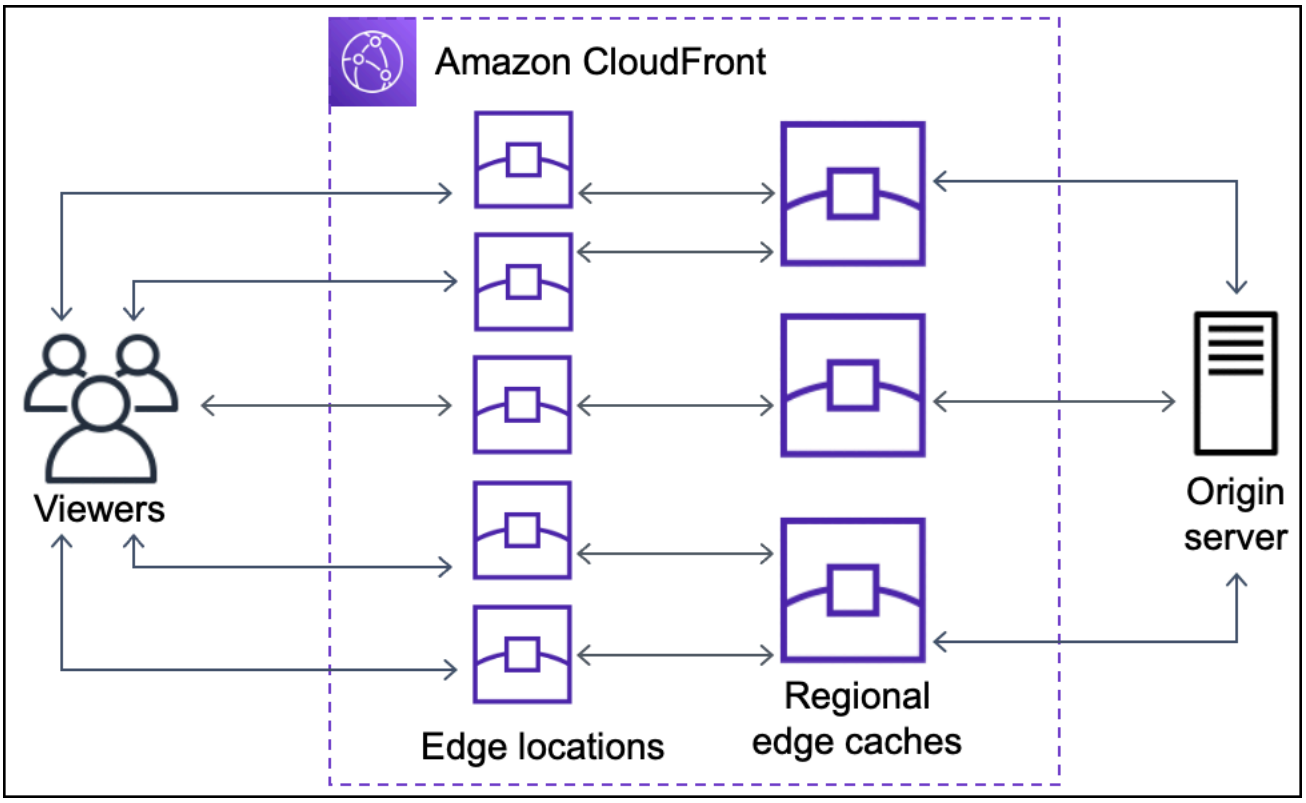
Bei Objekten, die weder am POP- noch am regionalen Edge-Cache-Speicherort zwischengespeichert wurden, CloudFront vergleicht die Anfrage mit den Spezifikationen in Ihren Distributionen und leitet die Anfrage an den Ursprungsserver weiter. Nachdem Ihr Ursprungsserver das Objekt an den regionalen Edge-Cache-Speicherort zurückgesendet hat, wird es an den POP und dann an den CloudFront Benutzer weitergeleitet. In diesem Fall fügt das Objekt zusätzlich zum POP CloudFront auch dem Cache am regionalen Edge-Cache-Speicherort hinzu, wenn ein Betrachter es das nächste Mal anfordert. Dadurch wird sichergestellt, dass sich alle POPs in einer Region einen lokalen Cache teilen, wodurch mehrere Anfragen an die Originalserver vermieden werden. CloudFront hält außerdem persistente Verbindungen zu den Ursprungsservern aufrecht, sodass Objekte so schnell wie möglich von den Ursprungsservern abgerufen werden.

> **Note**
>
> - Regionale Edge-Caches haben eine Feature-Parität mit. POPs Eine Cache-Invalidierungsanforderung entfernt ein Objekt z. B. sowohl aus POP-Caches als auch aus regionalen Edge-Caches, bevor es abgelaufen ist. Wenn einen Betrachter das Objekt das

nächste Mal anfordert, bezieht CloudFront die neueste Version des Objekts wieder von dem Ursprungsserver.

- Proxy-HTTP-Methoden (`PUT`,, `POST` `PATCH` `OPTIONS`, und`DELETE`) leiten von den regionalen Edge-Caches direkt zum Ursprung POPs und verwenden keinen Proxy über die regionalen Edge-Caches.
- Dynamische Anforderungen, die zum Anforderungszeitpunkt bestimmt werden, fließen nicht durch regionale Edge-Caches, sondern gehen direkt zum Ursprung.
- Wenn der Ursprung ein Amazon S3 S3-Bucket ist und sich der optimale regionale Edge-Cache der Anfrage im selben AWS-Region wie der S3-Bucket befindet, überspringt der POP den regionalen Edge-Cache und geht direkt zum S3-Bucket.

Das folgende Diagramm zeigt, wie Anfragen und Antworten durch CloudFront Edge-Standorte und regionale Edge-Caches fließen.



Standorte und IP-Adressbereiche von CloudFront Edge-Servern

Eine Liste der Standorte der CloudFront Edge-Server finden Sie auf der Amazon CloudFront Global Edge Network-Seite.

Amazon Web Services (AWS) veröffentlicht seine aktuellen IP-Adressbereiche im JSON-Format. Laden Sie die Datei ip-ranges.json herunter, um die aktuellen Bereiche anzuzeigen. Weitere Informationen finden Sie unter AWS IP-Adressbereiche im Allgemeine Amazon Web Services-Referenz.

Um die IP-Adressbereiche zu finden, die CloudFront Edge-Servern zugeordnet sind, suchen Sie ip-ranges.json nach der folgenden Zeichenfolge:

```
"region": "GLOBAL",
"service": "CLOUDFRONT"
```

Alternativ können Sie nur die IP-Bereiche unter anzeigen. CloudFront https://d7uri8nf7uskq.cloudfront.net/tools/list-cloudfront-ips

Verwenden Sie die Liste der CloudFront verwalteten Präfixe

Die Liste der CloudFront verwalteten Präfixe enthält die IP-Adressbereiche aller CloudFront weltweit verteilten Ursprungsserver. Wenn Ihr Ursprung auf einer Amazon VPC-Sicherheitsgruppe gehostet AWS und von dieser geschützt wird, können Sie die Liste der CloudFront verwalteten Präfixe verwenden, um eingehenden Datenverkehr zu Ihrem Ursprung nur von CloudFront Servern zuzulassen, die mit dem Ursprung verbunden sind, sodass kein CloudFront Datenverkehr Ihren Ursprung erreicht. CloudFront verwaltet die Liste der verwalteten Präfixe, sodass sie immer auf dem neuesten Stand mit den IP-Adressen aller globalen Ursprungsserver CloudFront ist. Mit der CloudFront verwalteten Präfixliste müssen Sie keine Liste mit IP-Adressbereichen selbst lesen oder verwalten.

Stellen Sie sich beispielsweise vor, dass Ihr Ursprung eine EC2 Amazon-Instance in der Region Europa (London) ist (`eu-west-2`). Wenn sich die Instance in einer VPC befindet, können Sie eine Sicherheitsgruppenregel erstellen, die eingehenden HTTPS-Zugriff aus der Liste der CloudFront verwalteten Präfixe zulässt. Auf diese Weise können alle globalen Server, CloudFront die mit dem Ursprung verbunden sind, die Instanz erreichen. Wenn Sie alle anderen Regeln für eingehenden Datenverkehr aus der Sicherheitsgruppe entfernen, verhindern Sie, dass kein CloudFront Datenverkehr die Instance erreicht.

Die Liste der CloudFront verwalteten Präfixe trägt den Namen com.amazonaws.global.cloudfront.origin-facing. Weitere Informationen finden Sie unter Verwenden einer AWS-verwalteten Präfixliste im Amazon VPC-Benutzerhandbuch.

> **Important**
>
> Die Liste der CloudFront verwalteten Präfixe ist insofern einzigartig, als sie auf Amazon VPC-Kontingente angewendet wird. Weitere Informationen finden Sie im Abschnitt zur Gewichtung der AWS-verwalteten Präfixliste im Amazon-VPC-Benutzerhandbuch.

Verwendung CloudFront mit einem SDK AWS

AWS Software Development Kits (SDKs) sind für viele gängige Programmiersprachen verfügbar. Jedes SDK bietet eine API, Codebeispiele und Dokumentation, die es Entwicklern erleichtern, Anwendungen in ihrer bevorzugten Sprache zu erstellen.

SDK-Dokumentation	Codebeispiele
AWS SDK für C++	AWS SDK für C++ Codebeispiele
AWS CLI	AWS CLI Code-Beispiele
AWS SDK für Go	AWS SDK für Go Code-Beispiele
AWS SDK für Java	AWS SDK für Java Code-Beispiele
AWS SDK für JavaScript	AWS SDK für JavaScript Code-Beispiele
AWS SDK für Kotlin	AWS SDK für Kotlin Code-Beispiele
AWS SDK für .NET	AWS SDK für .NET Code-Beispiele
AWS SDK für PHP	AWS SDK für PHP Code-Beispiele
AWS -Tools für PowerShell	AWS -Tools für PowerShell Code-Beispiele
AWS SDK für Python (Boto3)	AWS SDK für Python (Boto3) Code-Beispiele
AWS SDK für Ruby	AWS SDK für Ruby Code-Beispiele
AWS SDK für Rust	AWS SDK für Rust Code-Beispiele
AWS SDK für SAP ABAP	AWS SDK für SAP ABAP Code-Beispiele

SDK-Dokumentation	Codebeispiele
AWS SDK für Swift	AWS SDK für Swift Code-Beispiele

> **Beispiel für die Verfügbarkeit**
>
> Sie können nicht finden, was Sie brauchen? Fordern Sie ein Codebeispiel an, indem Sie unten den Link Feedback geben auswählen.

CloudFront technische Ressourcen

Verwenden Sie die folgenden Ressourcen, um Antworten auf technische Fragen zu folgenden Themen zu erhalten CloudFront:

- AWS re:POST — Eine Community-basierte Frage- und Antwortseite, auf der Entwickler technische Fragen zu diesem Thema erörtern können. CloudFront
- Support Center — Auf dieser Website finden Sie Informationen zu Ihren aktuellen Support-Fällen und zu den Ergebnissen von AWS Trusted Advisor Zustandsprüfungen. Sie enthält auch Links zu Diskussionsforen, technischen Foren FAQs, dem Service Health Dashboard und Informationen zu Support Tarifen.
- AWS Premium-Support — Erfahren Sie mehr über AWS Premium Support one-on-one, einen Support-Kanal mit schnellen Reaktionszeiten, der Sie bei der Entwicklung und Ausführung von Anwendungen unterstützt. AWS
- AWS IQ — Holen Sie sich Hilfe von AWS zertifizierten Fachleuten und Experten.

Fangen Sie an mit CloudFront

Die Themen in diesem Abschnitt zeigen Ihnen, wie Sie mit der Bereitstellung Ihrer Inhalte über Amazon beginnen können CloudFront.

In Richten Sie Ihre ein AWS-Konto diesem Thema werden die Voraussetzungen für die folgenden Tutorials beschrieben, z. B. das Erstellen eines Benutzers AWS-Konto und das Erstellen eines Benutzers mit Administratorzugriff.

Das grundlegende Verteilungs-Tutorial zeigt Ihnen, wie Sie Origin Access Control (OAC) einrichten, um authentifizierte Anfragen an einen Amazon S3 S3-Ursprung zu senden.

Das Tutorial zu sicheren statischen Websites zeigt Ihnen, wie Sie mithilfe von OAC mit einem Amazon S3 S3-Ursprung eine sichere statische Website für Ihren Domainnamen erstellen. Das Tutorial verwendet eine Amazon CloudFront (CloudFront) -Vorlage für Konfiguration und Bereitstellung.

Themen

Richten Sie Ihre ein AWS-Konto

In diesem Thema werden vorbereitende Schritte beschrieben, z. B. das Erstellen eines AWS-Konto, um Sie auf die Nutzung von Amazon vorzubereiten CloudFront.

Themen

Melden Sie sich für eine an AWS-Konto

Wenn Sie noch keine haben AWS-Konto, führen Sie die folgenden Schritte aus, um eine zu erstellen.

Um sich für eine anzumelden AWS-Konto

1. Öffnen Sie https://portal.aws.amazon.com/billing/die Anmeldung.
2. Folgen Sie den Online-Anweisungen.

   Ein Teil des Anmeldevorgangs umfasst den Empfang eines Telefonanrufs oder einer Textnachricht und die Eingabe eines Bestätigungscodes auf der Telefontastatur.

   Wenn Sie sich für eine anmelden AWS-Konto, wird eine Root-Benutzer des AWS-Kontoserstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte Sicherheitsmethode weisen Sie einem Administratorbenutzer Administratorzugriff zu und verwenden Sie nur den Root-Benutzer, um Aufgaben auszuführen, die Root-Benutzerzugriff erfordern.

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Du kannst jederzeit deine aktuellen Kontoaktivitäten einsehen und dein Konto verwalten, indem du zu https://aws.amazon.com/gehst und Mein Konto auswählst.

Erstellen eines Benutzers mit Administratorzugriff

Nachdem Sie sich für einen angemeldet haben AWS-Konto, sichern Sie Ihren Root-Benutzer des AWS-Kontos AWS IAM Identity Center, aktivieren und erstellen Sie einen Administratorbenutzer, sodass Sie den Root-Benutzer nicht für alltägliche Aufgaben verwenden.

Sichern Sie Ihre Root-Benutzer des AWS-Kontos

1. Melden Sie sich AWS Management Consoleals Kontoinhaber an, indem Sie Root-Benutzer auswählen und Ihre AWS-Konto E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

   Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter Anmelden als Root-Benutzer im AWS-Anmeldung Benutzerhandbuch zu.
2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer.

   Anweisungen finden Sie unter Aktivieren eines virtuellen MFA-Geräts für Ihren AWS-Konto Root-Benutzer (Konsole) im IAM-Benutzerhandbuch.

Erstellen eines Benutzers mit Administratorzugriff

1. Aktivieren Sie das IAM Identity Center.

   Anweisungen finden Sie unter Aktivieren AWS IAM Identity Center im AWS IAM Identity Center Benutzerhandbuch.

2. Gewähren Sie einem Administratorbenutzer im IAM Identity Center Benutzerzugriff.

   Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie IAM-Identity-Center-Verzeichnis im Benutzerhandbuch unter Benutzerzugriff mit der Standardeinstellung konfigurieren.AWS IAM Identity Center

Anmelden als Administratorbenutzer

- Um sich mit Ihrem IAM-Identity-Center-Benutzer anzumelden, verwenden Sie die Anmelde-URL, die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM-Identity-Center-Benutzer erstellt haben.

  Hilfe bei der Anmeldung mit einem IAM Identity Center-Benutzer finden Sie im AWS-Anmeldung Benutzerhandbuch unter Anmeldung beim AWS Access-Portal.

Weiteren Benutzern Zugriff zuweisen

1. Erstellen Sie im IAM-Identity-Center einen Berechtigungssatz, der den bewährten Vorgehensweisen für die Anwendung von geringsten Berechtigungen folgt.

   Anweisungen hierzu finden Sie unter Berechtigungssatz erstellen im AWS IAM Identity Center Benutzerhandbuch.

2. Weisen Sie Benutzer einer Gruppe zu und weisen Sie der Gruppe dann Single Sign-On-Zugriff zu.

   Eine genaue Anleitung finden Sie unter Gruppen hinzufügen im AWS IAM Identity Center Benutzerhandbuch.

Wählen Sie, wie Sie darauf zugreifen möchten CloudFront

Sie können auf folgende Weise auf Amazon CloudFront zugreifen:

- AWS Management Console— Die Verfahren in diesem Handbuch erläutern, wie Sie mit AWS Management Console dem Aufgaben ausführen können.
- AWS SDKs— Wenn Sie eine Programmiersprache verwenden, die ein SDK für AWS bereitstellt, können Sie ein SDK für den Zugriff verwenden CloudFront. SDKs Vereinfachen Sie die Authentifizierung, lassen Sie sich problemlos in Ihre Entwicklungsumgebung integrieren und bieten Sie Zugriff auf CloudFront Befehle. Weitere Informationen finden Sie unter Verwendung CloudFront mit einem SDK AWS.
- CloudFront API — Wenn Sie eine Programmiersprache verwenden, für die kein SDK verfügbar ist, finden Sie in der Amazon CloudFront API-Referenz Informationen zu API-Aktionen und zum Stellen von API-Anfragen.
- AWS CLI— Das AWS Command Line Interface (AWS CLI) ist ein einheitliches Tool für die Verwaltung AWS-Services. Informationen zur Installation und Konfiguration von finden Sie AWS CLI im AWS Command Line Interface Benutzerhandbuch unter Installation oder Aktualisierung auf die neueste Version von. AWS CLI
- Tools für Windows PowerShell — Wenn Sie Erfahrung mit Windows haben PowerShell, bevorzugen Sie möglicherweise die Verwendung von AWS Tools for Windows PowerShell. Weitere Informationen finden Sie unter Installieren der AWS Tools for Windows PowerShell im AWS -Tools für PowerShell -Benutzerhandbuch.

Beginnen Sie mit einer CloudFront Standarddistribution

Die Verfahren in diesem Abschnitt zeigen Ihnen, wie Sie eine Standarddistribution einrichten, die Folgendes bietet: CloudFront

- Erstellt einen S3-Bucket, der als Verteilungsquelle verwendet werden soll.
- Speichert die Originalversionen Ihrer Objekte in einem Amazon Simple Storage Service (Amazon S3) -Bucket.
- Verwendet Origin Access Control (OAC), um authentifizierte Anfragen an Ihren Amazon S3 S3-Ursprung zu senden. OAC sendet Anfragen durch, CloudFront um zu verhindern, dass Zuschauer direkt auf Ihren S3-Bucket zugreifen. Weitere Informationen zu OAC finden Sie unter. Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung
- Verwendet den CloudFront Domainnamen in URLs für Ihre Objekte (z. B.`https://d111111abcdef8.cloudfront.net/index.html`).
- Bewahrt Ihre Objekte für die Standarddauer von 24 Stunden an CloudFront Randpositionen (die Mindestdauer beträgt 0 Sekunden).

Das meiste davon wird automatisch für Sie konfiguriert, wenn Sie eine CloudFront Verteilung erstellen.

Themen

Voraussetzungen

Bevor Sie beginnen, sollten Sie sicherstellen, dass Sie die in beschriebenen Schritte ausgeführt habe Richten Sie Ihre ein AWS-Konto.

Erstellen eines Amazon-S3-Buckets

Ein Amazon S3 S3-Bucket ist ein Container für Dateien (Objekte) oder Ordner. CloudFront kann fast jeden Dateityp für Sie verteilen, wenn ein S3-Bucket die Quelle ist. CloudFront kann beispielsweise Text, Bilder und Videos verteilen. Die Menge der Daten, die Sie in Amazon S3 speichern können, ist nicht begrenzt.

Für dieses Tutorial erstellen Sie einen S3-Bucket mit den bereitgestellten `hello world` Beispieldateien, die Sie verwenden werden, um eine einfache Webseite zu erstellen.

So erstellen Sie einen Bucket

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wir empfehlen Ihnen, für diese Erste Schritte unser Hello World-Beispiel zu verwenden. Laden Sie die Hello World-Webseite herunter: hello-world-html.zip. Entpacken Sie es und speichern Sie den `css` Ordner und die `index` Datei an einem geeigneten Ort, z. B. auf dem Desktop, auf dem Sie Ihren Browser ausführen.
3. Wählen Sie Create Bucket (Bucket erstellen) aus.

4. Geben Sie einen eindeutigen Bucket-Namen ein, der den Benennungsregeln für allgemeine Buckets im Amazon Simple Storage Service-Benutzerhandbuch entspricht.
5. Als Region empfehlen wir, eine Region zu wählen AWS-Region , die sich geografisch in Ihrer Nähe befindet. (Dies reduziert die Latenz und die Kosten.)
   - Die Auswahl einer anderen Region funktioniert ebenfalls. Sie könnten dies beispielsweise tun, um regulatorische Anforderungen zu erfüllen.
6. Belassen Sie alle anderen Einstellungen auf ihren Standardeinstellungen und wählen Sie dann Bucket erstellen.

Laden Sie den Inhalt in den Bucket hoch

Nachdem Sie Ihren Amazon S3 S3-Bucket erstellt haben, laden Sie den Inhalt der entpackten `hello world` Datei in ihn hoch. (Sie haben diese Datei heruntergeladen und entpackt.) Erstellen eines Amazon-S3-Buckets

So laden Sie den Inhalt in Amazon S3 hoch

1. Wählen Sie im Abschnitt Allgemeine Buckets den Namen Ihres neuen Buckets aus.
2. Klicken Sie auf Upload.
3. Ziehen Sie auf der Upload-Seite den `css` Ordner und die `index` Datei in den Drop-Bereich.
4. Belassen Sie alle anderen Einstellungen auf ihren Standardeinstellungen und wählen Sie dann Hochladen.

Erstellen Sie eine CloudFront Distribution, die einen Amazon S3 S3-Ursprung mit OAC verwendet

In diesem Tutorial erstellen Sie eine CloudFront Distribution, die einen Amazon S3 S3-Ursprung mit Origin Access Control (OAC) verwendet. OAC hilft Ihnen dabei, authentifizierte Anfragen sicher an Ihren Amazon S3 S3-Ursprung zu senden. Weitere Informationen zu OAC finden Sie unter. Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung

Um eine CloudFront Distribution mit einem Amazon S3 S3-Ursprung zu erstellen, der OAC verwendet

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie Verteilung erstellen aus.

3. Geben Sie einen Distributionsnamen für die Standarddistribution ein. Der Name wird als Wert für den `Name` Schlüssel als Tag angezeigt. Sie können diesen Wert später ändern. Sie können bis zu 50 Tags für Ihre Standarddistribution hinzufügen. Weitere Informationen finden Sie unter Kennzeichnen Sie eine Distribution.
4. Wählen Sie Einzelne Website oder App, Weiter.
5. Wählen Sie Weiter aus.
6. Wählen Sie für die Seite vom Typ Origin Amazon S3 aus.
7. Wählen Sie für S3 Origin Browse S3 und dann den S3-Bucket aus, den Sie für dieses Tutorial erstellt haben.
8. Wählen Sie unter Einstellungen die Option Empfohlene Origin-Einstellungen verwenden aus. CloudFront verwendet die empfohlenen Standardeinstellungen für Cache und Origin für Ihren Amazon S3 S3-Ursprung, einschließlich der Einrichtung von Origin Access Control (OAC). Weitere Informationen zu den empfohlenen Einstellungen finden Sie unterReferenz für vorkonfigurierte Verteilungseinstellungen.
9. Wählen Sie Weiter aus.
10. Wählen Sie auf der Seite Sicherheitsvorkehrungen aktivieren aus, ob der AWS WAF Sicherheitsschutz aktiviert werden soll.
11. Wählen Sie Weiter aus.
12. Wählen Sie Verteilung erstellen aus. CloudFront aktualisiert die S3-Bucket-Richtlinie für Sie.
13. Sehen Sie sich den Abschnitt „Details“ für Ihre neue Distribution an. Wenn Ihre Distribution mit der Bereitstellung fertig ist, ändert sich das Feld Letzte Änderung von Bereitstellen in ein Datum und eine Uhrzeit.
14. Notieren Sie sich den Domainnamen, der Ihrer Distribution CloudFront zugewiesen wurde. Er sieht in etwa wie folgt aus: `d111111abcdef8.cloudfront.net`.

Bevor Sie die Distribution und den S3-Bucket aus diesem Tutorial in einer Produktionsumgebung verwenden, stellen Sie sicher, dass sie Ihren spezifischen Anforderungen entsprechen. Informationen zur Konfiguration des Zugriffs in einer Produktionsumgebung finden Sie unterKonfigurieren Sie den sicheren Zugriff und beschränken Sie den Zugriff auf Inhalte.

Greifen Sie auf Ihre Inhalte zu über CloudFront

Um über auf Ihre Inhalte zuzugreifen CloudFront, kombinieren Sie den Domainnamen für Ihre CloudFront Distribution mit der Hauptseite für Ihre Inhalte. (Sie haben Ihren Vertriebs-Domainnamen

in aufgezeichnetErstellen Sie eine CloudFront Distribution, die einen Amazon S3 S3-Ursprung mit OAC verwendet.)

- Ihr Vertriebsdomänenname könnte so aussehen: `d111111abcdef8.cloudfront.net`.
- Der Pfad zur Hauptseite einer Website ist in der Regel `/index.html`.

Daher CloudFront könnte die URL, über die Sie auf Ihre Inhalte zugreifen, wie folgt aussehen:

`https://d111111abcdef8.cloudfront.net/index.html`.

Wenn Sie die vorherigen Schritte befolgt und die Hello World-Webseite verwendet haben, sollten Sie eine Webseite mit der Aufschrift Hello World! sehen. .

Wenn Sie weitere Inhalte in diesen S3-Bucket hochladen, können Sie auf die Inhalte zugreifen, CloudFront indem Sie den Namen der CloudFront Distributionsdomain mit dem Pfad zum Objekt im S3-Bucket kombinieren. Wenn Sie eine neue Datei mit dem Namen `new-page.html` zum Stammverzeichnis Ihres S3-Buckets hochladen, sieht die URL beispielsweise wie folgt aus:

`https://d111111abcdef8.cloudfront.net/new-page.html`.

Bereinigen

Wenn Sie Ihre Distribution und Ihren S3-Bucket nur zu Lernzwecken erstellt haben, löschen Sie sie, damit keine Gebühren mehr anfallen. Löschen Sie zuerst die Verteilung. Weitere Informationen finden Sie unter den folgenden Links:

- Löschen einer -Verteilung
- Einen Bucket löschen

Verbessern Sie Ihre Basisdistribution

Dieses Tutorial „Erste Schritte“ bietet ein minimales Framework für die Erstellung einer Distribution. Wir empfehlen Ihnen, sich mit den folgenden Verbesserungen vertraut zu machen:

- Sie können die Funktion für CloudFront private Inhalte verwenden, um den Zugriff auf die Inhalte in den Amazon S3 S3-Buckets einzuschränken. Weitere Informationen zum Verteilen von privaten Inhalten finden Sie unter Stellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit.

- Sie können Ihre CloudFront Distribution so konfigurieren, dass sie einen benutzerdefinierten Domainnamen verwendet (z. B. `www.example.com` anstelle von`d111111abcdef8.cloudfront.net`). Weitere Informationen finden Sie unter Benutzerdefiniert verwenden URLs.
- Dieses Tutorial verwendet einen Amazon S3 S3-Ursprung mit Origin Access Control (OAC). Sie können OAC jedoch nicht verwenden, wenn Ihr Ursprung ein S3-Bucket ist, der als Website-Endpunkt konfiguriert ist. In diesem Fall müssen Sie Ihren Bucket CloudFront als benutzerdefinierten Ursprung einrichten. Weitere Informationen finden Sie unter Verwenden Sie einen Amazon S3 S3-Bucket, der als Website-Endpunkt konfiguriert ist. Weitere Informationen zu OAC finden Sie unterBeschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung.

Beginnen Sie mit einer Standarddistribution (AWS CLI)

Die Verfahren in diesem Abschnitt zeigen Ihnen, wie Sie mithilfe von AWS CLI with CloudFront eine grundlegende Konfiguration einrichten, die Folgendes umfasst:

- Erstellen eines Amazon S3 S3-Buckets, der als Vertriebsursprung verwendet werden soll.
- Speichern der Originalversionen Ihrer Objekte im S3-Bucket.
- Verwenden Sie Origin Access Control (OAC), um authentifizierte Anfragen an Ihren Amazon S3 S3-Ursprung zu senden. OAC sendet Anfragen durch, CloudFront um zu verhindern, dass Zuschauer direkt auf Ihren S3-Bucket zugreifen. Weitere Informationen zu OAC finden Sie unter. Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung
- Verwenden Sie den CloudFront Domainnamen in URLs für Ihre Objekte (z. B.`https://d111111abcdef8.cloudfront.net/index.html`).
- Halten Sie Ihre Objekte für die Standarddauer von 24 Stunden an CloudFront Randpositionen (die Mindestdauer beträgt 0 Sekunden).

Die meisten dieser Optionen sind anpassbar. Weitere Informationen dazu, wie Sie Ihre CloudFront - Verteilungsoptionen anpassen, finden Sie unter Eine Verteilung erstellen.

Voraussetzungen

Bevor Sie beginnen, sollten Sie sicherstellen, dass Sie die in beschriebenen Schritte ausgeführt habe Richten Sie Ihre ein AWS-Konto.

Installieren Sie das AWS CLI und konfigurieren Sie es mit Ihren Anmeldeinformationen. Weitere Informationen finden Sie unter Erste Schritte mit dem AWS CLI im AWS CLI -Benutzerhandbuch.

Erstellen eines Amazon-S3-Buckets

Ein Amazon S3 S3-Bucket ist ein Container für Dateien (Objekte) oder Ordner. CloudFront kann fast jeden Dateityp für Sie verteilen, wenn ein S3-Bucket die Quelle ist. CloudFront kann beispielsweise Text, Bilder und Videos verteilen. Die Menge der Daten, die Sie in Amazon S3 speichern können, ist nicht begrenzt.

Für dieses Tutorial erstellen Sie einen S3-Bucket und laden eine HTML-Datei hoch, mit der Sie eine einfache Webseite erstellen werden.

```
aws s3 mb s3://amzn-s3-demo-bucket/ --region us-east-1
```

*amzn-s3-demo-bucket*Ersetzen Sie es durch einen weltweit eindeutigen Bucket-Namen. Für den empfehlen wir AWS-Region, eine Region zu wählen, die sich geografisch in Ihrer Nähe befindet. Dies reduziert die Latenz und die Kosten, aber die Auswahl einer anderen Region funktioniert auch. Sie könnten dies beispielsweise tun, um regulatorische Anforderungen zu erfüllen.

Laden Sie den Inhalt in den Bucket hoch

Laden Sie für dieses Tutorial die Beispielinhaltsdateien für eine einfache „Hello World“ -Webseite herunter und extrahieren Sie sie.

```
# Create a temporary directory
mkdir -p ~/cloudfront-demo

# Download the sample Hello World files
curl -o ~/cloudfront-demo/hello-world-html.zip https://docs.aws.amazon.com/
AmazonCloudFront/latest/DeveloperGuide/samples/hello-world-html.zip

# Extract the zip file
unzip ~/cloudfront-demo/hello-world-html.zip -d ~/cloudfront-demo/hello-world
```

Dadurch wird ein Verzeichnis mit einer `index.html` Datei und einem `css` Ordner erstellt. Laden Sie diese Dateien in Ihren S3-Bucket hoch.

```
aws s3 cp ~/cloudfront-demo/hello-world/ s3://amzn-s3-demo-bucket/ --recursive>
```

Erstelle eine Origin Access Control (OAC)

In diesem Tutorial erstellen Sie eine Origin Access Control (OAC). OAC hilft Ihnen dabei, authentifizierte Anfragen sicher an Ihren Amazon S3 S3-Ursprung zu senden. Weitere Informationen zu OAC finden Sie unter. Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung

```
aws cloudfront create-origin-access-control \
    --origin-access-control-config Name="oac-for-
s3",SigningProtocol=sigv4,SigningBehavior=always,OriginAccessControlOriginType=s3
```

Speichern Sie die OAC-ID aus der Ausgabe als Umgebungsvariable. Ersetzen Sie den Beispielwert durch Ihre eigene OAC-ID. Sie werden dies im nächsten Schritt verwenden.

```
OAC_ID="E1ABCD2EFGHIJ"
```

Erstellen Sie eine Standarddistribution

Erstellen Sie eine Distributionskonfigurationsdatei mit dem Namen`distribution-config.json`. Ersetzen Sie den Beispiel-Bucket-Namen durch Ihren Bucket-Namen für die `TargetOriginId` Werte `IdDomainName`, und.

```
cat > distribution-config.json << EOF
{
    "CallerReference": "cli-example-$(date +%s)",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
                "Id": "S3-amzn-s3-demo-bucket",
                "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                },
                "OriginAccessControlId": "$OAC_ID"
            }
        ]
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "S3-amzn-s3-demo-bucket",
        "ViewerProtocolPolicy": "redirect-to-https",
```

```
        "AllowedMethods": {
            "Quantity": 2,
            "Items": ["GET", "HEAD"],
            "CachedMethods": {
                "Quantity": 2,
                "Items": ["GET", "HEAD"]
            }
        },
        "DefaultTTL": 86400,
        "MinTTL": 0,
        "MaxTTL": 31536000,
        "Compress": true,
        "ForwardedValues": {
            "QueryString": false,
            "Cookies": {
                "Forward": "none"
            }
        }
    },
    "Comment": "CloudFront distribution for S3 bucket",
    "Enabled": true
}
EOF
```

Erstellen Sie die Standardverteilung.

```
aws cloudfront create-distribution --distribution-config file://distribution-
config.json
```

Speichern Sie die Distributions-ID und den Domainnamen aus der Ausgabe als Umgebungsvariablen. Ersetzen Sie die Beispielwerte durch Ihre eigenen Werte. Sie werden diese später in diesem Tutorial verwenden.

```
DISTRIBUTION_ID="EABCD1234XMPL"
DOMAIN_NAME="d111111abcdef8.cloudfront.net"
```

Bevor Sie die Distribution und den S3-Bucket aus diesem Tutorial in einer Produktionsumgebung verwenden, stellen Sie sicher, dass sie Ihren spezifischen Anforderungen entsprechen. Informationen zur Konfiguration des Zugriffs in einer Produktionsumgebung finden Sie unterKonfigurieren Sie den sicheren Zugriff und beschränken Sie den Zugriff auf Inhalte.

Aktualisieren Sie Ihre S3-Bucket-Richtlinie

Aktualisieren Sie Ihre S3-Bucket-Richtlinie, um den Zugriff auf die Objekte CloudFront zu ermöglichen. Ersetzen Sie den Beispiel-Bucket-Namen durch Ihren Bucket-Namen.

```
# Get your AWS account ID
ACCOUNT_ID=$(aws sts get-caller-identity --query 'Account' --output text)

# Create the bucket policy
cat > bucket-policy.json << EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCloudFrontServicePrincipal",
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudfront.amazonaws.com"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
            "Condition": {
                "StringEquals": {
                    "AWS:SourceArn": "arn:aws:cloudfront::$ACCOUNT_ID:distribution/
$DISTRIBUTION_ID"
                }
            }
        }
    ]
}
EOF

# Apply the bucket policy
aws s3api put-bucket-policy \
    --bucket amzn-s3-demo-bucket \
    --policy file://bucket-policy.json
```

Bestätigen Sie die Bereitstellung der Distribution

Nachdem Sie Ihre Distribution erstellt haben, wird es einige Zeit dauern, bis die Bereitstellung abgeschlossen ist. Wenn sich der Verteilungsstatus von `InProgress` zu ändert`Deployed`, fahren Sie mit dem nächsten Schritt fort.

```
aws cloudfront get-distribution --id $DISTRIBUTION_ID --query 'Distribution.Status'
```

Alternativ können Sie den `wait` Befehl verwenden, um auf die Bereitstellung der Verteilung zu warten.

```
aws cloudfront wait distribution-deployed --id $DISTRIBUTION_ID
```

Greifen Sie auf Ihre Inhalte zu über CloudFront

Um über auf Ihre Inhalte zuzugreifen CloudFront, kombinieren Sie den Domainnamen für Ihre CloudFront Distribution mit der Hauptseite für Ihre Inhalte. Ersetzen Sie den CloudFront Beispiel-Domainnamen durch Ihren eigenen.

```
https://d111111abcdef8.cloudfront.net/index.html
```

Wenn Sie die vorherigen Schritte befolgt und die HTML-Datei erstellt haben, sollten Sie eine Webseite mit der Aufschrift Hello world! sehen. .

Wenn Sie weitere Inhalte in diesen S3-Bucket hochladen, können Sie auf die Inhalte zugreifen, CloudFront indem Sie den Namen der CloudFront Distributionsdomain mit dem Pfad zum Objekt im S3-Bucket kombinieren. Wenn Sie eine neue Datei mit dem Namen `new-page.html` zum Stammverzeichnis Ihres S3-Buckets hochladen, sieht die URL beispielsweise wie folgt aus:

`https://d111111abcdef8.cloudfront.net/new-page.html`.

Bereinigen

Wenn Sie Ihre Distribution und Ihren S3-Bucket nur zu Lernzwecken erstellt haben, löschen Sie sie, damit keine Gebühren mehr anfallen. Deaktivieren und löschen Sie zuerst die Distribution.

Um eine Standarddistribution zu deaktivieren und zu löschen (AWS CLI)

1. Deaktivieren Sie zunächst die Distribution.

   ```
   # Get the current configuration and ETag
   ETAG=$(aws cloudfront get-distribution-config --id $DISTRIBUTION_ID --query 'ETag'
    --output text)
   ```

```
# Create a modified configuration with Enabled=false
aws cloudfront get-distribution-config --id $DISTRIBUTION_ID | \
jq '.DistributionConfig.Enabled = false' > temp_disabled_config.json

# Update the distribution to disable it
aws cloudfront update-distribution \
    --id $DISTRIBUTION_ID \
    --distribution-config file://<(jq '.DistributionConfig'
 temp_disabled_config.json) \
    --if-match $ETAG
```

2. Warten Sie, bis die Distribution deaktiviert ist.

```
aws cloudfront wait distribution-deployed --id $DISTRIBUTION_ID
```

3. Löschen Sie die Distribution.

```
# Get the current ETag
ETAG=$(aws cloudfront get-distribution-config --id $DISTRIBUTION_ID --query 'ETag'
 --output text)

# Delete the distribution
aws cloudfront delete-distribution --id $DISTRIBUTION_ID --if-match $ETAG
```

Um einen S3-Bucket zu löschen (AWS CLI)

- Löschen Sie den S3-Bucket und seinen Inhalt. Ersetzen Sie den Beispiel-Bucket-Namen durch Ihren eigenen.

```
# Delete the bucket contents
aws s3 rm s3://amzn-s3-demo-bucket --recursive

# Delete the bucket
aws s3 rb s3://amzn-s3-demo-bucket
```

Führen Sie die folgenden Befehle aus, um die für dieses Tutorial erstellten lokalen Dateien zu bereinigen:

```
# Clean up local files
rm -f distribution-config.json bucket-policy.json temp_disabled_config.json
```

```
rm -rf ~/cloudfront-demo
```

Optional können Sie das OAC löschen, das Sie für dieses Tutorial erstellt haben.

```
# Get the OAC ETag
OAC_ETAG=$(aws cloudfront get-origin-access-control --id $OAC_ID --query 'ETag' --
output text)

# Delete the OAC
aws cloudfront delete-origin-access-control --id $OAC_ID --if-match $OAC_ETAG
```

Beginnen Sie mit einer sicheren statischen Website

Sie können mit Amazon beginnen, CloudFront indem Sie die in diesem Thema beschriebene Lösung verwenden, um eine sichere statische Website für Ihren Domainnamen zu erstellen. Eine statische Website verwendet nur statische Dateien wie HTML, CSS JavaScript, Bilder und Videos und benötigt keine Server oder serverseitige Verarbeitung. Mit dieser Lösung erhält Ihre Website folgende Vorteile:

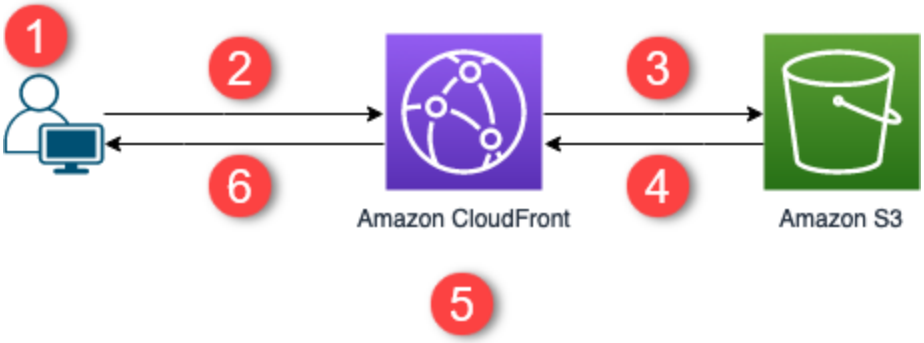
- Verwendet den dauerhaften Speicher von Amazon Simple Storage Service (Amazon S3) – Mit dieser Lösung wird ein Amazon S3-Bucket erstellt, um den Inhalt Ihrer statischen Website zu hosten. Um Ihre Website zu aktualisieren, laden Sie einfach Ihre neuen Dateien in den S3-Bucket hoch.
- Wird durch das Amazon CloudFront Content Delivery Network beschleunigt — Diese Lösung erstellt eine CloudFront Distribution, um Ihre Website Zuschauern mit geringer Latenz zur Verfügung zu stellen. Die Verteilung ist mit Origin Access Control (OAC) konfiguriert, um sicherzustellen, dass auf die Website nur über CloudFront S3 und nicht direkt von S3 aus zugegriffen werden kann.
- Ist durch HTTPS und Sicherheitsheader gesichert — Diese Lösung erstellt ein SSL/TLS-Zertifikat in AWS Certificate Manager (ACM) und hängt es an die Distribution an. CloudFront Dieses Zertifikat ermöglicht es der Verteilung, die Website Ihrer Domäne sicher mit HTTPS zu bedienen.
- Ist konfiguriert und bereitgestellt mit AWS CloudFormation— Diese Lösung verwendet eine AWS CloudFormation Vorlage zur Einrichtung aller Komponenten, sodass Sie sich mehr auf den Inhalt Ihrer Website und weniger auf die Konfiguration der Komponenten konzentrieren können.

Diese Lösung ist Open Source auf GitHub. Um den Code anzuzeigen, eine Pull-Anforderung einzureichen oder ein Problemticket zu öffnen, gehen Sie zu https://github.com/aws-samples/amazon-cloudfront-secure-static-site.

Themen

Übersicht über die Lösung

Das folgende Diagramm zeigt einen Überblick über die Funktionsweise dieser statischen Website-Lösung:



1. Der Betrachter fordert die Website unter www.example.com an.
2. Wenn das angeforderte Objekt zwischengespeichert ist, wird das Objekt aus seinem Cache an den Viewer CloudFront zurückgegeben.
3. Wenn sich das Objekt nicht im CloudFront Cache befindet, CloudFront wird das Objekt vom Ursprung (einem S3-Bucket) angefordert.
4. S3 gibt das Objekt an zurück CloudFront.
5. CloudFront speichert das Objekt im Cache.
6. Das Objekt wird an den Viewer zurückgegeben. Nachfolgende Anfragen für das Objekt, die an derselben CloudFront Edge-Position ankommen, werden vom CloudFront Cache aus bedient.

Stellen Sie die Lösung bereit

Um diese sichere statische Website-Lösung bereitzustellen, können Sie eine der folgenden Optionen auswählen:

- Verwenden Sie die AWS CloudFormation Konsole, um die Lösung mit Standardinhalten bereitzustellen, und laden Sie dann den Inhalt Ihrer Website auf Amazon S3 hoch.

- Klonen Sie die Lösung auf Ihren Computer, um Ihre Website-Inhalte hinzuzufügen. Stellen Sie dann die Lösung mit AWS Command Line Interface (AWS CLI) bereit.

> **Note**
>
> Sie müssen die Region USA Ost (Nord-Virginia) verwenden, um die CloudFormation Vorlage bereitzustellen.

Themen

Voraussetzungen

Um diese Lösung verwenden zu können, müssen Sie die folgenden Voraussetzungen haben:

- Ein registrierter Domänenname, z. B. example.com, der auf eine von Amazon Route 53 gehostete Zone verweist. Die Hosting-Zone muss sich in derselben Zone befinden AWS-Konto , in der Sie diese Lösung bereitstellen. Wenn Sie keinen registrierten Domänennamen haben, können Sie einen bei Route 53 registrieren. Wenn Sie einen registrierten Domänennamen haben, der aber nicht auf eine von Route 53 gehostete Zone verweist, konfigurieren Sie Route 53 als Ihren DNS-Service.
- AWS Identity and Access Management (IAM) Berechtigungen zum Starten von CloudFormation Vorlagen, die IAM-Rollen erstellen, und Berechtigungen zum Erstellen aller AWS Ressourcen in der Lösung. Weitere Informationen finden Sie unter Steuern des Zugriffs mit AWS Identity and Access Management im AWS CloudFormation Benutzerhandbuch.

Sie sind für die Kosten verantwortlich, die bei der Nutzung dieser Lösung entstehen. Weitere Informationen zu den Kosten finden Sie auf den jeweiligen Preisseiten AWS-Service.

Verwendung der AWS CloudFormation -Konsole

Zur Bereitstellung über die CloudFormation Konsole

1. Starten Sie diese Lösung in der AWS CloudFormation Konsole. Melden Sie sich bei Bedarf bei Ihrem an AWS-Konto.
2. Der Assistent zum Erstellen von Stacks wird in der CloudFormation Konsole geöffnet. Er enthält vorausgefüllte Felder, die die CloudFormation Vorlage dieser Lösung angeben.

   Wählen Sie unten auf der Seite Next aus.
3. Geben Sie auf der Seite Specify stack details (Stackdetails angeben) Werte für die folgenden Felder ein:

   - SubDomain— Geben Sie die Subdomain ein, die Sie für Ihre Website verwenden möchten. Wenn die Subdomain beispielsweise www ist, ist Ihre Website verfügbar unter *`www.example.com.`* (Ersetzen Sie example.com durch Ihren Domainnamen, wie im folgenden Punkt bullet).
   - DomainName— Geben Sie Ihren Domainnamen ein, z. B. *`example.com`* Diese Domäne muss auf eine von Route 53 gehostete Zone verweisen.
   - HostedZoneId— Die Route 53-Hosting-Zonen-ID Ihres Domainnamens.
   - CreateApex— (Optional) Erstellen Sie in Ihrer CloudFront Konfiguration einen Alias für die Domain Apex (example.com).
4. Wenn Sie fertig sind, wählen Sie Next (Weiter).
5. (Optional) Auf der Seite Configure stack options (Stack-Optionen konfigurieren) können Sie Tags und andere Stack-Optionen hinzufügen.
6. Wenn Sie fertig sind, wählen Sie Next (Weiter).
7. Scrollen Sie auf der Seite Review (Überprüfen) zum Ende der Seite und wählen Sie dann die beiden Felder im Abschnitt Capabilities (Funktionen) aus. Diese Funktionen ermöglichen es CloudFormation , eine IAM-Rolle zu erstellen, die den Zugriff auf die Ressourcen des Stacks ermöglicht, und die Ressourcen dynamisch zu benennen.
8. Wählen Sie Create stack (Stack erstellen) aus.
9. Warten Sie, bis der Stack erstellt wurde. Der Stack erstellt einige verschachtelte Stacks. Dieser Vorgang kann einige Minuten dauern. Wenn der Vorgang fertig ist, wechselt der Status zu CREATE_COMPLETE.

Wenn der Status CREATE_COMPLETE lautet, wechseln Sie zu https://, *www.example.com* um Ihre Website aufzurufen (ersetzen Sie www.example.com durch die Subdomain und den Domainnamen, die Sie in Schritt 3 angegeben haben). Sie sollten den Standardinhalt der Website sehen:

So ersetzen Sie den Standardinhalt der Website durch Ihren eigenen

1. Öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie den Bucket aus, dessen Name mit amazon-cloudfront-secure-static-site-s3bucketroot - beginnt.

   > **Note**
   >
   > Stellen Sie sicher, dass Sie den Bucket mit s3bucketroot im Namen auswählen, nicht s3bucketlogs. Der Bucket mit s3bucketroot im Namen enthält den Inhalt der Website. Der Bucket mit s3bucketlogs im Namen enthält nur Protokolldateien.

3. Löschen Sie den Standardinhalt der Website, und laden Sie dann Ihren eigenen hoch.

   > **Note**
   >
   > Wenn Sie Ihre Website mit dem Standardinhalt dieser Lösung angesehen haben, ist es wahrscheinlich, dass ein Teil des Standardinhalts an einem Edge-Speicherort zwischengespeichert wurde. CloudFront Um sicherzustellen, dass Zuschauer Ihre aktualisierten Website-Inhalte sehen, machen Sie die Dateien ungültig, um die zwischengespeicherten Kopien von CloudFront den Edge-Speicherorten zu entfernen. Weitere Informationen finden Sie unter Machen Sie Dateien ungültig, um Inhalte zu entfernen.

Klonen Sie die Lösung lokal

Voraussetzungen

Um Ihren Website-Inhalt hinzuzufügen, bevor Sie diese Lösung bereitstellen, müssen Sie die Artefakte der Lösung lokal verpacken, wofür Node.js und npm erforderlich sind. Weitere Informationen finden Sie unter https://www.npmjs.com/get-npm.

So fügen Sie Ihre Website-Inhalte hinzu und stellen die Lösung bereit

1. Klonen oder laden Sie die Lösung von herunte https://github.com/aws-samples/amazon-cloudfront-secure-static-site. Öffnen Sie nach dem Klonen oder Herunterladen eine Eingabeaufforderung oder ein Terminal, und navigieren Sie zum Ordner `amazon-cloudfront-secure-static-site`.
2. Führen Sie den folgenden Befehl aus, um die Artefakte der Lösung zu installieren und zu verpacken:

   ```
   make package-static
   ```

3. Kopieren Sie den Inhalt Ihrer Website in den Ordner `www` und überschreiben Sie den Standard-Website-Inhalt.
4. Führen Sie den folgenden AWS CLI Befehl aus, um einen Amazon S3 S3-Bucket zum Speichern der Lösungsartefakte zu erstellen. *amzn-s3-demo-bucket-for-artifacts*Ersetzen Sie es durch Ihren eigenen Bucket-Namen.

   ```
   aws s3 mb s3://amzn-s3-demo-bucket-for-artifacts --region us-east-1
   ```

5. Führen Sie den folgenden AWS CLI Befehl aus, um die Artefakte der Lösung als CloudFormation Vorlage zu verpacken. *amzn-s3-demo-bucket-for-artifacts*Ersetzen Sie es durch den Namen des Buckets, den Sie im vorherigen Schritt erstellt haben.

   ```
   aws cloudformation package \
       --region us-east-1 \
       --template-file templates/main.yaml \
       --s3-bucket amzn-s3-demo-bucket-for-artifacts \
       --output-template-file packaged.template
   ```

6. Führen Sie den folgenden Befehl aus, um die Lösung bereitzustellen CloudFormation, und ersetzen Sie dabei die folgenden Werte:

- *your-CloudFormation-stack-name*— Durch einen Namen für den CloudFormation Stack ersetzen.
- *example.com*— Ersetze es durch deinen Domainnamen. Diese Domain muss auf eine von Route 53 gehostete Zone in derselben verweisen AWS-Konto.
- *www*— Ersetzen Sie es durch die Subdomain, die Sie für Ihre Website verwenden möchten. Wenn die Subdomäne beispielsweise www ist, ist Ihre Website unter www.example.com verfügbar.
- *hosted-zone-ID*— Ersetzen Sie es durch die Route 53-Hosting-Zonen-ID Ihres Domainnamens.

```
aws cloudformation deploy \
    --region us-east-1 \
    --stack-name your-CloudFormation-stack-name \
    --template-file packaged.template \
    --capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND \
    --parameter-overrides DomainName=example.com SubDomain=www HostedZoneId=hosted-
zone-ID
```

- (Optional) Um den Stack mit einem Domain-Apex bereitzustellen, führen Sie stattdessen den folgenden Befehl aus.

```
aws --region us-east-1 cloudformation deploy \
    --stack-name your-CloudFormation-stack-name \
    --template-file packaged.template \
    --capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND \
    --parameter-overrides  DomainName=example.com SubDomain=www
 HostedZoneId=hosted-zone-ID CreateApex=yes
```

7. Warten Sie, bis die Erstellung des CloudFormation Stacks abgeschlossen ist. Der Stack erstellt einige verschachtelte Stacks. Dieser Vorgang kann einige Minuten dauern. Wenn der Vorgang fertig ist, wechselt der Status zu CREATE_COMPLETE.

   Wenn sich der Status in CREATE_COMPLETE ändert, gehen Sie zu, https://www.example.com um Ihre Website aufzurufen (ersetzen Sie www.example.com durch die Subdomain und den Domainnamen, die Sie im vorherigen Schritt angegeben haben). Sie sollten den Inhalt Ihrer Website sehen.

Suchen von Zugriffsprotokollen

Diese Lösung ermöglicht Zugriffsprotokolle für die CloudFront-Verteilung. Führen Sie die folgenden Schritte aus, um die Zugriffsprotokolle der Verteilung zu finden.

So finden Sie die Zugriffsprotokolle der Verteilung

1. Öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie den Bucket aus, dessen Name mit amazon-cloudfront-secure-static-site-s3bucketlogs - beginnt.

   > **Note**
   >
   > Stellen Sie sicher, dass Sie den Bucket mit s3bucketlogs im Namen auswählen, nicht s3bucketroot. Der Bucket mit s3bucketlogs im Namen enthält Protokolldateien. Der Bucket mit s3bucketroot im Namen enthält den Inhalt der Website.

3. Der Ordner mit dem Namen cdn enthält die Zugriffsprotokolle. CloudFront

Distributionen konfigurieren

Sie erstellen eine CloudFront Amazon-Distribution, um mitzuteilen, CloudFront von wo aus Inhalte geliefert werden sollen, und um zu erfahren, wie Sie die Inhaltszustellung verfolgen und verwalten können.

Wenn Sie Ihre CloudFront Distribution erstellen, konfiguriert die meisten Vertriebseinstellungen CloudFront automatisch für Sie, basierend auf der Herkunft Ihrer Inhalte. Weitere Informationen zu den vorkonfigurierten Einstellungen finden Sie unter. Referenz für vorkonfigurierte Verteilungseinstellungen Optional können Sie wählen, ob Sie Ihre Verteilungseinstellungen manuell bearbeiten möchten. Weitere Informationen finden Sie unter Referenz für alle Verteilungseinstellungen.

Die folgenden Einstellungen können konfiguriert werden:

- Ihr Inhaltsursprung — Der Amazon S3 S3-Bucket, der AWS Elemental MediaPackage Kanal, der AWS Elemental MediaStore Container, der Elastic Load Balancing Balancing-Load Balancer oder der HTTP-Server, von dem die CloudFront zu verteilenden Dateien abgerufen werden. Sie können eine beliebige Kombination von bis zu 25 Ursprüngen für eine einzelne Verteilung angeben.
- Zugriff – Ob Sie möchten, dass die Dateien öffentlich verfügbar sind, oder ob Sie den Zugriff auf bestimmte Benutzer beschränken möchten.
- Sicherheit – Ob Sie den AWS WAF -Schutz aktivieren und möchten, dass Benutzer für den Zugriff auf Ihre Inhalte HTTPS verwenden müssen. Für Multi-Tenant-Distributionen werden nur AWS WAF V2-Web-Zugriffskontrolllisten (ACLs) unterstützt.
- Cache-Schlüssel – Welche Werte, wenn überhaupt, in den Cache-Schlüssel eingefügt werden sollen. Der Cache-Schlüssel identifiziert eindeutig jede Datei im Cache für eine bestimmte Verteilung.
- Einstellungen für Origin-Anfragen — Gibt CloudFront an, ob Sie HTTP-Header, Cookies oder Abfragezeichenfolgen in Anfragen einbeziehen möchten, die an Ihren Ursprung gesendet werden.
- Geografische Einschränkungen — Gibt an, ob Sie verhindern CloudFront möchten, dass Nutzer in ausgewählten Ländern auf Ihre Inhalte zugreifen.
- Protokolle — Ob Sie Standardprotokolle oder Echtzeitprotokolle erstellen CloudFront möchten, die die Zuschaueraktivitäten aufzeigen.

Weitere Informationen finden Sie unter Referenz für alle Verteilungseinstellungen.

Die aktuelle maximale Anzahl von Verteilungen, die Sie für jede Verteilung erstellen können AWS-Konto, finden Sie unter. Allgemeine Kontingente für Verteilungen Die Anzahl von Dateien, die Sie pro Verteilung bereitstellen können, ist nicht begrenzt.

Sie können Verteilungen verwenden, um die folgenden Inhalte über HTTP oder HTTPS bereitzustellen:

- Statische und dynamische Inhalte wie HTML- JavaScript, CSS- und Bilddateien werden über HTTP oder HTTPS heruntergeladen.
- Video-on-Demand in verschiedenen Formaten, wie Apple HTTP Live Streaming (HLS) und Microsoft Smooth Streaming. (Bei Distributionen mit mehreren Mandanten wird Smooth Streaming nicht unterstützt.) Weitere Informationen finden Sie unter Stellen Sie Video-on-Demand bereit mit CloudFront.
- Live-Events, z. B. ein Meeting, eine Konferenz oder ein Konzert – in Echtzeit. Für Live-Streaming können Sie die Verteilung mithilfe eines AWS CloudFormation Stacks automatisch erstellen. Weitere Informationen finden Sie unter Stellen Sie Videostreaming mit CloudFront und AWS Media Services bereit.

In den folgenden Themen finden Sie weitere Informationen zu CloudFront Distributionen und dazu, wie Sie sie entsprechend Ihren Geschäftsanforderungen konfigurieren können. Weitere Informationen zum Erstellen einer Verteilung finden Sie unter Eine Verteilung erstellen.

Themen

- Verwenden Sie Benutzerdefiniert, URLs indem Sie alternative Domainnamen hinzufügen (CNAMEs)
- WebSockets Mit CloudFront Distributionen verwenden
- Fordere Anycast static an, um es für die Zulassungsliste zu verwenden IPs
- gRPC mit CloudFront Distributionen verwenden

Erfahren Sie, wie Distributionen mit mehreren Mandanten funktionieren

Sie können CloudFront Mehrmandantenverteilungen mit Einstellungen erstellen, die für mehrere Verteilungsmandanten wiederverwendet werden können. Bei einer Verteilung mit mehreren Mandanten können Sie Ihre Verteilungseinstellungen auf der Grundlage Ihres Inhaltstyps CloudFront konfigurieren lassen. Weitere Informationen zu den vorkonfigurierten Einstellungen finden Sie unter. Referenz für vorkonfigurierte Verteilungseinstellungen

Die Verwendung einer Multi-Tenant-Distribution anstelle einer Standarddistribution bietet unter anderem folgende Vorteile:

- Verringerung der betrieblichen Belastung.
- Wiederverwendbare Konfigurationen für Webadministratoren und Softwareanbieter zur Verwaltung der CloudFront Verteilung mehrerer Webanwendungen, die Inhalte für Endbenutzer bereitstellen.
- Verbesserte Integrationen mit anderen AWS-Services , um automatisiertes Zertifikatsmanagement, einheitliche Sicherheitskontrollen und problemlose Konfigurationskontrolle im großen Maßstab zu ermöglichen.
- Aufrechterhaltung konsistenter Ressourcenmuster in Ihren Implementierungen. Definieren Sie Einstellungen, die gemeinsam genutzt werden müssen, und geben Sie dann Anpassungen an, die außer Kraft gesetzt werden sollen.
- Individuell anpassbare Herkunfts- und Sicherheitseinstellungen, um spezifische Anforderungen auf der Ebene des Distributionsmandanten zu erfüllen.
- Ordnen Sie Ihre Distributionsmandanten in verschiedene Stufen ein. Wenn beispielsweise einige Distributionsmandanten Origin Shield benötigen und andere nicht, können Sie Verteilungsmandanten in verschiedene Mehrmandantenverteilungen gruppieren.
- Gemeinsame Nutzung einer gemeinsamen DNS-Konfiguration für mehrere Domains.

Im Gegensatz zu einer Standardverteilung kann auf eine Mehrmandantenverteilung nicht direkt zugegriffen werden, da sie keinen CloudFront Routing-Endpunkt hat. Daher muss sie in Verbindung mit einer Verbindungsgruppe und einem oder mehreren Verteilungsmandanten verwendet werden. Standardverteilungen haben zwar ihren eigenen CloudFront Endpunkt und können von Endbenutzern direkt aufgerufen werden, sie können jedoch nicht als Vorlage für andere Distributionen verwendet werden.

Informationen zu Verteilungsquoten für mehrere Mandanten finden Sie unter. Kontingente für Distributionen mit mehreren Mandanten

Themen

Funktionsweise

In einer Standardverteilung enthält die Verteilung alle Einstellungen, die Sie für Ihre Website oder Anwendung aktivieren möchten, z. B. die Originalkonfigurationen, das Cache-Verhalten und die Sicherheitseinstellungen. Wenn Sie eine separate Website erstellen und viele der gleichen Einstellungen verwenden möchten, müssten Sie jedes Mal eine neue Distribution erstellen.

CloudFront Mehrmandantenverteilungen unterscheiden sich darin, dass Sie eine anfängliche Mehrmandantenverteilung erstellen können. Für jede neue Website erstellen Sie einen Verteilungsmandanten, der automatisch die definierten Werte der Quellverteilung erbt. Anschließend passen Sie spezifische Einstellungen für Ihren Distributionsmandanten an.

Übersicht

1. Zu Beginn erstellen Sie zunächst eine Mehrmandantenverteilung. CloudFront konfiguriert Ihre Verteilungseinstellungen für Sie auf der Grundlage Ihres Inhaltstyps. Sie können die Einstellungen für alle Ursprünge außer VPC-Ursprüngen anpassen. Die VPC-Ursprungseinstellungen werden auf

der VPC-Ursprungsressource selbst angepasst. Weitere Informationen zu den Einstellungen für die Verteilung mit mehreren Mandanten, die Sie anpassen können, finden Sie unter. Referenz für vorkonfigurierte Verteilungseinstellungen

- Das TLS-Zertifikat, das Sie für die Verteilung mit mehreren Mandanten verwenden, kann von Ihren Distributionsmandanten übernommen werden. Die Mehrmandantenverteilung selbst ist nicht routingfähig, sodass ihr kein Domainname zugeordnet ist.

2. CloudFront Erstellt standardmäßig eine Verbindungsgruppe für Sie. Die Verbindungsgruppe steuert, mit welcher Verbindung Zuschaueranfragen nach Inhalten verbunden CloudFront werden. Sie können einige Routing-Einstellungen in der Verbindungsgruppe anpassen.

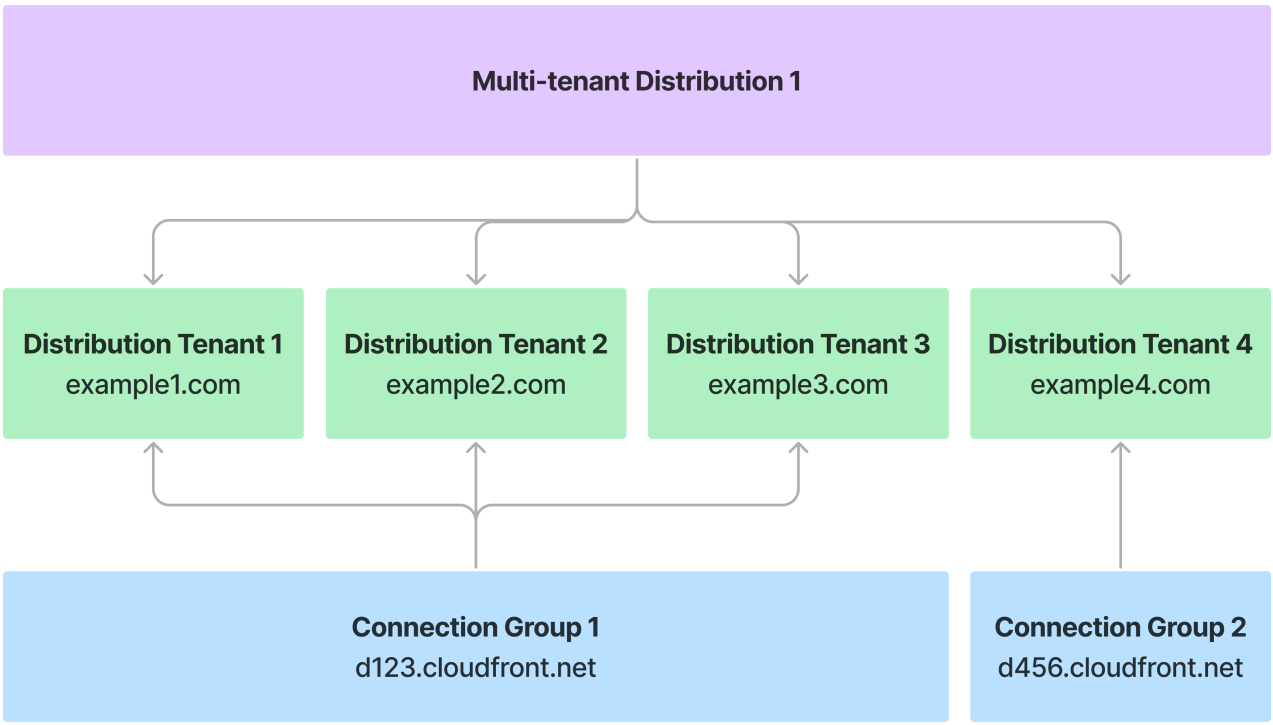
   Sie können dies ändern, indem Sie manuell Ihre eigene Verbindungsgruppe erstellen. Weitere Informationen finden Sie unter Erstellen Sie eine benutzerdefinierte Verbindungsgruppe (optional).
3. Anschließend erstellen Sie einen oder mehrere Verteilungsmandanten. Der Distributionsmandant ist die „Eingangstür“, über die Zuschauer auf Ihre Inhalte zugreifen können. Jeder Verteilungsmandant verweist auf die Mehrmandantenverteilung und wird automatisch der Verbindungsgruppe zugeordnet, die für Sie CloudFront erstellt wurde. Der Verteilungsmandant unterstützt eine einzelne Domäne oder Subdomäne.
4. Anschließend können Sie einige Einstellungen für den Verteilungsmandanten anpassen, z. B. Vanity-Domains und Herkunftspfade. Weitere Informationen finden Sie unter Anpassungen des Distributionsmandanten.
5. Schließlich müssen Sie den DNS-Eintrag in Ihrem DNS-Host aktualisieren, um den Datenverkehr an den Verteilungsmandanten weiterzuleiten. Rufen Sie dazu den CloudFront Endpunktwert aus Ihrer Verbindungsgruppe ab und erstellen Sie einen CNAME-Eintrag, der auf den CloudFront Endpunkt verweist.

Example Beispiel

Die folgende Grafik zeigt, wie eine Mehrmandantenverteilung, Verteilungsmandanten und Verbindungsgruppen zusammenarbeiten, um Ihren Zuschauern Inhalte für mehrere Domänen bereitzustellen.

1. Die Verteilung mit mehreren Mandanten definiert die geerbten Einstellungen für jeden Verteilungsmandanten. Sie verwenden die Mehrmandantenverteilung als Vorlage.
2. Jeder aus der Mehrmandantenverteilung erstellte Verteilungsmandant hat seine eigene Domäne.
3. Die Verteilungsmandanten werden automatisch der Verbindungsgruppe hinzugefügt, die Sie bei der CloudFront Erstellung der Mehrmandantenverteilung für Sie erstellt haben.

Verbindungsgruppen steuern, wie Viewer-Anfragen mit dem CloudFront Netzwerk verbunden werden.



Eine ausführliche Anleitung zur Erstellung von Mehrmandantenverteilungen finden Sie unter. Erstellen Sie eine CloudFront Distribution in der Konsole

Bedingungen

Die folgenden Konzepte beschreiben Komponenten von Mehrmandantenverteilungen:

Verteilung für mehrere Mandanten

Eine Blueprint-Verteilung für mehrere Mandanten, die alle gemeinsam genutzten Konfigurationseinstellungen für alle Verteilungsmandanten spezifiziert, einschließlich Cache-Verhalten, Sicherheitsvorkehrungen und Herkunft. Verteilungen mit mehreren Mandanten können den Datenverkehr nicht direkt bedienen. Sie müssen in Verbindung mit Verbindungsgruppen und Verteilungsmandanten verwendet werden.

Standardverteilung

Eine Distribution, die keine Multi-Tenant-Funktionalität bietet. Diese Distributionen eignen sich am besten für die Unterstützung einzelner Websites oder Apps.

Distributionsmandant

Ein Verteilungsmandant erbt die Mehrmandanten-Verteilungskonfiguration. Einige Konfigurationseinstellungen können auf der Ebene des Verteilungsmandanten angepasst werden. Der Verteilungsmandant muss über ein gültiges TLS-Zertifikat verfügen, das von der Mehrmandantenverteilung übernommen werden kann, sofern es die Domäne oder Subdomäne des Verteilungsmandanten abdeckt.

Der Verteilungsmandant muss einer Verbindungsgruppe zugeordnet sein. CloudFront erstellt eine Verbindungsgruppe für Sie, wenn Sie einen Verteilungsmandanten erstellen, und weist dieser Verbindungsgruppe automatisch alle Verteilungsmandanten zu.

Mehrmandantenfähigkeit

Sie können die Mehrmandantenverteilung verwenden, um Inhalte domänenübergreifend bereitzustellen und gleichzeitig Konfiguration und Infrastruktur gemeinsam zu nutzen. Dieser Ansatz ermöglicht es verschiedenen Domänen (so genannten Mandanten), gemeinsame Einstellungen aus der Mehrmandantenverteilung zu verwenden und gleichzeitig ihre eigenen Anpassungen beizubehalten.

Verbindungsgruppe

Stellt den CloudFront Routing-Endpunkt bereit, der Inhalte für Zuschauer bereitstellt. Sie müssen jeden Verteilungsmandanten einer Verbindungsgruppe zuordnen, um den entsprechenden CloudFront Routing-Endpunkt für den CNAME-Datensatz zu erhalten, den Sie für Ihre Verteilermandantendomäne oder -subdomäne erstellen. Verbindungsgruppen können von mehreren Verteilungsmandanten gemeinsam genutzt werden. Verbindungsgruppen verwalten die Routingeinstellungen für Verteilungsmandanten, z. B. IPv6 die Einstellungen für Anycast-IP-Listen.

Parameter

Eine Liste von Schlüssel-Wert-Paaren für Platzhalterwerte, z. B. Herkunftspfade und Domainnamen. Sie können Parameter in Ihrer Mehrmandantenverteilung definieren und Werte für diese Parameter auf der Ebene des Verteilungsmandanten angeben. Sie wählen aus, ob die Parameterwerte für den Verteilungsmandanten eingegeben werden müssen.

Wenn Sie keinen Wert für einen optionalen Parameter in einem Verteilungsmandanten angeben, wird der Standardwert aus der Mehrmandantenverteilung als Wert verwendet.

CloudFront Routing-Endpunkt

Kanonisches DNS für die Verbindungsgruppe, z. B. `d123.cloudfront.net` Wird im CNAME-Eintrag für Ihre Distributionsmandantendomain oder -Subdomain verwendet.

Anpassungen

Sie können Ihre Distributionsmandanten so anpassen, dass sie andere Einstellungen als die Mehrmandantenverteilung verwenden. Für jeden Verteilungsmandanten können Sie eine andere AWS WAF Web Access Control List (ACL), TLS-Zertifikate und geografische Einschränkungen angeben.

Nicht unterstützte Funktionen

Die folgenden Funktionen können nicht mit einer Mehrmandantenverteilung verwendet werden. Wenn Sie eine neue Mehrmandantenverteilung mit denselben Einstellungen wie Ihre Standarddistribution erstellen möchten, beachten Sie, dass einige Einstellungen nicht verfügbar sind.

Hinweise

- Derzeit gelten die AWS Firewall Manager Richtlinien nur für Ihre Standardverteilungen. Firewall Manager wird in einer future Version Unterstützung für Multi-Tenant-Distributionen hinzufügen.
- Im Gegensatz zu Standardverteilungen geben Sie Ihren Domainnamen (Alias) auf der Ebene des Verteilungsmandanten an. Weitere Informationen finden Sie unter Fordern Sie Zertifikate für Ihren CloudFront Distributionsmandanten an und zur CreateDistributionTenantAPI-Operation.

- Kontinuierlicher Einsatz
- Origin Access Identity (OAI) — Verwenden Sie stattdessen Origin Access Control (OAC).
- Dedizierte benutzerdefinierte IP-SSL-Unterstützung — Nur die `sni-only` Methode wird unterstützt.
- AWS WAF Klassische (V1) Web-ACL — Nur AWS WAF V2-Web ACLs werden unterstützt.
- Standardprotokollierung (Legacy)

- Minimale TTL
- Standard-TTL
- Maximale TTL
- ForwardedValues
- PriceClass
- Vertrauenswürdige Unterzeichner
- Reibungsloses Streaming
- AWS Identity and Access Management (IAM) -Serverzertifikate
- Dedizierte IP-Adressen
- Minimale Protokollversion SSLv3

Die folgenden Einstellungen können nicht in einer Mehrmandantenverteilung oder einem Verteilungsmandanten konfiguriert werden. Legen Sie stattdessen die Werte fest, die Sie in einer Verbindungsgruppe haben möchten. Alle Verteilungsmandanten, die der Verbindungsgruppe zugeordnet sind, verwenden diese Einstellungen. Weitere Informationen finden Sie unter Erstellen Sie eine benutzerdefinierte Verbindungsgruppe (optional).

- Aktivieren IPv6
- Statische Anycast-IP-Liste

Anpassungen des Distributionsmandanten

Wenn Sie eine Mehrmandantenverteilung verwenden, erben Ihre Verteilungsmandanten die Mehrmandanten-Verteilungskonfiguration. Sie können jedoch einige Einstellungen auf der Ebene der Verteilungsmandanten anpassen.

Sie können Folgendes anpassen:

- Parameter — Parameter sind Schlüssel-Wert-Paare, die Sie für die Ursprungsdomäne oder die Quellpfade verwenden können. Siehe Wie funktionieren Parameter mit Verteilungsmandanten.
- AWS WAF Web-ACL (V2) — Sie können eine separate Web-ACL für den Verteilungsmandanten angeben, wodurch die für die Mehrmandantenverteilung verwendete Web-ACL außer Kraft gesetzt wird. Sie können diese Einstellung auch für einen bestimmten Verteilungsmandanten deaktivieren, was bedeutet, dass der Verteilungsmandant den Web-ACL-Schutz nicht von der Mehrmandantenverteilung erbt. Weitere Informationen finden Sie unter AWS WAF Web-ACL.

- Geografische Einschränkungen — Geografische Einschränkungen, die Sie für einen Verteilungsmandanten angeben, haben Vorrang vor allen geografischen Einschränkungen für die Mehrmandantenverteilung. Wenn Sie beispielsweise Deutschland (DE) in Ihrer Mehrmandantenverteilung blockieren, blockieren alle zugehörigen Vertriebsmandanten auch DE. Wenn Sie DE jedoch für einen bestimmten Verteilungsmandanten zulassen, überschreiben diese Einstellungen für den Verteilungsmandanten die Einstellungen für die Mehrmandantenverteilung. Weitere Informationen finden Sie unter Beschränken Sie die geografische Verteilung Ihrer Inhalte.
- Invalidierungspfade — Geben Sie die Dateipfade zu den Inhalten an, die Sie für den Verteilungsmandanten ungültig machen möchten. Weitere Informationen finden Sie unter Dateien ungültig machen.
- Benutzerdefinierte TLS-Zertifikate — AWS Certificate Manager (ACM) -Zertifikate, die Sie für Verteilungsmandanten angeben, ergänzen das in der Mehrmandantenverteilung bereitgestellte Zertifikat. Wenn jedoch dieselbe Domäne sowohl durch das Mehrmandantenverteilungs- als auch durch das Verteilungsmandantenzertifikat abgedeckt ist, wird das Mandantenzertifikat verwendet. Weitere Informationen finden Sie unter Fordern Sie Zertifikate für Ihren CloudFront Distributionsmandanten an.
- Domainnamen — Sie müssen mindestens einen Domainnamen pro Distributionsmandant angeben.

Wie funktionieren Parameter mit Verteilungsmandanten

Ein Parameter ist ein Schlüssel-Wert-Paar, das Sie für Platzhalterwerte verwenden können. Definieren Sie die Parameter, die Sie in Ihrer Mehrmandantenverteilung verwenden möchten, und geben Sie an, ob sie erforderlich sind.

Wenn Sie Parameter in Ihrer Mehrmandantenverteilung definieren, wählen Sie aus, ob diese Parameter auf der Ebene der Verteilungsmandanten eingegeben werden müssen.

- Wenn Sie die Parameter so definieren, wie sie in der Mehrmandantenverteilung erforderlich sind, müssen sie auf der Ebene der Verteilungsmandanten eingegeben werden. (Sie werden nicht vererbt).
- Wenn die Parameter nicht erforderlich sind, können Sie in der Mehrmandantenverteilung einen Standardwert angeben, der vom Verteilungsmandanten geerbt wird.

Sie können Parameter in den folgenden Eigenschaften verwenden:

- Domänenname des Ursprungs
- Ursprungspfad

In der Mehrmandantenverteilung können Sie bis zu zwei Parameter für jede der vorherigen Eigenschaften definieren.

Beispielparameter

In den folgenden Beispielen finden Sie Informationen zur Verwendung von Parametern für den Domainnamen und den Herkunftspfad.

Parameter für Domainnamen

In der Konfiguration für die Verteilung mit mehreren Mandanten können Sie einen Parameter für den Ursprungsdomänennamen wie in den folgenden Beispielen definieren:

Amazon S3

- {{*parameter1*}}.amzn-s3-demo-logging-bucket.s3.us-east-1.amazonaws.com
- {{*parameter1*}}-amzn-s3-demo-logging-bucket.s3.us-east-1.amazonaws.com

Benutzerdefinierte Ursprünge

- {{*parameter1*}}.lambda-url.us-east-1.on.aws
- {{*parameter1*}}.mediapackagev2.ap-south-1.amazonaws.com

Wenn Sie einen Distributionsmandanten erstellen, geben Sie den Wert an, für den Sie ihn verwenden möchten*parameter1*.

```
"Parameters": [
  {
    "Name": "parameter1",
    "Value": "mycompany-website"
  }
]
```

Unter Verwendung der vorherigen Beispiele, die in der Mehrmandantenverteilung angegeben wurden, wird der Ursprungsdomänenname für den Verteilungsmandanten wie folgt aufgelöst:

- *mycompany-website*.amzn-s3-demo-bucket3.s3.us-east-1.amazonaws.com
- *mycompany-website*-amzn-s3-demo-bucket3.s3.us-east-1.amazonaws.com
- *mycompany-website*.lambda-url.us-east-1.on.aws
- *mycompany-website*.mediapackagev2.ap-south-1.amazonaws.com

Parameter für den Ursprungspfad

In ähnlicher Weise können Sie Parameter für den Origin-Pfad in der Multi-Tenant-Distribution wie in den folgenden Beispielen definieren:

- /{{*parameter2*}}
- /{{*parameter2*}}/test
- /public/{{*parameter2*}}/test
- /search?name={{*parameter2*}}

Wenn Sie einen Verteilungsmandanten erstellen, geben Sie den Wert an, für *parameter2* den Sie ihn verwenden möchten.

```
"Parameters": [
  {
    "Name": "parameter2",
    "Value": "myBrand"
  }
]
```

Unter Verwendung der vorherigen Beispiele, die in der Mehrmandantenverteilung angegeben wurden, wird der Quellpfad für den Verteilungsmandanten wie folgt aufgelöst:

- /*myBrand*
- /*myBrand*/test
- /public/*myBrand*/test
- /search?name=*myBrand*

Example Beispiel

Sie möchten mehrere Websites (Mandanten) für Ihre Kunden erstellen und müssen sicherstellen, dass jede Verteilungsmandantenressource die richtigen Werte verwendet.

1. Sie erstellen eine Mehrmandantenverteilung und geben zwei Parameter für die Konfiguration des Verteilungsmandanten an.
2. Für den Namen der Ursprungsdomäne erstellen Sie einen Parameter mit dem Namen *customer-name* und geben an, dass er erforderlich ist. Sie geben den Parameter vor dem S3-Bucket ein, sodass er wie folgt aussieht:

   `{{customer-name}}.amzn-s3-demo-bucket3.s3.us-east-1.amazonaws.com`.
3. Für Origin Path erstellen Sie einen zweiten Parameter mit dem Namen und geben an*my-theme*, dass er optional ist, mit dem Standardwert*basic*. Ihr Ursprungspfad wird wie folgt angezeigt: `/{{my-theme}}`
4. Wenn Sie einen Distributionsmandanten erstellen:
   - Für den Domainnamen müssen Sie einen Wert für angeben*customer-name*, da dieser in der Mehrmandantenverteilung als erforderlich gekennzeichnet ist.
   - Für den Quellpfad können Sie optional einen Wert für angeben *my-theme* oder den Standardwert verwenden.

Fordern Sie Zertifikate für Ihren CloudFront Distributionsmandanten an

Wenn Sie einen Verteilungsmandanten erstellen, erbt der Mandant das gemeinsame Zertifikat AWS Certificate Manager (ACM) von der Mehrmandantenverteilung. Dieses gemeinsame Zertifikat stellt HTTPS für alle Mandanten bereit, die der Mehrmandantenverteilung zugeordnet sind.

Wenn Sie einen CloudFront Verteilungsmandanten erstellen oder aktualisieren, um Domänen hinzuzufügen, können Sie ein verwaltetes CloudFront Zertifikat von ACM hinzufügen. CloudFront erhält dann in Ihrem Namen ein HTTP-validiertes Zertifikat von ACM. Sie können dieses ACM-Zertifikat auf Mandantenebene für benutzerdefinierte Domänenkonfigurationen verwenden. CloudFront optimiert den Verlängerungsablauf, um sicherzustellen, dass Zertifikate up-to-date und die sichere Bereitstellung von Inhalten unterbrechungsfrei erfolgen.

> **Note**
>
> Sie besitzen das Zertifikat, aber es kann nur mit CloudFront Ressourcen verwendet werden und der private Schlüssel kann nicht exportiert werden.

Sie können das Zertifikat anfordern, wenn Sie den Verteilungsmandanten erstellen oder aktualisieren.

Themen

- Fügen Sie eine Domäne und ein Zertifikat hinzu (Distributionsmandant)
- Schließen Sie die Domaineinrichtung ab
- Verweisen Sie Domains auf CloudFront
- Überlegungen zur Domain (Vertriebsmandant)
- Wildcard-Domains (Vertriebsmandant)

Fügen Sie eine Domäne und ein Zertifikat hinzu (Distributionsmandant)

Das folgende Verfahren zeigt Ihnen, wie Sie eine Domäne hinzufügen und das Zertifikat für einen Verteilungsmandanten aktualisieren.

So fügen Sie eine Domäne und ein Zertifikat hinzu (Distributionsmandant)

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie unter SaaS die Option Distribution Tenants aus.
3. Suchen Sie nach dem Distributionsmandanten. Verwenden Sie das Dropdownmenü in der Suchleiste, um nach Domäne, Name, Distributions-ID, Zertifikat-ID, Verbindungsgruppen-ID oder Web-ACL-ID zu filtern.
4. Wählen Sie den Namen des Verteilungsmandanten aus.
5. Wählen Sie für Domains die Option Domain verwalten aus.
6. Wählen Sie unter Zertifikat aus, ob Sie ein benutzerdefiniertes TLS-Zertifikat für Ihren Distributionsmandanten wünschen. Das Zertifikat bestätigt, ob Sie berechtigt sind, den Domainnamen zu verwenden. Das Zertifikat muss in der Region USA Ost (Nord-Virginia) existieren.

7. Wählen Sie für Domains die Option Add domain aus und geben Sie den Domainnamen ein. Abhängig von Ihrer Domain werden die folgenden Meldungen unter dem von Ihnen eingegebenen Domainnamen angezeigt.

   - Diese Domain ist durch das Zertifikat abgedeckt.
   - Diese Domain ist durch das Zertifikat abgedeckt, deren Validierung noch aussteht.
   - Diese Domain ist nicht durch ein Zertifikat abgedeckt. (Das bedeutet, dass Sie die Inhaberschaft der Domain verifizieren müssen.)

8. Wählen Sie „Distributionsmandant aktualisieren“.

   Auf der Seite mit den Mandantendetails finden Sie unter Domains die folgenden Felder:

   - Domainbesitz — Der Status des Domainbesitzes. Bevor Inhalte CloudFront bereitgestellt werden können, muss Ihr Domainbesitz mithilfe der TLS-Zertifikatsvalidierung verifiziert werden.
   - DNS-Status — Die DNS-Einträge Ihrer Domain müssen auf diese verweisen, CloudFront um den Datenverkehr korrekt weiterzuleiten.

9. Wenn Ihre Domaininhaberschaft nicht bestätigt wurde, wählen Sie auf der Seite mit den Mandantendetails unter Domains die Option Domaineinrichtung abschließen aus und führen Sie dann das folgende Verfahren aus, um den DNS-Eintrag auf Ihren CloudFront Domainnamen zu verweisen.

Schließen Sie die Domaineinrichtung ab

Gehen Sie wie folgt vor, um zu überprüfen, ob Sie Eigentümer der Domain für Ihre Vertriebsmandanten sind. Wählen Sie je nach Ihrer Domain eines der folgenden Verfahren aus.

> **Note**
>
> Wenn auf Ihre Domain bereits CloudFront mit einem Amazon Route 53-Aliaseintrag verwiesen wird, müssen Sie Ihren DNS-TXT-Eintrag mit `_cf-challenge.` vor dem Domainnamen hinzufügen. Dieser TXT-Eintrag bestätigt, dass Ihr Domainname mit verknüpft ist. CloudFront Wiederholen Sie diesen Schritt für jede Domain. Im Folgenden wird gezeigt, wie Sie Ihren TXT-Eintrag aktualisieren:
>
> - Name des Datensatzes: `_cf-challenge.`*`DomainName`*
> - Datensatztyp: `TXT`

- Wert des Datensatzes: *CloudFrontRoutingEndpoint*

Ihr TXT-Eintrag könnte beispielsweise wie folgt aussehen: `_cf-challenge.example.com TXT d111111abcdef8.cloudfront.net`
Sie finden Ihren CloudFront Routing-Endpunkt in der Konsole auf der Detailseite des Distributionsmandanten oder verwenden Sie die ListConnectionGroupsAPI-Aktion in der Amazon CloudFront API-Referenz, um ihn zu finden.

Tip

Wenn Sie ein SaaS-Anbieter sind und die Ausstellung von Zertifikaten zulassen möchten, ohne dass Ihre Kunden (Mandanten) einen TXT-Eintrag direkt zu ihrem DNS hinzufügen müssen, gehen Sie wie folgt vor:

1. Wenn Sie Eigentümer der Domain sind`example-saas-provider.com`, weisen Sie Ihren Mandanten Subdomains zu, z. B. `customer-123.example-saas-provider.com`
2. Fügen Sie in Ihrem DNS den `_cf_challenge.customer-123.example-saas-provider.com TXT d111111abcdef8.cloudfront.net` TXT-Eintrag zu Ihrer DNS-Konfiguration hinzu.
3. Als Nächstes können Ihre Kunden (die Mandanten) dann ihren eigenen DNS-Eintrag aktualisieren, um ihren Domainnamen der von Ihnen angegebenen Subdomain zuzuordnen.

   ```
   www.customer-domain.com CNAME customer-123.example-saas-provider.com
   ```

I have existing traffic

Wählen Sie diese Option, wenn Ihre Domain keine Ausfallzeiten toleriert. Sie müssen Zugriff auf Ihren origin/web Server haben. Gehen Sie wie folgt vor, um den Domainbesitz zu überprüfen.

Um die Einrichtung der Domain abzuschließen, wenn bereits Traffic vorhanden ist

1. Wählen Sie unter Spezifizieren Sie Ihren Web-Traffic die Option Ich habe vorhandenen Traffic und dann Weiter aus.

2. Wählen Sie für Domainbesitz verifizieren eine der folgenden Optionen aus:

   - Bestehendes Zertifikat verwenden — Suchen Sie nach einem vorhandenen ACM-Zertifikat oder geben Sie den Zertifikat-ARN ein, der die aufgelisteten Domänen abdeckt.
   - Manuelles Hochladen von Dateien — Wählen Sie aus, ob Sie direkten Zugriff auf das Hochladen von Dateien auf Ihren Webserver haben.

     Erstellen Sie für jede Domain eine Klartextdatei, die Ihr Validierungstoken vom Token-Speicherort enthält, und laden Sie es an Ihren Ursprung unter dem angegebenen Dateipfad auf Ihrem vorhandenen Server hoch. Der Pfad zu dieser Datei könnte wie das folgende Beispiel aussehen:`/.well-known/pki-validation/acm_9c2a7b2ec0524d09fa6013efb73ad123.txt`. Nachdem Sie diesen Schritt abgeschlossen haben, verifiziert ACM das Token und stellt dann das TLS-Zertifikat für die Domain aus.
   - HTTP-Weiterleitung — Wählen Sie aus, ob Sie keinen direkten Zugriff zum Hochladen von Dateien auf Ihren Webserver haben oder ob Sie einen CDN- oder Proxydienst verwenden.

     Erstellen Sie für jede Domain eine 301-Weiterleitung auf Ihrem vorhandenen Server. Kopieren Sie den bekannten Pfad unter Umleiten von und verweisen Sie auf den angegebenen Zertifikatsendpunkt unter Umleiten zu. Ihre Weiterleitung könnte wie das folgende Beispiel aussehen:

     ```
     If the URL matches: example.com/.well-known/pki-validation/
     leabe938a4fe077b31e1ff62b781c123.txt
     Then the settings are:Forwarding URL
     Then 301 Permanent Redirect:To validation.us-east-1.acm-
     validations.aws/123456789012/.well-known/pki-validation/
     leabe938a4fe077b31e1ff62b781c123.txt
     ```

   > **Note**
   >
   > Sie können Zertifikatsstatus überprüfen wählen, um zu überprüfen, wann ACM das Zertifikat für die Domain ausstellt.

3. Wählen Sie Weiter aus.
4. Führen Sie die Schritte für Verweisen Sie Domains auf CloudFront aus.

I don't have traffic

Wählen Sie diese Option, wenn Sie neue Domains hinzufügen. CloudFront verwaltet die Zertifikatsvalidierung für Sie.

Um die Domaineinrichtung abzuschließen, wenn Sie keinen Traffic haben

1. Wählen Sie unter Spezifizieren Sie Ihren Web-Traffic die Option Ich habe noch keinen Traffic aus.
2. Führen Sie für jeden Domainnamen die Schritte für ausVerweisen Sie Domains auf CloudFront.
3. Nachdem Sie Ihre DNS-Einträge für jeden Domainnamen aktualisiert haben, wählen Sie Weiter.
4. Warten Sie, bis das Zertifikat ausgestellt wurde.

   > **Note**
   >
   > Sie können Zertifikatsstatus überprüfen wählen, um zu überprüfen, wann ACM das Zertifikat für die Domain ausstellt.

5. Wählen Sie Absenden aus.

Verweisen Sie Domains auf CloudFront

Aktualisieren Sie Ihre DNS-Einträge, um den Verkehr von jeder Domain zum CloudFront Routing-Endpunkt weiterzuleiten. Sie können mehrere Domainnamen haben, aber sie müssen alle zu diesem Endpunkt führen.

Um Domains zu verweisen CloudFront

1. Kopieren Sie den Wert des CloudFront Routing-Endpunkts, z. B. d111111abcdef8.cloudfront.net.
2. Aktualisieren Sie Ihre DNS-Einträge, um den Verkehr von jeder Domain zum Routing-Endpunkt weiterzuleiten. CloudFront

   1. Melden Sie sich bei der Verwaltungskonsole Ihres Domain-Registrars oder DNS-Providers an.
   2. Navigieren Sie zum DNS-Verwaltungsbereich für Ihre Domain.
      - Für Subdomains — Erstellen Sie einen CNAME-Eintrag. Zum Beispiel:
        - Name — Ihre Subdomain (wie oder) `www app`

- Value//Target — Der CloudFront Routing-Endpunkt
- Datensatztyp — CNAME
- TTL — 3600 (oder was auch immer für Ihren Anwendungsfall geeignet ist)

- Für apex/root Domains — Dies erfordert eine eindeutige DNS-Konfiguration, da CNAME-Standardeinträge nicht auf Stamm- oder Apex-Domänenebene verwendet werden können. Da die meisten DNS-Anbieter ALIAS-Einträge nicht unterstützen, empfehlen wir, einen ALIAS-Eintrag in Route 53 zu erstellen. Zum Beispiel:
  - Name — Ihre Apex-Domain (z. B.`example.com`)
  - Datensatztyp — A
  - Alias — Ja
  - Alias-Ziel — Ihr CloudFront Routing-Endpunkt
  - Routing-Richtlinie — Einfach (oder was auch immer für Ihren Anwendungsfall geeignet ist)

3. Stellen Sie sicher, dass die DNS-Änderung weitergegeben wurde. (Dies passiert normalerweise, wenn die TTL abgelaufen ist. Manchmal kann es 24-48 Stunden dauern.) Verwenden Sie ein Tool wie `dig` oder`nslookup`.

```
dig www.example.com
# Should eventually return a CNAME pointing to your CloudFront routing endpoint
```

3. Kehren Sie zur CloudFront Konsole zurück und wählen Sie Submit. Wenn Ihre Domain aktiv ist, CloudFront aktualisiert sie den Domain-Status, um anzuzeigen, dass Ihre Domain bereit ist, Traffic bereitzustellen.

Weitere Informationen finden Sie in der Dokumentation Ihres DNS-Anbieters:

- Cloudflare
- ClouDNS
- DNSimple
- Gandi.net
- GoDaddy
- Google Cloud-DNS
- Nennen Sie billig

Überlegungen zur Domain (Vertriebsmandant)

Wenn eine Domäne aktiv ist, wurde die Domänensteuerung eingerichtet, CloudFront die auf alle Besucheranfragen an diese Domäne reagiert. Nach der Aktivierung kann eine Domain nicht deaktiviert oder in einen inaktiven Status geändert werden. Die Domain kann keiner anderen CloudFront Ressource zugeordnet werden, solange sie bereits verwendet wird. Um die Domain einer anderen Distribution zuzuordnen, verwenden Sie die UpdateDomainAssociationAnfrage, um die Domain von einer CloudFront Ressource auf eine andere zu verschieben.

Wenn eine Domain inaktiv ist, reagiert sie CloudFront nicht auf Zuschaueranfragen an die Domain. Beachten Sie Folgendes, solange die Domain inaktiv ist:

- Wenn Sie eine ausstehende Zertifikatsanfrage haben, CloudFront beantwortet Anfragen für den bekannten Pfad. Solange die Anfrage aussteht, kann die Domain keinen anderen CloudFront Ressourcen zugeordnet werden.
- Wenn Sie keine ausstehende Zertifikatsanforderung haben, beantwortet sie CloudFront keine Anfragen für die Domain. Sie können die Domain mit anderen CloudFront Ressourcen verknüpfen.
- Pro Verteilungsmandant kann nur eine ausstehende Zertifikatsanforderung vorliegen. Bevor Sie ein weiteres Zertifikat für weitere Domänen anfordern können, müssen Sie die bestehende ausstehende Anfrage stornieren. Durch das Stornieren einer vorhandenen Zertifikatsanforderung wird das zugehörige ACM-Zertifikat nicht gelöscht. Sie können das mithilfe der ACM-API löschen.
- Wenn Sie ein neues Zertifikat auf Ihren Distributionsmandanten anwenden, wird dadurch die Zuordnung zum vorherigen Zertifikat aufgehoben. Sie können das Zertifikat wiederverwenden, um die Domäne für einen anderen Verteilungsmandanten abzudecken.

Wie bei Verlängerungen von DNS-validierten Zertifikaten werden Sie benachrichtigt, wenn die Zertifikatserneuerung erfolgreich ist. Sie müssen jedoch nichts weiter tun. CloudFront verwaltet die Zertifikatsverlängerung für Ihre Domain automatisch.

> **Note**
>
> Sie müssen die ACM-API-Operationen nicht aufrufen, um Ihre Zertifikatsressourcen zu erstellen oder zu aktualisieren. Sie können Ihre Zertifikate verwalten, indem Sie die UpdateDistributionTenantAPI-Operationen CreateDistributionTenantund verwenden, um die Details für Ihre verwaltete Zertifikatsanforderung anzugeben.

Wildcard-Domains (Vertriebsmandant)

Platzhalterdomänen werden in den folgenden Situationen für Verteilungsmandanten unterstützt:

- Wenn der Platzhalter in dem gemeinsamen Zertifikat enthalten ist, das von der übergeordneten Mehrmandantenverteilung geerbt wurde
- Wenn Sie ein gültiges vorhandenes benutzerdefiniertes TLS-Zertifikat für Ihren Distributionsmandanten verwenden

Erstellen Sie eine benutzerdefinierte Verbindungsgruppe (optional)

CloudFront Erstellt standardmäßig eine Verbindungsgruppe für Sie, wenn Sie eine Verteilung mit mehreren Mandanten erstellen. Die Verbindungsgruppe steuert, mit welchen Inhaltsanfragen von Zuschauern eine Verbindung hergestellt wird. CloudFront

Es wird empfohlen, die Standardverbindungsgruppe zu verwenden. Wenn Sie jedoch Unternehmensanwendungen isolieren oder Gruppen von Verteilungsmandanten separat verwalten müssen, können Sie eine benutzerdefinierte Verbindungsgruppe erstellen. Beispielsweise müssen Sie möglicherweise einen Verteilungsmandanten in eine separate Verbindungsgruppe verschieben, wenn er von einem DDo S-Angriff betroffen ist. Auf diese Weise können Sie andere Verteilungsmandanten vor Stößen schützen.

Erstellen Sie eine benutzerdefinierte Verbindungsgruppe (optional)

Optional können Sie eine benutzerdefinierte Verbindungsgruppe für Ihre Verteilungsmandanten erstellen.

Um eine benutzerdefinierte Verbindungsgruppe zu erstellen (optional)

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Settings (Einstellungen).
3. Schalten Sie die Einstellungen für die Verbindungsgruppe ein.
4. Wählen Sie im Navigationsbereich Verbindungsgruppen und dann Verbindungsgruppe erstellen aus.
5. Geben Sie unter Name der Verbindungsgruppe einen Namen für die Verbindungsgruppe ein. Sie können diesen Namen nicht aktualisieren, nachdem Sie die Verbindungsgruppe erstellt haben.

6. Geben Sie für an IPv6, ob Sie dieses IP-Protokoll aktivieren möchten. Weitere Informationen finden Sie unter Aktivieren IPv6.
7. Geben Sie für die statische IP-Liste von Anycast an, ob Sie Traffic von einer Reihe von IP-Adressen aus an Ihre Verteilermandanten weiterleiten möchten. Weitere Informationen finden Sie unter Statische IP-Liste von Anycast.
8. (Optional) Fügen Sie Ihrer Verbindungsgruppe Tags hinzu.
9. Wählen Sie Verbindungsgruppe erstellen aus.

Wenn Ihre Verbindungsgruppe erstellt wurde, finden Sie die von Ihnen angegebenen Einstellungen sowie den ARN und den Endpunkt.

- Der ARN sieht wie das folgende Beispiel aus: `arn:aws:cloudfront::123456789012:connection-group/cg_2uVbA9KeWaADTbKzhj9lcKDoM25`
- Der Endpunkt sieht wie das folgende Beispiel aus: d111111abcdef8.cloudfront.net

Sie können Ihre benutzerdefinierte Verbindungsgruppe bearbeiten oder löschen, nachdem Sie sie erstellt haben. Bevor Sie eine Verbindungsgruppe löschen können, müssen Sie zunächst alle zugehörigen Verteilungsmandanten aus der Verbindungsgruppe löschen. Sie können die Standardverbindungsgruppe nicht löschen, die für Sie CloudFront erstellt wurde, als Sie Ihre Mehrmandantenverteilung erstellt haben.

Important

Wenn Sie die Verbindungsgruppe für einen Verteilungsmandanten ändern, CloudFront wird weiterhin Datenverkehr für den Verteilungsmandanten übertragen, allerdings mit erhöhter Latenz. Es wird empfohlen, den DNS-Eintrag für den Verteilungsmandanten so zu aktualisieren, dass er den CloudFront Routingendpunkt der neuen Verbindungsgruppe verwendet.
Solange Sie den DNS-Eintrag nicht aktualisieren, basiert CloudFront das Routing auf den Einstellungen, die für den Routing-Endpunkt definiert sind, auf den die Website derzeit mit DNS verweist. Gehen Sie beispielsweise davon aus, dass Ihre Standard-Verbindungsgruppe nicht Anycast static verwendet, Ihre neue, benutzerdefinierte Verbindungsgruppe IPs jedoch schon. Sie müssen den DNS-Eintrag aktualisieren, bevor CloudFront Sie Anycast static IPs für die Verteilungsmandanten in Ihrer benutzerdefinierten Verbindungsgruppe verwenden.

Migrieren Sie zu einer Mehrmandanten-Distribution

Wenn Sie über eine CloudFront Standarddistribution verfügen und zu einer Multi-Tenant-Distribution migrieren möchten, gehen Sie wie folgt vor.

So migrieren Sie von einer Standarddistribution zu einer Mehrmandantenverteilung

1. Überprüfen Sie das Nicht unterstützte Funktionen.
2. Erstellen Sie eine Mehrmandantenverteilung mit derselben Konfiguration wie Ihre Standarddistribution, abzüglich der nicht unterstützten Funktionen. Weitere Informationen finden Sie unter Erstellen Sie eine CloudFront Distribution in der Konsole.
3. Erstellen Sie einen Distributionsmandanten und fügen Sie einen alternativen Domainnamen hinzu, dessen Eigentümer Sie sind.

   > **Warning**
   >
   > Verwenden Sie nicht den aktuellen Domainnamen, der Ihrer Standarddistribution zugeordnet ist. Fügen Sie stattdessen eine Platzhalter-Domain hinzu. Sie werden Ihre Domain später verschieben. Weitere Informationen zum Erstellen eines Verteilungsmandanten finden Sie unterErstellen Sie eine CloudFront Distribution in der Konsole.

4. Stellen Sie ein vorhandenes Zertifikat für die Domäne des Verteilungsmandanten bereit. Dies ist das Zertifikat, das die Platzhalterdomäne und die Domäne, die Sie verschieben möchten, abdeckt.
5. Kopieren Sie den CloudFront Routing-Endpunkt von der Detailseite des Verteilungsmandanten in der Konsole. Sie können es auch mithilfe der ListConnectionGroupsAPI-Aktion in der Amazon CloudFront API-Referenz finden.
6. Um die Inhaberschaft der Domain zu verifizieren, erstellen Sie einen DCV-TXT-Eintrag mit einem Unterstrich (_), der auf den CloudFront Routing-Endpunkt für Ihren Distributionsmandanten verweist. Weitere Informationen finden Sie unter Verweisen Sie Domains auf CloudFront.
7. Wenn Ihre Änderungen übernommen wurden, aktualisieren Sie Ihren Verteilungsmandanten so, dass er die Domain verwendet, die Sie zuvor für Ihre Standardverteilung verwendet haben.

   - Konsole — Eine ausführliche Anleitung finden Sie unterFügen Sie eine Domäne und ein Zertifikat hinzu (Distributionsmandant).

- API — Verwenden Sie die UpdateDomainAssociationAPI-Aktion in der Amazon CloudFront API-Referenz.

> **Important**
>
> Dadurch wird der Cache-Schlüssel für Ihre Inhalte zurückgesetzt. Danach CloudFront beginnt das Zwischenspeichern Ihrer Inhalte mithilfe des neuen Cache-Schlüssels. Weitere Informationen finden Sie unter Den Cache-Schlüssel verstehen.

8. Aktualisieren Sie Ihren DNS-Eintrag so, dass Ihre Domain auf den CloudFront Routing-Endpunkt für Ihren Verteilungsmandanten verweist. Sobald Sie diesen Schritt abgeschlossen haben, ist Ihre Domain bereit, den Datenverkehr an Ihren Distributionsmandanten weiterzuleiten. Weitere Informationen finden Sie unter Verweisen Sie Domains auf CloudFront.
9. (Optional) Nachdem Sie Ihre Domain erfolgreich zu einem Verteilungsmandanten migriert haben, können Sie ein anderes CloudFront verwaltetes Zertifikat verwenden, das den Domainnamen für Ihren Verteilungsmandanten abdeckt. Um ein verwaltetes Zertifikat anzufordern, erstellen Sie einen separaten TXT-Eintrag, um das Zertifikat auszustellen, und folgen Sie den Schritten hier unterSchließen Sie die Domaineinrichtung ab.

Eine Verteilung erstellen

In diesem Thema wird erklärt, wie Sie mit der CloudFront Konsole eine Distribution erstellen.

Übersicht

1. Erstellen Sie mindestens einen Amazon-S3-Bucket oder konfigurieren Sie HTTP-Server als Ursprungs-Server. Ein Ursprung ist der Speicherort, an dem Sie die Originalversion Ihrer Inhalte speichern. Wenn CloudFront Sie eine Anfrage für Ihre Dateien erhalten, geht sie an den Ursprung, um die Dateien abzurufen, die sie an Edge-Standorten verteilt. Sie können jede Kombination von Amazon S3-Buckets und HTTP-Servern als Ursprungs-Server verwenden.

   - Wenn Sie Amazon S3 verwenden, darf der Name Ihres Buckets nur Kleinbuchstaben enthalten und er darf keine Leerzeichen enthalten.
   - Wenn Sie einen EC2 Amazon-Server oder einen anderen benutzerdefinierten Ursprung verwenden, überprüfen Sie diesVerwenden Sie Amazon EC2 (oder einen anderen benutzerdefinierten Ursprung).

- Informationen zur aktuell gültigen maximalen Anzahl von Ursprüngen, die Sie für eine Verteilung erstellen können, oder zum Anfordern eines höheren Kontingents finden Sie unter Allgemeine Kontingente für Verteilungen.

2. Laden Sie Ihre Inhalte auf Ihre Ursprungsserver hoch. Sie machen Ihre Objekte öffentlich lesbar oder Sie können CloudFront signiert verwenden, URLs um den Zugriff auf Ihre Inhalte einzuschränken.

   > **Important**
   >
   > Es liegt in Ihrer Verantwortung, die Sicherheit Ihres Ursprungsservers sicherzustellen. Sie müssen sicherstellen, dass diese Person über die erforderliche Zugriffsberechtigung für den Server CloudFront verfügt und dass Ihre Inhalte durch die Sicherheitseinstellungen geschützt sind.

3. Erstellen Sie Ihre CloudFront Distribution:

   - Ein detailliertes Verfahren zum Erstellen einer Distribution in der CloudFront Konsole finden Sie unterErstellen Sie eine CloudFront Distribution in der Konsole.
   - Informationen zum Erstellen einer Distribution mithilfe der CloudFront API finden Sie CreateDistributionin der Amazon CloudFront API-Referenz.

4. (Optional) Wenn Sie die CloudFront Konsole verwenden, um Ihre Distribution zu erstellen, erstellen Sie weitere Cache-Verhaltensweisen oder -Ursprünge für die Verteilung. Weitere Informationen zu Verhaltensweisen und Ursprüngen finden Sie unter Um eine Multi-Tenant-Distribution zu aktualisieren.

5. Testen Sie Ihre Verteilung. Weitere Informationen zum Testen finden Sie unter Testen Sie eine Distribution.

6. Entwickeln Sie Ihre Website oder Anwendung so, dass sie mithilfe des von CloudFront nach der Erstellung Ihrer Verteilung in Schritt 3 zurückgegebenen Domänennamens auf Ihre Inhalte zugreift. Wenn beispielsweise d111111abcdef8.cloudfront.net als Domainnamen für Ihre Distribution CloudFront zurückgegeben wird, lautet die URL für die Datei `image.jpg` in einem Amazon S3 S3-Bucket oder im Stammverzeichnis auf einem HTTP-Server. `https://d111111abcdef8.cloudfront.net/image.jpg`

   Wenn Sie bei der Erstellung Ihrer Distribution einen oder mehrere alternative Domainnamen (CNAMEs) angegeben haben, können Sie Ihren eigenen Domainnamen verwenden. In diesem

Fall könnte die URL für `image.jpg` folgendermaßen lauten: `https://www.example.com/image.jpg`.

Beachten Sie Folgendes:

- Wenn Sie signiert verwenden möchten, um URLs den Zugriff auf Ihre Inhalte einzuschränken, finden Sie weitere Informationen unterStellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit.
- Wenn Sie komprimierte Inhalte bereitstellen möchten, informieren Sie sich unter Komprimierte Dateien bereitstellen.
- Informationen zum CloudFront Anfrage- und Antwortverhalten für Amazon S3 und benutzerdefinierte Ursprünge finden Sie unterVerhalten von Anfragen und Antworten.

Themen

Erstellen Sie eine CloudFront Distribution in der Konsole

Wenn Sie eine Verteilung erstellen, CloudFront konfiguriert es Ihre Verteilungseinstellungen für Sie, die auf Ihrem Inhaltsherkunftstyp basieren. Weitere Informationen zu den vorkonfigurierten Einstellungen finden Sie unter. Referenz für vorkonfigurierte Verteilungseinstellungen Sie können auch Mehrmandantenverteilungen mit Einstellungen erstellen, die für mehrere Distributionsmandanten wiederverwendet werden können. Weitere Informationen finden Sie unter Erfahren Sie, wie Distributionen mit mehreren Mandanten funktionieren. Alternativ können Sie Ihre eigenen Verteilungseinstellungen manuell konfigurieren.

Multi-tenant

Um eine Mehrmandantenverteilung zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Distributionen und dann Distribution erstellen aus.

3. Wählen Sie Multi-Tenant-Architektur, Next.
4. Geben Sie einen Distributionsnamen für die Multi-Tenant-Distribution ein. Der Name wird als Wert für den `Name` Schlüssel angezeigt. Sie können diesen Wert später ändern. Sie können bis zu 50 Tags für Ihre Multi-Tenant-Distribution hinzufügen. Weitere Informationen finden Sie unter Kennzeichnen Sie eine Distribution.
5. (Optional) Wählen Sie für das Wildcard-Zertifikat das AWS Certificate Manager (ACM) - Zertifikat aus, das alle Subdomänen unter der Stammdomain abdeckt, z. B. *`*.example.com`* Das Zertifikat muss sich in der Region USA Ost (Nord-Virginia) befinden.
6. Wählen Sie Weiter aus.
7. Wählen Sie auf der Seite „Herkunft angeben“ den Herkunftstyp aus, von dem Ihre Inhalte abgerufen CloudFront werden sollen. CloudFront verwendet die empfohlenen Einstellungen für diesen Herkunftstyp für Ihre Multi-Tenant-Distribution. Weitere Informationen zu den empfohlenen Einstellungen finden Sie unterReferenz für vorkonfigurierte Verteilungseinstellungen.
8. Wählen Sie für Origin unter dem von Ihnen ausgewählten Quelltyp den zu verwendenden Ursprung aus, oder geben Sie ihn ein.
9. Geben Sie als Origin-Pfad den Schrägstrich (`/`) gefolgt vom Quellpfad ein.
10. (Optional) Um einen Parameter hinzuzufügen, wählen Sie entweder für den Namen der Ursprungsdomain oder den Quellpfad die Option Parameter einfügen aus. Sie können bis zu zwei Parameter für jedes Feld eingeben.

    a. Wählen Sie Neuen Parameter erstellen.

    b. Geben Sie im Dialogfeld Neuen Parameter erstellen unter Parametername einen eindeutigen Namen für den Parameter und optional eine Beschreibung ein.

    c. Aktivieren Sie für Erforderlicher Parameter das Kontrollkästchen, damit dieser Parameterwert auf der Ebene des Verteilungsmandanten erforderlich ist. Wenn er nicht erforderlich ist, geben Sie einen Standardwert ein, den der Verteilungsmandant erbt.

    d. Wählen Sie Parameter erstellen aus. Dieser Parameter wird im entsprechenden Feld angezeigt.
11. Wählen Sie für Optionen eine der folgenden Optionen aus:

    - Empfohlene Origin-Einstellungen verwenden — Verwenden Sie die empfohlenen Standard-Cache- und Origin-Einstellungen für den ausgewählten Origin-Typ.
    - Origin-Einstellungen anpassen — Passen Sie die Cache- und Origin-Einstellungen an. Wenn Sie diese Option wählen, geben Sie Ihre eigenen Werte an, die angezeigt werden.

12. Wählen Sie Weiter aus.
13. Wählen Sie auf der Seite Sicherheitsvorkehrungen aktivieren aus, ob der AWS WAF Sicherheitsschutz aktiviert werden soll. Sie können die Web-ACL später für bestimmte Verteilungsmandanten anpassen. Weitere Informationen finden Sie unter AWS WAF Für eine neue Distribution aktivieren.
14. Wählen Sie Weiter, Verteilung erstellen aus.
15. Auf der Seite Verteilungen wird Ihre Mehrmandantenverteilung in der Ressourcenliste angezeigt. Sie können das Drop-down-Menü Alle Verteilungen auswählen, um nach Standardverteilung oder Mehrmandantenverteilung zu filtern. Sie können auch die Spalte Typ auswählen, um nach Standard- oder Mehrmandantenverteilung zu filtern.

CloudFront Erstellt standardmäßig eine Verbindungsgruppe für Sie. Die Verbindungsgruppe steuert, mit welcher Verbindung Zuschaueranfragen nach Inhalten verbunden CloudFront werden. Sie können einige Routing-Einstellungen in der Verbindungsgruppe anpassen. Weitere Informationen finden Sie unter Erfahren Sie, wie Distributionen mit mehreren Mandanten funktionieren.

Sie können zusätzliche Verteilungsmandanten erstellen, indem Sie die Mehrmandantenverteilung als Vorlage verwenden.

Um einen Verteilungsmandanten zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Führen Sie im Navigationsbereich einen der folgenden Schritte aus:
   - Wählen Sie Verteilungen, wählen Sie eine Mehrmandantenverteilung aus, und klicken Sie dann auf Mandant erstellen.
   - Wählen Sie Distribution Tenants und dann Create Tenant aus.
3. Geben Sie unter Name des Distributionsmandanten den Namen ein. Der Name muss in Ihrem Namen eindeutig sein AWS-Konto und kann nach der Erstellung nicht mehr geändert werden.
4. Wählen Sie für die Vorlagenverteilung eine Distributions-ID für mehrere Mandanten aus der Liste aus.

5. Fügen Sie unter Tags verwalten bis zu 50 Schlüssel-Wert-Paare für den Distributionsmandanten hinzu. Weitere Informationen finden Sie unter Kennzeichnen Sie eine Distribution.
6. Wählen Sie Weiter aus.
7. Wählen Sie auf der Seite Domänen hinzufügen für Zertifikat aus, ob Sie ein benutzerdefiniertes TLS-Zertifikat für Ihren Distributionsmandanten wünschen. Das Zertifikat bestätigt, ob Sie berechtigt sind, den Domainnamen zu verwenden. Das Zertifikat muss in der Region USA Ost (Nord-Virginia) existieren.
8. Geben Sie für Domains Ihren Domainnamen ein.

> **Note**
>
> Wenn Sie einen Domainnamen eingegeben haben, der nicht durch ein Zertifikat abgedeckt ist, müssen Sie überprüfen, ob Sie Eigentümer der Domain sind. Sie können weiterhin vorerst den Verteilungsmandanten erstellen und den Domainbesitz später überprüfen. Weitere Informationen finden Sie unter Fordern Sie Zertifikate für Ihren CloudFront Distributionsmandanten an.

9. Wählen Sie Weiter aus.
10. Auf der Seite „Parameter definieren“ werden die Parameter angezeigt, die Sie in der Mehrmandantenverteilung angegeben haben. Geben Sie für die erforderlichen Parameter einen Wert neben dem Parameternamen ein und speichern Sie Ihre Änderungen.
11. Um einen weiteren Parameter hinzuzufügen, wählen Sie Parameter hinzufügen und geben Sie einen Namen und einen Wert ein.
12. Wählen Sie Weiter aus.
13. (Optional) Wenn Sie sich für Sicherheitsanpassungen entscheiden, Verteilungseinstellungen außer Kraft setzen, wählen Sie die Option für Ihren Anwendungsfall aus.
14. (Optional) Wenn Sie Verteilungseinstellungen überschreiben möchten, wählen Sie für die Anpassung geografischer Einschränkungen den entsprechenden Einschränkungstyp und die entsprechenden Länder für den Verteilungsmandanten aus. Weitere Informationen finden Sie unter Beschränken Sie die geografische Verteilung Ihrer Inhalte.
15. Wählen Sie Weiter aus.
16. Wählen Sie Verteilungsmandant erstellen aus.

Sie finden alle Ihre Distributionsmandanten auf der Seite Distributionsmandanten. Sie können nach folgenden Kriterien filtern:

Zuordnung

- Verteilungs-ID
- Zertifikat-ID
- ID der Verbindungsgruppe
- Web-ACL-ID

Eigenschaften

- Name
- Domain

Sie können Ihre Distributionsmandanten bearbeiten, um bestimmte Einstellungen anzupassen. Weitere Informationen finden Sie unter Anpassungen des Distributionsmandanten.

Standard

Um eine Standardverteilung zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Distributionen und dann Distribution erstellen aus.
3. Geben Sie einen Distributionsnamen für die Standarddistribution ein. Der Name wird als Wert für den `Name` Schlüssel als Tag angezeigt. Sie können diesen Wert später ändern. Sie können bis zu 50 Tags für Ihre Standarddistribution hinzufügen. Weitere Informationen finden Sie unter Kennzeichnen Sie eine Distribution.
4. Wählen Sie Einzelne Website oder App, Weiter.
5. (Optional) Geben Sie für das Domain-Setup eine Domain ein, die bereits bei Route 53 registriert ist AWS-Konto, in Ihre oder registrieren Sie eine neue Domain. Schließen Sie die Einrichtungsschritte ab.

   - Wenn Ihre Domain einen anderen DNS-Anbieter als Route 53 verwendet, können Sie die Domain trotzdem hinzufügen, müssen dies jedoch tun, nachdem Sie die Distribution erstellt haben. Überspringen Sie vorerst das Domain-Setup, um mit der Erstellung der Distribution

fortzufahren. Sie müssen die Domain und das TLS-Zertifikat später manuell konfigurieren. Weitere Informationen finden Sie unter Fügen Sie Ihrer CloudFront Standarddistribution eine Domain hinzu.

6. Wählen Sie Weiter aus.
7. Wählen Sie auf der Seite „Herkunft angeben“ den Herkunftstyp aus, von dem Ihre Inhalte abgerufen CloudFront werden sollen. CloudFront verwendet die empfohlenen Einstellungen für diesen Herkunftstyp für Ihre Distribution. Weitere Informationen zu den empfohlenen Einstellungen finden Sie unterReferenz für vorkonfigurierte Verteilungseinstellungen.
8. Wählen Sie für Origin Ihren Ursprung aus, oder geben Sie ihn ein.
9. Wähle unter Einstellungen eine der folgenden Optionen:

   - Empfohlene Origin-Einstellungen verwenden — Verwenden Sie die empfohlenen Standard-Cache- und Origin-Einstellungen für den ausgewählten Origin-Typ.
   - Origin-Einstellungen anpassen — Passen Sie die Cache- und Origin-Einstellungen an. Wenn Sie diese Option wählen, geben Sie Ihre eigenen Werte an.

10. Wählen Sie Weiter aus.
11. Wählen Sie auf der Seite Sicherheitsvorkehrungen aktivieren aus, ob der AWS WAF Sicherheitsschutz aktiviert werden soll.
12. Wählen Sie Weiter aus.
13. (Optional) Wenn Sie Route 53 für Ihre Domain verwenden, wird die Seite mit dem TLS-Zertifikat angezeigt. Wenn CloudFront Sie in Ihrem kein vorhandenes AWS Certificate Manager (ACM-) Zertifikat für Ihre Domain finden können `us-east-1` AWS-Region, können Sie wählen, ob Sie ein Zertifikat automatisch oder manuell erstellen möchten. AWS-Konto Nachdem das Zertifikat erstellt wurde, wählen Sie Weiter.
14. Überprüfen Sie Ihre Vertriebsdetails und wählen Sie Verteilung erstellen aus.
15. Nachdem Sie Ihre Verteilung CloudFront erstellt haben, ändert sich der Wert in der Spalte Status für Ihre Verteilung von Bereitstellen auf das Datum und die Uhrzeit der Bereitstellung der Verteilung.

    Der Domainname, der Ihrer Distribution CloudFront zugewiesen wurde, wird in der Liste der Distributionen angezeigt. (Der Status wird auch auf der Registerkarte General für die ausgewählte Verteilung angezeigt.)

> **Tip**
>
> Sie können anstelle des Namens, der Ihnen von zugewiesen wurde, einen alternativen Domainnamen verwenden CloudFront, indem Sie die Schritte unter befolgen. Verwenden Sie Benutzerdefiniert, URLs indem Sie alternative Domainnamen hinzufügen (CNAMEs)

16. Stellen Sie nach der Bereitstellung Ihrer Distribution sicher, dass Sie mit Ihrer neuen CloudFront URL (d111111abcdef8.cloudfront.net) oder dem CNAME auf Ihre Inhalte zugreifen können. Weitere Informationen finden Sie unter Testen Sie eine Distribution.
17. Achten Sie darauf, Ihre DNS-Einträge so zu aktualisieren, dass sie darauf hinweisen, wann Sie bereit sind, Traffic an Ihre Distribution zu senden CloudFront . Weitere Informationen finden Sie unter Verweisen Sie Domains auf CloudFront (Standardverteilung).

Werte, die in der Konsole CloudFront angezeigt werden

Wenn Sie eine neue Distribution erstellen oder eine bestehende Distribution aktualisieren, CloudFront werden die folgenden Informationen in der CloudFront Konsole angezeigt.

> **Note**
>
> Aktive vertrauenswürdige Unterzeichner, AWS-Konten die über ein aktives CloudFront key pair verfügen und verwendet werden können, um gültige signierte Unterzeichner zu erstellen URLs, sind derzeit in der CloudFront Konsole nicht sichtbar.

Verteilungs-ID

Wenn Sie mithilfe der CloudFront API eine Aktion für eine Distribution ausführen, verwenden Sie die Verteilungs-ID, um anzugeben, welche Distribution verwendet werden soll, z. B. `EDFDVBD6EXAMPLE` Sie können die Verteilungs-ID einer Verteilung nicht ändern.

Bereitstellung und Status

Wenn Sie eine Distribution bereitstellen, wird der Status Bereitgestellt in der Spalte Letzte Änderung angezeigt. Warten Sie, bis die Bereitstellung der Verteilung abgeschlossen ist, und stellen Sie sicher,

dass in der Statusspalte Aktiviert angezeigt wird. Weitere Informationen finden Sie unter Status der Verteilung.

Letzte Änderung

Datum und Uhrzeit der letzten Änderung der Verteilung im ISO 8601-Format, z. B. 2012-05-19T19:37:58Z. Weitere Informationen finden Sie unter https://www.w3.org/TR/NOTE-datetime.

Domainname

Sie verwenden den Domänennamen der Verteilung in den Links zu Ihren Objekten. Wenn beispielsweise der Domänenname der Verteilung `d111111abcdef8.cloudfront.net` ist, würde der Link zu `/images/image.jpg` dann `https://d111111abcdef8.cloudfront.net/images/image.jpg` lauten. Sie können den CloudFront -Domänennamen für Ihre Verteilung nicht ändern. Weitere Informationen zu Links CloudFront URLs zu Ihren Objekten finden Sie unterPassen Sie das URL-Format für Dateien an in CloudFront.

Wenn Sie einen oder mehrere alternative Domänennamen (CNAMEs) angegeben haben, können Sie anstelle des Domänennamens Ihre eigenen Domänennamen für Links zu Ihren Objekten verwenden. CloudFront Weitere Informationen zu finden CNAMEs Sie unterAlternative Domainnamen (CNAMEs).

> **Note**
>
> CloudFront Domainnamen sind einzigartig. Der Domänenname Ihrer Verteilung ist zuvor niemals für eine Verteilung verwendet worden und wird zukünftig für keine andere Verteilung verwendet werden.

Zusätzliche Links

Weitere Informationen zum Erstellen einer Distribution finden Sie unter den folgenden Links.

- Informationen zum Erstellen einer Distribution, die einen Amazon Simple Storage Service (Amazon S3) -Bucket-Ursprung mit Origin Access Control (OAC) verwendet, finden Sie unterBeginnen Sie mit einer CloudFront Standarddistribution.
- Informationen zur Verwendung von CloudFront APIs zum Erstellen einer Distribution finden Sie CreateDistributionin der Amazon CloudFront API-Referenz.

- Informationen zur Aktualisierung einer Distribution (z. B. zum Hinzufügen von Cache-Verhalten zu Standardverteilungen oder zum Anpassen von Distributionsmandanten) finden Sie unterEine Verteilung aktualisieren.
- Informationen zu den aktuellen Höchstwerten für die Anzahl der Verteilungen, die Sie für jedes AWS -Konto erstellen können, oder zum Anfordern eines höheren Kontingents (früher als Limit bezeichnet) finden Sie unter Allgemeine Kontingente für Verteilungen.

Fügen Sie Ihrer CloudFront Standarddistribution eine Domain hinzu

Nachdem Sie eine neue CloudFront Standarddistribution erstellt haben, können Sie ihr eine Domain hinzufügen. Optional können Sie bei der Erstellung eine Amazon Route 53-Domain für Ihre Standarddistribution einrichten. Weitere Informationen finden Sie unter Erstellen Sie eine CloudFront Distribution in der Konsole.

Fügen Sie Ihrer bestehenden Standarddistribution eine Domain hinzu

Um Ihrer Standarddistribution eine Domain hinzuzufügen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Distributionen und dann die Distributions-ID aus.
3. Wählen Sie unter Einstellungen, Alternative Domainnamen die Option Domain hinzufügen aus.
4. Geben Sie bis zu fünf Domains ein, die bedient werden sollen.
5. Wählen Sie Weiter aus.
6. Wenn CloudFront Sie für ein TLS-Zertifikat kein vorhandenes AWS Certificate Manager (ACM) Zertifikat für Ihre Domain AWS-Konto in Ihrem finden `us-east-1` AWS-Region, können Sie eines erstellen.

   - Wenn Sie Amazon Route 53 (Route 53) verwenden, CloudFront wird automatisch ein Zertifikat für Sie erstellt.
7. Wenn Ihr Zertifikat bereitgestellt wird, müssen Sie Ihre DNS-Einträge bei Ihrem DNS-Anbieter aktualisieren, um die Inhaberschaft der Domain nachzuweisen. Wählen Sie dann Zertifikat validieren aus. Weitere Informationen finden Sie unter Verweisen Sie Domains auf CloudFront (Standardverteilung).

   - Wenn Sie Route 53 verwenden, CloudFront aktualisiert Ihre DNS-Einträge für Sie.

8. Wählen Sie Weiter aus.
9. Überprüfen Sie Ihre Änderungen und wählen Sie Domains hinzufügen.
10. Bevor Sie Traffic an Ihre Distribution senden, stellen Sie sicher, dass Sie Ihre DNS-Einträge aktualisieren, auf die verwiesen wird CloudFront. Weitere Informationen erhalten Sie, wenn Sie auf der Seite mit den Vertriebsdetails CloudFront im Abschnitt Einstellungen die Option Domains weiterleiten an auswählen.

    - Wenn Sie Route 53 verwenden, können Sie das DNS-Routing automatisch für Sie CloudFront einrichten lassen.

Verweisen Sie Domains auf CloudFront (Standardverteilung)

Aktualisieren Sie Ihre DNS-Einträge, um den Verkehr von jeder Domain zum CloudFront Hostnamen weiterzuleiten. Sie können mehrere Domainnamen haben, aber sie müssen alle zu diesem Hostnamen führen.

Um Domains zu verweisen CloudFront

1. Kopieren Sie den CloudFront Hostnamenwert, z. B. d111111abcdef8.cloudfront.net.
2. Aktualisieren Sie Ihre DNS-Einträge, um den Verkehr von jeder Domain zum Hostnamen weiterzuleiten. CloudFront

    1. Melden Sie sich bei der Verwaltungskonsole Ihres Domain-Registrars oder DNS-Providers an.
    2. Navigieren Sie zum DNS-Verwaltungsbereich für Ihre Domain.
        - Für Subdomains — Erstellen Sie einen CNAME-Eintrag. Zum Beispiel:
            - Name — Ihre Subdomain (wie oder) `www` `app`
            - Value//Target — Ihr Hostname CloudFront
            - Datensatztyp — CNAME
            - TTL — 3600 (oder was auch immer für Ihren Anwendungsfall geeignet ist)
        - Für apex/root Domains — Dies erfordert eine eindeutige DNS-Konfiguration, da CNAME-Standardeinträge nicht auf Stamm- oder Apex-Domänenebene verwendet werden können. Da die meisten DNS-Anbieter ALIAS-Einträge nicht unterstützen, empfehlen wir, einen ALIAS-Eintrag in Route 53 zu erstellen. Zum Beispiel:
            - Name — Ihre Apex-Domain (z. B.`example.com`)
            - Datensatztyp — A

- Alias — Ja
- Alias-Ziel — Ihr CloudFront Hostname
- Routing-Richtlinie — Einfach (oder was auch immer für Ihren Anwendungsfall geeignet ist)

3. Stellen Sie sicher, dass die DNS-Änderung weitergegeben wurde. (Dies passiert normalerweise, wenn die TTL abgelaufen ist. Manchmal kann es 24-48 Stunden dauern.) Verwenden Sie ein Tool wie `dig` oder`nslookup`.

```
dig www.example.com
# Should eventually return a CNAME pointing to your CloudFront hostname
```

3. Kehren Sie zur CloudFront Konsole zurück und wählen Sie Submit. Wenn Ihre Domain aktiv ist, CloudFront aktualisiert sie den Domain-Status, um anzuzeigen, dass Ihre Domain bereit ist, Traffic bereitzustellen.

Weitere Informationen finden Sie in der Dokumentation Ihres DNS-Anbieters:

- Cloudflare
- ClouDNS
- DNSimple
- Gandi.net
- GoDaddy
- Google Cloud-DNS
- Nennen Sie billig

Referenz für vorkonfigurierte Verteilungseinstellungen

Wenn Sie Ihre CloudFront Distribution erstellen, CloudFront werden die meisten Distributionseinstellungen automatisch für Sie konfiguriert, basierend auf Ihrem Inhaltstyp. Optional können Sie Ihre Vertriebseinstellungen manuell bearbeiten. Weitere Informationen finden Sie unter Referenz für alle Verteilungseinstellungen.

In den folgenden Abschnitten werden die standardmäßigen Vorkonfigurationseinstellungen für Distributionen und die Einstellungen beschrieben, die Sie anpassen können.

Amazon S3 S3-Ursprung

Im Folgenden finden Sie die Origin-Einstellungen, die für Ihren Amazon S3 S3-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Origin-Einstellungen (vorkonfiguriert)

- Origin Access Control (nur Konsole) — CloudFront richtet das für dich ein. CloudFront versucht, die S3-Bucket-Richtlinie für Standardverteilungen und für Multi-Tenant-Distributionen hinzuzufügen, bei denen in der Ursprungsdomäne keine Parameter verwendet werden.
- Benutzerdefinierten Header hinzufügen — Keine
- Origin Shield aktivieren — Nein
- Verbindungsversuche — 3

Im Folgenden finden Sie die Cache-Einstellungen, die für Ihren Amazon S3 S3-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Cache-Einstellungen (vorkonfiguriert)

- Objekte automatisch komprimieren — Ja
- Viewer-Protokollrichtlinie — Zu HTTPS weiterleiten
- Zulässige HTTP-Methode — `GET, HEAD`
- Zuschauerzugriff einschränken — Nein
- Cache-Richtlinie — `CachingOptimized`
- Richtlinie für Origin-Anfragen — Keine
- Richtlinie für den Antwort-Header — Keine
- Reibungsloses Streaming — Nein
- Verschlüsselung auf Feldebene — Nein
- Echtzeitprotokolle aktivieren — Nein
- Funktionen — Nein

Im Folgenden finden Sie die Einstellungen, die Sie für Ihren Amazon S3 S3-Ursprung in einer Multi-Tenant-Distribution anpassen können.

Individuell anpassbare Einstellungen

- S3-Zugriff — CloudFront legt dies für Sie fest, basierend auf Ihren S3-Bucket-Einstellungen:
  - Wenn dein Bucket öffentlich ist, ist keine Origin Access Control (OAC) -Richtlinie erforderlich.
  - Wenn Ihr Bucket privat ist — Sie können eine zu verwendende OAC-Richtlinie auswählen oder erstellen.
- Origin Shield aktivieren — Nein
- Objekte automatisch komprimieren — Ja
  - Wenn Sie Ja wählen, wird die `CachingOptimized` Caching-Richtlinie verwendet.
  - Wenn Sie Nein wählen, wird die `CachingOptimizedForUncompressedObjects` Caching-Richtlinie verwendet.

Herkunft des API Gateway

Im Folgenden finden Sie die Ursprungseinstellungen, die für Ihren API-Gateway-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Origin-Einstellungen (vorkonfiguriert)

- Protokoll — Nur HTTPS
- HTTPS-Port — 443
- Minimaler Ursprung des SSL-Protokolls — TLSv1 2.
- Herkunftspfad — Keiner
- Origin Access Control (nur Konsole) — CloudFront richtet das für dich ein
- Benutzerdefinierten Header hinzufügen — Keine
- Origin Shield aktivieren — Nein
- Verbindungsversuche — 3
- Timeout für die Antwort — 30
- Keep-Alive-Timeout — 5

Im Folgenden finden Sie die Cache-Einstellungen, die für Ihren API-Gateway-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Cache-Einstellungen (vorkonfiguriert)

- Objekte automatisch komprimieren — Ja
- Viewer-Protokollrichtlinie — Zu HTTPS weiterleiten
- Zulässige HTTP-Methode — `GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE`
- HTTP-Methoden zwischenspeichern — Nein
- gRPC-Anfragen über HTTP/2 zulassen — Nein
- Zuschauerzugriff einschränken — Nein
- Cache-Richtlinie — `CachingDisabled` (Mögliche Werte:`UseOriginCacheControlHeaders`,`UseOriginCacheControlHeaders-QueryStrings`)
- Origin-Anforderungsrichtlinie — `AllViewerExceptHostHeader` (Mögliche Werte:`AllViewer`,`AllViewerandCloudFrontHeaders-2022-06`)
- Richtlinie für den Antwort-Header — Keine
- Reibungsloses Streaming — Nein
- Verschlüsselung auf Feldebene — Nein
- Echtzeitprotokolle aktivieren — Nein
- Funktionen — Nein

Im Folgenden finden Sie die Einstellungen, die Sie für Ihren API-Gateway-Ursprung in einer Multi-Tenant-Distribution anpassen können.

Individuell anpassbare Einstellungen

- Origin Shield aktivieren — (Standardeinstellung: Nein)
- Objekte automatisch komprimieren — (Standard: Ja)

Benutzerdefinierter Ursprung und EC2 Instanz

Im Folgenden finden Sie die Origin-Einstellungen, die für Ihren benutzerdefinierten Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Origin-Einstellungen (vorkonfiguriert)

- Protokoll — Match Viewer

- HTTP-Port — 80
- HTTPS-Port — 443
- Minimaler Ursprung des SSL-Protokolls — TLSv1 2.
- Herkunftspfad — Keiner
- Benutzerdefinierten Header hinzufügen — Keine
- Origin Shield aktivieren — Nein
- Verbindungsversuche — 3
- Timeout für die Antwort — 30
- Keep-Alive-Timeout — 5

Im Folgenden finden Sie die Cache-Einstellungen, die für Ihren benutzerdefinierten Ursprung und Ihre benutzerdefinierte EC2 Instanz in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Cache-Einstellungen (vorkonfiguriert)

- Objekte automatisch komprimieren — Ja
- Viewer-Protokollrichtlinie — Zu HTTPS weiterleiten
- Zulässige HTTP-Methode — `GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE`
- HTTP-Methoden zwischenspeichern — Nein
- gRPC-Anfragen über HTTP/2 zulassen — Nein
- Zuschauerzugriff einschränken — Nein
- Cache-Richtlinie — `UseOriginCacheControlHeaders` (Mögliche Werte:`UseOriginCacheControlHeaders-QueryStrings`,`CachingDisabled`,`CacheOptimized`,`CachingOptimizedForUncompressedObjec
- Origin-Anforderungsrichtlinie — `AllViewer` (Mögliche Werte:`AllViewerExceptHostHeader`,`AllViewerandCloudFrontHeaders-2022-06`)
- Richtlinie für den Antwort-Header — Keine
- Reibungsloses Streaming — Nein
- Verschlüsselung auf Feldebene — Nein
- Echtzeitprotokolle aktivieren — Nein
- Funktionen — Nein

Im Folgenden finden Sie die Einstellungen, die Sie für Ihren benutzerdefinierten Ursprung und Ihre benutzerdefinierte EC2 Instanz in einer Multi-Tenant-Distribution anpassen können.

Individuell anpassbare Einstellungen

- Origin Shield aktivieren — (Standardeinstellung: Nein)
- Objekte automatisch komprimieren — (Standard: Ja)
- Zwischenspeichern — (Standard:`Cache by Default`)
  - Wenn ausgewählt, `Cache by Default` wird die `UseOriginCacheControlHeaders` Cache-Richtlinie verwendet.
  - Wenn ausgewählt, `Do Not Cache by Default` wird die `CachingDisabled` Cache-Richtlinie verwendet.
- Abfragezeichenfolge in den Cache einbeziehen — (Standard: Ja, falls `Cache by Default` bereits ausgewählt)
  - Wenn die Option bereits ausgewählt `Do Not Cache by Default` ist und Sie sich dann dafür entscheiden, die Abfragezeichenfolge in den Cache aufzunehmen, wird die `UseOriginCacheControlHeaders-QueryStrings` Cache-Richtlinie verwendet.

Ursprung von Elastic Load Balancing

Im Folgenden finden Sie die Origin-Einstellungen, die für Ihren Elastic Load Balancing Balancing-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Origin-Einstellungen (vorkonfiguriert)

- Protokoll — Nur HTTPS
- HTTPS-Port — 443
- Minimaler Ursprung des SSL-Protokolls — TLSv1 2.
- Herkunftspfad — Keiner
- Benutzerdefinierten Header hinzufügen — Keine
- Origin Shield aktivieren — Nein
- Verbindungsversuche — 3
- Timeout für die Antwort — 30
- Keep-Alive-Timeout — 5

Im Folgenden finden Sie die Cache-Einstellungen, die für Ihren Elastic Load Balancing Balancing-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Cache-Einstellungen (vorkonfiguriert)

- Objekte automatisch komprimieren — Ja
- Viewer-Protokollrichtlinie — Zu HTTPS weiterleiten
- Zulässige HTTP-Methode — `GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE`
- HTTP-Methoden zwischenspeichern — Nein
- gRPC-Anfragen über HTTP/2 zulassen — Nein
- Zuschauerzugriff einschränken — Nein
- Caching — (Standard:`Cache by Default`)
  - Wenn ausgewählt, `Cache by Default` wird die `UseOriginCacheControlHeaders` Cache-Richtlinie verwendet.
  - Wenn ausgewählt, `Do Not Cache by Default` wird die `CachingDisabled` Cache-Richtlinie verwendet.
- Abfragezeichenfolge in den Cache einbeziehen — (Standard: Ja, falls `Cache by Default` bereits ausgewählt)
  - Wenn die Option bereits ausgewählt `Do Not Cache by Default` ist und Sie sich dann dafür entscheiden, die Abfragezeichenfolge in den Cache aufzunehmen, wird die `UseOriginCacheControlHeaders-QueryStrings` Cache-Richtlinie verwendet.
- Origin-Anforderungsrichtlinie — `All Viewer` (Mögliche Werte:`AllViewerExceptHostHeader`,`AllViewerandCloudFrontHeaders-2022-06`)
- Richtlinie für den Antwort-Header — Keine
- Reibungsloses Streaming — Nein
- Verschlüsselung auf Feldebene — Nein
- Echtzeitprotokolle aktivieren — Nein
- Funktionen — Nein

Im Folgenden finden Sie die Einstellungen, die Sie für Ihren Elastic Load Balancing Balancing-Ursprung in einer Multi-Tenant-Distribution anpassen können.

Individuell anpassbare Einstellungen

- Origin Shield aktivieren — (Standardeinstellung: Nein)
- Objekte automatisch komprimieren — (Standard: Ja)
- Zwischenspeichern — (Standard:`Cache by Default`)
  - Wenn ausgewählt, `Cache by Default` wird die `UseOriginCacheControlHeaders` Cache-Richtlinie verwendet.
  - Wenn ausgewählt, `Do Not Cache by Default` wird die `CachingDisabled` Cache-Richtlinie verwendet.
- Abfragezeichenfolge in den Cache einbeziehen — (Standard: Ja, falls `Cache by Default` bereits ausgewählt)
  - Wenn die Option bereits ausgewählt `Do Not Cache by Default` ist und Sie sich dann dafür entscheiden, die Abfragezeichenfolge in den Cache aufzunehmen, wird die `UseOriginCacheControlHeaders-QueryStrings` Cache-Richtlinie verwendet.

URL-Ursprung der Lambda-Funktion

Im Folgenden finden Sie die Ursprungseinstellungen, mit denen der URL-Ursprung Ihrer Lambda-Funktion in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert ist.

Origin-Einstellungen (vorkonfiguriert)

- Origin Access Control — CloudFront richtet das für dich ein und fügt die Richtlinie hinzu
- Protokoll — Nur HTTPS
- HTTPS-Port — 443
- Minimaler Ursprung des SSL-Protokolls — TLSv1 2.
- Herkunftspfad — Keiner
- Benutzerdefinierten Header hinzufügen — Keine
- Origin Shield aktivieren — Nein
- Verbindungsversuche — 3
- Timeout für die Antwort — 30
- Keep-Alive-Timeout — 5

Im Folgenden finden Sie die Cache-Einstellungen, die für den URL-Ursprung Ihrer Lambda-Funktion in einer Mehrmandantenverteilung CloudFront vorkonfiguriert sind.

Cache-Einstellungen (vorkonfiguriert)

- Objekte automatisch komprimieren — Ja
- Viewer-Protokollrichtlinie — Zu HTTPS weiterleiten
- Zulässige HTTP-Methode — `GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE`
- HTTP-Methoden zwischenspeichern — Nein
- gRPC-Anfragen über HTTP/2 zulassen — Nein
- Zuschauerzugriff einschränken — Nein
- Cache-Richtlinie — `CachingDisabled` (Mögliche Werte:`UseOriginCacheControlHeaders`,`UseOriginCacheControlHeaders-QueryStrings`)
- Richtlinie für Origin-Anfragen — `AllViewerExceptHostHeader`
- Richtlinie für den Antwort-Header — Keine
- Reibungsloses Streaming — Nein
- Verschlüsselung auf Feldebene — Nein
- Echtzeitprotokolle aktivieren — Nein
- Funktionen — Nein

Im Folgenden finden Sie die Einstellungen, die Sie für den URL-Ursprung Ihrer Lambda-Funktion in einer Multi-Tenant-Distribution anpassen können.

Individuell anpassbare Einstellungen

- Origin Shield aktivieren — (Standardeinstellung: Nein)
- Objekte automatisch komprimieren — (Standard: Ja)
- Zwischenspeichern — (Standard:`Cache by Default`)
  - Wenn ausgewählt, `Cache by Default` wird die `UseOriginCacheControlHeaders` Cache-Richtlinie verwendet.
  - Wenn ausgewählt, `Do Not Cache by Default` wird die `CachingDisabled` Cache-Richtlinie verwendet.

- Abfragezeichenfolge in den Cache einbeziehen — (Standard: Ja, falls `Cache by Default` bereits ausgewählt)
  - Wenn die Option bereits ausgewählt `Do Not Cache by Default` ist und Sie sich dann dafür entscheiden, die Abfragezeichenfolge in den Cache aufzunehmen, wird die `UseOriginCacheControlHeaders-QueryStrings` Cache-Richtlinie verwendet.

MediaPackage v1-Ursprung

Im Folgenden finden Sie die Origin-Einstellungen, die für Ihren MediaPackage v1-Origin in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Origin-Einstellungen (vorkonfiguriert)

- Protokoll — Nur HTTPS
- HTTPS-Port — 443
- Minimaler Ursprung des SSL-Protokolls — TLSv1 2.
- Herkunftspfad — Sie geben diesen Pfad an, indem Sie Ihre MediaPackage URL eingeben.
- Benutzerdefinierten Header hinzufügen — Keine
- Origin Shield aktivieren — Nein
- Verbindungsversuche — 3
- Timeout für die Antwort — 30
- Keep-Alive-Timeout — 5

Im Folgenden finden Sie die Cache-Einstellungen, die für Ihren MediaPackage v1-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Cache-Einstellungen (vorkonfiguriert)

- Objekte automatisch komprimieren — Ja
- Viewer-Protokollrichtlinie — Zu HTTPS weiterleiten
- Zulässige HTTP-Methode — `GET, HEAD`
- HTTP-Methoden zwischenspeichern — Nein
- gRPC-Anfragen über HTTP/2 zulassen — Nein
- Zuschauerzugriff einschränken — Nein

- Cache-Richtlinie — `Elemental-MediaPackage`
- Richtlinie für Origin-Anfragen — Keine
- Richtlinie für den Antwort-Header — Keine
- Reibungsloses Streaming — Nein
- Verschlüsselung auf Feldebene — Nein
- Echtzeitprotokolle aktivieren — Nein
- Funktionen — Nein

MediaPackage v2-Ursprung

Im Folgenden finden Sie die Origin-Einstellungen, die für Ihren MediaPackage v2-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Origin-Einstellungen (vorkonfiguriert)

- Origin Access Control — CloudFront richtet das für dich ein und fügt die Richtlinie hinzu
- Protokoll — Nur HTTPS
- HTTPS-Port — 443
- Minimaler Ursprung des SSL-Protokolls — TLSv1 2.
- Herkunftspfad — Keiner
- Benutzerdefinierten Header hinzufügen — Keine
- Origin Shield aktivieren — Nein
- Verbindungsversuche — 3
- Timeout für die Antwort — 30
- Keep-Alive-Timeout — 5

Im Folgenden finden Sie die Cache-Einstellungen, die für Ihren MediaPackage v2-Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Cache-Einstellungen (vorkonfiguriert)

- Objekte automatisch komprimieren — Ja
- Viewer-Protokollrichtlinie — Zu HTTPS weiterleiten
- Zulässige HTTP-Methode — `GET, HEAD`

- HTTP-Methoden zwischenspeichern — Nein
- gRPC-Anfragen über HTTP/2 zulassen — Nein
- Zuschauerzugriff einschränken — Nein
- Cache-Richtlinie — `Elemental-MediaPackage`
- Richtlinie für Origin-Anfragen — Keine
- Richtlinie für den Antwort-Header — Keine
- Reibungsloses Streaming — Nein
- Verschlüsselung auf Feldebene — Nein
- Echtzeitprotokolle aktivieren — Nein
- Funktionen — Nein

MediaTailor Herkunft

Im Folgenden finden Sie die Origin-Einstellungen, die für Ihren MediaTailor Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Origin-Einstellungen (vorkonfiguriert)

- Protokoll — Nur HTTPS
- HTTPS-Port — 443
- Minimaler Ursprung des SSL-Protokolls — TLSv1 2.
- Herkunftspfad — Sie geben diesen Pfad an, indem Sie Ihre MediaPackage URL eingeben.
- Benutzerdefinierten Header hinzufügen — Keine
- Origin Shield aktivieren — Nein
- Verbindungsversuche — 3
- Timeout für die Antwort — 30
- Keep-Alive-Timeout — 5

Im Folgenden finden Sie die Cache-Einstellungen, die für Ihren MediaTailor Ursprung in einer Multi-Tenant-Distribution CloudFront vorkonfiguriert sind.

Cache-Einstellungen (vorkonfiguriert)

- Objekte automatisch komprimieren — Ja

- Viewer-Protokollrichtlinie — Zu HTTPS weiterleiten
- Zulässige HTTP-Methode — `GET, HEAD`
- HTTP-Methoden zwischenspeichern — Nein
- gRPC-Anfragen über HTTP/2 zulassen — Nein
- Zuschauerzugriff einschränken — Nein
- Cache-Richtlinie — Keine
- Richtlinie für Origin-Anfragen — `Elemental-MediaTailor-PersonalizedManifests`
- Richtlinie für den Antwort-Header — Keine
- Reibungsloses Streaming — Nein
- Verschlüsselung auf Feldebene — Nein
- Echtzeitprotokolle aktivieren — Nein
- Funktionen — Nein

Referenz für alle Verteilungseinstellungen

Sie können wählen, ob Sie Ihre CloudFront Vertriebseinstellungen manuell bearbeiten möchten, wenn Sie Ihre Distribution erstellen oder aktualisieren. Im Folgenden finden Sie die Einstellungen, die Sie bearbeiten können.

CloudFront Konfiguriert jedoch die meisten Verteilungseinstellungen für Sie, basierend auf der Herkunft Ihres Inhalts. Weitere Informationen finden Sie unter Referenz für vorkonfigurierte Verteilungseinstellungen.

Weitere Informationen zum Erstellen oder Aktualisieren einer Verteilung mithilfe der CloudFront-Konsole finden Sie unter the section called “Eine Verteilung erstellen” oder the section called “Eine Verteilung aktualisieren”.

Themen

- Ursprungseinstellungen
- Einstellungen für das Cache-Verhalten
- Distribution Settings (Einstellungen für die Verteilung)
- Benutzerdefinierte Fehlerseiten und Zwischenspeicherung von Fehlern
- Geografische Einschränkungen

Ursprungseinstellungen

Wenn Sie die CloudFront Konsole verwenden, um eine Distribution zu erstellen oder zu aktualisieren, geben Sie Informationen zu einem oder mehreren Speicherorten, den sogenannten Origins, an denen Sie die Originalversionen Ihrer Webinhalte speichern, an. CloudFront ruft Ihre Webinhalte von Ihren Ursprüngen ab und stellt sie den Zuschauern über ein weltweites Netzwerk von Edge-Servern zur Verfügung.

Informationen zur aktuell gültigen maximalen Anzahl von Ursprüngen, die Sie für eine Verteilung erstellen können, oder zum Anfordern eines höheren Kontingents finden Sie unter the section called “Allgemeine Kontingente für Verteilungen”.

Wenn Sie einen Ursprung löschen möchten, müssen Sie zunächst die Cache-Verhalten bearbeiten oder löschen, die mit diesem Ursprung verknüpft sind.

> **Important**
>
> Vergewissern Sie sich beim Löschen eines Ursprungs, dass die zuvor von diesem Ursprung bereitgestellten Dateien in einem anderen Ursprung verfügbar sind und dass Ihre Cache-Verhalten Anfragen für diese Dateien jetzt an den neuen Ursprung weiterleiten.

Beim Erstellen oder Aktualisieren einer Verteilung geben Sie die folgenden Werte für jeden Ursprung an.

Themen

- Timeout bei Abschluss der Antwort
- Keep-Alive-Timeout (nur benutzerdefinierte und VPC-Ursprünge)
- Quoten für Antwort- und Keep-Alive-Timeouts

Ursprungsdomäne

Die Ursprungsdomain ist der DNS-Domainname der Ressource, von der Objekte für Ihren Ursprung abgerufen CloudFront werden, z. B. ein Amazon S3 S3-Bucket oder ein HTTP-Server. Zum Beispiel:

- Amazon S3-Bucket – *amzn-s3-demo-bucket*.s3.*us-west-2*.amazonaws.com

> **Note**
>
> Wenn Sie den S3-Bucket kürzlich erstellt haben, gibt die CloudFront Verteilung möglicherweise `HTTP 307 Temporary Redirect` Antworten für bis zu 24 Stunden zurück. Es kann bis zu 24 Stunden dauern, bis der S3-Bucket-Name in alle AWS -Regionen weitergegeben wird. Wenn die Weitergabe abgeschlossen ist, stoppt die Verteilung das Senden dieser Umleitungsantworten automatisch. Sie müssen keine Maßnahmen ergreifen. Weitere Informationen finden Sie unter Warum erhalte ich eine HTTP 307 Temporary Redirect-Antwort von Amazon S3? und Temporäre Anforderungsumleitung.

- Amazon S3-Bucket, der als Website konfiguriert ist – *amzn-s3-demo-bucket*.s3-website.*us-west-2*.amazonaws.com
- MediaStore Behälter — *examplemediastore*.data.mediastore.*us-west-1*.amazonaws.com
- MediaPackage Endpunkt — *examplemediapackage*.mediapackage.*us-west-1*.amazonaws.com
- EC2 Amazon-Instanz — *ec2-203-0-113-25*.compute-1.amazonaws.com
- Elastic Load Balancing-Load Balancer – *example-load-balancer-1234567890*.*us-west-2*.elb.amazonaws.com
- Ihr eigener Webserver – www.example.com

Wählen Sie den Domänennamen im Feld Origin domain (Ursprungsdomäne) aus oder geben Sie den Namen ein. Ressourcen aus Opt-in-Regionen müssen manuell eingegeben werden. Bei dem

Domänennamen wird die Groß- und Kleinschreibung nicht berücksichtigt. Ihre Ursprungsdomain muss über einen öffentlich auflösbaren DNS-Namen verfügen, der Anfragen von Clients über das Internet an Ziele weiterleitet.

Wenn Sie so konfigurieren, dass eine Verbindung CloudFront zu Ihrem Ursprung über HTTPS hergestellt wird, muss einer der Domainnamen im Zertifikat mit dem Domainnamen übereinstimmen, den Sie für Origin Domain Name angeben. Wenn kein Domainname übereinstimmt, wird der HTTP-Statuscode 502 (Bad Gateway) an den Betrachter CloudFront zurückgegeben. Weitere Informationen erhalten Sie unter Domainnamen in der CloudFront Distribution und im Zertifikat und Fehler bei der SSL/TLS-Aushandlung zwischen CloudFront und einem benutzerdefinierten Ursprungsserver.

> **Note**
>
> Wenn Sie eine Herkunftsanforderungsrichtlinie verwenden, die den Viewer-Host-Header an den Ursprung weiterleitet, muss der Ursprung mit einem Zertifikat antworten, das dem Viewer-Host-Header entspricht. Weitere Informationen finden Sie unter CloudFront Anforderungsheader hinzufügen.

Wenn es sich bei Ihrem Ursprung um einen Amazon S3-Bucket handelt, beachten Sie Folgendes:

- Wenn der Bucket als Website konfiguriert ist, geben Sie den statischen Amazon-S3-Endpunkt für das Hosten von Websites für Ihren Bucket ein. Sie dürfen den Bucket-Namen nicht aus der Liste im Feld Origin domain (Ursprungsdomäne) auswählen. Der statische Endpunkt für das Hosten von Websites wird in der Amazon-S3-Konsole auf der Seite Properties (Eigenschaften) unter Static Website Hosting (Hosting der statischen Website) angezeigt. Weitere Informationen finden Sie unter the section called “Verwenden Sie einen Amazon S3 S3-Bucket, der als Website-Endpunkt konfiguriert ist”.
- Wenn Sie für Ihren Bucket Amazon S3 Transfer Acceleration konfiguriert haben, geben Sie den `s3-accelerate`-Endpunkt für Origin domain (Ursprungsdomäne) nicht an.
- Wenn Sie einen Bucket aus einem anderen AWS Konto verwenden und der Bucket nicht als Website konfiguriert ist, geben Sie den Namen im folgenden Format ein:

  ```
  bucket-name.s3.region.amazonaws.com
  ```

  Wenn sich Ihr Bucket in einer US-Region befindet und Amazon S3 Anforderungen an einen Standort in Nord-Virginia weiterleiten soll, verwenden Sie das folgende Format:

bucket-name.s3.us-east-1.amazonaws.com

- Die Dateien müssen öffentlich lesbar sein, es sei denn, Sie sichern Ihre Inhalte in Amazon S3 mithilfe einer CloudFront Ursprungszugriffskontrolle. Weitere Informationen zur Zugriffskontrolle finden Sie unter the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”.

> **Important**
>
> Wenn der Ursprung ein Amazon S3-Bucket ist, muss der Bucket-Name den Anforderungen für DNS-Namen entsprechen. Weitere Informationen finden Sie unter Bucket-Einschränkungen und -Limits im Benutzerhandbuch zu Amazon Simple Storage Service.

Wenn Sie den Wert der Origin-Domain für einen Ursprung ändern, beginnt CloudFront sofort, die Änderung an den CloudFront Edge-Standorten zu replizieren. Leitet weiterhin Anfragen an den vorherigen Ursprung CloudFront weiter, bis die Verteilungskonfiguration an einem bestimmten Edge-Standort aktualisiert ist. Sobald die Verteilungskonfiguration an diesem Edge-Standort aktualisiert wurde, CloudFront beginnt die Weiterleitung von Anfragen an den neuen Ursprung.

Wenn Sie den Ursprung ändern, CloudFront müssen die Edge-Caches nicht erneut mit Objekten aus dem neuen Ursprung gefüllt werden. Solange die Anzeigeanfragen in Ihrer Anwendung nicht geändert wurden, stellt CloudFront weiterhin Objekte bereit, die sich bereits in einem Edge-Cache befinden, bis die TTL für jedes Objekt abläuft oder bis selten angefragte Objekte entfernt wurden.

Protokoll (nur benutzerdefinierte Ursprünge)

> **Note**
>
> Dies gilt nur für benutzerdefinierte Ursprünge.

Die Protokollrichtlinie, die Sie beim Abrufen CloudFront von Objekten von Ihrem Ursprung verwenden möchten.

Wählen Sie einen der folgenden Werte aus:

- Nur HTTP: CloudFront verwendet nur HTTP für den Zugriff auf den Ursprung.

> **Important**
>
> HTTP only (Nur HTTP) ist die Standardeinstellung, wenn der Ursprung ein statischer Amazon S3-Endpunkt für das Hosten von Websites ist, da Amazon S3 keine HTTPS-Verbindungen für statische Endpunkte für das Hosten von Websites unterstützt. Die CloudFront Konsole unterstützt das Ändern dieser Einstellung für statische Amazon S3 S3-Website-Hosting-Endpunkte nicht.

- Nur HTTPS: CloudFront verwendet nur HTTPS für den Zugriff auf den Ursprung.
- Match Viewer: CloudFront Kommuniziert je nach Protokoll der Viewer-Anfrage über HTTP oder HTTPS mit deinem Ursprung. CloudFront speichert das Objekt nur einmal im Cache, auch wenn Zuschauer Anfragen sowohl mit HTTP- als auch mit HTTPS-Protokollen stellen.

> **Important**
>
> Bei HTTPS-Viewer-Anfragen, CloudFront die an diesen Ursprung weiterleiten, muss einer der Domainnamen im SSL/TLS Zertifikat auf Ihrem Ursprungsserver mit dem Domainnamen übereinstimmen, den Sie für die Origin-Domain angeben. Andernfalls CloudFront reagiert auf die Viewer-Anfragen mit dem HTTP-Statuscode 502 (Bad Gateway), anstatt das angeforderte Objekt zurückzugeben. Weitere Informationen finden Sie unter the section called “Anforderungen für die Verwendung von SSL/TLS Zertifikaten mit CloudFront”.

Themen

HTTP-Port

> **Note**
>
> Dies gilt nur für benutzerdefinierte Ursprünge.

(Optional) Sie können den HTTP-Port angeben, den der benutzerdefinierte Ursprung überwacht. Zu den gültigen Werten gehören die Ports 80, 443 und 1024 bis 65535. Der Standardwert ist Port 80.

> **Important**
>
> Port 80 ist die Standardeinstellung, wenn der Ursprung ein statischer Amazon S3-Endpunkt für das Hosten von Websites ist, da Amazon S3 den Port 80 nur für statische Endpunkte für das Hosten von Websites unterstützt. Die CloudFront Konsole unterstützt das Ändern dieser Einstellung für statische Amazon S3 S3-Website-Hosting-Endpunkte nicht.

HTTPS-Port

> **Note**
>
> Dies gilt nur für benutzerdefinierte Ursprünge.

(Optional) Sie können den HTTPS-Port angeben, den der benutzerdefinierte Ursprung überwacht. Zu den gültigen Werten gehören die Ports 80, 443 und 1024 bis 65535. Der Standardwert ist Port 443. Wenn Protocol (Protokoll) auf HTTP only (Nur HTTP) festgelegt wird, können Sie keinen Wert für HTTPS port (HTTPS-Port) angeben.

Mindest-SSL-Protokoll für Ursprung

> **Note**
>
> Dies gilt nur für benutzerdefinierte Ursprünge.

Wählen Sie das TLS/SSL Mindestprotokoll aus, das verwendet CloudFront werden kann, wenn es eine HTTPS-Verbindung zu Ihrem Ursprung herstellt. Da niedrigere TLS-Protokollversionen weniger sicher sind, sollten Sie das neueste TLS-Protokoll auswählen, das Ihr Ursprung unterstützt. Wenn Protocol (Protokoll) auf HTTP only (Nur HTTP) festgelegt wird, können Sie keinen Wert für Minimum origin SSL protocol (SSL-Mindestursprungsprotokoll) angeben.

Wenn Sie die CloudFront API verwenden, um das CloudFront zu verwendende TLS/SSL Protokoll festzulegen, können Sie kein Mindestprotokoll festlegen. Stattdessen geben Sie alle TLS/SSL

Protokolle an, die Sie mit Ihrem Origin verwenden CloudFront können. Weitere Informationen finden Sie OriginSslProtocolsin der Amazon CloudFront API-Referenz.

Ursprungspfad

Wenn Sie Ihre Inhalte aus einem Verzeichnis in Ihrem Ursprungsverzeichnis anfordern möchten CloudFront , geben Sie den Verzeichnispfad ein, der mit einem Schrägstrich (/) beginnt. CloudFront hängt den Verzeichnispfad an den Wert der Origin-Domain an, zum Beispiel. **`cf-origin.example.com/production/images`** Fügen Sie keinen Schrägstrich (/) am Pfadende hinzu.

Angenommen, Sie haben die folgenden Werte für Ihre Verteilung angegeben:

- Origin domain (Ursprungsdomäne) – Ein Amazon-S3-Bucket mit dem Namen **`amzn-s3-demo-bucket`**
- Origin path (Ursprungspfad) – **`/production`**
- Alternate domain names (Alternative Domänennamen) (CNAME) – **`example.com`**

Wenn ein Benutzer einen Browser `example.com/index.html` aufruft, CloudFront sendet eine Anfrage an Amazon S3 für`amzn-s3-demo-bucket/production/index.html`.

Wenn ein Benutzer einen Browser `example.com/acme/index.html` aufruft, CloudFront sendet eine Anfrage an Amazon S3 für`amzn-s3-demo-bucket/production/acme/index.html`.

Name

Eine Zeichenfolge, die diesen Ursprung eindeutig in dieser Verteilung identifiziert. Wenn Sie zusätzlich zum Standard-Cache-Verhalten Cache-Verhalten erstellen, verwenden Sie den Namen, den Sie hier angeben, um den Ursprung zu identifizieren, an den Sie eine Anfrage weiterleiten möchten CloudFront , wenn die Anforderung dem Pfadmuster für dieses Cache-Verhalten entspricht.

Ursprungszugriff (nur Amazon-S3-Ursprünge)

> **Note**
>
> Dies gilt nur für Amazon S3-Bucket-Ursprünge (d. h. Ursprünge, die nicht den statischen S3-Website-Endpunkt verwenden).

Wählen Sie Origin-Zugriffskontrolleinstellungen (empfohlen), wenn Sie den Zugriff auf einen Amazon S3 S3-Bucket-Ursprung auf bestimmte CloudFront Distributionen beschränken möchten.

Wählen Sie Public (Öffentlich) aus, wenn der Amazon-S3-Bucket-Ursprung öffentlich zugänglich ist.

Weitere Informationen finden Sie unter the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”.

Informationen darüber, wie Sie Benutzern den Zugriff auf Objekte mit einem benutzerdefinierten Ursprung vorschreiben können, indem Sie Only verwenden CloudFront URLs, finden Sie unterthe section called “Beschränken Sie den Zugriff auf Dateien mit benutzerdefinierten Ursprüngen”.

Benutzerdefinierten Header hinzufügen

Wenn Sie benutzerdefinierte Header hinzufügen CloudFront möchten, wann immer eine Anfrage an Ihren Ursprung gesendet wird, geben Sie den Header-Namen und seinen Wert an. Weitere Informationen finden Sie unter the section called “Fügen Sie benutzerdefinierte Header zu ursprünglichen Anfragen hinzu”.

Informationen zur derzeit maximalen Anzahl von benutzerdefinierten Headern, die Sie hinzufügen können, zur maximalen Länge eines benutzerdefinierten Header-Namens und -Werts und zur maximalen Gesamtlänge aller Header-Namen und -Werte finden Sie unter Kontingente.

Origin Shield aktivieren

Wähle Ja, um CloudFront Origin Shield zu aktivieren. Weitere Informationen über Origin Shield erhalten Sie unter the section called “Benutze Origin Shield”.

Verbindungsversuche

Du kannst festlegen, wie oft CloudFront versucht wird, eine Verbindung zum Origin herzustellen. Sie können 1, 2 oder 3 als Anzahl der Versuche angeben. Die Standardanzahl (wenn Sie nichts anderes angeben) ist 3.

Verwenden Sie diese Einstellung zusammen mit dem Verbindungs-Timeout, um anzugeben, wie lange es CloudFront dauert, bis versucht wird, eine Verbindung zum sekundären Ursprung herzustellen, oder eine Fehlerantwort an den Viewer zurückgegeben wird. CloudFront Wartet standardmäßig bis zu 30 Sekunden (3 Versuche à 10 Sekunden), bevor versucht wird, eine Verbindung zum sekundären Ursprung herzustellen, oder eine Fehlerantwort zurückgegeben wird. Sie können diese Zeit reduzieren, indem Sie weniger Versuche, ein kürzeres Verbindungs-Timeout, oder beides angeben.

Wenn die angegebene Anzahl von Verbindungsversuchen fehlschlägt, CloudFront führt eine der folgenden Aktionen aus:

- Wenn der Ursprung Teil einer Ursprungsgruppe ist, wird CloudFront versucht, eine Verbindung zum sekundären Ursprung herzustellen. Schlägt die angegebene Anzahl von Verbindungsversuchen mit dem sekundären Ursprung fehl, wird eine Fehlerantwort an den Betrachter CloudFront zurückgegeben.
- Wenn der Ursprung nicht Teil einer Ursprungsgruppe ist, wird eine Fehlerantwort an den Betrachter CloudFront zurückgegeben.

Für einen benutzerdefinierten Ursprung (einschließlich eines Amazon S3 S3-Buckets, der mit statischem Website-Hosting konfiguriert ist) gibt diese Einstellung auch an, wie oft CloudFront versucht wird, eine Antwort vom Ursprung zu erhalten. Weitere Informationen finden Sie unter the section called “Timeout bei der Antwort”.

Verbindungstimeout

Das Verbindungs-Timeout ist die Anzahl der Sekunden, die CloudFront gewartet werden, wenn versucht wird, eine Verbindung zum Ursprung herzustellen. Sie können eine Anzahl von Sekunden zwischen 1 und 10 (inklusive) angeben. Das Standardtimeout (wenn Sie nichts anderes angeben) beträgt 10 Sekunden.

Verwenden Sie diese Einstellung zusammen mit Verbindungsversuchen, um anzugeben, wie lange CloudFront es dauert, bis versucht wird, eine Verbindung zum sekundären Ursprung herzustellen, oder bis eine Fehlerantwort an den Viewer zurückgegeben wird. CloudFront Wartet standardmäßig bis zu 30 Sekunden (3 Versuche à 10 Sekunden), bevor versucht wird, eine Verbindung zum sekundären Ursprung herzustellen, oder eine Fehlerantwort zurückgegeben wird. Sie können diese Zeit reduzieren, indem Sie weniger Versuche, ein kürzeres Verbindungs-Timeout, oder beides angeben.

Wenn innerhalb der angegebenen Anzahl von Sekunden CloudFront keine Verbindung zum Ursprung hergestellt wird, CloudFront wird eine der folgenden Aktionen ausgeführt:

- Wenn die angegebene Anzahl von Verbindungsversuchen mehr als 1 beträgt, wird erneut CloudFront versucht, eine Verbindung herzustellen. CloudFront versucht bis zu dreimal, je nach dem Wert der Verbindungsversuche.
- Wenn alle Verbindungsversuche fehlschlagen und der Ursprung Teil einer Ursprungsgruppe ist, versucht CloudFront , eine Verbindung mit dem sekundären Ursprung herzustellen. Wenn die

angegebene Anzahl von Verbindungsversuchen mit dem sekundären Ursprung fehlschlägt, wird eine Fehlerantwort an den Viewer CloudFront zurückgegeben.

- Wenn alle Verbindungsversuche fehlschlagen und der Ursprung nicht Teil einer Ursprungsgruppe ist, wird eine Fehlerantwort an den Viewer CloudFront zurückgegeben.

Timeout bei der Antwort

Das Ursprungs-Reaktions-Timeout, das auch als Ursprungs-Lese-Timeout oder Ursprungs-Anforderungs-Timeout bezeichnet wird, gilt für folgende Werte:

- Wie lange (in Sekunden) auf eine Antwort CloudFront gewartet wird, nachdem eine Anfrage an den Ursprung weitergeleitet wurde.
- Wie lange (in Sekunden) nach dem Empfang eines Antwortpakets vom Ursprung und vor dem Empfang des nächsten Pakets CloudFront gewartet wird.

> **Tip**
>
> Wenn Sie den Timeout-Wert erhöhen möchten, da die Viewer HTTP 504-Statuscodefehler erhalten, suchen Sie nach anderen Möglichkeiten zur Vermeidung dieser Fehler, bevor Sie den Timeout-Wert ändern. Vorschläge zur Fehlerbehebung finden Sie unter the section called “HTTP 504-Statuscode (Gateway Timeout)”.

CloudFront Das Verhalten hängt von der HTTP-Methode in der Viewer-Anfrage ab:

- `GET`und `HEAD` Anfragen — Wenn der Ursprung nicht oder nicht innerhalb der Dauer des Antwort-Timeouts reagiert, wird die CloudFront Verbindung unterbrochen. CloudFront versucht erneut, eine Verbindung entsprechend dem Wert von the section called “Verbindungsversuche” herzustellen.
- `DELETE`,`OPTIONS`, `PATCH`PUT, und `POST` Anfragen — Wenn der Absender für die Dauer des Lese-Timeouts nicht reagiert, CloudFront bricht er die Verbindung ab und versucht nicht erneut, den Ursprung zu kontaktieren. Der Client kann die Anfrage erneut senden, falls erforderlich.

Timeout bei Abschluss der Antwort

> **Note**
>
> Das Timeout für den Abschluss der Antwort unterstützt die Funktion zur kontinuierlichen Bereitstellung nicht.

Die Zeit (in Sekunden), während der eine Anfrage vom CloudFront ursprünglichen Server geöffnet bleiben und auf eine Antwort warten kann. Wenn bis zu diesem Zeitpunkt keine vollständige Antwort vom Ursprung eingegangen ist, wird die Verbindung CloudFront beendet.

Im Gegensatz zum Antwort-Timeout, bei dem es sich um die Wartezeit für einzelne Antwortpakete handelt, ist der Timeout für den Antwortabschluss die maximal zulässige Zeit, die auf CloudFront den Abschluss der Antwort warten darf. Sie können diese Einstellung verwenden, um sicherzustellen, dass CloudFront nicht unbegrenzt auf einen langsamen oder nicht reagierenden Ursprung gewartet wird, auch wenn andere Timeout-Einstellungen eine längere Wartezeit zulassen.

Dieses maximale Timeout beinhaltet das, was Sie für andere Timeout-Einstellungen angegeben haben, und die Anzahl der Verbindungsversuche bei jedem erneuten Versuch. Sie können diese Einstellungen zusammen verwenden, um anzugeben, wie lange CloudFront auf die vollständige Anfrage gewartet wird und wann die Anfrage beendet werden soll, unabhängig davon, ob sie abgeschlossen ist oder nicht.

Wenn Sie beispielsweise die folgenden Einstellungen festlegen:

- Die Anzahl der Verbindungsversuche beträgt 3
- Das Verbindungs-Timeout beträgt 10 Sekunden
- Das Antwort-Timeout beträgt 30 Sekunden
- Das Timeout für den Abschluss der Antwort beträgt 60 Sekunden

Das bedeutet, dass versucht CloudFront wird, eine Verbindung zum Ursprung herzustellen (bis zu 3 Versuche insgesamt), wobei bei jedem Verbindungsversuch eine Zeitüberschreitung von 10 Sekunden eintritt. Sobald die Verbindung hergestellt ist, CloudFront wird bis zu 30 Sekunden gewartet, bis der Ursprung auf die Anfrage antwortet, bis er das letzte Paket der Antwort erhält.

Unabhängig von der Anzahl der Verbindungsversuche oder dem Antwort-Timeout CloudFront wird die Verbindung beendet, wenn die vollständige Antwort vom Ursprung länger als 60 Sekunden

dauert. CloudFront gibt dem Betrachter dann eine the section called “HTTP 504-Statuscode (Gateway Timeout)” Fehlerantwort oder eine benutzerdefinierte Fehlerantwort zurück, falls Sie eine angegeben haben.

> **Hinweise**
>
> - Wenn Sie einen Wert für das Timeout für den Abschluss der Antwort festlegen, muss der Wert gleich oder größer als der Wert für das Antwort-Timeout (Origin-Read-Timeout) sein.
> - Wenn Sie keinen Wert für das Timeout für den Abschluss der Antwort festlegen, wird CloudFront kein Höchstwert erzwungen.

Keep-Alive-Timeout (nur benutzerdefinierte und VPC-Ursprünge)

Das Keep-Alive-Timeout gibt an, wie lange (in Sekunden) CloudFront versucht wird, eine Verbindung zu Ihrem benutzerdefinierten Ursprung aufrechtzuerhalten, nachdem dieser das letzte Paket einer Antwort erhalten hat. Durch eine persistente Verbindung wird die Zeit gespart, die erforderlich ist, um die TCP-Verbindung erneut herzustellen und einen weiteren TLS-Handshake für nachfolgende Anfragen durchzuführen. Eine Erhöhung des Keep-Alive-Timeouts trägt zur Verbesserung der Metrik für Distributionen bei. request-per-connection

> **Note**
>
> Damit der Wert für das Keepalive-Timeout Auswirkungen hat, muss Ihr Ursprungsserver ständige Verbindungen zulassen.

Quoten für Antwort- und Keep-Alive-Timeouts

- Für das Antwort-Timeout ist die Standardeinstellung 30 Sekunden.
- Für das Keep-Alive-Timeout ist der Standardwert 5 Sekunden.

Wenn Sie eine Erhöhung des Timeouts für Ihre anfordern AWS-Konto, aktualisieren Sie Ihre Distributionsquellen so, dass sie die gewünschten Werte für das Antwort-Timeout und das Keep-Alive-Timeout haben. Bei einer Erhöhung des Kontingents für Ihr Konto werden Ihre Herkunftsländer nicht automatisch aktualisiert. Wenn Sie beispielsweise eine Lambda @Edge -Funktion verwenden, um ein Keep-Alive-Timeout von 90 Sekunden festzulegen, muss Ihr Origin bereits ein Keep-Alive-

Timeout von 90 Sekunden oder mehr haben. Andernfalls kann Ihre Lambda @Edge -Funktion möglicherweise nicht ausgeführt werden.

Weitere Informationen zu Verteilungsquoten, einschließlich der Beantragung einer Erhöhung, finden Sie unterAllgemeine Kontingente für Verteilungen.

Einstellungen für das Cache-Verhalten

Durch die Einstellung des Cache-Verhaltens können Sie eine Vielzahl von CloudFront Funktionen für ein bestimmtes URL-Pfadmuster für Dateien auf Ihrer Website konfigurieren. Beispielsweise kann ein Cache-Verhalten für alle `.jpg`-Dateien im `images`-Verzeichnis auf einem Webserver gelten, den Sie als Ursprungs-Server für CloudFront verwenden. Zu den Funktionen, die Sie für jedes Cache-Verhalten konfigurieren können, gehören folgende:

- Das Pfadmuster
- Wenn Sie mehrere Ursprünge für Ihre CloudFront Distribution konfiguriert haben, den Ursprung, an den Sie Ihre Anfragen weiterleiten CloudFront möchten
- Die Angabe, ob Abfragezeichenfolgen an Ihren Ursprung weiterleitet werden sollen
- Gibt an, ob für den Zugriff auf die angegebenen Dateien eine Signatur erforderlich ist URLs
- Die Angabe, ob Benutzer HTTPS für den Zugriff auf diese Dateien verwenden müssen
- Die Mindestdauer, für die diese Dateien im CloudFront Cache verbleiben, unabhängig vom Wert der `Cache-Control` Header, die Ihr Origin den Dateien hinzufügt

Wenn Sie eine neue Verteilung erstellen, geben Sie Einstellungen für das Standard-Cache-Verhalten an, das alle Anfragen automatisch an den Ursprung weiterleitet, den Sie beim Erstellen der Verteilung angeben. Nachdem Sie eine Verteilung erstellt haben, können Sie zusätzliche Cache-Verhaltensweisen erstellen, die definieren, wie sie CloudFront reagiert, wenn sie eine Anforderung für Objekte empfängt, die einem Pfadmuster entsprechen, `*.jpg` z. B. Wenn Sie zusätzliche Cache-Verhalten erstellen, wird das Standard-Cache-Verhalten immer als Letztes verarbeitet. Andere Cache-Verhaltensweisen werden in der Reihenfolge verarbeitet, in der sie in der CloudFront Konsole aufgeführt sind, oder, wenn Sie die CloudFront API verwenden, in der Reihenfolge, in der sie im `DistributionConfig` Element für die Verteilung aufgeführt sind. Weitere Informationen finden Sie unter Pfadmuster.

Wenn Sie ein Cache-Verhalten erstellen, geben Sie den einen Ursprung an, von dem Sie Objekte abrufen CloudFront möchten. Wenn Sie also Objekte aus all Ihren Ursprüngen verteilen CloudFront

möchten, müssen Sie mindestens so viele Cache-Verhalten (einschließlich des Standard-Cache-Verhaltens) wie Ursprünge haben. Wenn Sie beispielsweise zwei Ursprünge und nur das Standard-Cache-Verhalten haben, bewirkt CloudFront das Standard-Cache-Verhalten, dass Objekte von einem der Ursprünge abgerufen werden, der andere Ursprung jedoch nie verwendet wird.

Informationen zur aktuell gültigen maximalen Anzahl von Cache-Verhaltensweisen, die Sie einer Verteilung hinzufügen können, oder zum Anfordern eines höheren Kontingents (früher als Limit bezeichnet) finden Sie unter Allgemeine Kontingente für Verteilungen.

Themen

- Pfadmuster
- Ursprung oder Ursprungsgruppe
- Viewer-Protokollrichtlinien
- Zulässige HTTP-Methoden
- Verschlüsselungskonfiguration auf Feldebene
- Zwischengespeicherte HTTP-Methoden
- gRPC-Anfragen über HTTP/2 zulassen
- Basierend auf den ausgewählten Anforderungsheadern
- Zulassungslisten-Header
- Zwischenspeicherung von Objekten
- Mindest-TTL
- Höchst-TTL
- Standard-TTL
- Cookies weiterleiten
- Zulassungslisten-Cookies
- Weiterleitung und Zwischenspeicherung von Abfragezeichenfolgen
- Zulassungsliste für Abfragezeichenfolgen
- Smooth Streaming
- Beschränken Sie den Zuschauerzugriff (verwenden Sie signierte URLs oder signierte Cookies)
- Vertrauenswürdiger Signaturgeber
- AWS-Konto -Ziffern

- Objekte automatisch komprimieren
- CloudFront Ereignis
- ARN der Lambda-Funktion
- Include body (Text einschließen)

Pfadmuster

Ein Pfadmuster (z. B. `images/*.jpg`) gibt an, für welche Anfragen dieses Cache-Verhalten gelten soll. Wenn eine CloudFront Endbenutzeranfrage eingeht, wird der angeforderte Pfad mit Pfadmustern in der Reihenfolge verglichen, in der das Cache-Verhalten in der Distribution aufgeführt ist. Die erste Übereinstimmung bestimmt, welches Cache-Verhalten auf diese Anfrage angewendet wird. Nehmen wir beispielsweise an, dass Sie drei Cache-Verhalten mit den folgenden drei Pfadmustern haben, in dieser Reihenfolge:

- `images/*.jpg`
- `images/*`
- `*.gif`

> **Note**
>
> Sie können optional einen Schrägstrich (/) am Anfang des Pfadmusters angeben, zum Beispiel. `/images/*.jpg` CloudFront Das Verhalten ist mit oder ohne das führende/ identisch. Wenn Sie/nicht am Anfang des Pfads angeben, wird dieses Zeichen automatisch impliziert. CloudFront Behandelt den Pfad auf dieselbe Weise, mit oder ohne das führende /. CloudFront Behandelt zum Beispiel `/*product.jpg` dasselbe wie `*product.jpg`

Eine Anfrage für die Datei `images/sample.gif` entspricht nicht dem ersten Pfadmuster, sodass die zugehörigen Cache-Verhalten nicht auf die Anfrage angewendet werden. Die Datei entspricht dem zweiten Pfadmuster, sodass die mit dem zweiten Pfadmuster verknüpften Cache-Verhalten angewendet werden, auch wenn die Anfrage ebenfalls dem dritten Pfadmuster entspricht.

> **Note**
>
> Wenn Sie eine neue Verteilung erstellen, wird der Wert von Path Pattern für das Standard-Cache-Verhalten auf * (alle Dateien) gesetzt und kann nicht geändert werden. Dieser Wert

bewirkt CloudFront , dass alle Anfragen für Ihre Objekte an den Ursprung weitergeleitet werden, den Sie im Ursprungsdomäne Feld angegeben haben. Wenn die Anforderung für ein Objekt nicht mit dem Pfadmuster für eines der anderen Cache-Verhalten übereinstimmt, wird das Verhalten CloudFront angewendet, das Sie im Standard-Cache-Verhalten angeben.

Important

Gehen Sie beim Definieren von Pfadmustern und ihrer Reihenfolge sorgfältig vor. Andernfalls gewähren Sie Benutzern möglicherweise unerwünscht Zugriff auf Ihre Inhalte. Nehmen wir beispielsweise an, eine Anfrage stimmt mit dem Pfadmuster für zwei Cache-Verhalten überein. Für das erste Cache-Verhalten ist keine Signatur erforderlich URLs und für das zweite Cache-Verhalten ist eine Signatur erforderlich URLs. Benutzer können auf die Objekte zugreifen, ohne eine signierte URL zu verwenden, da das Cache-Verhalten CloudFront verarbeitet wird, das mit der ersten Übereinstimmung verknüpft ist.

Wenn Sie mit einem MediaPackage Kanal arbeiten, müssen Sie spezifische Pfadmuster für das Cache-Verhalten angeben, das Sie für den Endpunkttyp für Ihren Ursprung definieren. Für einen DASH-Endpunkt geben Sie beispielsweise `*.mpd` in Path Pattern (Pfadmuster) ein. Weitere Informationen und spezielle Anweisungen finden Sie unter Bereitstellen Sie Live-Videos, formatiert mit AWS Elemental MediaPackage.

Der von Ihnen angegebene Pfad gilt für Anfragen für alle Dateien im angegebenen Verzeichnis und in Unterverzeichnissen unterhalb des angegebenen Verzeichnisses. CloudFront berücksichtigt bei der Auswertung des Pfadmusters keine Abfragezeichenfolgen oder Cookies. Wenn ein `images`-Verzeichnis beispielsweise `product1`- und `product2`-Unterverzeichnisse enthält, gilt das Pfadmuster `images/*.jpg` für Anfragen für alle JPG-Dateien in den Verzeichnissen `images`, `images/product1` und `images/product2`. Wenn Sie auf die Dateien im Verzeichnis `images/product1` ein anderes Cache-Verhalten als auf die Dateien in den Verzeichnissen `images` und `images/product2` anwenden möchten, erstellen Sie ein separates Cache-Verhalten für `images/product1` und verschieben Sie dieses Cache-Verhalten an eine Stelle über (vor) dem Cache-Verhalten für das Verzeichnis `images`.

Sie können die folgenden Platzhalterzeichen in Ihrem Pfadmuster verwenden:

- `*` entspricht 0 oder mehr Zeichen.
- `?` entspricht genau 1 Zeichen.

Die folgenden Beispiele zeigen, wie die Platzhalterzeichen funktionieren:

Pfadmuster	Dateien, die dem Pfadmuster entsprechen
`*.jpg`	Alle .jpg-Dateien.
`images/*.jpg`	Alle .jpg-Dateien im `images`-Verzeichnis und in den Unterverzeichnissen unter dem `images`-Verzeichnis.
`a*.jpg`	• Alle .jpg-Dateien, deren Dateiname mit a beginnt, z. B. `apple.jpg` und `appalachian_trail_2012_05_21.jpg` . • Alle JPG-Dateien, deren Dateipfad mit a beginnt, z. B. `abra/cadabra/magic.jpg` .
`a??.jpg`	Alle .jpg-Dateien, deren Dateiname mit a beginnt, gefolgt von genau zwei anderen Zeichen, z. B. `ant.jpg` und `abe.jpg`.
`*.doc*`	Alle Dateien, deren Dateinamenerweiterung mit `.doc` beginnt, z. B. `.doc`-, `.docx`- und `.docm`-Dateien. Sie können das Pfadmuster `*.doc?` in diesem Fall nicht verwenden, weil dieses Pfadmuster nicht für Anfragen für `.doc`-Dateien gelten würde. Das Platzhalterzeichen ? ersetzt genau ein Zeichen.

Die maximale Länge eines Pfadmusters beträgt 255 Zeichen. Der Wert kann folgende Zeichen enthalten:

- A-Z, a-z

  Bei den Pfadmustern muss die Groß- und Kleinschreibung beachtet werden, so dass das Pfadmuster `*.jpg` nicht für die Datei `LOGO.JPG` gilt.
- 0-9
- _ - . * $ / ~ " ' @ : +

- & (übergeben und zurückgegeben als `&`)

Normalisierung des Pfads

CloudFront normalisiert URI-Pfade gemäß RFC 3986 und ordnet dem Pfad dann das richtige Cache-Verhalten zu. Sobald das Cache-Verhalten übereinstimmt, wird der unformatierte URI-Pfad CloudFront an den Ursprung gesendet. Wenn sie nicht übereinstimmen, werden Anfragen stattdessen mit Ihrem Standard-Cache-Verhalten abgeglichen.

Einige Zeichen werden normalisiert und aus dem Pfad entfernt, z. B. mehrere Schrägstriche (`//`) oder Punkte (`..`). Dadurch kann die URL, die CloudFront verwendet wird, so geändert werden, dass sie dem beabsichtigten Cache-Verhalten entspricht.

Example Beispiel

Sie geben die `/a*` Pfade `/a/b*` und die Pfade für Ihr Cache-Verhalten an.

- Ein Betrachter, der den `/a/b?c=1` Pfad sendet, entspricht dem Verhalten des `/a/b*` Caches.
- Ein Betrachter, der den `/a/b/..?c=1` Pfad sendet, entspricht dem Verhalten des `/a*` Caches.

Um die Normalisierung der Pfade zu umgehen, können Sie Ihre Anforderungspfade oder das Pfadmuster für das Cache-Verhalten aktualisieren.

Ursprung oder Ursprungsgruppe

Diese Einstellung gilt nur, wenn Sie ein Cache-Verhalten für eine bestehende Distribution erstellen oder aktualisieren.

Geben Sie den Wert eines vorhandenen Ursprungs oder einer Ursprungsgruppe ein. Dadurch wird der Ursprung oder die Ursprungsgruppe identifiziert, an die Sie Anfragen weiterleiten CloudFront möchten, wenn eine Anforderung (wie https://example.com /logo.jpg) dem Pfadmuster für ein Cache-Verhalten (z. B. *.jpg) oder für das Standard-Cache-Verhalten (*) entspricht.

Viewer-Protokollrichtlinien

Wählen Sie die Protokollrichtlinie aus, die Zuschauer für den Zugriff auf Ihre Inhalte an CloudFront Edge-Standorten verwenden sollen:

- HTTP und HTTPS: Viewer können beide Protokolle verwenden.

- Redirect HTTP to HTTPS: Viewer können zwar beide Protokolle verwenden, doch HTTP-Anfragen werden automatisch umgeleitet und als HTTPS-Anfragen gesendet.
- HTTPS Only: Viewer können nur auf Ihre Inhalte zugreifen, wenn sie HTTPS verwenden

Weitere Informationen finden Sie unter Erfordert HTTPS für die Kommunikation zwischen Zuschauern und CloudFront.

Zulässige HTTP-Methoden

Geben Sie die HTTP-Methoden an, die Sie verarbeiten und CloudFront an Ihren Ursprung weiterleiten möchten:

- GET, HEAD: Sie können es CloudFront nur verwenden, um Objekte von Ihrem Ursprung abzurufen oder um Objekt-Header abzurufen.
- GET, HEAD, OPTIONS: Sie können CloudFront nur für den Objektabruf aus dem Ursprung, den Abruf von Objekt-Headern oder den Abruf einer Liste mit Optionen, die vom Ursprungs-Server unterstützt werden, verwenden.
- GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE: Sie können CloudFront sie verwenden, um Objekte abzurufen, hinzuzufügen, zu aktualisieren und zu löschen sowie Objekt-Header abzurufen. Darüber hinaus können Sie andere POST-Vorgänge wie das Senden von Daten aus einem Webformular ausführen.

> **Note**
>
> Wenn Sie gRPC in Ihrem Workload verwenden, müssen Sie GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE auswählen gRPC gRPC-Workloads erfordern die Methode. `POST` Weitere Informationen finden Sie unter gRPC mit CloudFront Distributionen verwenden. CloudFront speichert Antworten auf `HEAD` Anfragen `GET` und optional Anfragen im Cache. `OPTIONS` Antworten auf `OPTIONS` Anfragen werden getrennt von Antworten auf `GET` und `HEAD` Anfragen zwischengespeichert (die `OPTIONS` Methode ist im Cache-Schlüssel für `OPTIONS` Anfragen enthalten). CloudFront speichert keine Antworten auf Anfragen, die andere Methoden verwenden.

Important

Wenn Sie GET, HEAD, OPTIONS oder GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE auswählen, müssen Sie möglicherweise den Zugriff auf den Amazon S3-Bucket oder den benutzerdefinierten Ursprung einschränken, um zu verhindern, dass Benutzer Operationen ausführen, die sie nicht ausführen sollen. Die folgenden Beispiele erläutern, wie Sie den Zugriff beschränken:

- Wenn Sie Amazon S3 als Quelle für Ihren Vertrieb verwenden: Erstellen Sie eine CloudFront Ursprungszugriffskontrolle, um den Zugriff auf Ihre Amazon S3 S3-Inhalte einzuschränken, und erteilen Sie der Ursprungszugriffskontrolle Berechtigungen. Wenn Sie beispielsweise so konfigurieren CloudFront , dass diese Methoden nur akzeptiert und weitergeleitet werden, weil Sie sie verwenden möchtenPUT, müssen Sie dennoch die Amazon S3 S3-Bucket-Richtlinien konfigurieren, um `DELETE` Anfragen angemessen zu behandeln. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung.
- Wenn Sie einen benutzerdefinierten Ursprungs-Server verwenden: Konfigurieren Sie Ihren Ursprungs-Server so, dass alle Methoden verarbeitet werden. Wenn Sie beispielsweise so konfigurieren CloudFront , dass diese Methoden nur akzeptiert und weitergeleitet werden, weil Sie sie verwenden möchtenPOST, müssen Sie Ihren Ursprungsserver trotzdem so konfigurieren, dass er `DELETE` Anfragen entsprechend verarbeitet.

Verschlüsselungskonfiguration auf Feldebene

Wenn Sie die Verschlüsselung auf Feldebene für bestimmte Datenfelder erzwingen möchten, wählen Sie in den Dropdown-Listen eine Verschlüsselungskonfiguration auf Feldebene.

Weitere Informationen finden Sie unter Vertrauliche Daten durch Verschlüsselung auf Feldebene schützen.

Zwischengespeicherte HTTP-Methoden

Geben Sie an CloudFront , ob Sie die Antwort von Ihrem Ursprung zwischenspeichern möchten, wenn ein Zuschauer eine `OPTIONS` Anfrage einreicht. CloudFront speichert die Antwort auf `GET` und `HEAD` Anfragen immer im Cache.

gRPC-Anfragen über HTTP/2 zulassen

Geben Sie an, ob Ihre Distribution gRPC-Anfragen zulassen soll. Um gRPC zu aktivieren, wählen Sie die folgenden Einstellungen:

- Wählen Sie für Zulässige HTTP-Methoden die Methoden GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE aus. gRPC benötigt die `POST` Methode.
- Aktivieren Sie das gRPC-Kontrollkästchen, das nach der Auswahl der `POST` Methode angezeigt wird.
- Wählen Sie für Unterstützte HTTP-VersionenHTTP/2 aus.

Weitere Informationen finden Sie unter gRPC mit CloudFront Distributionen verwenden.

Basierend auf den ausgewählten Anforderungsheadern

Geben Sie an, ob Sie CloudFront Objekte auf der Grundlage der Werte der angegebenen Header zwischenspeichern möchten:

- Keine (verbessert das Caching) — Ihre Objekte werden CloudFront nicht auf der Grundlage von Header-Werten zwischengespeichert.
- Allowlist — CloudFront speichert Ihre Objekte nur auf der Grundlage der Werte der angegebenen Header im Cache. Verwenden Sie Allowlist Headers, um die Header auszuwählen, auf denen das Caching basieren soll. CloudFront
- Alle — Die Objekte, die mit diesem Cache-Verhalten verknüpft sind, werden CloudFront nicht zwischengespeichert. CloudFront Sendet stattdessen jede Anfrage an den Ursprung. (Nicht empfohlen für Amazon S3-Ursprünge.)

Leitet unabhängig von der ausgewählten Option bestimmte Header CloudFront an Ihren Ursprung weiter und ergreift auf der Grundlage der von Ihnen weitergeleiteten Header bestimmte Aktionen. Weitere Hinweise zur CloudFront Handhabung der Header-Weiterleitung finden Sie unter. Header und CloudFront Verhalten von HTTP-Anfragen (benutzerdefiniert und Amazon S3 S3-Ursprünge)

Weitere Hinweise zur Konfiguration des Zwischenspeichers mithilfe CloudFront von Anforderungsheadern finden Sie unter. Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern

Zulassungslisten-Header

Diese Einstellungen gelten nur, wenn Sie Allowlist for Cache Based on Selected Request Headers auswählen.

Geben Sie die Header an, die Sie beim CloudFront Zwischenspeichern Ihrer Objekte berücksichtigen möchten. Wählen Sie die Header aus der Liste der verfügbaren Header aus und klicken Sie auf Add. Um einen benutzerdefinierten Header weiterzuleiten, geben Sie den Namen des Headers in das Feld ein und wählen Sie Add Custom.

Informationen zur aktuell gültigen maximalen Anzahl von Headern, die Sie für die einzelnen Cache-Verhaltensweisen auf die Zulassungsliste setzen können, oder zum Anfordern eines höheren Kontingents (früher als Limit bezeichnet) finden Sie unter Kontingente für Header.

Zwischenspeicherung von Objekten

Wenn Ihr Original-Server Ihren Objekten einen `Cache-Control` Header hinzufügt, um zu kontrollieren, wie lange die Objekte im CloudFront Cache bleiben, und wenn Sie den `Cache-Control` Wert nicht ändern möchten, wählen Sie Origin-Cache-Header verwenden.

Um eine Mindest- und Höchstdauer anzugeben, für die Ihre Objekte unabhängig von den **`Cache-Control`** Headern im CloudFront Cache verbleiben, und eine Standardzeit, für die Ihre Objekte im CloudFront Cache verbleiben, wenn der **`Cache-Control`** Header eines Objekts fehlt, wählen Sie Anpassen. Geben Sie dann in den Feldern Minimum TTL, Default TTL und Maximum TTL den entsprechenden Wert ein.

Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Mindest-TTL

Geben Sie die Mindestdauer in Sekunden an, für die Objekte im CloudFront Cache verbleiben sollen, bevor eine weitere Anfrage an den Ursprung CloudFront gesendet wird, um festzustellen, ob das Objekt aktualisiert wurde.

> **Warning**
>
> Wenn Ihre Mindest-TTL größer als 0 ist, CloudFront wird Inhalt mindestens für die in der Mindest-TTL der Cache-Richtlinie angegebene Dauer zwischengespeichert, auch wenn die

`private` Direktiven `Cache-Control: no-cacheno-store`, oder in den Origin-Headern vorhanden sind.

Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Höchst-TTL

Geben Sie die maximale Zeit in Sekunden an, für die Objekte in CloudFront Caches verbleiben sollen, bevor Ihr Ursprung CloudFront abgefragt wird, um festzustellen, ob das Objekt aktualisiert wurde. Der Wert, den Sie für Maximum TTL angeben, gilt nur, wenn Ihr Ursprung HTTP-Header, wie beispielsweise `Cache-Control max-age`, `Cache-Control s-maxage` oder `Expires` zu Objekten hinzufügt. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Um einen Wert für Maximum TTL anzugeben, müssen Sie die Option Customize für die Object Caching-Einstellung auswählen.

Der Standardwert für Maximum TTL beträgt 31 536 000 Sekunden (ein Jahr). Wenn Sie den Wert von Minimum TTL oder Default TTL in mehr als 31 536 000 Sekunden ändern, ändert sich der Standardwert von Maximum TTL in den Wert von Default TTL.

Standard-TTL

Geben Sie die Standarddauer in Sekunden an, CloudFront für die Objekte in CloudFront Caches verbleiben sollen, bevor eine weitere Anfrage an Ihren Ursprung weitergeleitet wird, um festzustellen, ob das Objekt aktualisiert wurde. Der Wert, den Sie für Default TTL (Standardgültigkeitsdauer) angeben, gilt nur, wenn Ihr Ursprung keine HTTP-Header wie `Cache-Control max-age`, `Cache-Control s-maxage` oder `Expires` zu Objekten hinzufügt. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Um einen Wert für Default TTL anzugeben, müssen Sie die Option Customize für die Object Caching-Einstellung auswählen.

Der Standardwert für Default TTL beträgt 86 400 Sekunden (ein Tag). Wenn Sie den Wert von Minimum TTL in mehr als 86 400 Sekunden ändern, ändert sich der Standardwert von Default TTL in den Wert von Minimum TTL.

Cookies weiterleiten

> **Note**
>
> Bei Amazon-S3-Ursprüngen gilt diese Option nur für Buckets, die als Website-Endpunkte konfiguriert sind.

Geben Sie an CloudFront , ob Sie Cookies an Ihren Ursprungsserver weiterleiten möchten und wenn ja, welche. Wenn nur ausgewählte Cookies (eine Zulassungsliste von Cookies) weitergeleitet werden sollen, geben Sie die Cookie-Namen im Feld Zulassungslisten-Cookies ein. Wenn Sie All auswählen, leitet CloudFront alle Cookies weiter, unabhängig davon, wie viele Ihre Anwendung verwendet.

Amazon S3 verarbeitet keine Cookies. Die Weiterleitung von Cookies an den Ursprung reduziert die Möglichkeit zum Caching. Wählen Sie für Cache-Verhalten, die Anforderungen an einem Amazon S3-Ursprung weiterleiten, None (Keine) in Forward Cookies (Cookies weiterleiten) aus.

Weitere Informationen zum Weiterleiten von Cookies an den Ursprung finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.

Zulassungslisten-Cookies

> **Note**
>
> Bei Amazon S3-Ursprüngen gilt diese Option nur für Buckets, die als Website-Endpunkte konfiguriert sind.

Wenn Sie in der Liste Forward-Cookies die Option Allowlist ausgewählt haben, geben Sie im Feld Allowlist-Cookies die Namen der Cookies ein, die Sie für dieses Cache-Verhalten an Ihren Ursprungsserver weiterleiten möchten CloudFront. Geben Sie die einzelnen Cookie-Namen jeweils in einer neuen Zeile ein.

Sie können die folgenden Platzhalter verwenden, wenn Sie Cookie-Namen angeben:

- * steht für 0 oder mehr Zeichen in dem Cookie-Namen
- ? steht für genau 1 Zeichen in dem Cookie-Namen

Nehmen wir beispielsweise an, dass Viewer-Anforderungen für ein Objekt ein Cookie mit dem Namen

`userid_`*`member-number`*

Wofür jeder Ihrer Benutzer einen eindeutigen Wert hat. *`member-number`* Sie CloudFront möchten für jedes Mitglied eine separate Version des Objekts zwischenspeichern. Sie könnten dies erreichen, indem Sie alle Cookies an Ihren Ursprung weiterleiten, aber die Anfragen von Zuschauern enthalten einige Cookies, die Sie nicht zwischenspeichern CloudFront möchten. Alternativ könnten Sie den folgenden Wert als Cookie-Namen angeben, wodurch CloudFront alle Cookies, die mit beginnen, an den Ursprung weitergeleitet werden`userid_`:

`userid_*`

Informationen zur aktuell gültigen maximalen Anzahl von Cookie-Namen, die Sie für die einzelnen Cache-Verhaltensweisen auf die Zulassungsliste setzen können, oder zum Anfordern eines höheren Kontingents (früher als Limit bezeichnet) finden Sie unter Kontingente für Cookies (Legacy-Cache-Einstellungen).

Weiterleitung und Zwischenspeicherung von Abfragezeichenfolgen

CloudFront kann verschiedene Versionen Ihres Inhalts auf der Grundlage der Werte von Abfragezeichenfolgenparametern zwischenspeichern. Wählen Sie eine der folgenden Optionen:

None (Improves Caching)

Wählen Sie diese Option aus, wenn Ihr Ursprung dieselbe Version eines Objekts unabhängig von den Werten der Abfragezeichenfolgeparameter zurückgibt. Dadurch steigt die Wahrscheinlichkeit, dass eine Anfrage aus dem Cache bedient werden CloudFront kann, wodurch die Leistung verbessert und die Belastung Ihres Quellservers verringert wird.

Alle weiterleiten, basierend auf Zulassungliste zwischenspeichern

Wählen Sie diese Option aus, wenn Ihr Ursprungs-Server verschiedene Versionen Ihrer Objekte auf der Grundlage von einem oder mehreren Abfragezeichenfolgeparametern zurückgibt. Geben Sie dann die Parameter CloudFront an, die Sie als Grundlage für das Caching im Zulassungsliste für Abfragezeichenfolgen Feld verwenden möchten.

Forward all, cache based on all

Wählen Sie diese Option aus, wenn Ihr Ursprungs-Server verschiedene Versionen Ihrer Objekte für alle Abfragezeichenfolgeparameter zurückgibt.

Weitere Informationen zur Zwischenspeicherung auf der Grundlage von Abfragezeichenfolgeparametern, einschließlich Informationen zur Verbesserung der Leistung, finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern.

Zulassungsliste für Abfragezeichenfolgen

Diese Einstellung gilt nur, wenn Sie Alle weiterleiten, auf der Zulassungsliste basierender Zwischenspeicher für wählen. Weiterleitung und Zwischenspeicherung von Abfragezeichenfolgen Sie können die Parameter der Abfragezeichenfolge angeben, die Sie als Grundlage für das Caching verwenden möchten CloudFront .

Smooth Streaming

Wählen Sie Yes, wenn Sie Mediendateien im Microsoft Smooth Streaming-Format verteilen möchten und keinen IIS-Server verwenden.

Wählen Sie No, wenn Sie einen Microsoft IIS-Server als Ursprung verwenden möchten, um Mediendateien im Microsoft Smooth Streaming-Format zu verteilen, oder wenn Sie keine Smooth Streaming-Mediendateien verteilen.

> **Note**
>
> Wenn Sie Yes angeben, können Sie weiterhin andere Inhalte mit diesem Cache-Verhalten verteilen, wenn diese Inhalte mit dem Wert von Path Pattern übereinstimmen.

Weitere Informationen finden Sie unter Video-on-Demand für Microsoft Smooth Streaming konfigurieren.

Beschränken Sie den Zuschauerzugriff (verwenden Sie signierte URLs oder signierte Cookies)

Wenn Sie möchten, dass Anfragen für Objekte, die dem Verhalten `PathPattern` für diesen Cache entsprechen URLs, öffentlich verwendet werden, wählen Sie Nein.

Wenn Anfragen für Objekte, die dem Verhalten `PathPattern` für diesen Cache entsprechen URLs, signiert werden sollen, wählen Sie Ja. Geben Sie dann die AWS Konten an, die Sie zum Erstellen signierter Konten verwenden möchten URLs. Diese Konten werden als vertrauenswürdige Unterzeichner bezeichnet.

Weitere Informationen zu vertrauenswürdigen Ausstellern finden Sie unter Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können.

Vertrauenswürdiger Signaturgeber

Diese Einstellung gilt nur, wenn Sie für Zuschauerzugriff einschränken (Signierte URLs oder signierte Cookies verwenden) die Option Ja wählen.

Wählen Sie aus, welche AWS Konten Sie als vertrauenswürdige Unterzeichner für dieses Cache-Verhalten verwenden möchten:

- Selbst: Verwende das Konto, mit dem du derzeit AWS Management Console als vertrauenswürdiger Unterzeichner angemeldet bist. Wenn Sie derzeit als IAM-Benutzer angemeldet sind, wird das zugehörige AWS Konto als vertrauenswürdiger Unterzeichner hinzugefügt.
- Wenn Sie das Kontrollkästchen Specify Accounts aktiviert haben, geben Sie die Konto-IDs der vertrauenswürdigen Signaturgeber im Feld AWS Account Numbers ein.

Um ein signiertes AWS Konto zu erstellen URLs, muss es mindestens ein aktives CloudFront key pair haben.

> **Important**
>
> Wenn Sie eine Distribution aktualisieren, die Sie bereits zum Verteilen von Inhalten verwenden, fügen Sie vertrauenswürdige Unterzeichner erst hinzu, wenn Sie bereit sind, mit der Generierung signierter Dateien URLs für Ihre Objekte zu beginnen. Nachdem Sie einer Distribution vertrauenswürdige Unterzeichner hinzugefügt haben, müssen Benutzer signiert verwenden, URLs um auf die Objekte zuzugreifen, die dem Verhalten `PathPattern` für diesen Cache entsprechen.

AWS-Konto -Ziffern

Diese Einstellung gilt nur, wenn Sie Konten für vertrauenswürdige Unterzeichner angeben wählen.

Wenn Sie zusätzlich oder anstelle des aktuellen Kontos ein signiertes URLs Konto erstellen möchten, geben Sie eine AWS-Konto Zahl pro Zeile in dieses Feld ein. AWS-Konten Beachten Sie Folgendes:

- Die angegebenen Konten müssen über mindestens ein aktives CloudFront-Schlüsselpaar verfügen. Weitere Informationen finden Sie unter Erstellen Sie Schlüsselpaare für Ihre Unterzeichner.
- Sie können keine CloudFront Schlüsselpaare für IAM-Benutzer erstellen, sodass Sie IAM-Benutzer nicht als vertrauenswürdige Unterzeichner verwenden können.
- Informationen zum Abrufen der AWS-Konto Nummer für ein Konto finden Sie im Management-Referenzhandbuch unter AWS-Konto Identifikatoren anzeigen.AWS-Konto
- Wenn Sie die Kontonummer für das Girokonto eingeben, CloudFront wird automatisch das Kontrollkästchen Selbst aktiviert und die Kontonummer wird aus der AWS Kontonummernliste entfernt.

Objekte automatisch komprimieren

Wenn Sie Dateien bestimmter Typen automatisch komprimieren CloudFront möchten, wenn Zuschauer komprimierte Inhalte unterstützen, wählen Sie Ja. Wenn CloudFront Ihre Inhalte komprimiert, werden Downloads schneller ausgeführt, da die Dateien kleiner sind, und Ihre Webseiten werden schneller für Ihre Benutzer wiedergegeben. Weitere Informationen finden Sie unter Komprimierte Dateien bereitstellen.

CloudFront Ereignis

Diese Einstellung gilt für Lambda-Funktionszuordnungen.

Sie können wählen, ob eine Lambda-Funktion ausgeführt werden soll, wenn eines oder mehrere der folgenden CloudFront Ereignisse eintreten:

- Wann CloudFront erhält er eine Anfrage von einem Zuschauer (Zuschaueranfrage)
- Bevor CloudFront eine Anfrage an den Ursprung weitergeleitet wird (ursprüngliche Anfrage)
- Wann CloudFront erhält er eine Antwort vom Ursprung (ursprüngliche Antwort)
- Before CloudFront gibt die Antwort an den Zuschauer zurück (Antwort des Betrachters)

Weitere Informationen finden Sie unter Wählen Sie das Ereignis aus, das die Funktion auslösen soll.

ARN der Lambda-Funktion

Diese Einstellung gilt für Lambda-Funktionszuordnungen.

Geben Sie den Amazon-Ressourcennamen (ARN) der Lambda-Funktion an, für die Sie einen Auslöser hinzufügen möchten. Informationen zum Abrufen des ARN für eine Funktion finden Sie in Schritt 1 des Verfahrens Hinzufügen von Triggern mithilfe der CloudFront Konsole.

Include body (Text einschließen)

Diese Einstellung gilt für Lambda-Funktionszuordnungen.

Weitere Informationen finden Sie unter Text einbeziehen.

Distribution Settings (Einstellungen für die Verteilung)

Die folgenden Werte gelten für die gesamte Verteilung.

Themen

Preisklasse

Wählen Sie die Preisklasse, die dem Höchstpreis entspricht, den Sie für den CloudFront Service zahlen möchten. Standardmäßig werden Ihre Objekte von Randstandorten in allen CloudFront Regionen aus CloudFront bedient.

Weitere Informationen zu Preisklassen und dazu, wie sich Ihre Wahl der Preisklasse auf die CloudFront Leistung Ihres Vertriebs auswirkt, finden Sie unter CloudFront Preisgestaltung.

AWS WAF Web-ACL

Sie können Ihre CloudFront Distribution mit AWS WAFeiner Firewall für Webanwendungen schützen, mit der Sie Ihre Webanwendungen sichern und APIs Anfragen blockieren können, bevor sie Ihre Server erreichen. Dies ist möglichAWS WAF Für Distributionen aktivieren, wenn Sie eine CloudFront Distribution erstellen oder bearbeiten.

Optional können Sie später in der AWS WAF Konsole unter zusätzliche Sicherheitsvorkehrungen für andere anwendungsspezifische Bedrohungen konfigurieren. https://console.aws.amazon.com/wafv2/

Weitere Informationen AWS WAF dazu finden Sie im AWS WAF Entwicklerhandbuch.

Alternative Domainnamen (CNAMEs)

Optional. Geben Sie einen oder mehrere Domainnamen an, die Sie URLs für Ihre Objekte verwenden möchten, und nicht den Domänennamen, den Sie bei der Erstellung Ihrer Distribution CloudFront zugewiesen haben. Sie müssen Eigentümer des Domainnamens sein oder über die Autorisierung verfügen, ihn zu verwenden. Dies überprüfen Sie, indem Sie ein SSL/TLS Zertifikat hinzufügen.

Wenn beispielsweise die URL für das Objekt:

```
/images/image.jpg
```

wie folgt angezeigt werden soll:

```
https://www.example.com/images/image.jpg
```

und nicht wie folgt:

```
https://d111111abcdef8.cloudfront.net/images/image.jpg
```

Fügen Sie einen CNAME für `www.example.com` hinzu.

> **Important**
>
> Wenn Sie einen CNAME für `www.example.com` zu Ihrer Verteilung hinzufügen, müssen Sie auch die folgenden Schritte ausführen:

- Erstellen (oder aktualisieren) Sie einen CNAME-Datensatz für Ihren DNS-Service, um Abfragen für `www.example.com` an `d111111abcdef8.cloudfront.net` weiterzuleiten.
- Fügen Sie ein Zertifikat CloudFront von einer vertrauenswürdigen Zertifizierungsstelle (CA) hinzu, das den Domainnamen (CNAME) abdeckt, den Sie Ihrer Distribution hinzufügen, um Ihre Autorisierung zur Verwendung des Domainnamens zu überprüfen.

Sie müssen die Berechtigung besitzen, beim DNS-Dienstanbieter für die Domäne einen CNAME-Datensatz zu erstellen. Das bedeutet in der Regel, dass Sie der Besitzer der Domäne sind oder Anwendungen für den Besitzer der Domäne entwickeln.

Informationen zur aktuell gültigen maximalen Anzahl von alternativen Domänennamen, die Sie einer Verteilung hinzufügen können, oder zum Anfordern eines höheren Kontingents (früher als Limit bezeichnet) finden Sie unter Allgemeine Kontingente für Verteilungen.

Weitere Informationen zu alternativen Domänennamen finden Sie unter Verwenden Sie Benutzerdefiniert, URLs indem Sie alternative Domainnamen hinzufügen (CNAMEs). Weitere Informationen zu finden Sie CloudFront URLs unterPassen Sie das URL-Format für Dateien an in CloudFront.

SSL-Zertifikat

Wenn Sie einen alternativen Domänennamen angegeben haben, der für Ihre Verteilung verwendet werden soll, wählen Sie Custom SSL Certificate (Benutzerdefiniertes SSL-Zertifikat) und anschließend ein Zertifikat aus, das diesen Domänennamen abdeckt, um Ihre Berechtigung zur Verwendung des alternativen Domänennamens zu bestätigen. Wenn Sie möchten, dass Viewer HTTPS für den Zugriff auf Ihre Objekte verwenden, wählen Sie die entsprechende Einstellung aus.

- CloudFront Standardzertifikat (*.cloudfront.net) — Wählen Sie diese Option, wenn Sie den CloudFront Domainnamen in URLs für Ihre Objekte verwenden möchten, z. B. `https://d111111abcdef8.cloudfront.net/image1.jpg`
- Benutzerdefiniertes SSL-Zertifikat — Wählen Sie diese Option, wenn Sie Ihren eigenen Domainnamen in der URLs für Ihre Objekte als alternativen Domainnamen verwenden möchten, z. B. `https://example.com/image1.jpg` Wählen Sie anschließend ein Zertifikat aus, das den alternativen Domänennamen abdeckt. Die Liste der Zertifikate kann folgende Arten von Zertifikaten enthalten:

- Zertifikate bereitgestellt von AWS Certificate Manager
- Zertifikate, die Sie von einer externen Zertifizierungsstelle erworben und zu ACM hochgeladen haben
- Zertifikate, die Sie von einer externen Zertifizierungsstelle erworben und zum IAM-Zertifikatspeicher hochgeladen haben

Wenn Sie diese Einstellung wählen, empfehlen wir, dass Sie in Ihrem Objekt nur einen alternativen Domainnamen verwenden URLs (https://example.com/logo.jpg). If you use your CloudFront distribution domain name (https://d111111abcdef8.cloudfront.net/logo.jpg) und ein Client einen älteren Viewer verwendet, der SNI nicht unterstützt. Wie der Viewer reagiert, hängt von dem Wert ab, den Sie für Unterstützte Clients wählen:

- Alle Clients: Der Viewer zeigt eine Warnung an, da der Domainname nicht mit dem Domainnamen in Ihrem Zertifikat übereinstimmt. CloudFront SSL/TLS
- Nur Clients, die Server Name Indication (SNI) Support: CloudFront unterbricht die Verbindung mit dem Viewer, ohne das Objekt zurückzugeben.

Benutzerdefinierte SSL-Clientunterstützung

Gilt nur, wenn Sie Benutzerdefiniertes SSL-Zertifikat (example.com) als SSL-Zertifikat wählen. Wenn Sie einen oder mehrere alternative Domainnamen und ein benutzerdefiniertes SSL-Zertifikat für die Verteilung angegeben haben, wählen Sie aus, wie Sie HTTPS-Anfragen bearbeiten CloudFront möchten:

- Clients that Support Server Name Indication (SNI) - (Recommended) (Clients, die SNI (Server Name Indication) unterstützen – (Empfohlen)): Mit dieser Einstellung können nahezu alle modernen Webbrowser und Clients eine Verbindung zur Verteilung herstellen, da sie SNI unterstützen. Einige Viewer verwenden jedoch möglicherweise ältere Webbrowser oder Clients, die SNI nicht unterstützen, was bedeutet, dass sie keine Verbindung zur Verteilung herstellen können.

  Um diese Einstellung mithilfe der CloudFront API anzuwenden, geben Sie dies `sni-only` in dem `SSLSupportMethod` Feld an. In AWS CloudFormation wird das Feld `SslSupportMethod` genannt (Beachten Sie die geänderte Groß-/Kleinschreibung).
- Legacy Clients Support (Unterstützung für ältere Clients): Mit dieser Einstellung können ältere Webbrowser und Clients, die SNI nicht unterstützen, eine Verbindung zur Distribution herstellen. Für diese Einstellung fallen jedoch zusätzliche monatliche Gebühren an. Den genauen Preis

finden Sie auf der CloudFront Amazon-Preisseite und suchen Sie auf der Seite nach Dedicated IP Custom SSL.

Um diese Einstellung mithilfe der CloudFront API anzuwenden, geben Sie dies `vip` in das `SSLSupportMethod` Feld ein. In ist AWS CloudFormation das Feld benannt `SslSupportMethod` (beachten Sie die unterschiedliche Groß-/Kleinschreibung).

Weitere Informationen finden Sie unter Wählen Sie aus, wie CloudFront HTTPS-Anfragen bearbeitet werden.

Sicherheitsrichtlinie (Mindestversion von SSL/TLS)

Geben Sie die Sicherheitsrichtlinie CloudFront an, die Sie für HTTPS-Verbindungen mit Zuschauern (Clients) verwenden möchten. Eine Sicherheitsrichtlinie bestimmt zwei Einstellungen:

- Das SSL/TLS Mindestprotokoll, das für die Kommunikation mit Zuschauern CloudFront verwendet wird.
- Die Chiffren, mit denen der Inhalt verschlüsselt werden CloudFront kann, der an die Zuschauer zurückgegeben wird.

Weitere Informationen zu den Sicherheitsrichtlinien einschließlich der jeweils enthaltenen Protokolle und Verschlüsselungen finden Sie unter Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront.

Die verfügbaren Sicherheitsrichtlinien hängen von den Werten ab, die Sie für SSL-Zertifikate und benutzerdefinierte SSL-Clientunterstützung (bekannt als `CloudFrontDefaultCertificate` und `SSLSupportMethod` in der CloudFront API) angeben:

- Wenn das SSL-Zertifikat das CloudFront Standardzertifikat (*.cloudfront.net) ist (wenn `CloudFrontDefaultCertificate` es `true` in der API enthalten ist), wird die CloudFront Sicherheitsrichtlinie automatisch auf festgelegt. TLSv1
- Wenn SSL Certificate (SSL-Zertifikat) auf Custom SSL Certificate (example.com) (Benutzerdefiniertes SSL-Zertifikat (example.com)) und Custom SSL Client Support (Benutzerdefinierte SSL-Client-Unterstützung) auf Clients that Support Server Name Indication (SNI) - (Recommended) (Clients, die Server Name Indication (SNI) unterstützen – (empfohlen)) eingestellt ist (wenn `CloudFrontDefaultCertificate` in der API auf `false` und `SSLSupportMethod` auf `sni-only` festgelegt ist), können Sie aus den folgenden Sicherheitsrichtlinien auswählen:

- TLSv1.2_2021
- TLSv1.2_2019
- TLSv1.2_2018
- TLSv1.1_2016
- TLSv1_2016
- TLSv1

- Wenn SSL Certificate (SSL-Zertifikat) auf Custom SSL Certificate (example.com) (Benutzerdefiniertes SSL-Zertifikat (example.com)) und Custom SSL Client Support (Benutzerdefinierte SSL-Client-Unterstützung) auf Legacy Clients Support (Unterstützung für Legacy-Clients) eingestellt ist (wenn `CloudFrontDefaultCertificate` in der API auf `false` und `SSLSupportMethod` auf `vip` festgelegt ist), können Sie aus den folgenden Sicherheitsrichtlinien auswählen:

  - TLSv1
  - SSLv3

  In dieser Konfiguration sind die Sicherheitsrichtlinien TLSv1 .2_2021, TLSv1 .2_2019, TLSv1 .2_2018, TLSv1 .1_2016 und TLSv1 _2016 nicht in der Konsole oder API verfügbar. CloudFront Wenn Sie eine dieser Sicherheitsrichtlinien verwenden möchten, stehen Ihnen folgende Optionen zur Verfügung:

  - Prüfen Sie, ob Ihre Verteilung Unterstützung für Legacy-Clients mit dedizierten IP-Adressen benötigt. Wenn Ihre Viewer Server Name Indication (SNI) unterstützen, empfehlen wir, die Einstellung Custom SSL Client Support (Benutzerdefinierte SSL-Client-Unterstützung) Ihrer Verteilung auf Clients that Support Server Name Indication (SNI) (Clients, die Server Name Indication (SNI) unterstützen) zu aktualisieren (`SSLSupportMethod` in der API auf `sni-only` einzustellen). Auf diese Weise können Sie alle verfügbaren TLS-Sicherheitsrichtlinien verwenden, und es kann auch Ihre Gebühren senken. CloudFront
  - Wenn Sie Legacy Clients Support mit dedizierten IP-Adressen beibehalten müssen, können Sie eine der anderen TLS-Sicherheitsrichtlinien (TLSv1.2_2021, TLSv1 .2_2019, TLSv1 .2_2018, TLSv1 .1_2016 oder TLSv1 _2016) anfordern, indem Sie im Support Center einen Fall erstellen.AWS

    > **Note**
    >
    > Bevor Sie sich an den AWS Support wenden, um diese Änderung zu beantragen, sollten Sie Folgendes beachten:

- Wenn Sie eine dieser Sicherheitsrichtlinien (TLSv1.2_2021, TLSv1 .2_2019, TLSv1 .2_2018, TLSv1 .1_2016 oder TLSv1 _2016) zu einer Legacy Clients Support-Distribution hinzufügen, wird die Sicherheitsrichtlinie auf alle Nicht-SNI-Viewer-Anfragen für alle Legacy Clients Support-Distributionen in Ihrem Konto angewendet. AWS Wenn Viewer jedoch SNI-Anforderungen an eine Verteilung mit Unterstützung für Legacy-Clients senden, gilt die Sicherheitsrichtlinie dieser Verteilung. Um sicherzustellen, dass Ihre gewünschte Sicherheitsrichtlinie auf alle Zuschaueranfragen angewendet wird, die an alle Legacy Clients Support-Distributionen in Ihrem AWS Konto gesendet werden, fügen Sie die gewünschte Sicherheitsrichtlinie jeder Distribution einzeln hinzu.
- Per Definition unterstützt die neue Sicherheitsrichtlinie nicht dieselben Verschlüsselungen und Protokolle wie die alte. Wenn Sie sich beispielsweise dafür entscheiden, die Sicherheitsrichtlinie einer Distribution von TLSv1 auf TLSv1 .1_2016 zu aktualisieren, unterstützt diese Distribution die DES- CBC3 -SHA-Verschlüsselung nicht mehr. Weitere Informationen zu den Verschlüsselungen und Protokollen, die von den einzelnen Sicherheitsrichtlinien unterstützt werden, finden Sie unter Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront.

Unterstützte HTTP-Versionen

Wählen Sie die HTTP-Versionen aus, die Ihre Distribution unterstützen soll, wenn Zuschauer mit ihnen kommunizieren. CloudFront

Damit Zuschauer HTTP/2 verwenden können, müssen die Zuschauer TLSv1 2.2 oder höher und Server Name Indication (SNI) unterstützen. CloudFront

Zuschauer müssen HTTP/3 und Server Name Indication (SNI TLSv1) unterstützen, CloudFront damit sie HTTP/3 verwenden können. CloudFront unterstützt die HTTP/3-Verbindungsmigration, sodass der Zuschauer zwischen Netzwerken wechseln kann, ohne die Verbindung zu verlieren. Weitere Angaben zur Verbindungsmigration finden Sie in den Informationen zur Verbindungsmigration in RFC 9000.

> **Note**
>
> Weitere Hinweise zu unterstützten TLSv1 3.3-Chiffren finden Sie unter. Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront

> **Note**
>
> Wenn Sie Amazon Route 53 verwenden, können Sie HTTPS-Einträge verwenden, um Protokollverhandlungen als Teil der DNS-Suche zu ermöglichen, sofern der Client dies unterstützt. Weitere Informationen finden Sie unter Create alias resource record set.

Standardstammobjekt

Optional. Das Objekt, das Sie von Ihrem Ursprung aus anfordern CloudFront möchten (z. B.`index.html`), wenn ein Betrachter die Stamm-URL Ihrer Distribution (`https://www.example.com/`) anstelle eines Objekts in Ihrer Distribution (`https://www.example.com/product-description.html`) anfordert. Durch die Festlegung eines Angeben eines Standardstammobjekt wird vermieden, dass die Inhalte Ihrer Verteilung preisgegeben werden.

Die maximale Länge des Namens beträgt 255 Zeichen. Der Name kann folgende Zeichen enthalten:

- A-Z, a-z
- 0-9
- _ - . * $ / ~ " '
- & (übergeben und zurückgegeben als `&`)

Geben Sie bei der Angabe des Standardstammobjekts nur den Objektnamen ein, z. B. `index.html`. Fügen Sie keinen `/` vor dem Objektnamen hinzu.

Weitere Informationen finden Sie unter Geben Sie ein Standard-Stammobjekt an.

Standardprotokollierung

Geben Sie an CloudFront , ob Sie Informationen zu jeder Anforderung für ein Objekt protokollieren und die Protokolldateien speichern möchten. Sie können die Protokollierung jederzeit aktivieren oder deaktivieren. Wenn Sie die Protokollierung aktivieren, fallen keine zusätzlichen Kosten an, aber es können Gebühren für das Speichern und Zugreifen auf die Dateien anfallen. Sie können die Protokolle jederzeit löschen.

CloudFront unterstützt die folgenden Standard-Protokollierungsoptionen:

- Standardprotokollierung (v2) — Sie können Protokolle an Lieferziele wie Amazon CloudWatch Logs, Amazon Data Firehose und Amazon Simple Storage Service (Amazon S3) senden.

- Standardprotokollierung (veraltet) — Sie können Protokolle nur an einen Amazon S3 S3-Bucket senden.

Protokollpräfix

(Optional) Wenn Sie die Standardprotokollierung (Legacy) aktivieren, geben Sie gegebenenfalls die Zeichenfolge CloudFront an, die Sie den Namen der Zugriffsprotokolldateien für diese Distribution voranstellen möchten, `exampleprefix/` z. B. Der abschließende Schrägstrich (/) ist optional, jedoch empfohlen, um das Durchsuchen Ihrer Protokolldateien zu vereinfachen. Weitere Informationen finden Sie unter Standardprotokollierung konfigurieren (Legacy).

Protokollierung von Cookies

Wenn Sie Cookies in CloudFront die Zugriffsprotokolle aufnehmen möchten, wählen Sie On. Wenn Sie Cookies in die Protokolle aufnehmen möchten, werden alle Cookies unabhängig davon CloudFront protokolliert, wie Sie das Cache-Verhalten für diese Verteilung konfigurieren: alle Cookies weiterleiten, keine Cookies weiterleiten oder eine bestimmte Liste von Cookies an den Ursprung weiterleiten.

Amazon S3 verarbeitet keine Cookies. Sofern Ihre Distribution nicht auch Amazon EC2 oder eine andere benutzerdefinierte Herkunft beinhaltet, empfehlen wir Ihnen, für den Wert von Cookie Logging die Option Aus zu wählen.

Weitere Informationen zu Cookies finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.

Aktivieren IPv6

IPv6 ist eine neue Version des IP-Protokolls. Es ist der letztendliche Ersatz für einen größeren Adressraum IPv4 und verwendet einen größeren Adressraum. CloudFront reagiert immer auf IPv4 Anfragen. Wenn CloudFront Sie auf Anfragen von IPv4 IP-Adressen (wie 192.0.2.44) und Anfragen von IPv6 Adressen (wie 2001:0 db 8:85 a3: :8a2e: 0370:7334) antworten möchten, wählen Sie Aktivieren. IPv6

Im Allgemeinen sollten Sie die Option aktivieren IPv6 , wenn Sie Benutzer in Netzwerken haben, die auf Ihre Inhalte zugreifen möchten. IPv6 Wenn Sie jedoch signierte URLs oder signierte Cookies verwenden, um den Zugriff auf Ihre Inhalte einzuschränken, und wenn Sie eine benutzerdefinierte Richtlinie verwenden, die den `IpAddress` Parameter zum Einschränken der IP-Adressen enthält, die auf Ihre Inhalte zugreifen können, sollten Sie diese Option nicht aktivieren IPv6. Wenn Sie

den Zugriff auf bestimmte Inhalte nach IP-Adresse einschränken und den Zugriff auf andere Inhalte nicht einschränken möchten (oder den Zugriff einschränken möchten, jedoch nicht nach IP-Adresse), können Sie zwei Verteilungen erstellen. Informationen zum Erstellen signierter URLs mithilfe einer benutzerdefinierten Richtlinie finden Sie unterErstellen Sie eine signierte URL mithilfe einer benutzerdefinierten Richtlinie. Informationen zum Erstellen von signierten Cookies mit einer benutzerdefinierten Richtlinie finden Sie unter Legen Sie signierte Cookies mithilfe einer benutzerdefinierten Richtlinie fest.

Wenn Sie einen Route 53-Aliasressourcendatensatz verwenden, um den Verkehr an Ihre CloudFront Verteilung weiterzuleiten, müssen Sie einen zweiten Alias-Ressourcendatensatz erstellen, wenn beide der folgenden Bedingungen zutreffen:

- Sie aktivieren IPv6 für die Verteilung
- Sie verwenden alternative Domainnamen in der URLs für Ihre Objekte

Weitere Informationen finden Sie unter Weiterleiten von Traffic an eine CloudFront Amazon-Distribution mithilfe Ihres Domainnamens im Amazon Route 53-Entwicklerhandbuch.

Wenn Sie einen CNAME-Ressourcendatensatz über Route 53 oder einen anderen DNS-Service erstellt haben, müssen Sie keine Änderungen ausführen. Ein CNAME-Datensatz leitet den Datenverkehr unabhängig von dem IP-Adressformat der Viewer-Anforderung an Ihre Verteilung weiter.

Wenn Sie Protokolle aktivieren IPv6 und CloudFront darauf zugreifen, enthält die `c-ip` Spalte Werte im IPv6 Format IPv4 und Format. Weitere Informationen finden Sie unter Felder in der Protokolldatei.

> **Note**
>
> Um eine hohe Kundenverfügbarkeit zu gewährleisten, CloudFront beantwortet es Anfragen von Zuschauern unter IPv4 Angabe der Daten, IPv4 die für ein besseres Nutzererlebnis sprechen. Um herauszufinden, wie viel Prozent der Anfragen CloudFront zugestellt werdenIPv6, aktivieren Sie die CloudFront Protokollierung für Ihre Distribution und analysieren Sie die `c-ip` Spalte, die die IP-Adresse des Betrachters enthält, der die Anfrage gestellt hat. Dieser Prozentsatz sollte im Laufe der Zeit steigen, aber er wird weiterhin eine Minderheit des Traffics sein, da IPv6 er noch nicht von allen Zuschauernetzwerken weltweit unterstützt wird. Einige Zuschauernetzwerke bieten eine hervorragende IPv6 Unterstützung, andere dagegen IPv6 überhaupt nicht. (Ein Viewer-Netzwerk ist mit dem Betreiber Ihres stationären Internets bzw. Ihrem Mobilfunkbetreiber vergleichbar.)

Weitere Informationen zu unserem Support für IPv6 finden Sie in den CloudFront häufig gestellten Fragen. Informationen zur Aktivierung von Zugriffsprotokollen finden Sie in den Feldern Standardprotokollierung undProtokollpräfix.

Kommentar

Optional. Wenn Sie eine Verteilung erstellen, können Sie einen Kommentar mit einer Länge von bis zu 128 Zeichen einfügen. Sie können den Kommentar jederzeit aktualisieren.

Status der Verteilung

Gibt an, ob die Verteilung nach der Bereitstellung aktiv oder inaktiv sein soll:

- Enabled bedeutet, dass Sie sofort Links mit dem Domänennamen der Verteilung verwenden können und dass Benutzer die Inhalte darüber abrufen können, sobald die Verteilung vollständig bereitsteht. Wenn eine Verteilung aktiviert ist, akzeptiert und verarbeitet CloudFront alle Anfragen zu diesen Inhalten von Endbenutzern, die den Domänennamen verwenden, welcher der Verteilung zugewiesen wurde.

  Wenn Sie eine CloudFront Distribution erstellen, ändern oder löschen, dauert es einige Zeit, bis Ihre Änderungen in der CloudFront Datenbank übernommen werden. Bei einer unmittelbaren Anfrage für Informationen zu einer Verteilung wird die Änderung möglicherweise nicht angezeigt. Die Weiterleitung wird in der Regel innerhalb weniger Minuten abgeschlossen. Dieser Zeitraum kann sich bei einer hohen Zeitauslastung oder Netzwerkpartition jedoch verlängern.
- Disabled bedeutet, dass die Verteilung möglicherweise bereitgestellt und betriebsbereit ist, aber Benutzer sie noch nicht verwenden können. Immer wenn eine Verteilung deaktiviert ist, akzeptiert CloudFront sie keine Anfragen von Endbenutzern, die den mit dieser Verteilung verknüpften Domainnamen verwenden. Die Verteilung kann erst verwendet werden, wenn Sie den Status von Deaktiviert zu Aktiviert ändern (indem Sie die Konfiguration der Verteilung aktualisieren).

Sie können in Bezug auf den Status einer Verteilung so oft zwischen Deaktiviert und Aktiviert wechseln, wie Sie möchten. Führen Sie die Schritte zum Aktualisieren der Konfiguration einer Verteilung aus. Weitere Informationen finden Sie unter Eine Verteilung aktualisieren.

Benutzerdefinierte Fehlerseiten und Zwischenspeicherung von Fehlern

Sie können ein Objekt an den Viewer CloudFront zurücksenden lassen (z. B. eine HTML-Datei), wenn Ihr Amazon S3- oder benutzerdefinierter Origin einen HTTP-Statuscode 4xx oder 5xx-an zurückgibt. CloudFront Sie können auch angeben, wie lange eine Fehlerantwort von Ihrem Absender oder einer benutzerdefinierten Fehlerseite in CloudFront Edge-Caches zwischengespeichert wird. Weitere Informationen finden Sie unter Erstellen Sie eine benutzerdefinierte Fehlerseite für bestimmte HTTP-Statuscodes.

> **Note**
>
> Die folgenden Werte sind im Assistenten zum Erstellen von Verteilungen nicht enthalten, deshalb können Sie benutzerdefinierte Fehlerseiten nur konfigurieren, wenn Sie eine Verteilung aktualisieren.

Themen

HTTP-Fehlercode

Der HTTP-Statuscode, für den Sie eine benutzerdefinierte Fehlerseite zurückgeben CloudFront möchten. Sie können so konfigurieren CloudFront , dass benutzerdefinierte Fehlerseiten für keinen, einige oder alle CloudFront zwischengespeicherten HTTP-Statuscodes zurückgegeben werden.

Pfad zur Antwortseite

Der Pfad zu der benutzerdefinierten Fehlerseite (z. B. `/4xx-errors/403-forbidden.html`), die CloudFront an Betrachter zurückgeben soll, wenn Ihr Ursprung den HTTP-Statuscode zurückgibt, den Sie für Error Code angegeben haben (z. B. 403). Wenn Sie Ihre Objekte und Ihre benutzerdefinierten Fehlerseiten an verschiedenen Orten speichern möchten, muss Ihre Verteilung ein Cache-Verhalten mit den folgenden Eigenschaften enthalten:

- Der Wert von Path Pattern stimmt mit dem Pfad zu Ihren benutzerdefinierten Fehlermeldungen überein. Angenommen, Sie haben benutzerdefinierte Fehlerseiten für 4xx-Fehler in einem Amazon S3-Bucket in einem Verzeichnis mit dem Namen gespeicher `/4xx-errors`. Ihre Verteilung muss ein Cache-Verhalten beinhalten, für welches das Pfadmuster Anfragen für Ihre benutzerdefinierten Fehlerseiten an diesen Ort weiterleitet, z. B. /4xx-errors/*.
- Der Wert von Origin legt den Wert von Origin ID für den Ursprung fest, der Ihre benutzerdefinierten Fehlerseiten enthält.

HTTP-Antwortcode

Der HTTP-Statuscode, den Sie zusammen mit der benutzerdefinierten Fehlerseite an den Viewer zurückgeben möchten CloudFront .

Mindest-TTL für die Zwischenspeicherung von Fehlern (Sekunden)

Die Mindestdauer, für die Sie Fehlerantworten von Ihrem Ursprungsserver zwischenspeichern möchten CloudFront .

Geografische Einschränkungen

Wenn Sie verhindern möchten, dass Benutzer in ausgewählten Ländern auf Ihre Inhalte zugreifen, können Sie Ihre CloudFront Distribution mit einer Zulassungsliste oder einer Sperrliste konfigurieren. Für die Konfiguration von geografischen Einschränkungen fallen keine zusätzlichen Kosten an. Weitere Informationen finden Sie unter Beschränken Sie die geografische Verteilung Ihrer Inhalte.

Testen Sie eine Distribution

Nachdem Sie Ihre Distribution erstellt haben, CloudFront weiß er, wo sich Ihr Ursprungsserver befindet, und Sie kennen den Domainnamen, der mit der Distribution verknüpft ist. Gehen Sie wie folgt vor, um Ihre Distribution zu testen:

1. Warten Sie, bis Ihre Distribution bereitgestellt ist.
   - Sehen Sie sich Ihre Distributionsdetails in der Konsole an. Wenn Ihre Distribution mit der Bereitstellung fertig ist, ändert sich das Feld Letzte Änderung von Bereitstellen in ein Datum und eine Uhrzeit.
2. Verwenden Sie das folgende Verfahren, um Links zu Ihren Objekten mit dem CloudFront Domänennamen zu erstellen.

3. Testen Sie die Links. CloudFront stellt die Objekte Ihrer Webseite oder Anwendung zur Verfügung.

Erstellen Sie Links zu Ihren Objekten

Gehen Sie wie folgt vor, um Testlinks für die Objekte in Ihrer CloudFront Webdistribution zu erstellen.

So erstellen Sie Links zu Objekten in einer Web-Verteilung

1. Kopieren Sie den folgenden HTML-Code in eine neue Datei, *domain-name* ersetzen Sie ihn durch den Domainnamen Ihrer Distribution und *object-name* ersetzen Sie ihn durch den Namen Ihres Objekts.

   ```
   <html>
   <head>
    <title>My CloudFront Test</title>
   </head>
   <body>
    <p>My text content goes here.</p>
    <p><img src="https://domain-name/object-name" alt="my test image"></p>
   </body>
   </html>
   ```

   Wenn Ihr Domänenname beispielsweise `d111111abcdef8.cloudfront.net` ist und Ihr Objekt `image.jpg`, würde die URL für den Link folgendermaßen lauten:

   `https://d111111abcdef8.cloudfront.net/image.jpg`.

   Wenn sich das Objekt in einem Ordner auf Ihrem Ursprungs-Server befindet, muss der Ordner ebenfalls in der URL enthalten sein. Wenn sich image.jpg beispielsweise im Ordner „images” auf Ihrem Ursprungs-Server befindet, würde die URL folgendermaßen lauten:

   `https://d111111abcdef8.cloudfront.net/images/image.jpg`
2. Speichern Sie den HTML-Code unter einem Dateinamen mit der Erweiterung .html.
3. Öffnen Sie Ihre Webseite in einem Browser, um zu verifizieren, dass das Objekt angezeigt wird.

Der Browser gibt Ihre Seite mit der eingebetteten Bilddatei zurück, die von der Edge-Position aus bereitgestellt wurde, CloudFront die für das Objekt als geeignet erachtet wurde.

Eine Verteilung aktualisieren

In der CloudFront Konsole können Sie die CloudFront Distributionen sehen, die mit Ihrer Distribution verknüpft sind AWS-Konto, die Einstellungen für eine Distribution einsehen und die meisten Einstellungen aktualisieren. Beachten Sie, dass die von Ihnen vorgenommenen Einstellungsänderungen erst wirksam werden, nachdem die Verteilung an die AWS -Edge-Standorte weitergegeben wurde.

Aktualisieren Sie eine Distribution in der Konsole

Die folgenden Verfahren zeigen Ihnen, wie Sie eine CloudFront Distribution in der Konsole aktualisieren.

Multi-tenant

Um eine Multi-Tenant-Distribution zu aktualisieren

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Suchen Sie nach der ID der Multi-Tenant-Distribution und wählen Sie sie aus.
3. Wählen Sie die Registerkarte für die Einstellungen, die Sie aktualisieren möchten.
4. Nehmen Sie die Aktualisierungen vor, und wählen Sie dann Änderungen speichern, um Ihre Änderungen zu speichern. Weitere Informationen zu den Einstellungen, die Sie aktualisieren können, finden Sie unterReferenz für vorkonfigurierte Verteilungseinstellungen.

Sie können eine Distribution auch mithilfe der CloudFront API aktualisieren:

- Informationen zum Aktualisieren einer Distribution finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.

> **Important**
>
> Beachten Sie bei der Aktualisierung Ihrer Distribution, dass eine Reihe zusätzlicher Felder erforderlich sind, die bei der ersten Erstellung einer Distribution jedoch nicht erforderlich sind. Um sicherzustellen, dass alle erforderlichen Felder enthalten sind, wenn

Sie die CloudFront API zum Aktualisieren einer Distribution verwenden, folgen Sie den UpdateDistributionin der Amazon CloudFront API-Referenz beschriebenen Schritten.

Um die Mehrmandantenverteilung für einen Distributionsmandanten zu ändern, aktualisieren Sie den Distributionsmandanten. Sie aktualisieren auch den Verteilungsmandanten, um dessen Domäne, Zertifikat, Anpassungen oder Parameterwerte zu aktualisieren. Weitere Informationen zur Aktualisierung des Zertifikats für den Verteilungsmandanten finden Sie unterFügen Sie eine Domäne und ein Zertifikat hinzu (Distributionsmandant).

So aktualisieren Sie einen Verteilungsmandanten

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie unter SaaS die Option Distribution Tenants aus.
3. Suchen Sie nach dem Distributionsmandanten. Verwenden Sie das Dropdownmenü in der Suchleiste, um nach Domäne, Name, Distributions-ID, Zertifikat-ID, Verbindungsgruppen-ID oder Web-ACL-ID zu filtern.
4. Wählen Sie den Namen des Distributionsmandanten aus.
5. Um die allgemeinen Details zu aktualisieren, wählen Sie Bearbeiten, nehmen Sie die Aktualisierungen vor und wählen Sie dann Verteilungsmandant aktualisieren aus.
6. Wählen Sie die entsprechende Registerkarte für alle anderen Einstellungen, die Sie aktualisieren möchten, nehmen Sie Ihre Aktualisierungen vor und speichern Sie sie. Weitere Informationen zu den Einstellungen für den Verteilungsmandanten, die Sie anpassen können, finden Sie unterAnpassungen des Distributionsmandanten.

Standard

So aktualisieren Sie eine Standarddistribution

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die ID einer Verteilung aus. Die Liste enthält alle Distributionen, die dem AWS Konto zugeordnet sind, mit dem Sie sich bei der CloudFront Konsole angemeldet haben.
3. Um die allgemeinen Einstellungen zu aktualisieren, wählen Sie Edit (Bearbeiten) aus. Wählen Sie andernfalls die Registerkarte für die Einstellungen, die Sie aktualisieren möchten.

4. Nehmen Sie die Aktualisierungen vor und wählen Sie dann Änderungen speichern. Informationen zu den einzelnen Feldern finden Sie unter den folgenden Themen:

   - General settings (Allgemeine Einstellungen: Distribution Settings (Einstellungen für die Verteilung)
   - Origin settings (Ursprungseinstellungen: Ursprungseinstellungen
   - Cache behavior settings (Einstellungen für das Cache-Verhalten: Einstellungen für das Cache-Verhalten

5. Wenn Sie einen Ursprung in Ihrer Verteilung löschen möchten, gehen Sie wie folgt vor:

   a. Wählen Sie Behaviors (Verhalten) und stellen Sie sicher, dass Sie alle dem Ursprung zugeordneten Standard-Cache-Verhalten auf einen anderen Ursprung verschoben haben.

   b. Wählen Sie Origins (Ursprünge) und wählen Sie dann einen Ursprung aus.

   c. Wählen Sie Delete (Löschen).

Sie können eine Distribution auch mithilfe der CloudFront API aktualisieren:

- Informationen zum Aktualisieren einer Distribution finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.

> **Important**
>
> Wenn Sie eine Verteilung aktualisieren, sollten Sie daran denken, dass eine Reihe von zusätzlichen Felder erforderlich sind, die bei der Erstellung der Verteilung nicht erforderlich sind. Um sicherzustellen, dass alle erforderlichen Felder enthalten sind, wenn Sie die CloudFront API zum Aktualisieren einer Distribution verwenden, folgen Sie den UpdateDistributionin der Amazon CloudFront API-Referenz beschriebenen Schritten.

Wenn Sie Änderungen an Ihrer Distributionskonfiguration speichern, CloudFront beginnt die Übertragung der Änderungen auf alle Edge-Standorte. Aufeinanderfolgende Konfigurationsänderungen werden in ihrer jeweiligen Reihenfolge verbreitet. Solange die Konfiguration an einem Edge-Standort aktualisiert wird, stellt CloudFront Ihre Inhalte von diesem Standort aus auf Basis der vorherigen Konfiguration bereit. Wenn die Konfiguration an einem Edge-

Standort aktualisiert wurde, beginnt CloudFront sofort damit, Ihre Inhalte von diesem Standort aus auf Basis der neuen Konfiguration bereitzustellen.

Ihre Änderungen werden nicht gleichzeitig auf alle Edge-Standorte übertragen. Während CloudFront Ihre Änderungen übertragen werden, können wir anhand der vorherigen Konfiguration oder der neuen Konfiguration nicht feststellen, ob ein bestimmter Edge-Standort Ihre Inhalte bereitstellt.

> **Note**
>
> In seltenen Fällen, wenn ein Host oder eine Netzwerkverbindung unterbrochen wird, kann es vorkommen, dass ein Teil des Datenverkehrs für den Verteilermandanten für einen kurzen Zeitraum mit älteren Konfigurationen bedient wird, bis Ihre Änderungen im Netzwerk ankommen.

Um zu sehen, wann Ihre Änderungen übernommen werden, sehen Sie sich Ihre Distributionsdetails in der Konsole an. Das Feld Letzte Änderung ändert sich von Deploying zu einem Datum und einer Uhrzeit, zu der die Bereitstellung abgeschlossen ist.

Kennzeichnen Sie eine Distribution

Tags sind Wörter oder Ausdrücke, mit denen Sie Ihre AWS Ressourcen identifizieren und organisieren können. Sie können jeder Ressource mehrere Tags hinzufügen, und jedes Tag enthält einen Schlüssel und einen Wert, den Sie festlegen. Der Schlüssel könnte beispielsweise „domain“ heißen und der Wert „example.com“. Sie können die Ressourcen auf Grundlage der hinzugefügten Tags durchsuchen und filtern.

Sie können Tags zusammen verwenden CloudFront, z. B. in den folgenden Beispielen:

- Erzwingen Sie tagbasierte Berechtigungen für CloudFront Distributionen. Weitere Informationen finden Sie unter ABAC mit CloudFront.
- Verfolge Rechnungsinformationen in verschiedenen Kategorien. Wenn Sie Tags auf CloudFront Distributionen oder andere AWS Ressourcen (wie EC2 Amazon-Instances oder Amazon S3-Buckets) anwenden und die Tags aktivieren, wird ein Kostenzuordnungsbericht als kommagetrennter Wert (CSV-Datei) AWS generiert, in dem Ihre Nutzung und die Kosten nach Ihren aktiven Tags aggregiert werden.

  Sie können Tags anwenden, die geschäftliche Kategorien (wie Kostenstellen, Anwendungsnamen oder Eigentümer) darstellen, um die Kosten für mehrere Services zu organisieren. Weitere

Informationen zur Verwendung von Tags für die Kostenzuordnung finden Sie unter Verwenden von Kostenzuordnungs-Tags im AWS Billing -Benutzerhandbuch.

Hinweise

- Sie können Verteilungen mit Tags markieren, nicht jedoch Ursprungszugriffsidentitäten oder Invalidierungen.
- Tag-Editor und Ressourcengruppen werden derzeit nicht unterstützt. CloudFront
- Informationen zur zurzeit maximalen Anzahl von Tags, die Sie einer Verteilung hinzufügen können, finden Sie unter Allgemeine Kontingente.

Inhalt

Tag-Einschränkungen

Die folgenden grundlegenden Einschränkungen gelten für Tags (Markierungen):

- Die maximale Anzahl von Tags pro Distribution finden Sie unterAllgemeine Kontingente.
- Maximale Schlüssellänge – 128 Unicode-Zeichen.
- Maximale Wertlänge – 256 Unicode-Zeichen.
- Gültige Werte für Schlüssel- und Wert sind a-z, A-Z, 0-9, Leerzeichen und die folgenden Zeichen: _ . : / = + - und @.
- Bei Tag-Schlüsseln und -Werten muss die Groß-/Kleinschreibung beachtet werden
- Verwenden Sie `aws:` nicht als Präfix für Schlüssel. Dieses Präfix ist zur Verwendung in AWS reserviert.

Tags für Distributionen hinzufügen, bearbeiten und löschen

Sie können die CloudFront Konsole verwenden, um Tags für Ihre Distributionen zu verwalten.

So fügen Sie einer Verteilung Tags hinzu, bzw. bearbeiten oder löschen sie

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die ID für die Verteilung aus, die Sie aktualisieren möchten.
3. Wählen Sie die Registerkarte Tags aus.
4. Wählen Sie Tags verwalten aus.
5. Auf der Seite Tags verwalten haben Sie folgende Möglichkeiten:
   - Um ein Tag hinzuzufügen, geben Sie einen Schlüssel und optional einen Wert für das Tag ein. Wählen Sie Neues Tag hinzufügen, um weitere Tags hinzuzufügen.
   - Um ein Tag zu bearbeiten, ändern Sie den Schlüssel des Tags oder seinen Wert oder beides. Sie können den Wert für ein Tag löschen, aber der Schlüssel ist erforderlich.
   - Zum Entfernen eines Tags wählen Sie Remove (Entfernen).
6. Wählen Sie Änderungen speichern aus.

Programmatisches Tagging

Sie können auch die CloudFront API AWS Command Line Interface (AWS CLI), und verwenden AWS SDKs, AWS Tools for Windows PowerShell um Tags anzuwenden. Weitere Informationen finden Sie unter den folgenden Themen:

- CloudFront API-Operationen:
  - ListTagsForResource
  - TagResource
  - UntagResource
- AWS CLI — Siehe Cloudfront in der AWS CLI Befehlsreferenz
- AWS SDKs — Die entsprechende SDK-Dokumentation finden Sie auf der AWS Dokumentationsseite
- Tools für Windows PowerShell — siehe Amazon CloudFront in der AWS -Tools für PowerShell Cmdlet-Referenz

Löschen einer -Verteilung

Mit dem folgenden Verfahren wird eine Distribution mithilfe der CloudFront Konsole gelöscht. Informationen zum Löschen mit der CloudFront API finden Sie DeleteDistributionin der Amazon CloudFront API-Referenz.

Wenn Sie eine Distribution löschen müssen, bei der ein OAC an einen S3-Bucket angehängt ist, finden Sie wichtige Löschen Sie eine Distribution, bei der ein OAC an einen S3-Bucket angehängt ist Informationen unter.

> **Note**
>
> Wichtig: Bevor Sie eine Verteilung löschen können, müssen Sie sie deaktivieren. Hierzu benötigen Sie die Berechtigung zum Aktualisieren der Verteilung.
> Wenn Sie eine Distribution deaktivieren, der ein alternativer Domainname zugeordnet ist, wird kein Traffic CloudFront mehr für diesen Domainnamen akzeptiert (z. B. www.example.com), auch wenn eine andere Distribution einen alternativen Domainnamen mit einem Platzhalter (*) hat, der derselben Domain entspricht (z. B. *.example.com).

Multi-tenant

Bevor Sie eine Mehrmandantenverteilung löschen können, müssen Sie zunächst alle zugehörigen Verteilungsmandanten aus der Verteilung löschen.

Um eine Mehrmandantenverteilung zu löschen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im rechten Bereich der CloudFront Konsole den Namen der Mehrmandantenverteilung aus, die Sie löschen möchten.
3. Wählen Sie für Mandanten alle zugehörigen Distributionsmandanten aus und löschen Sie sie.
4. Wählen Sie Deaktivieren, um die Verteilung zu deaktivieren, und wählen Sie Verteilung deaktivieren, um zu bestätigen.
5. Warten Sie, bis der neue Zeitstempel in der Spalte Letzte Änderung angezeigt wird.

   - Es kann einige Minuten dauern, CloudFront bis Ihre Änderung auf alle Edge-Standorte übertragen ist.

6. Wählen Sie Löschen, Verteilung löschen aus.

Um einen Distributionsmandanten zu löschen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie unter SaaS die Option Distribution Tenants aus.
3. Suchen Sie nach dem Distributionsmandanten. Verwenden Sie das Dropdownmenü in der Suchleiste, um nach Domäne, Name, Distributions-ID, Zertifikat-ID, Verbindungsgruppen-ID oder Web-ACL-ID zu filtern.
4. Wählen Sie den zu löschenden Verteilungsmandanten aus.
5. Wählen Sie Mandant löschen, Verteilungsmandant löschen aus.

Standard

Um eine Standardverteilung zu löschen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Suchen Sie im rechten Bereich der CloudFront Konsole nach der Distribution, die Sie löschen möchten.

   - Wenn in der Spalte Status angezeigt wird, dass die Verteilung bereits deaktiviert ist, fahren Sie mit Schritt 6 fort.
   - Wenn der Status Aktiviert angezeigt wird, die Verteilung aber in der Spalte Letzte Änderung immer noch Bereitstellen anzeigt, warten Sie, bis die Bereitstellung abgeschlossen ist, bevor Sie mit Schritt 3 fortfahren.
3. Aktivieren Sie im rechten Bereich der CloudFront Konsole das Kontrollkästchen für die Distribution, die Sie löschen möchten.
4. Klicken Sie auf Disable (Deaktivieren), um die Verteilung zu deaktivieren, und wählen Sie Yes, Disable (Ja, deaktivieren) aus, um den Vorgang zu bestätigen. Wählen Sie anschließend Close (Schließen) aus.

   - Der Wert der Spalte Status ändert sich sofort in Deaktiviert.
5. Warten Sie, bis der neue Zeitstempel in der Spalte Letzte Änderung angezeigt wird.

- Es kann einige Minuten dauern, CloudFront bis Ihre Änderung auf alle Edge-Standorte übertragen ist.

6. Aktivieren Sie das Kontrollkästchen für die Verteilung, die Sie löschen möchten.
7. Wählen Sie Löschen, Löschen aus.

   - Wenn die Option Löschen nicht verfügbar ist, bedeutet dies, dass Ihre Änderung immer noch auf die Edge-Standorte übertragen CloudFront wird. Warten Sie, bis der neue Zeitstempel in der Spalte Letzte Änderung angezeigt wird, und wiederholen Sie dann die Schritte 6-7.

Verwenden Sie bei Verteilungen verschiedene Ursprünge CloudFront

Wenn Sie eine Distribution erstellen, geben Sie den Ursprung an, von dem Anfragen für die Dateien CloudFront gesendet werden. Sie können mehrere verschiedene Arten von Ursprüngen mit verwenden CloudFront. Sie können beispielsweise einen Amazon S3 S3-Bucket, einen MediaStore Container, einen MediaPackage Channel, einen Application Load Balancer oder eine AWS Lambda Funktions-URL verwenden. Wenn Sie Ihre CloudFront Distribution erstellen, konfiguriert die meisten Distributionseinstellungen CloudFront automatisch für Sie, basierend auf Ihrem Inhaltstyp. Weitere Informationen finden Sie unter Referenz für vorkonfigurierte Verteilungseinstellungen.

Wenn Sie einen Application Load Balancer, Network Load Balancer oder eine EC2 Instance in einem privaten Subnetz haben, können Sie ihn als VPC-Ursprung verwenden. Mit VPC-Ursprüngen kann auf Ihre Anwendungen nur in einem privaten Subnetz mit einer CloudFront Verteilung zugegriffen werden, wodurch verhindert wird, dass Ihre Anwendung im öffentlichen Internet zugänglich ist. Weitere Informationen finden Sie unter the section called “Beschränken Sie den Zugriff mit VPC-Ursprüngen”.

> **Note**
>
> Sie können Edge-Funktionen verwenden, um dynamisch den geeigneten Ursprung für jede Anfrage auszuwählen. Mithilfe von CloudFront Functions oder Lambda @Edge können Sie Anfragen anhand von Faktoren wie dem geografischen Standort des Betrachters, den Anforderungsheadern oder Abfragezeichenfolgenparametern an verschiedene Ursprünge

weiterleiten. Weitere Informationen finden Sie unter Personalisieren Sie am Rand mit Funktionen.

Themen

- Verwenden Sie einen Amazon S3 S3-Bucket
- Verwenden Sie einen MediaStore Container oder einen MediaPackage Channel
- Verwenden Sie einen Application Load Balancer
- Verwenden Sie einen Network Load Balancer
- Verwenden Sie eine Lambda-Funktions-URL
- Verwenden Sie Amazon EC2 (oder einen anderen benutzerdefinierten Ursprung)
- Verwenden Sie CloudFront Ursprungsgruppen
- Verwenden Sie Amazon API Gateway

Verwenden Sie einen Amazon S3 S3-Bucket

In den folgenden Themen werden die verschiedenen Möglichkeiten beschrieben, wie Sie einen Amazon S3 S3-Bucket als Ursprung für eine CloudFront Distribution verwenden können.

Themen

- Verwenden Sie einen standardmäßigen Amazon S3 S3-Bucket
- Verwenden Sie Amazon S3 Object Lambda
- Amazon S3 Access Point verwenden
- Verwenden Sie einen Amazon S3 S3-Bucket, der als Website-Endpunkt konfiguriert ist
- CloudFront Zu einem vorhandenen Amazon S3 S3-Bucket hinzufügen
- Verschieben Sie einen Amazon S3 S3-Bucket in einen anderen AWS-Region

Verwenden Sie einen standardmäßigen Amazon S3 S3-Bucket

Wenn Sie Amazon S3 als Quelle für Ihre Distribution verwenden, platzieren Sie die Objekte, die Sie liefern CloudFront möchten, in einem Amazon S3 S3-Bucket. Sie können alle von Amazon S3 unterstützten Methoden zum Bereitstellen Ihrer Objekte in Amazon S3 verwenden. Sie können beispielsweise die Amazon-S3-Konsole oder -API oder ein Drittanbieter-Tool verwenden. Sie können

in Ihrem Bucket wie für jeden anderen standardmäßigen Amazon-S3-Bucket eine Hierarchie zum Speichern der Objekte erstellen.

Wenn Sie einen vorhandenen Amazon S3 S3-Bucket als Ihren CloudFront Ursprungsserver verwenden, ändert sich der Bucket in keiner Weise. Sie können ihn weiterhin wie gewohnt verwenden, um Amazon S3-Objekte zum Amazon S3-Standardpreis zu speichern und darauf zuzugreifen. Es fallen die regulären Amazon S3-Gebühren für die Speicherung der Objekte im Bucket an. Weitere Informationen zu den zu CloudFront verwendenden Gebühren finden Sie unter CloudFrontAmazon-Preise. Weitere Informationen zur Verwendung CloudFront mit einem vorhandenen S3-Bucket finden Sie unterthe section called “ CloudFront Zu einem vorhandenen Amazon S3 S3-Bucket hinzufügen”.

> **Important**
>
> Damit Ihr Bucket verwendet werden kann CloudFront, muss der Name den DNS-Benennungsanforderungen entsprechen. Weitere Informationen finden Sie unter Benennungsregeln für Buckets im Benutzerhandbuch zu Amazon Simple Storage Service.

Wenn Sie einen Amazon S3 S3-Bucket als Ursprung für angeben CloudFront, empfehlen wir Ihnen, das folgende Format zu verwenden:

```
bucket-name.s3.region.amazonaws.com
```

Wenn Sie den Bucket-Namen in diesem Format angeben, können Sie die folgenden CloudFront -Funktionen verwenden:

- Konfigurieren Sie CloudFront die Kommunikation mit Ihrem Amazon S3 S3-Bucket über SSL/TLS. Weitere Informationen finden Sie unter the section called “Verwenden Sie HTTPS mit CloudFront”.
- Verwenden Sie eine Origin-Zugriffskontrolle, um zu verlangen, dass Zuschauer über Amazon S3 auf Ihre Inhalte zugreifen CloudFrontURLs, nicht über Amazon S3 URLs. Weitere Informationen finden Sie unter the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”.
- Aktualisieren Sie den Inhalt Ihres Buckets, indem Sie PUT Anfragen an senden POST CloudFront. Weitere Informationen finden Sie unter the section called “HTTP-Methoden” im Thema the section called “Wie CloudFront verarbeitet und leitet Anfragen an Ihren Amazon S3 S3-Ursprung weiter”.

Verwenden Sie für die Angabe des Buckets nicht die folgenden Formate:

- Den Amazon-S3-Pfadstil: `s3.amazonaws.com/bucket-name`
- Den Amazon-S3-CNAME

> **Note**
>
> CloudFront unterstützt S3-Ursprünge unter Verwendung beliebiger Speicherklassen, einschließlich S3 Intelligent-Tiering. Wenn CloudFront Objekte von einem S3-Ursprung angefordert werden, werden die Objekte unabhängig von der Speicherebene abgerufen, in der sie sich derzeit befinden. Die Verwendung CloudFront mit S3 Intelligent-Tiering hat keine Auswirkungen auf die Leistung oder Funktionalität Ihrer Distribution. Weitere Informationen finden Sie unter Verwaltung der Speicherkosten mit Amazon S3 Intelligent-Tiering im Amazon Simple Storage Service-Benutzerhandbuch.

Verwenden Sie Amazon S3 Object Lambda

Wenn Sie einen Object-Lambda-Zugriffspunkt erstellen, generiert Amazon S3 automatisch ein eindeutiges Alias für Ihren Object-Lambda-Zugriffspunkt. Sie können diesen Alias anstelle eines Amazon S3 S3-Bucket-Namens als Ursprung für Ihre CloudFront Distribution verwenden.

Wenn Sie einen Object Lambda Access Point-Alias als Ursprung für verwenden CloudFront, empfehlen wir Ihnen, das folgende Format zu verwenden:

`alias.s3.region.amazonaws.com`

Weitere Informationen zum Finden des *alias* finden Sie unter Verwenden eines Alias im Bucket-Stil für Ihren Object-Lambda-Zugriffspunkt für S3-Buckets im Amazon-S3-Benutzerhandbuch.

> **Important**
>
> Wenn Sie einen Object Lambda Access Point als Ursprung für verwenden CloudFront, müssen Sie die Origin-Zugriffskontrolle verwenden.

Ein Beispiel für einen Anwendungsfall finden Sie unter Verwenden von Amazon S3 Object Lambda mit Amazon CloudFront zur Anpassung von Inhalten für Endbenutzer.

CloudFront behandelt einen Object Lambda Access Point-Ursprung genauso wie einen standardmäßigen Amazon S3 S3-Bucket-Ursprung.

Wenn Sie Amazon S3 Object Lambda als Quelle für Ihre Distribution verwenden, müssen Sie die folgenden vier Berechtigungen konfigurieren.

Object Lambda Access Point

So fügen Sie Berechtigungen für den Object Lambda Access Point hinzu

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Object-Lambda-Zugriffspunkte aus.
3. Wählen Sie den Object-Lambda-Zugriffspunkt aus, den Sie verwenden möchten.
4. Wählen Sie die Registerkarte Berechtigungen.
5. Wählen Sie im Abschnitt Object-Lambda-Zugriffspunktrichtlinie die Option Bearbeiten aus.
6. Fügen Sie die folgende Richtlinie in das Feld Richtlinie ein.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudfront.amazonaws.com"
            },
            "Action": "s3-object-lambda:Get*",
            "Resource": "arn:aws:s3-object-lambda:region:AWS-account-
ID:accesspoint/Object-Lambda-Access-Point-name",
            "Condition": {
                "StringEquals": {
                    "aws:SourceArn": "arn:aws:cloudfront::AWS-account-
ID:distribution/CloudFront-distribution-ID"
                }
            }
        }
    ]
}
```

7. Wählen Sie Änderungen speichern aus.

Amazon S3 Access Point

So fügen Sie Berechtigungen für den Amazon S3 Access Point hinzu

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Zugriffspunkte aus.
3. Wählen Sie den Amazon S3 Access Point aus, den Sie verwenden möchten.
4. Wählen Sie die Registerkarte Berechtigungen.
5. Wählen Sie im Abschnitt Zugriffspunktrichtlinie die Option Bearbeiten aus.
6. Fügen Sie die folgende Richtlinie in das Feld Richtlinie ein.

   JSON

```
{
    "Version": "2012-10-17",
    "Id": "default",
    "Statement": [
        {
            "Sid": "s3objlambda",
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudfront.amazonaws.com"
            },
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:region:AWS-account-ID:accesspoint/Access-Point-name",
                "arn:aws:s3:region:AWS-account-ID:accesspoint/Access-Point-name/object/*"
            ],
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "s3-object-lambda.amazonaws.com"
                }
            }
        }
    ]
}
```

7. Wählen Sie Speichern.

Amazon S3 bucket

So fügen Sie dem Amazon S3 S3-Bucket Berechtigungen hinzu

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich die Option Buckets aus.
3. Wählen Sie den Amazon-S3-Bucket aus, den Sie verwenden möchten.
4. Wählen Sie die Registerkarte Berechtigungen.
5. Wählen Sie im Abschnitt Bucket-Richtlinie die Option Bearbeiten aus.
6. Fügen Sie die folgende Richtlinie in das Feld Richtlinie ein.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": "*",
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ],
            "Condition": {
                "StringEquals": {
                    "s3:DataAccessPointAccount": "AWS-account-ID"
                }
            }
        }
    ]
}
```

7. Wählen Sie Änderungen speichern aus.

AWS Lambda function

So fügen Sie der Lambda-Funktion Berechtigungen hinzu

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Lambda Konsole unter https://console.aws.amazon.com/lambda/.
2. Wählen Sie im Navigationsbereich Funktionen aus.
3. Wählen Sie die AWS Lambda Funktion aus, die Sie verwenden möchten.
4. Wählen Sie die Registerkarte Konfiguration und dann Berechtigungen aus.
5. Wählen Sie im Abschnitt Ressourcenbasierte Richtlinienanweisungen die Option Berechtigungen hinzufügen aus.
6. Wählen Sie AWS-Konto.
7. Geben Sie einen Namen für die Anweisungs-ID ein.
8. Geben Sie `cloudfront.amazonaws.com` als Prinzipal ein.
9. Wählen Sie `lambda:InvokeFunction` im Drop-down-Menü Aktion aus:
10. Wählen Sie Speichern.

Amazon S3 Access Point verwenden

Wenn Sie einen S3 Access Point verwenden, generiert Amazon S3 automatisch einen eindeutigen Alias für Sie. Sie können diesen Alias anstelle eines Amazon S3 S3-Bucket-Namens als Ursprung für Ihre CloudFront Distribution verwenden.

Wenn Sie einen Amazon S3 Access Point-Alias als Ursprung für verwenden CloudFront, empfehlen wir Ihnen, das folgende Format zu verwenden:

`alias``.s3.`*`region`*`.amazonaws.com`

Weitere Informationen zum Auffinden *`alias`* von finden Sie unter Verwenden eines Alias im Bucket-Stil für Ihren S3-Bucket-Zugriffspunkt im Amazon S3 S3-Benutzerhandbuch.

> **Important**
>
> Wenn Sie einen Amazon S3 Access Point als Ausgangspunkt für verwenden CloudFront, müssen Sie die Origin-Zugriffskontrolle verwenden.

CloudFront behandelt einen Amazon S3 Access Point-Ursprung genauso wie einen standardmäßigen Amazon S3 S3-Bucket-Ursprung.

Wenn Sie Amazon S3 Object Lambda als Quelle für Ihre Distribution verwenden, müssen Sie die folgenden beiden Berechtigungen konfigurieren.

Amazon S3 Access Point

So fügen Sie Berechtigungen für den Amazon S3 Access Point hinzu

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Zugriffspunkte aus.
3. Wählen Sie den Amazon S3 Access Point aus, den Sie verwenden möchten.
4. Wählen Sie die Registerkarte Berechtigungen.
5. Wählen Sie im Abschnitt Zugriffspunktrichtlinie die Option Bearbeiten aus.
6. Fügen Sie die folgende Richtlinie in das Feld Richtlinie ein.

   JSON

```
{
    "Version": "2012-10-17",
    "Id": "default",
    "Statement": [
        {
            "Sid": "s3objlambda",
            "Effect": "Allow",
            "Principal": {"Service": "cloudfront.amazonaws.com"},
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:region:AWS-account-ID:accesspoint/Access-
Point-name",
                "arn:aws:s3:region:AWS-account-ID:accesspoint/Access-
Point-name/object/*"
            ],
            "Condition": {
                "StringEquals": {"aws:SourceArn":
 "arn:aws:cloudfront::AWS-account-ID:distribution/CloudFront-distribution-
ID"}
            }
```

```
        }
    ]
}
```

7. Wählen Sie Speichern.

Amazon S3 bucket

So fügen Sie dem Amazon S3 S3-Bucket Berechtigungen hinzu

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich die Option Buckets aus.
3. Wählen Sie den Amazon-S3-Bucket aus, den Sie verwenden möchten.
4. Wählen Sie die Registerkarte Berechtigungen.
5. Wählen Sie im Abschnitt Bucket-Richtlinie die Option Bearbeiten aus.
6. Fügen Sie die folgende Richtlinie in das Feld Richtlinie ein.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": "*",
            "Resource": [
                "arn:aws:s3:::bucket-name",
                "arn:aws:s3:::bucket-name/*"
            ],
            "Condition": {
                "StringEquals": {
                    "s3:DataAccessPointAccount": "AWS-account-ID"
                }
            }
        }
    ]
```

```
}
```

7. Wählen Sie Änderungen speichern aus.

Verwenden Sie einen Amazon S3 S3-Bucket, der als Website-Endpunkt konfiguriert ist

Sie können einen Amazon S3 S3-Bucket, der als Website-Endpunkt konfiguriert ist, als benutzerdefinierten Ursprung mit verwendenCloudFront. Wenn Sie Ihre CloudFront Distribution konfigurieren, geben Sie für den Ursprung den statischen Amazon S3 S3-Website-Hosting-Endpunkt für Ihren Bucket ein. Dieser Wert wird in der Amazon-S3-Konsole auf der Registerkarte Properties (Eigenschaften) im Bereich Static Website Hosting (Hosten statischer Websites) angezeigt. Beispiel:

`http://`*`bucket-name`*`.s3-website-`*`region`*`.amazonaws.com`

Weitere Informationen zur Angabe von Amazon-S3-Endpunkten für statische Websites finden Sie unter Website-Endpunkte im Benutzerhandbuch zu Amazon Simple Storage Service.

Wenn Sie den Bucket-Namen in diesem Format als Ursprung angeben, können Sie Amazon S3-Umleitungen und benutzerdefinierte Amazon S3-Fehlerdokumente verwenden. Weitere Informationen finden Sie unter Konfiguration eines benutzerdefinierten Fehlerdokuments und Konfiguration einer Umleitung im Amazon Simple Storage Service-Benutzerhandbuch. (bietet CloudFront auch benutzerdefinierte Fehlerseiten. Weitere Informationen finden Sie unterthe section called “Erstellen Sie eine benutzerdefinierte Fehlerseite für bestimmte HTTP-Statuscodes”.)

Wenn Sie einen Amazon S3 S3-Bucket als Ihren CloudFront Ursprungsserver verwenden, ändert sich der Bucket in keiner Weise. Sie können ihn weiter wie normal zu den regulären Amazon S3-Gebühren verwenden. Weitere Informationen zu den zu CloudFront verwendenden Gebühren finden Sie unter CloudFront Amazon-Preise.

> **Note**
>
> Wenn Sie die CloudFront API verwenden, um Ihre Distribution mit einem Amazon S3 S3-Bucket zu erstellen, der als Website-Endpunkt konfiguriert ist, müssen Sie ihn mithilfe von konfigurieren`CustomOriginConfig`, obwohl die Website in einem Amazon S3 S3-Bucket gehostet wird. Weitere Informationen zum Erstellen von Verteilungen mithilfe der CloudFront API finden Sie CreateDistributionin der Amazon CloudFront API-Referenz.

CloudFront Zu einem vorhandenen Amazon S3 S3-Bucket hinzufügen

Wenn Sie Ihre Objekte in einem Amazon S3 S3-Bucket speichern, können Sie entweder festlegen, dass Benutzer Ihre Objekte direkt von S3 abrufen, oder Sie können so konfigurieren, CloudFront dass Ihre Objekte von S3 abgerufen und dann an Ihre Benutzer verteilt werden. Die Verwendung CloudFront kann kostengünstiger sein, wenn Ihre Benutzer häufig auf Ihre Objekte zugreifen, da bei höherer Nutzung der Preis für die CloudFront Datenübertragung niedriger ist als der Preis für die Amazon S3 S3-Datenübertragung. Darüber hinaus sind Downloads mit Amazon S3 allein schneller CloudFront als mit Amazon S3 allein, da Ihre Objekte näher an Ihren Benutzern gespeichert werden.

> **Note**
>
> Wenn Sie CloudFront die ursprungsübergreifenden Amazon S3-Einstellungen für die gemeinsame Nutzung von Ressourcen respektieren möchten, konfigurieren Sie die Konfiguration so, CloudFront dass der `Origin` Header an Amazon S3 weitergeleitet wird. Weitere Informationen finden Sie unter the section called “Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern”.

Wenn Sie derzeit Inhalte direkt aus Ihrem Amazon S3 S3-Bucket unter Verwendung Ihres eigenen Domainnamens (z. B. example.com) anstelle des Domainnamens Ihres Amazon S3-Buckets (z. B. amzn-s3-demo-bucket. s3.us-west-2.amazonaws.com) verteilen, können Sie ohne Unterbrechung hinzufügen CloudFront, indem Sie das folgende Verfahren verwenden.

Um hinzuzufügen CloudFront , wenn Sie Ihre Inhalte bereits über Amazon S3 verteilen

1. Erstellen Sie eine CloudFront Distribution. Weitere Informationen finden Sie unter the section called “Eine Verteilung erstellen”.

   Wenn Sie die Verteilung erstellen, geben Sie den Namen Ihres Amazon S3-Buckets als Ursprungs-Server an.

   > **Important**
   >
   > Damit Ihr Bucket verwendet werden kann CloudFront, muss der Name den DNS-Benennungsanforderungen entsprechen. Weitere Informationen finden Sie unter Benennungsregeln für Buckets im Benutzerhandbuch zu Amazon Simple Storage Service.

Wenn Sie einen CNAME mit Amazon S3 verwenden, geben Sie den CNAME für Ihre Verteilung ebenfalls an.

2. Erstellen Sie eine Testwebseite mit Links, die auf öffentlich verfügbare Objekte in Ihrem Amazon S3-Bucket verweisen, und testen Sie diese Links. Verwenden Sie für diesen ersten Test den CloudFront Domainnamen Ihrer Distribution im Objekt URLs, z. B. `https://d111111abcdef8.cloudfront.net/images/image.jpg`

   Weitere Hinweise zum Format von CloudFront URLs finden Sie unterthe section called “Datei anpassen URLs”.

3. Wenn Sie Amazon S3 verwenden CNAMEs, verwendet Ihre Anwendung Ihren Domainnamen (z. B. example.com), um auf die Objekte in Ihrem Amazon S3 S3-Bucket zu verweisen, anstatt den Namen Ihres Buckets zu verwenden (z. B. amzn-s3-demo-bucket.s3.amazonaws.com). Wenn Sie Ihren Domainnamen weiterhin verwenden möchten, um auf Objekte zu verweisen, anstatt den CloudFront Domainnamen für Ihre Distribution zu verwenden (z. B. d111111abcdef8.cloudfront.net), müssen Sie Ihre Einstellungen bei Ihrem DNS-Dienstanbieter aktualisieren.

   Damit Amazon S3 CNAMEs funktioniert, muss Ihr DNS-Dienstanbieter einen CNAME-Ressourceneintrag für Ihre Domain eingerichtet haben, der derzeit Abfragen für die Domain an Ihren Amazon S3-Bucket weiterleitet. Wenn ein Benutzer z. B. dieses Objekt anfordert:

   `https://example.com/images/image.jpg`

   Die Anfrage wird automatisch umgeleitet und dem Benutzer wird dieses Objekt angezeigt:

   `https://amzn-s3-demo-bucket.s3.amazonaws.com/images/image.jpg`

   Um Anfragen an Ihre CloudFront Distribution und nicht an Ihren Amazon S3 S3-Bucket weiterzuleiten, müssen Sie die von Ihrem DNS-Dienstanbieter bereitgestellte Methode verwenden, um den CNAME-Ressourceneintrag für Ihre Domain zu aktualisieren. Dieser aktualisierte CNAME-Eintrag leitet DNS-Anfragen von Ihrer Domain an den CloudFront Domainnamen für Ihre Distribution weiter. Weitere Informationen finden Sie in der Dokumentation Ihres DNS-Serviceanbieters.

> **Note**
>
> Wenn Sie Route 53 als DNS-Service verwenden, können Sie entweder einen CNAME-Ressourcendatensatz oder einen Alias-Ressourcendatensatz verwenden. Informationen zum Bearbeiten von Ressourceneintragssätzen finden Sie unter Bearbeiten von Datensätzen. Informationen zu Alias-Ressourceneintragssätzen finden Sie unter Wählen zwischen Alias- und Nicht-Alias-Datensätzen. Beide Themen finden Sie im Amazon Route 53-Entwicklerhandbuch.

Weitere Hinweise zur Verwendung von CNAMEs with finden Sie CloudFront unterthe section called “Benutzerdefiniert verwenden URLs”.

Wenn Sie den CNAME-Ressourcendatensatz aktualisiert haben, kann es bis zu 72 Stunden dauern, bis die Änderungen im gesamten DNS-System übernommen werden, auch wenn das in der Regel schneller geschieht. Während dieser Zeit werden einige Anfragen für Ihre Inhalte weiterhin an Ihren Amazon S3 S3-Bucket und andere an Ihren Amazon S3-Bucket weitergeleitet. CloudFront

Verschieben Sie einen Amazon S3 S3-Bucket in einen anderen AWS-Region

Wenn Sie Amazon S3 als Ursprung für eine CloudFront Distribution verwenden und den Bucket in einen anderen verschieben AWS-Region, CloudFront kann es bis zu einer Stunde dauern, bis die Datensätze aktualisiert sind, sodass die neue Region verwendet wird, wenn beide der folgenden Bedingungen zutreffen:

- Sie verwenden eine CloudFront Origin Access Identity (OAI), um den Zugriff auf den Bucket einzuschränken.
- Sie verschieben den Bucket in eine Amazon S3-Region, die Signature Version 4 für die Authentifizierung erfordert.

Wenn Sie verwenden OAIs, CloudFront verwendet die Region (neben anderen Werten), um die Signatur zu berechnen, anhand derer Objekte aus Ihrem Bucket angefordert werden. Weitere Informationen zu finden OAIs Sie unterthe section called “Verwenden Sie eine ursprüngliche Zugriffsidentität (veraltet, nicht empfohlen)”. Eine Liste der AWS-Regionen unterstützten Signature

Version 2 finden Sie unter Signaturprozess für Signature Version 2 in der Allgemeine Amazon Web Services-Referenz.

Um eine schnellere Aktualisierung CloudFront der Datensätze zu erzwingen, können Sie Ihre CloudFront Distribution aktualisieren, indem Sie beispielsweise das Feld Beschreibung auf der Registerkarte Allgemein in der CloudFront Konsole aktualisieren. Wenn Sie eine Distribution aktualisieren, wird CloudFront sofort die Region überprüft, in der sich Ihr Bucket befindet. Das Übertragen der Änderung auf alle Edge-Standorte sollte nur wenige Minuten in Anspruch nehmen.

Verwenden Sie einen MediaStore Container oder einen MediaPackage Channel

Um Videos zu streamen CloudFront, können Sie einen Amazon S3 S3-Bucket einrichten, der als MediaStore Container konfiguriert ist, oder einen Kanal und Endpunkte mit MediaPackage erstellen. Anschließend erstellen und konfigurieren Sie eine Distribution, um das Video CloudFront zu streamen.

Weitere Informationen und step-by-step Anweisungen finden Sie in den folgenden Themen:

- the section called “Stellen Sie das Video bereit, indem Sie AWS Elemental MediaStore es als Quelle verwenden”
- the section called “Bereitstellen Sie Live-Videos, formatiert mit AWS Elemental MediaPackage”

Verwenden Sie einen Application Load Balancer

Sie können ihn verwenden CloudFront , um Datenverkehr sowohl an interne als auch an mit dem Internet verbundene Application Load Balancer weiterzuleiten.

Wenn Ihr Ursprung ein oder mehrere HTTP (S) -Server (Webserver) sind, die auf einer oder mehreren EC2 Amazon-Instances gehostet werden, können Sie wählen, ob Sie einen mit dem Internet verbundenen Application Load Balancer verwenden möchten, um den Traffic auf die Instances zu verteilen. Ein mit dem Internet verbundener Load Balancer hat einen öffentlich auflösbaren DNS-Namen und leitet Anfragen von Clients über das Internet an Ziele weiter.

Weitere Informationen zur Verwendung eines mit dem Internet verbundenen Application Load Balancer als Ausgangspunkt CloudFront, einschließlich der Frage, wie Sie sicherstellen können, dass Zuschauer nur über den Load Balancer auf Ihre Webserver zugreifen können CloudFront und nicht direkt auf den Load Balancer zugreifen können, finden Sie unter. the section called “Beschränken Sie den Zugriff auf Application Load Balancers”

Alternativ können Sie VPC-Ursprünge verwenden, um Inhalte aus Anwendungen bereitzustellen, die mit einem internen Application Load Balancer in Ihren privaten VPC-Subnetzen (Virtual Private Cloud) gehostet werden. VPC-Ursprünge verhindern, dass Ihre Anwendung im öffentlichen Internet zugänglich ist. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff mit VPC-Ursprüngen.

Verwenden Sie einen Network Load Balancer

Sie können sowohl interne als auch mit dem Internet verbundene Network Load Balancer mit Amazon verwenden. CloudFront Sie können interne Network Load Balancer in privaten Subnetzen verwenden, CloudFront indem Sie VPC-Ursprünge verwenden. CloudFront VPC-Ursprünge ermöglichen es Ihnen, Inhalte von Anwendungen bereitzustellen, die in privaten VPC-Subnetzen gehostet werden, ohne sie dem öffentlichen Internet zugänglich zu machen. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff mit VPC-Ursprüngen.

Alternativ können Sie es auch CloudFront für die Übertragung von Datenverkehr von mit dem Internet verbundenen Network Load Balancers verwenden. Ein mit dem Internet verbundener Load Balancer hat einen öffentlich auflösbaren DNS-Namen und kann Anfragen sowohl von Clients im Internet als auch von Distributionen empfangen. CloudFront

Verwenden Sie eine Lambda-Funktions-URL

Eine Lambda-Funktions-URL ist ein dedizierter HTTPS-Endpunkt für eine Lambda-Funktion. Sie können eine Lambda-Funktions-URL verwenden, um eine serverlose Webanwendung vollständig in Lambda zu erstellen. Sie können die Lambda-Webanwendung direkt über die Funktions-URL aufrufen, ohne dass eine Integration in API Gateway oder einen Application Load Balancer erforderlich ist.

Wenn Sie eine serverlose Webanwendung mithilfe von Lambda-Funktionen mit Funktion erstellen, können Sie sie hinzufügenURLs, um die folgenden Vorteile CloudFront zu erhalten:

- Beschleunigen Ihrer Anwendung, indem Sie Inhalte näher an den Viewern zwischenspeichern
- Verwenden eines benutzerdefinierten Domänennamens für Ihre Webanwendung
- Verschiedene URL-Pfade mithilfe von CloudFront Cache-Verhalten an verschiedene Lambda-Funktionen weiterleiten
- Blockieren Sie bestimmte Anfragen mithilfe CloudFront geografischer Einschränkungen oder AWS WAF (oder beidem)

- Verwenden Sie AWS WAF with CloudFront , um Ihre Anwendung vor bösartigen Bots zu schützen, häufige Anwendungs-Exploits zu verhindern und den Schutz vor DDo S-Angriffen zu verbessern

Um eine Lambda-Funktions-URL als Ursprung für eine CloudFront Distribution zu verwenden, geben Sie den vollständigen Domainnamen der Lambda-Funktions-URL als Ursprungsdomäne an. Ein Domänenname der Lambda-Funktions-URL weist das folgende Format auf:

function-URL-ID.lambda-url.*AWS-Region*.on.aws

Wenn Sie eine Lambda-Funktions-URL als Ursprung für eine CloudFront Distribution verwenden, muss die Funktions-URL öffentlich zugänglich sein. Verwenden Sie dazu eine der folgenden Optionen:

- Wenn Sie Origin Access Control (OAC) verwenden, muss der `AuthType` Parameter der Lambda-Funktions-URL den `AWS_IAM` Wert verwenden und die `lambda:InvokeFunctionUrl` Berechtigung in einer ressourcenbasierten Richtlinie zulassen. Weitere Hinweise zur Verwendung der Lambda-Funktion URLs für OAC finden Sie unter. Beschränken Sie den Zugriff auf den URL-Ursprung einer AWS Lambda Funktion
- Wenn Sie OAC nicht verwenden, können Sie den `AuthType` Parameter der Funktions-URL auf festlegen `NONE` und die `lambda:InvokeFunctionUrl` Erlaubnis in einer ressourcenbasierten Richtlinie gewähren.

Sie können den Anfragen, die an den Ursprung CloudFront gesendet werden, auch einen benutzerdefinierten Origin-Header hinzufügen und Funktionscode schreiben, um eine Fehlerantwort zurückzugeben, wenn der Header nicht in der Anfrage enthalten ist. Dadurch wird sichergestellt, dass Benutzer nur über die URL der Lambda-Funktion und nicht direkt über die CloudFront URL der Lambda-Funktion auf Ihre Webanwendung zugreifen können.

Weitere Informationen zur Lambda-Funktion URLs finden Sie in den folgenden Themen im AWS Lambda Entwicklerhandbuch:

- Lambda-Funktion URLs — Ein allgemeiner Überblick über die Lambda-Funktionsfunktion URLs
- Lambda-Funktion aufrufen URLs — Enthält Details zu den Anforderungs- und Antwort-Payloads, die für die Codierung Ihrer serverlosen Webanwendung verwendet werden sollen
- Sicherheits- und Authentifizierungsmodell für die Lambda-Funktion URLs — Enthält Details zu den Lambda-Authentifizierungstypen

Verwenden Sie Amazon EC2 (oder einen anderen benutzerdefinierten Ursprung)

Sie können sowohl interne als auch mit dem Internet verbundene EC2 Instances mit Amazon verwenden. CloudFront Sie können interne EC2 Instances in privaten Subnetzen verwenden, CloudFront indem Sie VPC-Ursprünge verwenden. CloudFront VPC-Ursprünge ermöglichen es Ihnen, Inhalte von Anwendungen bereitzustellen, die in privaten VPC-Subnetzen gehostet werden, ohne sie dem öffentlichen Internet zugänglich zu machen. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff mit VPC-Ursprüngen.

Ein benutzerdefinierter Ursprung ist ein HTTP (S) -Webserver mit einem öffentlich auflösbaren DNS-Namen, der Anfragen von Clients über das Internet an Ziele weiterleitet. Der HTTP (S) -Server kann auf AWS— z. B. einer EC2 Amazon-Instance — oder an einem anderen Ort gehostet werden. Ein als Website-Endpunkt konfigurierter Amazon S3-Ursprung gilt ebenfalls als benutzerdefinierter Ursprung. Weitere Informationen finden Sie unter the section called “Verwenden Sie einen Amazon S3 S3-Bucket, der als Website-Endpunkt konfiguriert ist”.

Wenn Sie Ihren eigenen HTTP-Server als benutzerdefinierten Ursprung verwenden, geben Sie den DNS-Namen des Servers zusammen mit den HTTP- und HTTPS-Ports und dem Protokoll an, das Sie beim Abrufen von Objekten von Ihrem Ursprung verwenden CloudFront möchten.

Die meisten CloudFront Funktionen werden unterstützt, wenn Sie einen benutzerdefinierten Ursprung verwenden, mit Ausnahme von privaten Inhalten. Sie können zwar eine signierte URL verwenden, um Inhalte von einem benutzerdefinierten Ursprung zu verteilen, aber CloudFront um auf den benutzerdefinierten Ursprung zugreifen zu können, muss der Ursprung öffentlich zugänglich bleiben. Weitere Informationen finden Sie unter the section called “Beschränken Sie Inhalte mit signierten URLs und signierten Cookies”.

Folgen Sie diesen Richtlinien für die Verwendung von EC2 Amazon-Instances und anderen benutzerdefinierten Ursprüngen mitCloudFront.

- Hosten Sie denselben Inhalt auf allen Servern bzw. stellen Sie ihn auf diesen bereit, die Inhalt für den gleichen CloudFront-Ursprung bereitstellen. Weitere Informationen finden Sie unter the section called “Ursprungseinstellungen” im Thema the section called “Alle Verteilungseinstellungen”.
- Protokollieren Sie die `X-Amz-Cf-Id` Header-Einträge auf allen Servern, falls Sie diesen Wert für CloudFront das Debuggen benötigen Support oder verwenden möchten.
- Beschränken Sie Anforderungen von HTTP- und HTTPS-Ports, die Ihr benutzerdefinierter Ursprung überwacht.

- Synchronisieren Sie die Uhrzeit von allen Servern in der Implementierung. Beachten Sie, dass für signierte URLs und signierte Cookies, für Protokolle und Berichte die koordinierte Weltzeit (Coordinated Universal Time, UTC) CloudFront verwendet wird. Beachten Sie außerdem, dass bei der Überwachung von CloudFront Aktivitäten mithilfe von CloudWatch Messwerten CloudWatch auch UTC verwendet wird.
- Verwenden Sie redundante Server für die Behandlung von Ausfällen.
- Informationen zur Verwendung eines benutzerdefinierten Ursprungs für die Bereitstellung privater Inhalte finden Sie unter the section called “Beschränken Sie den Zugriff auf Dateien mit benutzerdefinierten Ursprüngen”.
- Informationen zu Anfrage- und Antwortverhalten sowie zu unterstützten HTTP-Statuscodes finden Sie unter Verhalten von Anfragen und Antworten.

Wenn Sie Amazon EC2 für eine benutzerdefinierte Herkunft verwenden, empfehlen wir Ihnen, wie folgt vorzugehen:

- Verwenden Sie ein Amazon Machine Image, das die Software für einen Webserver automatisch installiert. Weitere Informationen finden Sie in der EC2 Amazon-Dokumentation.
- Verwenden Sie einen Elastic Load Balancing Load Balancer, um den Datenverkehr zwischen mehreren EC2 Amazon-Instances zu verarbeiten und Ihre Anwendung von Änderungen an EC2 Amazon-Instances zu isolieren. Wenn Sie beispielsweise einen Load Balancer verwenden, können Sie EC2 Amazon-Instances hinzufügen und löschen, ohne Ihre Anwendung zu ändern. Weitere Informationen finden Sie im Elastic Load Balancing-Benutzerhandbuch.
- Wenn Sie Ihre CloudFront Distribution erstellen, geben Sie die URL des Load Balancers für den Domainnamen Ihres Ursprungsservers an. Weitere Informationen finden Sie unter the section called “Eine Verteilung erstellen”.

Verwenden Sie CloudFront Ursprungsgruppen

Sie können eine Ursprungsgruppe für Ihren CloudFront Ursprung angeben, wenn Sie beispielsweise das Origin-Failover für Szenarien konfigurieren möchten, in denen Sie hohe Verfügbarkeit benötigen. Verwenden Sie das Origin-Failover, um einen primären Ursprung CloudFront sowie einen zweiten Ursprung festzulegen, der CloudFront automatisch zu diesem wechselt, wenn der primäre Ursprung bestimmte HTTP-Statuscode-Fehlerantworten zurückgibt.

Weitere Informationen, einschließlich der Schritte für die Einrichtung einer Ursprungsgruppe, finden Sie unter the section called “Erhöhen Sie die Verfügbarkeit mit Origin Failover”.

Verwenden Sie Amazon API Gateway

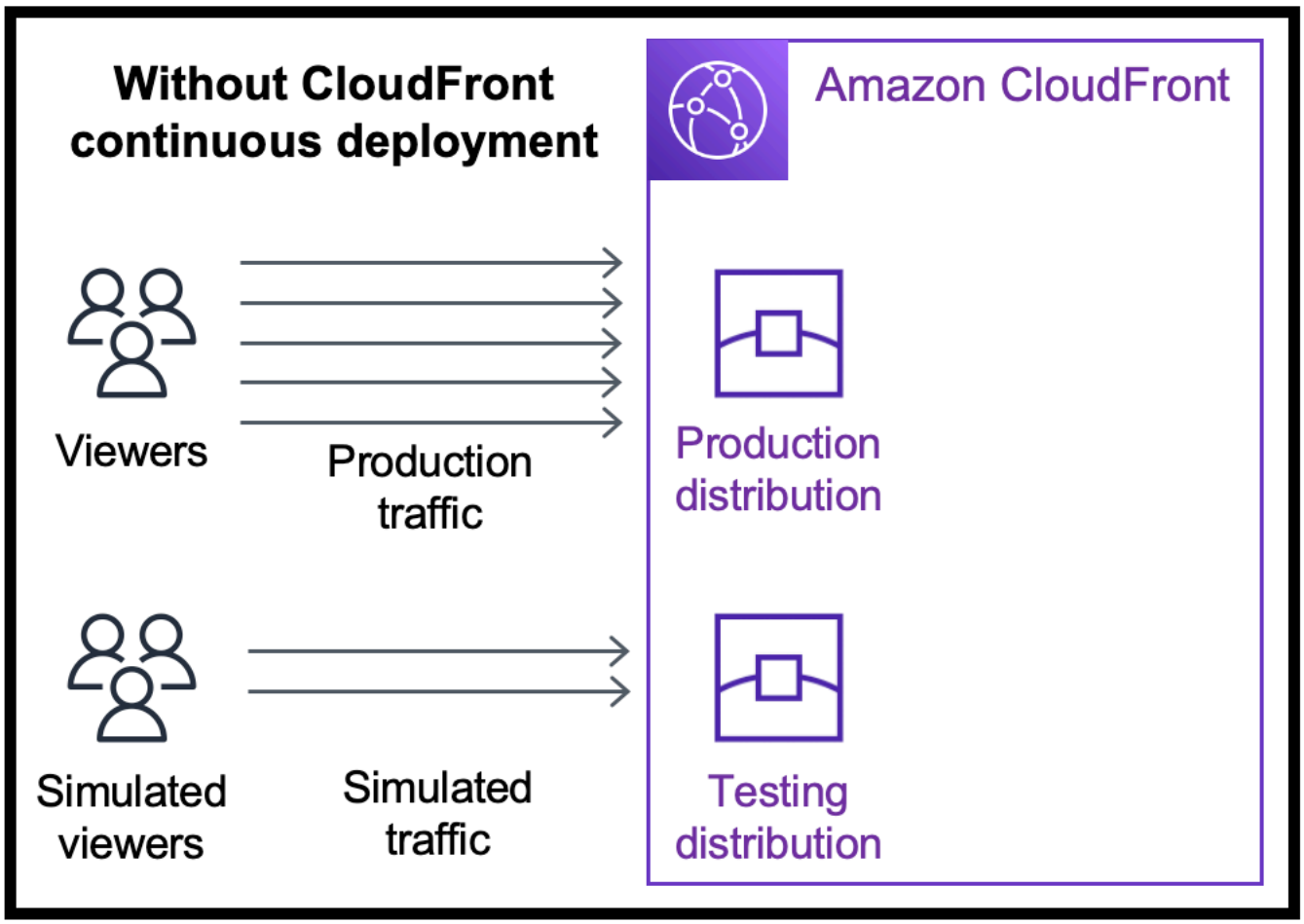
Sie können API Gateway als benutzerdefinierten Ursprung für Ihre CloudFront Distribution verwenden. Weitere Informationen finden Sie unter den folgenden Themen:

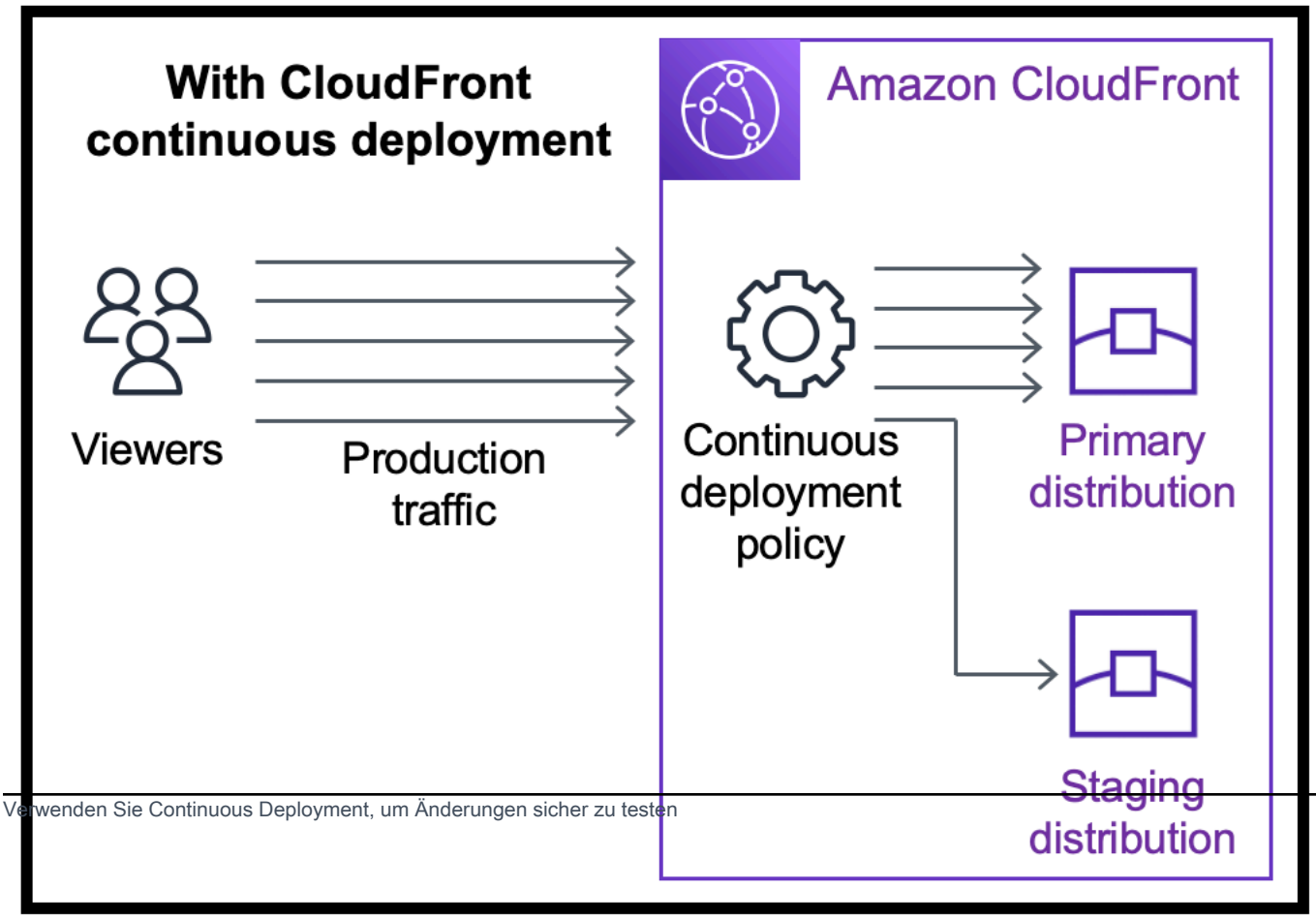
- Sicherung von Amazon API Gateway mit sicheren Chiffren mithilfe eines Amazon-Blogbeitrags CloudFront AWS
- Wie richte ich API Gateway mit meiner eigenen CloudFront Distribution ein? AWS re:Post

Verwenden Sie CloudFront Continuous Deployment, um CDN-Konfigurationsänderungen sicher zu testen

Mit Amazon CloudFront Continuous Deployment können Sie Änderungen an Ihrer CDN-Konfiguration sicher implementieren, indem Sie zunächst mit einer Teilmenge des Produktionsdatenverkehrs testen. Sie können eine Staging-Verteilung und eine Richtlinie für die kontinuierliche Bereitstellung verwenden, um einen Teil des Datenverkehrs von tatsächlichen Viewern (in der Produktion) an die neue CDN-Konfiguration zu senden und zu überprüfen, ob diese wie erwartet funktioniert. Sie können die Leistung der neuen Konfiguration in Echtzeit überwachen und, wenn Sie bereit sind, die neue Konfiguration so hochstufen, dass sie für den gesamten Datenverkehr über die primäre Verteilung gilt.

Das folgende Diagramm zeigt die Vorteile der CloudFront kontinuierlichen Bereitstellung. Ohne diese Funktionalität müssten Sie CDN-Konfigurationsänderungen mit simuliertem Datenverkehr testen. Bei Verwendung der kontinuierlichen Bereitstellung können Sie die Änderungen an einem Teil des Produktionsdatenverkehrs testen und dann auf die primäre Verteilung hochstufen, wenn Sie dazu bereit sind.


Without CloudFront continuous deployment
Amazon CloudFront
Viewers
Production traffic
Production distribution
Simulated viewers
Simulated traffic
Testing distribution


With CloudFront continuous deployment
Amazon CloudFront
Viewers
Production traffic
Continuous deployment policy
Primary distribution
Staging distribution

In den folgenden Themen erfahren Sie mehr über die Arbeit mit kontinuierlicher Bereitstellung.

Themen

CloudFront Arbeitsablauf für die kontinuierliche Bereitstellung

Der folgende allgemeine Workflow erklärt, wie Sie Konfigurationsänderungen bei CloudFront kontinuierlicher Bereitstellung sicher testen und bereitstellen können.

1. Wählen Sie die Verteilung aus, die Sie als primäre Verteilung verwenden möchten. Die primäre Verteilung ist die Verteilung, die zurzeit für den Produktionsdatenverkehr zuständig ist.
2. Erstellen Sie von der primären Verteilung ausgehend eine Staging-Verteilung. Eine Staging-Verteilung ist zunächst eine Kopie der primären Verteilung.
3. Erstellen Sie eine Datenverkehrskonfiguration in einer Richtlinie für die kontinuierliche Bereitstellung und fügen Sie diese der primären Verteilung an. Dadurch wird bestimmt, wie CloudFront der Verkehr an die Staging-Verteilung weitergeleitet wird. Weitere Informationen zum Weiterleiten von Anforderungen an eine Staging-Verteilung finden Sie unter the section called “Anfragen an die Staging-Distribution weiterleiten”.
4. Aktualisieren Sie die Konfiguration der Staging-Verteilung. Weitere Informationen zu den aktualisierbaren Einstellungen finden Sie unter the section called “Aktualisieren Sie die Primär- und Staging-Distributionen”.
5. Überwachen Sie die Staging-Verteilung, um festzustellen, ob die Konfigurationsänderungen erwartungsgemäß funktionieren. Weitere Informationen zum Überwachen einer Staging-Verteilung finden Sie unter the section called “Überwachen Sie eine Staging-Verteilung”.

   Während der Überwachung der Staging-Verteilung können Sie folgende Aktionen ausführen:

   - Erneutes Aktualisieren der Konfiguration der Staging-Verteilung, um die Konfigurationsänderungen weiter zu testen
   - Aktualisieren der Richtlinie für die kontinuierliche Bereitstellung (Datenverkehrskonfiguration), um mehr oder weniger Datenverkehr an die Staging-Verteilung zu senden

6. Wenn Sie mit der Leistung der Staging-Verteilung zufrieden sind, können Sie die Konfiguration der Staging-Verteilung auf die primäre Verteilung hochstufen. Dabei wird die Konfiguration der Staging-Verteilung in die primäre Verteilung kopiert. Dadurch wird auch die Continuous Deployment Policy deaktiviert, was bedeutet, dass der gesamte CloudFront Datenverkehr an die primäre Verteilung weitergeleitet wird.

Sie können eine Automatisierung erstellen, die die Leistung der Staging-Verteilung überwacht (Schritt 5) und die Konfiguration automatisch hochstuft (Schritt 6), wenn bestimmte Kriterien erfüllt sind.

Nachdem Sie eine Konfiguration hochgestuft haben, können Sie die betreffende Staging-Verteilung erneut verwenden, wenn Sie das nächste Mal eine Konfigurationsänderung testen möchten.

Weitere Informationen zum Arbeiten mit Staging-Verteilungen und Richtlinien für die kontinuierliche Bereitstellung in der CloudFront Konsole AWS CLI, der oder der CloudFront API finden Sie im folgenden Abschnitt.

Arbeiten Sie mit einer Richtlinie für Staging-Verteilung und kontinuierliche Bereitstellung

Sie können Staging-Verteilungen und Richtlinien für die kontinuierliche Bereitstellung in der CloudFront Konsole, mit der AWS Command Line Interface (AWS CLI) oder mit der CloudFront API erstellen, aktualisieren und ändern.

Erstellen Sie eine Staging-Verteilung mit einer Richtlinie für die kontinuierliche Bereitstellung

Die folgenden Verfahren zeigen Ihnen, wie Sie eine Staging-Verteilung mit einer kontinuierlichen Bereitstellungsrichtlinie erstellen.

Console

Sie können eine Staging-Verteilung mit einer kontinuierlichen Bereitstellungsrichtlinie erstellen, indem Sie die verwenden. AWS Management Console

So erstellen Sie eine Staging-Verteilung und eine Richtlinie für die kontinuierliche Bereitstellung (Konsole)

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Rufen Sie im Navigationsbereich Distributions auf.
3. Wählen Sie die Verteilung aus, die Sie als primäre Verteilung verwenden möchten. Die primäre Verteilung ist die Verteilung, die zurzeit für den Produktionsdatenverkehr zuständig ist. Aus dieser Verteilung erstellen Sie die Staging-Verteilung.
4. Wählen Sie im Abschnitt Continuous deployment (Kontinuierliche Bereitstellung) die Option Create Staging Distribution (Staging-Verteilung erstellen) aus. Dadurch wird der Assistent Create Staging Distribution (Staging-Verteilung erstellen) geöffnet.
5. Führen Sie im Assistenten Create Staging Distribution (Create Staging Distribution) folgende Schritte aus:

   a. (Optional) Geben Sie eine Beschreibung für die Staging-Verteilung ein.

   b. Wählen Sie Weiter aus.

   c. Ändern Sie die Konfiguration der Staging-Verteilung. Weitere Informationen zu den aktualisierbaren Einstellungen finden Sie unter the section called “Aktualisieren Sie die Primär- und Staging-Distributionen”.

      Wenn Sie mit den Änderungen an der Konfiguration der Staging-Verteilung fertig sind, wählen Sie Next (Weiter) aus.

   d. Geben Sie über die Konsole die Traffic configuration (Datenverkehrskonfiguration) an. Dies bestimmt, wie der CloudFront Datenverkehr an die Staging-Verteilung weitergeleitet wird. (CloudFront Speichert die Verkehrskonfiguration in einer Richtlinie für die kontinuierliche Bereitstellung.)

      Weitere Informationen zu den Optionen für die Datenverkehrskonfiguration finden Sie unter the section called “Anfragen an die Staging-Distribution weiterleiten”.

      Wenn Sie die Traffic configuration (Datenverkehrskonfiguration) abgeschlossen haben, wählen Sie Next (Weiter) aus.

   e. Überprüfen Sie die Konfiguration der Staging-Verteilung, einschließlich der Datenverkehrskonfiguration, und wählen Sie dann Create Staging Distribution (Staging-Verteilung erstellen) aus.

Wenn Sie den Assistenten zum Erstellen einer Staging-Verteilung in der CloudFront Konsole abgeschlossen haben, CloudFront geht er wie folgt vor:

- Erstellen einer Staging-Verteilung mit den von Ihnen (in Schritt 5c) angegebenen Einstellungen
- Erstellen einer Richtlinie für die kontinuierliche Bereitstellung mit der von Ihnen (in Schritt 5d) angegebenen Datenverkehrskonfiguration
- Anfügen der Richtlinie für die kontinuierliche Bereitstellung an die primäre Verteilung, aus der Sie die Staging-Verteilung erstellt haben

Wenn die Konfiguration der Primärverteilung mit der beigefügten Continuous Deployment Policy an Edge-Standorten bereitgestellt wird, CloudFront beginnt das Senden des angegebenen Datenverkehrs auf der Grundlage der Verkehrskonfiguration an die Staging-Verteilung.

CLI

Gehen Sie wie folgt vor, um eine Staging-Verteilung und eine Richtlinie für die kontinuierliche Bereitstellung mit der AWS CLI zu erstellen.

So erstellen Sie eine Staging-Verteilung (CLI)

1. Verwenden Sie die Befehle aws cloudfront get-distribution und grep gemeinsam, um den `ETag`-Wert der Verteilung zu ermitteln, die Sie als primäre Verteilung verwenden möchten. Die primäre Verteilung ist die Verteilung, die zurzeit für den Produktionsdatenverkehr zuständig ist. Aus dieser Verteilung erstellen Sie die Staging-Verteilung.

   Im Folgenden wird ein Beispielbefehl gezeigt. Im folgenden Beispiel ersetzen Sie es *`primary_distribution_ID`* durch die ID der Primärdistribution.

   ```
   aws cloudfront get-distribution --id primary_distribution_ID | grep 'ETag'
   ```

   Kopieren Sie den `ETag`-Wert, da Sie ihn für den folgenden Schritt benötigen.
2. Verwenden Sie den Befehl aws cloudfront copy-distribution, um eine Staging-Verteilung zu erstellen. Im folgenden Beispielbefehl werden zur besseren Lesbarkeit Escape-Zeichen (\) und Zeilenumbrüche verwendet, Sie sollten diese jedoch im Befehl weglassen. Beachten Sie in dem Beispielbefehl Folgendes:

   - *`primary_distribution_ID`*Ersetzen Sie es durch die ID der Primärdistribution.

- *primary_distribution_ETag*Ersetzen Sie durch den `ETag` Wert der Primärverteilung (den Sie im vorherigen Schritt erhalten haben).
- (Optional) *CLI_example* Ersetzen Sie durch die gewünschte Anruferreferenz-ID.

```
aws cloudfront copy-distribution --primary-distribution-
id primary_distribution_ID \
                                 --if-match primary_distribution_ETag \
                                 --staging \
                                 --caller-reference 'CLI_example'
```

Die Ausgabe des Befehls enthält Informationen über die Staging-Verteilung und ihre Konfiguration. Kopieren Sie den CloudFront Domainnamen der Staging-Distribution, da Sie ihn für einen folgenden Schritt benötigen.

So erstellen Sie eine Richtlinie für die kontinuierliche Bereitstellung (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um eine Datei mit dem Namen `continuous-deployment-policy.yaml` zu erstellen, die alle Eingabeparameter für den Befehl create-continuous-deployment-policy enthält. Im folgenden Befehl werden zur besseren Lesbarkeit Escape-Zeichen (\) und Zeilenumbrüche verwendet, Sie sollten diese jedoch im Befehl weglassen.

```
aws cloudfront create-continuous-deployment-policy --generate-cli-skeleton yaml-
input \
                                                    > continuous-deployment-
policy.yaml
```

2. Öffnen Sie die Datei mit dem Namen `continuous-deployment-policy.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, um die gewünschten Einstellungen für die Richtlinie für die kontinuierliche Bereitstellung anzugeben, und speichern Sie die Datei. Gehen Sie beim Bearbeiten der Datei wie folgt vor:

   - Im Abschnitt `StagingDistributionDnsNames`:
     - Ändern Sie den Wert von `Quantity` in 1.

- Fügen Sie für `Items` den CloudFront Domainnamen der Staging-Distribution (den Sie aus einem vorherigen Schritt gespeichert haben) ein.

- Im Abschnitt `TrafficConfig`:
  - Wählen Sie einen `Type` aus, entweder `SingleWeight` oder `SingleHeader`.
  - Entfernen Sie die Einstellungen für den anderen Typ. Wenn Sie beispielsweise eine gewichtete Datenverkehrskonfiguration wünschen, legen Sie für `Type` `SingleWeight` fest und entfernen Sie dann die Einstellungen für `SingleHeaderConfig`.
  - Um eine gewichtete Datenverkehrskonfiguration zu verwenden, legen Sie als Wert für `Weight` eine Dezimalzahl zwischen `.01` (ein Prozent) und `.15` (fünfzehn Prozent) fest.

  Weitere Informationen zu den Optionen in `TrafficConfig` finden Sie unter the section called “Anfragen an die Staging-Distribution weiterleiten” und the section called “Sitzungs-Stickiness bei gewichtsbasierten Konfigurationen”.

3. Verwenden Sie den folgenden Befehl, um die Richtlinie für die kontinuierliche Bereitstellung unter Verwendung von Eingabeparametern aus der Datei `continuous-deployment-policy.yaml` zu erstellen.

```
aws cloudfront create-continuous-deployment-policy --cli-input-yaml file://
continuous-deployment-policy.yaml
```

   Kopieren Sie den `Id`-Wert in die Ausgabe des Befehls. Dies ist die ID der Richtlinie für die kontinuierliche Bereitstellung, die Sie in einem nachfolgenden Schritt benötigen werden.

So fügen Sie eine Richtlinie für die kontinuierliche Bereitstellung an eine primäre Verteilung an (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um die Konfiguration der primären Verteilung in einer Datei namens `primary-distribution.yaml` zu speichern. *`primary_distribution_ID`*Ersetzen Sie ihn durch die ID der Primärdistribution.

```
aws cloudfront get-distribution-config --id primary_distribution_ID --output
 yaml > primary-distribution.yaml
```

2. Öffnen Sie die Datei mit dem Namen `primary-distribution.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und nehmen Sie die folgenden Änderungen vor:

- Fügen Sie die ID der Richtlinie für die kontinuierliche Bereitstellung (die Sie aus einem vorherigen Schritt kopiert haben) in das Feld `ContinuousDeploymentPolicyId` ein.
- Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die primäre Verteilung so zu aktualisieren, dass die Richtlinie für die kontinuierliche Bereitstellung verwendet wird. Ersetze es *`primary_distribution_ID`* durch die ID der Primärdistribution.

```
aws cloudfront update-distribution --id primary_distribution_ID --cli-input-yaml
 file://primary-distribution.yaml
```

Wenn die Konfiguration der Primärverteilung mit der beigefügten Continuous Deployment Policy an Edge-Standorten bereitgestellt wird, CloudFront beginnt, den angegebenen Teil des Datenverkehrs auf der Grundlage der Verkehrskonfiguration an die Staging-Verteilung zu senden.

API

Verwenden Sie die folgenden API-Operationen, um mit der CloudFront API eine Richtlinie für die Staging-Verteilung und kontinuierliche Bereitstellung zu erstellen:

- CopyDistribution
- CreateContinuousDeploymentPolicy

Weitere Informationen zu den Feldern, die Sie in diesen API-Aufrufen angeben, finden Sie:

- the section called “Anfragen an die Staging-Distribution weiterleiten”
- the section called “Sitzungs-Stickiness bei gewichtsbasierten Konfigurationen”
- Die API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client

Nachdem Sie eine Staging-Verteilung und eine Continuous Deployment Policy erstellt haben, verwenden Sie UpdateDistribution(auf der Primärdistribution), um die Continuous Deployment Policy der Primärdistribution zuzuordnen.

Aktualisieren Sie eine Staging-Verteilung

Die folgenden Verfahren zeigen Ihnen, wie Sie eine Staging-Verteilung mit einer kontinuierlichen Bereitstellungsrichtlinie aktualisieren.

Console

Sie können bestimmte Konfigurationen sowohl für die Primär- als auch für die Staging-Distribution aktualisieren. Weitere Informationen finden Sie unter Aktualisieren Sie die Primär- und Staging-Distributionen.

So aktualisieren Sie eine Staging-Verteilung (Konsole)

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Rufen Sie im Navigationsbereich Distributions auf.
3. Wählen Sie die primäre Verteilung aus. Dies ist die Verteilung, die zurzeit für den Produktionsdatenverkehr zuständig ist. Aus dieser Verteilung haben Sie die Staging-Verteilung erstellt.
4. Wählen Sie View Staging Distribution (Staging-Verteilung anzeigen) aus.
5. Ändern Sie die Konfiguration der Staging-Verteilung über die Konsole. Weitere Informationen zu den aktualisierbaren Einstellungen finden Sie unter the section called “Aktualisieren Sie die Primär- und Staging-Distributionen”.

Sobald die Konfiguration der Staging-Verteilung an Edge-Standorten bereitgestellt ist, wird sie für den eingehenden Datenverkehr wirksam, der an die Staging-Verteilung weitergeleitet wird.

CLI

So aktualisieren Sie eine Staging-Verteilung (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um die Konfiguration der Staging-Verteilung in einer Datei namens `staging-distribution.yaml` zu speichern. *`staging_distribution_ID`*Ersetzen Sie es durch die ID der Staging-Distribution.

```
aws cloudfront get-distribution-config --id staging_distribution_ID --output
 yaml > staging-distribution.yaml
```

2. Öffnen Sie die Datei mit dem Namen `staging-distribution.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und nehmen Sie die folgenden Änderungen vor:

   - Ändern Sie die Konfiguration der Staging-Verteilung. Weitere Informationen zu den aktualisierbaren Einstellungen finden Sie unter the section called “Aktualisieren Sie die Primär- und Staging-Distributionen”.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

   Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Konfiguration der Staging-Verteilung zu aktualisieren. Ersetzen Sie es *`staging_distribution_ID`* durch die ID der Staging-Distribution.

```
aws cloudfront update-distribution --id staging_distribution_ID --cli-input-yaml
 file://staging-distribution.yaml
```

Sobald die Konfiguration der Staging-Verteilung an Edge-Standorten bereitgestellt ist, wird sie für den eingehenden Datenverkehr wirksam, der an die Staging-Verteilung weitergeleitet wird.

API

Um die Konfiguration einer Staging-Distribution zu aktualisieren, verwenden Sie UpdateDistribution(auf der Staging-Distribution), um die Konfiguration der Staging-Distribution zu ändern. Weitere Informationen zu den aktualisierbaren Einstellungen finden Sie unter the section called “Aktualisieren Sie die Primär- und Staging-Distributionen”.

Aktualisieren Sie eine Richtlinie für die kontinuierliche Bereitstellung

Die folgenden Verfahren zeigen Ihnen, wie Sie eine Richtlinie für die kontinuierliche Bereitstellung aktualisieren.

Console

Sie können die Verkehrskonfiguration Ihrer Distribution aktualisieren, indem Sie die Continuous Deployment Policy aktualisieren.

So aktualisieren Sie eine Richtlinie für die kontinuierliche Bereitstellung (Konsole)

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Rufen Sie im Navigationsbereich Distributions auf.
3. Wählen Sie die primäre Verteilung aus. Dies ist die Verteilung, die zurzeit für den Produktionsdatenverkehr zuständig ist. Aus dieser Verteilung haben Sie die Staging-Verteilung erstellt.
4. Wählen Sie im Abschnitt Continuous deployment (Kontinuierliche Bereitstellung) Edit policy (Richtlinie bearbeiten) aus.
5. Ändern Sie die Datenverkehrskonfiguration in der Richtlinie für die kontinuierliche Bereitstellung. Klicken Sie auf Save changes (Änderungen speichern), wenn Sie fertig sind.

Wenn die Konfiguration der Primärverteilung mit der aktualisierten Continuous Deployment Policy an Edge-Standorten bereitgestellt wird, CloudFront beginnt das Senden von Datenverkehr an die Staging-Verteilung auf der Grundlage der aktualisierten Datenverkehrskonfiguration.

CLI

So aktualisieren Sie eine Richtlinie für die kontinuierliche Bereitstellung (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um die Konfiguration der Richtlinie für die kontinuierliche Bereitstellung in einer Datei namens `continuous-deployment-policy.yaml` zu speichern. Ersetzen Sie es *`continuous_deployment_policy_ID`* durch die ID der Continuous Deployment Policy. Im folgenden Befehl werden zur besseren Lesbarkeit Escape-Zeichen (\) und Zeilenumbrüche verwendet, Sie sollten diese jedoch im Befehl weglassen.

```
aws cloudfront get-continuous-deployment-policy-config --
id continuous_deployment_policy_ID \
                                                    --output yaml >
 continuous-deployment-policy.yaml
```

2. Öffnen Sie die Datei mit dem Namen `continuous-deployment-policy.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und nehmen Sie die folgenden Änderungen vor:

- Ändern Sie die Konfiguration der Richtlinie für die kontinuierliche Bereitstellung wie gewünscht. Sie können beispielsweise von einer Header-basierten zu einer gewichteten Datenverkehrskonfiguration übergehen oder den Prozentsatz des Datenverkehrs (Gewichtung) durch eine gewichtete Konfiguration ersetzen. Weitere Informationen erhalten Sie unter the section called “Anfragen an die Staging-Distribution weiterleiten” und the section called “Sitzungs-Stickiness bei gewichtsbasierten Konfigurationen”.
- Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Richtlinie für die kontinuierliche Bereitstellung zu aktualisieren. *`continuous_deployment_policy_ID`*Ersetzen Sie es durch die ID der Richtlinie für die kontinuierliche Bereitstellung. Im folgenden Befehl werden zur besseren Lesbarkeit Escape-Zeichen (\) und Zeilenumbrüche verwendet, Sie sollten diese jedoch im Befehl weglassen.

```
aws cloudfront update-continuous-deployment-policy --
id continuous_deployment_policy_ID \
                                                    --cli-input-yaml file://
continuous-deployment-policy.yaml
```

Wenn die Konfiguration der Primärverteilung mit der aktualisierten Continuous Deployment Policy an Edge-Standorten bereitgestellt wird, CloudFront beginnt das Senden von Datenverkehr an die Staging-Verteilung auf der Grundlage der aktualisierten Verkehrskonfiguration.

API

Um eine Richtlinie für die kontinuierliche Bereitstellung zu aktualisieren, verwenden Sie. UpdateContinuousDeploymentPolicy

Werben Sie für eine Konfiguration der Staging-Verteilung

Die folgenden Verfahren zeigen Ihnen, wie Sie eine Staging-Verteilungskonfiguration heraufstufen.

Console

Wenn Sie eine Staging-Distribution heraufstufen, wird die Konfiguration von der Staging-Distribution in die primäre Distribution CloudFront kopiert. CloudFront deaktiviert außerdem die Continuous Deployment Policy und leitet den gesamten Datenverkehr an die Primärverteilung weiter.

Nachdem Sie eine Konfiguration hochgestuft haben, können Sie die betreffende Staging-Verteilung erneut verwenden, wenn Sie das nächste Mal eine Konfigurationsänderung testen möchten.

So stufen Sie die Konfiguration einer Staging-Verteilung hoch (Konsole)

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Rufen Sie im Navigationsbereich Distributions auf.
3. Wählen Sie die primäre Verteilung aus. Dies ist die Verteilung, die zurzeit für den Produktionsdatenverkehr zuständig ist. Aus dieser Verteilung haben Sie die Staging-Verteilung erstellt.
4. Wählen Sie im Abschnitt Continuous deployment (Kontinuierliche Bereitstellung) Promote (Hochstufen) aus.
5. Geben Sie **`confirm`** ein und wählen Sie dann Promote (Hochstufen) aus.

CLI

Wenn Sie eine Staging-Distribution heraufstufen, CloudFront kopiert die Konfiguration aus der Staging-Distribution in die primäre Distribution. CloudFront deaktiviert außerdem die Continuous Deployment Policy und leitet den gesamten Datenverkehr an die Primärverteilung weiter.

Nachdem Sie eine Konfiguration hochgestuft haben, können Sie die betreffende Staging-Verteilung erneut verwenden, wenn Sie das nächste Mal eine Konfigurationsänderung testen möchten.

So stufen Sie die Konfiguration einer Staging-Verteilung hoch (CLI)

- Verwenden Sie den Befehl aws cloudfront update-distribution-with-staging-config, um die Konfiguration der Staging-Verteilung auf die primäre Verteilung hochzustufen. Im folgenden Beispielbefehl werden zur besseren Lesbarkeit Escape-Zeichen (\) und Zeilenumbrüche

verwendet, Sie sollten diese jedoch im Befehl weglassen. Beachten Sie in dem Beispielbefehl Folgendes:

- *primary_distribution_ID*Ersetzen Sie es durch die ID der Primärverteilung.
- *staging_distribution_ID*Ersetzen Sie es durch die ID der Staging-Distribution.
- Ersetzen Sie *primary_distribution_ETag* und *staging_distribution_ETag* durch die `ETag` Werte der Primär- und Stagingverteilung. Stellen Sie sicher, dass der Wert der primären Verteilung an erster Stelle steht, wie im Beispiel dargestellt.

```
aws cloudfront update-distribution-with-staging-config --
id primary_distribution_ID \
                                                    --staging-distribution-
id staging_distribution_ID \
                                                    --if-match
 'primary_distribution_ETag,staging_distribution_ETag'
```

API

Um die Konfiguration einer Staging-Distribution auf die Primärdistribution hochzustufen, verwenden Sie. UpdateDistributionWithStagingConfig

Überwachen Sie eine Staging-Verteilung

Um die Leistung einer Staging-Verteilung zu überwachen, können Sie dieselben Metriken, Protokolle und Berichte verwenden, die für alle Verteilungen CloudFront verfügbar sind. Zum Beispiel:

- Sie können die CloudFrontStandard-Verteilungsmetriken (wie Gesamtzahl der Anfragen und Fehlerrate) in der CloudFront Konsole anzeigen und gegen Aufpreis zusätzliche Messwerte (wie Cache-Trefferrate und Fehlerrate nach Statuscode) aktivieren. Zudem können Sie Alarme basierend auf diesen Metriken erstellen.
- Sie können Standardprotokolle und Echtzeitprotokolle einsehen, um detaillierte Informationen zu den Anforderungen zu erhalten, die von der Staging-Verteilung empfangen werden. Standardprotokolle enthalten die folgenden zwei Felder, anhand derer Sie die primäre Distribution identifizieren können, an die die Anfrage ursprünglich gesendet wurde, bevor sie an die Staging-

Verteilung CloudFront weitergeleitet wurde: `primary-distribution-id` und. `primary-distribution-dns-name`

- In der CloudFront Konsole können Sie Berichte anzeigen und herunterladen, z. B. den Cache-Statistikbericht.

Erfahren Sie, wie Continuous Deployment funktioniert

In den folgenden Themen wird erklärt, wie CloudFront Continuous Deployment funktioniert.

Themen

Anfragen an die Staging-Distribution weiterleiten

Wenn Sie CloudFront Continuous Deployment verwenden, müssen Sie nichts an den Viewer-Anfragen ändern. Viewer haben nicht die Möglichkeit, unter Verwendung eines DNS-Namens, einer IP-Adresse oder eines CNAME Anforderungen direkt an eine Staging-Verteilung zu senden. Stattdessen senden Zuschauer Anfragen an die primäre (Produktions-) Distribution und leiten einige dieser Anfragen auf CloudFront der Grundlage der Datenverkehrskonfigurationse instellungen in der Continuous Deployment Policy an die Staging-Distribution weiter. Es gibt zwei Arten von Datenverkehrskonfigurationen:

Gewichtet

Bei einer gewichtsbasierten Konfiguration wird der angegebene Prozentsatz der Viewer-Anforderungen an die Staging-Verteilung weitergeleitet. Wenn Sie eine gewichtsbasierte Konfiguration verwenden, können Sie auch die Sitzungsbindung aktivieren, wodurch sichergestellt wird, dass Anfragen desselben Betrachters als Teil einer einzigen Sitzung CloudFront behandelt werden. Weitere Informationen finden Sie unter the section called “Sitzungs-Stickiness bei gewichtsbasierten Konfigurationen”.

Header-basiert

Bei einer Header-basierten Konfiguration werden Anforderungen an die Staging-Verteilung weitergeleitet, wenn die Viewer-Anforderung einen bestimmten HTTP-Header enthält (den Header

und Wert geben Sie selbst an). Anforderungen, die den angegebenen Header und Wert nicht enthalten, werden an die primäre Verteilung weitergeleitet. Diese Konfiguration ist hilfreich, wenn Sie lokale Tests durchführen oder die Kontrolle über die Viewer-Anforderungen haben.

> **Note**
>
> Header, die an Ihre Staging-Verteilung weitergeleitet werden, müssen das Präfix `aws-cf-cd-` enthalten.

Sitzungs-Stickiness bei gewichtsbasierten Konfigurationen

Wenn Sie eine gewichtsbasierte Konfiguration verwenden, um Traffic an eine Staging-Verteilung weiterzuleiten, können Sie auch die Sitzungsbindung aktivieren, wodurch sichergestellt wird, dass Anfragen von demselben Viewer als eine einzelne Sitzung CloudFront behandelt werden. Wenn Sie Session Stickiness aktivieren, CloudFront wird ein Cookie gesetzt, sodass alle Anfragen desselben Viewers in einer einzigen Sitzung von einer Distribution bedient werden, entweder von der Primär- oder der Staging-Distribution.

Wenn Sie Sitzungs-Stickiness aktivieren, können Sie auch die Leerlaufdauer (idle duration) angeben. Wenn der Viewer für diese Zeit inaktiv ist (keine Anfragen sendet), läuft die Sitzung ab und CloudFront behandelt future Anfragen von diesem Viewer als neue Sitzung. Sie geben die Leerlaufdauer in Sekunden an. Möglich sind dabei Werte von 300 (fünf Minuten) bis 3 600 Sekunden (einer Stunde).

CloudFront Setzt in den folgenden Fällen alle Sitzungen (auch aktive) zurück und betrachtet alle Anfragen als neue Sitzung:

- Sie deaktivieren oder aktivieren die Richtlinie für die kontinuierliche Bereitstellung.
- Sie deaktivieren oder aktivieren die Einstellung für Sitzungs-Stickiness.

Aktualisieren Sie die Primär- und Staging-Distributionen

Wenn einer primären Verteilung eine Richtlinie für die kontinuierliche Bereitstellung angefügt ist, sind die folgenden Konfigurationsänderungen sowohl für die primäre Verteilung als auch für die Staging-Verteilung verfügbar:

- Alle Einstellungen in Bezug auf das Cache-Verhalten, einschließlich des Standard-Cache-Verhaltens

- Alle Ursprungseinstellungen (Ursprünge und Ursprungsgruppen)
- Benutzerdefinierte Fehlerreaktionen (Fehlerseiten)
- Geografische Einschränkungen
- Standardstammobjekt
- Protokollierungseinstellungen
- Beschreibung (Kommentar)

Sie können auch externe Ressourcen aktualisieren, auf die in der Konfiguration einer Distribution verwiesen wird, z. B. eine Cache-Richtlinie, eine Response-Header-Richtlinie, eine CloudFront Funktion oder eine Lambda @Edge -Funktion.

Primäre Verteilungen und Staging-Verteilungen nutzen nicht denselben Cache

Primäre Verteilungen und Staging-Verteilungen nutzen nicht denselben Cache. Wenn die erste Anfrage an eine Staging-Distribution CloudFront gesendet wird, ist deren Cache leer. Wenn Anforderungen bei der Staging-Verteilung ankommen, beginnt diese mit dem Zwischenspeichern der Antworten (sofern entsprechend konfiguriert).

Kontingente und andere zu berücksichtigende Aspekte bei der kontinuierlichen Bereitstellung

CloudFront Für die kontinuierliche Bereitstellung gelten die folgenden Kontingente und weitere Überlegungen.

Kontingente

- Maximale Anzahl von Staging-Verteilungen pro AWS-Konto: 20
- Maximale Anzahl von Richtlinien für die kontinuierliche Bereitstellung pro AWS-Konto: 20
- Maximaler Prozentsatz des Datenverkehrs, den Sie bei einer gewichtsbasierten Konfiguration an eine Staging-Verteilung senden können: 15 %
- Mindest- und Höchstwerte für die Leerlaufdauer bei Sitzungs-Stickiness: 300–3 600 Sekunden

Weitere Informationen finden Sie unter Kontingente.

> **Note**
>
> Wenn Sie Continuous Deployment verwenden und Ihre primäre Distribution mit OAC für den S3-Bucket-Zugriff eingerichtet ist, aktualisieren Sie Ihre S3-Bucket-Richtlinie, um den Zugriff für die Staging-Distribution zu ermöglichen. Beispiele für S3-Bucket-Richtlinien finden Sie unter. the section called “Erteilen Sie die CloudFront Erlaubnis, auf den S3-Bucket zuzugreifen”

AWS WAF Web ACLs

Wenn Sie den kontinuierlichen Vertrieb für Ihre Distribution aktivieren, gelten die folgenden Überlegungen für AWS WAF:

- Sie können der Verteilung keine AWS WAF Web Access Control List (ACL) zuordnen, wenn dies das erste Mal ist, dass diese ACL der Verteilung zugeordnet wurde.
- Sie können eine AWS WAF Web-ACL nicht von der Verteilung trennen.

Bevor Sie die oben genannten Aufgaben ausführen können, müssen Sie die Continuous Deployment Policy für Ihre Produktionsdistribution löschen. Dadurch wird auch die Staging-Verteilung gelöscht. Weitere Informationen finden Sie unter AWS WAF Schutzmaßnahmen verwenden.

Fälle, in denen alle Anfragen an die Primärverteilung CloudFront gesendet werden

In bestimmten Fällen, z. B. in Zeiten hoher Ressourcenauslastung, werden CloudFront möglicherweise alle Anfragen an die primäre Verteilung gesendet, unabhängig davon, was in der Richtlinie für die kontinuierliche Bereitstellung festgelegt ist.

CloudFront sendet zu Spitzenzeiten alle Anfragen an die primäre Distribution, unabhängig davon, was in der Richtlinie für die kontinuierliche Bereitstellung festgelegt ist. Spitzenverkehr bezieht sich auf den Verkehr auf dem CloudFront Service und nicht auf den Verkehr auf Ihrer Distribution.

HTTP/3

Sie können die kontinuierliche Bereitstellung nicht mit einer Verteilung verwenden, die HTTP/3 unterstützt.

Verwenden Sie Benutzerdefiniert, URLs indem Sie alternative Domainnamen hinzufügen (CNAMEs)

Wenn Sie eine Distribution erstellen, CloudFront stellt sie einen Domainnamen bereit, z. B. d111111abcdef8.cloudfront.net. Anstatt diesen bereitgestellten Domainnamen zu verwenden, können Sie einen alternativen Domainnamen (auch als CNAME bezeichnet) verwenden.

In den folgenden Themen erfahren Sie, wie Sie Ihren eigenen Domainnamen verwenden können, z. B. www.example.com:

Themen

Voraussetzungen für die Verwendung von alternativen Domänennamen

Wenn Sie einer CloudFront Distribution einen alternativen Domainnamen wie www.example.com hinzufügen, gelten die folgenden Voraussetzungen:

Alternative Domänennamen müssen Kleinbuchstaben verwenden

Alle alternativen Domainnamen (CNAMEs) müssen in Kleinbuchstaben geschrieben werden.

Alternative Domainnamen müssen durch ein gültiges TLS-Zertifikat abgedeckt sein

Um einer Distribution einen alternativen Domainnamen (CNAME) hinzuzufügen, müssen Sie Ihrer CloudFront Distribution ein vertrauenswürdiges, gültiges TLS-Zertifikat beifügen, das den alternativen Domainnamen abdeckt. Dadurch wird sichergestellt, dass nur Personen, die Zugriff auf das Zertifikat Ihrer Domain haben, eine Verbindung zu CloudFront einem CNAME herstellen können, der sich auf Ihre Domain bezieht.

Ein vertrauenswürdiges Zertifikat ist ein Zertifikat, das von AWS Certificate Manager (ACM) oder einer anderen gültigen Zertifizierungsstelle (CA) ausgestellt wurde. Sie können ein

selbstsigniertes Zertifikat verwenden, um einen vorhandenen CNAME zu validieren, aber nicht für einen neuen CNAME. CloudFront unterstützt dieselben Zertifizierungsstellen wie Mozilla. Die aktuelle Liste finden Sie unter Liste der CA-Zertifikate für Mozilla. Hinweise zu Zwischenzertifikaten bei der Verwendung einer Zertifizierungsstelle eines Drittanbieters finden Sie unterZwischenzertifikate.

Um einen alternativen Domainnamen mithilfe des von Ihnen angehängten Zertifikats zu verifizieren, einschließlich alternativer Domänennamen, die Platzhalter enthalten, wird der alternative Name (SAN) auf dem Zertifikat CloudFront überprüft. Der alternative Domänenname, den Sie hinzufügen, muss vom SAN abgedeckt werden.

> **Note**
>
> Einer CloudFront Distribution kann jeweils nur ein Zertifikat angehängt werden.

Sie weisen Ihre Berechtigung zum Hinzufügen eines spezifischen alternativen Domänennamens zu Ihrer Distribution nach, indem Sie eine der folgenden Aktionen ausführen:

- Anfügen eines Zertifikats, das den alternativen Domänennamen enthält, z. B product-name.example.com.
- Anfügen eines Zertifikats, das mit dem Platzhalterzeichen * am Anfang eines Domänennamens beginnt, um mehrere Unterdomänen durch ein einziges Zertifikat abzudecken. Wenn Sie ein Platzhalterzeichen verwenden, können Sie mehrere Unterdomänen als alternative Domänennamen in CloudFront hinzufügen.

Die folgenden Beispiele zeigen, wie die Verwendung von Platzhalterzeichen in Domänennamen in einem Zertifikat funktioniert und Sie dazu berechtigt, spezifische alternative Domänennamen in CloudFront hinzuzufügen.

- Sie möchten marketing.example.com als einen alternativen Domänennamen hinzufügen. Sie listen in Ihrem Zertifikat den folgenden Domänennamen auf: *.example.com. Wenn Sie dieses Zertifikat anhängen CloudFront, können Sie einen beliebigen alternativen Domainnamen für Ihre Distribution hinzufügen, der den Platzhalter auf dieser Ebene ersetzt, einschließlich marketing.example.com. Sie können beispielsweise auch die folgenden alternativen Domänennamen hinzufügen:
  - product.example.com
  - api.example.com

Sie können jedoch keine alternativen Domänennamen auf höheren oder niedrigeren Ebenen als der Platzhalterebene hinzufügen. Beispielsweise können Sie nicht die alternativen Domänennamen example.com oder marketing.product.example.com hinzufügen.

- Sie möchten example.com als einen alternativen Domänennamen hinzufügen. Hierzu müssen Sie den Domänennamen example.com in dem Zertifikat auflisten, das Sie Ihrer Verteilung anfügen.
- Sie möchten marketing.product.example.com als einen alternativen Domänennamen hinzufügen. Dazu können Sie *.product.example.com im Zertifikat auflisten oder marketing.product.example.com selbst im Zertifikat auflisten.

Berechtigung zum Ändern der DNS-Konfiguration

Wenn Sie alternative Domainnamen hinzufügen, müssen Sie CNAME-Einträge erstellen, um DNS-Abfragen für die alternativen Domainnamen an Ihre Distribution weiterzuleiten. CloudFront Hierzu müssen Sie beim DNS-Serviceanbieter für die von Ihnen verwendeten alternativen Domänennamen die Berechtigung zum Erstellen von CNAME-Datensätzen besitzen. Das bedeutet in der Regel, dass Sie der Besitzer der Domänen sind. Es kann aber auch bedeuten, dass Sie eine Anwendung für den Besitzer der Domäne entwickeln.

Alternative Domänennamen und HTTPS

Wenn Sie möchten, dass Viewer in Verbindung mit alternativen Domänennamen HTTPS verwenden, sind zusätzliche Konfigurationsschritte erforderlich. Weitere Informationen finden Sie unter Verwenden Sie alternative Domainnamen und HTTPS.

Einschränkungen bei der Verwendung alternativer Domänennamen

Beachten Sie die folgenden Einschränkungen bei der Verwendung alternativer Domänennamen:

Maximale Anzahl der alternativen Domänennamen

Informationen zur aktuell gültigen maximalen Anzahl von alternativen Domänennamen, die Sie einer Verteilung hinzufügen können, oder zum Anfordern eines höheren Kontingents (früher als Limit bezeichnet) finden Sie unter Allgemeine Kontingente für Verteilungen.

Doppelte und sich überschneidende alternative Domänennamen

Sie können einer CloudFront Distribution keinen alternativen Domainnamen hinzufügen, wenn derselbe alternative Domainname bereits in einer anderen CloudFront Distribution existiert, auch wenn Ihnen die andere Distribution AWS-Konto gehört.

Sie können jedoch einen alternativen Domänennamen mit Platzhalterzeichen hinzufügen (z. B. *.example.com), der einen alternativen Domänennamen ohne Platzhalterzeichen enthält (d. h. sich mit diesem überschneidet), z. B. www.example.com. Wenn sich alternative Domainnamen in zwei Verteilungen überschneiden, CloudFront sendet die Anfrage an die Distribution mit der genaueren Namensübereinstimmung, unabhängig von der Verteilung, auf die der DNS-Eintrag verweist. Beispielsweise ist marketing.domain.com spezifischer als *.domain.com.

Wenn Sie bereits über einen DNS-Platzhaltereintrag verfügen, der auf eine CloudFront Verteilung verweist, und Sie beim Versuch, einen neuen CNAME mit einem genaueren Namen hinzuzufügen, einen falsch konfigurierten DNS-Fehler erhalten, finden Sie weitere Informationen unter. CloudFront gibt einen falsch konfigurierten DNS-Eintragsfehler zurück, wenn ich versuche, einen neuen CNAME hinzuzufügen

Domänen-Fronting

CloudFront umfasst Schutz vor Domain-Fronting, das zwischen verschiedenen Konten auftritt. AWS Domain-Fronting ist ein Szenario, in dem ein nicht standardmäßiger Client eine TLS-Verbindung zu einem Domainnamen in einem Konto herstellt AWS-Konto, dann aber eine HTTPS-Anfrage für einen nicht verwandten Namen in einem anderen Konto stellt. AWS Die TLS-Verbindung kann beispielsweise eine Verbindung zu www.example.com herstellen und anschließend eine HTTP-Anfrage für www.example.org ausgeben.

Um zu verhindern, dass sich CloudFront das Domain-Fronting nicht überschneidet AWS-Konten, sollten Sie sicherstellen, AWS-Konto dass das Zertifikat, das es für eine bestimmte Verbindung bereitstellt, immer mit dem übereinstimmt AWS-Konto , dem die Anfrage gehört, die es für dieselbe Verbindung bearbeitet.

Wenn die beiden AWS-Konto Zahlen nicht übereinstimmen, CloudFront antwortet die Antwort mit einer fehlgeleiteten HTTP-421-Anfrage, sodass der Client die Möglichkeit hat, über die richtige Domäne eine Verbindung herzustellen.

Hinzufügen eines alternativen Domänennamens im obersten Knoten (Zone Apex) einer Domäne

Wenn Sie einer Distribution einen alternativen Domainnamen hinzufügen, erstellen Sie in der Regel einen CNAME-Eintrag in Ihrer DNS-Konfiguration, um DNS-Abfragen für den Domainnamen an Ihre CloudFront Distribution weiterzuleiten. Sie können jedoch keinen CNAME-Datensatz für den obersten Knoten eines DNS-Namespace (auch als Zone Apex bezeichnet) erstellen; das DNS-Protokoll lässt dies nicht zu. Wenn Sie beispielsweise den DNS-Namen example.com registriert haben, lautet der Zone Apex example.com. Sie können

keinen CNAME-Datensatz für example.com erstellen, Sie können jedoch CNAME-Datensätze für www.example.com, newproduct.example.com und so weiter erstellen.

Wenn Sie Route 53 als DNS-Dienst verwenden, können Sie einen Alias-Ressourcendatensatz erstellen, der gegenüber CNAME-Einträgen die folgenden Vorteile bietet:

- Sie können einen Alias-Ressourcendatensatz für einen Domänennamen im obersten Knoten (example.com) erstellen.
- Sie können einen HTTPS-Eintrag für einen alternativen Domainnamen erstellen, um die Protokollaushandlung als Teil der DNS-Suche zu ermöglichen, sofern der Client dies unterstützt. Weitere Informationen finden Sie unter Create alias resource record set.
- Sie zahlen nicht für Route 53-Abfragen, wenn Sie einen Alias-Ressourcendatensatz verwenden.

> **Note**
>
> Wenn Sie diese Option aktivieren IPv6, müssen Sie zwei Alias-Ressourcendatensätze erstellen: einen für die Weiterleitung des IPv4 Datenverkehrs (ein A-Datensatz) und einen für die Weiterleitung des IPv6 Datenverkehrs (ein AAAA-Datensatz). Weitere Informationen finden Sie unter Aktivieren IPv6 im Thema Referenz für alle Verteilungseinstellungen.

Weitere Informationen finden Sie unter Weiterleiten von Datenverkehr an eine CloudFront Amazon-Webdistribution mithilfe Ihres Domainnamens im Amazon Route 53-Entwicklerhandbuch.

Wenn Sie Route 53 nicht für Ihr DNS verwenden, können Sie statische Anycast-IP-Adressen anfordern, an die Apex-Domains wie example.com weitergeleitet werden. CloudFront Weitere Informationen finden Sie unter Fordere Anycast static an, um es für die Zulassungsliste zu verwenden IPs .

Fügen Sie einen alternativen Domainnamen hinzu

In der folgenden Aufgabenliste wird beschrieben, wie Sie mit der CloudFront Konsole Ihrer Distribution einen alternativen Domainnamen hinzufügen, sodass Sie in Ihren Links Ihren eigenen Domainnamen anstelle des CloudFront Domainnamens verwenden können. Informationen zur Aktualisierung Ihrer Distribution mithilfe der CloudFront API finden Sie unterDistributionen konfigurieren.

> **Note**
>
> Wenn Sie möchten, dass Viewer HTTPS in Verbindung mit Ihrem alternativen Domain-Namen verwenden, finden Sie weitere Informationen unter Verwenden Sie alternative Domainnamen und HTTPS.

Bevor Sie beginnen: Sie müssen die folgenden Schritte ausführen, bevor Sie Ihre Verteilung aktualisieren, um einen alternativen Domänennamen hinzuzufügen:

- Registrieren Sie den Domänennamen bei Route 53 oder einem anderen Domänenregister.
- Besorgen Sie sich ein TLS-Zertifikat von einer autorisierten Zertifizierungsstelle (CA), die den Domainnamen abdeckt. Fügen Sie das Zertifikat Ihrer Verteilung hinzu, um zu überprüfen, ob Sie berechtigt sind, die Domäne zu verwenden. Weitere Informationen finden Sie unter Voraussetzungen für die Verwendung von alternativen Domänennamen.

Fügen Sie einen alternativen Domainnamen hinzu

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die ID für die Verteilung aus, die Sie aktualisieren möchten.
3. Wählen Sie auf der Registerkarte Allgemein die Option Domain hinzufügen aus.
4. Geben Sie bis zu fünf Domains ein, die bedient werden sollen.
5. Wählen Sie Weiter aus.
6. Wenn CloudFront Sie beim TLS-Zertifikat kein vorhandenes AWS Certificate Manager (ACM-) Zertifikat für Ihre Domain AWS-Konto in Ihrem finden, können Sie wählen `us-east-1` AWS-Region, ob Sie ein Zertifikat automatisch oder manuell in ACM erstellen möchten.
7. Wenn Ihr Zertifikat bereitgestellt wird, müssen Sie Ihre DNS-Einträge bei Ihrem DNS-Anbieter aktualisieren, um den Besitz der Domain nachzuweisen. Die Einträge, die Sie in Ihren DNS-Einträgen vornehmen müssen, werden Ihnen in der CloudFront Konsole zur Verfügung gestellt.
8. Nachdem Sie Ihre DNS-Einträge aktualisiert haben, wählen Sie Zertifikat validieren aus.
9. Wenn das Zertifikat validiert ist, wählen Sie Weiter.
10. Überprüfen Sie Ihre Änderungen und wählen Sie Domains hinzufügen.
11. Vergewissern Sie sich auf der Registerkarte General für die Verteilung, dass unter Distribution Status der Wert Deployed angezeigt wird. Wenn Sie versuchen, einen alternativen Domain-

Namen zu verwenden, bevor die Aktualisierungen für Ihre Verteilung übertragen wurden, kann es sein, dass die Links, die Sie in den folgenden Schritten erstellen, nicht funktionieren.

12. Konfigurieren Sie den DNS-Dienst für den alternativen Domainnamen (z. B. www.example.com), um den Datenverkehr an den CloudFront Domainnamen für Ihre Distribution weiterzuleiten (z. B. d111111abcdef8.cloudfront.net). Die verwendete Methode ist davon abhängig, ob Sie Route 53 als DNS-Serviceanbieter oder einen anderen Anbieter für die Domäne verwenden. Weitere Informationen finden Sie unter Fügen Sie Ihrer CloudFront Standarddistribution eine Domain hinzu.

    Route 53

    Erstellen Sie einen Alias-Ressourcendatensatz. Mit einem Alias-Ressourcendatensatz bezahlen Sie keine Gebühren für Route 53-Abfragen. Sie können auch einen Alias-Ressourceneintrag für den Root-Domainnamen (example.com) erstellen, was DNS nicht zulässt. CNAMEs Anweisungen zum Erstellen eines Alias-Ressourcendatensatzes finden Sie unter Weiterleiten von Datenverkehr an eine CloudFront Amazon-Webdistribution mithilfe Ihres Domainnamens im Amazon Route 53-Entwicklerhandbuch.

    Optional können Sie einen HTTPS-Eintrag für einen alternativen Domainnamen erstellen, um die Protokollaushandlung als Teil der DNS-Suche zu ermöglichen, sofern der Client dies unterstützt.

    Um einen Alias-Ressourcendatensatz mit einem HTTPS-Eintrag zu erstellen (optional)

    1. Aktivieren Sie HTTP/2 oder HTTP/3 in Ihren CloudFront Distributionseinstellungen. Weitere Informationen erhalten Sie unter Unterstützte HTTP-Versionen und Eine Verteilung aktualisieren.
    2. Erstellen Sie in der Route 53-Konsole einen Alias-Ressourcendatensatz. Folgen Sie den Anweisungen zur Weiterleitung des Datenverkehrs an eine CloudFront Amazon-Webdistribution, indem Sie Ihr Domainnamen-Verfahren verwenden.
    3. Während Sie den Alias-Ressourcendatensatz erstellen, erstellen Sie einen Alias-Datensatz mit dem Datensatztyp HTTPS.

    Anderer DNS-Serviceanbieter

    Verwenden Sie die von Ihrem DNS-Serviceanbieter bereitgestellte Methode, um einen CNAME-Eintrag für Ihre Domäne hinzuzufügen. Dieser neue CNAME-Eintrag leitet DNS-Anfragen von Ihrem alternativen Domainnamen (z. B. www.example.com) an den CloudFront

Domainnamen für Ihre Distribution weiter (z. B. d111111abcdef8.cloudfront.net). Weitere Informationen finden Sie in der Dokumentation Ihres DNS-Serviceanbieters.

> **Important**
>
> Wenn Sie bereits einen CNAME-Eintrag für Ihren alternativen Domainnamen haben, aktualisieren Sie diesen Eintrag oder ersetzen Sie ihn durch einen neuen, der auf den Domainnamen für Ihre Distribution verweist. CloudFront

13. Wenn Sie `dig` oder ein ähnliches DNS-Tool verwenden, vergewissern Sie sich, dass die DNS-Konfiguration, die Sie im vorherigen Schritt erstellt haben, auf den Domänennamen für Ihre Verteilung verweist.

    Das folgende Beispiel zeigt eine `dig`-Anfrage für die Domäne images.example.com sowie den relevanten Teil der Antwort.

    ```
    PROMPT> dig www.example.com

    ; <<> DiG 9.3.3rc2 <<> www.example.com
    ;; global options: printcmd
    ;; Got answer:
    ;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15917
    ;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 2, ADDITIONAL: 0

    ;; QUESTION SECTION:
    ;www.example.com.     IN    A

    ;; ANSWER SECTION:
    www.example.com. 10800 IN CNAME d111111abcdef8.cloudfront.net.
    ...
    ```

    Der Antwortbereich zeigt einen CNAME-Eintrag, der Anfragen für www.example.com an den CloudFront Distributionsdomänennamen d111111abcdef8.cloudfront.net weiterleitet. Wenn der Name auf der rechten Seite von der Domainname für Ihre Distribution ist, `CNAME` ist der CNAME-Eintrag korrekt konfiguriert. CloudFront Wird hier ein anderer Wert angezeigt, z. B. der Domänenname für Ihren Amazon-S3-Bucket, dann ist der CNAME-Datensatz nicht korrekt konfiguriert. Beginnen Sie in diesem Fall erneut mit Schritt 7 und korrigieren Sie den CNAME-Datensatz so, dass er auf den Domänennamen für Ihre Verteilung verweist.

14. Testen Sie den alternativen Domainnamen, indem URLs Sie Ihren Domainnamen anstelle des CloudFront Domainnamens für Ihren Vertrieb angeben.
15. Ändern Sie in Ihrer Anwendung die Option URLs für Ihre Objekte, sodass Ihr alternativer Domainname anstelle des Domainnamens Ihrer CloudFront Distribution verwendet wird.

Verschieben Sie einen alternativen Domainnamen

Wenn Sie versuchen, einem Standardverteilungs- oder Distributionsmandanten einen alternativen Domainnamen hinzuzufügen, und der alternative Domainname bereits mit einer anderen Ressource verknüpft ist, erhalten Sie eine Fehlermeldung.

Wenn Sie beispielsweise versuchen, www.example.com zu einem Standardverteilungs- oder Distributionsmandanten hinzuzufügen, erhalten CNAMEs Sie beispielsweise die `CNAMEAlreadyExists` Fehlermeldung (Ein oder mehrere der von Ihnen angegebenen sind bereits mit einer anderen Ressource verknüpft), dieser alternative Domainname jedoch bereits mit einer anderen Ressource verknüpft ist.

In diesem Fall möchten Sie möglicherweise den vorhandenen alternativen Domainnamen von einer Ressource auf eine andere verschieben. Dies ist die Quellverteilung und die Zielverteilung. Sie können alternative Domainnamen zwischen beiden and/or Standardverteilungsmandanten verschieben.

Informationen zum Verschieben des alternativen Domainnamens finden Sie in den folgenden Themen:

Themen

- Richten Sie den Ziel-Standardverteilungs- oder Distributionsmandanten ein
- Suchen Sie den Mandanten für die Standardverteilung oder -verteilung als Quelle
- Einen alternativen Domänennamen verschieben

Richten Sie den Ziel-Standardverteilungs- oder Distributionsmandanten ein

Bevor Sie einen alternativen Domainnamen verschieben können, müssen Sie die Zielressource einrichten. Dies ist der Ziel-Standardverteilungs- oder Distributionsmandant, auf den Sie den alternativen Domainnamen verschieben.

Standard distribution

Um eine Ziel-Standardverteilung einzurichten

1. Fordern Sie ein TLS-Zertifikat an. Dieses Zertifikat enthält den alternativen Domainnamen als Subject oder Subject Alternative Domain (SAN) oder einen Platzhalter (*), der den alternativen Domainnamen abdeckt, den Sie verschieben möchten. Wenn Sie noch keines haben, können Sie eines bei AWS Certificate Manager (ACM) oder einer anderen Zertifizierungsstelle (CA) anfordern und in ACM importieren.

   > **Note**
   >
   > Sie müssen das Zertifikat in der Region USA Ost (Nord-Virginia) (`us-east-1`) anfordern oder importieren.

   Weitere Informationen finden Sie unter Anfordern eines öffentlichen Zertifikats über die Konsole und Importieren eines Zertifikats AWS Certificate Manager im im AWS Certificate Manager Benutzerhandbuch.

2. Wenn Sie die Ziel-Standarddistribution noch nicht erstellt haben, erstellen Sie jetzt eine. Ordnen Sie im Rahmen der Erstellung der Standardverteilung das Zertifikat dieser Standardverteilung zu. Weitere Informationen finden Sie unter Eine Verteilung erstellen.

   Wenn Sie bereits über eine Ziel-Standardverteilung verfügen, ordnen Sie das Zertifikat der Standardverteilung zu. Weitere Informationen finden Sie unter Eine Verteilung aktualisieren.

3. Wenn Sie alternative Domainnamen innerhalb derselben Domain verschieben AWS-Konto, überspringen Sie diesen Schritt.

   Um einen alternativen Domainnamen von einem AWS-Konto auf einen anderen zu verschieben, müssen Sie in Ihrer DNS-Konfiguration einen TXT-Eintrag erstellen. Dieser Bestätigungsschritt trägt dazu bei, unbefugte Domainübertragungen zu verhindern. CloudFront verwendet diesen TXT-Eintrag, um Ihre Inhaberschaft des alternativen Domainnamens zu bestätigen.

   Erstellen Sie in Ihrer DNS-Konfiguration einen DNS-TXT-Eintrag, der den alternativen Domainnamen der Zielstandardverteilung zuordnet. Das Format des TXT-Eintrags kann je nach Domaintyp variieren.

- Geben Sie für Subdomains einen Unterstrich (_) vor dem alternativen Domainnamen ein. Im Folgenden wird ein Beispiel für einen TXT-Eintrag gezeigt.

  ```
  _www.example.com TXT d111111abcdef8.cloudfront.net
  ```

- Geben Sie für eine Apex- (oder Root-Domain) einen Unterstrich und einen Punkt (`_.`) vor dem Domainnamen ein. Im Folgenden wird ein Beispiel für einen TXT-Eintrag gezeigt.

  ```
  _.example.com TXT d111111abcdef8.cloudfront.net
  ```

Distribution tenant

Um den Mandanten für die Zielverteilung einzurichten

1. Fordern Sie ein TLS-Zertifikat an. Dieses Zertifikat enthält den alternativen Domainnamen als Subject oder Subject Alternative Domain (SAN) oder einen Platzhalter (*), der den alternativen Domainnamen abdeckt, den Sie verschieben möchten. Wenn Sie noch keines haben, können Sie eines bei AWS Certificate Manager (ACM) oder einer anderen Zertifizierungsstelle (CA) anfordern und in ACM importieren.

   > **Note**
   >
   > Sie müssen das Zertifikat in der Region USA Ost (Nord-Virginia) (`us-east-1`) anfordern oder importieren.

   Weitere Informationen finden Sie unter Anfordern eines öffentlichen Zertifikats über die Konsole und Importieren eines Zertifikats AWS Certificate Manager im im AWS Certificate Manager Benutzerhandbuch.

2. Wenn Sie den Mandanten für die Zielverteilung noch nicht erstellt haben, erstellen Sie jetzt einen. Ordnen Sie im Rahmen der Erstellung des Verteilungsmandanten das Zertifikat dem Verteilungsmandanten zu. Weitere Informationen finden Sie unter Eine Verteilung erstellen.

   Wenn Sie bereits einen Zielverteilungsmandanten haben, ordnen Sie das Zertifikat dem Verteilungsmandanten zu. Weitere Informationen finden Sie unter Fügen Sie eine Domäne und ein Zertifikat hinzu (Distributionsmandant).

3. Wenn Sie alternative Domainnamen innerhalb derselben Domain verschieben AWS-Konto, überspringen Sie diesen Schritt.

Um einen alternativen Domainnamen von einem AWS-Konto auf einen anderen zu verschieben, müssen Sie in Ihrer DNS-Konfiguration einen TXT-Eintrag erstellen. Dieser Bestätigungsschritt trägt dazu bei, unbefugte Domainübertragungen zu verhindern, und CloudFront verwendet diesen TXT-Eintrag, um zu überprüfen, ob Sie der Eigentümer des alternativen Domainnamens sind.

Erstellen Sie in Ihrer DNS-Konfiguration einen DNS-TXT-Eintrag, der den alternativen Domainnamen dem Zielverteilungsmandanten zuordnet. Das Format des TXT-Eintrags kann je nach Domaintyp variieren.

- Geben Sie für Subdomains einen Unterstrich (_) vor dem alternativen Domainnamen ein. Im Folgenden wird ein Beispiel für einen TXT-Eintrag gezeigt.

  ```
  _www.example.com TXT d111111abcdef8.cloudfront.net
  ```

- Geben Sie für eine Apex- (oder Root-Domain) einen Unterstrich und einen Punkt (_.) vor dem Domainnamen ein. Im Folgenden wird ein Beispiel für einen TXT-Eintrag gezeigt.

  ```
  _.example.com TXT d111111abcdef8.cloudfront.net
  ```

Als Nächstes finden Sie im folgenden Thema den Quell-Standardverteilungs- oder Distributionsmandanten, der bereits mit dem alternativen Domänennamen verknüpft ist.

Suchen Sie den Mandanten für die Standardverteilung oder -verteilung als Quelle

Bevor Sie einen alternativen Domainnamen von einer Distribution (Standard- oder Mandantendistribution) in eine andere verschieben können, müssen Sie die Quelldistribution ermitteln. Dies ist die Ressource, der der alternative Domainname bereits zugeordnet ist. Wenn Sie die AWS-Konto ID sowohl der Quell- als auch der Zielvertriebsressourcen kennen, können Sie festlegen, wie der alternative Domainname verschoben werden soll.

Hinweise

- Es wird empfohlen, den ListDomainConflictsAPI-Vorgang zu verwenden, da er sowohl Standardverteilungen als auch Verteilungsmandanten unterstützt.
- Der ListConflictingAliasesAPI-Vorgang unterstützt nur Standardverteilungen.

Folgen Sie diesen Beispielen, um die Quelldistribution (Standard oder Tenant) zu finden.

list-domain-conflicts

> **Tip**
>
> - Für eine Standardverteilung benötigen Sie die `cloudfront:ListDomainConflicts` Berechtigungen `cloudfront:GetDistribution` und.
> - Für einen Verteilungsmandanten benötigen Sie die `cloudfront:ListDomainConflicts` Berechtigungen `cloudfront:GetDistributionTenant` und.

Wird verwendet, **`list-domain-conflicts`** um den Quell-Standardverteilungs- oder Distributionsmandanten zu finden

1. Verwenden Sie den Befehl `list-domain-conflicts` wie im folgenden Beispiel gezeigt.

   a. Durch *`www.example.com`* den Domainnamen ersetzen.

   b. Geben Sie für die `domain-control-validation-resource` die ID des Ziel-Standardverteilungs- oder Distributionsmandanten an, den Sie zuvor eingerichtet haben. Sie benötigen einen Standardverteilungs- oder -verteilungsmandanten, dem ein Zertifikat zugeordnet ist, das die angegebene Domäne abdeckt.

   c. Führen Sie diesen Befehl mit den Anmeldeinformationen aus, die sich auf dem AWS-Konto Zielmandanten für die Standardverteilung oder -verteilung befinden.

   Anforderung

   In diesem Beispiel wird ein Verteilungsmandant angegeben.

   ```
   aws cloudfront list-domain-conflicts \
   --domain www.example.com \
   --domain-control-validation-resource
    "DistributionTenantId=dt_2x9GhoK0TZRsohWzv1b9It8JABC"
   ```

   Antwort

   Für jeden Domainnamen in der Ausgabe des Befehls wird Folgendes angezeigt:

- Der Ressourcentyp, dem die Domain zugeordnet ist
- Die Ressourcen-ID.
- Die AWS-Konto ID, der die Ressource gehört

Die Ressourcen-ID und die Konto-ID sind teilweise ausgeblendet. Auf diese Weise können Sie den Standardverteilungs- oder Distributionsmandanten identifizieren, der zu Ihrem Konto gehört, und die Informationen von Mandanten schützen, die Ihnen nicht gehören.

```
{
    "DomainConflicts": [
        {
            "Domain": "www.example.com",
            "ResourceType": "distribution-tenant",
            "ResourceId": "***************ohWzv1b9It8JABC",
            "AccountId": "******112233"
        }
    ]
}
```

In der Antwort werden alle Domänennamen aufgeführt, die mit dem von Ihnen angegebenen Namen in Konflikt stehen oder sich überschneiden.

Beispiel

- Wenn Sie angeben *tenant1.example.com*, enthält die Antwort tenant1.example.com und den sich überschneidenden alternativen Platzhalter-Domänennamen (*.example.com, falls vorhanden).
- Wenn Sie angeben **.tenant1.example.com*, enthält die Antwort *.tenant1.example.com und alle alternativen Domainnamen, die von diesem Platzhalter abgedeckt werden (z. B. test.tenant1.example.com, dev.tenant1.example.com usw.).

2. Suchen Sie in der Antwort nach dem Quell-Standardverteilungs- oder Distributionsmandanten für den alternativen Domainnamen, den Sie verschieben möchten, und notieren Sie sich die ID. AWS-Konto
3. Vergleichen Sie die Konto-ID des Ausgangsmandanten für die Standardverteilung oder -verteilung mit der Konto-ID, für die Sie im vorherigen Schritt den Zielmandanten für die Standardverteilung oder -verteilung erstellt haben. Anschließend können Sie feststellen, ob

die Quelle und das Ziel identisch sind AWS-Konto. Dies hilft Ihnen zu bestimmen, wie Sie den alternativen Domänennamen verschieben.

Weitere Informationen finden Sie in der AWS Command Line Interface Referenz zu dem list-domain-conflictsBefehl.

list-conflicting-aliases (standard distributions only)

> **Tip**
>
> Sie müssen über die `cloudfront:ListConflictingAliases` Berechtigungen `cloudfront:GetDistribution` und für die Ziel-Standarddistribution verfügen.

Wird verwendet**`list-conflicting-aliases`**, um die Quell-Standarddistribution zu finden

1. Verwenden Sie den Befehl `list-conflicting-aliases` wie im folgenden Beispiel gezeigt.

   a. *`www.example.com`*Ersetzen Sie durch den alternativen Domainnamen und *`EDFDVBD6EXAMPLE`* durch die ID der Ziel-Standarddistribution, die Sie zuvor eingerichtet haben.

   b. Führen Sie diesen Befehl mit den Anmeldeinformationen aus, die AWS-Konto sich in derselben Standardverteilung befinden.

   Anforderung

   In diesem Beispiel wird eine Standardverteilung angegeben.

   ```
   aws cloudfront list-conflicting-aliases \
   --alias www.example.com \
   --distribution-id EDFDVBD6EXAMPLE
   ```

   Antwort

   Für jeden alternativen Domainnamen in der Befehlsausgabe können Sie die ID der Standarddistribution sehen, der er zugeordnet ist, und die AWS-Konto ID, der die Standarddistribution gehört. Die Standarddistribution und das Konto IDs sind teilweise

ausgeblendet, sodass Sie die Standardverteilungen und Konten identifizieren können, die Ihnen gehören, und die Informationen derjenigen, die Sie nicht besitzen, besser schützen können.

```
{
    "ConflictingAliasesList": {
        "MaxItems": 100,
        "Quantity": 1,
        "Items": [
            {
                "Alias": "www.example.com",
                "DistributionId": "*******EXAMPLE",
                "AccountId": "******112233"
            }
        ]
    }
}
```

In der Antwort werden die alternativen Domainnamen aufgeführt, die mit dem von Ihnen angegebenen Domainnamen in Konflikt stehen oder sich überschneiden.

Beispiel

- Wenn Sie angeben*www.example.com*, enthält die Antwort www.example.com und den sich überschneidenden alternativen Platzhalter-Domänennamen (*.example.com), falls vorhanden.
- Wenn Sie angeben**.example.com*, enthält die Antwort *.example.com und alle alternativen Domainnamen, die von diesem Platzhalter abgedeckt werden (z. B. www.example.com, test.example.com, dev.example.com usw.).

2. Suchen Sie die Standardverteilung für den alternativen Domainnamen, den Sie verschieben, und notieren Sie sich die ID. AWS-Konto Vergleichen Sie diese Konto-ID mit der Konto-ID, für die Sie im vorherigen Schritt die Ziel-Standardverteilung erstellt haben. Anschließend können Sie feststellen, ob sich diese beiden Standardverteilungen in derselben befinden AWS-Konto und wie der alternative Domainname verschoben werden soll.

   Weitere Informationen finden Sie in der AWS Command Line Interface Referenz zu dem list-conflicting-aliasesBefehl.

Lesen Sie als Nächstes das folgende Thema, um den alternativen Domainnamen zu verschieben.

Einen alternativen Domänennamen verschieben

Wählen Sie je nach Situation eine der folgenden Möglichkeiten, den alternativen Domänennamen zu verschieben:

Die Quell- und Zielverteilungen (Standard oder Tenant) befinden sich in derselben AWS-Konto

Verwenden Sie den update-domain-association Befehl in AWS Command Line Interface (AWS CLI), um den alternativen Domainnamen zu verschieben.

Dieser Befehl funktioniert für alle Verschiebungen desselben Kontos, auch wenn es sich bei dem alternativen Domainnamen um eine Apex-Domain handelt (auch als Stammdomain bezeichnet, wie example.com).

Die Quell- und Zieldistribution (Standard- oder Mandantendistribution) sind unterschiedlich AWS-Konten

Wenn Sie Zugriff auf den Quellmandanten für die Standardverteilung oder -verteilung haben, der alternative Domänenname keine Apex-Domäne ist und Sie nicht bereits einen Platzhalter verwenden, der sich mit diesem alternativen Domänennamen überschneidet, verwenden Sie einen Platzhalter, um den alternativen Domänennamen zu verschieben. Weitere Informationen finden Sie unter the section called “Einen alternativen Domänennamen mit einem Platzhalter verschieben”.

Wenn Sie keinen Zugriff auf den Mandanten haben AWS-Konto , der den ursprünglichen Standardverteilungs- oder Distributionsmandanten hat, können Sie versuchen, den alternativen Domänennamen mithilfe des update-domain-association Befehls zu verschieben. Der Quell-Standardverteilungs- oder Distributionsmandant muss deaktiviert werden, bevor Sie den alternativen Domänennamen verschieben können. Weitere Informationen finden Sie unter the section called “Wenden Sie sich AWS -Support an, um einen alternativen Domainnamen zu verschieben”.

> **Note**
>
> Sie können den associate-alias Befehl verwenden, aber dieser Befehl unterstützt nur Standardverteilungen. Weitere Informationen finden Sie AssociateAliasin der Amazon CloudFront API-Referenz.

update-domain-association (standard distributions and distribution tenants)

Wird verwendet**`update-domain-association`**, um einen alternativen Domainnamen zu verschieben

1. Verwenden Sie den Befehl `update-domain-association`, wie im folgenden Beispiel gezeigt.

   a. *`example.com`*Ersetzen Sie ihn durch den alternativen Domänennamen und geben Sie die ID des Zielmandanten für die Standardverteilung oder Verteilung an.

   b. Führen Sie diesen Befehl mit den Anmeldeinformationen aus, die denen des Zielmandanten für die Standardverteilung oder -verteilung entsprechen AWS-Konto .

> Beachten Sie die folgenden Einschränkungen
>
> - Zusätzlich zur `cloudfront:UpdateDomainAssociation` Berechtigung müssen Sie über die `cloudfront:UpdateDistribution` Berechtigung verfügen, eine Standarddistribution zu aktualisieren. Um einen Distributionsmandanten zu aktualisieren, benötigen Sie die `cloudfront:UpdateDistributionTenant` entsprechende Berechtigung.
> - Wenn sich die Quell- und Zieldistribution (Standard- oder Mandantendistribution) unterscheiden AWS-Konten, muss die Quell- und Zielverteilung deaktiviert werden, bevor Sie die Domain verschieben können.
> - Die Zielverteilung muss wie in the section called “Richten Sie den Ziel-Standardverteilungs- oder Distributionsmandanten ein” beschrieben eingerichtet werden.

Anforderung

```
aws cloudfront update-domain-association \
  --domain "www.example.com" \
  --target-resource DistributionTenantId=dt_9Fd3xTZq7Hl2KABC \
  --if-match E3UN6WX5ABC123
```

Antwort

```
{
    "ETag": "E7Xp1Y3N9DABC",
    "Domain": "www.example.com",
    "ResourceId": "dt_9Fd3xTZq7Hl2KABC"
}
```

Dieser Befehl entfernt den alternativen Domänennamen aus dem Quell-Standardverteilungs- oder Distributionsmandanten und fügt ihn dem Ziel-Standardverteilungs- oder -verteilungsmandanten hinzu.

2. Nachdem die Zielverteilung vollständig bereitgestellt ist, aktualisieren Sie Ihre DNS-Konfiguration so, dass Ihr Domainname auf den CloudFront Routing-Endpunkt verweist. Ihr DNS-Eintrag würde beispielsweise Ihren alternativen Domainnamen (`www.example.com`) auf den CloudFront angegebenen Domainnamen d111111abcdef8.cloudfront.net verweisen. Wenn das Ziel ein Verteilungsmandant ist, geben Sie den Endpunkt der Verbindungsgruppe an. Weitere Informationen finden Sie unter Verweisen Sie Domains auf CloudFront.

associate-alias (standard distributions only)

Wird verwendet**`associate-alias`**, um einen alternativen Domänennamen zu verschieben

1. Verwenden Sie den Befehl `associate-alias`, wie im folgenden Beispiel gezeigt.

   a. *`www.example.com`*Ersetzen Sie ihn durch den alternativen Domainnamen und *`EDFDVBD6EXAMPLE`* durch die Standard-Zielvertriebs-ID.

   b. Führen Sie diesen Befehl mit Anmeldeinformationen aus, die mit denen der Ziel-Standarddistribution AWS-Konto identisch sind.

   > Beachten Sie die folgenden Einschränkungen
   >
   > - Sie müssen über `cloudfront:AssociateAlias` `cloudfront:UpdateDistribution` Berechtigungen für die Ziel-Standarddistribution verfügen.
   > - Wenn sich die Quell- und Ziel-Standarddistribution in derselben befinden AWS-Konto, benötigen Sie die entsprechenden `cloudfront:UpdateDistribution` Berechtigungen für die Quell-Standarddistribution.

- Wenn sich die Quell-Standardverteilung und die Ziel-Standardverteilung unterscheiden AWS-Konten, müssen Sie zuerst die Quell-Standardverteilung deaktivieren.
- Die Ziel-Standardverteilung muss wie unter beschrieben eingerichtet werdenthe section called “Richten Sie den Ziel-Standardverteilungs- oder Distributionsmandanten ein”.

Anforderung

```
aws cloudfront associate-alias \
--alias www.example.com \
--target-distribution-id EDFDVBD6EXAMPLE
```

Dieser Befehl entfernt den alternativen Domänennamen aus der Quell-Standardverteilung und verschiebt ihn in die Ziel-Standardverteilung.

2. Nachdem die Zielstandardverteilung vollständig bereitgestellt wurde, aktualisieren Sie Ihre DNS-Konfiguration so, dass der DNS-Eintrag des alternativen Domainnamens auf den Distributionsdomänennamen der Zielstandardverteilung verweist. Ihr DNS-Eintrag würde beispielsweise Ihren alternativen Domainnamen (`www.example.com`) auf den CloudFront angegebenen Domainnamen d111111abcdef8.cloudfront.net verweisen.

Weitere Informationen finden Sie unter dem Befehl in der Befehlsreferenz. associate-aliasAWS CLI

Einen alternativen Domänennamen mit einem Platzhalter verschieben

Wenn sich die Quelldistribution in einer anderen Distribution AWS-Konto als die Zieldistribution befindet und die Quellverteilung aktiviert ist, können Sie einen Platzhalter verwenden, um den alternativen Domainnamen zu verschieben.

> **Note**
>
> Sie können keine Platzhalter verwenden, um eine Apex-Domäne zu verschieben (z. B. example.com). Um eine Apex-Domain zu verschieben, wenn sich Quell- und Zielverteilung unterscheiden AWS-Konten, wenden Sie sich an. Support Weitere Informationen finden

Sie unter the section called “Wenden Sie sich AWS -Support an, um einen alternativen Domainnamen zu verschieben”.

Verwenden eines Platzhalters, um einen alternativen Domänennamen zu verschieben

> **Note**
>
> Dieser Prozess beinhaltet mehrere Aktualisierungen Ihrer Verteilungen. Warten Sie, bis jede Verteilung die letzte Änderung vollständig bereitgestellt hat, bevor Sie mit dem nächsten Schritt fortfahren.

1. Aktualisieren Sie die Zielverteilung, um einen alternativen Platzhalterdomänennamen hinzuzufügen, der den alternativen Domänennamen abdeckt, den Sie verschieben. Wenn der alternative Domänenname, den Sie verschieben, beispielsweise www.example.com lautet, fügen Sie der Zielverteilung den alternativen Domänennamen *.example.com hinzu. Dazu muss das SSL/TLS Zertifikat auf der Zieldistribution den Platzhalter-Domänennamen enthalten. Weitere Informationen finden Sie unter the section called “Eine Verteilung aktualisieren”.
2. Aktualisieren Sie die DNS-Einstellungen für den alternativen Domänennamen, sodass sie auf den Domänennamen der Zielverteilung verweisen. Wenn der alternative Domänenname, den Sie verschieben, beispielsweise www.example.com lautet, aktualisieren Sie den DNS-Eintrag für www.example.com, um den Datenverkehr an den Domänennamen der Zielverteilung weiterzuleiten (z. B. d111111abcdef8.cloudfront.net). .

   > **Note**
   >
   > Auch nachdem Sie die DNS-Einstellungen aktualisiert haben, wird der alternative Domänenname weiterhin von der Quellverteilung bedient, da dort der alternative Domänenname derzeit konfiguriert ist.

3. Aktualisieren Sie die Quellverteilung, um den alternativen Domänennamen zu entfernen. Weitere Informationen finden Sie unter Eine Verteilung aktualisieren.
4. Aktualisieren Sie die Zielverteilung, um den alternativen Domänennamen hinzuzufügen. Weitere Informationen finden Sie unter Eine Verteilung aktualisieren.
5. Verwenden Sie dig (oder ein ähnliches DNS-Abfragetool), um zu überprüfen, ob der DNS-Eintrag für den alternativen Domänennamen in den Domänennamen der Zielverteilung aufgelöst wird.

6. (Optional) Aktualisieren Sie die Zielverteilung, um den alternativen Platzhalterdomänennamen zu entfernen.

Wenden Sie sich AWS -Support an, um einen alternativen Domainnamen zu verschieben

Wenn sich Quell- und Zieldistribution unterscheiden AWS-Konten und Sie keinen Zugriff auf die Verteilung der Quelldistribution haben AWS-Konto oder die Quelldistribution nicht deaktivieren können, können Sie sich an uns wenden, Support um den alternativen Domainnamen zu verschieben.

Wenden Sie sich an uns Support , um einen alternativen Domainnamen zu verschieben

1. Richten Sie eine Zielverteilung ein, einschließlich des DNS-TXT-Eintrags, der auf die Zielverteilung verweist. Weitere Informationen finden Sie unter Richten Sie den Ziel-Standardverteilungs- oder Distributionsmandanten ein.
2. Wenden Sie sich Support an den Anbieter, um zu überprüfen, ob Ihnen die Domain gehört, und die Domain für Sie auf die neue CloudFront Distribution umzustellen.
3. Nachdem die Zielverteilung vollständig bereitgestellt wurde, aktualisieren Sie Ihre DNS-Konfiguration, um den DNS-Eintrag des alternativen Domänennamens auf den Verteilungsdomänennamen der Zielverteilung zu verweisen.

Entfernen Sie einen alternativen Domainnamen

Wenn Sie den Datenverkehr für eine Domain oder Subdomain nicht mehr an eine CloudFront Distribution weiterleiten möchten, folgen Sie den Schritten in diesem Abschnitt, um sowohl die DNS-Konfiguration als auch die CloudFront Verteilung zu aktualisieren.

Es ist wichtig, dass Sie die alternativen Domänennamen aus der Verteilung entfernen und Ihre DNS-Konfiguration aktualisieren. Auf diese Weise können Sie später Probleme vermeiden, wenn Sie den Domainnamen einer anderen CloudFront Distribution zuordnen möchten. Wenn ein alternativer Domänenname bereits einer Verteilung zugeordnet ist, kann er nicht mit einer anderen Verteilung eingerichtet werden.

> **Note**
>
> Wenn Sie den alternativen Domänennamen aus dieser Verteilung entfernen möchten, damit Sie ihn einer anderen Verteilung hinzufügen können, führen Sie die in Verschieben Sie

einen alternativen Domainnamen genannten Schritte aus. Wenn Sie stattdessen die hier beschriebenen Schritte ausführen (um eine Domain zu entfernen) und die Domain dann einer anderen Distribution hinzufügen, wird es einen bestimmten Zeitraum geben, in dem die Domain nicht mit der neuen Distribution verknüpft CloudFront wird, weil sie sich auf die Aktualisierungen an den Edge-Standorten ausbreitet.

So entfernen Sie einen alternativen Domänennamen aus einer Verteilung

1. Leiten Sie zunächst den Internet-Traffic für Ihre Domain an eine andere Ressource weiter, die nicht zu Ihrer CloudFront Distribution gehört, z. B. einen Elastic Load Balancing Load Balancer. Oder Sie können den DNS-Eintrag löschen, zu dem der Datenverkehr weitergeleitet CloudFront wird.

   Führen Sie einen der folgenden Schritte aus, je nach dem DNS-Service für Ihre Domäne:

   - Wenn Sie Route 53 verwenden, aktualisieren oder löschen Sie Aliasdatensätze oder CNAME-Datensätze. Weitere Informationen finden Sie unter Bearbeiten von Datensätzen oder Löschen von Datensätzen.
   - Wenn Sie einen anderen DNS-Serviceanbieter nutzen, verwenden Sie die Methode des DNS-Serviceanbieters, um den CNAME-Datensatz, der den Datenverkehr zu CloudFront weiterleitet, zu aktualisieren oder zu löschen. Weitere Informationen finden Sie in der Dokumentation Ihres DNS-Serviceanbieters.
2. Nachdem Sie die DNS-Datensätze Ihrer Domäne aktualisiert haben, warten Sie, bis die Änderungen weitergegeben wurden und DNS-Resolver den Datenverkehr an die neue Ressource weiterleiten. Sie können feststellen, ob der Vorgang abgeschlossen ist, indem Sie einige Testlinks erstellen, die Ihre Domäne in der URL verwenden.
3. Melden Sie sich bei der AWS Management Console an, öffnen Sie die CloudFront Konsole unter und aktualisieren Sie Ihre CloudFront Distributionhttps://console.aws.amazon.com/cloudfront/v4/home, um den Domainnamen zu entfernen. Gehen Sie dazu wie folgt vor:

   a. Wählen Sie die ID für die Verteilung aus, die Sie aktualisieren möchten.

   b. Wählen Sie auf der Registerkarte General die Option Edit aus.

   c. Entfernen Sie unter Alternative Domainnamen (CNAMEs) den alternativen Domainnamen (oder die Domainnamen), den Sie nicht mehr für Ihren Vertrieb verwenden möchten.

   d. Wählen Sie Yes, Edit aus.

Verwenden Sie Platzhalter in alternativen Domainnamen

Wenn Sie alternative Domänennamen hinzufügen, können Sie am Anfang eines Domänennamens das Platzhalterzeichen * verwenden, statt Unterdomänen einzeln hinzuzufügen. Beispiel: Mit dem alternativen Domainnamen *.example.com können Sie in Ihrem Domainnamen jeden beliebigen Domainnamen verwenden, der auf example.com endet, z. B. www.example.com, product-name.example.com URLs, marketing.product-name.example.com usw. Der Pfad eines Objekts ist derselbe, unabhängig von dem Domänennamen, z. B.:

- www.Beispiel. com/images/image.jpg
- Produktname.Beispiel. com/images/image.jpg
- marketing.produktname.beispiel. com/images/image.jpg

Beachten Sie diese Anforderungen bei alternativen Domänennamen, die Platzhalterzeichen enthalten.

- Am Anfang des alternativen Domänennamens muss ein Sternchen und ein Punkt stehen (*.).
- Sie können keine Platzhalter verwenden, um einen Teil eines Namens einer Subdomäne zu ersetzen, beispielsweise *domain.example.com.
- Sie können eine Subdomäne nicht in der Mitte eines Domänennamens ersetzen, beispielsweis: subdomain.*.example.com.
- Alle alternativen Domänennamen, einschließlich alternativer Domänennamen, die Platzhalter verwenden, müssen durch den Subject Alternative Name (SAN) auf dem Zertifikat abgedeckt sein.

Ein alternativer Domänenname mit einem Platzhalter wie *.example.com kann einen weiteren alternativen Domänennamen wie z. B. example.com enthalten.

WebSockets Mit CloudFront Distributionen verwenden

Amazon CloudFront unterstützt die Verwendung eines TCP-basierten Protokolls WebSocket, das nützlich ist, wenn Sie langlebige bidirektionale Verbindungen zwischen Clients und Servern benötigen. Eine persistente Verbindung ist bei Echtzeit-Anwendungen oft erforderlich. Zu den Szenarien, die Sie verwenden könnten, WebSockets gehören Social-Chat-Plattformen, Arbeitsbereiche für die Online-Zusammenarbeit, Spiele für mehrere Spieler und Dienste, die Datenfeeds in Echtzeit bereitstellen, wie z. B. Finanzhandelsplattformen. Bei der WebSocket Vollduplex-Kommunikation können Daten über eine Verbindung in beide Richtungen fließen.

WebSocket Die Funktionalität wird automatisch aktiviert, sodass sie mit jeder Distribution funktioniert. Um sie zu verwendenWebSockets, konfigurieren Sie im Cache-Verhalten, das Ihrer Distribution zugewiesen ist, eine der folgenden Optionen:

- Leiten Sie alle Header von Zuschaueranfragen an Ihren Ursprung weiter. Sie können die Richtlinie für AllViewer verwaltete Anfragen mit Herkunft verwenden.
- Leiten Sie die Header `Sec-WebSocket-Key` und die `Sec-WebSocket-Version` Anforderungsheader in Ihrer Richtlinie für ursprüngliche Anfragen ausdrücklich weiter.

Wie funktioniert das WebSocket Protokoll

Das WebSocket Protokoll ist ein unabhängiges, TCP-basiertes Protokoll, mit dem Sie einen Teil des Mehraufwands — und die potenziell erhöhte Latenz — von HTTP vermeiden können.

Um eine WebSocket Verbindung herzustellen, sendet der Client eine reguläre HTTP-Anfrage, die die Upgrade-Semantik von HTTP verwendet, um das Protokoll zu ändern. Der Server kann dann den Handshake abschließen. Die WebSocket Verbindung bleibt geöffnet und entweder der Client oder der Server können Datenframes aneinander senden, ohne jedes Mal neue Verbindungen herstellen zu müssen.

Standardmäßig verwendet das WebSocket Protokoll Port 80 für reguläre WebSocket Verbindungen und Port 443 für WebSocket Verbindungen über TLS. Die Optionen, die Sie für Ihre CloudFront Viewer-Protokollrichtlinien und wählen, Protokoll (nur benutzerdefinierte Ursprünge) gelten für WebSocket Verbindungen und auch für HTTP-Verkehr.

WebSocket-Voraussetzungen

WebSocket Anfragen müssen RFC 6455 in den folgenden Standardformaten entsprechen.

Example Beispiel für eine Kundenanfrage

```
GET /chat HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: dGhlIHNhbXBsZSBub25jZQ==
Origin: https://example.com
Sec-WebSocket-Protocol: chat, superchat
Sec-WebSocket-Version: 13
```

Example Beispiel für eine Serverantwort

```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: s3pPLMBiTxaQ9kYGzzhZRbK+xOo=
Sec-WebSocket-Protocol: chat
```

Wenn die WebSocket Verbindung durch den Client oder Server oder durch eine Netzwerkunterbrechung unterbrochen wird, wird erwartet, dass die Client-Anwendungen die Verbindung mit dem Server wieder aufnehmen.

Empfohlene Header WebSocket

Um unerwartete Probleme im Zusammenhang mit der Komprimierung bei der Verwendung zu vermeiden, empfehlen wir WebSockets, die folgenden Header in eine Origin-Request-Richtlinie aufzunehmen:

- `Sec-WebSocket-Key`
- `Sec-WebSocket-Version`
- `Sec-WebSocket-Protocol`
- `Sec-WebSocket-Accept`
- `Sec-WebSocket-Extensions`

Fordere Anycast static an, um es für die Zulassungsliste zu verwenden IPs

Sie können Static Anycast IPs von anfordern, um es mit Ihren CloudFront Distributionen zu verwenden. Die statischen IP-Listen von Anycast enthalten IPv4 IP-Adressen, die nur für Sie bestimmt sind AWS-Konto und über verschiedene geografische Regionen verteilt sind.

Sie können 21 statische Anycast-IP-Adressen für die Aufnahme einer Zulassungsliste bei Netzwerkanbietern anfordern, sodass Sie auf Datengebühren für Zuschauer verzichten können, die auf Ihre Anwendung zugreifen. Alternativ können Sie diese statischen Daten IPs innerhalb ausgehender Sicherheitsfirewalls verwenden, um den Datenaustausch mit zugelassenen Anwendungen zu kontrollieren. Statische Anycast-IP-Listen können mit einer oder mehreren Distributionen verwendet werden.

Wenn Sie das Routing von Apex-Domänen (wie example.com) direkt an Ihre CloudFront Distributionen aktivieren möchten, können Sie für diesen Anwendungsfall 3 statische Anycast-IP-Adressen anfordern. Fügen Sie dann A-Einträge zu Ihrem DNS hinzu, auf die die Apex-Domain verweisen soll. CloudFront

Anycast Static IPs funktioniert mit Server Name Indication (SNI). Weitere Informationen finden Sie unter Verwenden Sie SNI, um HTTPS-Anfragen zu bearbeiten (funktioniert für die meisten Kunden).

Voraussetzungen

Um statische Anycast-IP-Listen mit Ihrer CloudFront Distribution zu verwenden, müssen Sie:

- Schalten Sie die IPv6 Option für die Distribution aus, die Sie mit Ihrer statischen Anycast-IP-Liste verwenden möchten.
- Wählen Sie Alle Edge-Standorte verwenden für die Preisklasse für die Verteilung aus. Weitere Informationen zu Preisen finden Sie unter CloudFront Preise.

Fordern Sie eine statische Anycast-IP-Liste an

Fordern Sie eine statische Anycast-IP-Liste an, die Sie mit Ihrer CloudFront Distribution verwenden können.

Um eine statische Anycast-IP-Liste anzufordern

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im linken Navigationsbereich Statisch aus IPs.
3. Wählen Sie unter Anfrage den Link, um den technischen CloudFront Support zu kontaktieren.
4. Geben Sie Ihre Workload-Informationen an (Anforderungs-Bytes pro Sekunde und Anfragen pro Sekunde).
5. CloudFront Support Engineering prüft Ihre Anfrage. Der Überprüfungsprozess kann bis zu zwei Tage dauern.

Nachdem Ihre Anfrage genehmigt wurde, können Sie eine statische Anycast-IP-Liste erstellen und sie einer oder mehreren Distributionen zuordnen.

Erstellen Sie eine statische Anycast-IP-Liste

Bevor Sie beginnen, fordern Sie eine statische Anycast-IP-Liste an, wie im vorherigen Abschnitt beschrieben.

Um eine statische Anycast-IP-Liste zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im linken Navigationsbereich Statisch aus IPs.
3. Wählen Sie „Anycast-IP-Liste erstellen“.
4. Geben Sie unter Name einen Namen ein.
5. Wählen Sie für statische IP-Anwendungsfälle den entsprechenden Anwendungsfall aus.
6. Lesen Sie sich die Servicebedingungen und Preise durch und wählen Sie Senden aus.

Nachdem Ihre statische IP-Liste erstellt wurde, können Sie die zugewiesenen IP-Adressen auf der Detailseite Ihrer statischen IP-Liste einsehen. Sie können der statischen IP-Liste auch Distributionen zuordnen.

Ordnen Sie eine statische Anycast-IP-Liste einer vorhandenen Distribution zu

Bevor Sie beginnen, fordern Sie eine statische Anycast-IP-Liste an und erstellen Sie sie, wie in den vorherigen Abschnitten beschrieben. Vergewissern Sie sich außerdem, dass Sie die Option IPv6 für Ihre Distribution deaktiviert und für die Preisklasse Alle Edge-Standorte verwenden (beste Leistung) ausgewählt haben.

Um eine statische Anycast-IP-Liste mit einer vorhandenen Distribution zu verknüpfen

- Führen Sie eine der folgenden Aktionen aus:
  - Ordnen Sie die statische IP-Liste auf der Detailseite der statischen IP-Liste zu:
    1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
    2. Wählen Sie IPs im linken Navigationsbereich Statisch aus.
    3. Wählen Sie den Namen Ihrer statischen IP-Liste.

4. Wählen Sie Associate Distributions aus.
5. Wählen Sie eine oder mehrere Verteilungen aus und wählen Sie Verteilungen zuordnen aus.

- Ordnen Sie die statische IP-Liste auf der Verteilungsdetailseite zu:
  1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
  2. Wählen Sie im linken Navigationsbereich Distributionen aus.
  3. Wählen Sie den Namen Ihrer Distribution.
  4. Wählen Sie auf der Registerkarte Allgemein unter Einstellungen die Option Bearbeiten aus.
  5. Wählen Sie für die Anycast-IP-Liste die statische Anycast-IP-Liste aus, die mit dieser Distribution verwendet werden soll.
  6. Wählen Sie Änderungen speichern aus.

Ordnen Sie einer neuen Distribution eine statische Anycast-IP-Liste zu

Bevor Sie beginnen, fordern Sie eine statische Anycast-IP-Liste an und erstellen Sie sie, wie in den vorherigen Abschnitten beschrieben.

Um eine statische Anycast-IP-Liste einer neuen Distribution zuzuordnen

- Legen Sie eine neue -Verteilung an. Weitere Informationen finden Sie unter Erstellen Sie eine CloudFront Distribution in der Konsole. Für Einstellungen müssen Sie die folgenden Auswahlen treffen, um Ihre statische Anycast-IP-Liste verwenden zu können:
  - Wählen Sie für die Anycast-IP-Liste Ihre statische Anycast-IP-Liste aus der Drop-down-Liste aus.
  - Wählen Sie für Preisklasse die Option Alle Edge-Standorte verwenden (beste Leistung) aus.
  - Wählen Sie für IPv6„Aus“.

Beenden Sie die Erstellung Ihrer Distribution. Sie können je nach Bedarf alle anderen Einstellungen und Konfigurationen wählen, die für statische IP-Listen von Anycast nicht erforderlich sind.

Weitere Informationen zu Kontingenten im Zusammenhang mit statischen Anycast-IP-Listen finden Sie unter CloudFront Amazon-Endpunkte und Kontingente in der. Allgemeine AWS-Referenz

gRPC mit CloudFront Distributionen verwenden

Amazon CloudFront unterstützt gRPC, ein Open-Source-RPC-Framework (Remote Procedure Call), das auf HTTP/2 basiert. gRPC bietet bidirektionales Streaming und ein binäres Protokoll, das Nutzlasten zwischenspeichert, wodurch es sich für Anwendungen eignet, die eine Kommunikation mit geringer Latenz erfordern.

CloudFront empfängt Ihre gRPC-Anfragen und leitet sie direkt an Ihre Ursprünge weiter. Sie können vier Arten von gRPC-Diensten als Proxy verwenden CloudFront :

- Unäres RPC
- Server-Streaming-RPC
- Client-Streaming-RPC
- Bidirektionales Streaming-RPC

So funktioniert gRPC in CloudFront

Um gRPC zu konfigurieren CloudFront, legen Sie einen Ursprung, der einen gRPC-Dienst bereitstellt, als Ursprung Ihrer Distribution fest. Sie können Origins verwenden, die sowohl Nicht-gRPC- als auch gRPC-Dienste bereitstellen. CloudFront bestimmt anhand des Headers, ob es sich bei der eingehenden Anfrage um eine gRPC-Anfrage oder eine HTTP/HTTPS-Anfrage handelt. `Content-Type` Wenn der `Content-Type` Header einer Anfrage den Wert von hat`application/grpc`, wird die Anfrage als gRPC-Anfrage betrachtet und CloudFront leitet die Anfrage an Ihren Ursprung weiter.

> **Note**
>
> Damit eine Distribution gRPC-Anfragen verarbeiten kann, schließen Sie HTTP/2 als eine der unterstützten HTTP-Versionen ein und lassen HTTP-Methoden zu, einschließlich. `POST` Ihr gRPC-Ursprungsendpunkt muss für die Unterstützung von HTTPS konfiguriert sein, da CloudFront nur sichere (HTTPS-basierte) gRPC-Verbindungen unterstützt werden. gRPC unterstützt nur HTTPS. end-to-end Wenn Sie einen benutzerdefinierten Ursprung verwenden, stellen Sie sicher, dass Ihre Protokolleinstellungen HTTPS unterstützen.

Gehen Sie wie folgt vor, um die gRPC-Unterstützung für Ihre Distribution zu aktivieren:

1. Aktualisieren Sie das Cache-Verhalten Ihrer Distribution, sodass HTTP-Methoden, einschließlich der `POST` Methode, zulässig sind.
2. Nachdem Sie die `POST` Methode ausgewählt haben, aktivieren Sie das angezeigte gRPC-Kontrollkästchen.
3. Geben Sie HTTP/2 als eine der unterstützten HTTP-Versionen an.

Weitere Informationen finden Sie unter den folgenden Themen:

- gRPC-Anfragen über HTTP/2 zulassen
- GrpcConfigin der Amazon CloudFront API-Referenz

Da gRPC nur für API-Verkehr verwendet wird, der nicht zwischengespeichert werden kann, wirken sich Ihre Cache-Konfigurationen nicht auf gRPC-Anfragen aus. Sie können eine Origin-Anforderungsrichtlinie verwenden, um benutzerdefinierte Header zu den gRPC-Anfragen hinzuzufügen, die an Ihren gRPC-Ursprung gesendet werden. Sie können AWS WAF with verwenden CloudFront , um den Zugriff auf Ihre gRPC-Distribution zu verwalten, Bots zu kontrollieren und Ihre gRPC-Anwendungen vor Web-Exploits zu schützen. CloudFront gRPC unterstützt CloudFront Funktionen.

Zusätzlich zum HTTPS-Status erhalten Sie zusammen mit Ihrer gRPC-Antwort den gRPC-Status. Eine Liste möglicher Werte für grpc-status finden Sie unter Statuscodes und ihre Verwendung in gRPC.

Hinweise

gRPC unterstützt die folgenden CloudFront Funktionen nicht:

- Kontinuierlicher Einsatz
- Benutzerdefinierte Fehlerantworten
- Origin-Failover wird mit gRPC nicht unterstützt, da gRPC die Methode verwendet. `POST` CloudFront führt nur dann einen Failover zum sekundären Ursprung durch, wenn die HTTP-Methode der Viewer-Anfrage`GET`,, `HEAD` oder ist. `OPTIONS`
- CloudFront leitet gRPC-Anfragen direkt an den Ursprung weiter und umgeht den Regional Edge Cache (REC). Da gRPC das REC umgeht, unterstützt gRPC weder Lambda @Edge noch Origin Shield.

- gRPC unterstützt keine Regeln für die Inspektion von AWS WAF Anforderungsstellen. Wenn Sie diese Regeln in der Web-ACL für eine Distribution aktiviert haben, ignoriert jede Anfrage, die gRPC verwendet, die Regeln zur Überprüfung des Anforderungstexts. Alle anderen AWS WAF Regeln gelten weiterhin. Weitere Informationen finden Sie unter AWS WAF Für Distributionen aktivieren.

Caching und Verfügbarkeit

Sie können CloudFront es verwenden, um die Anzahl der Anfragen zu reduzieren, auf die Ihr Ursprungsserver direkt antworten muss. Beim CloudFront Caching werden mehr Objekte von CloudFront Edge-Standorten aus bedient, die sich näher an Ihren Benutzern befinden. Dies reduziert die Belastung Ihres Ursprungs-Servers sowie die Latenz.

Je mehr Anfragen von Edge-Caches aus bedient werden CloudFront können, desto weniger Viewer-Anfragen CloudFront müssen an Ihren Ursprung weitergeleitet werden, um die neueste Version oder eine eindeutige Version eines Objekts zu erhalten. Um CloudFront zu optimieren und so wenig Anfragen wie möglich an deinen Absender zu stellen, solltest du die Verwendung eines CloudFront Origin Shield in Betracht ziehen. Weitere Informationen finden Sie unter Verwenden Sie Amazon CloudFront Origin Shield.

Der Anteil der Anfragen, die direkt aus dem CloudFront Cache bedient werden, im Vergleich zu allen Anfragen wird als Cache-Trefferquote bezeichnet. Sie können den Prozentsatz der Viewer-Anfragen, bei denen es sich um Treffer, Fehlschläge und Fehler handelt, in der CloudFront Konsole einsehen. Weitere Informationen finden Sie unter CloudFront Cache-Statistikberichte anzeigen.

Die Cache-Trefferrate wird von einer Reihe von Faktoren beeinflusst. Sie können Ihre CloudFront Verteilungskonfiguration anpassen, um die Cache-Trefferquote zu verbessern, indem Sie die Anweisungen unter befolgenErhöhen Sie den Anteil der Anfragen, die direkt aus den CloudFront Caches bedient werden (Cache-Trefferquote).

Weitere Informationen zum Hinzufügen und Entfernen von Inhalten, die Sie bereitstellen CloudFront möchten, finden Sie unterInhalte hinzufügen, entfernen oder ersetzen, die CloudFront verbreitet werden.

Themen

Erhöhen Sie den Anteil der Anfragen, die direkt aus den CloudFront Caches bedient werden (Cache-Trefferquote)

Sie können die Leistung verbessern, indem Sie den Anteil Ihrer Zuschaueranfragen erhöhen, die direkt aus dem CloudFront Cache bedient werden, anstatt Inhalte an Ihre Originalserver zu senden. Dies ist als Verbesserung der Cache-Trefferquote bekannt.

In den folgenden Abschnitten wird erläutert, wie Sie die Cache-Trefferrate erhöhen.

Themen

Geben Sie an, wie lange Ihre CloudFront Objekte zwischengespeichert werden

Um Ihre Cache-Zugriffsrate zu erhöhen, können Sie Ihren Ursprung so konfigurieren, dass eine Cache-Control-Max-Age-Richtlinie zu Ihren Objekten hinzugefügt wird, und den längsten Gebrauchswert für `max-age` festlegen. Je kürzer die Cache-Dauer, desto häufiger werden Anfragen CloudFront an Ihren Ursprung gesendet, um festzustellen, ob sich ein Objekt geändert hat, und um die neueste Version abzurufen. Sie können die `max-age` mit den `stale-while-revalidate`- und `stale-if-error`-Anweisungen hinzufügen, um die Cache-Trefferquote unter bestimmten Bedingungen weiter zu verbessern. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Benutze Origin Shield

CloudFront Origin Shield kann dazu beitragen, die Cache-Trefferquote deiner CloudFront Distribution zu verbessern, da es eine zusätzliche Caching-Ebene vor deinem Ursprung bietet. Wenn du Origin Shield verwendest, kommen alle Anfragen CloudFront von allen Caching-Ebenen an deinen

Ursprung von einem einzigen Ort. CloudFront kann jedes Objekt mit einer einzigen Origin-Anfrage von Origin Shield abrufen, und alle anderen Ebenen des CloudFront Caches (Edge-Standorte und regionale Edge-Caches) können das Objekt von Origin Shield abrufen.

Weitere Informationen finden Sie unter Verwenden Sie Amazon CloudFront Origin Shield.

Zwischenspeichern auf der Grundlage von Abfragezeichenfolgeparametern

Wenn Sie so konfigurieren CloudFront , dass der Cache auf der Grundlage von Abfragezeichenfolgenparametern gespeichert wird, können Sie das Caching verbessern, indem Sie wie folgt vorgehen:

- Konfigurieren CloudFront Sie so, dass nur die Parameter der Abfragezeichenfolge weitergeleitet werden, für die Ihr Ursprung eindeutige Objekte zurückgibt.
- Verwenden Sie die gleiche Schreibweise (Groß- oder Kleinschreibung) für alle Instances desselben Parameters. Wenn beispielsweise eine Anfrage enthält `parameter1=A` und eine andere enthält`parameter1=a`, werden separate Anfragen an Ihren Ursprung CloudFront weitergeleitet, wenn eine Anfrage enthält `parameter1=A` und wenn eine Anfrage enthält`parameter1=a`. CloudFront speichert dann die entsprechenden Objekte, die von Ihrem Ursprung zurückgegeben wurden, separat zwischen, auch wenn die Objekte identisch sind. Wenn Sie nur `A` oder `a` verwenden, leitet CloudFront weniger Anfragen an Ihren Ursprung weiter.
- Listen Sie Parameter in der gleichen Reihenfolge auf. Wie bei den Unterschieden in der Schreibweise gilt auch hier, dass wenn eine Anfrage für ein Objekt die Abfragezeichenfolge `parameter1=a&parameter2=b` und eine weitere Anfrage für dasselbe Objekt `parameter2=b&parameter1=a` enthält, leitet CloudFront beide Anfragen an Ihren Ursprung weiter und speichert die entsprechenden Objekte separat zwischen, auch wenn sie identisch sind. Wenn Sie immer dieselbe Reihenfolge für Parameter verwenden, CloudFront werden weniger Anfragen an Ihren Ursprung weitergeleitet.

Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern. Wenn Sie die Abfragezeichenfolgen überprüfen möchten, die CloudFront an Ihren Ursprung weitergeleitet werden, sehen Sie sich die Werte in der `cs-uri-query` Spalte Ihrer CloudFront Protokolldateien an. Weitere Informationen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).

Zwischenspeichern auf der Grundlage von Cookie-Werten

Wenn Sie so konfigurieren CloudFront , dass der Cache auf der Grundlage von Cookie-Werten gespeichert wird, können Sie das Caching verbessern, indem Sie wie folgt vorgehen:

- Konfigurieren Sie CloudFront , dass nur bestimmte Cookies weitergeleitet werden, anstatt alle Cookies weiterzuleiten. Bei den Cookies, die Sie so konfigurieren, dass CloudFront sie an Ihren Ursprung weitergeleitet werden, CloudFront wird jede Kombination aus Cookie-Name und Wert weitergeleitet. Die Objekte, die Ihr Ursprung zurückgibt, werden dann separat gespeichert, auch wenn sie alle identisch sind.

  Nehmen wir zum Beispiel an, dass Zuschauer jeder Anfrage zwei Cookies hinzufügen, dass jedes Cookie drei mögliche Werte hat und dass alle Kombinationen von Cookie-Werten möglich sind. CloudFront leitet für jedes Objekt bis zu neun verschiedene Anfragen an Ihren Ursprung weiter. Wenn Ihr Origin verschiedene Versionen eines Objekts zurückgibt, die nur auf einem der Cookies basieren, leitet CloudFront es mehr Anfragen an Ihren Ursprung weiter als nötig und speichert unnötig mehrere identische Versionen des Objekts im Cache.
- Erstellen Sie separate Cache-Verhaltensweisen für statische und dynamische Inhalte und konfigurieren Sie CloudFront sie so, dass Cookies nur für dynamische Inhalte an Ihren Ursprung weitergeleitet werden.

  Nehmen wir zum Beispiel an, Sie haben nur ein Cache-Verhalten für Ihre Verteilung und verwenden die Verteilung sowohl für dynamische Inhalte wie `.js` Dateien als auch für `.css` Dateien, die sich selten ändern. CloudFront speichert separate Versionen Ihrer `.css` Dateien auf der Grundlage von Cookie-Werten im Cache, sodass jeder CloudFront Edge-Standort für jeden neuen Cookie-Wert oder jede Kombination von Cookie-Werten eine Anfrage an Ihren Ursprung weiterleitet.

  Wenn Sie ein Cache-Verhalten erstellen, für das das Pfadmuster gilt `*.css` und für CloudFront das es nicht auf Cookie-Werten basiert, werden `.css` Dateianfragen nur für die erste Anfrage, die ein Edge-Standort für eine bestimmte Datei erhält, und für die erste Anfrage nach Ablauf einer `.css` `.css` Datei an Ihren Ursprung CloudFront weitergeleitet.
- Erstellen Sie nach Möglichkeit separate Cache-Verhaltensweisen für dynamische Inhalte, wenn Cookie-Werte für jeden Benutzer eindeutig sind (z. B. eine Benutzer-ID), und dynamische Inhalte, die je nach einer geringeren Anzahl eindeutiger Werte variieren.

Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern. Wenn Sie die Cookies überprüfen möchten, die an Ihren CloudFront Ursprung weitergeleitet werden, sehen Sie sich die Werte in der `cs(Cookie)` Spalte Ihrer CloudFront Protokolldateien an. Weitere Informationen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).

Zwischenspeichern auf der Grundlage von Anfrage-Headern

Wenn Sie so konfigurieren CloudFront , dass der Cache auf der Grundlage von Anforderungsheadern zwischengespeichert wird, können Sie das Caching verbessern, indem Sie wie folgt vorgehen:

- Konfigurieren Sie CloudFront die Weiterleitung und das Zwischenspeichern nur auf der Grundlage bestimmter Header, anstatt die Weiterleitung und das Zwischenspeichern auf der Grundlage aller Header. Leitet für die von Ihnen angegebenen Header jede Kombination aus CloudFront Header-Namen und -Wert weiter. Die Objekte, die Ihr Ursprung zurückgibt, werden dann separat gespeichert, auch wenn sie alle identisch sind.

  > **Note**
  >
  > CloudFront leitet die in den folgenden Themen angegebenen Header immer an Ihren Ursprung weiter:
  >
  > - So CloudFront werden Anfragen verarbeitet und an Ihren Amazon S3 S3-Ursprungsserver weitergeleitet > HTTP-Anforderungsheader, die CloudFront entfernt oder aktualisiert werden
  > - Wie CloudFront verarbeitet und leitet Anfragen an Ihren benutzerdefinierten Ursprungsserver weiter > Header und CloudFront Verhalten von HTTP-Anfragen (benutzerdefiniert und Amazon S3 S3-Ursprünge)

  Wenn Sie so konfigurieren CloudFront , dass CloudFront der Cache auf der Grundlage von Anforderungsheadern zwischengespeichert wird, ändern Sie nicht die Header, die weitergeleitet werden, sondern nur, ob Objekte auf der Grundlage der CloudFront Header-Werte zwischengespeichert werden.

- Vermeiden Sie die Zwischenspeicherung auf der Grundlage von Anfrage-Headern mit einer hohen Anzahl von eindeutigen Werten.

  Wenn Sie beispielsweise ein Bild je nach Gerät des Benutzers in unterschiedlichen Größen bereitstellen möchten, sollten Sie die Konfiguration nicht so konfigurieren CloudFront , dass

der `User-Agent` Header zwischengespeichert wird, der eine enorme Anzahl möglicher Werte enthält. Konfigurieren Sie stattdessen, dass der Cache CloudFront auf der Grundlage der Header`CloudFront-Is-Desktop-Viewer`,, `CloudFront-Is-Mobile-Viewer` und des CloudFront Gerätetyps zwischengespeichert wird. `CloudFront-Is-SmartTV-Viewer` `CloudFront-Is-Tablet-Viewer` Wenn Sie zudem dieselbe Version des Images für Tablets und Desktops zurückgeben, leiten Sie nur den `CloudFront-Is-Tablet-Viewer`-Header weiter und nicht den `CloudFront-Is-Desktop-Viewer`-Header.

Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern.

Entfernen des **`Accept-Encoding`**-Headers, wenn keine Kompression erforderlich ist

Wenn die Komprimierung nicht aktiviert ist, weil der Ursprung sie nicht unterstützt, sie nicht unterstützt oder weil der Inhalt CloudFront nicht komprimierbar ist, können Sie die Cache-Trefferquote erhöhen, indem Sie ein Cache-Verhalten in Ihrer Distribution einem Ursprung zuordnen, der Folgendes festlegt: Custom Origin Header

- Header name (Header-Name: `Accept-Encoding`
- Header value (Header-Wert): (Frei lassen)

Wenn Sie diese Konfiguration verwenden, wird der Header aus dem Cache-Schlüssel CloudFront entfernt und der `Accept-Encoding` Header nicht in die ursprünglichen Anfragen aufgenommen. Diese Konfiguration gilt für alle Inhalte, die mit CloudFront der Distribution von diesem Ursprung aus bereitgestellt werden.

Medieninhalte über HTTP bereitstellen

Weitere Informationen zum Optimieren von Video-on-Demand-(VOD)- und Streaming-Videoinhalten finden Sie unter Video-on-Demand und Live-Streaming-Video mit CloudFront.

Verwenden Sie Amazon CloudFront Origin Shield

CloudFront Origin Shield ist eine zusätzliche Ebene in der CloudFront Caching-Infrastruktur, die dazu beiträgt, die Auslastung Ihres Origins zu minimieren, seine Verfügbarkeit zu verbessern und die Betriebskosten zu senken. CloudFront Origin Shield bietet die folgenden Vorteile:

Bessere Cache-Zugriffsrate

Origin Shield kann dazu beitragen, die Cache-Trefferquote deiner CloudFront Distribution zu verbessern, da es eine zusätzliche Caching-Ebene vor deinem Origin bietet. Wenn du Origin Shield verwendest, werden alle Anfragen CloudFront von allen Cache-Layern an deinen Ursprung über Origin Shield geleitet, was die Wahrscheinlichkeit eines Cache-Treffers erhöht. CloudFrontkann jedes Objekt mit einer einzigen Ursprungsanfrage von Origin Shield an deinen Ursprung abrufen, und alle anderen Ebenen des CloudFront Caches (Edge-Standorte und regionale Edge-Caches) können das Objekt von Origin Shield abrufen.

Reduzierte Ursprungslast

Origin Shield kann die Anzahl der gleichzeitigen Anfragen die für dasselbe Objekt an Ihren Ursprung gesendet werden, weiter reduzieren. Anfragen für Inhalte, die sich nicht im Origin-Shield-Cache befinden, werden mit anderen Anforderungen für dasselbe Objekt konsolidiert, was dazu führt, dass nur eine Anfrage an Ihren Ursprung gelangt. Wenn Sie weniger Anfragen an Ihrem Ursprung bearbeiten, kann die Verfügbarkeit Ihres Ursprungs bei Spitzenlasten oder unerwarteten Verkehrsspitzen aufrechterhalten und die Kosten für Dinge wie just-in-time Verpackung, Bildtransformationen und ausgehende Datenübertragung (DTO) gesenkt werden.

Bessere Netzwerkleistung

Wenn du Origin Shield in der AWS Region aktivierst, die die niedrigste Latenz zu deinem Ursprung hat, kannst du eine bessere Netzwerkleistung erzielen. Bei Ursprüngen in einer AWS Region verbleibt der CloudFront Netzwerkverkehr bis zu deinem Ursprung im CloudFront Netzwerk mit hohem Durchsatz. Bei Ursprüngen außerhalb von AWS verbleibt der CloudFront Netzwerkverkehr im CloudFront Netzwerk bis hin zu Origin Shield, das über eine Verbindung mit niedriger Latenz zu deinem Ursprung verfügt.

Für die Nutzung von Origin Shield fallen zusätzliche Gebühren an. Weitere Informationen finden Sie unter CloudFront Preise.

> **Note**
>
> Origin Shield wird bei gRPC-Anfragen nicht unterstützt. Wenn bei einer Distribution, die gRPC unterstützt, Origin Shield aktiviert ist, funktionieren die gRPC-Anfragen weiterhin. Die Anfragen werden jedoch direkt an den gRPC-Ursprung weitergeleitet, ohne Origin Shield zu durchlaufen. Weitere Informationen finden Sie unter gRPC mit CloudFront Distributionen verwenden.

Themen

Anwendungsfälle für Origin Shield

CloudFront Origin Shield kann für viele Anwendungsfälle von Vorteil sein, darunter die folgenden:

- Viewer, die über verschiedene geografische Regionen verteilt sind
- Origins, die just-in-time Verpackungen für Live-Streaming oder on-the-fly Bildverarbeitung anbieten
- Lokale Ursprünge mit Kapazitäts- oder Bandbreitenbeschränkungen
- Workloads, die mehrere Netzwerke zur Inhaltsbereitstellung verwenden () CDNs

Origin Shield eignet sich möglicherweise in anderen Fällen nicht gut, z. B. dynamische Inhalte, die an den Ursprung weitergeleitet werden, Inhalte mit geringer Cachefähigkeit oder Inhalte, die selten angefordert werden.

In den folgenden Abschnitten werden die Vorteile von Origin Shield für die folgenden Anwendungsfälle erläutert.

Anwendungsfälle

Viewer in verschiedenen geografischen Regionen

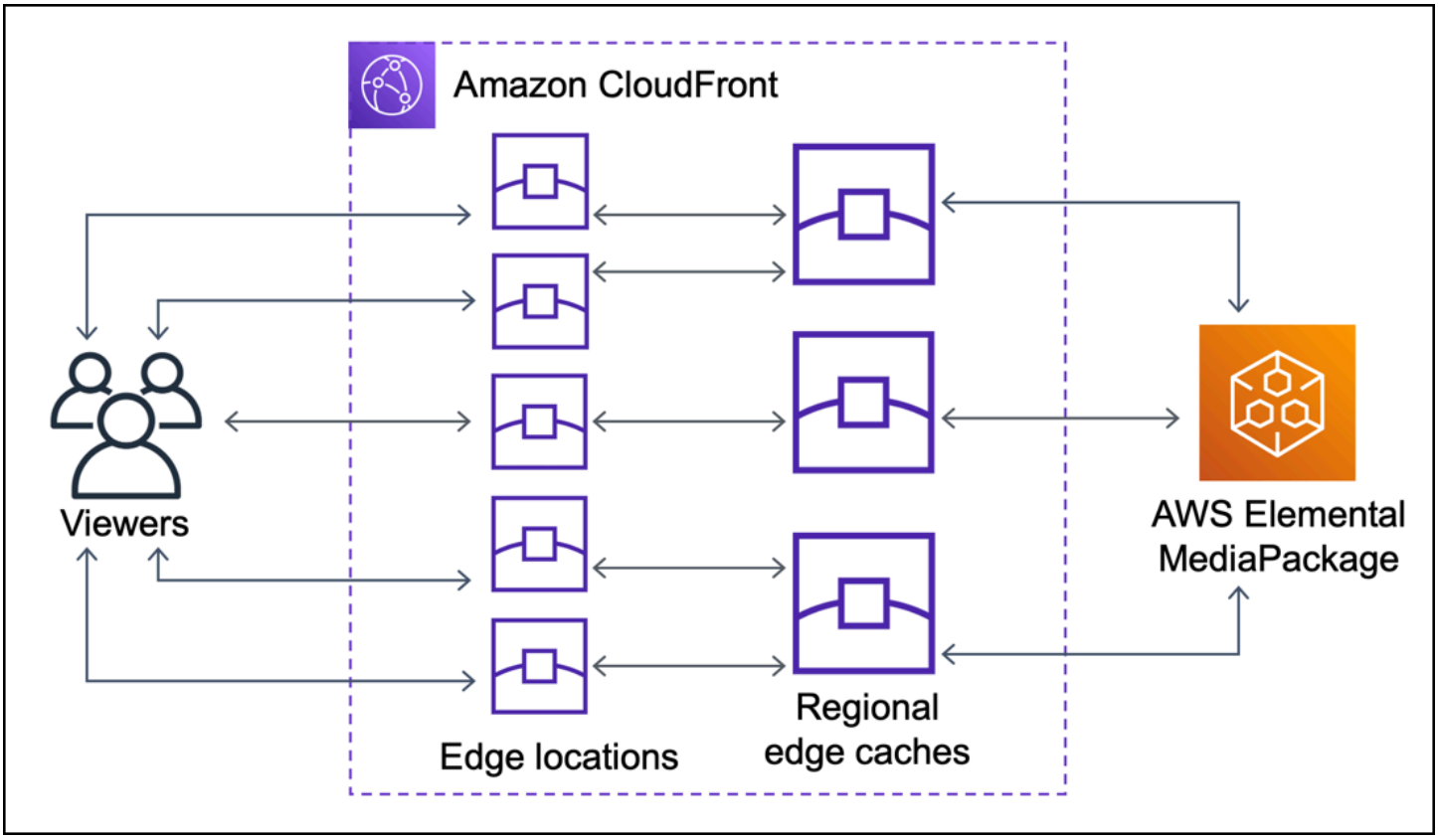
Mit Amazon wird die Belastung Ihres Ursprungs von Natur aus reduziert CloudFront, da Anfragen, die über den Cache bedient werden CloudFront können, nicht an Ihren Ursprung weitergeleitet werden. Zusätzlich zum CloudFront globalen Netzwerk von Edge-Standorten dienen regionale Edge-Caches als mittlere Caching-Ebene, um Cache-Treffer bereitzustellen und Anfragen von Zuschauern in nahegelegenen geografischen Regionen zu konsolidieren. Betrachteranfragen werden zuerst

an einen nahe gelegenen CloudFront -Edge-Standort weitergeleitet. Wenn das Objekt an diesem Speicherort nicht zwischengespeichert wird, wird die Anforderung an einen regionalen Edge-Cache gesendet.

Wenn sich Viewer in verschiedenen geografischen Regionen befinden, können Anfragen über verschiedene regionale Edge-Caches weitergeleitet werden, von denen jeder eine Anfrage für denselben Inhalt an Ihren Ursprung senden kann. Aber mit Origin Shield erhalten Sie eine zusätzliche Caching-Ebene zwischen den regionalen Edge-Caches und Ihrem Ursprung. Alle Anfragen von allen regionalen Edge-Caches laufen durch Origin Shield, wodurch die Belastung Ihres Ursprungs weiter reduziert wird. Das folgende Diagramm verdeutlicht dies. In den folgenden Diagrammen ist der Ursprung AWS Elemental MediaPackage.

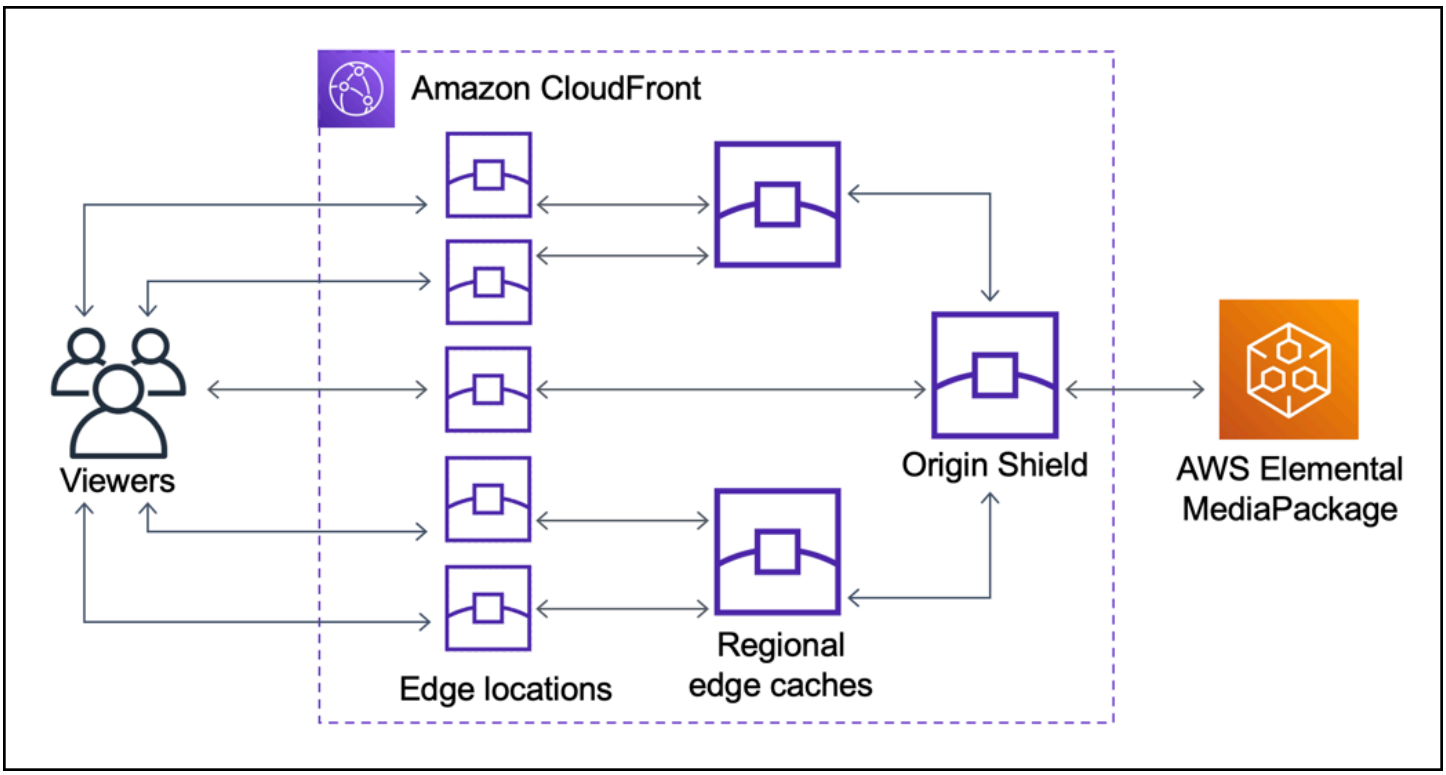
Ohne Origin Shield

Ohne Origin Shield erhält Ihr Ursprung möglicherweise doppelte Anfragen für denselben Inhalt, wie im folgenden Diagramm dargestellt.



Mit Origin Shield

Die Verwendung von Origin Shield kann dazu beitragen, die Belastung Ihres Ursprungs zu reduzieren, wie in der folgenden Abbildung gezeigt.



Mehrfach CDNs

Um Live-Videoveranstaltungen oder beliebte On-Demand-Inhalte bereitzustellen, können Sie mehrere Content Delivery Networks (CDNs) verwenden. Die Verwendung mehrerer Optionen CDNs kann gewisse Vorteile bieten, bedeutet aber auch, dass Ihr Absender möglicherweise viele doppelte Anfragen für denselben Inhalt erhält, die jeweils von unterschiedlichen CDNs oder unterschiedlichen Standorten innerhalb desselben CDN stammen. Diese redundanten Anfragen können sich negativ auf die Verfügbarkeit Ihrer Quelle auswirken oder zusätzliche Betriebskosten für Prozesse wie just-in-time Paketierung oder Datenübertragung (DTO) ins Internet verursachen.

Wenn du Origin Shield mit der Nutzung deiner CloudFront Distribution als Quelle für andere kombinierst CDNs, kannst du die folgenden Vorteile nutzen:

- Es gehen weniger redundante Anfragen bei deinem Absender ein, was dazu beiträgt, die negativen Auswirkungen der Verwendung mehrerer Anfragen zu reduzieren CDNs.
- Ein einheitlicher Cache-Schlüssel für alle und eine zentrale Verwaltung für Funktionen CDNs, die den Ursprung betreffen.
- Verbesserte Netzwerkleistung. Netzwerkverkehr von anderen CDNs wird an einem nahegelegenen CloudFront Edge-Standort beendet, was zu einem Treffer aus dem lokalen Cache führen kann. Wenn sich das angeforderte Objekt nicht im Edge-Location-Cache befindet, verbleibt die Anfrage

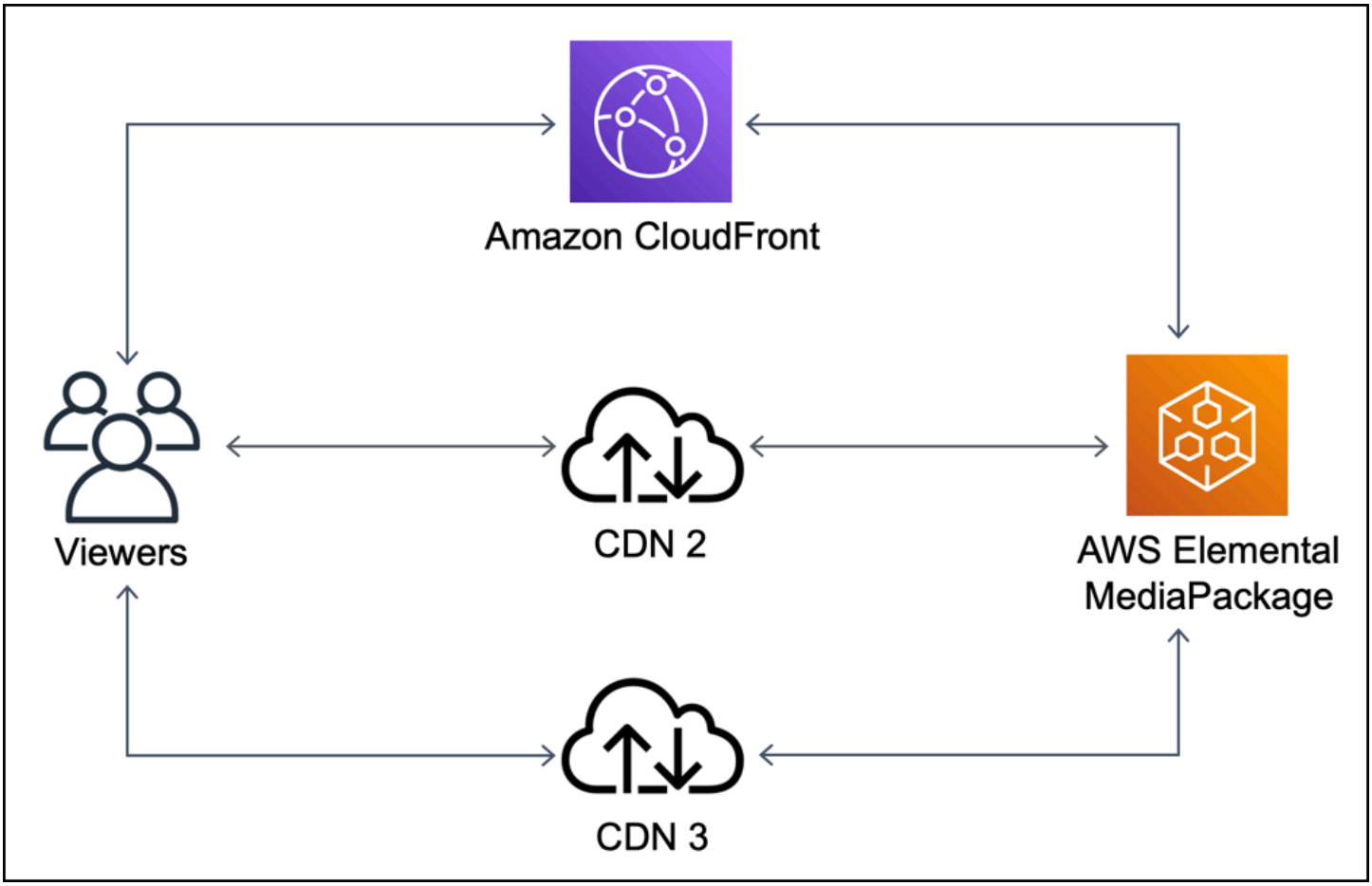
an den Ursprung bis Origin Shield im CloudFront Netzwerk, was einen hohen Durchsatz und eine geringe Latenz für den Ursprung bietet. Wenn sich das angeforderte Objekt im Origin-Shield-Cache befindet, wird die Anfrage an Ihren Ursprung vollständig vermieden.

> **Important**
>
> Wenn Sie Origin Shield in einer Multi-CDN-Architektur verwenden möchten und vergünstigte Preise erhalten möchten, kontaktieren Sie uns oder Ihren AWS Vertriebsmitarbeiter für weitere Informationen. Es können zusätzliche Gebühren anfallen.

Die folgenden Diagramme zeigen, wie diese Konfiguration dazu beitragen kann, die Belastung Ihres Origin zu minimieren, wenn Sie beliebte Live-Videoveranstaltungen mit mehreren Personen veranstalten. CDNs In den folgenden Diagrammen ist der Ursprung AWS Elemental MediaPackage.

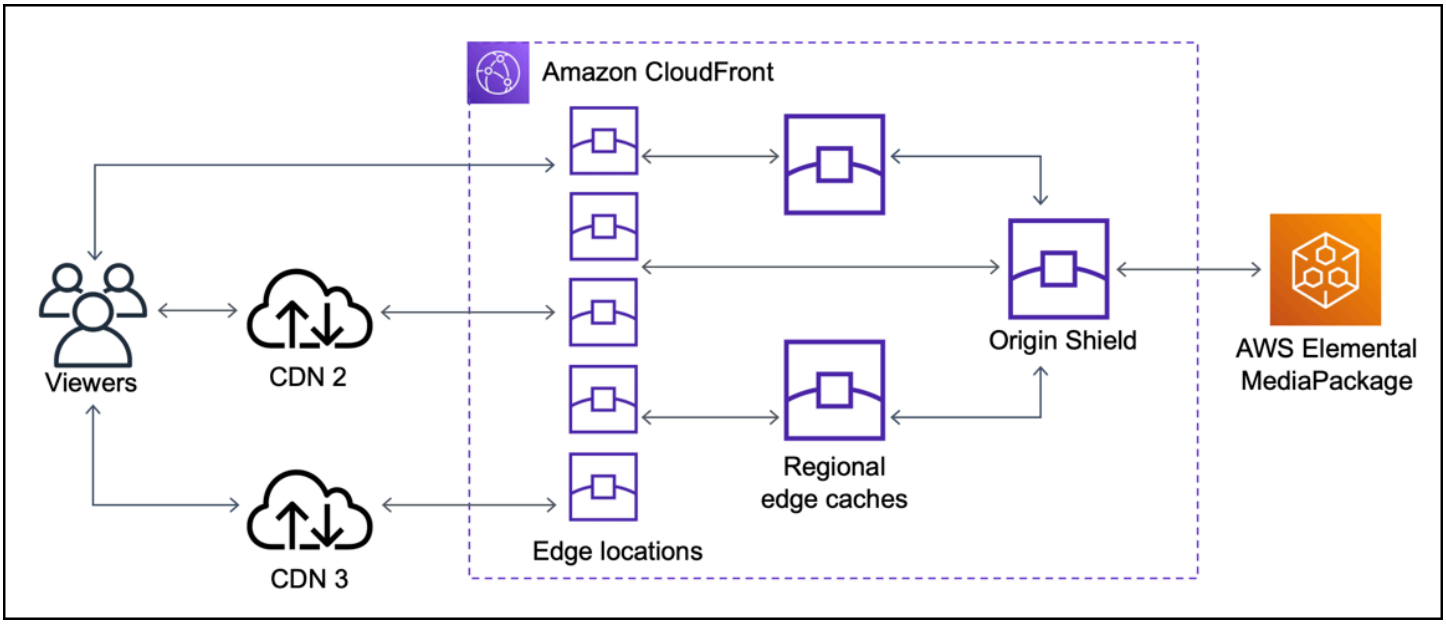
Ohne Origin Shield (mehrfach CDNs)

Ohne Origin Shield kann Ihr Ursprung viele doppelte Anfragen für denselben Inhalt erhalten, die jeweils von einem anderen CDN stammen, wie im folgenden Diagramm dargestellt.



Mit Origin Shield (mehrfach CDNs)

Die Verwendung von Origin Shield CloudFront als Ursprung für deinen anderen kann dazu beitragen CDNs, die Belastung deines Originals zu reduzieren, wie in der folgenden Abbildung dargestellt.



Wähle die AWS Region für Origin Shield

Amazon CloudFront bietet Origin Shield in AWS Regionen an, in denen CloudFront es einen regionalen Edge-Cache gibt. Wenn du Origin Shield aktivierst, wählst du die AWS Region für Origin Shield aus. Sie sollten die AWS -Region auswählen, die die niedrigste Latenz zu Ihrem Ursprung aufweist. Du kannst Origin Shield mit Ursprüngen verwenden, die in einer AWS Region liegen, und mit Ursprüngen, die sich nicht in einer Region befinden AWS.

Bei Ursprüngen in einer AWS -Region

Wenn dein Ursprung in einer AWS Region liegt, stelle zunächst fest, ob dein Ursprung in einer Region liegt, in der Origin Shield CloudFront angeboten wird. CloudFront bietet Origin Shield in den folgenden AWS Regionen an.

- US East (Ohio) – `us-east-2`
- USA Ost (Nord-Virginia) – `us-east-1`
- USA West (Oregon) – `us-west-2`
- Asia Pacific (Mumbai) – `ap-south-1`
- Asien-Pazifik (Seoul) – `ap-northeast-2`
- Asia Pacific (Singapore) – `ap-southeast-1`
- Asien-Pazifik (Sydney) – `ap-southeast-2`
- Asien-Pazifik (Tokio) – `ap-northeast-1`

- Europa (Frankfurt) – `eu-central-1`
- Europa (Ireland) – `eu-west-1`
- Europa (London) – `eu-west-2`
- South America (São Paulo) – `sa-east-1`
- Naher Osten (VAE) — `me-central-1`

Wenn dein Ursprung in einer AWS Region liegt, in der Origin Shield CloudFront angeboten wird

Wenn dein Ursprung in einer AWS Region liegt, in der Origin Shield CloudFront angeboten wird (siehe vorherige Liste), aktiviere Origin Shield in derselben Region wie dein Heimatland.

Wenn dein Ursprung nicht in einer AWS Region liegt, in der Origin Shield CloudFront angeboten wird

Wenn dein Ursprung nicht in einer AWS Region liegt, in der Origin Shield CloudFront angeboten wird, kannst du anhand der folgenden Tabelle herausfinden, in welcher Region Origin Shield aktiviert werden soll.

Bei einem Ursprung in...	Origin Shield aktivieren in...
US West (N. California) – `us-west-1`	US West (Oregon) – `us-west-2`
Africa (Cape Town) – `af-south-1`	Europe (Ireland) – `eu-west-1`
Asia Pacific (Hong Kong) – `ap-east-1`	Asia Pacific (Singapore) – `ap-southeast-1`
Canada (Central) – `ca-central-1`	US East (N. Virginia) – `us-east-1`
Europe (Milan) – `eu-south-1`	Europe (Frankfurt) – `eu-central-1`
Europe (Paris) – `eu-west-3`	Europe (London) – `eu-west-2`
Europe (Stockholm) – `eu-north-1`	Europe (London) – `eu-west-2`
Middle East (Bahrain) – `me-south-1`	Asia Pacific (Mumbai) – `ap-south-1`

Für Ursprünge außerhalb von AWS

Sie können Origin Shield mit einem Ursprung verwenden, der lokal ist oder sich nicht in einer AWS -Region befindet. Aktiviere in diesem Fall Origin Shield in der AWS Region, die die niedrigste Latenz

zu deinem Ursprung hat. Wenn du dir nicht sicher bist, welche AWS Region die niedrigste Latenz zu deinem Ursprung hat, kannst du die folgenden Vorschläge verwenden, um dir bei der Entscheidung zu helfen.

- In der obigen Tabelle finden Sie eine Näherung darüber, welche AWS -Region basierend auf der geografischen Lage Ihres Ursprungs die niedrigste Latenz zu Ihrem Ursprung hat.
- Sie können EC2 Amazon-Instances in verschiedenen AWS Regionen starten, die sich geografisch in der Nähe Ihres Ursprungs befinden, und einige Tests durchführen, `ping` um die typischen Netzwerklatenzen zwischen diesen Regionen und Ihrem Ursprung zu messen.

Origin Shield aktivieren

Sie können Origin Shield aktivieren, um Ihre Cache-Trefferquote zu verbessern, die Belastung Ihres Ursprungs zu reduzieren und die Leistung zu verbessern. Um Origin Shield zu aktivieren, ändere die Origin-Einstellungen in einer CloudFront Distribution. Origin Shield ist eine Eigenschaft des Ursprungs. Für jeden Ursprung in deinen CloudFront Distributionen kannst du Origin Shield separat in der AWS Region aktivieren, die für diesen Ursprung die beste Leistung bietet.

Du kannst Origin Shield in der CloudFront Konsole AWS CloudFormation, mit oder mit der CloudFront API aktivieren.

Console

So aktivieren Sie Origin Shield für einen vorhandenen Ursprung (Konsole)

1. Melde dich bei der an AWS Management Console und öffne die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die Verteilung mit dem Ursprung aus, den Sie aktualisieren möchten.
3. Wählen Sie den Tab Ursprünge aus.
4. Wählen Sie den zu aktualisierenden Ursprung und dann Edit (Bearbeiten) aus.
5. Wählen Sie bei Enable Origin Shield (Origin Shield aktivieren) die Option Yes (Ja) aus.
6. Wählen Sie bei Origin Shield Region (Origin-Shield-Region) die AWS -Region aus, in der Sie Origin Shield aktivieren möchten. Informationen zur Auswahl einer Region finden Sie unter Wähle die AWS Region für Origin Shield.
7. Wählen Sie Änderungen speichern aus.

Wenn der Verteilungsstatus Deployed (Bereitgestellt) lautet, ist Origin Shield bereit. Das dauert ein paar Minuten.

So aktivieren Sie Origin Shield für einen neuen Ursprung (Konsole)

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Gehen Sie wie folgt vor, um den neuen Ursprung in einer vorhandenen Verteilung zu erstellen:

   1. Wählen Sie die Verteilung aus, in der Sie den Ursprung erstellen möchten.
   2. Wählen Sie Create Origin (Ursprung erstellen) aus und fahren Sie mit Schritt 3 fort.

   Gehen Sie wie folgt vor, um den neuen Ursprung in einer neuen Standarddistribution zu erstellen:

   1. Folgen Sie den Schritten, um eine Standarddistribution in der Konsole zu erstellen. Weitere Informationen finden Sie unter Erstellen Sie eine CloudFront Distribution in der Konsole.
   2. Wählen Sie im Bereich Einstellungen die Option Origin-Einstellungen anpassen aus. Fahren Sie mit Schritt 3 fort.
3. Wählen Sie bei Enable Origin Shield (Origin Shield aktivieren) die Option Yes (Ja) aus.
4. Wählen Sie bei Origin Shield Region (Origin-Shield-Region) die AWS -Region aus, in der Sie Origin Shield aktivieren möchten. Informationen zur Auswahl einer Region finden Sie unter Wähle die AWS Region für Origin Shield.
5. Folgen Sie den Schritten in der Konsole, um die Erstellung Ihrer Herkunft oder Distribution abzuschließen.

Wenn der Verteilungsstatus Deployed (Bereitgestellt) lautet, ist Origin Shield bereit. Das dauert ein paar Minuten.

AWS CloudFormation

Um Origin Shield mit zu aktivieren AWS CloudFormation, verwende die `OriginShield` `Origin` Eigenschaft im Eigenschaftstyp einer `AWS::CloudFront::Distribution` Ressource. Sie können die `OriginShield`-Eigenschaft einem vorhandenen `Origin` hinzufügen oder sie einschließen, wenn Sie einen neuen `Origin` erstellen.

Das folgende Beispiel zeigt die Syntax im YAML-Format für die Aktivierung von `OriginShield` in der Region USA West (Oregon) (`us-west-2`). Informationen zur Auswahl einer Region finden Sie unter the section called “Wähle die AWS Region für Origin Shield”. In diesem Beispiel wird nur der `Origin`-Eigenschaftstyp und nicht die gesamte `AWS::CloudFront::Distribution`-Ressource angezeigt.

```
Origins:
- DomainName: 3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com
  Id: Example-EMP-3ae97e9482b0d011
  OriginShield:
    Enabled: true
    OriginShieldRegion: us-west-2
  CustomOriginConfig:
    OriginProtocolPolicy: match-viewer
    OriginSSLProtocols: TLSv1
```

Weitere Informationen findest du unter AWS::CloudFront::Distribution Origin im Abschnitt Ressourcen- und Eigenschaftenreferenzen des AWS CloudFormation Benutzerhandbuchs.

API

Um Origin Shield mit der CloudFront API mithilfe von AWS SDKs oder AWS Command Line Interface (AWS CLI) zu aktivieren, verwende den `OriginShield` Typ. Sie geben `OriginShield` in einem `Origin` in einer `DistributionConfig` an. Informationen zu diesem `OriginShield` Typ finden Sie in den folgenden Informationen in der Amazon CloudFront API-Referenz.

- OriginShield(Typ)
- Ursprung (Typ)
- DistributionConfig(Typ)
- UpdateDistribution(Betrieb)
- CreateDistribution(Betrieb)

Die spezifische Syntax für die Verwendung dieser Typen und Vorgänge variiert je nach SDK, CLI oder API-Client. Weitere Informationen finden Sie in der Referenzdokumentation zu Ihrem SDK, der CLI oder dem Client.

Schätzung der Origin Shield-Kosten

Für Origin Shield fallen Gebühren basierend auf der Anzahl der Anfragen an, die als inkrementelle Ebene an Origin Shield gelangen.

Bei dynamischen (nicht zwischenspeicherbaren) Anforderungen, die an den Ursprung weitergeleitet werden, ist Origin Shield immer eine inkrementelle Ebene. Dynamische Anfragen verwenden die HTTP-Methoden `PUT`,`POST`,`PATCH`, und`DELETE`.

`GET`und `HEAD` Anfragen mit einer Time-to-Live-Einstellung (TTL) von weniger als 3600 Sekunden werden als dynamische Anfragen betrachtet. Darüber hinaus gelten `HEAD` Anfragen, `GET` bei denen das Caching deaktiviert wurde, auch als dynamische Anfragen.

Verwenden Sie die folgende Formel, um Ihre Gebühren für Origin Shield für dynamische Anforderungen zu schätzen:

Gesamtzahl der dynamischen Anforderungen x Origin Shield-Ladung pro 10 000 Anforderungen/10 000

Für nicht dynamische Anfragen mit den HTTP-Methoden `GET` `HEAD`,`OPTIONS`, und ist Origin Shield manchmal eine inkrementelle Ebene. Wenn du Origin Shield aktivierst, wählst du das AWS-Region für Origin Shield. Für Anfragen, die natürlich an den regionalen Edge-Cache in derselben Region wie Origin Shield gehen, ist Origin Shield kein inkrementeller Layer. Für diese Anfragen fallen keine Origin Shield-Gebühren an. Für Anfragen, die an einen regionalen Edge-Cache in einer anderen Region als Origin Shield gehen und dann zu Origin Shield gehen, ist Origin Shield eine inkrementelle Ebene. Für diese Anfragen fallen Origin-Shield-Gebühren an.

Verwenden Sie die folgende Formel, um Ihre Gebühren für Origin Shield für zwischenspeicherbare Anfragen zu schätzen:

Gesamtzahl der zwischengespeicherten Anforderungen x (1 – Cache-Zugriffsrate) x Prozentsatz der Anforderungen, die von einem regionalen Edge-Cache in einer anderen Region an Origin Shield gelangen x Origin Shield-Ladung pro 10 000 Anforderungen /10 000

Weitere Informationen über die Gebühr pro 10 000 Anfragen für Origin Shield finden Sie unter CloudFront – Preise.

Hochverfügbarkeit bei Origin Shield

Origin Shield nutzt die Funktion für CloudFront regionale Edge-Caches. Jeder dieser Edge-Caches wird in einer AWS Region erstellt und verwendet mindestens drei Availability Zones mit Flotten von

auto-scaling Amazon-Instances. EC2 Verbindungen von CloudFront-Standorten mit Origin Shield verwenden auch die aktive Fehlerverfolgung für jede Anforderung, um die Anforderung automatisch an einen sekundären Origin Shield-Speicherort weiterzuleiten, wenn der primäre Speicherort von Origin Shield nicht verfügbar ist.

Wie Origin Shield mit anderen CloudFront Funktionen interagiert

In den folgenden Abschnitten wird erläutert, wie Origin Shield mit anderen CloudFront-Funktionen interagiert.

Origin Shield und CloudFront Protokollierung

Um zu sehen, wann Origin Shield eine Anforderung verarbeitet hat, müssen Sie eine der folgenden Optionen aktivieren:

- CloudFront Standardprotokolle (Zugriffsprotokolle). Standardprotokolle werden kostenlos zur Verfügung gestellt.
- CloudFront Protokolle in Echtzeit. Für die Verwendung von Echtzeitprotokollen fallen zusätzliche Gebühren an. Weitere Informationen finden Sie unter CloudFrontAmazon-Preise.

Cache-Treffer von Origin Shield werden wie `OriginShieldHit` im `x-edge-detailed-result-type` Feld in den CloudFront Protokollen angezeigt. Origin Shield nutzt die regionalen Edge-Caches CloudFront von Amazon. Wenn eine Anfrage von einem CloudFront Edge-Standort an den regionalen Edge-Cache weitergeleitet wird, der als Origin Shield fungiert, wird sie `Hit` in den Protokollen als a gemeldet, nicht als`OriginShieldHit`.

Origin Shield und Ursprungsgruppen

Origin Shield ist mit CloudFront -Ursprungsgruppen kompatibel. Da das Origin Shield eine Eigenschaft des Ursprungs ist, werden Anfragen für jeden Ursprung immer über Origin Shield geleitet, selbst wenn der Ursprung Teil einer Ursprungsgruppe ist. Leitet die Anfrage für eine bestimmte Anfrage über das Origin Shield des primären Ursprungs an den primären Ursprung in der Ursprungsgruppe weiter. CloudFront Wenn diese Anfrage fehlschlägt (gemäß den Failover-Kriterien der Ursprungsgruppe), wird CloudFront die Anfrage über Origin Shield des sekundären Ursprungs an den sekundären Ursprung weitergeleitet.

Origin Shield und Lambda@Edge

Origin Shield wirkt sich nicht auf die Funktionalität von Lambda @Edge-Funktionen aus, kann sich jedoch auf die AWS -Region auswirken, in der diese Funktionen ausgeführt werden.

Wenn du Origin Shield mit Lambda @Edge verwendest, werden auf den Ursprung gerichtete Trigger (Origin Request und Origin Response) in der AWS Region ausgeführt, in der Origin Shield aktiviert ist. Wenn der primäre Origin Shield-Standort nicht verfügbar ist und Anfragen CloudFront an einen sekundären Origin Shield-Standort weitergeleitet werden, wechseln Lambda @Edge -Trigger ebenfalls dazu, den sekundären Origin Shield-Standort zu verwenden.

Viewerorientierte Auslöser sind nicht betroffen.

Optimieren Sie die Hochverfügbarkeit mit CloudFront Origin Failover

Sie können Origin Failover für Szenarien einrichten CloudFront , die eine hohe Verfügbarkeit erfordern. Zunächst erstellen Sie eine Ursprungsgruppe mit zwei Ursprüngen: einem primären und einem sekundären. Wenn der primäre Ursprung nicht verfügbar ist oder bestimmte HTTP-Antwortstatuscodes zurückgibt, die auf einen Fehler hinweisen, CloudFront wird automatisch zum sekundären Ursprung gewechselt.

Zum Einrichten eines Origin Failovers müssen Sie eine Verteilung mit mindestens zwei Ursprüngen haben. Anschließend erstellen Sie eine Ursprungsgruppe für Ihre Verteilung, die die beiden Ursprünge enthält, und von denen Sie einen als primär einstellen. Schließlich erstellen oder aktualisieren Sie ein Cache-Verhalten, um die Ursprungsgruppe zu verwenden.

Informationen zu den Schritten zum Einrichten von Ursprungsgruppen und zum Konfigurieren bestimmter Ursprungs-Failover-Optionen finden Sie unter Erstellen Sie eine Ursprungsgruppe.

Nachdem Sie das Origin-Failover für ein Cache-Verhalten konfiguriert haben, CloudFront geht es für Viewer-Anfragen wie folgt vor:

- Gibt bei einem Cache-Treffer das angeforderte Objekt CloudFront zurück.
- Wenn ein Cache-Fehler auftritt, CloudFront leitet die Anfrage an den primären Ursprung in der Ursprungsgruppe weiter.
- Wenn der primäre Ursprung einen Statuscode zurückgibt, der nicht für Failover konfiguriert ist, z. B. einen HTTP-Statuscode 2xx oder 3xx, wird das angeforderte CloudFront Objekt dem Betrachter zugestellt.

- Wenn eine der folgenden Bedingungen eintritt:
  - Der primäre Ursprung gibt einen HTTP-Statuscode zurück, den Sie für das Failover konfiguriert haben
  - CloudFront kann keine Verbindung zum primären Ursprung herstellen (wenn 503 als Failover-Code festgelegt ist)
  - Die Antwort vom primären Ursprung dauert zu lange (Timeout) (wenn 504 als Failover-Code festgelegt ist)

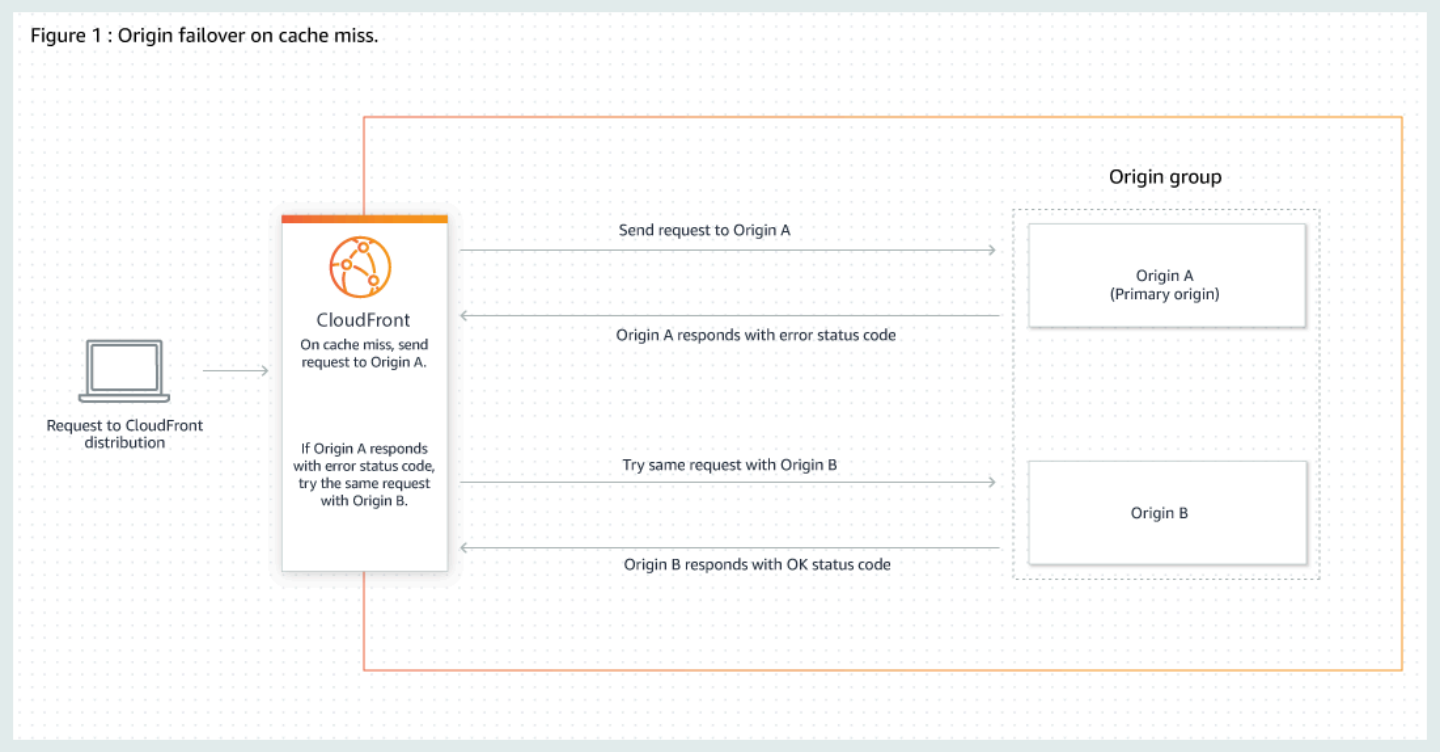
  CloudFront Leitet die Anfrage dann an den sekundären Ursprung in der Ursprungsgruppe weiter.

  > **Note**
  >
  > In einigen Anwendungsfällen, z. B. beim Streamen von Videoinhalten, CloudFront möchten Sie möglicherweise schnell auf den sekundären Ursprung umschalten. Informationen darüber, wie schnell ein CloudFront Failover zum sekundären Ursprung erfolgen soll, finden Sie unterKontrolliere Timeouts und Versuche bei der Herkunft.

CloudFront leitet alle eingehenden Anfragen an den primären Ursprung weiter, auch wenn bei einer vorherigen Anfrage ein Failover zum sekundären Ursprung erfolgt ist. CloudFront sendet Anfragen nur dann an den sekundären Ursprung, wenn eine Anfrage an den primären Ursprung fehlschlägt.

CloudFront führt nur dann einen Failover zum sekundären Ursprung durch, wenn die HTTP-Methode der Viewer-Anfrage`GET`,`HEAD`, oder ist`OPTIONS`. CloudFront führt nicht zu einem Failover, wenn der Betrachter eine andere HTTP-Methode sendet (z. B. `POSTPUT`, usw.).

Das folgende Diagramm veranschaulicht, wie Origin Failover funktioniert.



Themen

- Erstellen Sie eine Ursprungsgruppe
- Kontrolliere Timeouts und Versuche bei der Herkunft
- Verwenden von Origin Failover mit Lambda@Edge-Funktionen
- Verwenden von benutzerdefinierten Fehlerseiten mit Ursprung-Failover

Erstellen Sie eine Ursprungsgruppe

So erstellen Sie eine Ursprungsgruppe

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die Verteilung aus, für die Sie die Ursprungsgruppe erstellen möchten.
3. Wählen Sie den Tab Ursprünge aus.
4. Stellen Sie sicher, dass die Verteilung mehr als einen Ursprung hat. Wenn dies nicht der Fall ist, fügen Sie einen zweiten Ursprung hinzu.
5. Wählen Sie auf der Registerkarte Ursprünge im Bereich Ursprungsgruppen die Option Ursprungsgruppe erstellen aus.

6. Wählen Sie die Ursprünge für die Ursprungsgruppe aus. Nachdem Sie Ursprünge hinzugefügt haben, verwenden Sie die Pfeile, um die Priorität festzulegen, d. h., welcher Ursprung primär und welcher sekundär ist.
7. Geben Sie einen Namen für die Ursprungsgruppe ein.
8. Wählen Sie die HTTP-Statuscodes aus, die als Failover-Kriterien verwendet werden sollen. Sie können eine beliebige Kombination der folgenden Statuscodes wählen: 400, 403, 404, 416, 500, 502, 503 oder 504. Wenn CloudFront Sie eine Antwort mit einem der von Ihnen angegebenen Statuscodes erhalten, erfolgt ein Failover zum sekundären Ursprung.

   > **Note**
   >
   > CloudFront führt nur dann ein Failover zum sekundären Ursprung durch, wenn die HTTP-Methode der Viewer-Anfrage `GET`, `HEAD`, oder ist `OPTIONS`. CloudFront führt nicht zu einem Failover, wenn der Betrachter eine andere HTTP-Methode sendet (z. B. `POST`, `PUT`, usw.).

9. Geben Sie unter Auswahlkriterien für Herkunft an, wie Ihre Ursprünge ausgewählt werden, wenn Ihre Distribution Viewer-Anfragen weiterleitet. Sie können die folgenden Optionen wählen.

   Standard

   CloudFront verwendet die standardmäßige Ausgangspriorität, die Sie auf der Seite „Einstellungen“ angeben.

   Bewertung der Medienqualität

   CloudFront verfolgt und verwendet diesen Wert, um den ersten Ursprung zu bestimmen, an den die Anfrage weitergeleitet werden soll. Dies berechtigt auch CloudFront dazu, asynchrone `HEAD` Anfragen an den alternativen Ursprung in der Ursprungsgruppe zu stellen, um dessen Medienqualitätsfaktor zu ermitteln. Sie können diese Option nur für AWS Elemental MediaPackage v2-Ursprünge wählen. Weitere Informationen finden Sie unter Resilienz im Hinblick auf Medienqualität.
10. Wählen Sie Ursprungsgruppe erstellen aus.

Stellen Sie sicher, dass Sie Ihre Ursprungsgruppe als Ursprung für das Cache-Verhalten Ihrer Distribution angeben. Weitere Informationen finden Sie unter Name.

Kontrolliere Timeouts und Versuche bei der Herkunft

CloudFront Versucht standardmäßig, bis zu 30 Sekunden lang eine Verbindung zum primären Ursprung in einer Ursprungsgruppe herzustellen (3 Verbindungsversuche von jeweils 10 Sekunden), bevor ein Failover zum sekundären Ursprung erfolgt. In einigen Anwendungsfällen, z. B. beim Streamen von Videoinhalten, CloudFront möchten Sie möglicherweise schneller auf den sekundären Ursprung umschalten. Sie können die folgenden Einstellungen anpassen, um festzulegen, wie schnell ein CloudFront Failover zum sekundären Ursprung erfolgen soll. Handelt es sich bei dem Ursprung um einen sekundären Ursprung oder um einen Ursprung, der nicht Teil einer Ursprungsgruppe ist, beeinflussen diese Einstellungen, wie schnell eine HTTP 504-Antwort an den Betrachter CloudFront zurückgegeben wird.

Wenn Sie ein Failover schneller ausführen möchten, geben Sie ein kürzeres Verbindungstimeout, weniger Verbindungsversuche, oder beides an. Für benutzerdefinierte Ursprünge (einschließlich Amazon S3-Bucket-Ursprünge, die mit statischem Website-Hosting konfiguriert sind) können Sie auch das Timeout der Ursprungsantwort anpassen.

Timeout der Ursprungsverbindung

Die Einstellung für das Timeout für die ursprüngliche Verbindung beeinflusst, wie lange CloudFront gewartet wird, wenn versucht wird, eine Verbindung zum Ursprung herzustellen. CloudFront Wartet standardmäßig 10 Sekunden, bis eine Verbindung hergestellt wird. Sie können jedoch 1—10 Sekunden (einschließlich) angeben. Weitere Informationen finden Sie unter Verbindungstimeout.

Verbindungsversuche zum Ursprung

Die Einstellung für Verbindungsversuche über den Ursprung wirkt sich darauf aus, wie oft CloudFront versucht wird, eine Verbindung zum Ursprung herzustellen. Standardmäßig wird dreimal CloudFront versucht, eine Verbindung herzustellen, aber Sie können 1—3 (einschließlich) angeben. Weitere Informationen finden Sie unter Verbindungsversuche.

Für einen benutzerdefinierten Ursprung (einschließlich eines Amazon S3 S3-Buckets, der mit statischem Website-Hosting konfiguriert ist) wirkt sich diese Einstellung auch darauf aus, wie oft CloudFront versucht wird, eine Antwort vom Ursprung zu erhalten, falls es zu einem Timeout für die ursprüngliche Antwort kommt.

Ursprungs-Reaktions-Timeout

Das Timeout für die ursprüngliche Antwort, auch bekannt als Origin-Read-Timeout, beeinflusst, wie lange auf CloudFront den Empfang einer Antwort (oder auf den Erhalt der vollständigen

Antwort) vom Ursprung gewartet wird. CloudFront Wartet standardmäßig 30 Sekunden, aber Sie können 1—120 Sekunden (einschließlich) angeben. Weitere Informationen finden Sie unter Timeout bei der Antwort.

So ändern Sie diese Einstellungen

So ändern Sie diese Einstellungen in der CloudFront-Konsole

- Für einen neuen Ursprung oder eine neue Verteilung geben Sie diese Werte an, wenn Sie die Ressource erstellen.
- Für einen vorhandenen Ursprung in einer vorhandenen Verteilung geben Sie diese Werte an, wenn Sie den Ursprung bearbeiten.

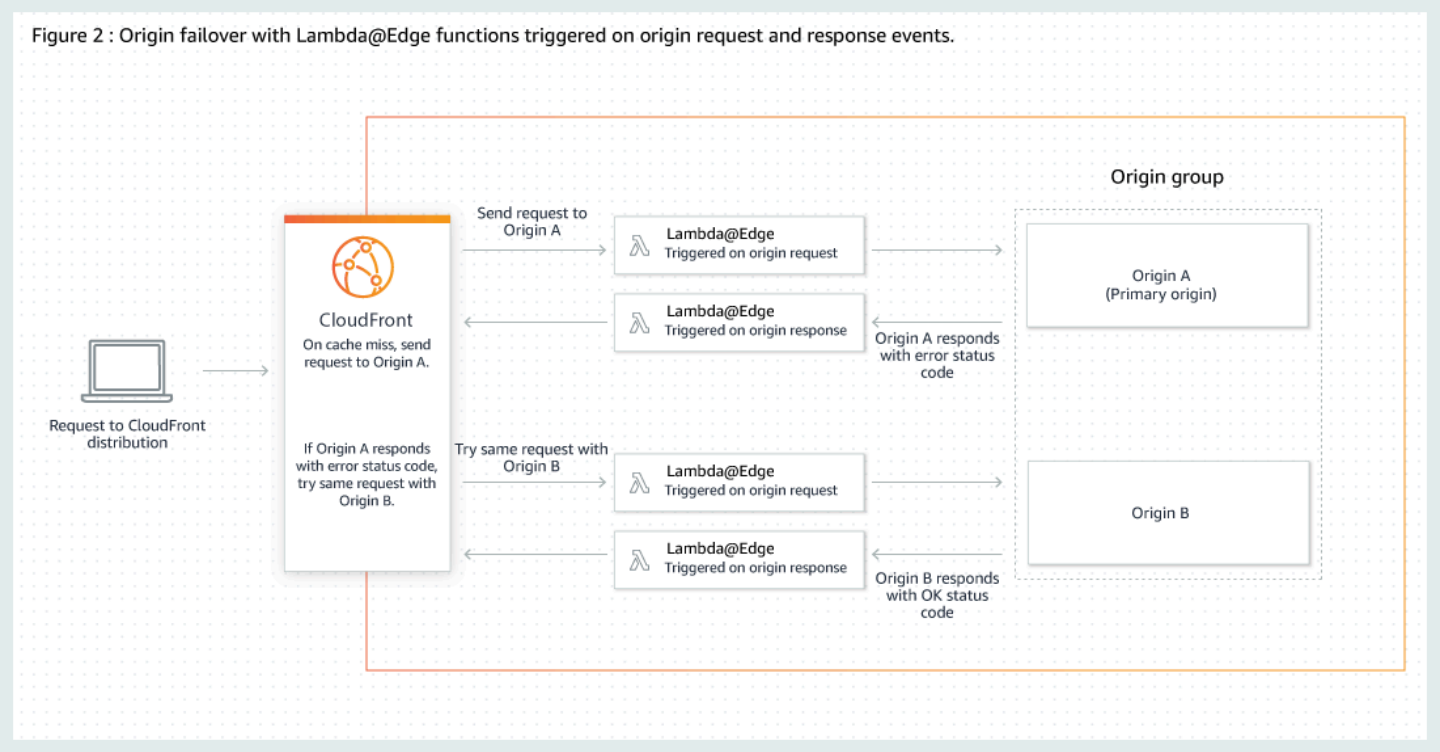
Weitere Informationen finden Sie unter Referenz für alle Verteilungseinstellungen.

Verwenden von Origin Failover mit Lambda@Edge-Funktionen

Sie können Lambda @Edge -Funktionen mit CloudFront Verteilungen verwenden, die Sie mit Ursprungsgruppen eingerichtet haben. Um eine Lambda-Funktion zu verwenden, geben Sie sie in einem Ursprungsanforderungs- oder Ursprungsantwort-Auslöser für eine Ursprungsgruppe an, wenn Sie die Cache-Verhaltenweise erstellen. Wenn Sie eine Lambda@Edge-Funktion mit einer Ursprungsgruppe verwenden, kann die Funktion zweimal für eine einzelne Viewer-Anfrage ausgelöst werden. Betrachten Sie beispielsweise folgendes Szenario:

1. Sie erstellen eine Lambda@Edge-Funktion mit einem Ursprungsanfrage-Auslöser.
2. Die Lambda-Funktion wird einmal ausgelöst, wenn eine Anfrage an den primären Ursprung CloudFront gesendet wird (bei einem Cache-Fehler).
3. Der primäre Ursprung antwortet mit einem HTTP-Statuscode, der für das Failover konfiguriert ist.
4. Die Lambda-Funktion wird erneut ausgelöst, wenn dieselbe Anfrage CloudFront an den sekundären Ursprung gesendet wird.

Das folgende Diagramm illustriert die Funktionsweise von Origin Failover, wenn Sie eine Lambda@Edge-Funktion in eine Ursprungsanfrage oder einen Antwort-Auslöser einbeziehen.



Weitere Informationen zur Verwendung von Lambda@Edge-Auslösern finden Sie unter the section called “Trigger für eine Lambda @Edge -Funktion hinzufügen”.

Weitere Informationen zur Verwaltung von DNS-Failover finden Sie unter Konfiguration des DNS-Failovers im Amazon Route 53-Entwicklerhandbuch.

Verwenden von benutzerdefinierten Fehlerseiten mit Ursprung-Failover

Sie können benutzerdefinierte Fehlerseiten mit Ursprungsgruppen ähnlich verwenden wie Ursprünge, die nicht für Origin Failover konfiguriert wurden.

Wenn Sie Origin-Failover verwenden, können Sie so konfigurieren, CloudFront dass eine benutzerdefinierte Fehlerseite für den primären oder sekundären Ursprung (oder beide) zurückgegeben wird:

- Eine benutzerdefinierte Fehlerseite für den primären Ursprung zurückgeben — Wenn der primäre Ursprung einen HTTP-Statuscode zurückgibt, der nicht für Failover konfiguriert ist, wird die benutzerdefinierte Fehlerseite an die Zuschauer CloudFront zurückgegeben.
- Eine benutzerdefinierte Fehlerseite für den sekundären Ursprung zurückgeben — Wenn ein Fehlerstatuscode vom sekundären Ursprung CloudFront empfangen wird, wird die benutzerdefinierte Fehlerseite CloudFront zurückgegeben.

Weitere Informationen zur Verwendung von benutzerdefinierten Fehlerseiten mit CloudFront finden Sie unterGenerieren Sie benutzerdefinierte Fehlerantworten.

Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf)

Sie können kontrollieren, wie lange Ihre Dateien in einem CloudFront Cache bleiben, bevor Sie CloudFront eine weitere Anfrage an Ihren Ursprung weiterleiten. Eine Reduzierung der Dauer ermöglicht Ihnen, dynamische Inhalte bereitzustellen. Eine Erhöhung der Dauer bedeutet, dass Ihre Benutzer eine bessere Leistung erhalten, da es wahrscheinlicher ist, dass Ihre Dateien direkt vom Edge-Cache bereitgestellt werden. Eine längere Dauer verringert darüber hinaus die Last auf Ihrem Ursprung.

In der Regel CloudFront wird eine Datei von einem Edge-Standort aus bereitgestellt, bis die von Ihnen angegebene Cache-Dauer abgelaufen ist, d. h. bis die Datei abläuft. Wenn der Edge-Standort nach Ablauf das nächste Mal eine Anfrage für die Datei erhält, CloudFront leitet er die Anfrage an den Ursprung weiter, um zu überprüfen, ob der Cache die neueste Version der Datei enthält. Die Antwort vom Ursprung hängt davon ab, ob die Datei geändert wurde:

- Wenn der CloudFront Cache bereits über die neueste Version verfügt, gibt der Ursprung einen Statuscode `304 Not Modified` zurück.
- Wenn der CloudFront Cache nicht über die neueste Version verfügt, gibt der Ursprung einen Statuscode `200 OK` und die neueste Version der Datei zurück.

Wenn eine Datei an einem Edge-Speicherort nicht häufig angefordert wird, kann es CloudFront sein, dass die Datei entfernt wird, d. h. sie wird vor dem Ablaufdatum entfernt, um Platz für Dateien zu schaffen, die in jüngerer Zeit angefordert wurden.

Wir empfehlen, die Cache-Dauer zu verwalten, indem Sie die Cache-Richtlinie Ihrer Distribution aktualisieren. Wenn Sie die Verwendung einer Cache-Richtlinie deaktivieren, beträgt die Standard-TTL (Time to Live) 24 Stunden. Sie können jedoch die folgenden Einstellungen aktualisieren, um die Standardeinstellung zu überschreiben:

- Um die Cachedauer für alle Dateien zu ändern, die demselben Pfadmuster entsprechen, können Sie die CloudFront Einstellungen für Minimale TTL, Maximale TTL und Standard-TTL für ein Cacheverhalten ändern. Informationen zu den einzelnen Einstellungen finden Sie unterMindest-TTL, undHöchst-TTL. Standard-TTL

- Um die Cache-Dauer für eine einzelne Datei zu ändern, können Sie Ihren Ursprung so konfigurieren, dass ein `Cache-Control`-Header mit der `max-age`- oder `s-maxage`-Richtlinie oder ein `Expires`-Header zu der Datei hinzugefügt wird. Weitere Informationen finden Sie unter Verwenden Sie Header, um die Cache-Dauer für einzelne Objekte zu steuern.

Weitere Informationen dazu, wie Mindest-TTL, Standard-TTL und Höchst-TTL mit `max-age`- und `s-maxage`-Richtlinien und dem `Expires`-Header-Feld interagieren, finden Sie unter the section called “Geben Sie an, wie lange Objekte zwischengespeichert werden CloudFront ”.

Sie können auch steuern, wie lange Fehler (z. B.`404 Not Found`) in einem CloudFront Cache verbleiben, bevor erneut CloudFront versucht wird, das angeforderte Objekt abzurufen, indem Sie eine weitere Anfrage an Ihren Ursprung weiterleiten. Weitere Informationen finden Sie unter the section called “Wie CloudFront werden die HTTP-Statuscodes 4xx und 5xx von Ihrem Ursprung verarbeitet”.

Themen

- Verwenden Sie Header, um die Cache-Dauer für einzelne Objekte zu steuern
- Stellt veraltete (abgelaufene) Inhalte bereit
- Geben Sie an, wie lange Objekte zwischengespeichert werden CloudFront
- Fügen Sie mithilfe der Amazon S3 S3-Konsole Header zu Ihren Objekten hinzu

Verwenden Sie Header, um die Cache-Dauer für einzelne Objekte zu steuern

Sie können die Header `Cache-Control` und `Expires` verwenden, um zu steuern, wie lange Objekte im Cache zwischengespeichert werden. Die Einstellungen für Mindest-TTL, Standard-TTL und Höchst-TTL wirken sich auch auf die Cache-Dauer aus. Im Folgenden finden Sie einen Überblick darüber, wie sich Header auf die Cache-Dauer auswirken können:

- Mit der `Cache-Control max-age` Direktive können Sie angeben, wie lange (in Sekunden) ein Objekt im Cache verbleiben soll, bevor CloudFront es erneut vom Ursprungsserver abgerufen wird. Die unterstützte Mindestablaufzeit CloudFront beträgt 0 Sekunden. Die Höchstwert beträgt 100 Jahre. Geben Sie den Wert im folgenden Format an:

  ```
  Cache-Control: max-age=seconds
  ```

Die folgende Direktive weist beispielsweise darauf CloudFront hin, dass das zugehörige Objekt 3600 Sekunden (eine Stunde) im Cache aufbewahrt werden soll:

```
Cache-Control: max-age=3600
```

Wenn Sie möchten, dass Objekte für eine andere Dauer in CloudFront Edge-Caches verbleiben als in Browser-Caches, können Sie die Direktiven `Cache-Control max-age` und `Cache-Control s-maxage` zusammen verwenden. Weitere Informationen finden Sie unter Geben Sie an, wie lange Objekte zwischengespeichert werden CloudFront .

- Im `Expires`-Header-Feld können Sie ein Ablaufdatum und eine Ablaufzeit festlegen. Verwenden Sie dafür das in RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1 Abschnitt 3.3.1, Full Date angegebene Format, zum Beispiel:

  ```
  Sat, 27 Jun 2015 23:59:59 GMT
  ```

Wir empfehlen, die `Cache-Control max-age`-Richtlinie anstelle des `Expires`-Header-Felds zum Steuern der Zwischenspeicherung von Objekten zu verwenden. Wenn Sie sowohl für `Cache-Control max-age` als auch für `Expires` Werte festlegen, verwendet CloudFront nur den Wert von `Cache-Control max-age`.

Weitere Informationen finden Sie unter Geben Sie an, wie lange Objekte zwischengespeichert werden CloudFront .

Sie können die HTTP `Cache-Control` - oder `Pragma` Header-Felder nicht in einer `GET` Anfrage eines Betrachters verwenden, um CloudFront zu erzwingen, für das Objekt zum Ursprungsserver zurückzukehren. CloudFront ignoriert diese Header-Felder in Viewer-Anfragen.

Weitere Informationen zu den Header-Feldern `Cache-Control` und `Expires` finden Sie in den folgenden Abschnitt in RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1:

- Abschnitt 14.9 Cache-Kontrolle
- Abschnitt 14.21 Läuft ab

Stellt veraltete (abgelaufene) Inhalte bereit

CloudFront unterstützt die Steueranweisungen `Stale-While-Revalidate` und die `Stale-If-Error` Cache-Steueranweisungen. Sie können diese Direktiven verwenden, um anzugeben, wie lange veraltete Inhalte für Zuschauer verfügbar sind.

Themen

`Stale-While-Revalidate`

Diese Direktive CloudFront ermöglicht es, veraltete Inhalte aus dem Cache bereitzustellen und gleichzeitig CloudFront asynchron eine neue Version vom Ursprung abzurufen. Dies verbessert die Latenz, da Zuschauer sofort Antworten von Edge-Standorten erhalten, ohne auf den Abruf im Hintergrund warten zu müssen. Für future Anfragen werden neue Inhalte im Hintergrund geladen.

Example Beispiel: **`Stale-While-Revalidate`**

CloudFront macht Folgendes, wenn Sie den `Cache-Control` Header so einstellen, dass er diese Direktiven verwendet.

```
Cache-Control: max-age=3600, stale-while-revalidate=600
```

1. CloudFront speichert eine Antwort für eine Stunde im Cache (`max-age=3600`).
2. Wenn eine Anfrage nach Ablauf dieser Dauer gestellt wird, wird der CloudFront veraltete Inhalt bereitgestellt und gleichzeitig eine Anfrage an den Ursprung gesendet, um den zwischengespeicherten Inhalt erneut zu validieren und zu aktualisieren.
3. Während der Inhalt erneut validiert wird, CloudFront wird der veraltete Inhalt bis zu 10 Minuten lang bereitgestellt (). `stale-while-revalidate=600`

> **Note**
>
> CloudFront stellt den veralteten Inhalt bis zum Wert der `stale-while-revalidate` Direktive oder bis zum Wert der CloudFront maximalen TTL bereit, je nachdem, welcher Wert

niedriger ist. Nach Ablauf der maximalen TTL-Dauer ist das veraltete Objekt unabhängig vom Wert nicht mehr im Edge-Cache verfügbar. `stale-while-revalidate`

`Stale-If-Error`

Diese Direktive ermöglicht CloudFront die Bereitstellung veralteter Inhalte aus dem Cache, wenn der Ursprung nicht erreichbar ist, oder gibt einen Fehlercode zurück, der zwischen 500 und 600 liegt. Dadurch wird sichergestellt, dass Viewer auch während eines Ausfalls des Ursprungs auf Inhalte zugreifen können.

Example Beispiel: **`Stale-If-Error`**

CloudFront macht Folgendes, wenn Sie den `Cache-Control` Header so einstellen, dass er diese Direktiven verwendet.

```
Cache-Control: max-age=3600, stale-if-error=86400
```

1. CloudFront speichert die Antwort für eine Stunde im Cache (`max-age=3600`).
2. Wenn der Ursprung nicht verfügbar ist oder nach Ablauf dieser Zeit eine Fehlermeldung zurückgibtCloudFront , wird der veraltete Inhalt bis zu 24 Stunden lang bereitgestellt () `stale-if-error=86400`
3. Wenn Sie benutzerdefinierte Fehlerantworten konfiguriert haben, CloudFront wird versucht, den veralteten Inhalt bereitzustellen, falls innerhalb der angegebenen `stale-if-error` Dauer ein Fehler auftritt. Wenn der veraltete Inhalt nicht verfügbar ist, CloudFront werden dann die benutzerdefinierten Fehlerantworten bereitgestellt, die Sie für den entsprechenden Fehlerstatuscode konfiguriert haben. Weitere Informationen finden Sie unter Generieren Sie benutzerdefinierte Fehlerantworten.

Hinweise

- CloudFront liefert den veralteten Inhalt bis zum Wert der `stale-if-error` Direktive oder bis zum Wert der CloudFront maximalen TTL, je nachdem, welcher Wert niedriger ist. Nach Ablauf der maximalen TTL-Dauer ist das veraltete Objekt unabhängig vom Wert nicht mehr im Edge-Cache verfügbar. `stale-if-error`
- Wenn Sie keine Fehlerantworten konfigurieren `stale-if-error` oder anpassen, CloudFront wird das veraltete Objekt zurückgegeben oder die Fehlerantwort zurück an

den Viewer weitergeleitet, je nachdem, ob sich das angeforderte Objekt im Edge-Cache befindet oder nicht. Weitere Informationen finden Sie unter Wie CloudFront werden Fehler verarbeitet, wenn Sie keine benutzerdefinierten Fehlerseiten konfiguriert haben.

Verwenden Sie beide Direktiven

Bei beiden `stale-while-revalidate` `stale-if-error` handelt es sich um unabhängige Cache-Steuerungsanweisungen, die Sie zusammen verwenden können, um die Latenz zu reduzieren und einen Puffer hinzuzufügen, damit Ihr Ursprung reagieren oder wiederherstellen kann.

Example Beispiel: Verwendung beider Direktiven

CloudFront macht Folgendes, wenn Sie den `Cache-Control` Header so einstellen, dass er die folgenden Direktiven verwendet.

```
Cache-Control: max-age=3600, stale-while-revalidate=600, stale-if-error=86400
```

1. CloudFront speichert die Antwort für eine Stunde im Cache (`max-age=3600`).
2. Wenn eine Anfrage nach Ablauf dieser Dauer gestellt wird, wird CloudFront der veraltete Inhalt bis zu 10 Minuten lang (`stale-while-revalidate=600`) bereitgestellt, während der Inhalt erneut validiert wird.
3. Wenn der Ursprungsserver beim CloudFront Versuch, den Inhalt zu revalidieren, einen Fehler zurückgibt, CloudFront stellt er den veralteten Inhalt bis zu 24 Stunden lang weiter bereit (). `stale-if-error=86400`

Caching stellt ein Gleichgewicht zwischen Leistung und Aktualität her. Die Verwendung von Richtlinien wie `stale-while-revalidate` und `stale-if-error` kann die Leistung und den Benutzerkomfort verbessern. Achten Sie jedoch darauf, dass die Konfigurationen darauf abgestimmt sind, wie aktuell Ihre Inhalte sein sollen. Richtlinien für veraltete Inhalte eignen sich am besten für Anwendungsfälle, in denen Inhalte aktualisiert werden müssen, die neueste Version jedoch nicht unbedingt erforderlich ist. Wenn sich Ihr Inhalt nicht oder nur selten ändert, kann `stale-while-revalidate` außerdem zu unnötigen Netzwerkanforderungen führen. Erwägen Sie stattdessen, eine lange Cachedauer festzulegen.

Geben Sie an, wie lange Objekte zwischengespeichert werden CloudFront

Um zu kontrollieren, wie lange ein Objekt im Cache CloudFront aufbewahrt wird, bevor eine weitere Anfrage an den Ursprung gesendet wird, können Sie:

- Legen Sie die minimalen, maximalen und standardmäßigen TTL-Werte für das Cache-Verhalten einer CloudFront Distribution fest. Sie können diese Werte in einer Cache-Richtlinie festlegen, die an das Cache-Verhalten (empfohlen) oder in den Legacy-Cache-Einstellungen angehängt ist.
- Die `Cache-Control`- oder `Expires`-Header in Antworten vom Ursprung einschließen. Diese Header helfen auch dabei zu bestimmen, wie lange ein Browser ein Objekt im Browser-Cache aufbewahrt, bevor eine weitere Anfrage an gesendet wird. CloudFront

In der folgenden Tabelle wird erläutert, wie die vom Ursprung gesendeten `Cache-Control`- und `Expires`-Header mit den TTL-Einstellungen in einem Cache-Verhalten zusammenarbeiten, um das Caching zu beeinflussen.

Urspung-Header	Mindest-TTL = 0	Mindest-TTL > 0
Der Ursprung fügt dem Objekt eine **`Cache-Control: max-age`** -Richtlinie hinzu	CloudFront Zwischens peichern CloudFront speichert das Objekt für den kleineren Wert der `Cache-Control: max-age` Direktive oder den Wert der CloudFront maximalen TTL im Cache. Browser-Caching Browser speichern das Objekt für den Wert der `Cache-Control: max-age` -Richtlinie zwischen.	CloudFront Zwischens peichern CloudFront Das Zwischens peichern hängt von den Werten der CloudFront minimalen TTL und der maximalen TTL und der Direktive ab: `Cache-Control max-age` • Wenn minimale TTL < `max-age` < maximale TTL ist, wird das Objekt für den CloudFront Wert der Direktive zwischeng espeichert. `Cache-Control: max-age`

Ursprung-Header	Mindest-TTL = 0	Mindest-TTL > 0
		• Bei `max-age` < minimaler TTL wird das Objekt für den Wert der minimalen TTL CloudFront zwischeng espeichert. CloudFront • Wenn `max-age` > maximale TTL, wird das Objekt für den Wert der maximalen TTL CloudFront zwischeng espeichert. CloudFront Browser-Caching Browser speichern das Objekt für den Wert der `Cache-Con trol: max-age` -Richtlinie zwischen.
Der Ursprung fügt dem Objekt keine **`Cache-Control: max-age`**-Richtlinie hinzu	CloudFront Zwischens peichern CloudFront speichert das Objekt für den Wert der CloudFront Standard-TTL im Cache. Browser-Caching Abhängig vom Browser.	CloudFront Zwischens peichern CloudFront zwischenspeichert das Objekt für den Wert der CloudFront minimalen TTL oder der Standard-TTL, je nachdem, welcher Wert größer ist. Browser-Caching Abhängig vom Browser.

Ursprung-Header	Mindest-TTL = 0	Mindest-TTL > 0
Der Ursprung fügt dem Objekt **`Cache-Control: max-age`** - und **`Cache-Control: s-maxage`** -Richtlinien hinzu	CloudFront Zwischenspeichern CloudFront speichert das Objekt für den kleineren Wert der `Cache-Control: s-maxage` Direktive oder den Wert der CloudFront maximalen TTL im Cache. Browser-Caching Browser speichern das Objekt für den Wert der `Cache-Control max-age` -Richtlinie zwischen.	CloudFront Zwischenspeichern CloudFront Das Zwischenspeichern hängt von den Werten der CloudFront minimalen TTL und der maximalen TTL und der Direktive ab: `Cache-Control: s-maxage` • Wenn minimale TTL < `s-maxage` < maximale TTL ist, wird das Objekt für den CloudFront Wert der Direktive zwischengespeichert. `Cache-Control: s-maxage` • Bei `s-maxage` < minimaler TTL wird das Objekt für den Wert der minimalen TTL CloudFront zwischengespeichert. CloudFront • Wenn `s-maxage` > maximale TTL, wird das Objekt für den Wert der maximalen TTL CloudFront zwischengespeichert. CloudFront Browser-Caching

Ursprung-Header	Mindest-TTL = 0	Mindest-TTL > 0
		Browser speichern das Objekt für den Wert der `Cache-Control: max-age` -Richtlinie zwischen.

Ursprung-Header	Mindest-TTL = 0	Mindest-TTL > 0
Der Ursprung fügt dem Objekt einen **`Expires`**-Header hinzu	CloudFront Zwischens peichern CloudFront zwischenspeichert das Objekt bis zu dem Datum in der `Expires` Kopfzeile oder bis zum Wert der CloudFront maximalen TTL, je nachdem, was früher eintritt. Browser-Caching Browser speichern das Objekt bis zum Datum im `Expires`-Header zwischen.	CloudFront Zwischens peichern CloudFront Das Zwischens peichern hängt von den Werten der CloudFront minimalen TTL und der maximalen TTL und dem Header ab: `Expires` • Wenn minimale TTL < `Expires` < maximale TTL ist, wird das Objekt bis zum Datum und der Uhrzeit im Header CloudFron t zwischengespeichert. `Expires` • Bei `Expires` < minimaler TTL wird das Objekt für den Wert der minimalen TTL CloudFront zwischeng espeichert. CloudFront • Wenn `Expires` > maximale TTL, wird das Objekt für den Wert der maximalen TTL CloudFront zwischeng espeichert. CloudFront Browser-Caching

Ursprung-Header	Mindest-TTL = 0	Mindest-TTL > 0
		Browser speichern das Objekt bis zum Datum und der Uhrzeit im `Expires`-Header zwischen.
Der Ursprung fügt Objekten **`Cache-Control: no-cache`**-, **`no-store`**-, und/oder **`private`**-Richtlinien hinzu	CloudFront und Browser respektieren die Header.	CloudFront Zwischens peichern CloudFront speichert das Objekt für den Wert der CloudFront minimalen TTL im Cache. Siehe die Warnung unter dieser Tabelle. Browser-Caching Browser berücksichtigen die Header.

Warning

Wenn Ihre Mindest-TTL größer als 0 ist, wird die Mindest-TTL der Cache-Richtlinie CloudFront verwendet, auch wenn die `private` Direktiven, `Cache-Control: no-cacheno-store`, und/oder in den Origin-Headern vorhanden sind.
Wenn der Ursprung erreichbar ist, CloudFront ruft das Objekt vom Ursprung ab und gibt es an den Viewer zurück.
Wenn der Ursprung nicht erreichbar ist und der minimale oder maximale TTL-Wert größer als 0 ist, CloudFront wird das Objekt angezeigt, das es zuvor vom Ursprung erhalten hat.
Um dieses Verhalten zu vermeiden, schließen Sie die `Cache-Control: stale-if-error=0`-Richtlinie in das vom Ursprung zurückgegebene Objekt ein. Dies führt CloudFront dazu, dass als Antwort auf future Anfragen ein Fehler zurückgegeben wird, wenn der Ursprung nicht erreichbar ist, anstatt das Objekt zurückzugeben, das es zuvor vom Ursprung erhalten hat.

Informationen zum Ändern der Einstellungen für Distributionen mithilfe der CloudFront Konsole finden Sie unter. Eine Verteilung aktualisieren Informationen zum Ändern der Einstellungen für Distributionen mithilfe der CloudFront API finden Sie unter. UpdateDistribution

Fügen Sie mithilfe der Amazon S3 S3-Konsole Header zu Ihren Objekten hinzu

Sie können das `Expires` Header-Feld `Cache-Control` oder zu Ihren Amazon S3 S3-Objekten hinzufügen. Dazu ändern Sie die Metadatenfelder für das Objekt.

So fügen Sie Amazon S3 S3-Objekten ein **`Cache-ControlExpires`** Oder-Header-Feld hinzu

1. Folgen Sie den Anweisungen im Abschnitt Ersetzen systemdefinierter Metadaten im Thema Objektmetadaten in der Amazon S3 S3-Konsole bearbeiten im Amazon S3 S3-Benutzerhandbuch.
2. Wählen Sie für Schlüssel den Namen des Headers aus, den Sie hinzufügen (Cache-Control oder Expires).
3. Geben Sie für Wert einen Header-Wert ein. Zum Beispiel könnten Sie für einen `Cache-Control`-Header `max-age=86400` eingeben. Für `Expires` könnten Sie ein Ablaufdatum und eine Uhrzeit wie `Wed, 30 Jun 2021 09:28:00 GMT` eingeben.
4. Folgen Sie den restlichen Schritten, um Ihre Metadatenänderungen zu speichern.

Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern

Einige Webanwendungen verwenden zum Senden von Informationen an den Ursprung Abfragezeichenfolgen. Eine Abfragezeichenfolge ist der Teil einer Webanfrage nach dem Zeichen `?`. Die Zeichenfolge kann einen oder mehrere durch das Zeichen & getrennte Parameter enthalten. Im folgenden Beispiel umfasst die Abfragezeichenfolge zwei Parameter *`color=red`* und*`size=large`*:

`https://d111111abcdef8.cloudfront.net/images/image.jpg?`*`color=red&size=large`*

Bei Verteilungen können Sie wählen, ob Sie Abfragezeichenfolgen CloudFront an Ihren Ursprung weiterleiten möchten und ob Ihre Inhalte auf der Grundlage aller Parameter oder anhand ausgewählter Parameter zwischengespeichert werden sollen. Warum kann dies sinnvoll sein? Betrachten Sie das folgende Beispiel.

Angenommen, Ihre Website ist in fünf Sprachen verfügbar. Die Verzeichnisstruktur und Dateinamen für alle fünf Versionen der Website sind identisch. Wenn ein Benutzer Ihre Website aufruft, CloudFront enthalten Anfragen, die weitergeleitet werden, einen Sprachabfragezeichenfolgenparameter, der auf der vom Benutzer ausgewählten Sprache basiert. Sie können so konfigurieren CloudFront , dass Abfragezeichenfolgen an den Ursprung weitergeleitet und auf der Grundlage des Sprachparameters zwischengespeichert werden. Wenn Sie Ihren Webserver so konfigurieren, dass er die Version einer bestimmten Seite zurückgibt, die der ausgewählten Sprache entspricht, CloudFront speichert er jede Sprachversion separat, basierend auf dem Wert des Parameters für die Sprachabfrage.

Wenn in diesem Beispiel die Hauptseite Ihrer Website lautet, führen die folgenden fünf Anfragen CloudFront dazu`main.html`, dass fünfmal zwischengespeichert wird, und `main.html` zwar einmal für jeden Wert des Zeichenfolgenparameters für die Sprachabfrage:

- `https://d111111abcdef8.cloudfront.net/main.html?language=de`
- `https://d111111abcdef8.cloudfront.net/main.html?language=en`
- `https://d111111abcdef8.cloudfront.net/main.html?language=es`
- `https://d111111abcdef8.cloudfront.net/main.html?language=fr`
- `https://d111111abcdef8.cloudfront.net/main.html?language=jp`

Beachten Sie Folgendes:

- Einige HTTP-Server verarbeiten keine Abfragezeichenfolgeparameter und geben daher nicht unterschiedliche Versionen eines Objekts auf der Grundlage der Parameterwerte zurück. Wenn Sie für diese Ursprünge die Weiterleitung von Abfragezeichenfolgenparametern an den Ursprung konfigurieren CloudFront , werden CloudFront trotzdem basierend auf den Parameterwerten zwischengespeichert, obwohl der Ursprung CloudFront für jeden Parameterwert identische Versionen des Objekts zurückgibt.
- Damit Abfragezeichenfolgeparameter wie im obigen Beispiel beschrieben mit den Sprachen funktionieren, müssen Sie das Zeichen & als Trennzeichen zwischen den Abfragezeichenfolgeparametern verwenden. Wenn Sie ein anderes Trennzeichen verwenden, erhalten Sie möglicherweise unerwartete Ergebnisse, je nachdem, welche Parameter Sie angeben, um sie als Grundlage für das Zwischenspeichern CloudFront zu verwenden, und von der Reihenfolge, in der die Parameter in der Abfragezeichenfolge erscheinen.

Die folgenden Beispiele zeigen, was passiert, wenn Sie ein anderes Trennzeichen verwenden und so konfigurieren CloudFront , dass der Cache nur auf der Grundlage des Parameters zwischengespeichert wird: `color`

- In der folgenden Anfrage werden Ihre Inhalte basierend auf dem Wert des `color` Parameters CloudFront zwischengespeichert, der Wert wird jedoch wie folgt CloudFront interpretiert: *`red;size=large`*

  `https://d111111abcdef8.cloudfront.net/images/image.jpg?`*`color=red;size=large`*
- In der folgenden Anfrage werden Ihre Inhalte CloudFront zwischengespeichert, die Zwischenspeicherung basiert jedoch nicht auf den Parametern der Abfragezeichenfolge. Das liegt daran, dass Sie das Zwischenspeichern CloudFront auf der Grundlage des `color` Parameters konfiguriert haben, die folgende Zeichenfolge jedoch so CloudFront interpretiert haben, dass sie nur einen `size` Parameter enthält, der den Wert hat: *`large;color=red`*

  `https://d111111abcdef8.cloudfront.net/images/image.jpg?`*`size=large;color=red`*

Sie können eine der folgenden Optionen konfigurieren CloudFront :

- Keine Abfragezeichenfolge an den Ursprung weiterleiten. Wenn Sie keine Abfragezeichenfolgen weiterleiten, wird der Cache CloudFront nicht auf der Grundlage von Abfragezeichenfolgenparametern zwischengespeichert.
- Abfragezeichenfolgen an den Ursprung weiterleiten und auf der Grundlage aller Parameter in der Abfragezeichenfolge zwischenspeichern.
- Abfragezeichenfolgen an den Ursprung weiterleiten und auf der Grundlage spezifischer Parameter in der Abfragezeichenfolge zwischenspeichern.

Weitere Informationen finden Sie unter the section called “Optimieren Sie das Caching”.

Themen

Konsolen- und API-Einstellungen für die Weiterleitung und Zwischenspeicherung von Abfragezeichenfolgen

Wenn Sie eine Verteilung in der CloudFront Konsole erstellen, CloudFront konfiguriert die Weiterleitung und das Zwischenspeichern von Abfragezeichenfolgen für Sie auf der Grundlage Ihres Quelltyps. Optional können Sie diese Einstellungen manuell bearbeiten. Weitere Informationen finden Sie in den folgenden Einstellungen imthe section called “Alle Verteilungseinstellungen”:

- the section called “Weiterleitung und Zwischenspeicherung von Abfragezeichenfolgen”
- the section called “Zulassungsliste für Abfragezeichenfolgen”

Informationen zur Konfiguration der Weiterleitung und Zwischenspeicherung von Abfragezeichenfolgen mit der CloudFront API finden Sie unter CachePolicyund OriginRequestPolicyin der Amazon CloudFront API-Referenz.

Optimieren Sie das Caching

Wenn Sie CloudFront den Cache auf der Grundlage von Abfragezeichenfolgenparametern konfigurieren, können Sie die folgenden Schritte ausführen, um die Anzahl der Anfragen zu reduzieren, die an CloudFront Ihren Ursprung weitergeleitet werden. Wenn CloudFront Edge-Standorte Objekte bereitstellen, reduzieren Sie die Belastung Ihres Ursprungsservers und reduzieren die Latenz, da Objekte von Standorten aus bedient werden, die sich näher an Ihren Benutzern befinden.

Zwischenspeichern auf der ausschließlichen Grundlage von Parametern, für die Ihr Ursprung verschiedene Versionen eines Objekts zurückgibt

Für jeden Abfragezeichenfolgenparameter, an den Ihre Webanwendung CloudFront weiterleitetCloudFront, werden Anfragen für jeden Parameterwert an Ihren Ursprung weitergeleitet und für jeden Parameterwert eine separate Version des Objekts zwischengespeichert. Dies gilt auch, wenn Ihr Ursprung unabhängig vom Parameterwert immer dasselbe Objekt zurückgibt. Bei mehreren Parametern multiplizieren sich die Anzahl der Anfragen und die Anzahl der Objekte.

Wir empfehlen Ihnen, den Cache so CloudFront zu konfigurieren, dass er nur auf der Grundlage der Abfragezeichenfolgenparameter zwischengespeichert wird, für die Ihr Origin unterschiedliche Versionen zurückgibt, und dass Sie die Vorteile des Cachings anhand der einzelnen Parameter sorgfältig abwägen. Nehmen wir beispielsweise an, dass Sie eine Einzelhandels-Website

besitzen. Sie haben Bilder einer Jacke in sechs verschiedenen Farben und es gibt die Jacke in zehn verschiedenen Größen. Die Bilder, die Sie von der Jacke haben, zeigen die verschiedenen Farben, jedoch nicht die verschiedenen Größen. Um das Caching zu optimieren, sollten Sie so konfigurieren, CloudFront dass der Cache nur anhand des Farbparameters und nicht anhand des Größenparameters zwischengespeichert wird. Dadurch steigt die Wahrscheinlichkeit, dass eine Anfrage aus dem Cache bearbeitet werden CloudFront kann, was die Leistung verbessert und die Belastung Ihres Ursprungsservers reduziert.

Parameter immer in der gleichen Reihenfolge auflisten

Die Reihenfolge der Parameter in Abfragezeichenfolgen ist wichtig. Im folgenden Beispiel sind die Abfragezeichenfolgen identisch, abgesehen davon, dass die Parameter eine andere Reihenfolge aufweisen. Dies führt CloudFront dazu, dass zwei separate Anfragen für image.jpg an Ihren Ursprung weitergeleitet und zwei separate Versionen des Objekts zwischengespeichert werden:

- `https://d111111abcdef8.cloudfront.net/images/image.jpg?color=red&size=large`
- `https://d111111abcdef8.cloudfront.net/images/image.jpg?size=large&color=red`

Wir empfehlen, Parameternamen immer in der gleichen Reihenfolge aufzulisten, beispielsweise in alphabetischer Reihenfolge.

Immer dieselbe Schreibweise für Parameternamen und -werte verwenden

CloudFront berücksichtigt die Groß-/Kleinschreibung von Parameternamen und -werten beim Zwischenspeichern auf der Grundlage von Abfragezeichenfolgenparametern. Im folgenden Beispiel sind die Abfragezeichenfolgen identisch, abgesehen von der Schreibweise der Parameternamen und -werte. Dies führt CloudFront dazu, dass vier separate Anfragen für image.jpg an Ihren Ursprung weitergeleitet und vier separate Versionen des Objekts zwischengespeichert werden:

- `https://d111111abcdef8.cloudfront.net/images/image.jpg?color=red`
- `https://d111111abcdef8.cloudfront.net/images/image.jpg?color=Red`
- `https://d111111abcdef8.cloudfront.net/images/image.jpg?Color=red`
- `https://d111111abcdef8.cloudfront.net/images/image.jpg?Color=Red`

Wir empfehlen, konsistent dieselbe Schreibweise für Parameternamen und -werte zu verwenden, beispielsweise nur Kleinbuchstaben.

Verwenden Sie keine Parameternamen, die mit signed in Konflikt stehen URLs

Wenn Sie signiert verwenden URLs , um den Zugriff auf Ihre Inhalte einzuschränken (wenn Sie Ihrer Distribution vertrauenswürdige Unterzeichner hinzugefügt haben), CloudFront entfernt die folgenden Abfragezeichenfolge-Parameter, bevor der Rest der URL an Ihren Ursprung weitergeleitet wird:

- `Expires`
- `Key-Pair-Id`
- `Policy`
- `Signature`

Wenn Sie signed verwenden URLs und so konfigurieren möchten, dass Abfragezeichenfolgen CloudFront an Ihren Ursprung weitergeleitet werden, können Ihre eigenen Abfragezeichenfolge-Parameter nicht mit`Expires`, `Key-Pair-Id``Policy`, oder `Signature` benannt werden.

Abfrageparameter und CloudFront -Standardprotokolle abfragen (Zugriffsprotokolle)

Wenn Sie die Protokollierung aktivieren, wird die vollständige URL einschließlich der Parameter für die Abfragezeichenfolge CloudFront protokolliert. Dies gilt unabhängig davon, ob Sie die Konfiguration für die Weiterleitung von Abfragezeichenfolgen an den Ursprung konfiguriert CloudFront haben. Weitere Hinweise zur CloudFront Protokollierung finden Sie unterthe section called “Standardprotokollierung (Zugriffsprotokolle)”.

Auf Cookies basierender Inhalt zwischenspeichern

Standardmäßig werden Cookies bei der Verarbeitung von Anfragen und Antworten oder beim Zwischenspeichern Ihrer Objekte an Edge-Standorten CloudFront nicht berücksichtigt. Wenn zwei Anfragen CloudFront empfangen werden, die bis auf das, was im `Cookie` Header steht, identisch sind, werden die Anfragen standardmäßig als identisch CloudFront behandelt und für beide Anfragen dasselbe Objekt zurückgegeben.

Sie können so konfigurieren CloudFront , dass einige oder alle Cookies in Viewer-Anfragen an Ihren Ursprung weitergeleitet werden und dass separate Versionen Ihrer Objekte auf der Grundlage der weitergeleiteten Cookie-Werte zwischengespeichert werden. Wenn Sie dies tun, CloudFront

verwendet einige oder alle Cookies in Viewer-Anfragen — unabhängig davon, welche Cookies für die Weiterleitung konfiguriert sind —, um ein Objekt im Cache eindeutig zu identifizieren.

Nehmen wir beispielsweise an, dass Anfragen für `locations.html` ein `country`-Cookie enthalten, das entweder den Wert `uk` oder `fr` hat. Wenn Sie so konfigurieren CloudFront , dass Ihre Objekte auf der Grundlage des Werts des Cookies zwischengespeichert werden, CloudFront leitet Anfragen für `locations.html` an den Ursprung weiter und schließt das `country` Cookie und seinen Wert ein. `country` Ihr Ursprung kehrt zurück und CloudFront speichert das Objekt einmal für Anfragen`locations.html`, in denen sich der Wert des `country` Cookies befindet, `uk` und einmal für Anfragen, in denen sich der Wert befindet. `fr`

> **Important**
>
> Amazon S3 und einige HTTP-Server verarbeiten keine Cookies. Konfigurieren Sie nicht CloudFront , dass Cookies an eine Quelle weitergeleitet werden, die keine Cookies verarbeitet oder ihre Antwort nicht anhand von Cookies variiert. Dies kann CloudFront dazu führen, dass mehr Anfragen für dasselbe Objekt an den Ursprung weitergeleitet werden, was die Leistung verlangsamt und die Belastung des Ursprungs erhöht. Wenn Sie das vorherige Beispiel berücksichtigen und Ihr Ursprung das `country` Cookie nicht verarbeitet oder CloudFront unabhängig vom Wert des Cookies immer dieselbe Version von `locations.html` zurückgibt, konfigurieren Sie es nicht so, dass dieses Cookie CloudFront weitergeleitet wird. `country`
>
> Wenn Ihre benutzerdefinierte Herkunft dagegen von einem bestimmten Cookie abhängt oder auf der Grundlage eines Cookies unterschiedliche Antworten sendet, stellen Sie sicher, dass Sie dieses Cookie so konfigurieren, dass dieses Cookie CloudFront an den Ursprung weitergeleitet wird. Andernfalls wird das Cookie CloudFront entfernt, bevor die Anfrage an Ihren Absender weitergeleitet wird.

Zum Konfigurieren der Cookie-Weiterleitung aktualisieren Sie das Cache-Verhalten Ihrer Verteilung. Weitere Informationen über Cache-Verhalten finden Sie unter Einstellungen für das Cache-Verhalten, insbesondere in den Abschnitten Cookies weiterleiten und Zulassungslisten-Cookies.

Sie können jedes Cache-Verhalten so konfigurieren, dass eine der folgenden Aktionen ausgeführt wird:

- Alle Cookies an Ihren Ursprung weiterleiten — CloudFront schließt alle Cookies ein, die der Betrachter sendet, wenn er Anfragen an den Ursprung weiterleitet. Wenn Ihr Absender eine

Antwort zurückgibt, CloudFront speichert er die Antwort unter Verwendung der Cookie-Namen und -Werte in der Viewer-Anfrage im Cache. Wenn die ursprüngliche Antwort `Set-Cookie` Header enthält, werden diese zusammen mit dem angeforderten Objekt an den Betrachter CloudFront zurückgegeben. CloudFront speichert auch die `Set-Cookie` Header mit dem vom Ursprung zurückgegebenen Objekt im Cache und sendet diese `Set-Cookie` Header bei allen Cache-Treffern an die Betrachter.

- Leiten Sie eine von Ihnen angegebene Gruppe von Cookies weiter — CloudFront entfernt alle Cookies, die der Betrachter sendet und die nicht auf der Zulassungsliste stehen, bevor eine Anfrage an den Ursprung weitergeleitet wird. CloudFront speichert die Antwort unter Verwendung der Namen und Werte der Cookies, die in der Viewer-Anfrage aufgeführt sind. Wenn die ursprüngliche Antwort `Set-Cookie` Header enthält, werden diese zusammen mit dem angeforderten Objekt an den Viewer CloudFront zurückgegeben. CloudFront speichert auch die `Set-Cookie` Header mit dem vom Ursprung zurückgegebenen Objekt im Cache und sendet diese `Set-Cookie` Header bei allen Cache-Treffern an die Betrachter.

  Weitere Informationen zum Angeben von Platzhaltern in Cookie-Namen finden Sie unter Zulassungslisten-Cookies.

  Informationen zum aktuellen Kontingent für die Anzahl von Cookie-Namen, die Sie für jedes Cache-Verhalten weiterleiten können, oder zum Anfordern eines höheren Kontingents finden Sie unter Kontingente für Abfragezeichenfolgen (Legacy-Cache-Einstellungen).

- Leiten Sie keine Cookies an Ihren Ursprung weiter — Ihre Objekte werden CloudFront nicht auf der Grundlage eines vom Betrachter gesendeten Cookies zwischengespeichert. Außerdem CloudFront werden Cookies entfernt, bevor Anfragen an Ihren Absender weitergeleitet werden, und entfernt `Set-Cookie` Header aus Antworten, bevor Antworten an Ihre Zuschauer zurückgegeben werden. Da dies keine optimale Art ist, deine ursprünglichen Ressourcen zu nutzen, solltest du bei der Auswahl dieses Cache-Verhaltens sicherstellen, dass deine Herkunft standardmäßig keine Cookies in den ursprünglichen Antworten enthält.

Beachten Sie die folgenden Informationen zur Angabe des Cookies, das Sie weiterleiten möchten:

Zugriffsprotokolle

Wenn Sie das Protokollieren von Anfragen und das CloudFront Protokollieren von Cookies konfigurieren CloudFront , werden alle Cookies und alle Cookie-Attribute protokolliert, auch wenn Sie so konfigurieren, dass Cookies CloudFront nicht an Ihren Ursprung weitergeleitet werden oder wenn Sie so konfigurieren CloudFront , dass nur bestimmte Cookies weitergeleitet werden.

Weitere Informationen zur CloudFront Protokollierung finden Sie unter[Standardprotokollierung (Zugriffsprotokolle)](#).

Groß-/Kleinschreibung

Bei Cookie-Namen und -Werten muss die Groß-/Kleinschreibung beachtet werden. Wenn beispielsweise so konfiguriert CloudFront ist, dass alle Cookies weitergeleitet werden und zwei Viewer-Anfragen für dasselbe Objekt Cookies enthalten, die bis auf die Groß- und Kleinschreibung identisch sind, wird das Objekt zweimal CloudFront zwischengespeichert.

CloudFront sortiert Cookies

Wenn CloudFront es so konfiguriert ist, dass Cookies (alle oder eine Teilmenge) weitergeleitet werden, CloudFront sortiert die Cookies in natürlicher Reihenfolge nach dem Namen des Cookies, bevor die Anfrage an Ihren Ursprung weitergeleitet wird.

> **Note**
>
> Cookie-Namen, die mit dem $ Zeichen beginnen, werden nicht unterstützt. CloudFront entfernt das Cookie, bevor die Anfrage an den Ursprung weitergeleitet wird. Sie können das $ Zeichen entfernen oder ein anderes Zeichen am Anfang des Cookie-Namens angeben.

`If-Modified-Since` und `If-None-Match`

`If-Modified-Since`und `If-None-Match` bedingte Anfragen werden nicht unterstützt, wenn die Konfiguration so konfiguriert CloudFront ist, dass Cookies (alle oder ein Teil davon) weitergeleitet werden.

Standard-Name-Wert-Paar-Format erforderlich

CloudFront leitet einen Cookie-Header nur weiter, wenn der Wert dem [Standardformat für Name-Wert-Paare entspricht, zum Beispiel:](#) `"Cookie: cookie1=value1; cookie2=value2"`

Deaktivieren der Zwischenspeicherung von `Set-Cookie`-Headern

Wenn CloudFront es so konfiguriert ist, dass es Cookies an den Ursprung weiterleitet (unabhängig davon, ob es sich um alle oder um bestimmte Cookies handelt), speichert es auch die in der ursprünglichen Antwort empfangenen `Set-Cookie` Header im Cache. CloudFront schließt diese `Set-Cookie` Header in die Antwort an den ursprünglichen Betrachter ein und schließt sie auch in nachfolgende Antworten ein, die aus dem Cache bereitgestellt werden. CloudFront

Wenn Sie Cookies an Ihrem Ursprung empfangen möchten, aber die `Set-Cookie` Header in den Antworten Ihres Ursprungs nicht zwischenspeichern CloudFront möchten, konfigurieren Sie Ihren Ursprung so, dass ein `Cache-Control` Header mit einer `no-cache` Direktive hinzugefügt wird, die `Set-Cookie` als Feldname spezifiziert wird. Beispiel: `Cache-Control: no-cache="Set-Cookie"`. Weitere Informationen finden Sie unter Response Cache-Control-Direktiven im Hypertext Transfer Protocol (HTTP/1.1): Caching Standard.

Maximallänge von Cookie-Namen

Wenn Sie so konfigurieren CloudFront , dass bestimmte Cookies an Ihren Ursprung weitergeleitet werden, darf die Gesamtzahl der Byte in allen Cookie-Namen, die Sie für die Weiterleitung konfigurieren CloudFront , 512 nicht überschreiten, abzüglich der Anzahl der Cookies, die Sie weiterleiten. Wenn Sie beispielsweise so konfigurieren, CloudFront dass 10 Cookies an Ihren Ursprung weitergeleitet werden, darf die kombinierte Länge der Namen der 10 Cookies 502 Byte (512 — 10) nicht überschreiten.

Wenn Sie so konfigurieren CloudFront , dass alle Cookies an Ihren Ursprung weitergeleitet werden, spielt die Länge der Cookie-Namen keine Rolle.

Informationen zur Verwendung der CloudFront Konsole zum Aktualisieren einer Distribution, sodass Cookies an den Ursprung CloudFront weitergeleitet werden, finden Sie unterEine Verteilung aktualisieren. Informationen zur Verwendung der CloudFront API zur Aktualisierung einer Distribution finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.

Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern

CloudFront lässt Sie wählen, ob Sie Header CloudFront an Ihren Ursprung weiterleiten und separate Versionen eines angegebenen Objekts auf der Grundlage der Header-Werte in Viewer-Anfragen zwischenspeichern möchten. Auf diese Weise können Sie auf der Grundlage des Geräts, das der Benutzers verwendet, des Standorts des Viewers, der Spracheinstellung des Viewers und einer Vielzahl anderer Kriterien unterschiedliche Versionen Ihrer Inhalte bereitstellen.

Themen

- Header und Verteilungen – Übersicht
- Wählen Sie die Header aus, auf denen das Caching basieren soll
- Konfigurieren Sie so CloudFront , dass die CORS-Einstellungen respektiert werden

- Konfigurieren Sie das Caching auf der Grundlage des Gerätetyps
- Konfigurieren Sie das Caching basierend auf der Sprache des Betrachters
- Konfigurieren Sie das Caching auf der Grundlage des Standorts des Betrachters
- Konfigurieren Sie das Caching auf der Grundlage des Protokolls der Anfrage
- Konfigurieren Sie das Caching für komprimierte Dateien
- Auswirkungen der Zwischenspeicherung auf der Grundlage von Headern auf die Leistung
- Auswirkungen der Schreibweise von Headern und Header-Werten auf die Zwischenspeicherung
- Header, die zum CloudFront Viewer zurückkehren

Header und Verteilungen – Übersicht

Standardmäßig werden Header CloudFront nicht berücksichtigt, wenn Sie Ihre Objekte an Randpositionen zwischenspeichern. Wenn Ihr Origin zwei Objekte zurückgibt und diese sich nur durch die Werte in den Anforderungsheadern unterscheiden, wird nur eine Version des Objekts CloudFront zwischengespeichert.

Sie können so konfigurieren CloudFront , dass Header an den Ursprung weitergeleitet werden, was CloudFront dazu führt, dass mehrere Versionen eines Objekts auf der Grundlage der Werte in einem oder mehreren Anforderungsheadern zwischengespeichert werden. Um CloudFront zu konfigurieren, dass Objekte auf der Grundlage der Werte bestimmter Header zwischengespeichert werden, geben Sie die Einstellungen für das Cache-Verhalten für Ihre Distribution an. Weitere Informationen finden Sie unter Cache-Speicherung basierend auf ausgewählten Anforderungs-Headern.

Nehmen wir beispielsweise an, dass Viewer-Anforderungen für `logo.jpg` einen benutzerdefinierten `Product`-Header enthalten, der entweder den Wert `Acme` oder `Apex` hat. Wenn Sie so konfigurieren CloudFront , dass Ihre Objekte auf der Grundlage des Werts des `Product` Headers zwischengespeichert werden, werden Anfragen für `logo.jpg` an den Ursprung CloudFront weitergeleitet und die `Product` Header- und Header-Werte eingeschlossen. CloudFront zwischenspeichert `logo.jpg` einmal für Anfragen, in denen sich der Wert des `Product` Headers befindet, `Acme` und einmal für Anfragen, in denen sich der Wert befindet. `Apex`

Sie können jedes Cache-Verhalten in einer Verteilung so konfigurieren, dass eine der folgenden Aktionen ausgeführt wird:

- Weiterleiten aller Header an Ihren Ursprung

> **Note**
>
> Für ältere Cache-Einstellungen — Wenn Sie so konfigurieren CloudFront , dass alle Header an Ihren Ursprung weitergeleitet werden, werden die mit diesem Cache-Verhalten verknüpften Objekte CloudFront nicht zwischengespeichert. Stattdessen werden alle Anfragen an den Ursprung gesendet.

- Leitet eine Liste von Headern weiter, die Sie angeben. CloudFront speichert Ihre Objekte auf der Grundlage der Werte in allen angegebenen Headern im Cache. CloudFront leitet auch die Header weiter, die standardmäßig weitergeleitet werden, aber Ihre Objekte werden nur auf der Grundlage der von Ihnen angegebenen Header zwischengespeichert.
- Weiterleiten nur der Standard-Header. In dieser Konfiguration werden Ihre Objekte CloudFront nicht auf der Grundlage der Werte in den Anforderungsheadern zwischengespeichert.

Informationen zum aktuellen Kontingent für die Anzahl von Headern, die Sie für jedes Cache-Verhalten weiterleiten können, oder zum Anfordern eines höheren Kontingents finden Sie unter Kontingente für Header.

Hinweise zur Verwendung der CloudFront Konsole zur Aktualisierung einer Distribution, bei der Header an den Ursprung CloudFront weitergeleitet werden, finden Sie unter. Eine Verteilung aktualisieren Informationen zur Verwendung der CloudFront API zur Aktualisierung einer bestehenden Distribution finden Sie unter Distribution aktualisieren in der Amazon CloudFront API-Referenz.

Wählen Sie die Header aus, auf denen das Caching basieren soll

Die Header, die Sie an den Ursprung weiterleiten können und auf denen das Caching CloudFront basiert, hängen davon ab, ob es sich bei Ihrem Ursprung um einen Amazon S3 S3-Bucket oder einen benutzerdefinierten Ursprung handelt.

- Amazon S3 — Sie können so konfigurieren CloudFront , dass Ihre Objekte auf der Grundlage bestimmter Header weitergeleitet und zwischengespeichert werden (siehe die folgende Ausnahmeliste). Allerdings empfehlen wir, möglichst keine Header mit einem Amazon-S3-Ursprung weiterzuleiten, es sei denn, Sie implementieren Cross-Origin Resource Sharing (CORS, ursprungsübergreifende gemeinsame Nutzung von Ressourcen) oder möchten Inhalte mit Lambda@Edge in Ereignissen auf Ursprungsseite personalisieren.

  - Um CORS zu konfigurieren, müssen Sie Header weiterleiten, die die Verteilung von Inhalten für Websites CloudFront ermöglichen, die für Cross-Origin Resource Sharing (CORS) aktiviert sind. Weitere Informationen finden Sie unter Konfigurieren Sie so CloudFront , dass die CORS-Einstellungen respektiert werden.
  - Um Inhalte mithilfe von Headern zu personalisieren, die Sie an Ihren Amazon S3 S3-Ursprung weiterleiten, schreiben Sie Lambda @Edge -Funktionen, fügen sie hinzu und verknüpfen sie mit Ihrer CloudFront Verteilung, die durch ein Ereignis ausgelöst wird, das auf den Ursprung gerichtet ist. Weitere Informationen zum Verwenden von Headern, um Inhalte zu personalisieren, finden Sie unter Personalisieren von Inhalten nach Land oder Gerätetyp-Header – Beispiele.

    Wir empfehlen, möglichst keine Header weiterzuleiten, die Sie nicht zum Personalisieren von Inhalten verwenden, da das Weiterleiten zusätzlicher Header zu einer Verringerung der Cache-Trefferrate führen kann. Das heißt, es CloudFront können nicht so viele Anfragen aus Edge-Caches bearbeitet werden, als der Anteil aller Anfragen.
- Benutzerdefinierter Ursprung — Sie können so konfigurierenCloudFront , dass der Cache auf der Grundlage des Werts eines beliebigen Anforderungsheaders gespeichert wird, mit Ausnahme der folgenden:
  - `Connection`
  - `Cookie` – Wenn die Weiterleitung und Zwischenspeicherung auf der Grundlage von Cookies erfolgen soll, verwenden Sie eine separate Einstellung in Ihrer Verteilung. Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.
  - `Host (for Amazon S3 origins)`
  - `Proxy-Authorization`
  - `TE`
  - `Upgrade`

  Sie können so konfigurieren CloudFront , dass Objekte auf der Grundlage von Werten in den `User-Agent` Headern `Date` und zwischengespeichert werden. Dies wird jedoch nicht empfohlen. Diese Header haben zahlreiche mögliche Werte, und das Zwischenspeichern auf der Grundlage ihrer Werte könnte dazu führen, dass deutlich mehr Anfragen CloudFront an Ihren Ursprung weitergeleitet werden.

Eine vollständige Liste der HTTP-Anforderungsheader und deren Verarbeitung finden Sie CloudFront unter. Header und CloudFront Verhalten von HTTP-Anfragen (benutzerdefiniert und Amazon S3 S3-Ursprünge)

Konfigurieren Sie so CloudFront , dass die CORS-Einstellungen respektiert werden

Wenn Sie Cross-Origin Resource Sharing (CORS) auf einem Amazon S3-Bucket oder einem benutzerdefinierten Ursprung aktiviert haben, müssen Sie bestimmte Header zum Weiterleiten auswählen, um die CORS-Einstellungen zu berücksichtigen. Die Header, die Sie weiterleiten müssen, sind abhängig vom Ursprung (Amazon S3 oder benutzerdefiniert) und von der Tatsache, ob Sie `OPTIONS`-Antworten zwischenspeichern möchten, unterschiedlich.

Amazon S3

- Wenn Sie möchten, dass `OPTIONS`-Antworten im Cache gespeichert werden, führen Sie die folgenden Schritte aus:
  - Wählen Sie die Optionen für die Standardeinstellungen für das Cache-Verhalten aus, mit denen das Zwischenspeichern von `OPTIONS`-Antworten aktiviert wird.
  - Konfigurieren Sie CloudFront die Konfiguration so, dass die folgenden Header weitergeleitet werden: `OriginAccess-Control-Request-Headers`, und. `Access-Control-Request-Method`
- Wenn Sie nicht möchten, dass `OPTIONS`-Antworten im Cache gespeichert werden, konfigurieren Sie CloudFront so, dass der `Origin`-Header zusammen mit allen Headern, die für Ihren Ursprung erforderlich sind, weitergeleitet wird (z. B. `Access-Control-Request-Headers`, `Access-Control-Request-Method` oder andere).

Benutzerdefinierte Ursprünge – Leiten Sie den `Origin`-Header zusammen mit anderen Headern weiter, die für Ihren Ursprung erforderlich sind.

Um CloudFront zu konfigurieren, dass Antworten auf der Grundlage von CORS zwischengespeichert werden, müssen Sie die Weiterleitung von Headern mithilfe einer Cache-Richtlinie konfigurieren CloudFront . Weitere Informationen finden Sie unter Steuern Sie den Cache-Schlüssel mit einer Richtlinie.

Weitere Informationen zu CORS und Amazon S3 finden Sie unter Cross-Origin Resource Sharing (CORS) verwenden im Benutzerhandbuch zu Amazon Simple Storage Service.

Konfigurieren Sie das Caching auf der Grundlage des Gerätetyps

Wenn Sie je CloudFront nach dem Gerät, das ein Benutzer zum Ansehen Ihrer Inhalte verwendet, verschiedene Versionen Ihrer Objekte zwischenspeichern möchten, konfigurieren Sie die

Konfiguration so, CloudFront dass die entsprechenden Header an Ihren benutzerdefinierten Ursprung weitergeleitet werden:

- `CloudFront-Is-Desktop-Viewer`
- `CloudFront-Is-Mobile-Viewer`
- `CloudFront-Is-SmartTV-Viewer`
- `CloudFront-Is-Tablet-Viewer`

CloudFront Legt basierend auf dem Wert des `User-Agent` Headers den Wert dieser Header auf `true` oder `false` vor der Weiterleitung der Anfrage an Ihren Ursprung fest. Wenn ein Gerät in mehr als eine Kategorie fällt, können mehrere Werte sei `true`. Beispielsweise CloudFront könnte bei einigen Tablet-Geräten sowohl als auch `CloudFront-Is-Mobile-Viewer` auf festgelegt `CloudFront-Is-Tablet-Viewer` werden`true`.

Konfigurieren Sie das Caching basierend auf der Sprache des Betrachters

Wenn Sie verschiedene Versionen Ihrer Objekte basierend auf der in der Anfrage angegebenen Sprache zwischenspeichern möchten CloudFront , konfigurieren Sie die Konfiguration so, CloudFront dass der `Accept-Language` Header an Ihren Ursprung weitergeleitet wird.

Konfigurieren Sie das Caching auf der Grundlage des Standorts des Betrachters

Wenn Sie je CloudFront nach Land, aus dem die Anfrage kam, unterschiedliche Versionen Ihrer Objekte zwischenspeichern möchten, konfigurieren Sie die Konfiguration so CloudFront , dass der `CloudFront-Viewer-Country` Header an Ihren Ursprung weitergeleitet wird. CloudFront konvertiert automatisch die IP-Adresse, von der die Anfrage kam, in einen aus zwei Buchstaben bestehenden Ländercode. Eine easy-to-use Liste der Ländercodes, sortierbar nach Code und Ländernamen, finden Sie im Wikipedia-Eintrag ISO 3166-1 Alpha-2.

Konfigurieren Sie das Caching auf der Grundlage des Protokolls der Anfrage

Wenn Sie verschiedene Versionen Ihrer Objekte basierend auf dem Protokoll der Anfrage, HTTP oder HTTPS, zwischenspeichern möchten CloudFront , konfigurieren Sie die Konfiguration so, dass der `CloudFront-Forwarded-Proto` Header CloudFront an Ihren Ursprung weitergeleitet wird.

Konfigurieren Sie das Caching für komprimierte Dateien

Wenn Ihr Ursprung die Brotli-Komprimierung unterstützt, können Sie die Zwischenspeicherung basierend auf dem `Accept-Encoding`-Header durchführen. Konfigurieren Sie die Zwischenspeicherung nur dann basierend auf `Accept-Encoding`, wenn Ihr Ursprungs-Server abhängig vom Header unterschiedliche Inhalte bereitstellt.

Auswirkungen der Zwischenspeicherung auf der Grundlage von Headern auf die Leistung

Wenn Sie das Zwischenspeichern CloudFront auf der Grundlage eines oder mehrerer Header konfigurieren und die Header mehr als einen möglichen Wert haben, werden mehr Anfragen CloudFront für dasselbe Objekt an Ihren Ursprungsserver weitergeleitet. Dadurch wird die Leistung verringert und die Last auf Ihrem Ursprungs-Server erhöht. Wenn Ihr Ursprungsserver unabhängig vom Wert eines bestimmten Headers dasselbe Objekt zurückgibt, empfehlen wir Ihnen, die Konfiguration nicht so CloudFront zu konfigurieren, dass der Cache auf der Grundlage dieses Headers zwischengespeichert wird.

Wenn Sie so konfigurieren, CloudFront dass mehr als ein Header weitergeleitet wird, hat die Reihenfolge der Header in Viewer-Anfragen keinen Einfluss auf das Caching, solange die Werte identisch sind. Wenn beispielsweise eine Anfrage die Header A:1, B:2 und eine andere Anfrage B:2, A:1 enthält, wird nur eine Kopie des Objekts zwischengespeichert. CloudFront

Auswirkungen der Schreibweise von Headern und Header-Werten auf die Zwischenspeicherung

Bei CloudFront Caches, die auf Header-Werten basieren, wird die Groß- und Kleinschreibung des Header-Namens nicht berücksichtigt, wohl aber die Groß- und Kleinschreibung des Header-Werts:

- Wenn Viewer-Anfragen `Product:Acme` sowohl als auch enthaltenproduct:Acme, wird ein Objekt nur einmal CloudFront zwischengespeichert. Der einzige Unterschied zwischen ihnen ist die Schreibweise des Header-Namens, die sich nicht auf die Zwischenspeicherung auswirkt.
- Wenn Viewer-Anfragen `Product:Acme` sowohl als auch enthaltenProduct:acme, wird ein Objekt zweimal CloudFront zwischengespeichert, da der Wert `Acme` in einigen Anfragen und `acme` in anderen enthalten ist.

Header, die zum CloudFront Viewer zurückkehren

Die Konfiguration CloudFront für die Weiterleitung und Zwischenspeicherung von Headern hat keinen Einfluss darauf, welche Header zum CloudFront Viewer zurückkehren. CloudFront gibt mit wenigen Ausnahmen alle Header zurück, die es vom Ursprung erhält. Weitere Informationen finden Sie im entsprechenden Thema:

- Amazon S3-Ursprünge – siehe HTTP-Antwort-Header, die CloudFront entfernt oder aktualisiert werden.
- Benutzerdefinierte Ursprünge – Siehe HTTP-Antwort-Header, die CloudFront entfernen oder ersetzen.

Steuern Sie den Cache-Schlüssel mit einer Richtlinie

Mit einer CloudFront Cache-Richtlinie können Sie die HTTP-Header, Cookies und Abfragezeichenfolgen angeben, die im Cache-Schlüssel für Objekte CloudFront enthalten sind, die an CloudFront Edge-Standorten zwischengespeichert werden. Der Cache-Schlüssel ist der eindeutige Bezeichner für jedes Objekt im Cache und bestimmt, ob die HTTP-Anfrage eines Betrachters zu einem Cache-Treffer führt.

Zu einem Cache-Treffer kommt es, wenn eine Viewer-Anforderung denselben Cache-Schlüssel wie eine vorherige Anforderung generiert und das Objekt für diesen Cache-Schlüssel im Cache des Edge-Standorts vorhanden und gültig ist. Bei einem Cache-Treffer wird das Objekt dem Betrachter von einem CloudFront Edge-Standort aus bereitgestellt, was die folgenden Vorteile bietet:

- Geringere Auslastung Ihres Ursprungs-Servers
- Reduzierte Latenz für den Viewer

Weniger Werte im Cache-Schlüssel erhöhen die Wahrscheinlichkeit eines Cache-Treffers. Dadurch können Sie die Leistung Ihrer Website oder Anwendung verbessern, da die Cache-Trefferquote höher ist (ein höherer Anteil von Besucheranfragen, die zu einem Cache-Treffer führen). Weitere Informationen finden Sie unter Den Cache-Schlüssel verstehen.

Um den Cache-Schlüssel zu kontrollieren, verwenden Sie eine CloudFront Cache-Richtlinie. Sie fügen eine Cache-Richtlinie einem oder mehreren Cache-Verhalten in einer CloudFront Verteilung zu.

Sie können die Cache-Richtlinie auch verwenden, um TTL-Einstellungen (Time to Live) für Objekte im CloudFront Cache festzulegen und komprimierte Objekte CloudFront anzufordern und zwischenzuspeichern.

> **Note**
>
> Cache-Einstellungen wirken sich nicht auf gRPC-Anfragen aus, da gRPC-Verkehr nicht zwischengespeichert werden kann. Weitere Informationen finden Sie unter gRPC mit CloudFront Distributionen verwenden.

Themen

Verstehen Sie die Cache-Richtlinien

Sie können eine Cache-Richtlinie verwenden, um das Cache-Trefferverhältnis zu verbessern, indem Sie die Werte (URL-Abfragezeichenfolgen, HTTP-Header und Cookies) kontrollieren, die im Cache-Schlüssel enthalten sind. CloudFront enthält einige vordefinierte Cache-Richtlinien für häufige Anwendungsfälle, die als verwaltete Richtlinien bezeichnet werden. Sie können diese verwalteten Richtlinien verwenden oder eine eigene Cache-Richtlinie erstellen, die speziell auf Ihre Anforderungen zugeschnitten ist. Weitere Informationen zu verwalteten Richtlinien finden Sie unter Verwaltete Cache-Richtlinien verwenden.

Eine Cache-Richtlinie enthält die folgenden Einstellungen, die in Richtlinieninformationen, Einstellungen für Time to Live (TTL, Gültigkeitsdauer der Verbindung) und Cache-Schlüssel-Einstellungen unterteilt sind.

Richtlinieninformationen

Name

Ein Name zur Identifizierung der Cache-Richtlinie. Verwenden Sie den Namen in der Konsole, um die Cache-Richtlinie einem Cache-Verhalten anzufügen.

Beschreibung

Ein Kommentar zur Beschreibung der Cache-Richtlinie. Dies ist optional, kann Ihnen jedoch helfen, den Zweck der Cache-Richtlinie zu identifizieren.

Einstellungen für Time to Live (TTL, Gültigkeitsdauer der Verbindung)

Die Einstellungen für die Gültigkeitsdauer (TTL) bestimmen zusammen mit den Headern `Cache-Control` und den `Expires` HTTP-Headern (sofern sie in der ursprünglichen Antwort enthalten sind), wie lange Objekte im CloudFront Cache gültig bleiben.

Mindest-TTL

Die Mindestdauer in Sekunden, für die Objekte im CloudFront Cache verbleiben sollen, bevor beim Ursprung CloudFront überprüft wird, ob das Objekt aktualisiert wurde. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

> **Warning**
>
> Wenn Ihre Mindest-TTL größer als 0 ist, CloudFront wird Inhalt mindestens für die in der Mindest-TTL der Cache-Richtlinie angegebene Dauer zwischengespeichert, auch wenn die `private` Direktiven `Cache-Control: no-cacheno-store`, oder in den Origin-Headern vorhanden sind.

Höchst-TTL

Die maximale Zeit in Sekunden, für die Objekte im CloudFront Cache verbleiben, bevor beim Ursprung CloudFront überprüft wird, ob das Objekt aktualisiert wurde. CloudFront verwendet diese Einstellung nur, wenn der Ursprung das Objekt sendet `Cache-Control` oder den `Expires` Header mit dem Objekt teilt. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Standard-TTL

Die Standarddauer in Sekunden, für die Objekte im CloudFront Cache bleiben sollen, bevor beim Ursprung CloudFront überprüft wird, ob das Objekt aktualisiert wurde. CloudFront verwendet den Wert dieser Einstellung nur dann als TTL des Objekts, wenn der Ursprung keine `Expires` Kopfzeilen mit dem Objekt sendet`Cache-Control`. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

> **Note**
>
> Wenn die Einstellungen Minimale TTL, Maximale TTL und Standard-TTL alle auf 0 gesetzt sind, wird das Caching deaktiviert. CloudFront

Cache-Schlüssel-Einstellungen

Die Cache-Schlüsseleinstellungen geben die Werte in Viewer-Anfragen an, die im Cache-Schlüssel CloudFront enthalten sind. Bei den Werten kann es sich um URL-Abfragezeichenfolgen, HTTP-Header und Cookies handeln. Die Werte, die Sie in den Cache-Schlüssel einschließen, werden automatisch in Anforderungen aufgenommen, die CloudFront an den Ursprung sendet. Diese werden als Ursprungsanforderungen bezeichnet. Informationen zum Kontrollieren von Ursprungsanforderungen ohne Auswirkungen auf den Cache-Schlüssel finden Sie unter Kontrollieren Sie Herkunftsanfragen mit einer Richtlinie.

Zu den Cache-Schlüssel-Einstellungen gehören:

- Header
- Cookies
- Abfragezeichenfolgen
- Komprimierungsunterstützung

Header

Die HTTP-Header in Viewer-Anfragen, die im Cache-Schlüssel CloudFront enthalten sind, und in Ursprungsanfragen. Sie können für Header eine der folgenden Einstellungen auswählen:

- None (Keine) – Die HTTP-Header in Viewer-Anforderungen sind nicht im Cache-Schlüssel enthalten und werden nicht automatisch in Ursprungsanforderungen eingeschlossen.
- Folgende Header einschließen – Sie geben an, welche der HTTP-Header in Viewer-Anforderungen im Cache-Schlüssel enthalten sind und automatisch in Ursprungsanforderungen eingeschlossen werden.

Wenn Sie die Einstellung Folgende Header einschließen verwenden, geben Sie HTTP-Header nach ihrem Namen und nicht nach ihrem Wert an. Betrachten Sie beispielsweise den folgenden HTTP-Header:

```
Accept-Language: en-US,en;q=0.5
```

In diesem Fall geben Sie den Header als `Accept-Language` an, nicht als `Accept-Language: en-US,en;q=0.5` an. CloudFront enthält jedoch den vollständigen Header, einschließlich seines Wertes, im Cache-Schlüssel und in Ursprungsanforderungen.

Sie können auch bestimmte Header, die von generiert wurden, CloudFront in den Cache-Schlüssel aufnehmen. Weitere Informationen finden Sie unter the section called “ CloudFront Anforderungsheader hinzufügen”.

Cookies

Die Cookies in Viewer-Anfragen, die im Cache-Schlüssel CloudFront enthalten sind, und in Ursprungsanfragen. Für Cookies können Sie eine der folgenden Einstellungen auswählen:

- None (Keine) – Die Cookies in Viewer-Anforderungen sind nicht im Cache-Schlüssel enthalten und werden nicht automatisch in Ursprungsanforderungen aufgenommen.
- All (Alle) – Alle Cookies in Viewer-Anforderungen sind im Cache-Schlüssel enthalten und werden automatisch in Ursprungsanforderungen aufgenommen.
- Angegebene Cookies einschließen – Sie geben an, welche der Cookies in Viewer-Anforderungen im Cache-Schlüssel enthalten sind und automatisch in Ursprungsanforderungen eingeschlossen werden.
- Alle Cookies einschließen außer – Sie geben an, welche der Cookies in Viewer-Anforderungen nicht im Cache-Schlüssel enthalten sind und nicht automatisch in Ursprungsanforderungen enthalten sind. Alle anderen Cookies, mit Ausnahme der von ihnen angegebenen, sind im Cache-Schlüssel und automatisch in Ursprungsanforderungen enthalten.

Wenn Sie die Einstellung Angegebene Cookies einschließen oder Alle Cookies einschließen außer verwenden verwenden, geben Sie Cookies nach ihrem Namen und nicht nach ihrem Wert an. Betrachten Sie beispielsweise den folgenden `Cookie`-Header:

```
Cookie: session_ID=abcd1234
```

In diesem Fall geben Sie das Cookie als `session_ID`, nicht als `session_ID=abcd1234` an. CloudFront Schließt jedoch das vollständige Cookie, einschließlich seines Werts, in den Cache-Schlüssel und in Ursprungsanfragen ein.

Abfragezeichenfolgen

Die URL-Abfragezeichenfolgen in Viewer-Anfragen, die im Cache-Schlüssel CloudFront enthalten sind, und in ursprünglichen Anfragen. Für Abfragezeichenfolgen können Sie eine der folgenden Einstellungen auswählen:

- None (Keine) – Abfragezeichenfolgen in Viewer-Anforderungen sind nicht im Cache-Schlüssel enthalten und sind nicht automatisch in Ursprungsanforderungen enthalten.
- All (Alle) – Alle Abfragezeichenfolgen in Viewer-Anforderungen sind im Cache-Schlüssel enthalten und sind auch automatisch in Ursprungsanforderungen enthalten.

- Angegebene Abfragezeichenfolgen einschließne – Sie geben an, welche der Abfragezeichenfolgen in Viewer-Anforderungen im Cache-Schlüssel enthalten sind und automatisch in Ursprungsanforderungen eingeschlossen werden.
- Alle Abfragezeichenfolgen einschließen außer – Sie geben an, welche der Abfragezeichenfolgen in Viewer-Anforderungen nicht im Cache-Schlüssel enthalten sind und nicht automatisch in Ursprungsanforderungen eingeschlossen werden. Alle anderen Abfragezeichenfolgen außer denen, die Sie angegeben haben, sind im Cache-Schlüssel und automatisch in Ursprungsanforderungen enthalten.

Wenn Sie die Einstellung Angegebene Abfragezeichenfolgen einschließen oder Alle Abfragezeichenfolgen einschließen außer verwenden, geben Sie Abfragezeichenfolgen nach ihrem Namen und nicht nach ihrem Wert an. Betrachten Sie beispielsweise den folgenden URL-Pfad:

```
/content/stories/example-story.html?split-pages=false
```

In diesem Fall geben Sie die Abfragezeichenfolge als `split-pages`, nicht als `split-pages=false` an. CloudFront Schließt jedoch die vollständige Abfragezeichenfolge, einschließlich ihres Werts, in den Cache-Schlüssel und in Ursprungsanfragen ein.

> **Note**
>
> CloudFront Behandelt bei den Einstellungen für den Cacheschlüssel das Sternchen (*) für die Header, Abfragezeichenfolgen und Cookies als wörtliche Zeichenfolge und nicht als Platzhalter.

Komprimierungsunterstützung

Diese Einstellungen ermöglichen CloudFront das Anfordern und Zwischenspeichern von Objekten, die im Gzip- oder Brotli-Komprimierungsformat komprimiert sind, sofern der Viewer dies unterstützt. Diese Einstellungen ermöglichen auch die Funktion der CloudFront -Komprimierung . Viewer geben ihre Unterstützung für diese Komprimierungsformate mit dem `Accept-Encoding`-HTTP-Header an.

> **Note**
>
> Die Webbrowser Chrome und Firefox unterstützen die Brotli-Komprimierung nur, wenn die Anforderung über HTTPS gesendet wird. Diese Browser unterstützen Brotli mit HTTP-Anforderungen nicht.

Aktivieren Sie diese Einstellungen, wenn eine der folgenden Bedingungen zutrifft:

- Ihr Ursprung gibt mit Gzip komprimierte Objekte zurück, wenn Viewer diese unterstützen (Anforderungen enthalten den `Accept-Encoding`-HTTP-Header mit `gzip` als Wert). Verwenden Sie in diesem Fall die Gzip-Einstellung aktiviert (`true`in der CloudFront API `EnableAcceptEncodingGzip` auf, AWS SDKs, AWS CLI oder gesetzt). AWS CloudFormation
- Ihr Ursprung gibt mit Brotli komprimierte Objekte zurück, wenn Viewer diese unterstützen (Anforderungen enthalten den `Accept-Encoding`-HTTP-Header mit `br` als Wert). Verwenden Sie in diesem Fall die Einstellung Brotli enabled (`true`in der CloudFront API `EnableAcceptEncodingBrotli` auf, AWS SDKs AWS CLI, oder gesetzt). AWS CloudFormation
- Die Cache-Verhaltensweise, der diese Cache-Richtlinie angefügt ist, ist mit CloudFront-Komprimierung konfiguriert. In diesem Fall können Sie die Zwischenspeicherung für Gzip oder Brotli oder beides aktivieren. Wenn die CloudFront Komprimierung aktiviert ist, kann die Aktivierung des Zwischenspeichers für beide Formate dazu beitragen, Ihre Kosten für die Datenübertragung ins Internet zu senken.

> **Note**
>
> Wenn Sie das Caching für eines oder beide dieser Komprimierungsformate aktivieren, sollten Sie den `Accept-Encoding` Header nicht in eine Quellanforderungsrichtlinie aufnehmen, die demselben Cache-Verhalten zugeordnet ist. CloudFront bezieht diesen Header immer in ursprüngliche Anfragen ein, wenn das Caching für eines dieser Formate aktiviert ist, sodass die Aufnahme `Accept-Encoding` in eine Richtlinie für ursprüngliche Anfragen keine Auswirkung hat.

Wenn Ihr Ursprungsserver keine mit Gzip oder Brotli komprimierten Objekte zurückgibt oder das Cache-Verhalten nicht mit CloudFront Komprimierung konfiguriert ist, aktivieren Sie das Caching

für komprimierte Objekte nicht. Wenn Sie dies dennoch tun, kann dies zu einer Verringerung der Cache-Trefferquote führen.

Im Folgenden wird erklärt, wie sich diese Einstellungen auf eine Verteilung auswirken. CloudFront In allen folgenden Szenarien wird davon ausgegangen, dass die Viewer-Anforderung den `Accept-Encoding`-Header enthält. Wenn die Viewer-Anfrage den `Accept-Encoding` Header nicht enthält, CloudFront nimmt sie diesen Header nicht in den Cache-Schlüssel auf und nimmt ihn nicht in die entsprechende ursprüngliche Anfrage auf.

Wenn das Zwischenspeichern komprimierter Objekte für beide Komprimierungsformate aktiviert ist

Wenn der Viewer sowohl Gzip als auch Brotli unterstützt — das heißt, wenn sich die `br` Werte `gzip` und beide im `Accept-Encoding` Header der Viewer-Anfrage befinden — wird wie folgt vorgegangen: CloudFront

- Sie normalisiert den Header zu `Accept-Encoding: br,gzip` und schließt den normalisierten Header in den Cache-Schlüssel ein. Der Cache-Schlüssel enthält keine anderen Werte, die sich in dem vom Viewer gesendeten `Accept-Encoding`-Header befanden.
- Wenn der Edge-Standort ein mit Brotli oder Gzip komprimiertes Objekt im Cache enthält, das der Anforderung entspricht und nicht abgelaufen ist, gibt der Edge-Standort das Objekt an den Viewer zurück.
- Wenn der Edge-Standort kein komprimiertes Brotli- oder Gzip-Objekt im Cache hat, das der Anfrage entspricht und nicht abgelaufen ist, nimmt er den normalisierten Header () in die CloudFront entsprechende ursprüngliche Anfrage auf. `Accept-Encoding: br,gzip` Die Ursprungsanforderung enthält keine anderen Werte, die sich in dem vom Viewer gesendeten `Accept-Encoding`-Header befanden.

Wenn der Viewer ein Komprimierungsformat unterstützt, das andere aber nicht — zum Beispiel, wenn es sich um einen Wert im `Accept-Encoding` Header der Viewer-Anfrage `gzip` handelt, dies aber nicht `br` ist —, wird wie folgt vorgegangen: CloudFront

- Sie normalisiert den Header zu `Accept-Encoding: gzip` und schließt den normalisierten Header in den Cache-Schlüssel ein. Der Cache-Schlüssel enthält keine anderen Werte, die sich in dem vom Viewer gesendeten `Accept-Encoding`-Header befanden.
- Wenn der Edge-Standort ein mit Gzip komprimiertes Objekt im Cache enthält, das der Anforderung entspricht und nicht abgelaufen ist, gibt der Edge-Standort das Objekt an den Viewer zurück.

- Wenn der Edge-Standort kein Gzip-komprimiertes Objekt im Cache hat, das der Anfrage entspricht und nicht abgelaufen ist, CloudFront nimmt er den normalisierten Header (`Accept-Encoding: gzip`) in die entsprechende Ursprungsanforderung auf. Die Ursprungsanforderung enthält keine anderen Werte, die sich in dem vom Viewer gesendeten `Accept-Encoding`-Header befanden.

Um zu verstehen, was CloudFront passiert, wenn der Viewer Brotli, aber nicht Gzip unterstützt, ersetzen Sie die beiden Komprimierungsformate im vorherigen Beispiel miteinander.

Falls der Viewer Brotli oder GZip nicht unterstützt — das heißt, der `Accept-Encoding` Header in der Viewer-Anfrage enthält keine Werte oder als Werte —: `br` `gzip` CloudFront

- Schließt den `Accept-Encoding`-Header nicht in den Cache-Schlüssel ein.
- Schließt `Accept-Encoding: identity` in die entsprechende Ursprungsanforderung ein. Die Ursprungsanforderung enthält keine anderen Werte, die sich in dem vom Viewer gesendeten `Accept-Encoding`-Header befanden.

Wenn die Zwischenspeicherung komprimierter Objekte für ein Komprimierungsformat aktiviert ist, aber nicht für das andere

Wenn der Viewer das Format unterstützt, für das das Caching aktiviert ist — zum Beispiel, wenn das Zwischenspeichern komprimierter Objekte für Gzip aktiviert ist und der Viewer Gzip unterstützt (`gzip`ist einer der Werte im Header der Viewer-Anfrage) — geht Folgendes vor: `Accept-Encoding` CloudFront

- Sie normalisiert den Header zu `Accept-Encoding: gzip` und schließt den normalisierten Header in den Cache-Schlüssel ein.
- Wenn der Edge-Standort ein mit Gzip komprimiertes Objekt im Cache enthält, das der Anforderung entspricht und nicht abgelaufen ist, gibt der Edge-Standort das Objekt an den Viewer zurück.
- Wenn der Edge-Standort kein Gzip-komprimiertes Objekt im Cache hat, das der Anfrage entspricht und nicht abgelaufen ist, wird der normalisierte Header () in die entsprechende CloudFront Ursprungsanforderung aufgenommen. `Accept-Encoding: gzip` Die Ursprungsanforderung enthält keine anderen Werte, die sich in dem vom Viewer gesendeten `Accept-Encoding`-Header befanden.

Dieses Verhalten ist identisch, wenn der Viewer sowohl Gzip als auch Brotli unterstützt (der `Accept-Encoding`-Header in der Viewer-Anforderung enthält `gzip` und `br` als Werte), da in diesem Szenario das Zwischenspeichern komprimierter Objekte für Brotli nicht aktiviert ist.

Um zu verstehen, was CloudFront passiert, wenn das Zwischenspeichern komprimierter Objekte für Brotli, aber nicht für Gzip aktiviert ist, ersetzen Sie die beiden Komprimierungsformate im vorherigen Beispiel durch einander.

Falls der Viewer das Komprimierungsformat, für das das Caching aktiviert ist, nicht unterstützt (der `Accept-Encoding` Header in der Viewer-Anfrage enthält nicht den Wert für dieses Format), gilt Folgendes: CloudFront

- Schließt den `Accept-Encoding`-Header nicht in den Cache-Schlüssel ein.
- Schließt `Accept-Encoding: identity` in die entsprechende Ursprungsanforderung ein. Die Ursprungsanforderung enthält keine anderen Werte, die sich in dem vom Viewer gesendeten `Accept-Encoding`-Header befanden.

Wenn das Zwischenspeichern komprimierter Objekte für beide Komprimierungsformate deaktiviert ist

Wenn das Zwischenspeichern komprimierter Objekte für beide Komprimierungsformate deaktiviert ist, wird der `Accept-Encoding` Header genauso CloudFront behandelt wie jeder andere HTTP-Header in der Viewer-Anforderung. Standardmäßig ist dieser nicht im Cache-Schlüssel und nicht in Ursprungsanforderungen enthalten. Sie können diesen wie jeden anderen HTTP-Header in eine Cache-Richtlinie oder eine Ursprungsanforderungsrichtlinie in die Header-Liste einfügen.

Erstellen Sie Cache-Richtlinien

Sie können eine Cache-Richtlinie verwenden, um das Cache-Trefferverhältnis zu verbessern, indem Sie die Werte (URL-Abfragezeichenfolgen, HTTP-Header und Cookies) steuern, die im Cache-Schlüssel enthalten sind. Sie können eine Cache-Richtlinie in der CloudFront Konsole, mit der AWS Command Line Interface (AWS CLI) oder mit der CloudFront API erstellen.

Nachdem Sie eine Cache-Richtlinie erstellt haben, fügen Sie sie einer oder mehreren Cache-Verhaltensweisen in einer CloudFront -Verteilung an.

Console

So erstellen Sie eine Cache-Richtlinie (Konsole):

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Seite Richtlinien in der CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home?#/policies.
2. Wählen Sie Create cache policy (Cache-Richtlinie erstellen).

3. Wählen Sie die gewünschte Einstellung für diese Cache-Richtlinie aus. Weitere Informationen finden Sie unter Verstehen Sie die Cache-Richtlinien.
4. Wenn Sie fertig sind, wählen Sie Erstellen.

Nachdem Sie eine Cache-Richtlinie erstellt haben, können Sie sie an eine Cache-Verhaltensweise anfügen.

So fügen Sie eine Cache-Richtlinie an eine vorhandene Verteilung an (Konsole):

1. Öffnen Sie die Seite Distributions (Verteilungen) in der CloudFront-Konsole unter https://console.aws.amazon.com/cloudfront/v4/home#/distributions.
2. Wählen Sie die Verteilung aus, die Sie aktualisieren möchten, und anschließend die Registerkarte Verhaltensweisen aus.
3. Wählen Sie das Cacheverhalten, das Sie aktualisieren möchten, und anschließend Bearbeiten aus.

   Um ein neues Cacheverhalten zu erstellen, wählen Sie Verhalten erstellen aus.
4. Stellen Sie im Abschnitt Cache-Schlüssel- und Ursprungsanforderungen sicher, dass Cache-Richtlinie und Ursprungsanforderungsrichtlinie ausgewählt sind.
5. Wählen Sie für Cache Policy (Cache-Richtlinie) die Cache-Richtlinie aus, die diesem Cache-Verhalten angefügt werden soll.
6. Wählen Sie unten auf der Seite die Option Änderungen speichern aus.

So fügen Sie eine Cache-Richtlinie an eine neue Verteilung an (Konsole):

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie Verteilung erstellen.
3. Stellen Sie im Abschnitt Cache-Schlüssel- und Ursprungsanforderungen sicher, dass Cache-Richtlinie und Ursprungsanforderungsrichtlinie ausgewählt sind.
4. Wählen Sie unter Cache policy (Cache-Richtlinie) die Cache-Richtlinie aus, die an das Standard-Cacheverhalten dieser Verteilung angefügt werden soll.
5. Wählen Sie die gewünschten Einstellungen für den Ursprung, das Standard-Cacheverhalten und andere Verteilungseinstellungen aus. Weitere Informationen finden Sie unter Referenz für alle Verteilungseinstellungen.

6. Wenn Sie fertig sind, wählen Sie Verteilung erstellen aus.

CLI

Verwenden Sie den aws cloudfront create-cache-policy Befehl, um eine Cache-Richtlinie mit dem AWS Command Line Interface (AWS CLI) zu erstellen. Sie können die Eingabeparameter des Befehls in einer Eingabedatei bereitstellen, anstatt jeden einzelnen Parameter als Befehlszeileneingabe anzugeben.

So erstellen Sie eine Cache-Richtlinie (CLI mit Eingabedatei):

1. Verwenden Sie den folgenden Befehl, um eine Datei mit dem Namen `cache-policy.yaml` zu erstellen, die alle Eingabeparameter für den create-cache-policy-Befehl enthält.

```
aws cloudfront create-cache-policy --generate-cli-skeleton yaml-input > cache-policy.yaml
```

2. Öffnen Sie die Datei mit dem Namen `cache-policy.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, um die gewünschten Cache-Richtlinieneinstellungen anzugeben, und speichern Sie die Datei. Sie können optionale Felder aus der Datei entfernen, erforderliche Felder dürfen jedoch nicht entfernt werden.

   Weitere Informationen zu den Cache-Richtlinieneinstellungen finden Sie unter Verstehen Sie die Cache-Richtlinien.

3. Verwenden Sie den folgenden Befehl, um die Cache-Richtlinie mit Eingabeparametern aus der `cache-policy.yaml`-Datei zu erstellen.

```
aws cloudfront create-cache-policy --cli-input-yaml file://cache-policy.yaml
```

   Notieren Sie den `Id`-Wert in der Ausgabe des Befehls. Dies ist die Cache-Richtlinien-ID, und Sie benötigen sie, um die Cache-Richtlinie an das Cache-Verhalten einer CloudFront Distribution anzuhängen.

So fügen Sie eine Cache-Richtlinie an eine vorhandene Verteilung an (CLI mit Eingabedatei):

1. Verwenden Sie den folgenden Befehl, um die Verteilungskonfiguration für die CloudFront Distribution zu speichern, die Sie aktualisieren möchten. Ersetzen Sie *distribution_ID* durch die ID der Verteilung.

```
aws cloudfront get-distribution-config --id distribution_ID --output yaml >
 dist-config.yaml
```

2. Öffnen Sie die Datei mit dem Namen `dist-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, indem Sie die folgenden Änderungen an jeder Cache-Verhaltensweise vornehmen, die Sie aktualisieren, um eine Cache-Richtlinie zu verwenden.

   - Fügen Sie in der Cache-Verhaltensweise ein Feld mit dem Namen `CachePolicyId` hinzu. Verwenden Sie für den Wert des Feldes die Cache-Richtlinien-ID, die Sie nach dem Erstellen der Richtlinie notiert haben.
   - Entfernen Sie die Felder `MinTTL`, `MaxTTL`, `DefaultTTL` und `ForwardedValues` aus der Cache-Verhaltensweise. Diese Einstellungen werden in der Cache-Richtlinie angegeben, somit können Sie diese Felder und eine Cache-Richtlinie nicht in dasselbe Cache-Verhalten einschließen.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

   Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Verteilung so zu aktualisieren, dass die Cache-Richtlinie verwendet wird. Ersetzen Sie *distribution_ID* durch die ID der Verteilung.

```
aws cloudfront update-distribution --id distribution_ID --cli-input-yaml file://
dist-config.yaml
```

So fügen Sie eine Cache-Richtlinie an eine neue Verteilung an (CLI mit Eingabedatei):

1. Verwenden Sie den folgenden Befehl, um eine Datei mit dem Namen `distribution.yaml` zu erstellen, die alle Eingabeparameter für den create-distribution-Befehl enthält.

```
aws cloudfront create-distribution --generate-cli-skeleton yaml-input >
 distribution.yaml
```

2. Öffnen Sie die Datei mit dem Namen `distribution.yaml`, die Sie gerade erstellt haben. Geben Sie im Standard-Cacheverhalten in das `CachePolicyId`-Feld die Cache-Richtlinien-ID ein, die Sie nach dem Erstellen der Richtlinie notiert haben. Fahren Sie mit der Bearbeitung der Datei fort, um die gewünschten Verteilungseinstellungen anzugeben, und speichern Sie die Datei, wenn Sie fertig sind.

   Weitere Informationen zu den Verteilungseinstellungen finden Sie unter Referenz für alle Verteilungseinstellungen.
3. Verwenden Sie den folgenden Befehl, um die Verteilung mit Eingabeparametern aus der Datei `distribution.yaml` zu erstellen.

   ```
   aws cloudfront create-distribution --cli-input-yaml file://distribution.yaml
   ```

API

Um eine Cache-Richtlinie mit der CloudFront API zu erstellen, verwenden Sie CreateCachePolicy. Weitere Informationen zu den Feldern, die Sie in diesem API-Aufruf angeben, finden Sie in Verstehen Sie die Cache-Richtlinien und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Nachdem Sie eine Cache-Richtlinie erstellt haben, können Sie sie mit einem der folgenden API-Aufrufe an eine Cache-Verhaltensweise anfügen:

- Um es an ein Cache-Verhalten in einer vorhandenen Distribution anzuhängen, verwenden Sie UpdateDistribution.
- Um es an ein Cache-Verhalten in einer neuen Distribution anzuhängen, verwenden Sie CreateDistribution.

Geben Sie für beide API-Aufrufe die ID der Cache-Richtlinie im Feld `CachePolicyId` innerhalb eines Cache-Verhaltens an. Weitere Informationen zu den anderen Feldern, die Sie in diesen API-Aufrufen angeben, finden Referenz für alle Verteilungseinstellungen Sie in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Verwaltete Cache-Richtlinien verwenden

CloudFront bietet eine Reihe verwalteter Cache-Richtlinien, die Sie an jedes Cache-Verhalten Ihrer Distribution anhängen können. Bei Verwendung einer verwalteten Cache-Richtlinie müssen Sie keine eigene Cache-Richtlinie schreiben oder verwalten. Die verwalteten Richtlinien verwenden Einstellungen, die für bestimmte Anwendungsfälle optimiert sind.

Um eine verwaltete Cache-Richtlinie zu verwenden, fügen Sie sie einem Cache-Verhalten in Ihrer Verteilung an. Der Prozess ist der gleiche wie beim Erstellen einer Cache-Richtlinie. Anstatt jedoch eine neue Cache-Richtlinie zu erstellen, fügen Sie einfach eine verwaltete Cache-Richtlinie an. Sie fügen die Richtlinie entweder nach Namen (mit der Konsole) oder nach ID (mit AWS CLI oder SDKs) hinzu. Die Namen und IDs sind im folgenden Abschnitt aufgeführt.

Weitere Informationen finden Sie unter Erstellen Sie Cache-Richtlinien.

In den folgenden Themen werden die verwalteten Richtlinien beschrieben, die Sie verwenden können.

Themen

Amplify

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie wurde für die Verwendung mit einem Ursprung entwickelt, bei dem es sich um eine AWS Amplify-Web-App handelt.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

```
2e54312d-136d-493c-8eb9-b001f22f67d2
```

Diese Richtlinie hat folgende Einstellungen:

- Mindest-TTL: 2 Sekunden
- Höchst-TTL: 600 Sekunden (10 Minuten)
- Standard-TTL: 2 Sekunden
- Im Cache-Schlüssel enthaltene Header:
  - `Authorization`
  - `CloudFront-Viewer-Country`
  - `Host`

  Der normalisierte Header `Accept-Encoding` ist auch enthalten, da die Einstellung für komprimierte Cache-Objekte aktiviert ist. Weitere Informationen finden Sie unter Komprimierungsunterstützung.
- Im Cache-Schlüssel enthaltene Cookies: Alle Cookies sind enthalten.
- Im Cache-Schlüssel enthaltene Abfragezeichenfolgen: Alle Abfragezeichenfolgen sind enthalten.
- Einstellung für komprimierte Cache-Objekte: Aktiviert. Weitere Informationen finden Sie unter Komprimierungsunterstützung.

> **Warning**
>
> Da diese Richtlinie eine Mindest-TTL von mehr als 0 hat, CloudFront wird Inhalt mindestens für die in der Mindest-TTL der Cache-Richtlinie angegebene Dauer zwischengespeichert, auch wenn die `private` Direktiven `Cache-Control: no-cacheno-store`, oder in den ursprünglichen Headern vorhanden sind.

AWS Amplify Cache-Richtlinien hosten

Amplify verwendet die folgenden verwalteten Cache-Richtlinien, um die Standard-Cache-Konfiguration für die Anwendungen der Kunden zu optimieren:

- Verstärker-Standard
- Amplify- DefaultNoCookies
- Amplify- ImageOptimization
- Amplify- StaticContent

> **Note**
>
> Diese Richtlinien werden nur von Amplify verwendet. Wir empfehlen Ihnen nicht, diese Richtlinien für Ihre Distributionen zu verwenden.

Weitere Informationen zur Verwaltung der Cache-Konfiguration für Ihre von Amplify gehostete Anwendung finden Sie unter Verwaltung der Cache-Konfiguration im Amplify Hosting-Benutzerhandbuch.

CachingDisabled

Sehen Sie sich diese Richtlinie in der Konsole an CloudFront

Diese Richtlinie deaktiviert die Zwischenspeicherung. Diese Richtlinie ist für dynamische Inhalte und für Anforderungen nützlich, die nicht zwischengespeichert werden können.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

```
4135ea2d-6df8-44a3-9df3-4b5a84be39ad
```

Diese Richtlinie hat folgende Einstellungen:

- Mindest-TTL: 0 Sekunden
- Höchst-TTL: 0 Sekunden
- Standard-TTL: 0 Sekunden
- Im Cache-Schlüssel enthaltene Header: Keine
- Im Cache-Schlüssel enthaltene Cookies: Keine
- Im Cache-Schlüssel enthaltene Abfragezeichenfolgen: Keine
- Einstellung für komprimierte Cache-Objekte: Deaktiviert

CachingOptimized

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie wurde entwickelt, um die Cache-Effizienz zu optimieren, indem die Werte, die im Cache-Schlüssel CloudFront enthalten sind, minimiert werden. CloudFront enthält

keine Abfragezeichenfolgen oder Cookies im Cache-Schlüssel und bezieht nur den normalisierten `Accept-Encoding` Header ein. Dadurch können CloudFront Objekte in den Komprimierungsformaten Gzip und Brotli separat zwischengespeichert werden, wenn der Ursprung sie zurückgibt oder wenn die Kantenkomprimierung aktiviert ist. CloudFront

Wenn Sie die oder die CloudFront API verwenden AWS CloudFormation AWS CLI, lautet die ID für diese Richtlinie:

`658327ea-f89d-4fab-a63d-7e88639e58f6`

Diese Richtlinie hat folgende Einstellungen:

- Mindest-TTL: 1 Sekunde
- Höchst-TTL: 31.536.000 Sekunden (365 Tage).
- Standard-TTL: 86.400 Sekunden (24 Stunden).
- Im Cache-Schlüssel enthaltene Header: Es werden keine Header explizit eingefügt. Der normalisierte Header `Accept-Encoding` ist enthalten, da die Einstellung für komprimierte Cache-Objekte aktiviert ist. Weitere Informationen finden Sie unter Komprimierungsunterstützung.
- Im Cache-Schlüssel enthaltene Cookies: Keine.
- Im Cache-Schlüssel enthaltene Abfragezeichenfolgen: Keine.
- Einstellung für komprimierte Cache-Objekte: Aktiviert. Weitere Informationen finden Sie unter Komprimierungsunterstützung.

> **Warning**
>
> Da diese Richtlinie eine Mindest-TTL von mehr als 0 hat, CloudFront wird Inhalt mindestens für die in der Mindest-TTL der Cache-Richtlinie angegebene Dauer zwischengespeichert, auch wenn die `private` Direktiven `Cache-Control: no-cacheno-store`, oder in den ursprünglichen Headern vorhanden sind.

CachingOptimizedForUncompressedObjects

Diese Richtlinie in der Konsole anzeigen CloudFront

Diese Richtlinie wurde zur Optimierung der Cache-Effizienz entwickelt, indem die in den Cache-Schlüssel eingefügten Werte minimiert werden. Es sind keine Abfragezeichenfolgen, Header oder

Cookies enthalten. Diese Richtlinie ist identisch mit der vorherigen Richtlinie; die Einstellung für komprimierte Cache-Objekte ist jedoch deaktiviert.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`b2884449-e4de-46a7-ac36-70bc7f1ddd6d`

Diese Richtlinie hat folgende Einstellungen:

- Mindest-TTL: 1 Sekunde
- Höchst-TTL: 31.536.000 Sekunden (365 Tage)
- Standard-TTL: 86.400 Sekunden (24 Stunden)
- Im Cache-Schlüssel enthaltene Header: Keine
- Im Cache-Schlüssel enthaltene Cookies: Keine
- Im Cache-Schlüssel enthaltene Abfragezeichenfolgen: Keine
- Einstellung für komprimierte Cache-Objekte: Deaktiviert

> **Warning**
>
> Da diese Richtlinie eine Mindest-TTL von mehr als 0 hat, CloudFront wird Inhalt mindestens für die in der Mindest-TTL der Cache-Richtlinie angegebene Dauer zwischengespeichert, auch wenn die `private` Direktiven `Cache-Control: no-cache` `no-store`, oder in den ursprünglichen Headern vorhanden sind.

Elementar- MediaPackage

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie wurde für die Verwendung mit einem Ursprung entwickelt, bei dem es sich um einen AWS Elemental MediaPackage -Endpunkt handelt.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`08627262-05a9-4f76-9ded-b50ca2e3a84f`

Diese Richtlinie hat folgende Einstellungen:

- Mindest-TTL: 0 Sekunden
- Höchst-TTL: 31.536.000 Sekunden (365 Tage)
- Standard-TTL: 86.400 Sekunden (24 Stunden)
- Im Cache-Schlüssel enthaltene Header:
  - `Origin`

  Der normalisierte Header `Accept-Encoding` ist enthalten, da die Einstellung für komprimierte Cache-Objekte für Gzip aktiviert ist. Weitere Informationen finden Sie unter Komprimierungsunterstützung.
- Im Cache-Schlüssel enthaltene Cookies: Keine
- Im Cache-Schlüssel enthaltene Abfragezeichenfolgen:
  - `aws.manifestfilter`
  - `start`
  - `end`
  - `m`
- Einstellung für komprimierte Cache-Objekte: Für Gzip aktiviert. Weitere Informationen finden Sie unter Komprimierungsunterstützung.

UseOriginCacheControlHeaders

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie ist für die Verwendung mit einem Ursprung konzipiert, der `Cache-Control` HTTP-Antwortheader zurückgibt und keine unterschiedlichen Inhalte bereitstellt, die auf den Werten in der Abfragezeichenfolge basieren. Wenn Ihr Origin auf der Grundlage von Werten in der Abfragezeichenfolge unterschiedliche Inhalte bereitstellt, sollten Sie die Verwendung vonUseOriginCacheControlHeaders-QueryStrings.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`83da9c7e-98b4-4e11-a168-04f0df8e2c65`

Diese Richtlinie hat folgende Einstellungen:

- Mindest-TTL: 0 Sekunden

- Höchst-TTL: 31.536.000 Sekunden (365 Tage)
- Standard-TTL: 0 Sekunden
- Im Cache-Schlüssel enthaltene Header:
  - `Host`
  - `Origin`
  - `X-HTTP-Method-Override`
  - `X-HTTP-Method`
  - `X-Method-Override`

  Der normalisierte Header `Accept-Encoding` ist auch enthalten, da die Einstellung für komprimierte Cache-Objekte aktiviert ist. Weitere Informationen finden Sie unter Komprimierungsunterstützung.
- Im Cache-Schlüssel enthaltene Cookies: Alle Cookies sind enthalten.
- Im Cache-Schlüssel enthaltene Abfragezeichenfolgen: Keine.
- Einstellung für komprimierte Cache-Objekte: Aktiviert. Weitere Informationen finden Sie unter Komprimierungsunterstützung.

UseOriginCacheControlHeaders-QueryStrings

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie ist für die Verwendung mit einem Ursprung konzipiert, der `Cache-Control` HTTP-Antwortheader zurückgibt und basierend auf den in der Abfragezeichenfolge enthaltenen Werten unterschiedliche Inhalte bereitstellt. Wenn Ihr Origin keine unterschiedlichen Inhalte bereitstellt, die auf den in der Abfragezeichenfolge enthaltenen Werten basieren, sollten Sie die Verwendung vonUseOriginCacheControlHeaders.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`4cc15a8a-d715-48a4-82b8-cc0b614638fe`

Diese Richtlinie hat folgende Einstellungen:

- Mindest-TTL: 0 Sekunden
- Höchst-TTL: 31.536.000 Sekunden (365 Tage)

- Standard-TTL: 0 Sekunden
- Im Cache-Schlüssel enthaltene Header:
  - `Host`
  - `Origin`
  - `X-HTTP-Method-Override`
  - `X-HTTP-Method`
  - `X-Method-Override`

  Der normalisierte Header `Accept-Encoding` ist auch enthalten, da die Einstellung für komprimierte Cache-Objekte aktiviert ist. Weitere Informationen finden Sie unter Komprimierungsunterstützung.
- Im Cache-Schlüssel enthaltene Cookies: Alle Cookies sind enthalten.
- Im Cache-Schlüssel enthaltene Abfragezeichenfolgen: Alle Abfragezeichenfolgen sind enthalten.
- Einstellung für komprimierte Cache-Objekte: Aktiviert. Weitere Informationen finden Sie unter Komprimierungsunterstützung.

Den Cache-Schlüssel verstehen

Der Cache-Schlüssel bestimmt, ob eine Viewer-Anfrage an einen CloudFront Edge-Standort zu einem Cache-Treffer führt. Der Cache-Schlüssel ist der eindeutige Bezeichner für ein Objekt im Cache. Jedes Objekt im Cache verfügt über einen eindeutigen Cache-Schlüssel.

Ein Cache-Treffer tritt auf, wenn eine Viewer-Anforderung denselben Cache-Schlüssel wie eine vorherige Anforderung generiert und sich das Objekt für diesen Cache-Schlüssel im Cache des Edge-Standorts befindet und gültig ist. Bei einem Cache-Treffer wird das angeforderte Objekt dem Betrachter von einem CloudFront Edge-Standort aus zugestellt, was folgende Vorteile bietet:

- Geringere Auslastung Ihres Ursprungs-Servers
- Reduzierte Latenz für den Viewer

Sie können eine bessere Leistung für Ihre Website oder Anwendung erzielen, wenn Sie eine höhere Cache-Trefferquote haben (ein höherer Anteil an Viewer-Anforderungen, die zu einem Cache-Treffer führen). Eine Möglichkeit, die Cache-Trefferquote zu verbessern, besteht darin, nur die minimal notwendigen Werte in den Cache-Schlüssel aufzunehmen. Weitere Informationen finden Sie in den folgenden Abschnitten.

Sie können die Werte (URL-Abfragezeichenfolgen, HTTP-Header und Cookies) im Cache-Schlüssel mithilfe einer Cache-Richtlinie ändern. (Sie können den Cache-Schlüssel auch ändern, indem Sie ein Lambda @Edgefunction oder eine CloudFront Funktion für eine Viewer-Anfrage verwenden.) Bevor Sie den Cache-Schlüssel ändern, ist es wichtig zu verstehen, wie Ihre Anwendung entworfen ist und wann und wie sie verschiedene Antworten basierend auf den Eigenschaften der Viewer-Anforderung bereitstellen kann. Wenn ein Wert in der Viewer-Anforderung die Antwort bestimmt, die Ihr Ursprung zurückgibt, sollten Sie diesen Wert in den Cache-Schlüssel aufnehmen. Wenn Sie jedoch einen Wert in den Cache-Schlüssel einfügen, der sich nicht auf die Antwort auswirkt, die Ihr Ursprung zurückgibt, kann es passieren, dass Sie doppelte Objekte zwischenspeichern.

Standard-Cache-Schlüssel

Standardmäßig enthält der Cache-Schlüssel für eine CloudFront Distribution die folgenden Informationen:

- Der Domainname der CloudFront Distribution (z. B. d111111abcdef8.cloudfront.net)
- Der URL-Pfad des angeforderten Objekts (z. B, `/content/stories/example-story.html`)

> **Note**
>
> Die`OPTIONS`-Methode ist im Cache-Schlüssel für `OPTIONS`-Anforderungen enthalten. Dies bedeutet, dass Antworten auf `OPTIONS`-Anforderungen getrennt von Antworten auf `GET`- und `HEAD`-Anforderungen zwischengespeichert werden.

Andere Werte aus der Viewer-Anforderung sind standardmäßig nicht im Cache-Schlüssel enthalten. Betrachten Sie die folgende HTTP-Anforderung von einem Webbrowser.

```
            GET /content/stories/example-story.html?ref=0123abc&split-pages=false
HTTP/1.1
            Host: d111111abcdef8.cloudfront.net
            User-Agent: Mozilla/5.0 Gecko/20100101 Firefox/68.0
            Accept: text/html,*/*
            Accept-Language: en-US,en
            Cookie: session_id=01234abcd
            Referer: https://news.example.com/
```

Wenn eine Viewer-Anfrage wie in diesem Beispiel an einem CloudFront Edge-Standort eingeht, wird anhand des Cache-Schlüssels CloudFront ermittelt, ob ein Cache-Treffer vorliegt. Standardmäßig sind nur die folgenden Komponenten der Anforderung im Cache-Schlüssel enthalten: `/content/stories/example-story.html` und `d111111abcdef8.cloudfront.net`. Wenn sich das angeforderte Objekt nicht im Cache befindet (ein Cache-Fehler), wird eine Anfrage an den Ursprung CloudFront gesendet, um das Objekt abzurufen. Nachdem das Objekt abgerufen wurde, wird es an den Viewer CloudFront zurückgegeben und im Cache der Edge-Position gespeichert.

When CloudFront empfängt eine weitere Anfrage für dasselbe Objekt, die durch den Cache-Schlüssel bestimmt wird, und CloudFront stellt das zwischengespeicherte Objekt sofort dem Betrachter zur Verfügung, ohne eine Anfrage an den Ursprung zu senden. Beachten Sie beispielsweise die folgende HTTP-Anforderung, die nach der vorherigen Anforderung eingeht.

```
            GET /content/stories/example-story.html?ref=xyz987&split-pages=true
HTTP/1.1
            Host: d111111abcdef8.cloudfront.net
            User-Agent: Mozilla/5.0 AppleWebKit/537.36 Chrome/83.0.4103.116
            Accept: text/html,*/*
            Accept-Language: en-US,en
            Cookie: session_id=wxyz9876
            Referer: https://rss.news.example.net/
```

Diese Anforderung gilt für dasselbe Objekt wie die vorherige Anforderung, unterscheidet sich jedoch von der vorherigen Anforderung. Sie hat eine andere URL-Abfragezeichenfolge, verschiedene `User-Agent`- und `Referer`-Header und ein anderes `session_id`- Cookie. Keiner dieser Werte ist jedoch standardmäßig Teil des Cache-Schlüssels, so dass diese zweite Anforderung zu einem Cache-Treffer führt.

Passen Sie den Cache-Schlüssel an

In einigen Fällen möchten Sie möglicherweise mehr Informationen in den Cache-Schlüssel aufnehmen, obwohl dies zu weniger Cache-Treffern führen kann. Sie können angeben, was in dem Cache-Schlüssel enthalten sein soll, indem Sie eine Cache-Richtlinie verwenden.

Wenn Ihr Ursprungs-Server beispielsweise den `Accept-Language`-HTTP-Header in Viewer-Anforderungen verwendet, um unterschiedliche Inhalte basierend auf der Sprache des Viewers zurückzugeben, sollten Sie diesen Header möglicherweise in den Cache-Schlüssel einfügen. Wenn Sie das tun, CloudFront verwendet er diesen Header, um Cache-Treffer zu ermitteln, und bezieht den

Header in ursprüngliche Anfragen ein (Anfragen, die CloudFront an den Ursprung gesendet werden, wenn ein Cache-Fehler auftritt).

Eine mögliche Folge der Aufnahme zusätzlicher Werte in den Cache-Schlüssel besteht darin, dass aufgrund der Variation, die bei Viewer-Anfragen auftreten kann, CloudFront möglicherweise doppelte Objekte zwischengespeichert werden. Beispielsweise können Viewer einen der folgenden Werte für die `Accept-Language`-Header senden:

- `en-US,en`
- `en,en-US`
- `en-US, en`
- `en-US`

All diese unterschiedlichen Werte deuten darauf hin, dass die Sprache des Betrachters Englisch ist, aber die Variation kann CloudFront dazu führen, dass dasselbe Objekt mehrmals zwischengespeichert wird. Dies kann Cache-Treffer reduzieren und die Anzahl der Ursprungsanforderungen erhöhen. Sie könnten diese Duplizierung vermeiden, indem Sie den `Accept-Language` Header nicht in den Cache-Schlüssel aufnehmen und stattdessen Ihre Website oder Anwendung so konfigurieren, dass sie unterschiedliche URLs Inhalte in verschiedenen Sprachen verwendet (z. B.`/en-US/content/stories/example-story.html`).

Bei jedem gegebenen Wert, den Sie in den Cache-Schlüssel einschließen möchten, sollten Sie sicherstellen, dass Sie verstehen, wie viele verschiedene Variationen dieses Werts in Viewer-Anforderungen angezeigt werden können. Bei bestimmten Anforderungswerten ist es selten sinnvoll, sie in den Cache-Schlüssel aufzunehmen. Beispielsweise kann der `User-Agent`-Header Tausende von eindeutigen Variationen haben, daher ist er im Allgemeinen kein guter Kandidat für die Aufnahme in den Cache-Schlüssel. Cookies, die benutzerspezifische oder sitzungsspezifische Werte aufweisen und über Tausende (oder sogar Millionen) von Anforderungen eindeutig sind, sind auch keine guten Kandidaten für die Aufnahme in Cache-Schlüssel. Wenn Sie diese Werte in den Cache-Schlüssel aufnehmen, führt jede eindeutige Variation zu einer weiteren Kopie des Objekts im Cache. Wenn diese Kopien des Objekts nicht eindeutig sind oder wenn Sie eine so großen Anzahl von leicht unterschiedlichen Objekten haben, dass jedes Objekt nur eine kleine Anzahl von Cache-Treffern erhält, sollten Sie einen anderen Ansatz in Betracht ziehen. Sie können diese hochvariablen Werte aus dem Cache-Schlüssel ausschließen oder Objekte als nicht zwischenspeicherbar markieren.

Seien Sie vorsichtig, wenn Sie den Cache-Schlüssel anpassen. Manchmal ist dies wünschenswert, es kann aber unbeabsichtigte Konsequenzen haben, wie z. B. das Zwischenspeichern von

doppelten Objekten, die Verringerung der Cache-Trefferquote und die Erhöhung der Anzahl der Ursprungsanforderungen. Wenn Ihre Ursprungs-Website oder -Anwendung bestimmte Werte von Viewer-Anforderungen für Analysen, Telemetrie oder andere Verwendungszwecke erhalten muss, diese Werte jedoch das vom Ursprung zurückgegebene Objekt nicht ändern, verwenden Sie eine Ursprungsanforderungsrichtlinie um diese Werte in Ursprungsanforderungen einzuschließen, sie aber nicht in den Cache-Schlüssel aufzunehmen.

Kontrollieren Sie Herkunftsanfragen mit einer Richtlinie

Wenn eine Viewer-Anfrage zu einem Cache-Fehler CloudFront führt (das angeforderte Objekt wird am Edge-Standort nicht zwischengespeichert), CloudFront sendet er eine Anfrage an den Ursprung, um das Objekt abzurufen. Dies wird als Ursprungsanforderung bezeichnet. Die Ursprungsanforderung enthält stets die folgenden Informationen aus der Anforderung des Viewers:

- Den URL-Pfad (nur den Pfad, ohne URL-Abfragezeichenfolgen oder Domänennamen)
- Den Text der Anforderung (wenn vorhanden)
- Die HTTP-Header, die CloudFront automatisch in jeder ursprünglichen Anfrage enthalten sind, einschließlich`Host`, und `User-Agent` `X-Amz-Cf-Id`

Andere Informationen aus der Viewer-Anforderung, z. B. URL-Abfragezeichenfolgen, HTTP-Header und Cookies, sind standardmäßig nicht in der Ursprungsanforderung enthalten. (Ausnahme: Leitet bei älteren Cache-Einstellungen CloudFront die Header standardmäßig an Ihren Ursprung weiter.) Möglicherweise möchten Sie jedoch einige dieser anderen Informationen am Ursprung erhalten, um z. B. Daten für Analysen oder Telemetrie zu sammeln. Sie können eine Ursprungsanforderungsrichtlinie verwenden, um die Informationen zu steuern, die in einer Ursprungsanforderung enthalten sind.

Ursprungsanforderungsrichtlinien unterscheiden sich von Cache-Richtlinien. Diese steuern den Cache-Schlüssel. Auf diese Weise können Sie zusätzliche Informationen am Ursprung erhalten und gleichzeitig eine gute Cache-Trefferquote (den Anteil der Zuschaueranfragen, die zu einem Cache-Treffer führen) aufrechterhalten. Hierzu steuern Sie separat, welche Informationen in Ursprungsanforderungen enthalten sind (über die Ursprungsanforderungsrichtlinie) und welche Informationen im Cache-Schlüssel enthalten sind (über die Cache-Richtlinie).

Auch wenn dies zwei getrennte Arten von Richtlinien sind, sind sie verwandt. Alle URL-Abfragezeichenfolgen, HTTP-Header und Cookies, die Sie in den Cache-Schlüssel (über eine Cache-Richtlinie) einfügen, werden automatisch auch in Ursprungsanforderungen eingefügt. Mithilfe der Ursprungsanforderungsrichtlinie können Sie die Informationen angeben, die Sie in Ursprungsanforderungen, jedoch nicht in den Cache-Schlüssel einfügen möchten. Genau wie bei einer Cache-Richtlinie fügen Sie eine Quell-Anforderungsrichtlinie einem oder mehreren Cache-Verhaltensweisen in einer CloudFront Distribution zu.

Sie können eine Ursprungsanforderungsrichtlinie auch verwenden, um einer Ursprungsanforderung zusätzliche HTTP-Header hinzuzufügen, die nicht in der Viewer-Anforderung enthalten waren.

Diese zusätzlichen Header werden CloudFront vor dem Senden der ursprünglichen Anfrage hinzugefügt, wobei die Header-Werte automatisch auf der Grundlage der Viewer-Anfrage bestimmt werden. Weitere Informationen finden Sie unter the section called “ CloudFront Anforderungsheader hinzufügen”.

Themen

Verstehen Sie die Richtlinien für Anfragen mit Herkunft

CloudFront bietet einige vordefinierte Richtlinien für Anfragen mit Herkunft, die als verwaltete Richtlinien bezeichnet werden, für allgemeine Anwendungsfälle. Sie können diese verwalteten Richtlinien verwenden oder eine eigene Ursprungsanforderungsrichtlinie speziell für Ihre Anforderungen erstellen. Weitere Informationen zu verwalteten Richtlinien finden Sie unter Richtlinien für verwaltete Origin-Anfragen verwenden.

Eine Ursprungsanforderungsrichtlinie enthält die folgenden Einstellungen, die in Richtlinieninformationen und Ursprungsanforderungseinstellungen unterteilt werden.

Richtlinieninformationen

Name

Ein Name zur Identifizierung der Ursprungsanforderungsrichtlinie. Sie verwenden den Namen in der Konsole, um die Ursprungsanforderungsrichtlinie einem Cacheverhalten anzufügen.

Beschreibung

Ein Kommentar zur Beschreibung der Ursprungsanforderungsrichtlinie. Dieser Schritt ist optional.

Ursprungsanforderungseinstellungen

Die Einstellungen für Ursprungsanfragen geben die Werte in Viewer-Anfragen an, die in Anfragen enthalten sind, die an den Ursprung CloudFront gesendet werden (sogenannte Origin-Anfragen).

Bei den Werten kann es sich um URL-Abfragezeichenfolgen, HTTP-Header und Cookies handeln. Die von Ihnen angegebenen Werte werden in Ursprungsanforderungen, jedoch nicht in den Cache-Schlüssel eingefügt. Informationen zum Steuern des Cache-Schlüssels finden Sie unter Steuern Sie den Cache-Schlüssel mit einer Richtlinie.

Header

Die HTTP-Header in Viewer-Anfragen, die in CloudFront ursprünglichen Anfragen enthalten sind. Sie können für Header eine der folgenden Einstellungen auswählen:

- None (Keine) – Die HTTP-Header in Betrachteranfragen werden nicht in Ursprungsanforderungen eingefügt.
- All viewer headers (Alle Betrachter-Header) – Alle HTTP-Header in Betrachteranfragen werden in Ursprungsanforderungen eingefügt.
- Alle Viewer-Header und die folgenden CloudFront Header — Alle HTTP-Header in Viewer-Anfragen sind in den ursprünglichen Anfragen enthalten. Darüber hinaus geben Sie an, welche der CloudFront Header Sie zu den ursprünglichen Anfragen hinzufügen möchten. Weitere Hinweise zu den CloudFront Headern finden Sie unter. the section called “ CloudFront Anforderungsheader hinzufügen”
- Die folgenden Header einschließen – Sie geben an, welche HTTP-Header in Ursprungsanforderungen eingeschlossen werden.

  > **Note**
  >
  > Geben Sie keinen Header an, der bereits in Ihren Einstellungen für benutzerdefinierte Ursprungs-Header enthalten ist. Weitere Informationen finden Sie unter Konfiguriere CloudFront es so, dass benutzerdefinierte Header zu ursprünglichen Anfragen hinzugefügt werden.

- Alle Viewer-Header außer – Sie geben an, welche HTTP-Header nicht in Ursprungsanforderungen enthalten sind. Alle anderen HTTP-Header in Viewer-Anforderungen, mit Ausnahme der angegebenen, sind enthalten.

Wenn Sie die Einstellung Alle Viewer-Header und die folgenden CloudFront Header, Folgende Header einbeziehen oder Alle Viewer-Header außer verwenden, geben Sie HTTP-Header nur anhand des Header-Namens an. CloudFront schließt den vollständigen Header, einschließlich seines Werts, in ursprüngliche Anfragen ein.

> **Note**
>
> Wenn Sie die Einstellung Alle Viewer-Header außer verwenden, um den `Host` Header des Viewers zu entfernen, wird der ursprünglichen Anfrage ein neuer `Host` Header mit dem Domainnamen des Ursprungs CloudFront hinzugefügt.

Cookies

Die Cookies in Viewer-Anfragen, CloudFront einschließlich in ursprünglichen Anfragen. Für Cookies können Sie eine der folgenden Einstellungen auswählen:

- Keine – Die Cookies in Viewer-Anforderungen werden nicht in Ursprungsanforderungen eingefügt.
- Alle – Alle Cookies in Viewer-Anforderungen werden in Ursprungsanforderungen eingefügt.
- Die folgenden Cookies einschließen – Sie geben an, welche Cookies in Viewer-Anforderungen in Ursprungsanforderungen eingefügt werden.
- Alle Cookies außer – Sie geben an, welche Cookies in Viewer-Anforderungen nicht in Ursprungsanforderungen eingefügt werden. Alle anderen Cookies in Viewer-Anfragen werden eingefügt.

Wenn Sie die Einstellung „Folgende Cookies einbeziehen“ oder „Alle Cookies außer“ verwenden, geben Sie Cookies nur anhand ihres Namens an. CloudFront schließt das vollständige Cookie, einschließlich seines Werts, in ursprüngliche Anfragen ein.

Abfragezeichenfolgen

Die URL-Abfragezeichenfolgen in Viewer-Anfragen, CloudFront einschließlich in ursprünglichen Anfragen. Für Abfragezeichenfolgen können Sie eine der folgenden Einstellungen auswählen:

- None (Keine) – Die Abfragezeichenfolgen in Betrachteranfragen werden nicht in Ursprungsanforderungen eingefügt.
- All (Alle) – Alle Abfragezeichenfolgen in Betrachteranfragen werden in Ursprungsanforderungen eingefügt.
- Die folgenden Abfragezeichenfolgen einschließen – Sie geben an, welche Abfragezeichenfolgen in Viewer-Anforderungen in Ursprungsanforderungen eingefügt werden.
- Alle Abfragezeichenfolgen außer – Sie geben an, welche Abfragezeichenfolgen in Viewer-Anforderungen nicht in Ursprungsanforderungen eingefügt werden. Alle anderen Abfragezeichenfolgen werden eingefügt.

Wenn Sie die Einstellung Folgende Abfragezeichenfolgen einbeziehen oder Alle Abfragezeichenfolgen außer verwenden, geben Sie Abfragezeichenfolgen nur anhand ihres Namens an. CloudFront schließt die vollständige Abfragezeichenfolge, einschließlich ihres Werts, in ursprüngliche Anfragen ein.

Erstellen Sie Richtlinien für ursprüngliche Anfragen

Sie können eine Richtlinie für ursprüngliche Anfragen verwenden, um die Werte (URL-Abfragezeichenfolgen, HTTP-Header und Cookies) zu steuern, die in Anfragen enthalten sind, die CloudFront an Ihren Absender gesendet werden. Sie können eine Richtlinie für Ursprungsanfragen in der CloudFront Konsole, mit der AWS Command Line Interface (AWS CLI) oder mit der CloudFront API erstellen.

Nachdem Sie eine Origin-Anforderungsrichtlinie erstellt haben, fügen Sie sie einem oder mehreren Cache-Verhalten in einer CloudFront Distribution hinzu.

Ursprungsanforderungsrichtlinien sind nicht erforderlich. Wenn einem Cacheverhalten keine Ursprungsanforderungsrichtlinie angefügt ist, enthält die Ursprungsanforderung alle Werte, die in der Cache-Richtlinie angegeben sind, jedoch keine weiteren Werte.

> **Note**
>
> Um eine Ursprungsanforderungsrichtlinie verwenden zu können, muss das Cacheverhalten auch eine Cache-Richtlinie verwenden. Sie können eine Ursprungsanforderungsrichtlinie in einem Cacheverhalten nicht ohne eine Cache-Richtlinie verwenden.

Console

So erstellen Sie eine Ursprungsanforderungsrichtlinie (Konsole)

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Seite Richtlinien in der CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home?#/policies.
2. Wählen Sie Ursprungsanforderung und anschließend Ursprungsanforderungsrichtlinie erstellen aus.
3. Wählen Sie die gewünschte Einstellung für diese Ursprungsanforderungsrichtlinie aus. Weitere Informationen finden Sie unter Verstehen Sie die Richtlinien für Anfragen mit Herkunft.

4. Wenn Sie fertig sind, wählen Sie Erstellen.

Nach der Erstellung einer Ursprungsanforderungsrichtlinie können Sie diese einem Cacheverhalten anfügen.

So fügen Sie eine Ursprungsanforderungsrichtlinie an eine vorhandene Verteilung an (Konsole)

1. Öffnen Sie die Seite Distributionen in der CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home#/distributions.
2. Wählen Sie die Verteilung aus, die Sie aktualisieren möchten, und anschließend die Registerkarte Verhaltensweisen aus.
3. Wählen Sie das Cacheverhalten, das Sie aktualisieren möchten, und anschließend Bearbeiten aus.

   Um ein neues Cacheverhalten zu erstellen, wählen Sie Verhalten erstellen aus.
4. Stellen Sie im Abschnitt Cache-Schlüssel- und Ursprungsanforderungen sicher, dass Cache-Richtlinie und Ursprungsanforderungsrichtlinie ausgewählt sind.
5. Wählen Sie unter Ursprungsanforderungsrichtlinie die Ursprungsanforderungsrichtlinie aus, die diesem Cacheverhalten angefügt werden soll.
6. Wählen Sie unten auf der Seite die Option Änderungen speichern aus.

So fügen Sie eine Ursprungsanforderungsrichtlinie an eine neue Verteilung an (Konsole)

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie Verteilung erstellen.
3. Stellen Sie im Abschnitt Cache-Schlüssel- und Ursprungsanforderungen sicher, dass Cache-Richtlinie und Ursprungsanforderungsrichtlinie ausgewählt sind.
4. Wählen Sie für Origin request policy (Ursprungsanforderungsrichtlinie) die Ursprungsanforderungsrichtlinie aus, die dem Standard-Cacheverhalten dieser Verteilung angefügt werden soll.
5. Wählen Sie die gewünschten Einstellungen für den Ursprung, das Standard-Cacheverhalten und andere Verteilungseinstellungen aus. Weitere Informationen finden Sie unter Referenz für alle Verteilungseinstellungen.
6. Wenn Sie fertig sind, wählen Sie Verteilung erstellen aus.

CLI

Verwenden Sie den aws cloudfront create-origin-request-policy Befehl, um eine Origin-Request-Richtlinie mit dem AWS Command Line Interface (AWS CLI) zu erstellen. Sie können die Eingabeparameter des Befehls in einer Eingabedatei bereitstellen, anstatt jeden einzelnen Parameter als Befehlszeileneingabe anzugeben.

So erstellen Sie eine Ursprungsanforderungsrichtlinie (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um eine Datei mit dem Namen `origin-request-policy.yaml` zu erstellen, die alle Eingabeparameter für den create-origin-request-policy-Befehl enthält.

```
aws cloudfront create-origin-request-policy --generate-cli-skeleton yaml-input >
 origin-request-policy.yaml
```

2. Öffnen Sie die Datei mit dem Namen `origin-request-policy.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, um die gewünschten Einstellungen für die Ursprungsanforderungsrichtlinie anzugeben, und speichern Sie die Datei. Sie können optionale Felder aus der Datei entfernen, erforderliche Felder dürfen jedoch nicht entfernt werden.

   Weitere Informationen zu den Einstellungen für die Ursprungsanforderungsrichtlinie finden Sie unter Verstehen Sie die Richtlinien für Anfragen mit Herkunft.

3. Verwenden Sie den folgenden Befehl, um die Ursprungsanforderungsrichtlinie mit Eingabeparametern aus der Datei `origin-request-policy.yaml` zu erstellen.

```
aws cloudfront create-origin-request-policy --cli-input-yaml file://origin-
request-policy.yaml
```

   Notieren Sie den `Id`-Wert in der Ausgabe des Befehls. Dies ist die Richtlinien-ID für die ursprüngliche Anfrage. Sie benötigen sie, um die Richtlinie für die Ursprungsanforderung an das Cache-Verhalten einer CloudFront Distribution anzuhängen.

So fügen Sie eine Ursprungsanforderungsrichtlinie an eine vorhandene Verteilung an (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um die Verteilungskonfiguration für die CloudFront Distribution zu speichern, die Sie aktualisieren möchten. Ersetzen Sie *distribution_ID* durch die ID der Verteilung.

```
aws cloudfront get-distribution-config --id distribution_ID --output yaml >
 dist-config.yaml
```

2. Öffnen Sie die Datei mit dem Namen `dist-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und nehmen Sie die folgenden Änderungen an jedem Cacheverhalten vor, das Sie aktualisieren, um eine Ursprungsanforderungsrichtlinie zu verwenden.

   - Fügen Sie in der Cache-Verhaltensweise ein Feld mit dem Namen `OriginRequestPolicyId` hinzu. Verwenden Sie als Wert des Feldes die Ursprungsanforderungsrichtlinien-ID, die Sie nach dem Erstellen der Richtlinie notiert haben.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

   Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Verteilung zur Verwendung der Ursprungsanforderungsrichtlinie zu aktualisieren. Ersetzen Sie *distribution_ID* durch die ID der Verteilung.

```
aws cloudfront update-distribution --id distribution_ID --cli-input-yaml file://
dist-config.yaml
```

So fügen Sie eine Ursprungsanforderungsrichtlinie an eine neu Verteilung an (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um eine Datei mit dem Namen `distribution.yaml` zu erstellen, die alle Eingabeparameter für den create-distribution-Befehl enthält.

```
aws cloudfront create-distribution --generate-cli-skeleton yaml-input >
 distribution.yaml
```

2. Öffnen Sie die Datei mit dem Namen `distribution.yaml`, die Sie gerade erstellt haben. Geben Sie im Standard-Cacheverhalten in das Feld `OriginRequestPolicyId` die Ursprungsanforderungsrichtlinien-ID ein, die Sie nach dem Erstellen der Richtlinie notiert haben. Fahren Sie mit der Bearbeitung der Datei fort, um die gewünschten Verteilungseinstellungen anzugeben, und speichern Sie die Datei, wenn Sie fertig sind.

   Weitere Informationen zu den Verteilungseinstellungen finden Sie unter Referenz für alle Verteilungseinstellungen.

3. Verwenden Sie den folgenden Befehl, um die Verteilung mit Eingabeparametern aus der Datei `distribution.yaml` zu erstellen.

```
aws cloudfront create-distribution --cli-input-yaml file://distribution.yaml
```

API

Um mit der CloudFront API eine Origin-Request-Richtlinie zu erstellen, verwenden Sie CreateOriginRequestPolicy. Weitere Informationen zu den Feldern, die Sie in diesem API-Aufruf angeben, finden Sie in Verstehen Sie die Richtlinien für Anfragen mit Herkunft und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Nach der Erstellung einer Ursprungsanforderungsrichtlinie können Sie diese mit einem der folgenden API-Aufrufe an ein Cacheverhalten anfügen:

- Um es an ein Cache-Verhalten in einer vorhandenen Distribution anzuhängen, verwenden Sie UpdateDistribution.
- Um es an ein Cache-Verhalten in einer neuen Distribution anzuhängen, verwenden Sie CreateDistribution.

Geben Sie für beide API-Aufrufe die ID der Ursprungsanforderungsrichtlinie im Feld `OriginRequestPolicyId` innerhalb eines Cache-Verhaltens an. Weitere Informationen zu den anderen Feldern, die Sie in diesen API-Aufrufen angeben, finden Sie in Referenz für alle

Verteilungseinstellungen und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Richtlinien für verwaltete Origin-Anfragen verwenden

CloudFront bietet eine Reihe von Richtlinien für verwaltete Herkunftsanfragen, die Sie an jedes Cache-Verhalten Ihrer Distribution anhängen können. Bei Verwendung einer verwalteten Ursprungsanforderungsrichtlinie müssen Sie keine eigene Ursprungsanforderungsrichtlinie schreiben oder verwalten. Die verwalteten Richtlinien verwenden Einstellungen, die für bestimmte Anwendungsfälle optimiert sind.

Wenn Sie eine verwaltete Ursprungsanforderungsrichtlinie verwenden möchten, fügen Sie sie einem Cache-Verhalten in Ihrer Verteilung zu. Der Prozess ist der gleiche wie beim Erstellen einer Ursprungsanforderungsrichtlinie. Anstatt jedoch eine neue zu erstellen, fügen Sie einfach eine der verwalteten Ursprungsanforderungsrichtlinien an. Sie fügen die Richtlinie entweder nach Namen (mit der Konsole) oder nach ID (mit AWS CLI oder SDKs) hinzu. Die Namen und IDs sind im folgenden Abschnitt aufgeführt.

Weitere Informationen finden Sie unter Erstellen Sie Richtlinien für ursprüngliche Anfragen.

In den folgenden Themen werden die verwalteten Ursprungsanforderungsrichtlinien beschrieben, die Sie verwenden können.

Themen

- AllViewer
- AllViewerAndCloudFrontHeaders-2022—06
- AllViewerExceptHostHeader
- CORS- CustomOrigin
- CORS-S3Origin
- Elementar- - MediaTailor PersonalizedManifests
- UserAgentRefererHeaders

AllViewer

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie enthält alle Werte (Header, Cookies und Abfragezeichenfolgen) aus der Viewer-Anforderung.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`216adef6-5c7f-47e4-b989-5492eafa07d3`

Diese Richtlinie hat folgende Einstellungen:

- Header, die in Ursprungsanfragen enthalten sind: Alle Header in der Betrachteranfrage
- Cookies, die in Ursprungsanfragen enthalten sind: Alle
- Abfragezeichenfolgen, die in Ursprungsanforderungen enthalten sind: Alle

AllViewerAndCloudFrontHeaders-2022—06

Diese Richtlinie in der Konsole anzeigen CloudFront

Diese Richtlinie umfasst alle Werte (Header, Cookies und Abfragezeichenfolgen) aus der Viewer-Anfrage sowie alle CloudFront Header, die bis Juni 2022 veröffentlicht wurden (CloudFront Header, die nach Juni 2022 veröffentlicht wurden, sind nicht enthalten).

Wenn Sie die oder die CloudFront API verwenden AWS CloudFormation AWS CLI, lautet die ID für diese Richtlinie:

`33f36d7e-f396-46d9-90e0-52428a34d9dc`

Diese Richtlinie hat folgende Einstellungen:

- In ursprünglichen Anfragen enthaltene Header: Alle Header in der Viewer-Anfrage und die folgenden CloudFront Header:
  - `CloudFront-Forwarded-Proto`
  - `CloudFront-Is-Android-Viewer`
  - `CloudFront-Is-Desktop-Viewer`
  - `CloudFront-Is-IOS-Viewer`
  - `CloudFront-Is-Mobile-Viewer`
  - `CloudFront-Is-SmartTV-Viewer`

- `CloudFront-Is-Tablet-Viewer`
- `CloudFront-Viewer-Address`
- `CloudFront-Viewer-ASN`
- `CloudFront-Viewer-City`
- `CloudFront-Viewer-Country`
- `CloudFront-Viewer-Country-Name`
- `CloudFront-Viewer-Country-Region`
- `CloudFront-Viewer-Country-Region-Name`
- `CloudFront-Viewer-Http-Version`
- `CloudFront-Viewer-Latitude`
- `CloudFront-Viewer-Longitude`
- `CloudFront-Viewer-Metro-Code`
- `CloudFront-Viewer-Postal-Code`
- `CloudFront-Viewer-Time-Zone`
- `CloudFront-Viewer-TLS`

- Cookies, die in Ursprungsanfragen enthalten sind: Alle
- Abfragezeichenfolgen, die in Ursprungsanforderungen enthalten sind: Alle

AllViewerExceptHostHeader

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie beinhaltet nicht den `Host`-Header der Viewer-Anforderung, aber alle anderen Werte (Header, Cookies und Abfragezeichenfolgen) aus der Viewer-Anforderung.

Diese Richtlinie umfasst auch zusätzliche CloudFront Anforderungsheader für das HTTP-Protokoll, die HTTP-Version, die TLS-Version und alle Header für Gerätetyp und Viewer-Standort.

Diese Richtlinie ist für die Verwendung mit Amazon API Gateway und AWS Lambda Funktions-URL-Ursprüngen vorgesehen. Bei diesen Ursprüngen wird davon `Host` ausgegangen, dass der Header den Ursprungs-Domainnamen und nicht den Domainnamen der CloudFront Distribution enthält. Das Weiterleiten des `Host`-Headers von der Viewer-Anforderung an diese Ursprünge kann dazu führen, dass sie nicht funktionieren.

> **Note**
>
> Wenn Sie diese Richtlinie für verwaltete Herkunftsanfragen verwenden, um den `Host` Header des Betrachters zu entfernen, wird der ursprünglichen Anfrage ein neuer `Host` Header mit dem Domainnamen des Ursprungs CloudFront hinzugefügt.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`b689b0a8-53d0-40ab-baf2-68738e2966ac`

Diese Richtlinie hat folgende Einstellungen:

- Header, die in Ursprungsanforderungen enthalten sind: Alle Header in der Viewer-Anforderung mit Ausnahme des `Host`-Headers
- Cookies, die in Ursprungsanfragen enthalten sind: Alle
- Abfragezeichenfolgen, die in Ursprungsanforderungen enthalten sind: Alle

CORS- CustomOrigin

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie enthält den Header, der CORS-Anforderungen (Cross-Origin Resource Sharing) aktiviert, wenn der Ursprung ein benutzerdefinierter Ursprung ist.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`59781a5b-3903-41f3-afcb-af62929ccde1`

Diese Richtlinie hat folgende Einstellungen:

- Header, die in Ursprungsanfragen enthalten sind:
  - `Origin`
- Cookies, die in Ursprungsanfragen enthalten sind: Keine
- Abfragezeichenfolgen, die in Ursprungsanforderungen enthalten sind: Keine

CORS-S3Origin

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie enthält die Header, die CORS-Anforderungen (Cross-Origin Resource Sharing) aktivieren, wenn der Ursprung ein Amazon S3 Bucket ist.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`88a5eaf4-2fd4-4709-b370-b4c650ea3fcf`

Diese Richtlinie hat folgende Einstellungen:

- Header, die in Ursprungsanfragen enthalten sind:
  - `Origin`
  - `Access-Control-Request-Headers`
  - `Access-Control-Request-Method`
- Cookies, die in Ursprungsanfragen enthalten sind: Keine
- Abfragezeichenfolgen, die in Ursprungsanforderungen enthalten sind: Keine

Elementar- - MediaTailor PersonalizedManifests

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie wurde für die Verwendung mit einem Ursprung entwickelt, bei dem es sich um einen AWS Elemental MediaTailor -Endpunkt handelt.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`775133bc-15f2-49f9-abea-afb2e0bf67d2`

Diese Richtlinie hat folgende Einstellungen:

- Header, die in Ursprungsanfragen enthalten sind:
  - `Origin`
  - `Access-Control-Request-Headers`
  - `Access-Control-Request-Method`

  - `User-Agent`
  - `X-Forwarded-For`
- Cookies, die in Ursprungsanfragen enthalten sind: Keine
- Abfragezeichenfolgen, die in Ursprungsanforderungen enthalten sind: Alle

UserAgentRefererHeaders

Diese Richtlinie in der CloudFront Konsole anzeigen

Diese Richtlinie enthält nur die Header `User-Agent` und `Referer`. Sie enthält keine Abfragezeichenfolgen oder Cookies.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`acba4595-bd28-49b8-b9fe-13317c0390fa`

Diese Richtlinie hat folgende Einstellungen:

- Header, die in Ursprungsanfragen enthalten sind:
  - `User-Agent`
  - `Referer`
- Cookies, die in Ursprungsanfragen enthalten sind: Keine
- Abfragezeichenfolgen, die in Ursprungsanforderungen enthalten sind: Keine

CloudFront Anforderungsheader hinzufügen

Sie können so konfigurieren CloudFront , dass den Anfragen, die von Zuschauern CloudFront empfangen werden, spezifische HTTP-Header hinzugefügt und an Ihre Origin- oder Edge-Funktion weitergeleitet werden. Die Werte dieser HTTP-Header basieren auf den Eigenschaften des Viewers oder der Viewer-Anforderung. Die Header enthalten Informationen über den Gerätetyp, die IP-Adresse, den geografischen Standort, das Anforderungsprotokoll (HTTP oder HTTPS), die HTTP-Version, die TLS-Verbindungsdetails, den Fingerabdruck und den JA3 Fingerabdruck des Betrachters. JA4 Sie können das Cache-Verhalten Ihrer Distribution auch so konfigurieren, dass WebSocket Header weitergeleitet werden. Weitere Informationen finden Sie unter WebSockets Mit CloudFront Distributionen verwenden.

Mit diesen Headern kann Ihr Ursprung oder Ihre Edge-Funktion Informationen über den Viewer abrufen, sodass Sie keinen eigenen Code schreiben müssen, um diese Informationen zu ermitteln. Wenn Ihr Origin auf der Grundlage der Informationen in diesen Headern unterschiedliche Antworten zurückgibt, können Sie diese in den Cache-Schlüssel aufnehmen, sodass die Antworten separat CloudFront zwischengespeichert werden. Beispielsweise könnte der Ursprung basierend auf dem Land, in dem sich der Viewer befindet, mit Inhalten in einer spezifischen Sprache antworten oder er könnte mit Inhalten antworten, die auf einen spezifischen Gerätetyp zugeschnitten sind. Der Ursprung könnte diese Header auch in Protokolldateien schreiben, mit denen Sie Informationen über die Standorte und Gerätetypen der Viewer und vieles mehr ermitteln können.

Um diese Header in den Cache-Schlüssel einzufügen, verwenden Sie eine Cache-Richtlinie. Weitere Informationen erhalten Sie unter Steuern Sie den Cache-Schlüssel mit einer Richtlinie und the section called “Den Cache-Schlüssel verstehen”.

Wenn Sie Header am Ursprung empfangen, diese aber nicht in den Cache-Schlüssel einschließen möchten, verwenden Sie eine Ursprungsanforderungsrichtlinie. Weitere Informationen finden Sie unter Kontrollieren Sie Herkunftsanfragen mit einer Richtlinie.

Themen

Header vom Gerätetyp

Mit den folgenden Headern können Sie den Gerätetyp des Viewers ermitteln. CloudFront Legt den Wert dieser `User-Agent` Header basierend auf dem Wert des Headers auf `true` oder fest. `false` Wenn ein Gerät in mehr als eine Kategorie fällt, können mehrere Werte `true` sein. CloudFront Legt z. B. bei einigen Tablet-Geräten `CloudFront-Is-Mobile-Viewer` sowohl als auch `CloudFront-Is-Tablet-Viewer` auf `true` fest.

- `CloudFront-Is-Android-Viewer`— `true` Wird auf „Wenn“ gesetzt, wird CloudFront bestimmt, dass es sich bei dem Viewer um ein Gerät mit dem Betriebssystem Android handelt.
- `CloudFront-Is-Desktop-Viewer`— Auf „`true`Wenn“ eingestellt, wird CloudFront bestimmt, dass es sich bei dem Viewer um ein Desktop-Gerät handelt.

- `CloudFront-Is-IOS-Viewer`— Auf „`true`Wenn“ gesetzt, wird CloudFront bestimmt, dass es sich beim Viewer um ein Gerät mit einem mobilen Apple-Betriebssystem wie iPhone, iPod touch und einigen iPad-Geräten handelt.
- `CloudFront-Is-Mobile-Viewer`— Auf „`true`Wenn“ eingestellt, wird CloudFront bestimmt, dass es sich bei dem Betrachter um ein Mobilgerät handelt.
- `CloudFront-Is-SmartTV-Viewer`— Auf „`true`Wenn“ eingestellt, wird CloudFront bestimmt, dass es sich bei dem Zuschauer um ein Smart-TV handelt.
- `CloudFront-Is-Tablet-Viewer`— Auf „`true`Wenn“ eingestellt, wird CloudFront bestimmt, dass es sich bei dem Zuschauer um ein Tablet handelt.

Kopfzeilen zum Standort des Betrachters

Sie können die folgenden Header hinzufügen, um den Standort des Betrachters zu bestimmen. CloudFront bestimmt die Werte für diese Header anhand der IP-Adresse des Betrachters. Bei Nicht-ASCII-Zeichen in den Werten dieser Header wird das Zeichen gemäß Abschnitt 1.2 von RFC CloudFront 3986 mit Prozent codiert.

- `CloudFront-Viewer-Address` – Enthält die IP-Adresse des Viewers und den Quell-Port der Anforderung. Der Header-Wert `198.51.100.10:46532` bedeutet beispielsweise, dass die IP-Adresse des Viewers 198.51.100.10 lautet und der Quell-Port der Anforderung 46532 ist.
- `CloudFront-Viewer-ASN` – Enthält die autonome Systemnummer (ASN) des Viewers.

> **Note**
>
> `CloudFront-Viewer-Address` und `CloudFront-Viewer-ASN` können in einer Ursprungsanforderungsrichtlinie, aber nicht in einer Cache-Richtlinie hinzugefügt werden.

- `CloudFront-Viewer-Country` – Enthält den zweistelligen Ländercode für das Land des Viewers. Die Liste der Ländercodes finden Sie unter ISO 3166-1 alpha-2.
- `CloudFront-Viewer-City` – Enthält den Namen der Stadt des Viewers.

Wenn Sie die folgenden Header hinzufügen, werden sie auf alle Anfragen CloudFront angewendet, mit Ausnahme derjenigen, die aus dem Netzwerk stammen: AWS

- `CloudFront-Viewer-Country-Name` – Enthält den Namen des Landes des Viewers.

- `CloudFront-Viewer-Country-Region` – Enthält einen Code (bis zu drei Zeichen), der die Region des Viewers darstellt. Die Region ist die Unterteilung der ersten Ebene (die breiteste oder am wenigsten spezifische) des Codes ISO 3166-2.
- `CloudFront-Viewer-Country-Region-Name` – Enthält den Namen der Region des Viewers. Die Region ist die Unterteilung der ersten Ebene (die breiteste oder am wenigsten spezifische) des Codes ISO 3166-2.
- `CloudFront-Viewer-Latitude` – Enthält den ungefähren Breitengrad des Viewers.
- `CloudFront-Viewer-Longitude` – Enthält den ungefähren Längengrad des Viewers.
- `CloudFront-Viewer-Metro-Code` – Enthält den Metro-Code des Viewers. Dieser wird nur verwendet, wenn sich der Viewer in den Vereinigten Staaten befindet.
- `CloudFront-Viewer-Postal-Code` – Enthält die Postleitzahl des Viewers.
- `CloudFront-Viewer-Time-Zone` Enthält die Zeitzone des Viewers im IANA-Zeitzonen-Datenbankformat (z. B. `America/Los_Angeles`).

> **Note**
>
> `CloudFront-Viewer-CityCloudFront-Viewer-Metro-Code`, und ist `CloudFront-Viewer-Postal-Code` möglicherweise nicht für jede IP-Adresse verfügbar. Einige IP-Adressen können nicht mit ausreichender Genauigkeit geolokalisiert werden, um diese Informationen zu erhalten.

Header zur Bestimmung der Header-Struktur des Betrachters

Sie können jetzt die folgenden Header hinzufügen, um den Viewer leichter anhand der gesendeten Header identifizieren zu können. So können beispielsweise verschiedene Browser HTTP-Header in einer bestimmten Reihenfolge senden. Wenn der im `User-Agent`-Header angegebene Browser nicht mit der erwarteten Header-Reihenfolge dieses Browsers übereinstimmt, können Sie die Anforderung ablehnen. Auch wenn der `CloudFront-Viewer-Header-Count`-Wert nicht mit der Anzahl der Header in `CloudFront-Viewer-Header-Order` übereinstimmt, können Sie die Anforderung ablehnen.

- `CloudFront-Viewer-Header-Order` – Enthält die Header-Namen des Viewers in der angeforderten Reihenfolge, getrennt durch einen Doppelpunkt. Beispiel: `CloudFront-Viewer-`

`Header-Order: Host:User-Agent:Accept:Accept-Encoding`. Header, die das Zeichenlimit von 7 680 überschreiten, werden abgeschnitten.

- `CloudFront-Viewer-Header-Count` – Enthält die Gesamtzahl der Header des Viewers.

TLS-bezogene Header

Sie können die folgenden Header hinzufügen, um den Fingerabdruck, den JA3 Fingerabdruck und die TLS-Verbindungsdetails des Betrachters zu ermitteln: JA4

- `CloudFront-Viewer-JA3-Fingerprint`— Enthält den JA3 Fingerabdruck des Betrachters. Anhand des JA3 Fingerabdrucks können Sie feststellen, ob die Anfrage von einem bekannten Client stammt, ob es sich dabei um Malware oder einen böswilligen Bot oder um eine erwartete Anwendung handelt (auf der Zulassungsliste).
- `CloudFront-Viewer-JA4-Fingerprint`— Enthält den JA4 Fingerabdruck des Betrachters. Ähnlich wie beim JA3 Fingerabdruck können Sie anhand des JA4 Fingerabdrucks feststellen, ob die Anfrage von einem bekannten Client stammt, ob es sich dabei um Malware oder einen böswilligen Bot oder um eine erwartete Anwendung handelt (auf der Zulassungsliste). Sie können den Fingerabdruck verwenden, um eine Datenbank mit bekannten guten und schlechten Akteuren aufzubauen, die Sie bei der Überprüfung von HTTP-Anfragen verwenden können. Anschließend können Sie den Header-Wert auf Ihren Anwendungs-Webservern oder in Ihren Lambda @Edge und CloudFront Functions überprüfen, um den Header-Wert mit einer Liste bekannter Malware-Fingerabdrücke zu vergleichen, um böswillige Clients zu blockieren.
- `CloudFront-Viewer-TLS`— Enthält den SSL/TLS version, the cipher, and information about the SSL/TLS Handshake, der für die Verbindung zwischen dem Betrachter und verwendet wurde. CloudFront Der Header-Wert weist das folgende Format auf:

  ```
  SSL/TLS_version:cipher:handshake_information
  ```

  Der *`handshake_information`*-Header kann einen der folgenden Werte enthalten:

  - `fullHandshake` – Für die SSL/TLS-Sitzung wurde ein vollständiger Handshake durchgeführt.
  - `sessionResumed` – Eine vorherige SSL/TLS-Sitzung wurde wieder aufgenommen.
  - `connectionReused` – Eine frühere SSL/TLS-Verbindung wurde wiederverwendet.

  Im Folgenden finden Sie einige Beispielwerte für diesen Header:

```
TLSv1.3:TLS_AES_128_GCM_SHA256:sessionResumed
```

```
TLSv1.2:ECDHE-ECDSA-AES128-GCM-SHA256:connectionReused
```

```
TLSv1.1:ECDHE-RSA-AES128-SHA256:fullHandshake
```

```
TLSv1:ECDHE-RSA-AES256-SHA:fullHandshake
```

Eine vollständige Liste der möglichen SSL/TLS-Versionen und Verschlüsselungen, die in diesem Header-Wert enthalten sein können, finden Sie unter the section called “Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront”.

Hinweise

- Die JA4 Fingerabdrücke JA3 und die Fingerabdrücke werden aus dem `Client Hello` SSL/TLS-Paket abgeleitet. Sie sind nur für HTTPS-Anfragen vorhanden.
- Für diese TLS-bezogenen Header können Sie sie zu einer Ursprungsanforderungsrichtlinie hinzufügen, aber nicht zu einer Cache-Richtlinie.

CloudFront Andere Header

Sie können die folgenden Header hinzufügen, um den ursprünglichen Anforderungs-URI des Viewers, die Parameter und Werte der ursprünglichen Abfragezeichenfolge, das Protokoll und die Version zu ermitteln:

- `CloudFront-Error-Uri`— Enthält den ursprünglichen Anfrage-URI, der vom Viewer empfangen wurde.
- `CloudFront-Error-Args`— Enthält die ursprünglichen Parameter und Werte der Abfragezeichenfolge der Anfrage.
- `CloudFront-Forwarded-Proto` – Enthält das Protokoll der Anforderung des Viewers (HTTP oder HTTPS).
- `CloudFront-Viewer-Http-Version` – Enthält die HTTP-Version der Anforderung des Viewers.

Verstehen Sie, wie Origin-Request-Richtlinien und Cache-Richtlinien zusammenarbeiten

Sie können eine CloudFront Richtlinie für ursprüngliche Anfragen verwenden, um die Anfragen zu kontrollieren, die CloudFront an den Absender gesendet werden. Diese Anfragen werden als ursprüngliche Anfragen bezeichnet. Um eine Ursprungsanforderungsrichtlinie verwenden zu können, müssen Sie der Cache-Verhaltensweise eine Cache-Richtlinie anhängen. Sie können eine Ursprungsanforderungsrichtlinie in einem Cacheverhalten nicht ohne eine Cache-Richtlinie verwenden. Weitere Informationen finden Sie unter Kontrollieren Sie Herkunftsanfragen mit einer Richtlinie.

Die Richtlinien für ursprüngliche Anfragen und die Cache-Richtlinien bestimmen zusammen, welche Werte in den ursprünglichen Anfragen CloudFront enthalten sind. Alle URL-Abfragezeichenfolgen, HTTP-Header und Cookies, die Sie im Cache-Schlüssel (über eine Cache-Richtlinie) festlegen, werden automatisch auch in Ursprungsanforderungen eingefügt. Alle zusätzlichen Abfragezeichenfolgen, Header und Cookies, die Sie in einer Ursprungsanforderungsrichtlinie angeben, sind auch in Ursprungsanforderungen enthalten (nicht jedoch im Cache-Schlüssel).

Ursprungsanforderungs- und Cache-Richtlinien verfügen über Einstellungen, die anscheinend miteinander in Konflikt stehen. Beispielsweise kann eine Richtlinie bestimmte Werte zulassen, während eine andere sie blockiert. In der folgenden Tabelle wird erklärt, welche Werte in ursprünglichen Anfragen CloudFront enthalten sind, wenn Sie die Einstellungen einer Origin-Anforderungsrichtlinie und einer Cache-Richtlinie zusammen verwenden. Diese Einstellungen gelten im Allgemeinen für alle Arten von Werten (Abfragezeichenfolgen, Header und Cookies), mit der Ausnahme, dass Sie nicht alle Header angeben oder eine Header-Blockierliste in einer Cache-Richtlinie verwenden können.

	Ursprungsanforderungsrichtlinie			
	Keine	Alle	Zulassungsliste	Blockierliste
Cache-Richtlinie				
Keine	In der Ursprungsanforderung sind keine Werte aus der Viewer-Anforderung	Alle Werte aus der Viewer-Anforderung sind in der Ursprungs	Nur die in der Ursprungsanforderungsrichtlinie angegebenen	Alle Werte aus der Viewer-Anforderung mit Ausnahme der Werte, die in

	Ursprungsanforderungsrichtlinie			
	Keine	Alle	Zulassungsliste	Blockierliste
	enthalten, mit Ausnahme der Standardwerte, die in jeder Ursprungs anforderu ng enthalten sind. Weitere Informationen finden Sie unter Kontrollieren Sie Herkunfts anfragen mit einer Richtlinie.	anforderung enthalten.	Werte sind in der Ursprungs anforderung enthalten.	der Ursprungs anforderu ngsrichtlinie angegeben sind, sind in der Ursprungs anforderung enthalten.

	Ursprungsanforderungsrichtlinie			
	Keine	Alle	Zulassungsliste	Blockierliste
Alle Hinweis: Sie können nicht alle Header in einer Cache-Richtlinie angeben.	Alle Abfrageze ichenfolgen und Cookies aus der Viewer-An forderung sind in der Ursprungs anforderung enthalten.	Alle Werte aus der Viewer-An forderung sind in der Ursprungs anforderung enthalten.	Alle Abfrageze ichenfolgen und Cookies aus der Viewer-An forderung sowie alle Header, die in der Ursprungs anforderu ngsrichtlinie angegeben sind, sind in der Ursprungs anforderung enthalten.	Alle Abfrageze ichenfolgen und Cookies aus der Viewer-An forderung sind in der Ursprungs anforderung enthalten; auch die, die in der Blockierliste der Ursprungs anforderu ngsrichtlinie angegeben sind. Die Einstellu ng der Cache-Richtlinie hat Vorrang vor der Blockierliste der Ursprungs anforderu ngsrichtlinie.

	Ursprungsanforderungsrichtlinie			
	Keine	Alle	Zulassungsliste	Blockierliste
Zulassungsliste	Nur die angegebenen Werte aus der Viewer-Anforderung sind in der Ursprungsanforderung enthalten.	Alle Werte aus der Viewer-Anforderung sind in der Ursprungsanforderung enthalten.	Alle in der Cache-Richtlinie oder der Ursprungsanforderungsrichtlinie angegebenen Werte sind in der Ursprungsanforderung enthalten.	Die in der Cache-Richtlinie angegebenen Werte sind in der Ursprungsanforderung enthalten, auch wenn dieselben Werte in der Blockierliste der Ursprungsanforderungsrichtlinie angegeben sind. Die Zulassungsliste der Cache-Richtlinie hat Vorrang vor der Blockierliste der Ursprungsanforderungsrichtlinie.

	Ursprungsanforderungsrichtlinie			
	Keine	Alle	Zulassungsliste	Blockierliste
Blockierliste Hinweis: In einer Blockierliste für Cache-Richtlinien können Sie keine Header angeben.	Alle Abfragezeichenfolgen und Cookies aus der Viewer-Anforderung mit Ausnahme der angegebenen sind in der Ursprungsanforderung enthalten.	Alle Werte aus der Viewer-Anforderung sind in der Ursprungsanforderung enthalten.	Die in der Ursprungsanforderungsrichtlinie angegebenen Werte sind in der Ursprungsanforderung enthalten, auch wenn dieselben Werte in der Blockierliste der Cache-Richtlinie angegeben sind. Die Zulassungsliste der Ursprungsanforderungsrichtlinie hat Vorrang vor der Blockierliste der Cache-Richtlinie.	Alle Werte aus der Viewer-Anforderung mit Ausnahme der Werte, die in der Cache-Richtlinie oder der Ursprungsanforderungsrichtlinie angegeben sind, sind in der Ursprungsanforderung enthalten.

Hinzufügen oder Entfernen von HTTP-Headern in CloudFront Antworten mit einer Richtlinie

Sie können die HTTP-Header in den Antworten, die an Zuschauer (Webbrowser und andere Clients) gesendet werden, so konfigurieren CloudFront , dass sie geändert werden. CloudFront kann Header entfernen, die es vom Ursprung erhalten hat, oder Header zur Antwort hinzufügen, bevor die Antwort an die Zuschauer gesendet wird. Für diese Änderungen müssen Sie weder Code schreiben noch den Ursprung ändern.

Sie können beispielsweise Header wie `X-Powered-By` und entfernen, `Vary` sodass diese Header CloudFront nicht in den Antworten enthalten sind, die an die Zuschauer gesendet werden. Sie können auch HTTP-Header wie beispielsweise die folgenden hinzufügen:

- `Cache-Control`-Header zur Steuerung des Browser-Cachings
- `Access-Control-Allow-Origin`-Header zum Ermöglichen von Cross-Origin Resource Sharing (CORS). Sie können auch andere CORS-Header hinzufügen.
- Ein Satz von allgemeinen Sicherheits-Headern wie beispielsweise `Strict-Transport-Security`, `Content-Security-Policy` und `X-Frame-Options`
- Ein `Server-Timing` Header, in dem Informationen angezeigt werden, die sich auf die Leistung und die Weiterleitung der Anfrage und der Antwort beziehen. CloudFront

Um die Header anzugeben, die in HTTP-Antworten CloudFront hinzugefügt oder entfernt werden, verwenden Sie eine Antwort-Header-Richtlinie. Sie fügen einem weiteren Cache-Verhalten eine Antwort-Header-Richtlinie hinzu und CloudFront ändern die Header in den Antworten, die an Anfragen gesendet werden, die dem Cache-Verhalten entsprechen. CloudFrontändert die Header in den Antworten, die es vom Cache aus bereitstellt, und in den Antworten, die es vom Ursprung aus weiterleitet. Wenn die ursprüngliche Antwort einen oder mehrere der Header enthält, die in einer Antwort-Header-Richtlinie hinzugefügt wurden, kann die Richtlinie angeben, ob der Header CloudFront verwendet wird, den sie vom Ursprung erhalten hat, oder ob dieser Header mit dem Header in der Antwort-Header-Richtlinie überschrieben wird.

> **Note**
>
> Wenn Sie Ihren Antwort-Header-Richtlinien Header hinzufügen, die das Browser-Caching steuern, z. B. CloudFront fügt diese Header nur der Antwort des *`Cache-Control`*

Betrachters hinzu. Diese Header haben keinen Einfluss darauf, wie das angeforderte Objekt CloudFront zwischengespeichert wird.

CloudFront stellt vordefinierte Antwort-Header-Richtlinien, sogenannte verwaltete Richtlinien, für allgemeine Anwendungsfälle bereit. Sie können diese verwalteten Richtlinien verwenden oder eigene Richtlinien erstellen. Sie können eine einzelne Response-Header-Richtlinie mehreren Cache-Verhaltensweisen in mehreren Distributionen in Ihrem zuordnen. AWS-Konto

Weitere Informationen finden Sie hier:

Themen

- Verstehen Sie die Richtlinien für Antwort-Header
- Richtlinien für Antwort-Header erstellen
- Richtlinien für verwaltete Antwort-Header verwenden

Verstehen Sie die Richtlinien für Antwort-Header

Sie können eine Antwort-Header-Richtlinie verwenden, um die HTTP-Header anzugeben, die Amazon CloudFront entfernt oder zu Antworten hinzufügt, die es an Zuschauer sendet. Weitere Informationen zu Antwort-Header-Richtlinien und zu Gründen für ihre Verwendung finden Sie unter Fügen Sie Antwort-Header mit einer Richtlinie hinzu oder entfernen Sie sie.

In den folgenden Themen werden die Einstellungen in einer Antwort-Header-Richtlinie erläutert. Die Einstellungen sind in Kategorien unterteilt, die in den folgenden Themen behandelt werden.

Themen

- Richtliniendetails (Metadaten)
- CORS-Header
- Sicherheits-Header
- Benutzerdefinierte Header
- Entfernen von Headern
- Server-Timing-Header

Richtliniendetails (Metadaten)

Die Einstellungen für Richtliniendetails enthalten Metadaten zu einer Antwort-Header-Richtlinie.

- Name – Ein Name zur Identifizierung der Antwort-Header-Richtlinie. Verwenden Sie den Namen in der Konsole, um die Richtlinie einem Cache-Verhalten anzufügen.
- Beschreibung (optional) – Ein Kommentar zur Beschreibung der Antwort-Header-Richtlinie. Dies ist optional, kann Ihnen jedoch helfen, den Zweck der Richtlinie zu identifizieren.

CORS-Header

Mit den Cross-Origin Resource Sharing (CORS)-Einstellungen können Sie CORS-Header in einer Antwort-Header-Richtlinie hinzufügen und konfigurieren.

Bei dieser Liste liegt der Fokus darauf, wie Einstellungen und gültige Werte in einer Antwort-Header-Richtlinie angegeben werden. Weitere Informationen zu den jeweiligen Headern und deren Verwendung für reale CORS-Anforderungen und -Antworten finden Sie unter Cross-Origin Resource Sharing in den MDN-Webdokumenten und in der CORS-Protokollspezifikation.

Access-Control-Allow-Credentials

Dies ist eine boolesche Einstellung (`true`oder`false`), die bestimmt, ob der `Access-Control-Allow-Credentials` Header in Antworten auf CORS-Anfragen CloudFront hinzugefügt wird. Wenn diese Einstellung auf gesetzt ist`true`, wird der `Access-Control-Allow-Credentials: true` Header in Antworten auf CORS-Anfragen CloudFront hinzugefügt. Andernfalls wird dieser Header CloudFront nicht zu Antworten hinzugefügt.

Access-Control-Allow-Headers

Gibt die Header-Namen an, die als Werte für den `Access-Control-Allow-Headers` Header in Antworten auf CORS-Preflight-Anfragen CloudFront verwendet werden. Zu den gültigen Werten für diese Einstellung gehören HTTP-Header-Namen oder das Platzhalterzeichen (`*`), das anzeigt, dass alle Header zulässig sind.

> **Note**
>
> Der `Authorization` Header darf keinen Platzhalter verwenden und muss explizit aufgeführt werden.

Beispiele für die gültige Verwendung des Platzhalterzeichens

Beispiel	Übereinstimmung mit	Keine Übereinstimmung mit
`x-amz-*`	`x-amz-test` `x-amz-`	`x-amz`
`x-*-amz`	`x-test-amz` `x--amz`	
`*`	Alle Header außer `Authorization`	`Authorization`

Access-Control-Allow-Methods

Gibt die HTTP-Methoden an, die als Werte für den `Access-Control-Allow-Methods` Header in Antworten auf CORS-Preflight-Anfragen CloudFront verwendet werden. Gültige Werte sind `GET`, `DELETE`, `HEAD`, `OPTIONS`, `PATCH`, `POST`, `PUT` und `ALL`. `ALL` ist ein spezieller Wert, der alle aufgelisteten HTTP-Methoden enthält.

Access-Control-Allow-Origin

Gibt die Werte an, die im `Access-Control-Allow-Origin` Antwort-Header verwendet werden CloudFront können. Zu den gültigen Werten für diese Einstellung gehören ein spezifischer Ursprung (z. B. `http://www.example.com`) oder das Platzhalterzeichen (`*`), das angibt, dass alle Ursprünge zulässig sind.

Hinweise

- Das Platzhalterzeichen (`*`) ist als Unterdomäne ganz links () zulässig. `*.example.org`
- Das Platzhalterzeichen (`*`) ist an den folgenden Stellen nicht zulässig:
  - Top-Level-Domains (`example.*`)
  - Rechts neben Subdomains (`test.*.example.org`) oder innerhalb beliebiger Subdomains () `*test.example.org`
  - Innerhalb von Begriffen (`exa*mple.org`)

Beispiele für die Verwendung des Platzhalterzeichens finden Sie in der folgenden Tabelle.

Beispiel	Übereinstimmung mit	Keine Übereinstimmung mit

`http://*.example.org`	`http://www.example.org`
	
	`http://test.example.org`
		`https``://test.example.org`
		
		`https``://test.example.org:123`
		
		`http://test.example.org:123`

`*.example.org`	`test.example.org`
	
	`test.test.example.org`
	
	`.example.org`
	
	`http://test.example.org`
	
	`https://test.example.org`
		`http://test.example.org:123`
		
		`https://test.example.org:123`

`example.org`	`http://example.org`
	
	`https://example.org`	

`http://example.org`		**`https`**`://example.org` `http://example.org:123`

`http://example.org:*`	`http://example.org:123`
	
	`http://example.org`	

Beispiel	Übereinstimmung mit	Keine Übereinstimmung mit

`http://example.org:1*3`	`http://example.org:123`
	
	`http://example.org:1893`
	
	`http://example.org:13`	

`*.example.org:1*`	`test.example.org:123`	

Access-Control-Expose-Headers

Gibt die Header-Namen an, die als Werte für den `Access-Control-Expose-Headers` Header in Antworten auf CORS-Anfragen CloudFront verwendet werden. Zu den gültigen Werten für diese Einstellung gehören HTTP-Header-Namen oder das Platzhalterzeichen (`*`).

Access-Control-Max-Age

Eine Anzahl von Sekunden, die als Wert für den `Access-Control-Max-Age` Header in Antworten auf CORS-Preflight-Anfragen CloudFront verwendet wird.

Origin override (Ursprungsüberschreibung)

Eine boolesche Einstellung, die bestimmt, wie CloudFront sich verhält, wenn die Antwort vom Ursprung einen der CORS-Header enthält, die auch in der Richtlinie enthalten sind.

- Wenn diese Option auf gesetzt ist `true` und die ursprüngliche Antwort einen CORS-Header enthält, der auch in der Richtlinie enthalten ist, wird der Antwort der CORS-Header in der Richtlinie CloudFront hinzugefügt. CloudFront sendet diese Antwort dann an den Betrachter. CloudFront ignoriert den Header, den es vom Ursprung erhalten hat.
- Wenn diese Option auf gesetzt ist `false` und die ursprüngliche Antwort einen CORS-Header enthält (unabhängig davon, ob der CORS-Header in der Richtlinie enthalten ist), CloudFront schließt sie den CORS-Header ein, den sie vom Ursprung bis zur Antwort erhalten hat. CloudFront fügt der Antwort, die an den Betrachter gesendet wird, keine CORS-Header in der Richtlinie hinzu.

Sicherheits-Header

Mit den Einstellungen für Sicherheits-Header können Sie mehrere sicherheitsbezogene HTTP-Antwort-Header in einer Antwort-Header-Richtlinie hinzufügen und konfigurieren.

Diese Liste beschreibt, wie Sie Einstellungen und gültige Werte in einer Antwort-Header-Richtlinie angeben können. Weitere Informationen zu diesen Headern und ihrer Verwendung in realen HTTP-Antworten finden Sie unter den Links zu den MDN-Webdokumenten.

Content-Security-Policy

Gibt die Richtlinien zur Inhaltssicherheitsrichtlinie an, die als Werte für den `Content-Security-Policy` Antwort-Header CloudFront verwendet werden.

Weitere Hinweise zu diesem Header und zu gültigen Richtliniendirektiven finden Sie unter Content-Security-Policyin den MDN Web Docs.

> **Note**
>
> Der `Content-Security-Policy`-Header-Wert ist auf 1783 Zeichen begrenzt.

Referrer-Richtlinie

Gibt die Referrer-Richtliniendirektive an, die als Wert für den `Referrer-Policy` Antwort-Header CloudFront verwendet wird. Gültige Werte für diese Einstellung sind `no-referrer`, `no-referrer-when-downgrade`, `origin`, `origin-when-cross-origin`, `same-origin`, `strict-origin`, `strict-origin-when-cross-origin` und `unsafe-url`.

Weitere Hinweise zu diesem Header und diesen Direktiven finden Sie unter Referrer-Policyin den MDN Web Docs.

Strict-Transport-Security

Gibt die Direktiven und Einstellungen an, die als Wert für den `Strict-Transport-Security` Antwort-Header CloudFront verwendet werden. Für diese Einstellung geben Sie separat Folgendes an:

- Eine Anzahl von Sekunden, die als Wert für die `max-age` Direktive dieses Headers CloudFront verwendet wird

- Eine boolesche Einstellung (`true`oder`false`) für`preload`, die bestimmt, ob CloudFront die `preload` Direktive in den Wert dieses Headers aufgenommen wird
- Eine boolesche Einstellung (`true`oder`false`) für`includeSubDomains`, die bestimmt, ob die `includeSubDomains` Direktive in CloudFront den Wert dieses Headers aufgenommen wird

Weitere Hinweise zu diesem Header und diesen Direktiven finden Sie unter Strict-Transport-Securityin den MDN Web Docs.

X-Content-Type-Options

Dies ist eine boolesche Einstellung (`true`oder`false`), die bestimmt, ob der `X-Content-Type-Options` Header zu CloudFront Antworten hinzugefügt wird. Wenn diese Einstellung aktiviert ist`true`, wird der `X-Content-Type-Options: nosniff` Header zu Antworten CloudFront hinzugefügt. Andernfalls CloudFront wird dieser Header nicht hinzugefügt.

Weitere Informationen zu diesem Header finden Sie unter X-Content-Type-Optionsin den MDN Web Docs.

X-Frame-Options

Gibt die Direktive an, die als Wert für den `X-Frame-Options` Antwort-Header CloudFront verwendet wird. Gültige Werte für diese Einstellung sind `DENY` oder `SAMEORIGIN`.

Weitere Hinweise zu diesem Header und diesen Direktiven finden Sie unter X-Frame-Optionsin den MDN Web Docs.

X-XSS-Protection

Gibt die Direktiven und Einstellungen an, die als Wert für den `X-XSS-Protection` Antwort-Header CloudFront verwendet werden. Für diese Einstellung geben Sie separat Folgendes an:

- Die `X-XSS-Protection`-Einstellung `0` (deaktiviert XSS-Filterung) oder 1 (aktiviert XSS-Filterung)
- Eine boolesche Einstellung (`true`oder`false`) für`block`, die bestimmt, ob die `mode=block` Direktive CloudFront in den Wert für diesen Header aufgenommen wird
- Ein Berichts-URI, der bestimmt, ob CloudFront die `report=`*`reporting URI`* Direktive in den Wert für diesen Header aufgenommen wird

Sie können `true` für `block` oder einen Berichts-URI angeben. Sie können jedoch nicht beides zusammen angeben. Weitere Hinweise zu diesem Header und diesen Direktiven finden Sie unter X-XSS-Protectionin den MDN Web Docs.

Origin override (Ursprungsüberschreibung)

Jede dieser Sicherheits-Header-Einstellungen enthält eine boolesche Einstellung (`true`oder`false`), die bestimmt, wie CloudFront sich verhält, wenn die Antwort vom Ursprung diesen Header enthält.

Wenn diese Einstellung auf gesetzt ist `true` und die ursprüngliche Antwort den Header enthält, wird der Header in der Richtlinie der Antwort CloudFront hinzugefügt, die an den Betrachter gesendet wird. Der vom Ursprung empfangene Header wird dabei ignoriert.

Wenn diese Einstellung auf gesetzt ist `false` und die ursprüngliche Antwort den Header CloudFront enthält, wird der Header, den sie vom Ursprung erhalten hat, in die Antwort aufgenommen, die sie an den Betrachter sendet.

Wenn die ursprüngliche Antwort den Header nicht enthält, wird der Header in der Richtlinie der Antwort CloudFront hinzugefügt, die an den Betrachter gesendet wird. CloudFront tut dies, wenn diese Einstellung auf `true` oder gesetzt ist`false`.

Benutzerdefinierte Header

Sie können benutzerdefinierte Header-Einstellungen verwenden, um benutzerdefinierte HTTP-Header in einer Antwort-Header-Richtlinie hinzuzufügen und zu konfigurieren. CloudFront fügt diese Header zu jeder Antwort hinzu, die es an die Zuschauer zurückgibt. Bei benutzerdefinierten Headern geben Sie auch den Wert für den jeweiligen Header an, auch wenn die Angabe eines Werts optional ist. Dies liegt daran, dass ein Antwort-Header ohne Wert hinzugefügt werden CloudFront kann.

Jeder benutzerdefinierte Header verfügt zudem über eine eigene Einstellung für Origin override (Ursprungsüberschreibung):

- Wenn diese Einstellung auf gesetzt ist `true` und die ursprüngliche Antwort den benutzerdefinierten Header enthält, der in der Richtlinie enthalten ist, wird der benutzerdefinierte Header in der Richtlinie der Antwort CloudFront hinzugefügt, die sie an den Betrachter sendet. Der vom Ursprung empfangene Header wird dabei ignoriert.
- Wenn diese Einstellung aktiviert ist `false` und die ursprüngliche Antwort den benutzerdefinierten Header enthält, der in der Richtlinie enthalten ist, CloudFront wird der benutzerdefinierte Header, den sie vom Ursprung erhalten hat, in die Antwort aufgenommen, die sie an den Betrachter sendet.
- Wenn die ursprüngliche Antwort nicht den in der Richtlinie enthaltenen benutzerdefinierten Header enthält, wird der Antwort, die an den Betrachter gesendet wird, der benutzerdefinierte Header in

der Richtlinie CloudFront hinzugefügt. CloudFront tut dies, wenn diese Einstellung auf `true` oder gesetzt ist`false`.

Entfernen von Headern

Sie können Header angeben, die Sie aus den Antworten entfernen CloudFront möchten, die der Sender vom Absender erhält, sodass die Header nicht in den Antworten enthalten sind, die CloudFront an die Zuschauer gesendet werden. CloudFront entfernt die Header aus jeder Antwort, die es an Zuschauer sendet, unabhängig davon, ob die Objekte aus dem Cache oder aus dem CloudFront Ursprung bereitgestellt werden. Sie können beispielsweise Header entfernen, die für Browser nicht von Nutzen sind, wie z. B. `X-Powered-By` oder`Vary`, sodass diese Header aus den Antworten CloudFront entfernt werden, die an Zuschauer gesendet werden.

Wenn Sie mithilfe einer Antwort-Header-Richtlinie Header angeben, die entfernt werden sollen, werden zuerst die Header CloudFront entfernt und dann alle Header hinzugefügt, die in anderen Abschnitten der Antwort-Header-Richtlinie angegeben sind (CORS-Header, Sicherheitsheader, benutzerdefinierte Header usw.). Wenn Sie einen Header angeben, der entfernt werden soll, aber denselben Header auch in einem anderen Abschnitt der Richtlinie hinzufügen, wird der Header in die Antworten aufgenommen, die an CloudFront die Zuschauer gesendet werden.

> **Note**
>
> Sie können eine Richtlinie für Antwort-Header verwenden, um die Kopfzeilen `Server` und `Date` -Header zu entfernen, die vom Absender CloudFront empfangen wurden, sodass diese Header (wie sie vom Absender empfangen wurden) nicht in den Antworten enthalten sind, die an Zuschauer CloudFront gesendet werden. Wenn Sie das tun, CloudFront fügt es jedoch eine eigene Version dieser Header zu den Antworten hinzu, die es an Zuschauer sendet. Für den `Server` Header, der CloudFront etwas hinzufügt, lautet `CloudFront` der Wert des Headers.

Header, die Sie nicht entfernen können

Sie können die folgenden Header nicht mithilfe einer Antwort-Header-Richtlinie entfernen. Wenn Sie diese Header im Abschnitt Remove headers (Header entfernen) einer Antwort-Header-Richtlinie (`ResponseHeadersPolicyRemoveHeadersConfig` in der API) angeben, erhalten Sie eine Fehlermeldung.

- Connection
- Content-Encoding
- Content-Length
- Expect
- Host
- Keep-Alive
- Proxy-Authenticate
- Proxy-Authorization
- Proxy-Connection
- Trailer
- Transfer-Encoding
- Upgrade
- Via
- Warning
- X-Accel-Buffering
- X-Accel-Charset
- X-Accel-Limit-Rate
- X-Accel-Redirect
- X-Amz-Cf-.*
- X-Amzn-Auth
- X-Amzn-Cf-Billing
- X-Amzn-Cf-Id
- X-Amzn-Cf-Xff
- X-Amzn-ErrorType
- X-Amzn-Fle-Profile
- X-Amzn-Header-Count
- X-Amzn-Header-Order
- X-Amzn-Lambda-Integration-Tag
- X-Amzn-RequestId

- `X-Cache`
- `X-Edge-.*`
- `X-Forwarded-Proto`
- `X-Real-Ip`

Server-Timing-Header

Verwenden Sie die `Server-Timing` Header-Einstellung, um den `Server-Timing` Header in HTTP-Antworten zu aktivieren, die von gesendet werden CloudFront. Sie können diesen Header verwenden, um Metriken anzuzeigen, die Ihnen helfen können, Einblicke in das Verhalten und die Leistung von CloudFront und über Ihre Herkunft zu gewinnen. Sie können beispielsweise anzeigen, welche Cache-Ebene einen Cache-Treffer geliefert hat. Sie können auch die Latenz des ersten Byte vom Ursprung anzeigen, wenn ein Cache-Fehler vorliegt. Die Metriken in der `Server-Timing` Kopfzeile können dir dabei helfen, Probleme zu beheben oder die Effizienz deiner Konfiguration CloudFront oder deiner Origin-Konfiguration zu testen.

Weitere Informationen zur Verwendung des `Server-Timing` Headers mit CloudFront finden Sie in den folgenden Themen.

Um den Header `Server-Timing` zu aktivieren, müssen Sie eine Antwort-Header-Richtlinie erstellen (oder bearbeiten).

Themen

- Abtastrate und Pragma-Anforderungs-Header
- Server-Timing-Header vom Ursprung
- Metriken des Server-Timing-Headers
- Beispiele für Server-Timing-Header

Abtastrate und Pragma-Anforderungs-Header

Wenn Sie den Header `Server-Timing` in einer Antwort-Header-Richtlinie aktivieren, geben Sie auch die Abtastrate an. Die Stichprobenrate ist eine Zahl von 0 bis 100 (einschließlich), die den Prozentsatz der Antworten angibt, CloudFront zu denen Sie die `Server-Timing` Überschrift hinzufügen möchten. Wenn Sie die Samplerate auf 100 festlegen, wird der HTTP-Antwort der `Server-Timing` Header für jede Anfrage CloudFront hinzugefügt, die dem Cache-Verhalten

entspricht, an das die Antwort-Header-Richtlinie angehängt ist. Wenn Sie den Wert auf 50 setzen, wird der Header zu 50% der Antworten für Anfragen CloudFront hinzugefügt, die dem Cache-Verhalten entsprechen. Sie können als Abtastrate eine beliebige Zahl von 0 bis 100 mit bis zu vier Dezimalstellen festlegen.

Wenn die Samplerate auf eine Zahl unter 100 eingestellt ist, können Sie nicht kontrollieren, zu welcher Antwort der `Server-Timing` Header CloudFront hinzugefügt wird, sondern nur zu welchem Prozentsatz. Sie können jedoch den Header `Pragma` mit dem Wert `server-timing` in einer HTTP-Anforderung hinzufügen, um den Header `Server-Timing` in der Antwort auf diese Anforderung zu empfangen. Dies funktioniert unabhängig davon, auf welchen Wert die Abtastrate festgelegt ist. Selbst wenn die Samplerate auf Null (0) gesetzt ist, wird der Antwort der `Server-Timing` Header CloudFront hinzugefügt, wenn die Anfrage den `Pragma: server-timing` Header enthält.

Server-Timing-Header vom Ursprung

Wenn ein Cache-Fehler auftritt und CloudFront die Anfrage an den Ursprung weitergeleitet wird, kann der Ursprung in seiner Antwort auf CloudFront einen `Server-Timing` Header enthalten. In diesem Fall CloudFront fügt es seine Metriken dem `Server-Timing` Header hinzu, den es vom Ursprung erhalten hat. Die Antwort, die CloudFront an den Betrachter gesendet wird, enthält einen einzigen `Server-Timing` Header, der den Wert enthält, der vom Ursprung stammt, und die CloudFront hinzugefügten Metriken. Der Header-Wert aus dem Ursprung kann sich am Ende oder zwischen zwei Metriksätzen befinden, die CloudFront den Header ergänzen.

Bei einem Cache-Treffer enthält die Antwort, die an den Viewer CloudFront gesendet wird, einen einzelnen `Server-Timing` Header, der nur die CloudFront Metriken im Header-Wert enthält (der Wert aus dem Ursprung ist nicht enthalten).

Metriken des Server-Timing-Headers

Wenn der `Server-Timing` Header zu einer HTTP-Antwort CloudFront hinzugefügt wird, enthält der Wert des Headers eine oder mehrere Metriken, anhand derer Sie Erkenntnisse über das Verhalten und die Leistung von CloudFront und über Ihren Ursprung gewinnen können. Die folgende Liste enthält alle Metriken und ihre möglichen Werte. Ein `Server-Timing` Header enthält je nach Art der Anfrage und Antwort nur einige dieser Metriken CloudFront.

Einige dieser Metriken sind nur mit einem Namen (ohne Wert) im Header `Server-Timing` enthalten. Andere bestehen aus einem Namen und einem Wert. Wenn eine Metrik einen Wert aufweist, werden Name und Wert durch ein Semikolon (`;`) getrennt. Wenn der Header mehr als eine Metrik enthält, werden die Metriken durch ein Komma (`,`) getrennt.

cdn-cache-hit

CloudFront hat eine Antwort aus dem Cache bereitgestellt, ohne eine Anfrage an den Ursprung zu stellen.

cdn-cache-refresh

CloudFront hat nach dem Senden einer Anfrage an den Ursprung eine Antwort aus dem Cache bereitgestellt, um zu überprüfen, ob das zwischengespeicherte Objekt noch gültig ist. In diesem Fall CloudFront wurde nicht das vollständige Objekt vom Ursprung abgerufen.

cdn-cache-miss

CloudFront hat die Antwort nicht aus dem Cache bereitgestellt. In diesem Fall wurde das vollständige Objekt vom Ursprung CloudFront angefordert, bevor die Antwort zurückgegeben wurde.

cdn-pop

Enthält einen Wert, der beschreibt, welcher CloudFront Point of Presence (POP) die Anfrage bearbeitet hat.

cdn-rid

Enthält einen Wert mit dem CloudFront eindeutigen Bezeichner für die Anforderung. Sie können diese Anforderungskennung (Request Identifier, RID) bei der Fehlerbehebung mit dem Support verwenden.

cdn-hit-layer

Diese Metrik ist vorhanden, wenn CloudFront eine Antwort aus dem Cache bereitgestellt wird, ohne eine Anfrage an den Ursprung zu stellen. Sie enthält einen der folgenden Werte:

- EDGE — CloudFront hat die zwischengespeicherte Antwort von einem POP-Speicherort bereitgestellt.
- REC — CloudFront hat die zwischengespeicherte Antwort von einem regionalen Edge-Cache-Standort (REC) bereitgestellt.
- Origin Shield — CloudFront hat die zwischengespeicherte Antwort von der REC bereitgestellt, die als Origin Shield fungiert.

cdn-upstream-layer

Wenn CloudFront das vollständige Objekt vom Ursprung aus angefordert wird, ist diese Metrik vorhanden und enthält einen der folgenden Werte:

- EDGE – Ein POP-Standort hat die Anforderung direkt an den Ursprung gesendet.
- REC – Ein REC-Standort hat die Anforderung direkt an den Ursprung gesendet.
- Origin Shield – Der REC, der als Origin Shield fungiert, hat die Anforderung direkt an den Ursprung gesendet.

cdn-upstream-dns

Enthält einen Wert mit der Anzahl der Millisekunden, die für das Abrufen des DNS-Datensatzes für den Ursprung aufgewendet wurden. Ein Wert von Null (0) gibt an, dass CloudFront ein zwischengespeichertes DNS-Ergebnis oder eine bestehende Verbindung wiederverwendet wurde.

cdn-upstream-connect

Enthält einen Wert mit der Anzahl der Millisekunden zwischen dem Abschluss der DNS-Anforderung des Ursprungs und dem Abschluss einer TCP-Verbindung (und ggf. TLS-Verbindung) zum Ursprung. Der Wert Null (0) gibt an, dass eine bestehende Verbindung CloudFront wiederverwendet wurde.

cdn-upstream-fbl

Enthält einen Wert mit der Anzahl der Millisekunden zwischen dem Abschluss der HTTP-Anforderung des Ursprungs und dem Zeitpunkt, an dem das erste Byte in der Antwort vom Ursprung empfangen wird (Latenz des ersten Byte).

cdn-downstream-fbl

Enthält einen Wert mit der Anzahl der Millisekunden zwischen dem Zeitpunkt, an dem der Edge-Standort die Anforderung vollständig empfangen hat, und dem Zeitpunkt, an dem das erste Byte der Antwort an den Viewer gesendet wird.

Beispiele für Server-Timing-Header

Im Folgenden finden Sie Beispiele für einen `Server-Timing` Header, den ein Betrachter erhalten könnte, CloudFront wenn die `Server-Timing` Header-Einstellung aktiviert ist.

Example – Cache-Fehler

Das folgende Beispiel zeigt einen `Server-Timing` Header, den ein Betrachter erhalten könnte, wenn sich das angeforderte Objekt nicht im CloudFront Cache befindet.

```
Server-Timing: cdn-upstream-layer;desc="EDGE",cdn-upstream-dns;dur=0,cdn-upstream-
connect;dur=114,cdn-upstream-fbl;dur=177,cdn-cache-miss,cdn-pop;desc="PHX50-C2",cdn-
```

```
rid;desc="yNPsyYn7skvTzwWkq3Wcc8Nj_foxUjQUe9H1ifslzWhb0w7aLbFvGg==",cdn-downstream-
fbl;dur=436
```

Dieser `Server-Timing`-Header gibt Folgendes an:

- Die ursprüngliche Anfrage wurde von einem POP-Standort (CloudFront Point of Presence) (`cdn-upstream-layer;desc="EDGE"`) aus gesendet.
- CloudFront hat ein zwischengespeichertes DNS-Ergebnis für den Ursprung (`cdn-upstream-dns;dur=0`) verwendet.
- Es dauerte 114 Millisekunden CloudFront , bis die TCP-Verbindung (und gegebenenfalls TLS) zum Ursprung () hergestellt war. `cdn-upstream-connect;dur=114`
- Nach Abschluss der Anfrage () dauerte es 177 Millisekunden, CloudFront bis das erste Byte der Antwort vom Ursprung empfangen wurde. `cdn-upstream-fbl;dur=177`
- Das angeforderte Objekt befand sich nicht im CloudFront Cache (). `cdn-cache-miss`
- Die Anforderung wurde an dem durch den Code `PHX50-C2` identifizierten Edge-Standort empfangen (`cdn-pop;desc="PHX50-C2"`).
- Die CloudFront eindeutige ID für diese Anfrage war `yNPsyYn7skvTzwWkq3Wcc8Nj_foxUjQUe9H1ifslzWhb0w7aLbFvGg==` (`cdn-rid;desc="yNPsyYn7skvTzwWkq3Wcc8Nj_foxUjQUe9H1ifslzWhb0w7aLbFvGg=="`).
- Es dauerte 436 Millisekunden, CloudFront bis das erste Byte der Antwort an den Betrachter gesendet wurde, nachdem die Zuschaueranfrage () empfangen worden war. `cdn-downstream-fbl;dur=436`

Example – Cache-Treffer

Das folgende Beispiel zeigt einen `Server-Timing` Header, den ein Betrachter erhalten könnte, wenn sich das angeforderte Objekt im Cache befindet. CloudFront

```
Server-Timing: cdn-cache-hit,cdn-pop;desc="SEA19-C1",cdn-
rid;desc="nQBz4aJU2kP9iC3KHEq7vFxfMozu-VYBwGzkW9diOpeVc7xsrLKj-g==",cdn-hit-
layer;desc="REC",cdn-downstream-fbl;dur=137
```

Dieser `Server-Timing`-Header gibt Folgendes an:

- Das angeforderte Objekt war im Cache vorhanden (`cdn-cache-hit`).

- Die Anforderung wurde an dem durch den Code `SEA19-C1` identifizierten Edge-Standort empfangen (`cdn-pop;desc="SEA19-C1"`).
- Die CloudFront eindeutige ID für diese Anfrage war `nQBz4aJU2kP9iC3KHEq7vFxfMozu-VYBwGzkW9diOpeVc7xsrLKj-g==` (`cdn-rid;desc="nQBz4aJU2kP9iC3KHEq7vFxfMozu-VYBwGzkW9diOpeVc7xsrLKj-g=="`).
- Das angeforderte Objekt wurde an einem REC-Standort (regionaler Edge-Cache) zwischengespeichert (`cdn-hit-layer;desc="REC"`).
- Es dauerte 137 Millisekunden, CloudFront bis das erste Byte der Antwort an den Betrachter gesendet wurde, nachdem die Zuschaueranfrage () empfangen worden war. `cdn-downstream-fbl;dur=137`

Richtlinien für Antwort-Header erstellen

Sie können eine Antwort-Header-Richtlinie verwenden, um die HTTP-Header anzugeben, die Amazon in HTTP-Antworten CloudFront hinzufügt oder entfernt. Weitere Informationen zu Antwort-Header-Richtlinien und zu Gründen für ihre Verwendung finden Sie unter Fügen Sie Antwort-Header mit einer Richtlinie hinzu oder entfernen Sie sie.

Sie können in der Konsole eine Richtlinie für Antwort-Header erstellen. CloudFront Oder Sie können eine erstellen AWS CloudFormation, indem Sie die AWS Command Line Interface (AWS CLI) oder die CloudFront API verwenden. Nachdem Sie eine Response-Header-Richtlinie erstellt haben, fügen Sie sie einem oder mehreren Cache-Verhalten in einer CloudFront Distribution hinzu.

Prüfen Sie, ob eine der verwalteten Antwort-Header-Richtlinien für Ihren Anwendungsfall geeignet ist, bevor Sie eine benutzerdefinierte Antwort-Header-Richtlinie erstellen. Wenn dies der Fall ist, können Sie sie an Ihr Cache-Verhalten anfügen. Auf diese Weise müssen Sie keine eigene Antwort-Header-Richtlinie erstellen oder verwalten.

Console

So erstellen Sie eine Antwort-Header-Richtlinie (Konsole):

1. Melden Sie sich bei der an AWS Management Console und wechseln Sie dann auf der Seite Richtlinien in der CloudFront Konsole unter zur Registerkarte Antwortheader. https://console.aws.amazon.com/cloudfront/v4/home#/policies/responseHeaders
2. Wählen Sie Create response headers policy (Antwort-Header-Richtlinie erstellen) aus.

3. Gehen Sie im Formular Create response headers policy (Antwort-Header-Richtlinie erstellen) wie folgt vor:

   a. Geben Sie im Bereich Details unter Name einen Namen für die Antwort-Header-Richtlinie und (optional) unter Description (Beschreibung) eine Beschreibung des Zwecks der Richtlinie ein.

   b. Aktivieren Sie im Bereich Cross-Origin Resource Sharing (CORS) den Schalter Configure CORS (Konfigurieren von CORS) und konfigurieren Sie alle CORS-Header, die Sie der Richtlinie hinzufügen möchten. Wenn Sie möchten, dass die konfigurierten Header die vom Origin CloudFront empfangenen Header überschreiben, aktivieren Sie das Kontrollkästchen Origin-Override.

      Weitere Informationen zu den CORS-Header-Einstellungen finden Sie unter the section called “CORS-Header”.

   c. Aktivieren Sie im Bereich Security headers (Sicherheits-Header) den Schalter und konfigurieren Sie alle Sicherheits-Header, die der Richtlinie hinzugefügt werden sollen.

      Weitere Informationen zu den Sicherheits-Header-Einstellungen finden Sie unter the section called “Sicherheits-Header”.

   d. Fügen Sie im Bereich Custom headers (Benutzerdefinierte Header) alle benutzerdefinierten Header hinzu, die in der Richtlinie eingefügt werden sollen.

      Weitere Informationen zu benutzerdefinierten Header-Einstellungen finden Sie unter the section called “Benutzerdefinierte Header”.

   e. Fügen Sie im Bereich „Kopfzeilen entfernen“ die Namen aller Header hinzu, die Sie aus der Antwort des Originals entfernen und nicht in die Antwort aufnehmen möchten CloudFront , CloudFront die an die Zuschauer gesendet wird.

      Weitere Informationen zu den Einstellungen für das Entfernen von Headern finden Sie unter the section called “Entfernen von Headern”.

   f. Wählen Sie im Bereich Server-Timing header (Server-Timing-Header) die Umschalttaste Enable (Aktivieren) aus und geben Sie eine Abtastrate (eine Zahl zwischen 0 und 100, jeweils inklusive) ein.

      Weitere Informationen zum Header `Server-Timing` finden Sie unter the section called “Server-Timing-Header”.

4. Wählen Sie Create (Erstellen) aus, um die Richtlinie zu erstellen.

Nachdem Sie eine Richtlinie für Antwort-Header erstellt haben, können Sie sie an ein Cache-Verhalten in einer Verteilung anhängen. CloudFront

So fügen Sie eine Antwort-Header-Richtlinie an eine vorhandene Verteilung an (Konsole):

1. Öffnen Sie die Seite „Verteilungen“ in der CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home#/distributions
2. Wählen Sie die Verteilung aus, die Sie aktualisieren möchten, und anschließend die Registerkarte Verhaltensweisen aus.
3. Wählen Sie das Cache-Verhalten, das Sie aktualisieren möchten, und anschließend Edit (Bearbeiten) aus.

   Um ein neues Cacheverhalten zu erstellen, wählen Sie Verhalten erstellen aus.
4. Wählen Sie für Response headers policy (Antwort-Header-Richtlinie) die Richtlinie aus, die zum Cache-Verhalten hinzugefügt werden soll.
5. Wählen Sie Save changes (Änderungen speichern) aus, um das Cache-Verhalten zu aktualisieren. Wenn Sie ein neues Cache-Verhalten erstellen, wählen Sie Create behavior (Verhalten erstellen) aus.

So fügen Sie eine Antwort-Header-Richtlinie an eine neue Verteilung an (Konsole):

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie Verteilung erstellen aus.
3. Wählen Sie für Response headers policy (Antwort-Header-Richtlinie) die Richtlinie aus, die zum Cache-Verhalten hinzugefügt werden soll.
4. Wählen Sie die anderen Einstellungen für Ihre Verteilung aus. Weitere Informationen finden Sie unter the section called “Alle Verteilungseinstellungen”.
5. Wählen Sie Create distribution (Verteilung erstellen) aus, um die Verteilung zu erstellen.

AWS CloudFormation

Verwenden Sie den `AWS::CloudFront::ResponseHeadersPolicy` Ressourcentyp AWS CloudFormation, um eine Response-Header-Richtlinie mit zu erstellen. Das folgende Beispiel zeigt die AWS CloudFormation Vorlagensyntax im YAML-Format zum Erstellen einer Antwort-Header-Richtlinie.

```
Type: AWS::CloudFront::ResponseHeadersPolicy
Properties:
  ResponseHeadersPolicyConfig:
    Name: EXAMPLE-Response-Headers-Policy
    Comment: Example response headers policy for the documentation
    CorsConfig:
      AccessControlAllowCredentials: false
      AccessControlAllowHeaders:
        Items:
          - '*'
      AccessControlAllowMethods:
        Items:
          - GET
          - OPTIONS
      AccessControlAllowOrigins:
        Items:
          - https://example.com
          - https://docs.example.com
      AccessControlExposeHeaders:
        Items:
          - '*'
      AccessControlMaxAgeSec: 600
      OriginOverride: false
    CustomHeadersConfig:
      Items:
        - Header: Example-Custom-Header-1
          Value: value-1
          Override: true
        - Header: Example-Custom-Header-2
          Value: value-2
          Override: true
    SecurityHeadersConfig:
      ContentSecurityPolicy:
        ContentSecurityPolicy: default-src 'none'; img-src 'self'; script-src
 'self'; style-src 'self'; object-src 'none'; frame-ancestors 'none'
        Override: false
      ContentTypeOptions: # You don't need to specify a value for 'X-Content-Type-
Options'.
                          # Simply including it in the template sets its value to
 'nosniff'.
        Override: false
      FrameOptions:
        FrameOption: DENY
```

```
        Override: false
      ReferrerPolicy:
        ReferrerPolicy: same-origin
        Override: false
      StrictTransportSecurity:
        AccessControlMaxAgeSec: 63072000
        IncludeSubdomains: true
        Preload: true
        Override: false
      XSSProtection:
        ModeBlock: true # You can set ModeBlock to 'true' OR set a value for
ReportUri, but not both
        Protection: true
        Override: false
    ServerTimingHeadersConfig:
      Enabled: true
      SamplingRate: 50
    RemoveHeadersConfig:
      Items:
        - Header: Vary
        - Header: X-Powered-By
```

Weitere Informationen finden Sie unter AWS::CloudFront::ResponseHeadersRichtlinie im AWS CloudFormation Benutzerhandbuch.

CLI

Verwenden Sie den aws cloudfront create-response-headers-policy Befehl, um eine Response-Header-Richtlinie mit dem AWS Command Line Interface (AWS CLI) zu erstellen. Sie können eine Eingabedatei verwenden, um die Eingabeparameter für den Befehl bereitzustellen, anstatt jeden einzelnen Parameter als Befehlszeileneingabe anzugeben.

So erstellen Sie eine Antwort-Header-Richtlinie (CLI mit Eingabedatei):

1. Verwenden Sie den folgenden Befehl zum Erstellen einer Datei mit dem Namen `response-headers-policy.yaml`. Diese Datei enthält alle Eingabeparameter für den Befehl create-response-headers-policy.

```
aws cloudfront create-response-headers-policy --generate-cli-skeleton yaml-input
 > response-headers-policy.yaml
```

2. Öffnen Sie die Datei `response-headers-policy.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und geben Sie einen Richtliniennamen und die gewünschte Konfiguration der Antwort-Header-Richtlinie an. Speichern Sie anschließend die Datei.

   Weitere Informationen zu den Einstellungen für die Antwort-Header-Richtlinie finden Sie unter the section called “Verstehen Sie die Richtlinien für Antwort-Header”.

3. Verwenden Sie den folgenden Befehl, um die Antwort-Header-Richtlinie zu erstellen. Die Richtlinie, die Sie erstellen, verwendet die Eingabeparameter aus der Datei `response-headers-policy.yaml`.

   ```
   aws cloudfront create-response-headers-policy --cli-input-yaml file://response-headers-policy.yaml
   ```

   Notieren Sie den `Id`-Wert in der Befehlsausgabe. Dies ist die ID der Antwort-Header-Richtlinie. Sie benötigen ihn, um die Richtlinie an das Cache-Verhalten einer CloudFront Distribution anzuhängen.

So fügen Sie eine Antwort-Header-Richtlinie an eine vorhandene Verteilung an (CLI mit Eingabedatei):

1. Verwenden Sie den folgenden Befehl, um die Verteilungskonfiguration für die CloudFront Distribution zu speichern, die Sie aktualisieren möchten. Ersetzen Sie es *`distribution_ID`* durch die Distribution-ID.

   ```
   aws cloudfront get-distribution-config --id distribution_ID --output yaml > dist-config.yaml
   ```

2. Öffnen Sie die Datei mit dem Namen `dist-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, indem Sie die folgenden Änderungen am Cache-Verhalten vornehmen, um die Antwort-Header-Richtlinie zu verwenden.

   - Fügen Sie im Cache-Verhalten ein Feld mit dem Namen `ResponseHeadersPolicyId` hinzu. Verwenden Sie für den Wert des Feldes die ID der Antwort-Header-Richtlinie, die Sie nach dem Erstellen der Richtlinie notiert haben.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Verteilung so zu aktualisieren, dass die Antwort-Header-Richtlinie verwendet wird. Ersetzen Sie es *distribution_ID* durch die Distributions-ID.

```
aws cloudfront update-distribution --id distribution_ID --cli-input-yaml file://
dist-config.yaml
```

So fügen Sie eine Antwort-Header-Richtlinie an eine neue Verteilung an (CLI mit Eingabedatei):

1. Verwenden Sie den folgenden Befehl zum Erstellen einer Datei mit dem Namen `distribution.yaml`. Diese Datei enthält alle Eingabeparameter für den Befehl create-distribution.

```
aws cloudfront create-distribution --generate-cli-skeleton yaml-input >
 distribution.yaml
```

2. Öffnen Sie die Datei `distribution.yaml`, die Sie gerade erstellt haben. Geben Sie im Standard-Cache-Verhalten in das `ResponseHeadersPolicyId`-Feld die ID der Antwort-Header-Richtlinie ein, die Sie nach dem Erstellen der Richtlinie notiert haben. Fahren Sie mit der Bearbeitung der Datei fort, um die gewünschten Verteilungseinstellungen anzugeben, und speichern Sie die Datei, wenn Sie fertig sind.

   Weitere Informationen zu den Verteilungseinstellungen finden Sie unter Referenz für alle Verteilungseinstellungen.

3. Verwenden Sie den folgenden Befehl, um die Verteilung mit Eingabeparametern aus der Datei `distribution.yaml` zu erstellen.

```
aws cloudfront create-distribution --cli-input-yaml file://distribution.yaml
```

API

Um eine Response-Header-Richtlinie mit der CloudFront API zu erstellen, verwenden Sie CreateResponseHeadersPolicy. Weitere Informationen zu den Feldern, die Sie in diesem API-Aufruf angeben, finden Sie in the section called “Verstehen Sie die Richtlinien für Antwort-Header” und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Nachdem Sie eine Antwort-Header-Richtlinie erstellt haben, können Sie sie mit einem der folgenden API-Aufrufe an eine Cache-Verhaltensweise anfügen:

- Um es an ein Cache-Verhalten in einer vorhandenen Distribution anzuhängen, verwenden Sie UpdateDistribution.
- Um es an ein Cache-Verhalten in einer neuen Distribution anzuhängen, verwenden Sie CreateDistribution.

Geben Sie für beide API-Aufrufe die ID der Antwort-Header-Richtlinie in das Feld `ResponseHeadersPolicyId` innerhalb eines Cache-Verhaltens ein. Weitere Informationen zu den anderen Feldern, die Sie in diesen API-Aufrufen angeben, finden Sie in Referenz für alle Verteilungseinstellungen und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Richtlinien für verwaltete Antwort-Header verwenden

Mit einer CloudFront Antwort-Header-Richtlinie können Sie die HTTP-Header angeben, die Amazon CloudFront entfernt oder zu Antworten hinzufügt, die es an Zuschauer sendet. Weitere Informationen zu Antwort-Header-Richtlinien und zu Gründen für ihre Verwendung finden Sie unter Fügen Sie Antwort-Header mit einer Richtlinie hinzu oder entfernen Sie sie.

CloudFront bietet verwaltete Richtlinien für Antwort-Header, die Sie an das Cache-Verhalten in Ihren Distributionen anhängen können. CloudFront Bei Verwendung einer verwalteten Antwort-Header-Richtlinie müssen Sie keine eigene Richtlinie schreiben oder verwalten. Die verwalteten Richtlinien enthalten Sätze von HTTP-Antwort-Headern für häufige Anwendungsfälle.

Um eine verwaltete Richtlinie für Antwort-Header zu verwenden, fügen Sie sie einem Cache-Verhalten in Ihrer Verteilung an. Das Verfahren entspricht dem der Erstellung einer benutzerdefinierten Antwort-Header-Richtlinie. Anstatt jedoch eine neue Richtlinie zu erstellen, fügen Sie eine der verwalteten Richtlinien an. Sie hängen die Richtlinie entweder anhand des Namens (mit

der Konsole) oder anhand der ID (mit AWS CloudFormation AWS CLI, dem oder dem AWS SDKs) an. Die Namen und IDs sind im folgenden Abschnitt aufgeführt.

Weitere Informationen finden Sie unter the section called “Richtlinien für Antwort-Header erstellen”.

In den folgenden Themen werden die verwalteten Antwort-Header-Richtlinien beschrieben, die Sie verwenden können.

Themen

- Cors-und- SecurityHeadersPolicy
- CORS-With-Preflight
- CORS- - with-preflight-and SecurityHeadersPolicy
- SecurityHeadersPolicy
- SimpleCORS

Cors-und- SecurityHeadersPolicy

Diese Richtlinie in der Konsole anzeigen CloudFront

Verwenden Sie diese verwaltete Richtlinie, um einfache CORS-Anforderungen von jedem Ursprung zuzulassen. Diese Richtlinie fügt außerdem allen Antworten, die an Zuschauer CloudFront gesendet werden, eine Reihe von Sicherheitsheadern hinzu. Diese Richtlinie verbindet die the section called “SimpleCORS”- und the section called “SecurityHeadersPolicy”-Richtlinien in einer Richtlinie.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`e61eb60c-9c35-4d20-a928-2b84e02af89c`

Richtlinieneinstellungen

	Header-Name	Header-Wert	Override origin? (Ursprung überschreiben?)
CORS-Header:	`Access-Control-Allow-Origin`	*	Nein

	Header-Name	Header-Wert	Override origin? (Ursprung überschreiben?)
Sicherheits-Header:	`Referrer-Policy`	`strict-origin-when-cross-origin`	Nein
	`Strict-Transport-Security`	`max-age=31536000`	Nein
	`X-Content-Type-Options`	`nosniff`	Ja
	`X-Frame-Options`	`SAMEORIGIN`	Nein
	`X-XSS-Protection`	`1; mode=block`	Nein

CORS-With-Preflight

Diese Richtlinie in der CloudFront Konsole anzeigen

Mit dieser verwalteten Richtlinie werden CORS-Anforderungen von jedem Ursprung zugelassen, einschließlich Preflight-Anforderungen. CloudFront Fügt bei Preflight-Anfragen (mit der `OPTIONS` HTTP-Methode) alle drei der folgenden Header zur Antwort hinzu. Fügt bei einfachen CORS-Anfragen nur den CloudFront Header hinzu. `Access-Control-Allow-Origin`

Wenn die vom Ursprung CloudFront empfangene Antwort einen dieser Header enthält, CloudFront verwendet es den empfangenen Header (und seinen Wert) in seiner Antwort an den Betrachter. CloudFrontverwendet den Header in dieser Richtlinie nicht.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`5cc3b908-e619-4b99-88e5-2cf7f45965bd`

Richtlinieneinstellungen

	Header-Name	Header-Wert	Override origin? (Ursprung überschreiben?)
CORS-Header:	`Access-Control-Allow-Methods`	`DELETE`, `GET`, `HEAD`, `OPTIONS`, `PATCH`, `POST`, `PUT`	Nein
	`Access-Control-Allow-Origin`	`*`	
	`Access-Control-Expose-Headers`	`*`	

CORS- - with-preflight-and SecurityHeadersPolicy

Diese Richtlinie in der CloudFront Konsole anzeigen

Verwenden Sie diese verwaltete Richtlinie, um CORS-Anforderungen von jedem Ursprung zuzulassen. Dies umfasst auch Preflight-Anforderungen. Diese Richtlinie fügt außerdem allen Antworten, die an Zuschauer CloudFront gesendet werden, eine Reihe von Sicherheitsheadern hinzu. Diese Richtlinie verbindet die the section called “CORS-With-Preflight”- und the section called “SecurityHeadersPolicy”-Richtlinien in einer Richtlinie.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`eaab4381-ed33-4a86-88ca-d9558dc6cd63`

Richtlinieneinstellungen

	Header-Name	Header-Wert	Override origin? (Ursprung überschreiben?)
CORS-Header:	`Access-Control-Allow-Methods`	`DELETE, GET, HEAD, OPTIONS, PATCH, POST, PUT`	Nein
	`Access-Control-Allow-Origin`	`*`	
	`Access-Control-Expose-Headers`	`*`	
Sicherheits-Header:	`Referrer-Policy`	`strict-origin-when-cross-origin`	Nein
	`Strict-Transport-Security`	`max-age=31536000`	Nein
	`X-Content-Type-Options`	`nosniff`	Ja
	`X-Frame-Options`	`SAMEORIGIN`	Nein
	`X-XSS-Protection`	`1; mode=block`	Nein

SecurityHeadersPolicy

Diese Richtlinie in der CloudFront Konsole anzeigen

Verwenden Sie diese verwaltete Richtlinie, um allen Antworten, die an Zuschauer CloudFront gesendet werden, eine Reihe von Sicherheitsheadern hinzuzufügen. Weitere Informationen zu diesen Sicherheits-Headern finden Sie in den Websicherheitsrichtlinien von Mozilla.

Mit dieser Richtlinie für Antwort-Header werden alle Antworten CloudFront `X-Content-Type-Options: nosniff` hinzugefügt. Dies ist der Fall, wenn die Antwort, die von der Quelle CloudFront

empfangen wurde, diesen Header enthielt und wenn nicht. Für alle anderen Header in dieser Richtlinie gilt: Wenn die Antwort, die CloudFront vom Ursprung eingeht, den Header enthält, CloudFront wird der empfangene Header (und sein Wert) in der Antwort an den Betrachter verwendet. In dieser Richtlinie wird der Header nicht verwendet.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`67f7725c-6f97-4210-82d7-5512b31e9d03`

Richtlinieneinstellungen

	Header-Name	Header-Wert	Override origin? (Ursprung überschreiben?)
Sicherheits-Header:	`Referrer-Policy`	`strict-origin-when-cross-origin`	Nein
	`Strict-Transport-Security`	`max-age=31536000`	Nein
	`X-Content-Type-Options`	`nosniff`	Ja
	`X-Frame-Options`	`SAMEORIGIN`	Nein
	`X-XSS-Protection`	`1; mode=block`	Nein

SimpleCORS

Diese Richtlinie in der CloudFront Konsole anzeigen

Verwenden Sie diese verwaltete Richtlinie, um einfache CORS-Anforderungen von jedem Ursprung zuzulassen. CloudFront Fügt mit dieser Richtlinie den Header `Access-Control-Allow-Origin: *` zu allen Antworten für einfache CORS-Anfragen hinzu.

Wenn die Antwort, die vom Ursprung CloudFront empfangen wird, den `Access-Control-Allow-Origin` Header enthält, CloudFront verwendet er diesen Header (und seinen Wert) in seiner Antwort an den Betrachter. CloudFront verwendet den Header in dieser Richtlinie nicht.

Wenn Sie AWS CloudFormation die AWS CLI oder die CloudFront API verwenden, lautet die ID für diese Richtlinie:

`60669652-455b-4ae9-85a4-c4c02393f86c`

Richtlinieneinstellungen

	Header-Name	Header-Wert	Override origin? (Ursprung überschreiben?)
CORS-Header:	`Access-Control-Allow-Origin`	`*`	Nein

Verhalten von Anfragen und Antworten

In den folgenden Themen wird beschrieben, wie CloudFront mit Anfragen und Antworten umgegangen wird.

Sie erfahren, wie mit Amazon S3 oder benutzerdefinierten Ursprüngen CloudFront interagiert, mit verschiedenen HTTP-Methoden und -Headern umgeht, Statuscodes verarbeitet und Zwischenspeicherung und Fehlerantworten verwaltet.

Themen

Wie werden HTTP CloudFront - und HTTPS-Anfragen verarbeitet

CloudFront Akzeptiert für Amazon S3 S3-Ursprünge standardmäßig Anfragen sowohl im HTTP- als auch im HTTPS-Protokoll für Objekte in einer CloudFront Distribution. CloudFront leitet die Anfragen dann mit demselben Protokoll, in dem die Anfragen gestellt wurden, an Ihren Amazon S3 S3-Bucket weiter.

Bei benutzerdefinierten Ursprüngen können Sie beim Erstellen Ihrer Verteilung festlegen, wie CloudFront auf Ihren Ursprung zugreifen: nur HTTP oder entsprechend dem vom Betrachter verwendeten Protokoll. Weitere Informationen zum CloudFront Umgang mit HTTP- und HTTPS-Anfragen für benutzerdefinierte Ursprünge finden Sie unterProtokolle.

Weitere Informationen dazu, wie Sie Ihre Verteilung so einschränken, dass Endbenutzer nur mit HTTPS auf Objekte zugreifen können, finden Sie unter Verwenden Sie HTTPS mit CloudFront.

> **Note**
>
> Die Gebühr für HTTPS-Anfragen ist höher als die Gebühr für HTTP-Anfragen. Weitere Informationen zu den Abrechnungstarifen finden Sie unter CloudFront Preise.

Verhalten von Anfragen und Antworten für Amazon-S3-Ursprünge

In den folgenden Abschnitten erfahren Sie, wie Anfragen und Antworten CloudFront verarbeitet werden, wenn Sie Amazon S3 als Ausgangspunkt verwenden:

Themen

Wie CloudFront verarbeitet und leitet Anfragen an Ihren Amazon S3 S3-Ursprung weiter

Erfahren Sie, wie Zuschaueranfragen CloudFront verarbeitet und die Anfragen an Ihren Amazon S3 S3-Ursprung weiterleitet.

Inhalt

- Abfragezeichenfolgen
- Timeout der Ursprungsverbindung und Verbindungsversuche zum Ursprung
- Ursprungs-Reaktions-Timeout
- Gleichzeitige Anfragen für dasselbe Objekt (Zusammenfassung von Anfragen)

Cache-Dauer und Mindest-TTL

Um zu kontrollieren, wie lange Ihre Objekte in einem CloudFront Cache bleiben, bevor sie CloudFront eine weitere Anfrage an Ihren Ursprung weiterleiten, können Sie:

- Ihren Ursprungsserver so konfigurieren, dass jedem Objekt ein `Cache-Control`- oder `Expires`-Header-Feld hinzugefügt wird
- Geben Sie einen Wert für Minimale TTL in CloudFront Cache-Verhalten an.
- Den Standardwert von 24 Stunden verwenden

Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Client-IP-Adressen

Wenn ein Viewer eine Anfrage an sendet CloudFront und keinen Anforderungsheader enthält, CloudFront ruft er die IP-Adresse des Betrachters aus der TCP-Verbindung ab, fügt einen `X-Forwarded-For` Header hinzu, der die IP-Adresse enthält, und leitet die Anfrage an den Ursprung weiter. `X-Forwarded-For` Wenn CloudFront z. B. die IP-Adresse `192.0.2.2` von der TCP-Verbindung abruft, wird der folgende Header an den Ursprungs-Server weitergeleitet:

```
X-Forwarded-For: 192.0.2.2
```

Wenn ein Betrachter eine Anfrage an den Anforderungsheader sendet CloudFront und diesen einschließt, CloudFront ruft er die IP-Adresse des Viewers aus der TCP-Verbindung ab, hängt sie an das Ende des `X-Forwarded-For` Headers an und leitet die Anfrage an den Ursprung weiter. `X-Forwarded-For` Wenn die Viewer-Anfrage beispielsweise die IP-Adresse der TCP-Verbindung enthält `X-Forwarded-For: 192.0.2.4,192.0.2.3` und diese CloudFront `192.0.2.2` abruft, leitet sie den folgenden Header an den Ursprung weiter:

```
X-Forwarded-For: 192.0.2.4,192.0.2.3,192.0.2.2
```

> **Note**
>
> Der `X-Forwarded-For` Header enthält IPv4 Adressen (wie 192.0.2.44) und IPv6 Adressen (wie 2001:0 db 8:85 a3: :8a2e: 0370:7334).

Bedingte GET-Anfragen

Wenn eine Anfrage für ein Objekt CloudFront empfangen wird, das aus einem Edge-Cache abgelaufen ist, leitet es die Anfrage an den Amazon S3-Ursprung weiter, um die neueste Version des Objekts abzurufen oder um eine Bestätigung von Amazon S3 zu erhalten, dass der CloudFront Edge-Cache bereits über die neueste Version verfügt. Als Amazon S3 das Objekt ursprünglich an sendete CloudFront, enthielt es einen `ETag` Wert und einen `LastModified` Wert in der Antwort. Fügt in der neuen Anfrage CloudFront , die an Amazon S3 weitergeleitet wird, einen oder beide der folgenden Header CloudFront hinzu:

- Einen `If-Match`- oder `If-None-Match`-Header mit dem `ETag`-Wert für die abgelaufene Version des Objekts
- Einen `If-Modified-Since`-Header mit dem `LastModified`-Wert für die abgelaufene Version des Objekts

Amazon S3 verwendet diese Informationen, um festzustellen, ob das Objekt aktualisiert wurde und ob daher das gesamte Objekt an CloudFront oder nur ein HTTP-304-Statuscode (nicht geändert) zurückgegeben werden soll.

Cookies

Amazon S3 verarbeitet keine Cookies. Wenn Sie ein Cache-Verhalten so konfigurieren, dass Cookies an einen Amazon S3-Ursprung weitergeleitet werden CloudFront , werden die Cookies weitergeleitet, Amazon S3 ignoriert sie jedoch. Alle zukünftigen Anfragen zu demselben Objekt werden mithilfe des im Cache vorhandenen Objekts bedient – unabhängig davon, ob Sie Änderungen an den Cookies vornehmen.

Cross-Origin Resource Sharing (CORS)

Wenn Sie CloudFront die ursprungsübergreifenden Amazon S3-Einstellungen für die gemeinsame Nutzung von Ressourcen respektieren möchten, konfigurieren Sie die Konfiguration so, CloudFront

dass ausgewählte Header an Amazon S3 weitergeleitet werden. Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern.

GET-Anfragen mit Anfragetext

Wenn eine `GET` Viewer-Anfrage einen Text enthält, wird der HTTP-Statuscode 403 (Forbidden) an den Betrachter CloudFront zurückgegeben.

HTTP-Methoden

Wenn Sie so konfigurieren CloudFront , dass alle HTTP-Methoden verarbeitet werden, die es unterstützt, CloudFront akzeptiert es die folgenden Anfragen von Zuschauern und leitet sie an Ihren Amazon S3 S3-Ursprung weiter:

- `DELETE`
- `GET`
- `HEAD`
- `OPTIONS`
- `PATCH`
- `POST`
- `PUT`

CloudFront speichert Antworten auf `GET` und `HEAD` Anfragen immer im Cache. Sie können auch so konfigurieren CloudFront , dass Antworten auf `OPTIONS` Anfragen zwischengespeichert werden. CloudFront speichert keine Antworten auf Anfragen, die die anderen Methoden verwenden.

Wenn Sie mehrteilige Uploads verwenden möchten, um Objekte zu einem Amazon S3 S3-Bucket hinzuzufügen, müssen Sie Ihrer Distribution eine CloudFront Origin-Zugriffskontrolle (OAC) hinzufügen und dem OAC die erforderlichen Berechtigungen erteilen. Weitere Informationen finden Sie unter the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”.

> **Important**
>
> Wenn Sie so konfigurieren CloudFront , dass alle CloudFront unterstützten HTTP-Methoden akzeptiert und an Amazon S3 weitergeleitet werden, müssen Sie ein CloudFront OAC erstellen, um den Zugriff auf Ihre Amazon S3 S3-Inhalte einzuschränken und dem OAC die erforderlichen Berechtigungen zu erteilen. Wenn Sie beispielsweise so konfigurieren

CloudFront , dass diese Methoden akzeptiert und weitergeleitet werden, weil Sie die PUT Methode verwenden möchten, müssen Sie die Amazon S3 S3-Bucket-Richtlinien so konfigurieren, dass DELETE Anfragen angemessen behandelt werden, sodass Zuschauer keine Ressourcen löschen können, die Sie nicht möchten. Weitere Informationen finden Sie unter the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”.

Informationen zu den von Amazon S3 unterstützten Operationen finden Sie in der Amazon S3-Dokumentation.

HTTP-Anforderungsheader, die CloudFront entfernt oder aktualisiert werden

CloudFront entfernt oder aktualisiert einige Header, bevor Anfragen an Ihren Amazon S3 S3-Ursprung weitergeleitet werden. Für die meisten Header ist dieses Verhalten dasselbe wie für benutzerdefinierte Ursprünge. Eine vollständige Liste der HTTP-Anforderungsheader und deren Verarbeitung finden CloudFront Sie unter. Header und CloudFront Verhalten von HTTP-Anfragen (benutzerdefiniert und Amazon S3 S3-Ursprünge)

Maximale Länge einer Anfrage und maximale Länge einer URL

Die maximale Länge einer Anfrage – einschließlich des Pfads, der Abfragezeichenfolge (falls vorhanden) und der Header – beträgt 20 480 Byte.

CloudFront konstruiert aus der Anfrage eine URL. Die maximale Länge dieser URL beträgt 8 192 Byte.

Wenn eine Anfrage oder eine URL die maximale Länge überschreitet, CloudFront gibt es den HTTP-Statuscode 413 (Request Entity Too Large) an den Viewer zurück und beendet dann die TCP-Verbindung zum Viewer.

OCSP-Stapling

Wenn ein Betrachter eine HTTPS-Anfrage für ein Objekt sendet CloudFront oder der Betrachter bei der Zertifizierungsstelle (CA) bestätigen muss, dass das SSL-Zertifikat für die Domain nicht gesperrt wurde. OCSP-Stapling beschleunigt die Zertifikatsvalidierung, da CloudFront das Zertifikat validiert und die Antwort von der CA zwischengespeichert werden kann, sodass der Client das Zertifikat nicht direkt bei der CA validieren muss.

Die Leistungsverbesserung von OCSP-Stapling ist ausgeprägter, wenn CloudFront viele HTTPS-Anfragen für Objekte in derselben Domäne eingehen. Jeder Server an einem CloudFront -Edge-

Standort muss eine separate Validierungsanfrage senden. Wenn viele HTTPS-Anfragen für dieselbe Domain eingehen, CloudFront erhält jeder Server am Edge-Standort bald eine Antwort von der CA, die er im SSL-Handshake an ein Paket heften kann. Wenn der Betrachter davon überzeugt ist, dass das Zertifikat gültig ist, CloudFront kann er das angeforderte Objekt bereitstellen. Wenn Ihre Verteilung nicht viel Datenverkehr an einem CloudFront -Edge-Standort generiert, werden neue Anfragen mit einer höheren Wahrscheinlichkeit an einen Server weitergeleitet, der das Zertifikat noch nicht bei der CA validiert hat. In diesem Fall führt der Betrachter den Validierungsschritt separat durch und der CloudFront Server stellt das Objekt bereit. Dieser CloudFront Server sendet auch eine Überprüfungsanfrage an die CA. Wenn er also das nächste Mal eine Anfrage erhält, die denselben Domainnamen enthält, erhält er eine Validierungsantwort von der CA.

Protokolle

CloudFront leitet HTTP- oder HTTPS-Anfragen auf der Grundlage des Protokolls der Viewer-Anfrage, entweder HTTP oder HTTPS, an den Ursprungsserver weiter.

> **Important**
>
> Wenn Ihr Amazon S3-Bucket als Website-Endpunkt konfiguriert ist, können Sie die Verwendung von HTTPS für die Kommunikation mit Ihrem Ursprung nicht konfigurieren CloudFront , da Amazon S3 in dieser Konfiguration keine HTTPS-Verbindungen unterstützt.

Abfragezeichenfolgen

Sie können konfigurieren, ob CloudFront Abfragezeichenfolgenparameter an Ihren Amazon S3-Ursprung weitergeleitet werden. Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern.

Timeout der Ursprungsverbindung und Verbindungsversuche zum Ursprung

Das Timeout für die Origin-Verbindung ist die Anzahl der Sekunden, die beim Versuch, eine Verbindung zum CloudFront Ursprung herzustellen, gewartet wird.

Die Anzahl der Versuche, eine Verbindung zum Ursprung herzustellen, gibt an, wie oft CloudFront versucht wird, eine Verbindung zum Ursprung herzustellen.

Zusammen bestimmen diese Einstellungen, wie lange CloudFront versucht wird, eine Verbindung zum Ursprung herzustellen, bevor ein Failover zum sekundären Ursprung erfolgt (im Fall einer Ursprungsgruppe) oder eine Fehlermeldung an den Viewer zurückgegeben wird. CloudFront

Wartet standardmäßig bis zu 30 Sekunden (3 Versuche à 10 Sekunden), bevor versucht wird, eine Verbindung zum sekundären Ursprung herzustellen, oder eine Fehlermeldung zurückgegeben wird. Sie können diese Zeit reduzieren, indem Sie ein kürzeres Verbindungs-Timeout, weniger Versuche oder beides angeben.

Weitere Informationen finden Sie unter Kontrolliere Timeouts und Versuche bei der Herkunft.

Ursprungs-Reaktions-Timeout

Das Ursprungs-Reaktions-Timeout, das auch als Ursprungs-Lese-Timeout oder Ursprungs-Anforderungs-Timeout bezeichnet wird, gilt für Folgendes:

- Die Zeitspanne in Sekunden, die auf eine Antwort CloudFront wartet, nachdem eine Anfrage an den Ursprung weitergeleitet wurde.
- Die Zeitspanne in Sekunden, die nach dem Empfang eines Antwortpakets vom Ursprung und vor dem Empfang des nächsten Pakets CloudFront gewartet wird.

CloudFront Das Verhalten hängt von der HTTP-Methode der Viewer-Anfrage ab:

- `GET`und `HEAD` Anfragen — Wenn der Ursprung nicht innerhalb von 30 Sekunden reagiert oder 30 Sekunden lang nicht mehr reagiert, wird CloudFront die Verbindung unterbrochen. Wenn die angegebene Anzahl der ursprünglichen Verbindungsversuche mehr als 1 beträgt, wird erneut CloudFront versucht, eine vollständige Antwort zu erhalten. CloudFront versucht bis zu 3 Mal, je nach dem Wert der Einstellung für ursprüngliche Verbindungsversuche. Wenn der Ursprung beim letzten Versuch keine Antwort sendet, unternimmt CloudFront erst dann einen weiteren Versuch, wenn die nächste Anfrage für Inhalte auf demselben Ursprung empfangen wird.
- `DELETE`,`OPTIONS`, `PATCH`PUT, und `POST` Anfragen — Wenn der Absender nicht innerhalb von 30 Sekunden antwortet, wird die CloudFront Verbindung unterbrochen und es wird nicht erneut versucht, den Ursprung zu kontaktieren. Der Client kann die Anfrage erneut senden, falls erforderlich.

Sie können das Reaktions-Timeout für einen Amazon S3-Ursprung (ein S3-Bucket, der nicht mit statischem Website-Hosting konfiguriert ist) nicht ändern.

Gleichzeitige Anfragen für dasselbe Objekt (Zusammenfassung von Anfragen)

Wenn ein CloudFront Edge-Standort eine Anfrage für ein Objekt erhält und sich das Objekt nicht im Cache befindet oder das zwischengespeicherte Objekt abgelaufen ist, wird die Anfrage CloudFront

sofort an den Ursprung gesendet. Wenn es jedoch gleichzeitige Anfragen für dasselbe Objekt gibt, d. h. wenn zusätzliche Anfragen für dasselbe Objekt (mit demselben Cache-Schlüssel) am Edge-Standort ankommen, bevor die Antwort auf die erste Anfrage CloudFront empfangen wird, wird eine CloudFront Pause eingelegt, bevor die zusätzlichen Anfragen an den Ursprung weitergeleitet werden. Diese kurze Pause trägt dazu bei, die Belastung des Ursprungs zu reduzieren. CloudFront sendet die Antwort der ursprünglichen Anfrage auf alle Anfragen, die während der Pause eingegangen sind. Dies wird als Request Collapsing (Zusammenfassung von Anfragen) bezeichnet. In den CloudFront Protokollen wird die erste Anfrage `Miss` im `x-edge-result-type` Feld als eine identifiziert, und die ausgeblendeten Anfragen werden als a gekennzeichnet. `Hit` Weitere Hinweise zu CloudFront Protokollen finden Sie unterthe section called “CloudFront und Edge-Funktionsprotokollierung”.

CloudFront reduziert nur Anfragen, die sich einen Cache-Schlüssel teilen. Wenn die zusätzlichen Anfragen nicht denselben Cache-Schlüssel verwenden, weil Sie beispielsweise so konfiguriert haben, dass der Cache CloudFront auf der Grundlage von Anforderungsheadern, Cookies oder Abfragezeichenfolgen gespeichert wird, werden alle Anfragen mit einem eindeutigen Cache-Schlüssel an Ihren Ursprung CloudFront weitergeleitet.

Wenn Sie verhindern möchten, dass alle Anfragen kollabiert werden, können Sie die verwaltete Cache-Richtlinie verwenden`CachingDisabled`, die auch das Zwischenspeichern verhindert. Weitere Informationen finden Sie unter Verwaltete Cache-Richtlinien verwenden.

Wenn Sie verhindern möchten, dass Anfragen für bestimmte Objekte reduziert werden, können Sie die Mindest-TTL für das Cache-Verhalten auf 0 setzen und den Ursprung so konfigurieren, dass er`Cache-Control: private`,,`Cache-Control: no-store`, `Cache-Control: no-cache` oder sendet. `Cache-Control: max-age=0` `Cache-Control: s-maxage=0` Diese Konfigurationen erhöhen die Belastung Ihres Ursprungs und führen zu zusätzlicher Latenz für gleichzeitige Anfragen, die angehalten werden, während auf die Antwort auf die CloudFront erste Anfrage gewartet wird.

> **Important**
>
> Unterstützt derzeit CloudFront nicht das Zusammenklappen von Anfragen, wenn Sie die Cookie-Weiterleitung in der Cache-Richtlinie, der ursprünglichen Anforderungsrichtlinie oder den Legacy-Cache-Einstellungen aktivieren.

So CloudFront werden Antworten von Ihrem Amazon S3 S3-Absender verarbeitet

Erfahren Sie, wie Antworten von Ihrem Amazon S3 S3-Absender CloudFront verarbeitet werden.

Inhalt

Abgebrochene Anfragen

Wenn sich ein Objekt nicht im Edge-Cache befindet und ein Betrachter eine Sitzung beendet (z. B. einen Browser schließt), nachdem CloudFront er das Objekt von Ihrem Ursprung abgerufen hat, aber bevor es das angeforderte Objekt liefern kann, wird das Objekt CloudFront nicht am Edge-Standort zwischengespeichert.

HTTP-Antwort-Header, die CloudFront entfernt oder aktualisiert werden

CloudFront entfernt oder aktualisiert die folgenden Header-Felder, bevor die Antwort von Ihrem Amazon S3 S3-Ursprung an den Viewer weitergeleitet wird:

- `X-Amz-Id-2`
- `X-Amz-Request-Id`
- `Set-Cookie`— Wenn Sie CloudFront die Weiterleitung von Cookies konfigurieren, wird das `Set-Cookie` Header-Feld an Clients weitergeleitet. Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.
- `Trailer`
- `Transfer-Encoding`— Wenn Ihr Amazon S3 S3-Ursprung dieses Header-Feld zurückgibt, CloudFront setzt er den Wert auf, `chunked` bevor die Antwort an den Betrachter zurückgesendet wird.
- `Upgrade`
- `Via`— CloudFront setzt den Wert in der Antwort an den Zuschauer auf den folgenden Wert:

```
Via: http-version alphanumeric-string.cloudfront.net (CloudFront)
```

Der Wert ist beispielsweise in etwa wie folgt:

```
Via: 1.1 1026589cc7887e7a0dc7827b4example.cloudfront.net (CloudFront)
```

Maximale Dateigröße, die zwischengespeichert werden kann

Die maximale Größe eines Antworttextes, der in seinem Cache CloudFront gespeichert wird, beträgt 50 GB. Dazu gehören auch Antworten für aufgeteilte Übertragungen, in denen kein Wert für die `Content-Length`-Kopfzeile angegeben wurde.

Sie können ein Objekt zwischenspeichern, das größer als diese Größe ist, indem Sie Bereichsanforderungen verwenden, um die Objekte in Teilen anzufordern, die jeweils 50 GB oder weniger groß sind. CloudFront CloudFrontspeichert diese Teile im Cache, da jeder von ihnen 50 GB oder weniger groß ist. Nachdem der Viewer alle Teile des Objekts abgerufen hat, kann er das ursprüngliche, größere Objekt rekonstruieren. Weitere Informationen finden Sie unter Verwenden von Bereichsanforderungen zum Zwischenspeichern großer Objekte.

Umleitungen

Sie können einen Amazon S3-Bucket so konfigurieren, dass alle Anfragen an einen anderen Host-Namen umgeleitet werden. Dabei kann es sich um einen anderen Amazon S3-Bucket oder um einen HTTP-Server handeln. Wenn Sie einen Bucket so konfigurieren, dass er alle Anfragen umleitet, und wenn der Bucket der Ursprung für eine CloudFront Distribution ist, empfehlen wir, den Bucket so zu konfigurieren, dass er alle Anfragen entweder mit dem Domainnamen für die Distribution (z. B. d111111abcdef8.cloudfront.net) oder einem alternativen Domainnamen (einem CNAME), der einer Distribution zugeordnet ist (z. B. example.com), umleitet. CloudFront Andernfalls werden Viewer-Anfragen umgangen CloudFront und die Objekte werden direkt vom neuen Ursprung aus bedient.

> **Note**
>
> Wenn Sie Viewer-Anforderungen an einen alternativen Domain-Namen umleiten, müssen Sie auch den DNS-Service für Ihre Domain aktualisieren, indem Sie einen CNAME-Datensatz hinzufügen. Weitere Informationen finden Sie unter Verwenden Sie Benutzerdefiniert, URLs indem Sie alternative Domainnamen hinzufügen (CNAMEs).

Wenn Sie einen Bucket so konfigurieren, dass er alle Anfragen umleitet, geschieht Folgendes:

1. Ein Betrachter (z. B. ein Browser) fordert ein Objekt von an CloudFront.
2. CloudFront leitet die Anfrage an den Amazon S3 S3-Bucket weiter, der der Ursprung Ihrer Distribution ist.
3. Amazon S3 gibt einen HTTP-Statuscode 301 (dauerhaft verschoben) sowie den neuen Speicherort zurück.
4. CloudFront speichert den Umleitungsstatuscode und den neuen Standort im Cache und gibt die Werte an den Betrachter zurück. CloudFront folgt nicht der Weiterleitung, um das Objekt vom neuen Standort abzurufen.
5. Der Betrachter sendet eine weitere Anfrage für das Objekt, aber diesmal gibt der Betrachter den neuen Speicherort an, von dem es abgerufen wurde CloudFront:

    - Wenn der Amazon S3 S3-Bucket alle Anfragen an eine CloudFront Distribution umleitet und dabei entweder den Domainnamen für die Distribution oder einen alternativen Domainnamen verwendet, CloudFront fordert er das Objekt vom Amazon S3 S3-Bucket oder vom HTTP-Server am neuen Standort an. Wenn der neue Speicherort das Objekt zurückgibt, wird es an den Viewer CloudFront zurückgegeben und an einem Edge-Speicherort zwischengespeichert.
    - Wenn der Amazon S3 S3-Bucket Anfragen an einen anderen Standort umleitet, wird die zweite Anfrage umgangen. CloudFront Der Amazon S3 S3-Bucket oder der HTTP-Server am neuen Standort gibt das Objekt direkt an den Viewer zurück, sodass das Objekt niemals in einem CloudFront Edge-Cache zwischengespeichert wird.

Verhalten von Anfragen und Antworten für benutzerdefinierte Ursprungsserver

In den folgenden Abschnitten erfahren Sie, wie Anfragen und Antworten CloudFront verarbeitet werden, wenn Sie benutzerdefinierte Ursprünge verwenden:

Themen

Wie CloudFront verarbeitet und leitet Anfragen an Ihren benutzerdefinierten Absender weiter

Erfahren Sie, wie Zuschaueranfragen CloudFront verarbeitet und die Anfragen an Ihren benutzerdefinierten Ursprung weiterleitet.

Inhalt

Authentifizierung

Wenn Sie den `Authorization` Header an Ihren Ursprung weiterleiten, können Sie Ihren Ursprungsserver so konfigurieren, dass er für die folgenden Arten von Anfragen eine Client-Authentifizierung anfordert:

- `DELETE`
- `GET`
- `HEAD`
- `PATCH`
- `PUT`
- `POST`

Für `OPTIONS` Anfragen kann die Client-Authentifizierung nur konfiguriert werden, wenn Sie die folgenden CloudFront Einstellungen verwenden:

- CloudFront ist so konfiguriert, dass der `Authorization` Header an Ihren Ursprung weitergeleitet wird
- CloudFront ist so konfiguriert, dass die Antwort auf `OPTIONS` Anfragen nicht zwischengespeichert wird

Weitere Informationen finden Sie unter So konfigurieren CloudFront , dass der Header weitergeleitet wird `Authorization`.

Sie können HTTP oder HTTPS verwenden, um Anfragen an Ihren Ursprungsserver weiterzuleiten. Weitere Informationen finden Sie unter Verwenden Sie HTTPS mit CloudFront.

Cache-Dauer und Mindest-TTL

Um zu kontrollieren, wie lange Ihre Objekte in einem CloudFront Cache bleiben, bevor CloudFront sie eine weitere Anfrage an Ihren Ursprung weiterleiten, können Sie:

- Ihren Ursprungsserver so konfigurieren, dass jedem Objekt ein `Cache-Control`- oder `Expires`-Header-Feld hinzugefügt wird
- Geben Sie einen Wert für Minimale TTL in CloudFront Cache-Verhalten an.
- Den Standardwert von 24 Stunden verwenden

Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Client-IP-Adressen

Wenn ein Viewer eine Anfrage an sendet CloudFront und keinen Anforderungsheader enthält, CloudFront ruft er die IP-Adresse des Viewers aus der TCP-Verbindung ab, fügt einen `X-Forwarded-For` Header hinzu, der die IP-Adresse enthält, und leitet die Anfrage an den Ursprung weiter. `X-Forwarded-For` Wenn CloudFront z. B. die IP-Adresse `192.0.2.2` von der TCP-Verbindung abruft, wird der folgende Header an den Ursprungs-Server weitergeleitet:

```
X-Forwarded-For: 192.0.2.2
```

Wenn ein Betrachter eine Anfrage an den Anforderungsheader sendet CloudFront und diesen einschließt, CloudFront ruft er die IP-Adresse des Viewers aus der TCP-Verbindung ab, hängt sie an das Ende des `X-Forwarded-For` Headers an und leitet die Anfrage an den Ursprung weiter. `X-Forwarded-For` Wenn die Viewer-Anfrage beispielsweise die IP-Adresse der TCP-Verbindung enthält `X-Forwarded-For: 192.0.2.4,192.0.2.3` und diese CloudFront `192.0.2.2` abruft, leitet sie den folgenden Header an den Ursprung weiter:

```
X-Forwarded-For: 192.0.2.4,192.0.2.3,192.0.2.2
```

Einige Anwendungen, wie Load Balancer (einschließlich Elastic Load Balancing), Webanwendungs-Firewalls, Reverse-Proxys, Intrusion Prevention-Systeme und API Gateway, fügen die IP-Adresse des CloudFront Edge-Servers, der die Anfrage weitergeleitet hat, an das Ende des Headers an. `X-Forwarded-For` Wenn beispielsweise `X-Forwarded-For: 192.0.2.2` in einer Anfrage CloudFront enthalten ist, dass sie an ELB weitergeleitet wird, und wenn die IP-Adresse des CloudFront Edge-Servers 192.0.2.199 lautet, enthält die Anfrage, die Ihre Instance empfängt, den folgenden Header: EC2

```
X-Forwarded-For: 192.0.2.2,192.0.2.199
```

> **Note**
>
> Der `X-Forwarded-For` Header enthält IPv4 Adressen (wie 192.0.2.44) und IPv6 Adressen (wie 2001:0 db 8:85 a3: :8a2e: 0370:7334).
> Beachten Sie auch, dass der `X-Forwarded-For` Header von jedem Knoten auf dem Pfad zum aktuellen Server geändert werden kann (). CloudFront Weitere Informationen finden Sie

im Abschnitt 8.1 unter RFC 7239. Sie können den Header auch mithilfe von CloudFront Edge-Compute-Funktionen ändern.

Clientseitige SSL-Authentifizierung

CloudFront unterstützt keine Client-Authentifizierung mit clientseitigen SSL-Zertifikaten. Wenn ein Ursprung ein clientseitiges Zertifikat anfordert, CloudFront wird die Anfrage gelöscht.

Komprimierung

Weitere Informationen finden Sie unter Komprimierte Dateien bereitstellen.

Bedingte Anforderungen

Wenn CloudFront er eine Anforderung für ein Objekt erhält, das aus einem Edge-Cache abgelaufen ist, leitet er die Anfrage an den Ursprung weiter, um entweder die neueste Version des Objekts abzurufen oder um vom Ursprung eine Bestätigung zu erhalten, dass der CloudFront Edge-Cache bereits über die neueste Version verfügt. In der Regel hat der Ursprung, an den das Objekt zuletzt gesendet wurde CloudFront, einen `ETag` Wert, einen `LastModified` Wert oder beide Werte in die Antwort aufgenommen. Fügt in der neuen Anfrage, die an den Ursprung CloudFront weiterleitet, eine oder beide der folgenden Angaben CloudFront hinzu:

- Einen `If-Match`- oder `If-None-Match`-Header mit dem `ETag`-Wert für die abgelaufene Version des Objekts
- Einen `If-Modified-Since`-Header mit dem `LastModified`-Wert für die abgelaufene Version des Objekts

Der Ursprung verwendet diese Informationen, um zu ermitteln, ob das Objekt aktualisiert wurde und ob daher das gesamte Objekt CloudFront oder nur ein HTTP 304-Statuscode (nicht geändert) zurückgegeben werden soll.

> **Note**
>
> `If-Modified-Since`und `If-None-Match` bedingte Anfragen werden nicht unterstützt, wenn die Konfiguration so konfiguriert CloudFront ist, dass Cookies (alle oder ein Teil davon) weitergeleitet werden.
> Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.

Cookies

Sie können so konfigurieren CloudFront , dass Cookies an Ihren Ursprung weitergeleitet werden. Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.

Cross-Origin Resource Sharing (CORS)

Wenn Sie die Einstellungen CloudFront für die gemeinsame Nutzung von Ressourcen zwischen verschiedenen Quellen beibehalten möchten, konfigurieren Sie die Konfiguration so, CloudFront dass der `Origin` Header an Ihren Ursprung weitergeleitet wird. Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern.

Verschlüsselung

Sie können verlangen, dass Zuschauer HTTPS verwenden, um Anfragen an Ihren benutzerdefinierten Ursprung zu senden CloudFront, CloudFront und Anfragen an Ihren benutzerdefinierten Ursprung weiterleiten müssen, indem Sie das Protokoll verwenden, das vom Betrachter verwendet wird. Weitere Informationen finden Sie in den folgenden Verteilungseinstellungen:

- Viewer-Protokollrichtlinien
- Protokoll (nur benutzerdefinierte Ursprünge)

CloudFront leitet HTTPS-Anfragen mithilfe der Protokolle SSLv3, TLSv1 .0, TLSv1 .1 und TLSv1 .2 an den Ursprungsserver weiter. Für benutzerdefinierte Ursprünge können Sie die SSL-Protokolle auswählen, die Sie für die Kommunikation mit Ihrem Ursprung verwenden CloudFront möchten:

- Wenn du die CloudFront Konsole verwendest, wähle Protokolle mithilfe der Kontrollkästchen Origin SSL Protocols aus. Weitere Informationen finden Sie unter Eine Verteilung erstellen.
- Wenn Sie die CloudFront API verwenden, geben Sie Protokolle mithilfe des `OriginSslProtocols` Elements an. Weitere Informationen finden Sie unter OriginSslProtocolsund DistributionConfigin der Amazon CloudFront API-Referenz.

Wenn der Ursprung ein Amazon S3 S3-Bucket ist, wird CloudFront immer TLSv1 .2 verwendet.

> **Important**
>
> Andere Versionen von SSL und TLS werden nicht unterstützt.

Weitere Informationen zur Verwendung von HTTPS mit CloudFront finden Sie unterVerwenden Sie HTTPS mit CloudFront. Eine Liste der Chiffren, die die HTTPS-Kommunikation zwischen Zuschauern und und zwischen CloudFront und Ihrem Absender CloudFront unterstützen CloudFront, finden Sie unter. Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront

GET-Anfragen mit Anfragetext

Wenn eine `GET` Viewer-Anfrage einen Hauptteil enthält, wird der HTTP-Statuscode 403 (Forbidden) an den Betrachter CloudFront zurückgegeben.

HTTP-Methoden

Wenn Sie so konfigurieren CloudFront , dass alle HTTP-Methoden verarbeitet werden, die es unterstützt, CloudFront akzeptiert es die folgenden Anfragen von Zuschauern und leitet sie an Ihren benutzerdefinierten Ursprung weiter:

- `DELETE`
- `GET`
- `HEAD`
- `OPTIONS`
- `PATCH`
- `POST`
- `PUT`

CloudFront speichert immer Antworten auf `GET` und `HEAD` Anfragen im Cache. Sie können auch so konfigurieren CloudFront , dass Antworten auf `OPTIONS` Anfragen zwischengespeichert werden. CloudFront speichert keine Antworten auf Anfragen, die die anderen Methoden verwenden.

Informationen zur Konfiguration Ihres benutzerdefinierten Ursprungsservers für die Verarbeitung dieser Methoden finden Sie in der Dokumentation zu Ihrem Ursprungsserver.

> **Important**
>
> Wenn Sie so konfigurieren CloudFront , dass alle CloudFront unterstützten HTTP-Methoden akzeptiert und an Ihren Ursprung weitergeleitet werden, konfigurieren Sie Ihren Ursprungsserver so, dass er alle Methoden verarbeitet. Wenn Sie beispielsweise so konfigurieren CloudFront , dass diese Methoden akzeptiert und weitergeleitet werden, weil Sie sie verwenden möchten`POST`, müssen Sie Ihren Ursprungsserver so konfigurieren, dass

er `DELETE` Anfragen entsprechend verarbeitet, sodass Zuschauer keine Ressourcen löschen können, die Sie nicht möchten. Weitere Informationen finden Sie in der Dokumentation zu Ihrem HTTP-Server.

Header und CloudFront Verhalten von HTTP-Anfragen (benutzerdefiniert und Amazon S3 S3-Ursprünge)

In der folgenden Tabelle sind HTTP-Anfrage-Header aufgelistet, die Sie sowohl an benutzerdefinierte als auch Amazon S3-Ursprünge weiterleiten können (mit Ausnahmen, auf die hingewiesen wird). Für jeden Header umfasst die Tabelle Informationen über Folgendes:

- CloudFront Verhalten, wenn Sie nicht so konfigurieren CloudFront , dass der Header an Ihren Ursprung weitergeleitet wird, was dazu führt CloudFront , dass Ihre Objekte auf der Grundlage von Header-Werten zwischengespeichert werden.
- Ob Sie so konfigurieren können CloudFront , dass Objekte auf der Grundlage von Header-Werten für diesen Header zwischengespeichert werden.

  Sie können so konfigurieren CloudFront , dass Objekte auf der Grundlage von Werten in den `User-Agent` Kopfzeilen `Date` und zwischengespeichert werden, wir empfehlen dies jedoch nicht. Diese Header haben viele mögliche Werte, und das Zwischenspeichern auf der Grundlage ihrer Werte würde dazu führen, dass deutlich mehr Anfragen CloudFront an Ihren Ursprung weitergeleitet werden.

Weitere Informationen zum Zwischenspeichern auf Basis von Header-Werten finden Sie unter Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern.

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischenspeichern auf Basis von Header-Werten wird unterstützt
		Ja

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischens peichern auf Basis von Header-We rten wird unterstützt
Anderweitig definierte Header	Ältere Cache-Einstellungen — CloudFront leitet die Header an Ihren Ursprung weiter.	
`Accept`	CloudFront entfernt den Header.	Ja
`Accept-Charset`	CloudFront entfernt den Header.	Ja
`Accept-Encoding`	Wenn der Wert `gzip` oder enthält`br`, CloudFront leitet er einen normalisierten `Accept-Encoding` Header an Ihren Ursprung weiter. Weitere Informationen erhalten Sie unter Komprimierungsunterstützung und Komprimierte Dateien bereitstellen.	Ja
`Accept-Language`	CloudFront entfernt den Header.	Ja

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischens peichern auf Basis von Header-We rten wird unterstützt
`Authorization`	• `GET`und `HEAD` Anfragen — CloudFront entfernt das `Authorization` Header-Feld, bevor die Anfrage an Ihren Absender weitergeleitet wird. • `OPTIONS`Anfragen — CloudFront entfernt das `Authorization` Header-Feld, bevor die Anfrage an Ihren Absender weitergeleitet wird, wenn Sie so konfiguriert haben CloudFront , dass Antworten auf `OPTIONS` Anfragen zwischeng espeichert werden. CloudFront leitet das `Authorization` Header-Feld an Ihren Ursprung weiter, wenn Sie nicht so konfiguriert habenCloudFront , dass Antworten auf OPTIONS-Anfragen zwischengespeichert werden. • `DELETE`, `PATCH``POST`, und `PUT` Anfragen — entfernt das Header-Feld CloudFront nicht, bevor die Anfrage an Ihren Absender weitergeleitet wird.	Ja
`Cache-Control`	CloudFront leitet den Header an Ihren Ursprung weiter.	Nein

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischenspeichern auf Basis von Header-Werten wird unterstützt
`CloudFront-Forwarded-Proto`	CloudFront fügt den Header nicht hinzu, bevor die Anfrage an Ihren Ursprung weitergeleitet wurde. Weitere Informationen finden Sie unter Konfigurieren Sie das Caching auf der Grundlage des Protokolls der Anfrage.	Ja
`CloudFront-Is-Desktop-Viewer`	CloudFront fügt den Header nicht hinzu, bevor die Anfrage an Ihren Absender weitergeleitet wird. Weitere Informationen finden Sie unter Konfigurieren Sie das Caching auf der Grundlage des Gerätetyps.	Ja
`CloudFront-Is-Mobile-Viewer`	CloudFront fügt den Header nicht hinzu, bevor die Anfrage an Ihren Absender weitergeleitet wird. Weitere Informationen finden Sie unter Konfigurieren Sie das Caching auf der Grundlage des Gerätetyps.	Ja
`CloudFront-Is-Tablet-Viewer`	CloudFront fügt den Header nicht hinzu, bevor die Anfrage an Ihren Absender weitergeleitet wird. Weitere Informationen finden Sie unter Konfigurieren Sie das Caching auf der Grundlage des Gerätetyps.	Ja

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischens peichern auf Basis von Header-We rten wird unterstützt
`CloudFront-Viewer-Country`	CloudFront fügt den Header nicht hinzu, bevor die Anfrage an Ihren Absender weitergeleitet wird.	Ja
`Connection`	CloudFront ersetzt diesen Header durch, `Connection: Keep-Alive` bevor die Anfrage an Ihren Absender weitergeleitet wird.	Nein
`Content-Length`	CloudFront leitet den Header an Ihren Ursprung weiter.	Nein
`Content-MD5`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`Content-Type`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`Cookie`	Wenn Sie CloudFront die Weiterleitung von Cookies konfigurieren, wird das `Cookie` Header-Fe ld an Ihren Ursprung weitergeleitet. Wenn Sie dies nicht tun, CloudFront wird das `Cookie` Header-Fe ld entfernt. Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.	Nein

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischens peichern auf Basis von Header-We rten wird unterstützt
`Date`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja, wird aber nicht empfohlen
`Expect`	CloudFront entfernt den Header.	Ja
`From`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`Host`	CloudFront setzt den Wert auf den Domainnam en des Ursprungs, der dem angeforderten Objekt zugeordnet ist. Sie können nicht auf der Grundlage des Host-Headers für Amazon S3 oder MediaStore Origins zwischenspeichern.	Ja (benutzer definiert) Nein (S3 und MediaStore)
`If-Match`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`If-Modified-Since`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`If-None-Match`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischens peichern auf Basis von Header-We rten wird unterstützt
`If-Range`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`If-Unmodified-Sinc e`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`Max-Forwards`	CloudFront leitet den Header an Ihren Ursprung weiter.	Nein
`Origin`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`Pragma`	CloudFront leitet den Header an Ihren Ursprung weiter.	Nein
`Proxy-Authenticate`	CloudFront entfernt den Header.	Nein
`Proxy-Authorizatio n`	CloudFront entfernt den Header.	Nein
`Proxy-Connection`	CloudFront entfernt den Header.	Nein

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischens peichern auf Basis von Header-We rten wird unterstützt
`Range`	CloudFront leitet den Header an Ihren Ursprung weiter. Weitere Informationen finden Sie unter Wie CloudFront werden Teilanfragen für ein Objekt (BereichGETs) verarbeitet.	Ja, standardm äßig
`Referer`	CloudFront entfernt den Header.	Ja
`Request-Range`	CloudFront leitet den Header an Ihren Ursprung weiter.	Nein
`TE`	CloudFront entfernt den Header.	Nein
`Trailer`	CloudFront entfernt den Header.	Nein
`Transfer-Encoding`	CloudFront leitet den Header an Ihren Ursprung weiter.	Nein
`Upgrade`	CloudFront entfernt den Header, sofern Sie keine WebSocket Verbindung hergestellt haben.	Nein (außer für WebSocket Verbindun gen)

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischens peichern auf Basis von Header-We rten wird unterstützt
`User-Agent`	CloudFront ersetzt den Wert dieses Header-Fe lds durch`Amazon CloudFront` . Wenn Sie Ihre Inhalte je CloudFront nach dem vom Benutzer verwendeten Gerät zwischenspeichern möchten, finden Sie weitere Informationen unterKonfigurieren Sie das Caching auf der Grundlage des Gerätetyp s.	Ja, wird aber nicht empfohlen
`Via`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`Warning`	CloudFront leitet den Header an Ihren Ursprung weiter.	Ja
`X-Amz-Cf-Id`	CloudFront fügt den Header zur Viewer-Anfrage hinzu, bevor die Anfrage an Ihren Ursprung weitergeleitet wird. Der Header-Wert enthält eine verschlüsselte Zeichenfolge, die die Anfrage eindeutig bezeichnet.	Nein
`X-Edge-*`	CloudFront entfernt alle `X-Edge-*` Header.	Nein
`X-Forwarded-For`	CloudFront leitet den Header an Ihren Ursprung weiter. Weitere Informationen finden Sie unter Client-IP-Adressen.	Ja

Header	Verhalten, wenn Sie nicht so konfigurieren, dass es auf der Grundlage von CloudFront Header-Werten zwischengespeichert wird	Das Zwischens peichern auf Basis von Header-We rten wird unterstützt
`X-Forwarded-Proto`	CloudFront entfernt den Header.	Nein
`X-HTTP-Method-Over ride`	CloudFront entfernt den Header.	Ja
`X-Real-IP`	CloudFront entfernt den Header.	Nein

HTTP-Version

CloudFront leitet Anfragen über HTTP/1.1 an Ihren benutzerdefinierten Ursprung weiter.

Maximale Länge einer Anfrage und maximale Länge einer URL

Die maximale Länge einer Anfrage – einschließlich des Pfads, der Abfragezeichenfolge (falls vorhanden) und der Header – beträgt 20 480 Byte.

CloudFront konstruiert aus der Anfrage eine URL. Die maximale Länge dieser URL beträgt 8 192 Byte.

Wenn eine Anfrage oder eine URL diese Höchstwerte überschreitet, CloudFront gibt es den HTTP-Statuscode 413, Request Entity Too Large, an den Viewer zurück und beendet dann die TCP-Verbindung zum Viewer.

OCSP-Stapling

Wenn ein Betrachter eine HTTPS-Anfrage für ein Objekt sendet, muss einer CloudFront oder der Betrachter bei der Zertifizierungsstelle (CA) bestätigen, dass das SSL-Zertifikat für die Domain nicht gesperrt wurde. OCSP-Hefting beschleunigt die Zertifikatsvalidierung CloudFront , da das Zertifikat

validiert und die Antwort von der CA zwischengespeichert werden kann, sodass der Client das Zertifikat nicht direkt bei der CA validieren muss.

Die Leistungssteigerung durch OCSP-Stapling ist deutlicher spürbar, wenn CloudFront viele HTTPS-Anfragen für Objekte in derselben Domain erhält. Jeder Server an einem CloudFront Edge-Standort muss eine separate Überprüfungsanforderung einreichen. Wenn CloudFront viele HTTPS-Anfragen für dieselbe Domain erhält, hat jeder Server an dem Edge-Standort nach kurzer Zeit hat eine Antwort von der CA vorliegen, die er an ein Paket im SSL-Handshake „heften" (engl. „to staple") kann; wenn der Betrachter von der Gültigkeit des Zertifikats überzeugt ist, kann CloudFront das angeforderte Objekt übertragen. Wenn Ihre Distribution an einem CloudFront Edge-Standort nicht viel Verkehr erhält, ist es wahrscheinlicher, dass neue Anfragen an einen Server weitergeleitet werden, der das Zertifikat noch nicht bei der CA validiert hat. In diesem Fall führt der Viewer den Validierungsschritt separat durch und der CloudFront Server stellt das Objekt bereit. Dieser CloudFront Server sendet auch eine Überprüfungsanfrage an die CA. Wenn er also das nächste Mal eine Anfrage erhält, die denselben Domainnamen enthält, erhält er eine Validierungsantwort von der CA.

Persistente Verbindungen

Wenn CloudFront Sie eine Antwort von Ihrem Absender erhalten, versucht er, die Verbindung mehrere Sekunden lang aufrechtzuerhalten, falls in diesem Zeitraum eine weitere Anfrage eingeht. Durch eine persistente Verbindung wird die Zeit gespart, die erforderlich ist, um die TCP-Verbindung erneut herzustellen und einen weiteren TLS-Handshake für nachfolgende Anforderungen durchzuführen.

Weitere Informationen, einschließlich solcher zur Konfiguration der Dauer ständiger Verbindungen, finden Sie unter Keep-Alive-Timeout (nur benutzerdefinierte und VPC-Ursprünge) im Abschnitt Referenz für alle Verteilungseinstellungen.

Protokolle

CloudFront leitet HTTP- oder HTTPS-Anfragen auf folgender Grundlage an den Ursprungsserver weiter:

- Das Protokoll der Anfrage, an die der Betrachter sendet CloudFront, entweder HTTP oder HTTPS.
- Der Wert des Felds Origin Protocol Policy in der CloudFront Konsole oder, wenn Sie die CloudFront API verwenden, das `OriginProtocolPolicy` Element im `DistributionConfig` komplexen Typ. In der CloudFront Konsole stehen die Optionen „Nur HTTP", „Nur HTTPS" und „Match Viewer" zur Verfügung.

Wenn Sie „Nur HTTP“ oder „Nur HTTPS“ angeben, CloudFront werden Anfragen mit dem angegebenen Protokoll an den Ursprungsserver weitergeleitet, unabhängig vom Protokoll in der Viewer-Anfrage.

Wenn Sie Match Viewer angeben, CloudFront leitet Anfragen mithilfe des Protokolls in der Viewer-Anforderung an den Ursprungsserver weiter. Beachten Sie, dass CloudFront das Objekt nur einmal zwischenspeichert, auch wenn Betrachter ihre Anfragen sowohl über HTTP als auch über HTTPS übertragen.

> **Important**
>
> Wenn CloudFront eine Anfrage mithilfe des HTTPS-Protokolls an den Ursprung weitergeleitet wird und der Ursprungsserver ein ungültiges Zertifikat oder ein selbstsigniertes Zertifikat zurückgibt, wird die TCP-Verbindung CloudFront unterbrochen.

Hinweise zum Aktualisieren einer Distribution mithilfe der CloudFront Konsole finden Sie unter. Eine Verteilung aktualisieren Informationen zum Aktualisieren einer Distribution mithilfe der CloudFront API finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.

Abfragezeichenfolgen

Sie können konfigurieren, ob CloudFront Abfragezeichenfolgenparameter an Ihren Ursprung weitergeleitet werden. Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern.

Timeout der Ursprungsverbindung und Verbindungsversuche zum Ursprung

Das Timeout für die Origin-Verbindung ist die Anzahl der Sekunden, die beim Versuch, eine Verbindung zum CloudFront Ursprung herzustellen, gewartet wird.

Die Anzahl der Versuche, eine Verbindung zum Ursprung herzustellen, gibt an, wie oft CloudFront versucht wird, eine Verbindung zum Ursprung herzustellen.

Zusammen bestimmen diese Einstellungen, wie lange CloudFront versucht wird, eine Verbindung zum Ursprung herzustellen, bevor ein Failover zum sekundären Ursprung erfolgt (im Fall einer Ursprungsgruppe) oder eine Fehlermeldung an den Viewer zurückgegeben wird. CloudFront Wartet standardmäßig bis zu 30 Sekunden (3 Versuche à 10 Sekunden), bevor versucht wird, eine Verbindung zum sekundären Ursprung herzustellen, oder eine Fehlermeldung zurückgegeben wird.

Sie können diese Zeit reduzieren, indem Sie ein kürzeres Verbindungs-Timeout, weniger Versuche oder beides angeben.

Weitere Informationen finden Sie unter Kontrolliere Timeouts und Versuche bei der Herkunft.

Ursprungs-Reaktions-Timeout

Das Ursprungs-Reaktions-Timeout, das auch als Ursprungs-Lese-Timeout oder Ursprungs-Anforderungs-Timeout bezeichnet wird, gilt für Folgendes:

- Die Zeitspanne in Sekunden, die auf eine Antwort CloudFront wartet, nachdem eine Anfrage an den Ursprung weitergeleitet wurde.
- Die Zeitspanne in Sekunden, die nach dem Empfang eines Antwortpakets vom Ursprung und vor dem Empfang des nächsten Pakets CloudFront gewartet wird.

CloudFront Das Verhalten hängt von der HTTP-Methode der Viewer-Anfrage ab:

- `GET`und `HEAD` Anfragen — Wenn der Ursprung nicht oder nicht innerhalb der Dauer des Antwort-Timeouts reagiert, wird die CloudFront Verbindung unterbrochen. Wenn die angegebene Anzahl der ursprünglichen Verbindungsversuche mehr als 1 beträgt, wird erneut CloudFront versucht, eine vollständige Antwort zu erhalten. CloudFront versucht bis zu 3 Mal, je nach dem Wert der Einstellung für ursprüngliche Verbindungsversuche. Wenn der Ursprung beim letzten Versuch keine Antwort sendet, unternimmt CloudFront erst dann einen weiteren Versuch, wenn die nächste Anfrage für Inhalte auf demselben Ursprung empfangen wird.
- `DELETE`,`OPTIONS`, `PATCH`PUT, und `POST` Anfragen — Wenn der Absender für die Dauer des Lese-Timeouts nicht reagiert, wird die Verbindung CloudFront unterbrochen und es wird nicht erneut versucht, den Ursprung zu kontaktieren. Der Client kann die Anfrage erneut senden, falls erforderlich.

Weitere Informationen, einschließlich Informationen zum Konfigurieren des Reaktions-Timeouts für den Ursprungs-Server, finden Sie unter Timeout bei der Antwort.

Gleichzeitige Anfragen für dasselbe Objekt (Zusammenfassung von Anfragen)

Wenn ein CloudFront Edge-Standort eine Anfrage für ein Objekt empfängt und sich das Objekt nicht im Cache befindet oder das zwischengespeicherte Objekt abgelaufen ist, wird die Anfrage CloudFront sofort an den Ursprung gesendet. Wenn es jedoch gleichzeitige Anfragen für dasselbe Objekt gibt,

d. h. wenn zusätzliche Anfragen für dasselbe Objekt (mit demselben Cache-Schlüssel) am Edge-Standort ankommen, bevor die Antwort auf die erste Anfrage CloudFront empfangen wird, wird eine CloudFront Pause eingelegt, bevor die zusätzlichen Anfragen an den Ursprung weitergeleitet werden. Diese kurze Pause trägt dazu bei, die Belastung des Ursprungs zu reduzieren. CloudFront sendet die Antwort der ursprünglichen Anfrage auf alle Anfragen, die während der Pause eingegangen sind. Dies wird als Request Collapsing (Zusammenfassung von Anfragen) bezeichnet. In den CloudFront Protokollen wird die erste Anfrage `Miss` im `x-edge-result-type` Feld als eine identifiziert, und die ausgeblendeten Anfragen werden als a gekennzeichnet. `Hit` Weitere Hinweise zu CloudFront Protokollen finden Sie unterthe section called “CloudFront und Edge-Funktionsprotokollierung”.

CloudFront reduziert nur Anfragen, die sich einen Cache-Schlüssel teilen. Wenn die zusätzlichen Anfragen nicht denselben Cache-Schlüssel verwenden, weil Sie beispielsweise so konfiguriert haben, dass der Cache CloudFront auf der Grundlage von Anforderungsheadern, Cookies oder Abfragezeichenfolgen gespeichert wird, werden alle Anfragen mit einem eindeutigen Cache-Schlüssel an Ihren Ursprung CloudFront weitergeleitet.

Wenn Sie verhindern möchten, dass alle Anfragen kollabiert werden, können Sie die verwaltete Cache-Richtlinie verwenden`CachingDisabled`, die auch das Zwischenspeichern verhindert. Weitere Informationen finden Sie unter Verwaltete Cache-Richtlinien verwenden.

Wenn Sie verhindern möchten, dass Anfragen für bestimmte Objekte reduziert werden, können Sie die Mindest-TTL für das Cache-Verhalten auf 0 setzen und den Ursprung so konfigurieren, dass er`Cache-Control: private`,,`Cache-Control: no-store`, `Cache-Control: no-cache` oder sendet. `Cache-Control: max-age=0` `Cache-Control: s-maxage=0` Diese Konfigurationen erhöhen die Belastung Ihres Ursprungs und führen zu zusätzlicher Latenz für gleichzeitige Anfragen, die angehalten werden, während auf die Antwort auf die CloudFront erste Anfrage gewartet wird.

> **Important**
>
> Unterstützt derzeit CloudFront nicht das Zusammenklappen von Anfragen, wenn Sie die Cookie-Weiterleitung in der Cache-Richtlinie, der ursprünglichen Anforderungsrichtlinie oder den Legacy-Cache-Einstellungen aktivieren.

`User-Agent`-Header

Wenn Sie je CloudFront nach dem Gerät, das ein Nutzer zum Ansehen Ihrer Inhalte verwendet, unterschiedliche Versionen Ihrer Objekte zwischenspeichern möchten, empfehlen wir Ihnen, eine

oder mehrere der folgenden Header so CloudFront zu konfigurieren, dass eine oder mehrere der folgenden Header an Ihren benutzerdefinierten Ursprung weitergeleitet werden:

- `CloudFront-Is-Desktop-Viewer`
- `CloudFront-Is-Mobile-Viewer`
- `CloudFront-Is-SmartTV-Viewer`
- `CloudFront-Is-Tablet-Viewer`

CloudFront Legt basierend auf dem Wert des `User-Agent` Headers den Wert dieser Header auf `true` oder `false` vor der Weiterleitung der Anfrage an Ihren Ursprung fest. Wenn ein Gerät in mehr als eine Kategorie fällt, können mehrere Werte sei `true`. Beispielsweise setzt CloudFront bei einigen Tablet-Geräten möglicherweise `CloudFront-Is-Mobile-Viewer` und `CloudFront-Is-Tablet-Viewer` auf `true`. Weitere Hinweise zur Konfiguration der CloudFront Zwischenspeicherung auf der Grundlage von Anforderungsheadern finden Sie unter. Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern

Sie können so konfigurieren CloudFront , dass Objekte auf der Grundlage von Werten im `User-Agent` Header zwischengespeichert werden, dies wird jedoch nicht empfohlen. Der `User-Agent` Header hat viele mögliche Werte, und das Zwischenspeichern auf der Grundlage dieser Werte würde CloudFront dazu führen, dass deutlich mehr Anfragen an Ihren Ursprung weitergeleitet werden.

Wenn Sie nicht so konfigurieren CloudFront , dass Objekte auf der Grundlage von Werten im `User-Agent` Header zwischengespeichert werden, CloudFront fügt Sie einen `User-Agent` Header mit dem folgenden Wert hinzu, bevor eine Anfrage an Ihren Ursprung weitergeleitet wird:

```
User-Agent = Amazon CloudFront
```

CloudFront fügt diesen Header hinzu, unabhängig davon, ob die Anfrage des Viewers einen `User-Agent` Header enthält. Wenn die Anfrage des Viewers einen `User-Agent` Header enthält, CloudFront wird dieser entfernt.

Wie CloudFront werden Antworten von Ihrem benutzerdefinierten Ursprung verarbeitet

Erfahren Sie, wie Antworten aus Ihrer benutzerdefinierten Quelle CloudFront verarbeitet werden.

Inhalt

- 100 Continue Antworten
- Caching
- Abgebrochene Anfragen
- Inhaltsvereinbarung
- Cookies
- Abgebrochene TCP-Verbindungen
- HTTP-Antwort-Header, die CloudFront entfernen oder ersetzen
- Maximale Dateigröße, die zwischengespeichert werden kann
- Ursprung nicht verfügbar
- Umleitungen
- Transfer-Encoding-Header

`100 Continue` Antworten

Ihr Absender kann nicht mehr als eine 100-Continue-Antwort an CloudFront senden. CloudFront Erwartet nach der ersten 100-Continue-Antwort eine HTTP-200-OK-Antwort. Wenn Ihr Absender nach der ersten eine weitere 100-Continue-Antwort sendet, CloudFront wird ein Fehler zurückgegeben.

Caching

- Stellen Sie sicher, dass der Ursprungsserver in den Header-Feldern `Date` und `Last-Modified` gültige und korrekte Werte einsetzt.
- CloudFront respektiert normalerweise einen `Cache-Control: no-cache` Header in der Antwort von der Quelle. Eine Ausnahme von dieser Regel wird unter Gleichzeitige Anfragen für dasselbe Objekt (Zusammenfassung von Anfragen) beschrieben.

Abgebrochene Anfragen

Wenn sich ein Objekt nicht im Edge-Cache befindet und ein Betrachter eine Sitzung beendet (z. B. einen Browser schließt), nachdem CloudFront er das Objekt von Ihrem Ursprung abgerufen hat, aber bevor es das angeforderte Objekt liefern kann, wird das Objekt CloudFront nicht an der Edge-Position zwischengespeichert.

Inhaltsvereinbarung

Wenn Ihr Ursprung `Vary:*` in der Antwort zurückkehrt und der Wert von Minimum TTL für das entsprechende Cache-Verhalten 0 ist, wird das Objekt CloudFront zwischengespeichert, aber dennoch wird jede nachfolgende Anfrage für das Objekt an den Ursprung weitergeleitet, um zu bestätigen, dass der Cache die neueste Version des Objekts enthält. CloudFront enthält keine bedingten Header wie oder. `If-None-Match` `If-Modified-Since` Infolgedessen gibt Ihr CloudFront Origin das Objekt als Antwort auf jede Anfrage zurück.

Wenn Ihr Ursprung `Vary:*` in der Antwort zurückgegeben wird und wenn der Wert von Minimum TTL für das entsprechende Cache-Verhalten ein anderer Wert ist, CloudFront verarbeitet der `Vary` Header wie unter beschriebenHTTP-Antwort-Header, die CloudFront entfernen oder ersetzen.

Cookies

Wenn Sie Cookies für ein Cache-Verhalten aktivieren und der Ursprung Cookies mit einem Objekt zurückgibt, werden sowohl das Objekt als auch die Cookies CloudFront zwischengespeichert. Beachten Sie, dass diese Vorgehensweise die Zwischenspeicherbarkeit für ein Objekt reduziert. Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.

Abgebrochene TCP-Verbindungen

Wenn die TCP-Verbindung zwischen CloudFront und Ihrem Ursprung unterbrochen wird, während Ihr Ursprung ein Objekt zurückgibt CloudFront, hängt CloudFront das Verhalten davon ab, ob Ihr Ursprung einen `Content-Length` Header in der Antwort enthalten hat:

- Content-Length-Header — CloudFront gibt das Objekt an den Betrachter zurück, sobald dieser das Objekt von Ihrem Ursprung bezieht. Wenn der Wert des `Content-Length` Headers jedoch nicht der Größe des Objekts entspricht, wird das Objekt CloudFront nicht zwischengespeichert.
- Transfer-Encoding: Chunked — CloudFront gibt das Objekt so an den Betrachter zurück, wie es das Objekt von Ihrem Ursprung abgerufen hat. Wenn die Antwort in Teilen jedoch nicht vollständig ist, CloudFront wird das Objekt nicht zwischengespeichert.
- Kein Content-Length-Header — CloudFront gibt das Objekt an den Viewer zurück und speichert es im Cache, aber das Objekt ist möglicherweise nicht vollständig. Ohne einen `Content-Length` Header kann CloudFront nicht bestimmen, ob die TCP-Verbindung versehentlich oder absichtlich verworfen wurde.

Wir empfehlen, dass Sie Ihren HTTP-Server so konfigurieren, dass er einen `Content-Length` Header hinzufügt, um zu verhindern, dass Teile CloudFront von Objekten zwischengespeichert werden.

HTTP-Antwort-Header, die CloudFront entfernen oder ersetzen

CloudFront entfernt oder aktualisiert die folgenden Header-Felder, bevor die Antwort von Ihrem Ursprung an den Viewer weitergeleitet wird:

- `Set-Cookie`— Wenn Sie CloudFront die Weiterleitung von Cookies konfigurieren, wird das `Set-Cookie` Header-Feld an Clients weitergeleitet. Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.
- `Trailer`
- `Transfer-Encoding`— Wenn Ihr Origin dieses Header-Feld zurückgibt, CloudFront setzt er den Wert auf, `chunked` bevor die Antwort an den Betrachter zurückgegeben wird.
- `Upgrade`
- `Vary` – Beachten Sie Folgendes:
  - Wenn Sie konfigurieren CloudFront , dass einer der gerätespezifischen Header an Ihren Ursprung (`CloudFront-Is-Desktop-Viewer`,,`CloudFront-Is-Tablet-Viewer`) weitergeleitet wird `CloudFront-Is-Mobile-ViewerCloudFront-Is-SmartTV-Viewer`, und wenn Sie Ihren Ursprung so konfigurieren, dass er CloudFront zurückkehrt CloudFront, kehrt `Vary:User-Agent` er `Vary:User-Agent` zum Viewer zurück. Weitere Informationen finden Sie unter Konfigurieren Sie das Caching auf der Grundlage des Gerätetyps.
  - Wenn Sie Ihren Ursprung so konfigurieren, dass er entweder `Accept-Encoding` oder `Cookie` in den `Vary` Header CloudFront einschließt, werden die Werte in die Antwort an den Viewer aufgenommen.
  - Wenn Sie so konfigurieren, CloudFront dass Header an Ihren Ursprung weitergeleitet werden, und wenn Sie Ihren Ursprung so konfigurieren, dass die Header-Namen CloudFront in der `Vary` Kopfzeile CloudFront zurückgegeben werden (z. B.`Vary:Accept-Charset,Accept-Language`), wird der `Vary` Header mit diesen Werten an den Viewer zurückgegeben.
  - Hinweise dazu, wie ein Wert von `*` in der `Vary` Kopfzeile CloudFront verarbeitet wird, finden Sie unterInhaltsvereinbarung.
  - Wenn Sie Ihren Ursprung so konfigurieren, dass er andere Werte in den `Vary` Header einbezieht, CloudFront werden die Werte entfernt, bevor die Antwort an den Viewer zurückgegeben wird.

- `Via`— CloudFront setzt den Wert in der Antwort an den Betrachter auf den folgenden Wert:

  ```
  Via: http-version alphanumeric-string.cloudfront.net (CloudFront)
  ```

  Der Wert ist beispielsweise in etwa wie folgt:

  ```
  Via: 1.1 1026589cc7887e7a0dc7827b4example.cloudfront.net (CloudFront)
  ```

Maximale Dateigröße, die zwischengespeichert werden kann

Die maximale Größe eines Antworttextes, der in seinem Cache CloudFront gespeichert wird, beträgt 50 GB. Dazu gehören auch Antworten für aufgeteilte Übertragungen, in denen kein Wert für die `Content-Length`-Kopfzeile angegeben wurde.

Sie können ein Objekt zwischenspeichern, das größer als diese Größe ist, indem Sie Bereichsanforderungen verwenden, um die Objekte in Teilen anzufordern, die jeweils 50 GB oder weniger groß sind. CloudFront CloudFrontspeichert diese Teile im Cache, da jeder von ihnen 50 GB oder weniger groß ist. Nachdem der Viewer alle Teile des Objekts abgerufen hat, kann er das ursprüngliche, größere Objekt rekonstruieren. Weitere Informationen finden Sie unter Verwenden von Bereichsanforderungen zum Zwischenspeichern großer Objekte.

Ursprung nicht verfügbar

Wenn Ihr Ursprungsserver nicht verfügbar ist und eine Anfrage für ein Objekt CloudFront erhält, das sich im Edge-Cache befindet, aber abgelaufen ist (z. B. weil der in der `Cache-Control max-age` Direktive angegebene Zeitraum abgelaufen ist), wird CloudFront entweder die abgelaufene Version des Objekts oder eine benutzerdefinierte Fehlerseite angezeigt. Weitere Informationen zum CloudFront Verhalten bei der Konfiguration benutzerdefinierter Fehlerseiten finden Sie unterWie CloudFront werden Fehler verarbeitet, wenn Sie benutzerdefinierte Fehlerseiten konfiguriert haben.

In einigen Fällen wird ein Objekt, das selten angefordert wird, entfernt und ist nicht mehr im Edge-Cache verfügbar. CloudFront kann ein Objekt, das entfernt wurde, nicht bereitstellen.

Umleitungen

Wenn Sie den Speicherort eines Objekts auf dem Ursprungsserver ändern, können Sie Ihren Webserver so konfigurieren, dass Anfragen an den neuen Speicherort umgeleitet werden. Nachdem Sie die Umleitung konfiguriert haben, sendet ein Betrachter, wenn er zum ersten Mal eine Anfrage für das Objekt CloudFront sendet, die Anfrage an den Ursprung, und der Absender antwortet mit einer

Weiterleitung (z. B.). `302 Moved Temporarily` CloudFront speichert die Weiterleitung im Cache und gibt sie an den Betrachter zurück. CloudFront folgt der Weiterleitung nicht.

Sie können Ihren Webserver so konfigurieren, dass Anfragen an einen der folgenden Speicherorte umgeleitet werden:

- Die neue URL des Objekts auf dem Ursprungsserver. Wenn der Zuschauer der Weiterleitung zur neuen URL folgt, umgeht er die URL CloudFront und geht direkt zum Ursprung. Daher empfehlen wir, Anfragen nicht an die neue URL des Objekts am Ursprung weiterzuleiten.
- Die neue CloudFront URL für das Objekt. Wenn der Betrachter die Anfrage sendet, die die neue CloudFront URL enthält, CloudFront ruft er das Objekt von der neuen Position auf Ihrem Ursprung ab, speichert es am Edge-Standort im Cache und gibt das Objekt an den Viewer zurück. Nachfolgende Anfragen für das Objekt werden von dem Edge-Standort bedient. Dadurch werden Latenzzeiten und Arbeitslasten vermieden, die bei Viewer-Anforderungen für das Objekt an den Ursprungsserver entstehen. Allerdings fallen bei jeder neuen Anfrage für das Objekt Gebühren für zwei Anfragen an CloudFront berechnet.

`Transfer-Encoding`-Header

CloudFront unterstützt nur den `chunked` Wert des Headers. `Transfer-Encoding` Wenn Ihr Ursprung CloudFront zurückkehrt`Transfer-Encoding: chunked`, sendet er das Objekt an den Client zurück, sobald das Objekt am Edge-Standort empfangen wurde, und speichert das Objekt im Chunk-Format für nachfolgende Anfragen im Cache.

Wenn der Betrachter eine `Range GET` Anfrage stellt und der Ursprung CloudFront zurückkehrt`Transfer-Encoding: chunked`, wird das gesamte Objekt anstelle des angeforderten Bereichs an den Viewer zurückgegeben.

Wir empfehlen, dass Sie die Abschnittscodierung verwenden, wenn die Länge des Inhalts Ihrer Antwort nicht im Voraus ermittelt werden kann. Weitere Informationen finden Sie unter Abgebrochene TCP-Verbindungen.

Verhalten von Anfragen und Antworten für Ursprungsgruppen

Anfragen an eine Ursprungsgruppe funktionieren genauso wie Anfragen an einen Ursprung, der nicht als Ursprungsgruppe eingerichtet ist, außer wenn ein Ursprungs-Failover vorliegt. Wie bei jeder anderen Quelle gilt auch hier: CloudFront Wenn eine Anfrage eingeht und der Inhalt bereits an einem Edge-Standort zwischengespeichert ist, wird der Inhalt den Zuschauern aus dem Cache

bereitgestellt. Wenn ein Cache-Fehler vorliegt und der Ursprung eine Ursprungsgruppe ist, werden Viewer-Anforderungen an den primären Ursprung in der Ursprungsgruppe weitergeleitet.

Das Anfrage- und Antwortverhalten für den primären Ursprung ist identisch mit dem für einen Ursprung, der nicht zu einer Ursprungsgruppe gehört. Weitere Informationen erhalten Sie unter Verhalten von Anfragen und Antworten für Amazon-S3-Ursprünge und Verhalten von Anfragen und Antworten für benutzerdefinierte Ursprungsserver.

Nachfolgend werden die Verhaltensweisen bei Origin Failover beschrieben, wenn der primäre Ursprung bestimmte HTTP-Statuscodes ausgibt:

- HTTP 2xx-Statuscode (erfolgreich): Die Datei wird CloudFront zwischengespeichert und an den Betrachter zurückgegeben.
- HTTP 3xx-Statuscode (Umleitung): CloudFront Gibt den Statuscode an den Betrachter zurück.
- HTTP 4xx- oder 5xx-Statuscode (Client-/Server-Fehler): Wenn der zurückgegebene Statuscode für Failover konfiguriert wurde, wird dieselbe Anfrage an den sekundären Ursprung in der Ursprungsgruppe CloudFront gesendet.
- HTTP 4xx- oder 5xx-Statuscode (Client-/Server-Fehler): Wenn der zurückgegebene Statuscode nicht für Failover konfiguriert wurde, wird der Fehler an den Viewer zurückgegeben. CloudFront

CloudFront führt nur dann einen Failover zum sekundären Ursprung durch, wenn die HTTP-Methode der Viewer-Anfrage,, oder ist. `GET` `HEAD` `OPTIONS` CloudFront führt nicht zu einem Failover, wenn der Betrachter eine andere HTTP-Methode sendet (z. B. `POSTPUT`, usw.).

Wenn eine Anfrage CloudFront an einen sekundären Ursprung gesendet wird, ist das Antwortverhalten dasselbe wie bei einem CloudFront Ursprung, der sich nicht in einer Ursprungsgruppe befindet.

Weitere Informationen zu Ursprungsgruppen finden Sie unter Optimieren Sie die Hochverfügbarkeit mit CloudFront Origin Failover.

Fügen Sie benutzerdefinierte Header zu ursprünglichen Anfragen hinzu

Sie können so konfigurieren CloudFront , dass den Anfragen, die an Ihren Ursprung gesendet werden, benutzerdefinierte Header hinzugefügt werden. Sie können benutzerdefinierte Header verwenden, um Informationen von Ihrem Absender zu senden und zu sammeln, die Sie bei typischen

Zuschaueranfragen nicht erhalten. Sie können die Header sogar für jeden Ursprung anpassen. CloudFrontunterstützt benutzerdefinierte Header für benutzerdefinierte Ursprünge und Amazon S3 S3-Ursprünge.

Inhalt

Anwendungsfälle

Sie können benutzerdefinierte Header verwenden, z. B. in den folgenden Beispielen:

Identifizieren von Anfragen von CloudFront

Sie können die Anfragen identifizieren, von denen Ihr Absender empfängt CloudFront. Das kann nützlich sein, wenn du wissen möchtest, ob Nutzer sie umgehen CloudFront, oder wenn du mehr als ein CDN verwendest und Informationen darüber haben möchtest, welche Anfragen von jedem CDN kommen.

> **Note**
>
> Wenn Sie einen Amazon S3-Ursprung verwenden und die Protokollierung des Zugriffs auf Amazon S3 aktivieren, enthalten die Protokolle keine Header-Informationen.

Bestimmen, welche Anforderungen von einer bestimmten Verteilung stammen

Wenn Sie mehrere CloudFront Distributionen so konfigurieren, dass sie denselben Ursprung verwenden, können Sie jeder Verteilung unterschiedliche benutzerdefinierte Header hinzufügen. Anschließend können Sie mit den Protokollen Ihres Ursprungs bestimmen, welche Anforderungen aus welcher CloudFront-Verteilung stammen.

Cross-Origin Resource Sharing (CORS) aktivieren

Falls einige Ihrer Zuschauer Cross-Origin Resource Sharing (CORS) nicht unterstützen, können Sie so konfigurieren, dass der `Origin` Header immer CloudFront zu Anfragen hinzugefügt wird, die an Ihren Absender gesendet werden. Dann können Sie Ihren Ursprung so konfigurieren, dass der `Access-Control-Allow-Origin`-Header für jede Anforderung zurückgegeben wird. Sie müssen auch CloudFront so konfigurieren, dass die CORS-Einstellungen eingehalten werden.

Steuern des Zugriffs auf Inhalt

Mit benutzerdefinierten Header können Sie den Zugriff auf Inhalte steuern. Indem du deinen Absender so konfigurierst, dass er nur dann auf Anfragen reagiert, wenn sie einen benutzerdefinierten Header enthalten, der von hinzugefügt wird CloudFront, verhinderst du, dass Nutzer deine Inhalte umgehen CloudFront und direkt am Ursprung darauf zugreifen. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff auf Dateien mit benutzerdefinierten Ursprüngen.

Konfiguriere CloudFront es so, dass benutzerdefinierte Header zu ursprünglichen Anfragen hinzugefügt werden

Um eine Verteilung so zu konfigurieren, dass benutzerdefinierte Header Anforderungen hinzugefügt werden, die er an Ihren Ursprung sendet, aktualisieren Sie die Ursprungskonfiguration mit einer der folgenden Methoden:

- CloudFront Konsole — Wenn Sie eine Verteilung erstellen oder aktualisieren, geben Sie Header-Namen und -Werte in den Einstellungen Benutzerdefinierte Header hinzufügen an. Weitere Informationen finden Sie unter Benutzerdefinierten Header hinzufügen.
- CloudFront API — Geben Sie für jeden Ursprung, dem Sie benutzerdefinierte Header hinzufügen möchten, die Header-Namen und Werte im `CustomHeaders` Feld darin an. `Origin` Weitere Informationen finden Sie unter CreateDistributionoder UpdateDistributionin der Amazon CloudFront API-Referenz.

Wenn die von Ihnen angegebenen Header-Namen und Werte nicht bereits in der Viewer-Anfrage vorhanden sind, CloudFront fügt sie der ursprünglichen Anfrage hinzu. Wenn ein Header vorhanden ist, wird der Header-Wert CloudFront überschrieben, bevor die Anfrage an den Ursprung weitergeleitet wird.

Informationen zu den Kontingenten, die für benutzerdefinierte Origin-Header gelten, finden Sie unter. Kontingente für Header

Benutzerdefinierte Header, die nicht zu CloudFront ursprünglichen Anfragen hinzugefügt werden können

Sie können nicht so konfigurieren CloudFront , dass Anfragen, die an Ihren Absender gesendet werden, die folgenden Header hinzugefügt werden:

- `Cache-Control`
- `Connection`
- `Content-Length`
- `Cookie`
- `Host`
- `If-Match`
- `If-Modified-Since`
- `If-None-Match`
- `If-Range`
- `If-Unmodified-Since`
- `Max-Forwards`
- `Pragma`
- `Proxy-Authenticate`
- `Proxy-Authorization`
- `Proxy-Connection`
- `Range`
- `Request-Range`
- `TE`
- `Trailer`
- `Transfer-Encoding`
- `Upgrade`
- `Via`

- Header, die mit `X-Amz-` beginnen
- Header, die mit `X-Edge-` beginnen
- `X-Real-Ip`

So konfigurieren CloudFront , dass der Header weitergeleitet wird **`Authorization`**

Wenn Sie CloudFront eine Viewer-Anfrage an Ihren Ursprung weiterleiten, CloudFront werden standardmäßig einige Viewer-Header entfernt, einschließlich des `Authorization` Headers. Um sicherzustellen, dass Ihr Ursprung immer den `Authorization`-Header in Ursprungsanforderungen erhält, haben Sie folgende Möglichkeiten:

- Fügen Sie den `Authorization`-Header mithilfe einer Cache-Richtlinie zum Cache-Schlüssel hinzu. Alle Header im Cache-Schlüssel werden automatisch in Ursprungsanforderungen eingeschlossen. Weitere Informationen finden Sie unter Steuern Sie den Cache-Schlüssel mit einer Richtlinie.
- Verwenden Sie eine Herkunftsanforderungsrichtlinie, die alle Betrachter-Header an den Ursprung weiterleitet. Sie können den `Authorization` Header in einer ursprünglichen Anforderungsrichtlinie nicht einzeln weiterleiten, aber wenn Sie alle Viewer-Header weiterleiten, wird der `Authorization` Header in Viewer-Anfragen CloudFront miteinbezogen. CloudFront stellt für diesen Anwendungsfall eine verwaltete Ursprungsrichtlinie für Anfragen bereit, die als Managed- AllViewer bezeichnet wird. Weitere Informationen finden Sie unter Richtlinien für verwaltete Origin-Anfragen verwenden.

Wie CloudFront werden Teilanfragen für ein Objekt (BereichGETs) verarbeitet

Bei großen Objekten kann der Viewer (Webbrowser oder anderer Client) mehrere `GET`-Anforderungen stellen und verwendet den `Range`-Anforderungs-Header, um das Objekt in kleineren Teilen herunterzuladen. Diese Anfragen für Bereiche von Bytes, manchmal auch als `Range GET`-Anfragen bezeichnet, steigern die Effizienz von Teil-Downloads und die Wiederherstellung nach teilweise fehlgeschlagenen Übertragungen.

Wenn es eine `Range GET` Anfrage CloudFront empfängt, überprüft es den Cache an dem Edge-Standort, der die Anfrage empfangen hat. Wenn der Cache an dieser Edge-Position bereits das

gesamte Objekt oder den angeforderten Teil des Objekts enthält, wird CloudFront sofort der angeforderte Bereich aus dem Cache bereitgestellt.

Wenn der Cache den angeforderten Bereich nicht enthält, CloudFront wird die Anfrage an den Ursprung weitergeleitet. (Um die Leistung zu optimieren, wird CloudFront möglicherweise ein größerer Bereich angefordert, als der Client im angefordert hat`Range GET`.) Der nächste Schritt hängt davon ab, ob der Ursprung `Range GET`-Anfragen unterstützt:

- Wenn der Ursprung **`Range GET`** Anfragen unterstützt, gibt er den angeforderten Bereich zurück. CloudFront bedient den angeforderten Bereich und speichert ihn auch für future Anfragen. (Amazon S3 unterstützt `Range GET`-Anforderungen, ebenso wie viele HTTP-Server.)
- Wenn der Ursprung keine **`Range GET`** Anfragen unterstützt, wird das gesamte Objekt zurückgegeben. CloudFront bedient die aktuelle Anfrage, indem es das gesamte Objekt sendet und es gleichzeitig für future Anfragen zwischenspeichert. Nachdem CloudFront das gesamte Objekt in einem Edge-Cache zwischengespeichert wurde, reagiert es auf neue `Range GET` Anfragen, indem es den angeforderten Bereich bereitstellt.

In beiden Fällen CloudFront beginnt die Bereitstellung des angeforderten Bereichs oder Objekts für den Endbenutzer, sobald das erste Byte vom Ursprung eintrifft.

> **Note**
>
> Wenn der Betrachter eine `Range GET` Anfrage stellt und der Ursprung CloudFront zurückkehrt`Transfer-Encoding: chunked`, wird das gesamte Objekt anstelle des angeforderten Bereichs an den Betrachter zurückgegeben.

CloudFront folgt im Allgemeinen der RFC-Spezifikation für den `Range` Header. Wenn Ihre `Range`-Header die folgenden Anforderungen jedoch nicht erfüllen, gibt CloudFront HTTP-Statuscode `200` mit dem vollständigen Objekt anstelle von Statuscode `206` mit den angefragten Bereichen zurück:

- Die Bereiche müssen in aufsteigender Reihenfolge aufgeführt sein. Beispielweise ist `100-200,300-400` gültig, `300-400,100-200` hingegen nicht.
- Die Bereiche dürfen sich nicht überschneiden. Beispielsweise ist `100-200,150-250` nicht gültig.
- Alle Bereichsspezifikationen müssen gültig sein. Beispielsweise können Sie keinen negativen Wert als Teil eines Bereichs angeben.

Weitere Informationen zum `Range`-Anforderungs-Header finden Sie im Abschnitt zu Bereichsanforderungen in RFC 7233 oder im Abschnitt zum Bereich in den MDN-Webdokumenten.

Verwenden von Bereichsanforderungen zum Zwischenspeichern großer Objekte

Wenn das Caching aktiviert ist, wird ein Objekt, das größer als 50 GB ist, CloudFront nicht abgerufen oder zwischengespeichert. Wenn ein Ursprung angibt, dass das Objekt größer als diese Größe ist (im `Content-Length` Antwort-Header), CloudFront wird die Verbindung zum Ursprung geschlossen und es wird ein Fehler an den Viewer zurückgegeben. (CloudFront Kann bei deaktiviertem Caching ein Objekt, das größer als diese Größe ist, vom Ursprung abrufen und an den Betrachter weitergeben. Zwischenspeichert das Objekt jedoch CloudFront nicht.)

Bei Bereichsanforderungen können CloudFront Sie jedoch ein Objekt zwischenspeichern, das größer als die maximale zwischenspeicherbare Dateigröße ist.

Example Beispiel

1. Stellen Sie sich einen Ursprung mit einem 100-GB-Objekt vor. Ruft bei aktiviertem Caching CloudFront kein Objekt dieser Größe ab oder speichert es nicht im Cache. Der Viewer kann jedoch mehrere Bereichsanforderungen senden, um dieses Objekt in Teilen abzurufen, wobei die Teile jeweils kleiner als 50 GB sind.
2. Der Betrachter kann das Objekt in Teilen von 20 GB anfordern, indem er eine Anfrage mit dem Header `Range: bytes=0-21474836480` zum Abrufen des ersten Teils, eine weitere Anfrage mit dem Header `Range: bytes=21474836481-42949672960` zum Abrufen des nächsten Teils usw. sendet.
3. Nachdem der Viewer alle Teile empfangen hat, kann er sie kombinieren, um das ursprüngliche Objekt mit 100 GB zu erstellen.
4. In diesem Fall CloudFront speichert er jeden der 20 GB-Teile des Objekts im Cache und kann auf nachfolgende Anfragen für denselben Teil aus dem Cache antworten.

Bei einer Bereichsanforderung für ein komprimiertes Objekt basiert die Bytebereichsanforderung auf der komprimierten Größe und nicht auf der Originalgröße des Objekts. Weitere Hinweise zum Komprimieren von Dateien finden Sie unterKomprimierte Dateien bereitstellen.

Wie CloudFront werden HTTP 3xx-Statuscodes von Ihrem Ursprung verarbeitet

Wenn Sie CloudFront ein Objekt von Ihrem Amazon S3 S3-Bucket oder Ihrem benutzerdefinierten Ursprungsserver anfordern, gibt Ihr Origin manchmal einen HTTP 3xx-Statuscode zurück. Dies weist in der Regel auf einen der folgenden Umstände hin:

- Die URL des Objekts hat sich geändert (z. B. Statuscodes 301, 302, 307 oder 308).
- Das Objekt hat sich seit der letzten CloudFront Anfrage nicht geändert (Statuscode 304)

CloudFront speichert 3xx-Antworten entsprechend den Einstellungen in Ihrer CloudFront Distribution und den Headern in der Antwort im Cache. CloudFront speichert die Antworten 307 und 308 nur dann im Cache, wenn Sie den `Cache-Control` Header in die Antworten von der Quelle aufnehmen. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

Wenn dein Absender einen Umleitungsstatuscode zurückgibt (z. B. 301 oder 307), folgt er der Weiterleitung CloudFront nicht. CloudFront leitet die 301- oder 307-Antwort an den Zuschauer weiter, der der Weiterleitung folgen kann, indem er eine neue Anfrage sendet.

Wie CloudFront werden die HTTP-Statuscodes 4xx und 5xx von Ihrem Ursprung verarbeitet

Wenn CloudFront Sie ein Objekt von Ihrem Amazon S3 S3-Bucket oder Ihrem benutzerdefinierten Ursprungsserver anfordern, gibt Ihr Origin manchmal einen HTTP-Statuscode 4xx oder 5xx zurück, der darauf hinweist, dass ein Fehler aufgetreten ist. CloudFront Das Verhalten hängt ab von:

- Ob Sie benutzerdefinierte Fehlerseiten konfiguriert haben
- Ob Sie konfiguriert haben, wie lange Sie Fehlerantworten von Ihrem Ursprung zwischenspeichern CloudFront möchten (Mindest-TTL für Fehler-Caching)
- Der Statuscode
- Bei 5xx-Statuscodes, ob sich das angeforderte Objekt derzeit im CloudFront Edge-Cache befindet
- Gibt bei einigen 4xx-Statuscodes an, ob der Ursprung einen `Cache-Control max-age` Header oder zurückgibt `Cache-Control s-maxage`

CloudFront speichert immer Antworten auf `GET` und `HEAD` Anfragen im Cache. Sie können auch so konfigurieren CloudFront , dass Antworten auf `OPTIONS` Anfragen zwischengespeichert werden. CloudFront speichert keine Antworten auf Anfragen, die die anderen Methoden verwenden.

Wenn der Ursprung nicht antwortet, wird bei der CloudFront Anfrage an den Ursprung ein Timeout erreicht, was als HTTP 5xx-Fehler des Ursprungs angesehen wird, obwohl der Ursprung nicht mit diesem Fehler geantwortet hat. In diesem Szenario werden CloudFront weiterhin zwischengespeicherte Inhalte bereitgestellt. Weitere Informationen finden Sie unter Ursprung nicht verfügbar.

Wenn Sie die Protokollierung aktiviert haben, werden die Ergebnisse unabhängig vom HTTP-Statuscode in die Protokolle CloudFront geschrieben.

Weitere Informationen zu Funktionen und Optionen, die sich auf die Fehlermeldung beziehen CloudFront, von der zurückgegeben wurde, finden Sie im Folgenden:

- Informationen zu Einstellungen für benutzerdefinierte Fehlerseiten in der CloudFront Konsole finden Sie unterBenutzerdefinierte Fehlerseiten und Zwischenspeicherung von Fehlern.
- Hinweise zum Fehler beim Zwischenspeichern der Mindest-TTL in der CloudFront Konsole finden Sie unter. Mindest-TTL für die Zwischenspeicherung von Fehlern (Sekunden)
- Eine Liste der HTTP-Statuscodes, die CloudFront zwischengespeichert werden, finden Sie unter. HTTP-Statuscodes 4xx und 5xx, die zwischengespeichert werden CloudFront

Themen

Wie CloudFront werden Fehler verarbeitet, wenn Sie benutzerdefinierte Fehlerseiten konfiguriert haben

Wenn Sie benutzerdefinierte Fehlerseiten konfiguriert haben, hängt CloudFront das Verhalten davon ab, ob sich das angeforderte Objekt im Edge-Cache befindet.

Das angeforderte Objekt ist nicht im Edge-Cache vorhanden

CloudFront versucht weiterhin, das angeforderte Objekt von Ihrem Ursprung abzurufen, wenn alle der folgenden Bedingungen zutreffen:

- Ein Viewer fordert ein Objekt an.
- Das Objekt ist nicht im Edge-Cache vorhanden.
- Ihr Ursprungsserver gibt einen HTTP-Statuscode 4xx oder 5xx zurück und eines der Folgenden ist wahr:
  - Ihr Ursprungsserver gibt einen HTTP-Statuscode 5xx anstelle eines Statuscodes 304 (nicht geändert) oder eine aktualisierte Version des Objekts zurück.
  - Ihr Ursprungs-Server gibt einen HTTP-Statuscode 4xx zurück, der nicht durch einen Cache-Control-Header eingeschränkt und in der folgenden Statuscodeliste enthalten is: HTTP-Statuscodes 4xx und 5xx, die zwischengespeichert werden CloudFront .
  - Ihr Ursprungs-Server gibt einen HTTP-Statuscode 4xx ohne `Cache-Control max-age`- oder `Cache-Control s-maxage`-Header zurück und der Statuscode ist in der folgenden Statuscodeliste enthalten: HTTP 4xx-Statuscodes, die basierend auf CloudFront Headern zwischengespeichert werden `Cache-Control`.

CloudFront macht Folgendes:

1. CloudFront Überprüft im CloudFront Edge-Cache, der die Viewer-Anfrage empfangen hat, Ihre Distributionskonfiguration und ruft den Pfad der benutzerdefinierten Fehlerseite ab, die dem Statuscode entspricht, den Ihr Origin zurückgegeben hat.
2. CloudFront findet das erste Cache-Verhalten in Ihrer Distribution, dessen Pfadmuster dem Pfad der benutzerdefinierten Fehlerseite entspricht.
3. Der CloudFront Edge-Standort sendet eine Anforderung für die benutzerdefinierte Fehlerseite an den Ursprung, der im Cache-Verhalten angegeben ist.
4. Der Ursprungsserver gibt die benutzerdefinierte Fehlerseite an den Edge-Standort zurück.
5. CloudFront gibt die benutzerdefinierte Fehlerseite an den Viewer zurück, der die Anfrage gestellt hat, und speichert die benutzerdefinierte Fehlerseite für maximal die folgenden Werte im Cache:
   - Der Zeitraum, der durch die Mindest-TTL für die Zwischenspeicherung von Fehlern angegeben ist (standardmäßig zehn Sekunden)

- Der Zeitraum, der durch einen `Cache-Control max-age`- oder einen `Cache-Control s-maxage`-Header angegeben ist, der vom Ursprungsserver zurückgegeben wird, wenn die erste Anfrage den Fehler generiert hat

6. Nach Ablauf der (in Schritt 5 festgelegten) Cache-Zeit wird erneut CloudFront versucht, das angeforderte Objekt abzurufen, indem eine weitere Anfrage an Ihren Ursprung weitergeleitet wird. CloudFront wiederholt den Vorgang in Intervallen, die durch die Mindest-TTL für das Zwischenspeichern des Fehlers festgelegt sind.

Das angeforderte Objekt ist im Edge-Cache vorhanden

CloudFront bedient weiterhin das Objekt, das sich derzeit im Edge-Cache befindet, wenn alle der folgenden Bedingungen zutreffen:

- Ein Viewer fordert ein Objekt an.
- Das Objekt ist im Edge-Cache vorhanden, aber es ist abgelaufen.
- Ihr Ursprungsserver gibt einen HTTP-Statuscode 5xx anstelle eines Statuscodes 304 (nicht geändert) oder eine aktualisierte Version des Objekts zurück.

CloudFront macht Folgendes:

1. Wenn Ihr Origin einen 5xx-Statuscode zurückgibt, CloudFront wird das Objekt bereitgestellt, obwohl es abgelaufen ist. Reagiert für die Dauer des Fehler-Cachings (Mindest-TTL) CloudFront weiterhin auf Viewer-Anfragen, indem das Objekt aus dem Edge-Cache bereitgestellt wird.

   Wenn Ihr Ursprungsserver einen 4xx-Statuscode zurückgibt, sendet CloudFront den Statuscode anstelle des angeforderten Objekts an den Betrachter.

2. Wenn die Mindest-TTL für den Fehler beim Zwischenspeichern abgelaufen ist, wird erneut CloudFront versucht, das angeforderte Objekt abzurufen, indem eine weitere Anfrage an Ihren Ursprung weitergeleitet wird. Beachten Sie, dass das Objekt, wenn es nicht häufig angefordert wird, CloudFront möglicherweise aus dem Edge-Cache entfernt wird, während Ihr Ursprungsserver immer noch 5xx-Antworten zurückgibt. Informationen darüber, wie lange Objekte in CloudFront Edge-Caches verbleiben, finden Sie unter. Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf)

Wie CloudFront werden Fehler verarbeitet, wenn Sie keine benutzerdefinierten Fehlerseiten konfiguriert haben

Wenn Sie keine benutzerdefinierten Fehlerseiten konfiguriert haben, hängt CloudFront das Verhalten davon ab, ob sich das angeforderte Objekt im Edge-Cache befindet.

Themen

Das angeforderte Objekt ist nicht im Edge-Cache vorhanden

CloudFront versucht weiterhin, das angeforderte Objekt von Ihrem Ursprung abzurufen, wenn alle der folgenden Bedingungen zutreffen:

- Ein Viewer fordert ein Objekt an.
- Das Objekt ist nicht im Edge-Cache vorhanden.
- Ihr Ursprungsserver gibt einen HTTP-Statuscode 4xx oder 5xx zurück und eines der Folgenden ist wahr:
  - Ihr Ursprungsserver gibt einen HTTP-Statuscode 5xx anstelle eines Statuscodes 304 (nicht geändert) oder eine aktualisierte Version des Objekts zurück.
  - Ihr Ursprungs-Server gibt einen HTTP-Statuscode 4xx zurück, der nicht durch einen Cache-Control-Header eingeschränkt und in der folgenden Statuscodeliste enthalten is: HTTP-Statuscodes 4xx und 5xx, die zwischengespeichert werden CloudFront
  - Ihr Ursprungs-Server gibt einen HTTP-Statuscode 4xx ohne `Cache-Control max-age`- oder `Cache-Control s-maxage`-Header zurück und der Statuscode ist in der folgenden Statuscodeliste enthalten: HTTP 4xx-Statuscodes, die basierend auf CloudFront Headern zwischengespeichert werden `Cache-Control`.

CloudFront macht Folgendes:

1. CloudFront gibt den 4xx- oder 5xx-Statuscode an den Viewer zurück und speichert außerdem den Statuscode im Edge-Cache, der die Anforderung für maximal die folgenden Werte erhalten hat:
   - Der Zeitraum, der durch die Mindest-TTL für die Zwischenspeicherung von Fehlern angegeben ist (standardmäßig zehn Sekunden)

- Der Zeitraum, der durch einen `Cache-Control max-age`- oder einen `Cache-Control s-maxage`-Header angegeben ist, der vom Ursprungsserver zurückgegeben wird, wenn die erste Anfrage den Fehler generiert hat

2. Für die Dauer der Zwischenspeicherung (in Schritt 1 bestimmt) reagiert CloudFront auf nachfolgende Viewer-Anfragen für dasselbe Objekte mit den zwischengespeicherten Statuscodes 4xx und 5xx.
3. Nach Ablauf der (in Schritt 1 festgelegten) Caching-Zeit wird erneut CloudFront versucht, das angeforderte Objekt abzurufen, indem eine weitere Anfrage an Ihren Ursprung weitergeleitet wird. CloudFront wiederholt den Vorgang in Intervallen, die durch die Mindest-TTL für das Zwischenspeichern des Fehlers festgelegt sind.

Das angeforderte Objekt ist im Edge-Cache vorhanden

CloudFront bedient weiterhin das Objekt, das sich derzeit im Edge-Cache befindet, wenn alle der folgenden Bedingungen zutreffen:

- Ein Viewer fordert ein Objekt an.
- Das Objekt ist im Edge-Cache vorhanden, aber es ist abgelaufen. Das bedeutet, dass das Objekt veraltet ist.
- Ihr Ursprungsserver gibt einen HTTP-Statuscode 5xx anstelle eines Statuscodes 304 (nicht geändert) oder eine aktualisierte Version des Objekts zurück.

CloudFront macht Folgendes:

1. Wenn Ihr Origin einen 5xx-Fehlercode zurückgibt, CloudFront wird das Objekt versendet, obwohl es abgelaufen ist. Reagiert für die Dauer des Fehler-Cachings (mindestens TTL) (standardmäßig 10 Sekunden) CloudFront weiterhin auf Viewer-Anfragen, indem das Objekt aus dem Edge-Cache bereitgestellt wird.

   Wenn Ihr Ursprungsserver einen 4xx-Statuscode zurückgibt, sendet CloudFront den Statuscode anstelle des angeforderten Objekts an den Betrachter.
2. Nachdem die Mindest-TTL beim Zwischenspeichern des Fehlers abgelaufen ist, wird erneut CloudFront versucht, das angeforderte Objekt abzurufen, indem eine weitere Anfrage an Ihren Ursprung weitergeleitet wird. Wenn das Objekt nicht häufig angefordert wird, wird es CloudFront möglicherweise aus dem Edge-Cache entfernt, während Ihr Ursprungsserver immer noch 5xx-

Antworten zurückgibt. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

> **Tip**
>
> - Wenn Sie die `Stale-While-Revalidate` Direktive `stale-if-error` or konfigurieren, können Sie angeben, wie lange die veralteten Objekte im Edge-Cache verfügbar sind. Auf diese Weise können Sie Ihren Zuschauern weiterhin Inhalte bereitstellen, auch wenn Ihre Herkunft nicht verfügbar ist. Weitere Informationen finden Sie unter Stellt veraltete (abgelaufene) Inhalte bereit.
> - CloudFront stellt nur ein Objekt bereit, das bis zum angegebenen maximalen TTL-Wert veraltet ist. Nach Ablauf dieser Dauer ist das Objekt nicht mehr im Edge-Cache verfügbar.

HTTP-Statuscodes 4xx und 5xx, die zwischengespeichert werden CloudFront

CloudFront speichert die von Ihrem Ursprung zurückgegebenen HTTP 4xx- und 5xx-Statuscodes im Cache, abhängig vom jeweiligen Statuscode, der zurückgegeben wird, und davon, ob Ihr Ursprung in der Antwort bestimmte Header zurückgibt.

CloudFront speichert die folgenden HTTP-Statuscodes 4xx und 5xx, die von Ihrem Ursprung zurückgegeben wurden. Wenn Sie eine benutzerdefinierte Fehlerseite für einen HTTP-Statuscode konfiguriert haben, wird die benutzerdefinierte Fehlerseite CloudFront zwischengespeichert.

> **Note**
>
> Wenn Sie die Richtlinie für CachingDisabled verwaltete Caches verwenden, CloudFront werden diese Statuscodes oder benutzerdefinierten Fehlerseiten nicht zwischengespeichert.

404	Not Found
414	Anfrage-URI zu lang

500	Internal Server Error
501	Nicht implementiert
502	Bad Gateway
503	Service nicht verfügbar
504	Gateway-Timeout

HTTP 4xx-Statuscodes, die basierend auf CloudFront Headern zwischengespeichert werden **`Cache-Control`**

CloudFront speichert nur die folgenden HTTP 4xx-Statuscodes, die von Ihrem Ursprung zurückgegeben werden, zwischenspeichert, wenn Ihr Ursprung einen OR-Header zurückgibt. `Cache-Control max-age` `Cache-Control s-maxage` Wenn du eine benutzerdefinierte Fehlerseite für einen dieser HTTP-Statuscodes konfiguriert hast und dein Origin einen der Cache-Control-Header zurückgibt, wird die benutzerdefinierte Fehlerseite CloudFront zwischengespeichert.

400	Inkorrekte Anfrage
403	Forbidden
405	Method Not Allowed
412[1]	Vorbedingung fehlgeschlagen
415[1]	Unsupported Media Type (Nicht unterstützter Medientyp)

[1] unterstützt CloudFront nicht die Erstellung benutzerdefinierter Fehlerseiten für diese HTTP-Statuscodes.

Generieren Sie benutzerdefinierte Fehlerantworten

Wenn ein Objekt, über das Sie bereitstellen, aus irgendeinem Grund nicht verfügbar CloudFront ist, gibt Ihr Webserver in der Regel einen entsprechenden HTTP-Statuscode zurück, CloudFront um darauf hinzuweisen. Wenn ein Betrachter beispielsweise eine ungültige URL anfordert, gibt Ihr Webserver einen HTTP-Statuscode 404 (Not Found) CloudFront zurück und gibt diesen Statuscode dann an den Betrachter zurück. CloudFront Anstatt diese Standardfehlerantwort zu verwenden, können Sie eine benutzerdefinierte Antwort erstellen, die zum Viewer CloudFront zurückkehrt.

Wenn Sie die Rückgabe einer benutzerdefinierten Fehlerseite für einen HTTP-Statuscode konfigurieren CloudFront , die benutzerdefinierte Fehlerseite jedoch nicht verfügbar ist, wird der Statuscode, den Sie von der Quelle CloudFront erhalten haben und die benutzerdefinierten Fehlerseiten enthalten, an den Betrachter CloudFront zurückgegeben. Nehmen wir zum Beispiel an, Ihr benutzerdefinierter Ursprung gibt einen Statuscode 500 zurück und Sie haben konfiguriert CloudFront , dass eine benutzerdefinierte Fehlerseite für einen 500-Statuscode aus einem Amazon S3-Bucket abgerufen wird. Jemand hat jedoch versehentlich die benutzerdefinierte Fehlerseite aus Ihrem Amazon S3 S3-Bucket gelöscht. CloudFront gibt einen HTTP-404-Statuscode (Nicht gefunden) an den Betrachter zurück, der das Objekt angefordert hat.

Wenn eine benutzerdefinierte Fehlerseite an einen Betrachter CloudFront zurückgegeben wird, zahlen Sie die CloudFront Standardgebühren für die benutzerdefinierte Fehlerseite, nicht die Gebühren für das angeforderte Objekt. Weitere Informationen zu CloudFront Gebühren finden Sie unter CloudFrontAmazon-Preise.

Themen

Konfigurieren Sie das Verhalten bei der Fehlerreaktion

Sie haben mehrere Optionen, um zu verwalten, wie CloudFront auf einen Fehler reagiert wird. Um benutzerdefinierte Fehlerantworten zu konfigurieren, können Sie die CloudFront Konsole, die CloudFront API oder verwenden AWS CloudFormation. Unabhängig davon, wie Sie die Konfiguration aktualisieren, sollten Sie die folgenden Tipps und Empfehlungen beachten:

- Speichern Sie Ihre benutzerdefinierten Fehlerseiten an einem Ort, auf den zugegriffen werden kann CloudFront. Wir empfehlen Ihnen, sie in einem Amazon-S3-Bucket zu speichern und sie nicht am selben Ort wie den Rest der Inhalte Ihrer Website oder Anwendung zu speichern. Wenn Sie die benutzerdefinierten Fehlerseiten auf demselben Ursprung wie Ihre Website oder Anwendung speichern und der Ursprung beginnt, 5xx-Fehler zurückzugeben, CloudFront können Sie die benutzerdefinierten Fehlerseiten nicht abrufen, da der Ursprungsserver nicht verfügbar ist. Weitere Informationen finden Sie unter Speichern Sie Objekte und benutzerdefinierte Fehlerseiten an verschiedenen Orten.
- Stellen Sie sicher, dass dieser Benutzer berechtigt CloudFront ist, Ihre benutzerdefinierten Fehlerseiten abzurufen. Wenn die benutzerdefinierten Fehlerseiten in Amazon S3 gespeichert sind, müssen die Seiten öffentlich zugänglich sein oder Sie müssen eine CloudFront Origin Access Control (OAC) konfigurieren. Wenn die benutzerdefinierten Fehlerseiten in einem benutzerdefinierten Ursprung gespeichert sind, müssen die Seiten öffentlich zugänglich sein.
- (Optional) Konfigurieren Sie Ihren Ursprung so, dass ein `Cache-Control`- oder `Expires`-Header zusammen mit den benutzerdefinierten Fehlerseiten hinzugefügt wird, wenn Sie möchten. Sie können auch die Einstellung Minimum TTL für Error Caching verwenden, um zu steuern, wie lange die benutzerdefinierten Fehlerseiten CloudFront zwischengespeichert werden. Weitere Informationen finden Sie unter Steuern Sie, wie lange Fehler CloudFront zwischengespeichert werden.

Konfigurieren Sie benutzerdefinierte Fehlerantworten

Um benutzerdefinierte Fehlerantworten in der CloudFront Konsole zu konfigurieren, benötigen Sie eine CloudFront Distribution. In der Konsole stehen die Konfigurationseinstellungen für benutzerdefinierte Fehlerantworten nur für vorhandene Verteilungen zur Verfügung. Informationen zum Erstellen einer Verteilung finden Sie unter Beginnen Sie mit einer CloudFront Standarddistribution.

Console

Konfigurieren benutzerdefinierter Fehlerantworten (Konsole)

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Seite Verteilungen in der CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home#distributions.
2. Wählen Sie in der Liste der Verteilungen die zu aktualisierende Verteilung aus.

3. Wählen Sie die Registerkarte Fehlerseiten und wählen Sie dann Benutzerdefinierte Fehlerantwort erstellen.
4. Geben Sie die entsprechenden Werte ein. Weitere Informationen finden Sie unter Benutzerdefinierte Fehlerseiten und Zwischenspeicherung von Fehlern.
5. Nachdem Sie die gewünschten Werte eingegeben haben, wählen Sie Erstellen.

CloudFront API or AWS CloudFormation

Um benutzerdefinierte Fehlerantworten mit der CloudFront API oder zu konfigurieren AWS CloudFormation, verwenden Sie den `CustomErrorResponse` Typ in einer Distribution. Weitere Informationen finden Sie hier:

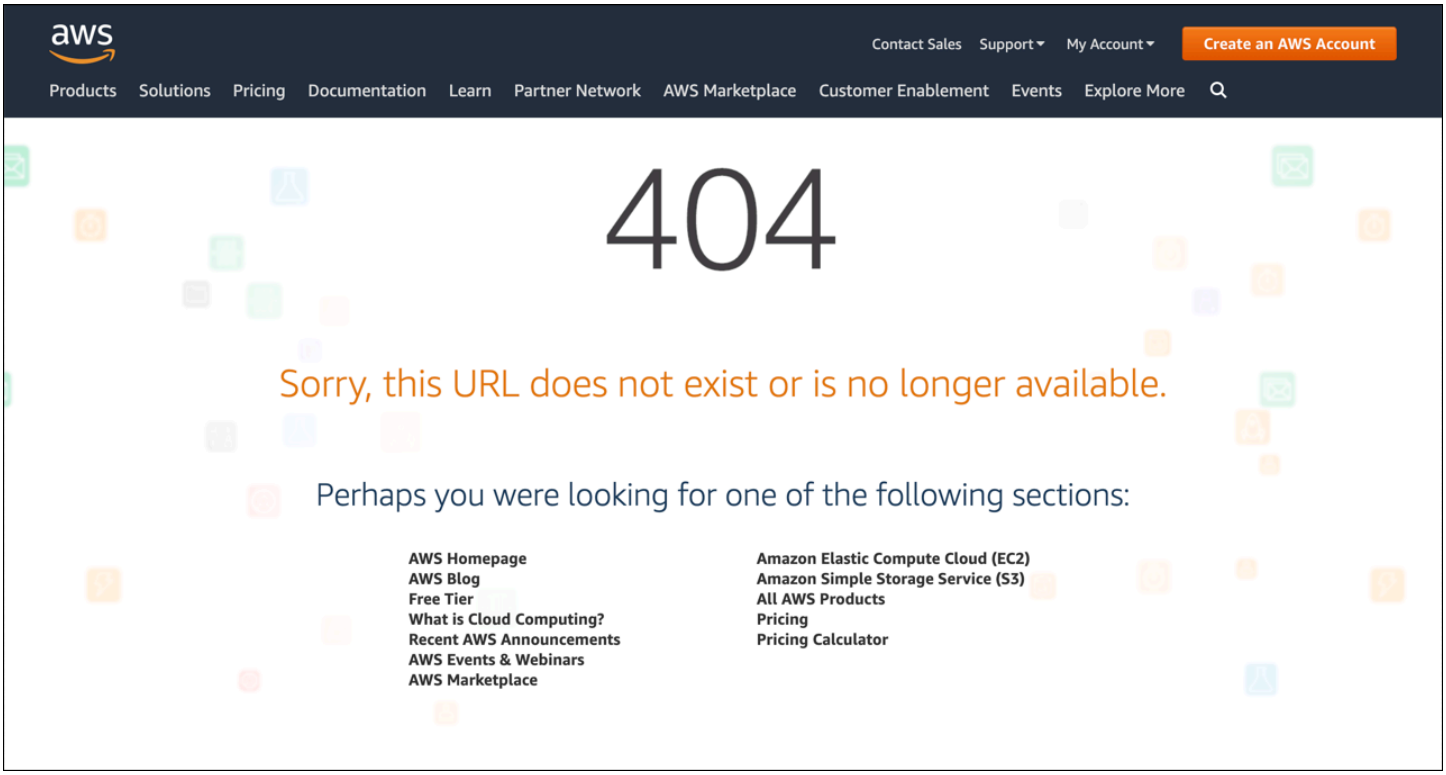
- AWS::CloudFront::Distribution CustomErrorResponse im AWS CloudFormation - Benutzerhandbuch
- CustomErrorResponsein der Amazon CloudFront API-Referenz

Erstellen Sie eine benutzerdefinierte Fehlerseite für bestimmte HTTP-Statuscodes

Wenn Sie statt der Standardmeldung lieber eine benutzerdefinierte Fehlermeldung anzeigen möchten, z. B. eine Seite, die dieselbe Formatierung wie der Rest Ihrer Website verwendet, können Sie ein Objekt (z. B. eine HTML-Datei), das Ihre benutzerdefinierte Fehlermeldung enthält, an den Viewer CloudFront zurückgeben lassen.

Um die Datei, die Sie zurückgeben möchten, und die Fehler, für die die Datei zurückgegeben werden soll, anzugeben, aktualisieren Sie Ihre CloudFront Distribution, sodass diese Werte angegeben werden. Weitere Informationen finden Sie unter Konfigurieren Sie das Verhalten bei der Fehlerreaktion.

Zum Beispiel ist das Folgende eine benutzerdefinierte Fehlerseite:



Sie können ein anderes Objekt für jeden unterstützten HTTP-Statuscode festlegen oder Sie können dasselbe Objekt für alle unterstützten Statuscodes verwenden. Es ist möglich, benutzerdefinierte Fehlerseiten für einige Statuscodes festzulegen und für andere nicht.

Die Objekte, über die Sie bereitstellen, CloudFront können aus verschiedenen Gründen nicht verfügbar sein. Diese gliedern sich in zwei große Kategorien:

- Client-Fehler weisen auf ein Problem mit der Anfrage hin. Beispielsweise ist ein Objekt mit dem angegebenen Namen nicht verfügbar oder der Benutzer verfügt nicht über die erforderlichen Berechtigungen, um ein Objekt in Ihrem Amazon S3-Bucket abzurufen. Wenn ein Client-Fehler auftritt, gibt der Ursprung einen HTTP-Statuscode im Bereich 4xx bis CloudFront zurück.
- Server-Fehler weisen auf ein Problem mit dem Ursprungs-Server hin. Beispielsweise ist der HTTP-Server ausgelastet oder nicht verfügbar. Wenn ein Serverfehler auftritt, gibt Ihr Ursprungsserver entweder einen HTTP-Statuscode im Bereich 5xx zurück oder CloudFront er erhält für einen bestimmten Zeitraum keine Antwort von Ihrem Ursprungsserver und nimmt den Statuscode 504 an (Gateway Timeout). CloudFront

Zu den HTTP-Statuscodes, für die eine benutzerdefinierte Fehlerseite zurückgegeben werden CloudFront kann, gehören die folgenden:

- 400, 403, 404, 405, 414, 416
- 500, 501, 502, 503, 504

> **Hinweise**
>
> - Wenn CloudFront erkannt wird, dass die Anfrage möglicherweise unsicher ist, wird anstelle einer benutzerdefinierten Fehlerseite ein 400-Fehler (Bad Request) CloudFront zurückgegeben.
> - Sie können eine benutzerdefinierte Fehlerseite für den HTTP-Statuscode 416 (Requested Range Not Satisfiable) erstellen und Sie können den HTTP-Statuscode ändern, der den Zuschauern CloudFront zurückgegeben wird, wenn Ihr Absender den Statuscode 416 an zurückgibt. CloudFront Weitere Informationen finden Sie unter Ändern Sie die Antwortcodes, die zurückgegeben wurden von CloudFront. Status-Code 416-Antworten werden jedoch CloudFront nicht zwischengespeichert, sodass dieser auch dann nicht verwendet wird, wenn Sie für den Statuscode 416 einen Wert für Error Caching Minimum TTL angeben. CloudFront
> - In einigen Fällen CloudFront gibt es keine benutzerdefinierte Fehlerseite für den HTTP-Statuscode 503 zurück, selbst wenn Sie dies konfigurieren CloudFront . Wenn der CloudFront Fehlercode `Capacity Exceeded` oder ist`Limit Exceeded`, wird ein 503-Statuscode an den Viewer CloudFront zurückgegeben, ohne Ihre benutzerdefinierte Fehlerseite zu verwenden.
> - Wenn Sie eine benutzerdefinierte Fehlerseite erstellt haben, CloudFront wird `Connection: keep-alive` für die folgenden Antwortcodes `Connection: close` oder zurückgegeben:
>   - CloudFront gibt `Connection: close` für Statuscodes zurück: 400, 405, 414, 416, 500, 501
>   - CloudFront gibt `Connection: keep-alive` für Statuscodes zurück: 403, 404, 502, 503, 504

Eine ausführliche Erläuterung, wie CloudFront mit Fehlerantworten aus Ihrem System umgegangen wird, finden Sie unterWie CloudFront werden die HTTP-Statuscodes 4xx und 5xx von Ihrem Ursprung verarbeitet.

Speichern Sie Objekte und benutzerdefinierte Fehlerseiten an verschiedenen Orten

Wenn Sie Ihre Objekte und Ihre benutzerdefinierten Fehlerseiten an verschiedenen Orten speichern möchten, muss Ihre Verteilung ein Cache-Verhalten mit den folgenden Eigenschaften enthalten:

- Der Wert von Path Pattern stimmt mit dem Pfad zu Ihren benutzerdefinierten Fehlermeldungen überein. Angenommen, Sie haben benutzerdefinierte Fehlerseiten für 4xx-Fehler in einem Amazon S3-Bucket in einem Verzeichnis mit dem Namen gespeicher `/4xx-errors`. Ihre Verteilung muss ein Cache-Verhalten beinhalten, für welches das Pfadmuster Anfragen für Ihre benutzerdefinierten Fehlerseiten an diesen Ort weiterleitet, z. B, `/4xx-errors/*`.
- Der Wert von Origin legt den Wert von Origin ID für den Ursprung fest, der Ihre benutzerdefinierten Fehlerseiten enthält.

Weitere Informationen finden Sie unter Einstellungen für das Cache-Verhalten.

Ändern Sie die Antwortcodes, die zurückgegeben wurden von CloudFront

Sie können so konfigurieren CloudFront , dass dem Betrachter ein anderer HTTP-Statuscode zurückgegeben wird als der, den CloudFront er vom Absender erhalten hat. Wenn Ihr Absender beispielsweise den Statuscode 500 an zurückgibtCloudFront, CloudFront möchten Sie möglicherweise eine benutzerdefinierte Fehlerseite und den Statuscode 200 (OK) an den Betrachter zurückgeben. Es gibt eine Vielzahl von Gründen, warum Sie dem Betrachter möglicherweise einen Statuscode zurückgeben CloudFront möchten, der sich von dem unterscheidet, zu dem Ihr Ursprung zurückgekehrt istCloudFront:

- Einige Internetgeräte (beispielsweise einige Firewalls und Unternehmens-Proxys) fangen HTTP 4xx- und 5xx-Statuscodes ab und verhindern, dass die Antwort an den Betrachter zurückgegeben wird. Wenn Sie in diesem Szenario `200` ersetzen, wird die Antwort nicht abgefangen.
- Wenn Sie nicht zwischen verschiedenen Client- oder Serverfehlern unterscheiden möchten, können Sie `400` oder `500` als den Wert angeben, der für alle 4xx- oder 5xx-Statuscodes CloudFront zurückgegeben wird.
- Möglicherweise möchten Sie einen `200`-Statuscode (OK) und eine statische Website zurückgeben, sodass Ihre Kunden nicht wissen, dass die Website nicht verfügbar ist.

Wenn Sie CloudFront Standardprotokolle aktivieren und so konfigurierenCloudFront , dass der HTTP-Statuscode in der Antwort geändert wird, enthält der Wert der `sc-status` Spalte in den Protokollen den von Ihnen angegebenen Statuscode. Der Wert der `x-edge-result-type`-Spalte wird jedoch nicht beeinflusst. Sie enthält den Ergebnistyp der Antwort vom Ursprung. Nehmen wir zum Beispiel an, Sie konfigurieren CloudFront , dass der Statuscode von `200` an den Betrachter zurückgegeben wird, wenn der Ursprung `404` (Not Found) zu zurückkehrt CloudFront. Wenn der Ursprung auf eine Anfrage mit dem Statuscode `404` antwortet, ist der Wert in der Spalte `sc-status` im Protokoll `200`, der Wert in der Spalte `x-edge-result-type` jedoch `Error`.

Sie können so konfigurieren CloudFront , dass jeder der folgenden HTTP-Statuscodes zusammen mit einer benutzerdefinierten Fehlerseite zurückgegeben wird:

- 200
- 400, 403, 404, 405, 414, 416
- 500, 501, 502, 503, 504

Steuern Sie, wie lange Fehler CloudFront zwischengespeichert werden

CloudFront speichert Fehlerantworten für eine Standarddauer von 10 Sekunden im Cache. CloudFront sendet dann die nächste Anfrage für das Objekt an Ihre Quelle, um zu überprüfen, ob das Problem, das den Fehler verursacht hat, behoben wurde und das angeforderte Objekt verfügbar ist.

Sie können für jeden 4xx- und 5xx-Statuscode, der zwischengespeichert wird, die Dauer des Fehler-Cachings — die Mindest-TTL für das Fehler-Caching angeben. CloudFront (Weitere Informationen finden Sie unter HTTP-Statuscodes 4xx und 5xx, die zwischengespeichert werden CloudFront .) Wenn Sie eine Dauer angeben, beachten Sie Folgendes:

- Wenn Sie eine kurze Dauer für das Zwischenspeichern von Fehlern angeben, werden mehr Anfragen an Ihren Ursprung weitergeleitet, CloudFront als wenn Sie eine längere Dauer angeben. Bei 5xx-Fehlern verschlimmert dies möglicherweise das Problem, das ursprünglich dazu geführt hat, dass Ihr Ursprung einen Fehler zurückgibt.
- Wenn Ihr Absender einen Fehler für ein Objekt zurückgibt, CloudFront beantwortet er Anfragen für das Objekt entweder mit der Fehlerantwort oder mit Ihrer benutzerdefinierten Fehlerseite, bis die Dauer der Fehlerzwischenspeicherung abgelaufen ist. Wenn Sie eine lange Dauer für das Zwischenspeichern von Fehlern angeben, reagiert das CloudFront Objekt möglicherweise noch lange auf Anfragen mit einer Fehlerantwort oder Ihrer benutzerdefinierten Fehlerseite, nachdem das Objekt wieder verfügbar ist.

> **Note**
>
> Sie können eine benutzerdefinierte Fehlerseite für HTTP-Statuscode 416 (Requested Range Not Satisfiable) erstellen und Sie können den HTTP-Statuscode ändern, den CloudFront an Betrachter zurückgibt, wenn Ihr Ursprung einen Statuscode 416 an CloudFront zurückgibt. (Weitere Informationen finden Sie unter Ändern Sie die Antwortcodes, die zurückgegeben wurden von CloudFront.) Status-Code 416-Antworten werden jedoch CloudFront nicht zwischengespeichert. Selbst wenn Sie einen Wert für Error Caching Minimum TTL für den Statuscode 416 angeben, wird dieser Wert nicht verwendet. CloudFront

Wenn Sie steuern möchten, wie lange Fehler für einzelne Objekte CloudFront zwischengespeichert werden, können Sie Ihren Ursprungsserver so konfigurieren, dass er der Fehlerantwort für dieses Objekt den entsprechenden Header hinzufügt.

Wenn der Ursprung eine `Cache-Control: s-maxage` OR-Direktive `Cache-Control: max-age` oder einen `Expires` Header hinzufügt, werden Fehlerantworten CloudFront zwischengespeichert, je nachdem, welcher Wert im Header oder der Mindest-TTL für Error Caching gilt.

> **Note**
>
> Beachten Sie, dass die Werte für `Cache-Control: max-age` und `Cache-Control: s-maxage` nicht größer als der Wert für Maximum TTL sein können, der für das Cache-Verhalten festgelegt wurde, für das die Fehlerseite abgerufen wird.

Fügt der Ursprung weitere **`Cache-Control`** Direktiven hinzu oder fügt er keine Header hinzu, werden die Fehlerantworten mit dem Wert von Error CloudFront Caching Minimum TTL zwischengespeichert.

Wenn die Ablaufzeit für einen 4xx- oder 5xx-Statuscode für ein Objekt länger ist, als Sie warten möchten, und das Objekt wieder verfügbar ist, können Sie den zwischengespeicherten Fehlercode mit der URL des angefragten Objekts aufheben. Wenn Ihr Ursprung eine Fehlermeldung für mehrere Objekte zurückgibt, müssen Sie die Gültigkeit aller Objekte einzeln aufheben. Weitere Informationen zur Aufhebung der Gültigkeit von Objekten finden Sie unter Machen Sie Dateien ungültig, um Inhalte zu entfernen.

Wenn Sie das Caching für einen S3-Bucket-Ursprung aktiviert haben und in Ihrer CloudFront Distribution einen Fehler beim Zwischenspeichern einer Mindest-TTL von 0 Sekunden konfigurieren,

wird immer noch ein Fehler beim Zwischenspeichern einer Mindest-TTL von 1 Sekunde für Fehler mit S3-Ursprung angezeigt. CloudFront tut dies, um Ihren Origin vor S-Angriffen zu schützen. DDo Dies gilt nicht für andere Arten von Ursprüngen.

Inhalte hinzufügen, entfernen oder ersetzen, die CloudFront verbreitet werden

In diesem Abschnitt wird erklärt, wie Sie sicherstellen, dass Sie auf die Inhalte zugreifen CloudFront können, die Ihren Zuschauern angezeigt werden sollen, wie Sie die Objekte auf Ihrer Website oder in Ihrer Anwendung angeben und wie Sie Inhalte entfernen oder ersetzen.

Themen

Fügen Sie Inhalte hinzu, die CloudFront verbreitet werden, und greifen Sie darauf zu

Wenn Sie Inhalte (Objekte) verteilen CloudFront möchten, fügen Sie Dateien zu einer der Quellen hinzu, die Sie für die Verteilung angegeben haben, und Sie stellen einen CloudFront Link zu den Dateien bereit. Ein CloudFront Edge-Standort ruft die neuen Dateien erst von einem Ursprung ab, wenn der Edge-Standort Viewer-Anfragen für sie erhält. Weitere Informationen finden Sie unter Wie liefert Inhalte CloudFront .

Wenn Sie eine Datei hinzufügen, die Sie verteilen CloudFront möchten, stellen Sie sicher, dass Sie sie zu einem der in Ihrer Distribution angegebenen Amazon S3 S3-Buckets oder, für einen benutzerdefinierten Ursprung, zu einem Verzeichnis in der angegebenen Domain hinzufügen. Überprüfen Sie darüber hinaus, ob das Pfadmuster im entsprechenden Cache-Verhalten Anfragen an den richtigen Ursprung sendet.

Nehmen wir beispielsweise an, dass das Pfadmuster für ein Cache-Verhalten is `*.html`. Wenn Sie kein anderes Cache-Verhalten für die Weiterleitung von Anfragen an diesen Ursprung konfiguriert haben, CloudFront werden nur `*.html` Dateien weitergeleitet. In diesem Szenario CloudFront

werden beispielsweise JPG-Dateien, die Sie hochladen, niemals an den Ursprungsserver verteilt, da Sie kein Cache-Verhalten erstellt haben, das JPG-Dateien einschließt.

CloudFront Server bestimmen nicht den MIME-Typ für die Objekte, die sie bereitstellen. Wenn Sie eine Datei auf Ihren Ursprungs-Server hochladen, sollten Sie das `Content-Type`-Header-Feld dafür festlegen.

Verwenden Sie die Dateiversionierung, um Inhalte mit einer CloudFront Distribution zu aktualisieren oder zu entfernen

Um bestehende Inhalte zu aktualisieren, die CloudFront für die Verteilung für Sie eingerichtet wurden, empfehlen wir, in Datei- oder Ordnernamen eine Versions-ID zu verwenden. Auf diese Weise haben Sie die Kontrolle über die Verwaltung der bereitgestellten CloudFront Inhalte.

Aktualisieren Sie vorhandene Dateien mit versionierten Dateinamen

Wenn Sie vorhandene Dateien in einer CloudFront Distribution aktualisieren, empfehlen wir Ihnen, eine Art Versionskennung entweder in Ihre Datei- oder in Ihre Verzeichnisnamen aufzunehmen, um eine bessere Kontrolle über Ihren Inhalt zu haben. Bei dieser Kennung kann es sich um einen Datum-Zeitstempel, eine fortlaufende Nummer oder eine andere Methode der Unterscheidung zwei Versionen desselben Objekts handeln.

Anstatt beispielsweise eine Grafikdatei mit image.jpg zu benennen, könnten Sie diese image_1.jpg nennen. Wenn Sie eine neue Version der Datei bereitstellen möchten, würden Sie die neue Datei mit image_2.jpg benennen und die Links in Ihrer Web-Anwendung oder auf Ihrer Website aktualisieren, damit sie auf image_2.jpg verweisen. Alternativ könnten Sie alle Grafiken in einem images_v1-Verzeichnis ablegen und, wenn Sie neue Versionen von einer oder mehreren Grafiken bereitstellen möchten, ein neues images_v2-Verzeichnis erstellen und Ihre Links aktualisieren, damit sie auf dieses Verzeichnis verweisen. Bei der Versionierung müssen Sie nicht warten, bis ein Objekt abläuft, bevor Sie mit der Bereitstellung einer neuen Version CloudFront beginnen, und Sie müssen auch nicht für die Objektinvalidierung bezahlen.

Auch wenn Sie Ihre Dateien versionieren, sollten Sie ein Ablaufdatum festlegen. Weitere Informationen finden Sie unter Verwalten Sie, wie lange Inhalte im Cache verbleiben (Ablauf).

> **Note**
>
> Das Festlegen von versionsgesteuerten Dateinamen oder Verzeichnisnamen steht nicht mit dem Amazon S3-Objekt-Versioning im Zusammenhang.

Inhalte entfernen, um sie CloudFront nicht zu verteilen

Sie können Dateien aus Ihrem Ursprungs-Server entfernen, die nicht länger in Ihrer CloudFront-Verteilung enthalten sein sollen. Zeigt den Zuschauern jedoch CloudFront weiterhin Inhalte aus dem Edge-Cache an, bis die Dateien ablaufen.

Wenn Sie eine Datei direkt entfernen möchten, müssen Sie einen der folgenden Schritte ausführen:

- Verwenden der Dateiversionierung. Wenn Sie die Versionsverwaltung verwenden, haben verschiedene Versionen einer Datei unterschiedliche Namen, die Sie in Ihrer CloudFront Distribution verwenden können, um zu ändern, welche Datei an die Betrachter zurückgegeben wird. Weitere Informationen finden Sie unter Aktualisieren Sie vorhandene Dateien mit versionierten Dateinamen.
- Aufheben der Gültigkeit der Datei. Weitere Informationen finden Sie unter Machen Sie Dateien ungültig, um Inhalte zu entfernen.

Passen Sie das URL-Format für Dateien an in CloudFront

Nachdem Sie Ihren Ursprung mit den Objekten (Inhalten) eingerichtet haben, die Sie Ihren Zuschauern CloudFront zur Verfügung stellen möchten, müssen Sie in Ihrer Website oder Ihrem Anwendungscode die richtige Option verwenden, URLs um auf diese Objekte zu verweisen, damit sie CloudFront bereitgestellt werden können.

Der Domainname, den Sie in den URLs for-Objekten auf Ihren Webseiten oder in Ihrer Webanwendung verwenden, kann einer der folgenden sein:

- Der Domainname, der z. `d111111abcdef8.cloudfront.net` B. CloudFront automatisch zugewiesen wird, wenn Sie eine Distribution erstellen
- Ihr eigener Domänenname, z. B. `example.com`

Sie können beispielsweise eine der folgenden Optionen verwenden, URLs um die Datei `image.jpg` zurückzugeben:

`https://d111111abcdef8.cloudfront.net/images/image.jpg`

`https://example.com/images/image.jpg`

Sie verwenden dasselbe URL-Format, unabhängig davon, ob Sie den Inhalt in Amazon S3-Buckets oder an einem benutzerdefinierten Ursprung wie Ihren eigenen Webservern speichern.

> **Note**
>
> Das URL-Format richtet sich zum Teil nach dem Wert, den Sie für Origin Path in Ihrer Verteilung festlegen. Dieser Wert gibt CloudFront einen obersten Verzeichnispfad für Ihre Objekte an. Weitere Informationen zum Einrichten eines Ursprungspfades beim Erstellen einer Verteilung finden Sie unter Ursprungspfad.

Weitere Informationen zu URL-Formaten finden Sie in den folgenden Abschnitten.

Verwenden Sie Ihren eigenen Domainnamen (example.com)

Anstatt den Standard-Domainnamen zu verwenden, der Ihnen bei der Erstellung einer Distribution CloudFront zugewiesen wird, können Sie einen alternativen Domainnamen hinzufügen, mit dem Sie einfacher arbeiten können, z. `example.com` Wenn Sie Ihren eigenen Domainnamen mit einrichten CloudFront, können Sie eine URL wie die folgende für Objekte in Ihrer Distribution verwenden:

`https://example.com/images/image.jpg`

Wenn Sie beabsichtigen, HTTPS zwischen Zuschauern und zu verwenden CloudFront, finden Sie weitere Informationen unterVerwenden Sie alternative Domainnamen und HTTPS.

Verwenden Sie einen abschließenden Schrägstrich (/) in URLs

Wenn Sie URLs für Verzeichnisse in Ihrer CloudFront Distribution angeben, wählen Sie entweder, ob Sie immer einen abschließenden Schrägstrich oder niemals einen abschließenden Schrägstrich verwenden möchten. Wählen Sie beispielsweise nur eines der folgenden Formate für alle Ihre: URLs

`https://d111111abcdef8.cloudfront.net/images/`

```
https://d111111abcdef8.cloudfront.net/images
```

Warum ist das wichtig?

Beide Formate funktionieren für Links zu CloudFront Objekten, aber wenn Sie konsistent sind, können Sie Probleme vermeiden, wenn Sie ein Verzeichnis später für ungültig erklären möchten. CloudFront speichert URLs exakt so, wie sie definiert sind, einschließlich nachgestellter Schrägstriche. Wenn Ihr Format also inkonsistent ist, müssen Sie das Verzeichnis URLs mit und ohne Schrägstrich ungültig machen, um sicherzustellen, dass das Verzeichnis gelöscht wird. CloudFront

Es ist lästig, beide URL-Formate deaktivieren zu müssen und kann zu zusätzlichen Kosten führen. Das liegt daran, dass Sie, wenn Sie die Anzahl der Ungültigerklärungen verdoppeln müssen URLs, um beide Typen abzudecken, die für diesen Monat zulässig sind, möglicherweise überschreiten. Tritt dieser Fall ein, müssen Sie für alle Invalidierungen bezahlen, auch wenn in CloudFront jeweils nur ein Format für jede Verzeichnis-URL vorhanden ist.

Erstellen Sie signierte Inhalte mit eingeschränktem Inhalt URLs

Wenn Sie Inhalte haben, auf die Sie den Zugriff einschränken möchten, können Sie signierte Inhalte erstellen URLs. Wenn Sie Ihre Inhalte beispielsweise nur an Benutzer verteilen möchten, die sich authentifiziert haben, können Sie Inhalte erstellen, URLs die nur für einen bestimmten Zeitraum gültig sind oder die nur über eine bestimmte IP-Adresse verfügbar sind. Weitere Informationen finden Sie unter Stellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit.

Geben Sie ein Standard-Stammobjekt an

Sie können so konfigurieren CloudFront , dass ein bestimmtes Objekt (das Standard-Root-Objekt) zurückgegeben wird, wenn ein Benutzer (Viewer) die Stamm-URL für Ihre Distribution anfordert, anstatt ein Objekt in Ihrer Distribution anzufordern. Sie können ein Standard-Stammobjekt verwenden, um zu verhindern, dass der Inhalt Ihrer Distribution offengelegt wird.

Inhalt

So geben Sie ein Standardstammobjekt an

Um zu vermeiden, dass der Inhalt Ihrer Distribution offengelegt wird oder ein Fehler zurückgegeben wird, geben Sie ein Standard-Stammobjekt für Ihre Distribution an. Sie können den genauen Dateinamen oder den Pfad zur Datei angeben. Wenn es sich bei Ihrem Stammobjekt beispielsweise um eine `index.html` Datei handelt, können Sie diesen Dateinamen angeben. Wenn sich Ihre `index.html` Datei in einem anderen Ordner befindet, geben Sie stattdessen den Pfad an, z. *`exampleFolderName`*`/index.html` B. Wenn Sie einen Pfad zum Standard-Stammobjekt festlegen, geben Viewer-Anfragen an die Stamm-URL der Distribution die angegebene Datei aus diesem Pfad zurück. Sie können einen Dateipfad verwenden, um Ihre Inhalte flexibler am Ursprung zu organisieren, da sich Ihr Standard-Stammobjekt in einem Ordner statt auf Stammebene befinden kann.

So geben Sie ein Standardstammobjekt für Ihre Verteilung an

1. Laden Sie das Standardstammobjekt auf den Ursprung hoch, auf den Ihre Verteilung zeigt.

   Der Dateityp kann jeder von CloudFront unterstützte Typ sein. Eine Liste der Einschränkungen für den Dateinamen finden Sie im `DefaultRootObject` Element in DistributionConfigder Amazon CloudFront API-Referenz.

   > **Note**
   >
   > Wenn der Dateiname des Standard-Stammobjekts zu lang ist oder ein ungültiges Zeichen enthält, wird der Fehler CloudFront zurückgegeben`HTTP 400 Bad Request - InvalidDefaultRootObject`. Außerdem wird der Code für 10 Sekunden CloudFront zwischengespeichert (standardmäßig) und die Ergebnisse werden in die Zugriffsprotokolle geschrieben.

2. Vergewissern Sie sich, dass die Berechtigungen für das Objekt CloudFront mindestens Lesezugriff gewähren.

   Weitere Informationen zu Amazon-S3-Berechtigungen finden Sie unter Identity and Access Management in Amazon S3 im Amazon-Simple-Storage-Service-Benutzerhandbuch.

3. Aktualisieren Sie Ihre Distribution mithilfe der CloudFront Konsole oder der CloudFront API so, dass sie auf das Standard-Root-Objekt verweist.

   So geben Sie mithilfe der CloudFront Konsole ein Standard-Root-Objekt an:

a. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.

b. Wählen Sie in der Liste der Verteilungen im oberen Bereich die zu aktualisierende Verteilung aus.

c. Wählen Sie im Bereich Settings (Einstellungen) auf der Registerkarte General (Allgemein) die Option Edit (Bearbeiten) aus.

d. Geben Sie im Dialogfeld Einstellungen bearbeiten im Feld Standard-Stammobjekt den Dateinamen oder Pfad zum Standard-Stammobjekt ein.

> **Tip**
>
> Ihre Zeichenfolge darf nicht mit einem Schrägstrich (`/`) beginnen. Geben Sie nur den Objektnamen oder den Pfad zum Objekt an. Verwenden Sie beispielsweise `index.html` oder*`exampleFolderName`*`/index.html`. Die Angabe eines `/`*`exampleFolderName`*`/index.html` oder `/index.html` kann zu dem Fehler 403 Access Denied führen.

e. Wählen Sie Änderungen speichern aus.

Um Ihre Konfiguration mithilfe der CloudFront API zu aktualisieren, geben Sie einen Wert für das `DefaultRootObject` Element in Ihrer Distribution an. Informationen zur Verwendung der CloudFront API zur Angabe eines Standard-Root-Objekts finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.

4. Vergewissern Sie sich, dass Sie das Standardstammobjekt aktiviert haben, indem Ihre Stamm-URL anfragen. Wenn Ihr Standardstammobjekt im Browser nicht angezeigt wird, führen Sie die folgenden Schritte aus:

   a. Vergewissern Sie sich, dass Ihre Distribution vollständig bereitgestellt wurde, indem Sie den Status Ihrer Distribution in der CloudFront Konsole einsehen.

   b. Wiederholen Sie die Schritte 2 und 3, um sicherzustellen, dass Sie die richtigen Berechtigungen erteilt und die Konfiguration Ihrer Verteilung ordnungsgemäß aktualisiert haben, um das Standardstammobjekt anzugeben.

So funktioniert das Standardstammobjekt

Angenommen, die folgende Anfrage zeigt beispielsweise auf das Objekt `image.jpg`:

`https://d111111abcdef8.cloudfront.net/image.jpg`

Im Gegensatz dazu zeigt die folgende Anfrage auf die Stamm-URL derselben Verteilung anstatt auf ein bestimmtes Objekt, wie im ersten Beispiel:

`https://d111111abcdef8.cloudfront.net/`

Wenn Sie ein Standardstammobjekt definieren, gibt eine Endbenutzeranfrage, welche den Stamm Ihrer Verteilung aufruft, das Standardstammobjekt zurück. Wenn Sie beispielsweise die Datei `index.html` als Standardstammobjekt bestimmen, gibt eine Anfrage für:

`https://d111111abcdef8.cloudfront.net/`

Rückgabe:

`https://d111111abcdef8.cloudfront.net/index.html`

> **Note**
>
> CloudFront bestimmt nicht, ob eine URL mit mehreren abschließenden Schrägstrichen (`https://d111111abcdef8.cloudfront.net///`) äquivalent zu ist. `https://d111111abcdef8.cloudfront.net/` Ihr Ursprungsserver führt diesen Vergleich durch.

Wenn Sie ein Standardstammobjekt definieren, gibt eine Endbenutzeranfrage für ein Unterverzeichnis Ihrer Verteilung nicht das Standardstammobjekt zurück. Nehmen wir zum Beispiel an, `index.html` es ist Ihr Standard-Root-Objekt, das eine Endbenutzeranfrage für das `install` Verzeichnis in Ihrer Distribution CloudFront empfängt: CloudFront

`https://d111111abcdef8.cloudfront.net/install/`

CloudFront gibt das Standard-Stammobjekt nicht zurück, auch wenn eine Kopie von im `install` Verzeichnis `index.html` erscheint. Wenn Sie jedoch einen Pfad zu Ihrem Standard-Root-Objekt angegeben haben, gibt (`install/index.html`) CloudFront das Standard-Root-Objekt für Endbenutzeranfragen für das `install` Verzeichnis zurück

Wenn Sie Ihre Verteilung so konfigurieren, dass alle HTTP-Methoden, die CloudFront unterstützt, zulässig sind, gilt das Standardstammobjekt für alle Methoden. Wenn Ihr Standard-Root-Objekt

beispielsweise index.php ist und Sie Ihre Anwendung schreiben, um eine `POST` Anfrage an das Stammverzeichnis Ihrer Domain (https://example.com) zu senden, CloudFront sendet die Anfrage an https://example.com /index.php.

Das Verhalten von CloudFront Standard-Root-Objekten unterscheidet sich vom Verhalten von Amazon S3 S3-Indexdokumenten. Wenn Sie einen Amazon S3-Bucket als Website konfigurieren und das Indexdokument angeben, gibt Amazon S3 das Indexdokument auch dann zurück, wenn ein Benutzer ein Unterverzeichnis im Bucket anfragt. (Eine Kopie des Indexdokuments muss in allen Unterverzeichnissen angezeigt werden.) Weitere Informationen zum Konfigurieren von Amazon-S3-Buckets als Websites und zu Indexdokumenten finden Sie im Kapitel Hosting von Websites auf Amazon S3 im Benutzerhandbuch zu Amazon Simple Storage Service.

> **Important**
>
> Denken Sie daran, dass ein Standard-Root-Objekt nur für Ihre CloudFront Distribution gilt. Sie müssen noch die Sicherheit für Ihren Ursprung verwalten. Wenn Sie beispielsweise einen Amazon S3 S3-Ursprung verwenden, müssen Sie Ihren Amazon S3-Bucket dennoch ACLs entsprechend einrichten, um die gewünschte Zugriffsebene für Ihren Bucket sicherzustellen.

Wie CloudFront funktioniert, wenn Sie kein Root-Objekt definieren

Wenn Sie kein Standardstammobjekt definieren, werden Anfragen für den Stamm Ihrer Verteilung an Ihren Ursprungs-Server weitergeleitet. Wenn Sie einen Amazon S3-Ursprung verwenden, kann Folgendes zurückgegeben werden:

- Eine Liste der Inhalte Ihres Amazon S3 S3-Buckets — Unter einer der folgenden Bedingungen sind die Inhalte Ihres Ursprungs für jeden sichtbar, der auf Ihre Distribution CloudFront zugreift:
  - Ihr Bucket ist nicht ordnungsgemäß konfiguriert.
  - Die Amazon S3-Berechtigungen für den mit Ihrer Verteilung verknüpften Bucket und für die Objekte im Bucket gewähren allen Benutzern den Zugriff.
  - Ein Endbenutzer greift mit der Stamm-URL Ihres Ursprungs auf den Ursprung zu.
- Eine Liste der privaten Inhalte Ihres Ursprungs — Wenn Sie Ihren Ursprung als private Distribution konfigurieren (nur Sie und Sie CloudFront haben Zugriff), ist der Inhalt des Amazon S3 S3-Buckets, der mit Ihrer Distribution verknüpft ist, für jeden sichtbar, der über die Anmeldeinformationen für den Zugriff auf Ihre Distribution verfügt CloudFront. In diesem Fall können Benutzer nicht über die Stamm-URL Ihres Ursprungs auf Ihre Inhalte zugreifen. Weitere Informationen zum Verteilen von

privaten Inhalten finden Sie unter the section called “Beschränken Sie Inhalte mit signierten URLs und signierten Cookies”.

- `Error 403 Forbidden`— CloudFront gibt diesen Fehler zurück, wenn die Berechtigungen für den Amazon S3 S3-Bucket, der mit Ihrer Distribution verknüpft ist, oder die Berechtigungen für die Objekte in diesem Bucket allen den Zugriff CloudFront verweigern.

Machen Sie Dateien ungültig, um Inhalte zu entfernen

Wenn Sie eine Datei aus CloudFront Edge-Caches entfernen müssen, bevor sie abläuft, können Sie einen der folgenden Schritte ausführen:

- Heben Sie die Gültigkeit der Datei in Edge-Caches auf. Wenn ein Betrachter die Datei das nächste Mal anfordert, CloudFront kehrt er zum Ursprung zurück, um die neueste Version der Datei abzurufen.
- Verwenden Sie Datei-Versioning, um eine andere Version der Datei mit einem anderen Namen bereitzustellen. Weitere Informationen finden Sie unter Aktualisieren Sie vorhandene Dateien mit versionierten Dateinamen.

Themen

Wählen Sie zwischen der Ungültigkeit von Dateien und der Verwendung versionierter Dateinamen

Um die Versionen von Dateien zu steuern, die von Ihrer Verteilung bereitgestellt werden, können Sie entweder die Gültigkeit der Dateien aufheben oder diesen versionierte Dateinamen zuweisen. Wenn Sie Ihre Dateien häufig aktualisieren möchten, empfehlen wir, hauptsächlich Datei-Versioning zu verwenden. Die Gründe dafür sind folgende:

- Mit Versioning können Sie steuern, welche Datei von einer Anfrage zurückgegeben wird, auch wenn der Benutzer über eine Version verfügt, die entweder lokal oder hinter einem Unternehmens-Caching-Proxy zwischengespeichert ist. Wenn Sie die Gültigkeit der Datei aufheben, kann der Benutzer möglicherweise noch die alte Version aus diesen Caches anzeigen, bis sie abläuft.
- CloudFront Zugriffsprotokolle enthalten die Namen Ihrer Dateien, sodass die Versionierung die Analyse der Ergebnisse von Dateiänderungen erleichtert.
- Versioning bietet eine Möglichkeit, verschiedenen Benutzern verschiedene Versionen von Dateien bereitzustellen.
- Versioning vereinfacht den Wechsel zwischen verschiedenen Dateirevisionen.
- Versioning ist kostengünstiger. Sie müssen immer noch dafür bezahlen CloudFront , neue Versionen Ihrer Dateien an Edge-Standorte zu übertragen, aber Sie müssen nicht für die Ungültigkeit von Dateien bezahlen.

Weitere Informationen zu Datei-Versioning finden Sie unter Aktualisieren Sie vorhandene Dateien mit versionierten Dateinamen.

Ermitteln Sie, welche Dateien für ungültig erklärt werden sollen

Wenn Sie die Gültigkeit mehrerer Dateien aufheben möchten, wie beispielsweise aller Dateien in einem Verzeichnis oder aller Dateien, deren Namen mit den gleichen Zeichen beginnen, können Sie den Platzhalter `*` am Ende des Aufhebungspfads einfügen. Weitere Informationen zur Verwendung des Platzhalters `*` finden Sie unter Invalidation paths.

Um die Gültigkeit von Dateien aufzuheben, können Sie entweder den Pfad für einzelne Dateien festlegen oder einen Pfad, der mit dem Platzhalter `*` endet und für ein oder mehrere Dateien gelten könnte, wie in den folgenden Beispielen gezeigt:

- `/images/image1.jpg`
- `/images/image*`
- `/images/*`

Wenn Sie die Gültigkeit ausgewählter Dateien aufheben möchten, Ihre Benutzer jedoch nicht unbedingt auf alle Dateien in Ihrem Ursprung zugreifen, können Sie ermitteln, welche Dateien Betrachter von CloudFront abgefragt haben, und nur die Gültigkeit dieser Dateien aufheben. Um festzustellen, welche Dateien von Zuschauern angefordert wurden, aktivieren Sie die

CloudFront Zugriffsprotokollierung. Weitere Informationen zu Zugriffsprotokollen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).

Was Sie wissen müssen, wenn Sie Dateien für ungültig erklären

Wenn Sie eine Datei angeben, die ungültig gemacht werden soll, beachten Sie die folgenden Informationen:

Groß-/Kleinschreibung

Bei Invalidierungspfaden wird zwischen Groß- und Kleinschreibung unterschieden. `/images/Image.jpg`Geben Sie beispielsweise `/images/image.jpg` zwei verschiedene Dateien an.

Ändern des URI mithilfe einer Lambda-Funktion

Wenn Ihre CloudFront Distribution bei Viewer-Anforderungsereignissen eine Lambda-Funktion auslöst und die Funktion den URI der angeforderten Datei ändert, empfehlen wir Ihnen, beide für ungültig zu erklären, URIs um die Datei aus den CloudFront Edge-Caches zu entfernen:

- Der URI in der Viewer-Anfrage
- Die URI, nachdem sie von der Funktion geändert wurde

Example Beispiel

Angenommen, Ihre Lambda-Funktion ändert den URI für eine Datei von:

```
https://d111111abcdef8.cloudfront.net/index.html
```

Zu einer URI, die ein Sprachverzeichnis enthält:

```
https://d111111abcdef8.cloudfront.net/en/index.html
```

Um die Datei ungültig zu machen, müssen Sie die folgenden Pfade angeben:

- `/index.html`
- `/en/index.html`

Weitere Informationen finden Sie unter Invalidation paths.

Standardstammobjekt

Um die Gültigkeit des Standardstammobjekts (Datei) aufzuheben, geben Sie den Pfad auf die Weise an, auf die Sie auch den Pfad für alle anderen Dateien angeben. Weitere Informationen finden Sie unter So funktioniert das Standardstammobjekt.

Weiterleiten von Cookies

Wenn Sie so konfiguriert CloudFront haben, dass Cookies an Ihren Ursprung weitergeleitet werden, können CloudFront Edge-Caches mehrere Versionen der Datei enthalten. Wenn Sie eine Datei für ungültig erklären, wird jede zwischengespeicherte Version der Datei CloudFront ungültig gemacht, unabhängig von den zugehörigen Cookies. Sie können nicht auf der Grundlage der zugehörigen Cookies selektiv die Gültigkeit einiger Versionen aufheben und anderer Versionen nicht. Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.

Weiterleiten von Headern

Wenn Sie so konfiguriert haben CloudFront , dass eine Liste von Headern an Ihren Ursprung weitergeleitet und anhand der Werte der Header zwischengespeichert wird, können CloudFront Edge-Caches mehrere Versionen der Datei enthalten. Wenn Sie die Gültigkeit einer Datei aufheben, hebt CloudFront die Gültigkeit aller zwischengespeicherten Versionen der Datei auf, unabhängig von den Header-Werten. Sie können nicht auf der Grundlage der Header-Werte selektiv die Gültigkeit einiger Versionen aufheben und anderer Versionen nicht. (Wenn Sie so konfigurieren CloudFront , dass alle Header an Ihren Ursprung weitergeleitet werden, werden Ihre Dateien CloudFront nicht zwischengespeichert.) Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern.

Weiterleiten von Abfragezeichenfolgen

Wenn Sie so konfiguriert haben CloudFront , dass Abfragezeichenfolgen an Ihren Ursprung weitergeleitet werden, müssen Sie die Abfragezeichenfolgen bei der Invalidierung von Dateien einbeziehen, wie in den folgenden Beispielen gezeigt:

- `/images/image.jpg?parameter1=a`
- `/images/image.jpg?parameter1=b`

Wenn Client-Anfragen fünf verschiedene Abfragezeichenfolgen für dieselbe Datei enthalten, können Sie entweder die Gültigkeit der Datei fünfmal aufheben, einmal für jede Abfragezeichenfolgen, oder den Platzhalter * im Aufhebungspfad verwenden wie im folgenden Beispiel gezeigt:

```
/images/image.jpg*
```

Weitere Informationen zur Verwendung von Platzhaltern im Aufhebungspfad finden Sie unter Invalidation paths.

Weitere Informationen zu Abfragezeichenfolgen finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern.

Um zu ermitteln, welche Abfragezeichenfolgen verwendet werden, können Sie die CloudFront-Protokollierung aktivieren. Weitere Informationen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).

Maximum erlaubt

Weitere Hinweise zur maximal zulässigen Anzahl von Invalidierungen finden Sie unter. Maximum für gleichzeitige Aufhebungsanfragen

Microsoft Smooth Streaming-Dateien

Sie können Mediendateien im Microsoft Smooth Streaming-Format nicht ungültig machen, wenn Sie Smooth Streaming für das entsprechende Cache-Verhalten aktiviert haben.

Nicht-ASCII- oder unsichere Zeichen im Pfad

Wenn der Pfad Nicht-ASCII-Zeichen oder unsichere Zeichen enthält, wie in RFC 1738, definiert, URL-codieren Sie diese Zeichen. Kodieren Sie keine anderen Zeichen im Pfad mit URL, da sonst die alte Version der aktualisierten Datei CloudFront nicht ungültig wird.

> **Important**
>
> Verwenden Sie das ~ Zeichen nicht in Ihrem Pfad. CloudFront unterstützt dieses Zeichen nicht für Ungültigerklärungen, unabhängig davon, ob es URL-kodiert ist oder nicht.

Aufhebungspfade

Der Pfad ist der relative Pfad in Bezug auf die Verteilung. Um die Datei beispielsweise für ungültig zu erklären, geben Sie an. `https://d111111abcdef8.cloudfront.net/images/image2.jpg` `/images/image2.jpg`

> **Note**
>
> In der CloudFront-Konsole können Sie den vorangestellten Schrägstrich im Pfad wie folgt weglassen: `images/image2.jpg`. Wenn Sie die CloudFront API direkt verwenden, müssen die Invalidierungspfade mit einem führenden Schrägstrich beginnen.

Mit dem Platzhalter * können Sie auch die Gültigkeit mehrerer Dateien gleichzeitig aufheben. Der Platzhalter *, der 0 oder mehr Zeichen ersetzt, muss das letzte Zeichen im Aufhebungspfad sein.

> **Important**
>
> Um Platzhalter (*) bei der Invalidierung zu verwenden, müssen Sie den Platzhalter am Ende des Pfads angeben. Sternchen (*), die an einer anderen Stelle eingefügt werden, werden als wörtliche Zeichenübereinstimmung und nicht als Ungültigerklärung von Platzhaltern behandelt.

Wenn Sie das AWS Command Line Interface (AWS CLI) verwenden, um Dateien für ungültig zu erklären, und Sie einen Pfad angeben, der den * Platzhalter enthält, müssen Sie den Pfad wie in Anführungszeichen () einschließen. " "/*"

Die maximale Länge eines Pfads beträgt 4 000 Zeichen.

Example Beispiel: Invalidierungspfade

- Um alle Dateien in einem Verzeichnis für ungültig zu erklären:

  */directory-path/**

- Um ein Verzeichnis, all seine Unterverzeichnisse und alle Dateien in dem Verzeichnis und den Unterverzeichnissen ungültig zu machen:

  */directory-path**

- So heben Sie die Gültigkeit aller Dateien auf, die denselben Namen, aber verschiedene Dateinamenerweiterungen haben, wie beispielsweise logo.jpg, logo.png und logo.gif:

  */directory-path/file-name.**

- So heben Sie die Gültigkeit aller Dateien in einem Verzeichnis auf, deren Dateinamen mit denselben Zeichen beginnen (beispielsweise alle Dateien für ein Video im HLS-Format), unabhängig von der Dateinamenerweiterung:

  */directory-path/initial-characters-in-file-name**

- Wenn Sie CloudFront den Cache auf der Grundlage von Abfragezeichenfolgenparametern konfigurieren und alle Versionen einer Datei für ungültig erklären möchten, gehen Sie wie folgt vor:

  */directory-path/file-name.file-name-extension**

- So machen Sie alle Dateien in einer Distribution ungültig:

```
/*
```

Weitere Informationen zum AUfheben der Gültigkeit von Dateien, wenn Sie eine Lambda-Funktion zum Ändern der URI verwenden, finden Sie unter Changing the URI Using a Lambda Function.

Wenn es sich bei dem Aufhebungspfad um ein Verzeichnis handelt und wenn Sie über keine Standard-Methode zum Angeben von Verzeichnissen verfügen – mit oder ohne einen abschließenden Schrägstrich (/) –, empfehlen wir, die Gültigkeit des Verzeichnisses sowohl mit als auch ohne abschließenden Schrägstrich aufzuheben, beispielsweise `/images` und `/images/`.

Signed URLs

Wenn Sie signiert verwenden URLs, machen Sie eine Datei ungültig, indem Sie nur den Teil der URL vor dem Fragezeichen (?) angeben.

Dateien ungültig machen

Sie können die CloudFront Konsole verwenden, um eine Ungültigerklärung zu erstellen und auszuführen, eine Liste der Ungültigerklärungen anzuzeigen, die Sie zuvor eingereicht haben, und detaillierte Informationen zu einer einzelnen Ungültigerklärung anzuzeigen. Sie können auch eine vorhandene Aufhebung kopieren, die Liste der Dateipfade bearbeiten und die bearbeiteten Aufhebungen ausführen. Sie können diese Aufhebungen nicht aus der Liste entfernen.

Inhalt

Dateien ungültig machen

Gehen Sie wie folgt vor, um Dateien mithilfe der CloudFront Konsole für ungültig zu erklären.

Console

Um Dateien für ungültig zu erklären (Konsole)

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die Verteilung aus, für die Sie Dateien ungültig machen möchten.
3. Wählen Sie die Registerkarte Invalidations aus.
4. Wählen Sie Invalidierung erstellen aus.
5. Geben Sie für die Dateien, die Sie ungültig machen möchten, einen Aufhebungspfad pro Zeile ein. Weitere Informationen zur Angabe von Aufhebungspfaden finden Sie unter Was Sie wissen müssen, wenn Sie Dateien für ungültig erklären.

   > **Important**
   >
   > Gehen Sie bei der Angabe der Dateipfade vorsichtig vor. Sie können einen Aufhebungsantrag nicht abbrechen, wenn Sie ihn begonnen haben.

6. Wählen Sie Invalidierung erstellen.

CloudFront API

Weitere Informationen zur Ungültigerklärung von Objekten und zur Anzeige von Informationen zu Invalidierungen finden Sie in den folgenden Themen in der CloudFront Amazon-API-Referenz:

- CreateInvalidation
- ListInvalidations
- GetInvalidation

> **Note**
>
> Wenn Sie das AWS Command Line Interface (AWS CLI) verwenden, um Dateien ungültig zu machen, und Sie einen Pfad angeben, der den * Platzhalter enthält, müssen Sie den Pfad in Anführungszeichen (") setzen, wie im folgenden Beispiel:

```
aws cloudfront create-invalidation --distribution-id distribution_ID --paths
 "/*"
```

Eine bestehende Invalidierung kopieren, bearbeiten und erneut ausführen

Sie können eine zuvor erstellte Aufhebung kopieren, die Liste der Aufhebungspfade aktualisieren und die aktualisierte Aufhebung ausführen. Sie können eine bestehende Invalidierung nicht kopieren, die Invalidierungspfade aktualisieren und die aktualisierte Invalidierung dann speichern, ohne sie auszuführen.

> **Important**
>
> Wenn Sie eine noch laufende Invalidierung kopieren, die Liste der Invalidierungspfade aktualisieren und dann die aktualisierte Invalidierung ausführen, CloudFront wird die kopierte Invalidierung weder gestoppt noch gelöscht. Falls im Original und in der Kopie Entwertungspfade auftauchen, versucht es zweimal, die Dateien für ungültig zu erklären. Beide Ungültigerklärungen CloudFront werden dann auf die maximale Anzahl kostenloser Invalidierungen für den Monat angerechnet. Wenn Sie die maximale Anzahl kostenloser Ungültigerklärungen bereits erreicht haben, werden Ihnen beide Ungültigerklärungen jeder Datei in Rechnung gestellt. Weitere Informationen finden Sie unter Maximum für gleichzeitige Aufhebungsanfragen.

So kopieren, bearbeiten und führen Sie eine vorhandene Aufhebung erneut aus

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Konsole unter. CloudFront https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie die Verteilung mit der Aufhebung aus, die Sie kopieren möchten.
3. Wählen Sie die Registerkarte Invalidations aus.
4. Wählen Sie die Verteilung aus, die Sie kopieren möchten.

   Wenn Sie sich nicht sicher sind, welche Ungültigerklärung Sie kopieren möchten, können Sie eine Ungültigerklärung auswählen und Details anzeigen wählen, um detaillierte Informationen zu dieser Ungültigerklärung anzuzeigen.
5. Wählen Sie „In neue Datei kopieren“.

6. Aktualisieren Sie die Liste der Aufhebungspfade, sofern zutreffend.
7. Wählen Sie „Invalidierung erstellen“.

Ungültigerklärungen stornieren

Wenn Sie eine Ungültigungsanforderung an senden CloudFront, CloudFront leitet die Anfrage innerhalb weniger Sekunden an alle Edge-Standorte weiter, und jeder Edge-Standort beginnt sofort mit der Verarbeitung der Invalidierung. Daher gibt es keine Möglichkeit, eine Aufhebung nach der Übermittlung zu stornieren.

Ungültigerklärungen auflisten

Sie können mit der Konsole eine Liste der letzten 100 Invalidierungen anzeigen, die Sie für eine Distribution erstellt und ausgeführt haben. CloudFront Wenn Sie eine Liste mit mehr als 100 Invalidierungen abrufen möchten, verwenden Sie die API-Operation. `ListInvalidations` Weitere Informationen finden Sie ListInvalidationsin der Amazon CloudFront API-Referenz.

So listen Sie Aufhebungen auf

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die Verteilung aus, für die Sie eine Liste der Aufhebungen anzeigen möchten.
3. Wählen Sie die Registerkarte Invalidations aus.

> **Note**
>
> Sie können diese Aufhebungen nicht aus der Liste entfernen.

Informationen zu einer Ungültigerklärung anzeigen

Sie können detaillierte Informationen zu einer Aufhebung anzeigen, einschließlich Verteilungs-ID, Aufhebungs-ID, Status der Aufhebung, Datum und Uhrzeit der Erstellung der Aufhebung sowie eine vollständige Liste der Aufhebungspfade.

So zeigen Sie Informationen zu einer Aufhebung auf

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die Verteilung mit der Aufhebung aus, zu der Sie detaillierte Informationen anzeigen möchten.
3. Wählen Sie die Registerkarte Invalidations aus.
4. Wählen Sie die entsprechende Invalidierungs-ID oder wählen Sie die Ungültigkeits-ID aus und klicken Sie dann auf Details anzeigen.

Maximum für gleichzeitige Aufhebungsanfragen

Wenn Sie die Gültigkeit von Dateien einzeln aufheben, können Aufhebungsanfragen für bis zu 3.000 Dateien pro Verteilung gleichzeitig in Bearbeitung sein. Dabei kann es sich um eine Aufhebungsanfrage für bis zu 3.000 Dateien, bis zu 3.000 Anfragen für eine Datei oder eine beliebige andere Kombination handeln, die 3.000 Dateien nicht überschreitet. Beispielsweise können Sie 30 Aufhebungsanfragen übermitteln, die jeweils die Gültigkeit von 100 Dateien aufheben. Solange alle 30 Aufhebungsanfragen noch in Bearbeitung sind, können Sie keine weiteren Aufhebungsanfragen übermitteln. Wenn Sie das Maximum überschreiten, wird eine Fehlermeldung CloudFront zurückgegeben.

Wenn Sie den Platzhalter * verwenden, können Anfragen für bis zu 15 Aufhebungspfade gleichzeitig in Bearbeitung sein. Es können darüber hinaus Aufhebungsanfragen für bis zu 3.000 einzelne Dateien pro Verteilung gleichzeitig in Bearbeitung sein. Das Maximum für zulässige Aufhebungsanfragen mit Platzhalter ist nicht abhängig vom Maximum für die einzelne Aufhebung der Gültigkeit von Dateien.

Zahlen Sie für die Ungültigerklärung der Datei

Die ersten 1 000 pro Monat übermittelten Aufhebungspfade sind kostenlos. Alle weiteren Aufhebungspfade (über 1 000) in einem Monat sind kostenpflichtig. Ein Aufhebungspfad kann für eine einzelne Datei (z. B. `/images/logo.jpg`) oder für mehrere Dateien (z. B. `/images/*`) gelten. Ein Pfad, der den * Platzhalter enthält, zählt als ein Pfad, auch wenn dadurch Tausende CloudFront von Dateien ungültig werden.

Das Maximum von 1.000 kostenlosen Invalidierungspfaden pro Monat gilt für die Gesamtzahl der Invalidierungspfade in allen Distributionen, die Sie mit einem Pfad erstellen. AWS-Konto Wenn Sie

beispielsweise die verwenden, um drei Verteilungen AWS-Konto `john@example.com` zu erstellen, und Sie für jede Verteilung in einem bestimmten Monat 600 Invalidierungspfade einreichen (also insgesamt 1.800 Invalidierungspfade), AWS wird Ihnen die Differenz zwischen der Gesamtzahl der Invalidierungspfade und dem freien Limit von 1000 in Rechnung gestellt. In diesem Beispiel AWS würden Ihnen 800 Invalidierungspfade in diesem Monat in Rechnung gestellt.

Die Gebühr für das Übermitteln eines Aufhebungspfads ist dieselbe, unabhängig von der Anzahl der Dateien, die Sie ungültig machen: eine einzelne Datei (`/images/logo.jpg`) oder alle Dateien, die mit einer Verteilung verknüpft sind (`/*`). Da Ihnen in Ihrer Ungültigungsanfrage pro Pfad berechnet wird, wird jeder Pfad für Abrechnungszwecke auch dann einzeln gezählt, wenn Sie mehrere Pfade zu einer einzigen Anfrage zusammenfassen.

Weitere Informationen zu den Preisen für Invalidierungen finden Sie unter CloudFront Amazon-Preise. Weitere Informationen zur Aufhebungspfaden finden Sie unter Invalidation paths.

Komprimierte Dateien bereitstellen

Wenn angeforderte Objekte komprimiert werden, können Downloads schneller sein, da die Objekte kleiner sind – in einigen Fällen weniger als ein Viertel der Größe des Originals. Schnellere Downloads können zu einem schnelleren Rendern von Webseiten für Ihre Zuschauer führen, insbesondere für JavaScript und CSS-Dateien. Darüber hinaus basieren die Kosten für die CloudFront Datenübertragung auf der Gesamtmenge der bereitgestellten Daten. Die Bereitstellung komprimierter Objekte kann kostengünstiger sein als die Bereitstellung unkomprimierter Objekte.

Themen

- Konfigurieren Sie CloudFront , um Objekte zu komprimieren
- Wie funktioniert die CloudFront Komprimierung
- Bedingungen für die Komprimierung
- Dateitypen, die von CloudFront komprimiert werden
- ETag-Header-Konvertierung

Konfigurieren Sie CloudFront , um Objekte zu komprimieren

Um das Komprimieren von Objekten CloudFront zu konfigurieren, aktualisieren Sie das Cache-Verhalten, das Sie für die komprimierten Objekte bereitstellen möchten.

So konfigurieren Sie CloudFront die Komprimierung von Objekten (Konsole)

1. Melden Sie sich an der CloudFront-Konsole an.
2. Wählen Sie Ihre Distribution und dann das zu bearbeitende Verhalten aus.
3. Wählen Sie für die Einstellung Objekte automatisch komprimieren die Option Ja aus.
4. Verwenden Sie eine Cache-Richtlinie, um die Cache-Einstellungen festzulegen, und aktivieren Sie sowohl die Gzip - als auch die Brotli-Komprimierungsformate.
5. Setzen Sie die TTL-Einstellungen in der Cache-Richtlinie auf einen Wert größer als Null. Wenn Sie den Mindest-TTL Wert auf Null setzen, CloudFront werden komprimierte Inhalte nicht zwischengespeichert.

> **Hinweise**
>
> - Sie müssen Cache-Richtlinien verwenden, um die Brotli-Komprimierung verwenden zu können. Brotli unterstützt keine älteren Cache-Einstellungen.
> - Um die Komprimierung mithilfe AWS CloudFormationder CloudFrontAPI zu aktivieren, setzen Sie die `EnableAcceptEncodingBrotli` Parameter `CompressEnableAcceptEncodingGzip`, auf. `true`

Informationen zur CloudFront Komprimierung von Objekten finden Sie im folgenden Abschnitt.

Wie funktioniert die CloudFront Komprimierung

1. Ein Viewer fordert ein Objekt an. Der Viewer fügt den HTTP-Header `Accept-Encoding` in die Anforderung ein. Die Header-Werte fügen `gzip`, `br` oder beides ein. Dadurch wird angezeigt, dass der Viewer komprimierte Objekte unterstützt. Wenn der Viewer sowohl Gzip als auch Brotli unterstützt, CloudFront verwendet er Brotli.

   > **Note**
   >
   > Die Webbrowser Chrome und Firefox unterstützen die Brotli-Komprimierung nur, wenn die Anfrage über HTTPS gesendet wird. Sie unterstützen Brotli nicht mit HTTP-Anfragen.

2. Sucht am Edge-Standort im CloudFront Cache nach einer komprimierten Kopie des angeforderten Objekts.

3. Führt je nachdem, ob sich das komprimierte Objekt im Cache befindet oder CloudFront nicht, eine der folgenden Aktionen aus:
   - Wenn sich das komprimierte Objekt bereits im Cache befindet, wird das Objekt an den Viewer CloudFront gesendet und die verbleibenden Schritte übersprungen.
   - Wenn sich das komprimierte Objekt nicht im Cache befindet, CloudFront leitet die Anfrage an den Ursprung weiter.

   > **Note**
   >
   > Wenn sich bereits eine unkomprimierte Kopie des Objekts im Cache befindet, wird sie CloudFront möglicherweise an den Betrachter gesendet, ohne die Anfrage an den Ursprung weiterzuleiten. Dies kann beispielsweise passieren, wenn die Komprimierung CloudFront zuvor übersprungen wurde. In diesem Fall wird das unkomprimierte Objekt CloudFront zwischengespeichert und solange bereitgestellt, bis das Objekt abläuft, entfernt oder ungültig gemacht wird.

4. Wenn der Ursprung ein komprimiertes Objekt zurückgibt (wie durch den `Content-Encoding` Header in der HTTP-Antwort angegeben), CloudFront sendet er das komprimierte Objekt an den Viewer, fügt es dem Cache hinzu und überspringt die verbleibenden Schritte. CloudFront komprimiert das Objekt nicht erneut.
5. Wenn der Ursprung ein unkomprimiertes Objekt CloudFront ohne den `Content-Encoding` Header in der HTTP-Antwort zurückgibt, CloudFront wird dann bestimmt, ob das Objekt komprimiert werden kann. Weitere Informationen finden Sie unter Bedingungen für die Komprimierung.
6. Wenn das Objekt komprimiert werden kann, wird es CloudFront komprimiert, an den Viewer gesendet und anschließend dem Cache hinzugefügt.
7. Wenn es nachfolgende Viewer-Anfragen für dasselbe Objekt CloudFront gibt, wird die erste zwischengespeicherte Version zurückgegeben. Wenn ein Viewer beispielsweise ein bestimmtes zwischengespeichertes Objekt anfordert, das Gzip-Komprimierung verwendet, und der Viewer das Gzip-Format akzeptiert, geben nachfolgende Anfragen an dasselbe Objekt immer die Gzip-Version zurück, auch wenn der Viewer sowohl Brotli als auch Gzip akzeptiert.

Einige benutzerdefinierte Ursprünge können auch Objekte komprimieren. Ihr Origin ist möglicherweise in der Lage, Objekte zu komprimieren, die nicht komprimiert werden. CloudFront Weitere Informationen finden Sie unter Dateitypen, die von CloudFront komprimiert werden.

Bedingungen für die Komprimierung

Die folgende Liste enthält weitere Informationen zu Szenarien, in denen Objekte CloudFront nicht komprimiert werden.

Anforderung verwendet HTTP 1.0

Wenn eine Anfrage HTTP 1.0 CloudFront verwendet, CloudFront wird der `Accept-Encoding` Header entfernt und das Objekt in der Antwort nicht komprimiert.

`Accept-Encoding`-Header der Anforderung

Wenn der `Accept-Encoding` Header in der Viewer-Anfrage fehlt oder wenn er keinen Wert enthält `gzip` oder `br` keinen Wert enthält, CloudFront wird das Objekt in der Antwort nicht komprimiert. Wenn der `Accept-Encoding` Header zusätzliche Werte enthält`deflate`, z. B. CloudFront werden diese entfernt, bevor die Anfrage an den Ursprung weitergeleitet wird.

Wenn für die Komprimierung von Objekten konfiguriert CloudFront ist, nimmt er den `Accept-Encoding` Header automatisch in den Cache-Schlüssel und in Ursprungsanfragen auf.

Der Inhalt wird bereits zwischengespeichert, wenn Sie das Komprimieren von CloudFront Objekten konfigurieren

CloudFront komprimiert Objekte, wenn sie vom Ursprung abgerufen werden. Wenn Sie CloudFront das Komprimieren von Objekten konfigurieren, werden Objekte, die bereits an Kantenpositionen zwischengespeichert sind, CloudFront nicht komprimiert. Wenn ein zwischengespeichertes Objekt an einem Edge-Standort abläuft und eine weitere Anforderung für das Objekt an Ihren Ursprung CloudFront weiterleitet, wird das Objekt außerdem CloudFront nicht komprimiert, wenn Ihr Ursprung den HTTP-Statuscode 304 zurückgibt. Das bedeutet, dass der Edge-Standort bereits über die neueste Version des Objekts verfügt. Wenn Sie Objekte komprimieren CloudFront möchten, die bereits an Kantenpositionen zwischengespeichert sind, müssen Sie diese Objekte für ungültig erklären. Weitere Informationen finden Sie unter Machen Sie Dateien ungültig, um Inhalte zu entfernen.

Ursprung ist bereits für die Kompression von Objekten konfiguriert

Wenn Sie CloudFront die Konfiguration so konfigurieren, dass Objekte komprimiert werden und der Ursprung auch Objekte komprimiert, sollte der Ursprung einen Header enthalten. `Content-Encoding` Dieser Header weist darauf hin CloudFront , dass das Objekt bereits komprimiert ist. Wenn eine Antwort von einem Ursprung den `Content-Encoding` Header enthält, wird das Objekt CloudFront nicht komprimiert, unabhängig vom Wert des Headers. CloudFrontsendet die Antwort an den Betrachter und speichert das Objekt an der Edge-Position im Cache.

Dateitypen, die CloudFront komprimiert werden

Eine vollständige Liste finden Sie hier: Dateitypen, die von CloudFront komprimiert werden.

Größe der Objekte, die CloudFront komprimiert werden

CloudFront komprimiert Objekte mit einer Größe zwischen 1.000 Byte und 10.000.000 Byte.

`Content-Length`-Header

Der Ursprung muss in der Antwort einen `Content-Length` Header enthalten, CloudFront anhand dessen bestimmt wird, ob die Größe des Objekts in dem Bereich liegt, der komprimiert wird. CloudFront Wenn der `Content-Length` Header fehlt, einen ungültigen Wert enthält oder einen Wert außerhalb des Größenbereichs für die CloudFront Komprimierung enthält, wird CloudFront das Objekt nicht komprimiert. Weitere Informationen zur Verarbeitung großer Objekte, die den Größenbereich überschreiten können, finden Sie unterWie CloudFront werden Teilanfragen für ein Objekt (BereichGETs) verarbeitet. CloudFront

Den HTTP-Statuscode der Antwort.

CloudFront komprimiert Objekte nur, wenn der HTTP-Statuscode der Antwort `200`, `403`, oder `404` lautet.

Antwort hat keinen Körper

Wenn die HTTP-Antwort vom Ursprung keinen Hauptteil hat, gibt es nichts, was komprimiert werden CloudFront könnte.

`ETag`-Header

CloudFront ändert manchmal den `ETag` Header in der HTTP-Antwort, wenn Objekte komprimiert werden. Weitere Informationen finden Sie unter the section called “`ETag`-Header-Konvertierung”.

CloudFront überspringt die Komprimierung

CloudFront komprimiert Objekte nach bestem Wissen. In seltenen Fällen CloudFront überspringt die Komprimierung eines Objekts bei hoher CloudFront Verkehrsbelastung. CloudFront trifft diese Entscheidung auf der Grundlage einer Vielzahl von Faktoren, einschließlich der Hostkapazität. Wenn die Komprimierung für ein Objekt CloudFront übersprungen wird, wird das unkomprimierte Objekt zwischengespeichert und Benutzern weiterhin zur Verfügung gestellt, bis das Objekt abläuft, entfernt oder ungültig gemacht wird.

Dateitypen, die von CloudFront komprimiert werden

Wenn Sie das Komprimieren von Objekten konfigurieren CloudFront , werden CloudFront nur Objekte komprimiert, die einen der folgenden Werte im Antwort-Header haben: `Content-Type`

- `application/dash+xml`
- `application/eot`
- `application/font`
- `application/font-sfnt`
- `application/javascript`
- `application/json`
- `application/opentype`
- `application/otf`
- `application/pdf`
- `application/pkcs7-mime`
- `application/protobuf`
- `application/rss+xml`
- `application/truetype`
- `application/ttf`
- `application/vnd.apple.mpegurl`
- `application/vnd.mapbox-vector-tile`
- `application/vnd.ms-fontobject`
- `application/wasm`
- `application/xhtml+xml`
- `application/xml`
- `application/x-font-opentype`
- `application/x-font-truetype`
- `application/x-font-ttf`
- `application/x-httpd-cgi`
- `application/x-javascript`
- `application/x-mpegurl`

- application/x-opentype
- application/x-otf
- application/x-perl
- application/x-ttf
- font/eot
- font/opentype
- font/otf
- font/ttf
- image/svg+xml
- text/css
- text/csv
- text/html
- text/javascript
- text/js
- text/plain
- text/richtext
- text/tab-separated-values
- text/xml
- text/x-component
- text/x-java-source
- text/x-script
- vnd.apple.mpegurl

ETag-Header-Konvertierung

Wenn das unkomprimierte Objekt aus dem Ursprung einen gültigen, starken ETag HTTP-Header enthält und das Objekt CloudFront komprimiert, konvertiert es CloudFront auch den starken ETag Header-Wert in einen schwachen ETag und gibt den schwachen ETag Wert an den Viewer zurück. Viewer können den schwachen ETag-Wert speichern und ihn verwenden, um bedingte Anforderungen mit dem HTTP-Header If-None-Match zu senden. Auf diese Weise können Betrachter und der Ursprung die komprimierten und unkomprimierten Versionen eines Objekts als

semantisch gleichwertig behandeln, wodurch unnötige Datenübertragungen vermieden werden. CloudFront

Ein gültiger, starker `ETag` Header-Wert beginnt und endet mit einem doppelten Anführungszeichen ()''. Um den starken `ETag`-Wert in einen schwachen Wert zu konvertieren, fügt CloudFront die Zeichen `W/` am Anfang des starken `ETag`-Werts hinzu.

Wenn das Objekt aus dem Ursprung einen schwachen `ETag` Header-Wert (einen Wert, der mit den Zeichen beginnt`W/`) enthält, CloudFront ändert dieser Wert nicht und gibt ihn so an den Betrachter zurück, wie er vom Ursprung empfangen wurde.

Wenn das Objekt aus dem Ursprung einen ungültigen `ETag` Header-Wert enthält (der Wert beginnt nicht mit '' oder mit`W/`), CloudFront wird der `ETag` Header entfernt und das Objekt ohne den `ETag` Antwort-Header an den Viewer zurückgegeben.

Weitere Informationen finden Sie auf den folgenden Seiten in den MDN-Webdokumenten:

- Direktiven (`ETag`-HTTP-Header)
- Schwache Validierung (bedingte HTTP-Anforderungen)
- If-None-Match-HTTP-Header

AWS WAF Schutzmaßnahmen verwenden

Sie können AWS WAFes zum Schutz Ihrer CloudFront Distributionen und Ursprungsserver verwenden. AWS WAF ist eine Firewall für Webanwendungen, die zum Schutz Ihrer Webanwendungen beiträgt und Anfragen APIs blockiert, bevor sie Ihre Server erreichen. Weitere Informationen finden Sie unter Beschleunigen und schützen Sie Ihre Websites mithilfe von CloudFront und AWS WAF und unter Richtlinien für die Implementierung AWS WAF.

Um den AWS WAF Schutz zu aktivieren, können Sie:

- Verwenden Sie den Ein-Klick-Schutz in der CloudFront Konsole. Der Ein-Klick-Schutz erstellt eine AWS WAF Web-Zugriffskontrollliste (Web ACL), konfiguriert Regeln zum Schutz Ihrer Server vor gängigen Internet-Bedrohungen und fügt die Web-ACL für Sie der Distribution hinzu. CloudFront Bei den Themen in diesem Abschnitt wird von der Verwendung der Ein-Klick-Sicherheitsvorkehrungen ausgegangen.
- Verwenden Sie eine vorkonfigurierte Web-ACL (Access Control List), die Sie in der AWS WAF Konsole erstellen, oder verwenden Sie die. AWS WAF APIs Weitere Informationen finden Sie unter Web Access Control Lists (ACLs) im AWS WAF Developer Guide und AssociateWebACL in der AWS WAF API-Referenz

Sie können es aktivieren AWS WAF , wenn Sie:

- Eine Verteilung erstellen
- Verwenden Sie das Sicherheits-Dashboard, um die Sicherheitseinstellungen einer vorhandenen Distribution zu bearbeiten.

Wenn Sie den Ein-Klick-Schutz verwenden, AWS wird eine Reihe von empfohlenen Schutzmaßnahmen CloudFront angewendet, die:

- Blockieren von IP-Adressen auf der Grundlage interner Bedrohungsinformationen von Amazon zu potenziellen Bedrohungen.
- Schutz vor den häufigsten Sicherheitslücken in Webanwendungen, wie in den OWASP Top 10 beschrieben.
- Schutz vor böswilligen Akteuren, die Anwendungsschwachstellen entdecken.

> **Important**
>
> Sie müssen AWS WAF diese Option aktivieren, wenn Sie Sicherheitsmetriken im Sicherheits-Dashboard anzeigen möchten. CloudFront Ist diese Option nicht aktiviert AWS WAF, können Sie das Sicherheits-Dashboard nur verwenden, um CloudFront geografische Einschränkungen zu aktivieren AWS WAF oder zu konfigurieren. Weitere Informationen zum Dashboard finden Sie unter AWS WAF Sicherheitsvorkehrungen im CloudFront Sicherheits-Dashboard verwalten in diesem Abschnitt.

Themen

AWS WAF Für Distributionen aktivieren

Sie können die Option aktivieren, AWS WAF wenn Sie eine Distribution erstellen, oder Sie können Sicherheitsvorkehrungen für eine bestehende Zugriffskontrollliste (ACL) aktivieren.

Wenn Sie AWS WAF die Option für Ihre CloudFront Distribution aktivieren, können Sie auch die Bot-Steuerung aktivieren und den Sicherheitsschutz nach Bot-Kategorien konfigurieren.

Themen

AWS WAF Für eine neue Distribution aktivieren

Das folgende Verfahren zeigt Ihnen, wie Sie die Option aktivieren AWS WAF , wenn Sie eine neue CloudFront Distribution erstellen.

Zur Aktivierung AWS WAF für eine neue Distribution

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Distributionen und dann Verteilung erstellen aus.
3. Folgen Sie bei Bedarf den Schritten unter Eine Verteilung erstellen.
4. Wählen Sie im Abschnitt Web Application Firewall die Option Bearbeiten und anschließend Sicherheitsvorkehrungen aktivieren aus.
5. Füllen Sie die folgenden Felder aus:

   - Überwachungsmodus verwenden — Sie aktivieren den Überwachungsmodus, wenn Sie zuerst Daten sammeln möchten, um zu testen, wie der Schutz funktioniert. Wenn der Überwachungsmodus aktiviert ist, werden Anfragen nicht blockiert, wenn der Schutz aktiv ist. Stattdessen sammelt der Überwachungsmodus Daten über Anforderungen, die bei aktiviertem Schutz blockiert werden würden. Wenn Sie bereit sind, mit dem Blockieren zu beginnen, können Sie das Blockieren auf der Seite Sicherheit aktivieren.
   - Zusätzliche Schutzmaßnahmen — Wählen Sie alle Optionen aus, die Sie aktivieren möchten. Wenn Sie die Ratenbegrenzung aktivieren, finden Sie weitere Informationen unter the section called “Festlegen der Ratenbegrenzung”.
   - Preisschätzung — Sie können den Abschnitt öffnen, um ein Feld anzuzeigen, in das Sie eine andere Anzahl von Anfragen pro Monat eingeben und eine neue Schätzung sehen können.
6. Prüfen Sie die übrigen Verteilungseinstellungen und wählen Sie dann Verteilung erstellen.

Nachdem Sie eine Verteilung erstellt haben, CloudFront wird ein Sicherheits-Dashboard erstellt. Sie können dieses Dashboard zum Deaktivieren oder Aktivieren verwenden AWS WAF. Wenn Sie es AWS WAF noch nicht aktiviert haben, bleiben die Diagramme und Grafiken im Dashboard leer.

Verwenden einer vorhandenen Web-ACL

Wenn Sie bereits über eine Web-ACL verfügen, können Sie diese anstelle des Schutzes von verwenden AWS WAF.

Um eine bestehende AWS WAF Konfiguration zu verwenden

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Führen Sie eine der folgenden Aktionen aus:

a. Wählen Sie Distribution erstellen aus und folgen Sie den Schritten unter Eine Verteilung erstellen. Kehren Sie dann zu diesem Thema zurück.

b. Wählen Sie eine bestehende Konfiguration und dann die Registerkarte Sicherheit aus.

3. Wählen Sie im Abschnitt Web Application Firewall (WAF) die Optionen Bearbeiten und dann Sicherheitsvorkehrungen aktivieren aus.
4. Wählen Sie Bestehende WAF-Konfiguration verwenden. Diese Option wird nur angezeigt, wenn Sie eine ACLs Webkonfiguration vorgenommen haben.
5. Wählen Sie Ihre vorhandene Web-ACL aus der Tabelle Wählen Sie eine Web-ACL aus.
6. Überprüfen Sie die übrigen Verteilungseinstellungen und wählen Sie dann Verteilung erstellen aus.

Aktivieren Sie die Bot-Steuerung

Wenn Sie AWS WAF die Option für Ihre CloudFront Distribution aktivieren, können Sie Bot-Anfragen für einen bestimmten Zeitraum im Sicherheits-Dashboard in der CloudFront Konsole einsehen. Sie können hier auch die Bot-Steuerung aktivieren oder deaktivieren.

Wenn Sie die Bot-Steuerung aktivieren, fallen Gebühren an. Das Sicherheits-Dashboard bietet einen Kostenvoranschlag.

Wenn Sie die Bot-Steuerung aktivieren, zeigt das Sicherheits-Dashboard den Bot-Verkehr nach Bot-Typ und Kategorie an. Wenn Sie die Bot-Steuerung deaktivieren, wird der Bot-Verkehr auf der Grundlage von Stichproben von Anfragen angezeigt.

So aktivieren Sie Bot Control:

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich die Option Distributionen und dann die Distribution, die Sie ändern möchten.
3. Wählen Sie die Registerkarte Sicherheit aus.
4. Scrollen Sie nach unten zum Abschnitt Bot-Anfragen für einen bestimmten Zeitraum und wählen Sie Bot Control aktivieren aus.
5. Aktivieren Sie im Dialogfeld „Bot-Kontrolle“ unter Konfiguration das Kontrollkästchen Bot-Kontrolle für allgemeine Bots aktivieren.
6. Wählen Sie Änderungen speichern.

Konfigurieren Sie den Schutz nach Bot-Kategorie

Wenn Sie die Bot-Steuerung aktivieren, können Sie konfigurieren, wie jeder nicht verifizierte Bot pro Bot-Kategorie behandelt wird. Sie können beispielsweise einen HTTP-Bibliotheks-Bot in den Überwachungsmodus versetzen und einem Link-Checker eine Challenge zuweisen.

> **Note**
>
> Bots, von denen bekannt ist, dass AWS sie häufig und überprüfbar sind, wie z. B. bekannte Suchmaschinen-Crawler, unterliegen nicht den hier festgelegten Aktionen. Die Bot Control bestätigt, dass validierte Bots aus der Quelle stammen, aus der sie zu stammen vorgeben, bevor sie als verifiziert markiert werden.

Um den Schutz für eine Bot-Kategorie zu konfigurieren

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich die Option Distributionen und dann die Distribution, die Sie ändern möchten.
3. Wählen Sie die Registerkarte Sicherheit aus.
4. Zeigen Sie im Diagramm Anfragen nach Bot-Kategorie auf eines der Elemente in der Spalte Unbestätigte Bot-Aktion und wählen Sie das Stiftsymbol, um es zu bearbeiten.
5. Öffnen Sie die Ergebnisliste und wählen Sie eine der nachstehenden Optionen aus:
   - Blockieren
   - Sobald Sie die Details auf dieser Seite überprüft haben, klicken Sie auf
   - Überwachungsmodus
   - CAPTCHA
   - Challenge
6. Klicken Sie auf das Häkchen neben der Liste, um Ihre Änderung zu speichern.

AWS WAF Sicherheitsvorkehrungen im CloudFront Sicherheits-Dashboard verwalten

CloudFront erstellt ein Sicherheits-Dashboard für jede Ihrer Distributionen. Sie verwenden die Dashboards in der CloudFront Konsole. Mit den Dashboards können Sie CloudFront und AWS WAF zusammen an einem einzigen Ort verwenden, um allgemeine Sicherheitsvorkehrungen für Ihre Webanwendungen zu überwachen und zu verwalten. Die Dashboards enthalten die folgenden Aufgaben und Daten:

- Sicherheitskonfiguration — Sie können Schutzmaßnahmen aktivieren und deaktivieren und alle anwendungsspezifischen AWS WAF Schutzmaßnahmen wie Schutzmaßnahmen einsehen. WordPress
- Sicherheitstrends — Dazu gehören zulässige und blockierte Anfragen, Challenge- und CAPTCHA-Anfragen sowie die häufigsten Angriffsarten. Sie können die Traffic-Zahlen und ihre Veränderung im Laufe der Zeit sehen. Wenn beispielsweise alle Anfragen um 3 % zunehmen, die zulässigen Anfragen jedoch um 14 % steigen, bedeutet das, dass Sie im aktuellen Zeitraum einen größeren Teil Ihres Traffics zugelassen haben.
- Bot-Anfragen — Sie können sehen, wie viel Traffic von Bots stammt, welche Arten von Bots (verifiziert oder nicht verifiziert) sind und wie sich die prozentualen Zuweisungen von Bot-Typen (verifiziert gegenüber nicht verifiziert) im Laufe der Zeit ändern. Weitere Informationen zur Aktivierung der Bot-Steuerung finden Sie unter. Aktivieren Sie die Bot-Steuerung
- Anforderungsprotokolle — Protokolldaten können bei der Beantwortung von Fragen zu Sicherheitstrends oder Bot-Anfragen helfen. Sie können Ihre Protokolle durchsuchen, ohne Abfragen schreiben zu müssen, und anhand von Aggregationsdiagrammen feststellen, ob ein gefilterter Satz von Protokollen hauptsächlich auf einer Teilmenge von HTTP-Methoden, IP-Adressen, URI-Pfaden oder Ländern basiert. Sie können den Mauszeiger über Werte in den Diagrammen bewegen und IP-Adressen und Länder blockieren. Weitere Informationen finden Sie unter Aktivieren Sie AWS WAF Protokolle.
- Verwaltung geografischer Beschränkungen — CloudFront und AWS WAF Bereitstellung von Funktionen für geografische Beschränkungen. CloudFront bietet geografische Einschränkungen kostenlos, aber Metriken für CloudFront geografische Einschränkungen werden nicht im Sicherheits-Dashboard angezeigt. Um die Anforderungsmetriken für blockierte Länderanfragen zu sehen, müssen Sie AWS WAF geografische Einschränkungen verwenden. Bewegen Sie dazu den Mauszeiger über eine Länderleiste im Sicherheits-Dashboard und blockieren Sie das Land. Weitere Informationen finden Sie unter Verwenden Sie CloudFront geografische Einschränkungen.

- Die Option Blockieren ist möglicherweise nicht verfügbar, wenn Sie zuvor außerhalb der CloudFront Konsole eine benutzerdefinierte AWS WAF Regel zum Sperren von Ländern erstellt haben.

Themen

Voraussetzungen

Sie müssen AWS WAF diese Option aktivieren, wenn Sie Sicherheitsmetriken im CloudFront Sicherheits-Dashboard anzeigen möchten. Wenn Sie die Option nicht aktivieren AWS WAF, können Sie das Sicherheits-Dashboard nur verwenden, um CloudFront geografische Einschränkungen zu aktivieren AWS WAF oder zu konfigurieren.

Weitere Informationen zur Aktivierung finden AWS WAF Sie unterAWS WAF Für Distributionen aktivieren.

Aktivieren Sie AWS WAF Protokolle

AWS WAF Mithilfe von Protokolldaten können Sie bestimmte Verkehrsmuster isolieren. Protokolle können Ihnen beispielsweise zeigen, woher bestimmter Traffic stammt oder was er bewirkt.

Wenn Sie die AWS WAF Protokollierung aktivieren CloudWatch, fragt das CloudFront Sicherheits-Dashboard die Erkenntnisse aus den CloudWatch Protokollen ab, aggregiert sie und zeigt sie an. Wir erheben keine Gebühren für die Nutzung des Sicherheits-Dashboards, aber die CloudWatch Preise gelten für Protokolle, die über das Dashboard abgefragt werden. Weitere Informationen finden Sie unter CloudWatch Amazon-Preise.

So aktivieren Sie die Protokollierung:

1. Geben Sie Ihr erwartetes Anforderungsvolumen in das Feld Anzahl der Anforderungen/Monat ein, um die Kosten für die Aktivierung von Protokollen abzuschätzen.
2. Aktivieren Sie das Kontrollkästchen AWS WAF Protokolle aktivieren.
3. Wählen Sie Enable (Aktivieren) aus.

CloudFront erstellt eine CloudWatch Protokollgruppe und aktualisiert Ihre AWS WAF Konfiguration, um mit der Protokollierung zu beginnen CloudWatch. Bei der erstmaligen Verwendung kann es einige Minuten dauern, bis die Protokolldaten angezeigt werden. Im Abschnitt Anfragen der Diagrammliste sind alle Anfragen aufgeführt. In den Balkendiagrammen unter den einzelnen Anfragen werden die Daten nach HTTP-Methode, den wichtigsten URI-Pfaden, den wichtigsten IP-Adressen und den wichtigsten Ländern zusammengefasst. Die Diagramme können Ihnen dabei helfen, Muster zu finden. Beispielsweise können Sie eine unverhältnismäßig hohe Anzahl an Anfragen von einer einzigen IP-Adresse oder von Daten aus einem Land sehen, die Sie zuvor nicht in Ihren Protokollen gesehen hatten. Sie können Anfragen nach Land, Host-Header und anderen Attributen filtern, um unerwünschten Datenverkehr zu finden. Sobald Sie diesen Traffic identifiziert haben, bewegen Sie den Mauszeiger über eine einzelne Anfrage oder ein Diagrammelement und blockieren eine IP-Adresse oder ein Land.

> **Note**
>
> Die angezeigten Messwerte basieren auf der Web-ACL. Wenn Sie also dieselbe Web-ACL mehreren Verteilungen zuordnen, werden Ihnen alle Metriken für Ihre Web-ACL angezeigt, nicht nur die AWS WAF Anfragen, die für diese Verteilung verarbeitet werden.

Festlegen der Ratenbegrenzung

Die Ratenbegrenzung ist eine der Empfehlungen, die Sie möglicherweise bei der Konfiguration von Sicherheitsschutzvorrichtungen erhalten.

CloudFront aktiviert im Monitormodus immer die Ratenbegrenzung. Wenn der Überwachungsmodus aktiviert ist, werden Messwerte CloudFront erfasst, anhand derer Sie feststellen können, ob die Rate, die Sie im Feld Ratenbegrenzung konfiguriert haben, überschritten wurde, wie oft und um wie viel.

Nachdem Sie die Verteilung gespeichert haben, CloudFront beginnt die Erfassung von Daten auf der Grundlage der Zahl im Feld Ratenbegrenzung.

Sie können die Einstellungen für die Ratenbegrenzung im Abschnitt Sicherheit — Web Application Firewall (WAF) auf der Registerkarte Sicherheit jeder CloudFront Distribution aktivieren oder verwalten.

> **Note**
>
> Die Option zur Ratenbegrenzung wird in der CloudFront Konsole nur angezeigt, wenn Sie für Ihre Distribution einen benutzerdefinierten Ursprung angegeben haben, der nicht auf S3 basiert. Andernfalls werden Ihnen nur die für die Distribution aktivierten Core-Schutzfunktionen angezeigt. Weitere Informationen zu den Herkunftstypen finden Sie unterVerwenden Sie bei Verteilungen verschiedene Ursprünge CloudFront.

So richten Sie die Ratenbegrenzung ein

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Verteilungen und dann die Verteilung aus, die Sie ändern möchten.
3. Wählen Sie die Registerkarte Sicherheit aus.
4. Wählen Sie im Abschnitt Sicherheit — Web Application Firewall (WAF) die Option Bearbeiten aus.
5. Wählen Sie unter Zusätzliche Schutzmaßnahmen die Option Ratenbegrenzung aus. Sie können das Ratenlimit optional ändern. Wenn Sie den Tarif verfeinert haben, wählen Sie Änderungen speichern.
6. Im Abschnitt Sicherheit — Web Application Firewall (WAF) neben Ratenbegrenzung können Sie den Überwachungsmodus und dann Blockierung aktivieren auswählen, um den Überwachungsmodus zu deaktivieren. CloudFront beginnt, Anfragen zu blockieren, die das angegebene Ratenlimit überschreiten.

Weitere Informationen zur Aktivierung AWS WAF und Ratenbegrenzung finden Sie im Blogbeitrag Introducing CloudFront Security Dashboard, ein einheitliches CDN und Security Experience.

Deaktivieren Sie die AWS WAF Sicherheitsvorkehrungen

Wenn Ihre Distribution keinen AWS WAF Sicherheitsschutz benötigt, können Sie diese Funktion mithilfe der CloudFront Konsole deaktivieren.

Wenn Sie zuvor den AWS WAF Schutz aktiviert und keine bestehende WAF-Konfiguration (auch bekannt als Ein-Klick-Schutz) ausgewählt haben, wurde CloudFront automatisch eine Web-ACL

für Sie erstellt. Bei auf diese Weise ACLs erstellten Websites trennt die CloudFront Konsole die Zuordnung zur Ressource und löscht die Web-ACL.

Das Trennen einer Web-ACL unterscheidet sich vom Löschen. Durch das Trennen der Zuordnung wird die Web-ACL aus Ihrer Distribution entfernt, sie wird jedoch nicht aus Ihrer entfernt. AWS-Konto Weitere Informationen finden Sie unter Zuordnen oder Aufheben der Zuordnung einer Web-ACL zu einer AWS Ressource im AWS WAF, AWS Firewall Manager und Developer Guide. AWS Shield Advanced

Gehen Sie wie folgt vor, um AWS WAF Schutzmaßnahmen zu deaktivieren und die Web-ACL von Ihrer Distribution zu trennen.

Um den AWS WAF Sicherheitsschutz zu deaktivieren in CloudFront

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie im Navigationsbereich Verteilungen und dann die Verteilung aus, die Sie ändern möchten.
3. Wählen Sie die Registerkarte Sicherheit und dann Bearbeiten aus.
4. Wählen Sie im Abschnitt Web Application Firewall (WAF) die Option AWS WAF Schutz deaktivieren aus.
5. Wählen Sie Änderungen speichern.

Hinweise

- Wenn Sie den AWS WAF Sicherheitsschutz deaktiviert haben und die Web-ACL trotzdem aus Ihrem löschen möchten AWS-Konto, können Sie sie manuell löschen. Gehen Sie wie folgt vor, um eine Web-ACL zu löschen. In der AWS WAF & Shield-Konsole müssen Sie für die ACLsWebseite die Liste Global (CloudFront) auswählen, um das Web zu finden ACLs.
- Wenn Sie eine Distribution aus der CloudFront Konsole löschen, CloudFront wird versucht, auch die Web-ACL zu löschen, falls Sie den Ein-Klick-Schutz gewählt haben. Dies ist die beste Lösung und kann nicht immer garantiert werden. Weitere Informationen finden Sie unter Löschen einer -Verteilung.

Konfigurieren Sie den sicheren Zugriff und beschränken Sie den Zugriff auf Inhalte

CloudFront bietet mehrere Optionen zum Sichern von Inhalten, die es bereitstellt. Im Folgenden finden Sie einige Methoden, mit denen Sie CloudFront den Zugriff auf Inhalte sichern und einschränken können:

- Konfigurieren von HTTPS-Verbindungen.
- Verhindern, dass Benutzer an bestimmten geografischen Standorten auf Inhalte zugreifen
- Erfordern Sie, dass Benutzer mithilfe CloudFront signierter URLs oder signierter Cookies auf Inhalte zugreifen
- Einrichten der Verschlüsselung auf Feldebene für bestimmte Inhaltsfelder
- Verwenden Sie diese AWS WAF Option, um den Zugriff auf Ihre Inhalte zu kontrollieren

Sie sollten auch eine DDo S-resistente Architektur für Ihre Infrastruktur und Anwendungen implementieren. Weitere Informationen finden Sie unter AWS Bewährte Methoden für DDo S Resiliency.

Weitere Informationen finden Sie unter:

- Schützen Sie Ihre Inhaltsbereitstellung mit CloudFront
- SIEM auf Amazon Service OpenSearch

Themen

Verwenden Sie HTTPS mit CloudFront

Sie können festlegen CloudFront , dass Zuschauer HTTPS verwenden müssen, sodass Verbindungen bei der CloudFront Kommunikation mit Zuschauern verschlüsselt werden. Sie können auch so konfigurieren CloudFront , dass HTTPS für Ihren Ursprung verwendet wird, sodass Verbindungen bei der CloudFront Kommunikation mit Ihrem Ursprung verschlüsselt werden.

Wenn du so konfigurierst CloudFront , dass HTTPS sowohl für die Kommunikation mit Zuschauern als auch für die Kommunikation mit deinem Absender erforderlich ist, passiert Folgendes, wenn du CloudFront eine Anfrage erhältst:

1. Ein Zuschauer sendet eine HTTPS-Anfrage an CloudFront. Hier gibt es einige SSL/TLS-Verhandlungen zwischen dem Betrachter und. CloudFront Am Ende sendet der Viewer die Anfrage in einem verschlüsselten Format.
2. Wenn der CloudFront Edge-Standort eine zwischengespeicherte Antwort enthält, CloudFront verschlüsselt die Antwort und gibt sie an den Viewer zurück, und der Viewer entschlüsselt sie.
3. Wenn der CloudFront Edge-Standort keine zwischengespeicherte Antwort enthält, CloudFront führt er eine SSL/TLS-Aushandlung mit Ihrem Ursprung durch und leitet die Anfrage nach Abschluss der Verhandlung in einem verschlüsselten Format an Ihren Ursprung weiter.
4. Ihr Absender entschlüsselt die Anfrage, verarbeitet sie (generiert eine Antwort), verschlüsselt die Antwort und gibt die Antwort an zurück. CloudFront
5. CloudFront entschlüsselt die Antwort, verschlüsselt sie erneut und leitet sie an den Betrachter weiter. CloudFrontspeichert die Antwort auch am Edge-Standort im Cache, sodass sie bei der nächsten Anforderung verfügbar ist.
6. Der Viewer entschlüsselt die Antwort.

Der Prozess funktioniert im Grunde auf die gleiche Weise, unabhängig davon, MediaStore ob es sich bei Ihrem Ursprung um einen Amazon S3 S3-Bucket oder um einen benutzerdefinierten Ursprung wie einen HTTP/S-Server handelt.

> **Note**
>
> Um Angriffe vom Typ SSL-Neuaushandlung zu verhindern, werden Neuverhandlungen für Zuschauer- und CloudFront Ursprungsanfragen nicht unterstützt.

In den folgenden Themen finden Sie Informationen dazu, wie Sie HTTPS zwischen Zuschauern und zwischen CloudFront und CloudFront Ihrem Absender verlangen können.

Themen

Erfordert HTTPS für die Kommunikation zwischen Zuschauern und CloudFront

Sie können ein oder mehrere Cache-Verhalten in Ihrer CloudFront Distribution so konfigurieren, dass HTTPS für die Kommunikation zwischen Zuschauern und erforderlich ist CloudFront. Sie können auch ein oder mehrere Cache-Verhalten so konfigurieren, dass sowohl HTTP als auch HTTPS zulässig sind, sodass HTTPS für einige Objekte CloudFront erforderlich ist, für andere jedoch nicht. Die Konfigurationsschritte hängen davon ab, welchen Domainnamen Sie im Objekt verwenden URLs:

- Wenn Sie den Domainnamen verwenden, der Ihrer Distribution CloudFront zugewiesen wurde, z. B. d111111abcdef8.cloudfront.net, ändern Sie die Einstellung der Viewer-Protokollrichtlinie für ein oder mehrere Cache-Verhaltensweisen dahingehend, dass HTTPS-Kommunikation erforderlich ist. In dieser Konfiguration stellt CloudFront das SSL-/TLS-Zertifikat bereit.

  Informationen zum Ändern des Werts der Viewer-Protokollrichtlinie mithilfe der Konsole finden Sie weiter unten in diesem Abschnitt CloudFront .

  Informationen darüber, wie Sie die CloudFront API verwenden können, um den Wert des `ViewerProtocolPolicy` Elements zu ändern, finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.
- Wenn Sie einen eigenen Domänennamen, z. B. beispiel.com, verwenden, müssen Sie mehrere CloudFront-Einstellungen ändern. Zudem benötigen Sie ein SSL-/TLS-Zertifikat, das entweder von AWS Certificate Manager (ACM) bereitgestellt wird oder das Sie erhalten, indem Sie es von einer

Drittanbieter-Zertifizierungsstelle beziehen und in ACM oder den IAM-Zertifikatspeicher importieren. Weitere Informationen finden Sie unter Verwenden Sie alternative Domainnamen und HTTPS.

> **Note**
>
> Wenn Sie sicherstellen möchten, dass die Objekte, von denen die Zuschauer empfangen, verschlüsselt CloudFront waren, CloudFront als sie von Ihrem Ursprung bezogen wurden, verwenden Sie immer HTTPS zwischen CloudFront und Ihrem Ursprung. Wenn Sie kürzlich zwischen CloudFront und Ihrem Ursprung von HTTP zu HTTPS gewechselt haben, empfehlen wir Ihnen, Objekte an CloudFront Edge-Standorten für ungültig zu erklären. CloudFront gibt ein Objekt an einen Viewer zurück, unabhängig davon, ob das vom Betrachter verwendete Protokoll (HTTP oder HTTPS) mit dem Protokoll übereinstimmt, CloudFront mit dem das Objekt abgerufen wurde. Weitere Informationen zum Entfernen oder Ersetzen von Objekten in einer Verteilung finden Sie unter Inhalte hinzufügen, entfernen oder ersetzen, die CloudFront verbreitet werden.

Erfordert HTTPS für Zuschauer

Gehen Sie wie folgt vor, um HTTPS zwischen Zuschauern und CloudFront für ein oder mehrere Cache-Verhaltensweisen vorzuschreiben.

So konfigurieren Sie CloudFront , dass HTTPS zwischen Zuschauern erforderlich ist und CloudFront

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im oberen Bereich der CloudFront Konsole die ID für die Distribution aus, die Sie aktualisieren möchten.
3. Wählen Sie auf der Registerkarte Verhalten das Cache-Verhalten aus, das Sie aktualisieren möchten, und klicken Sie dann auf Bearbeiten.
4. Geben Sie einen der folgenden Werte für die Viewer-Protokollrichtlinie an:

   Redirect HTTP to HTTPS

   Viewer können beide Protokolle verwenden. HTTP `GET` und `HEAD` Anfragen werden automatisch zu HTTPS-Anfragen umgeleitet. CloudFront gibt den HTTP-Statuscode 301

(Dauerhaft verschoben) zusammen mit der neuen HTTPS-URL zurück. Der Betrachter sendet die Anfrage dann erneut, um die CloudFront HTTPS-URL zu verwenden.

> **Important**
>
> Wenn Sie`POST`,, `PUT` `DELETE`, `OPTIONS`, oder `PATCH` über HTTP mit einem HTTP-zu-HTTPS-Cache-Verhalten und einer Anforderungsprotokollversion von HTTP 1.1 oder höher senden, CloudFront leitet die Anfrage an einen HTTPS-Standort mit dem HTTP-Statuscode 307 (Temporäre Umleitung) weiter. Dies gewährleistet, dass die Anforderung erneut unter Verwendung derselben Methode und derselben Nutzdaten an den neuen Speicherort gesendet wird.
> Wenn Sie`POST`,, `PUT` `DELETE`, `OPTIONS`, oder `PATCH` Anfragen über HTTP an das HTTPS-Cache-Verhalten mit einer Anforderungsprotokollversion unter HTTP 1.1 senden, wird der HTTP-Statuscode 403 (Forbidden) CloudFront zurückgegeben.

Wenn ein Viewer eine HTTP-Anforderung veranlasst, die an eine HTTPS-Adresse umgeleitet wird, berechnet CloudFront Gebühren für beide Anforderungen. Bei der HTTP-Anfrage fallen die Gebühren nur für die Anfrage und für die Header an, die an den Betrachter CloudFront zurückgesendet werden. Bezüglich der HTTPS-Anforderung fallen Gebühren für die Anforderung sowie für die vom Ursprung zurückgegebenen Header und das Objekt an, das vom Ursprung zurückgegeben wird.

Nur HTTPS

Viewer können auf Ihre Inhalte nur zugreifen, wenn sie HTTPS verwenden. Wenn ein Betrachter statt einer HTTPS-Anfrage eine HTTP-Anfrage sendet, wird der HTTP-Statuscode 403 (Forbidden) CloudFront zurückgegeben und das Objekt nicht zurückgegeben.

5. Wählen Sie Änderungen speichern aus.
6. Wiederholen Sie die Schritte 3 bis 5 für jedes weitere Cache-Verhalten, für das Sie HTTPS zwischen Zuschauern und benötigen CloudFront.
7. Vergewissern Sie sich, dass folgende Punkte erfüllt sind, bevor Sie die aktualisierte Konfiguration in einer Produktionsumgebung verwenden:

   - Das Pfadmuster eines jeden Cache-Verhaltens wird nur auf die Anforderungen angewendet, für die die Viewer HTTPS verwenden sollen.

- Die Cache-Verhaltensweisen werden in der Reihenfolge aufgeführt, in der Sie sie auswerten CloudFront möchten. Weitere Informationen finden Sie unter Pfadmuster.
- Die Cache-Verhaltensweisen leiten Anforderungen an die richtigen Ursprünge weiter.

Erfordern Sie HTTPS für die Kommunikation zwischen CloudFront und Ihrem benutzerdefinierten Ursprung

Sie können HTTPS für die Kommunikation zwischen CloudFront und Ihrem Ursprung verlangen.

> **Note**
>
> Wenn Ihr Ursprung ein Amazon S3-Bucket ist, der als Website-Endpunkt konfiguriert ist, können Sie die Verwendung von HTTPS mit Ihrem Ursprung nicht konfigurieren CloudFront , da Amazon S3 HTTPS für Website-Endpunkte nicht unterstützt.

Um HTTPS zwischen CloudFront und Ihrem Ursprung zu verlangen, folgen Sie den Verfahren in diesem Thema, um Folgendes zu tun:

1. Ändern Sie in Ihrer Verteilung die Einstellung Origin Protocol Policy (Ursprungsprotokollrichtlinie) für den Ursprung
2. Installieren Sie ein SSL/TLS-Zertifikat auf Ihrem Ursprungsserver (dies ist nicht erforderlich, wenn Sie einen Amazon S3 S3-Ursprung oder bestimmte andere AWS Ursprünge verwenden).

Themen

Erfordern Sie HTTPS für benutzerdefinierte Ursprünge

Das folgende Verfahren erklärt, wie Sie die Verwendung von HTTPS für die Kommunikation mit einem Elastic Load Balancing Load Balancer, einer EC2 Amazon-Instance oder einem anderen benutzerdefinierten Ursprung konfigurieren CloudFront . Informationen zur Verwendung der CloudFront API zur Aktualisierung einer Distribution finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.

Um CloudFront zu konfigurieren, dass HTTPS zwischen CloudFront und Ihrem benutzerdefinierten Ursprung erforderlich ist

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im oberen Bereich der CloudFront Konsole die ID für die Distribution aus, die Sie aktualisieren möchten.
3. Wählen Sie auf der Registerkarte Verhalten den Ursprung aus, den Sie aktualisieren möchten, und klicken Sie dann auf Bearbeiten.
4. Aktualisieren Sie die folgenden Einstellungen:

   Ursprungsprotokollrichtlinien

   Ändern Sie die Einstellung von Origin Protocol Policy für die den betroffenen Ursprung in Ihrer Verteilung:

   - Nur HTTPS — CloudFront verwendet nur HTTPS für die Kommunikation mit Ihrem benutzerdefinierten Ursprung.
   - Match Viewer — CloudFront kommuniziert mit deinem benutzerdefinierten Absender über HTTP oder HTTPS, je nach Protokoll der Zuschaueranfrage. Wenn Sie beispielsweise Match Viewer für Origin Protocol Policy wählen und der Viewer HTTPS verwendet, um ein Objekt anzufordern CloudFront, verwendet er CloudFront auch HTTPS, um die Anfrage an Ihren Ursprung weiterzuleiten.

     Wählen Sie Match Viewer nur dann, wenn Sie Redirect HTTP to HTTPS oder HTTPS Only für Viewer Protocol Policy auswählen.

     CloudFront speichert das Objekt nur einmal, auch wenn Zuschauer Anfragen sowohl mit HTTP- als auch mit HTTPS-Protokollen stellen.

   Origin SSL Protocols

   Wählen Sie die SSL-Protokolle für den ausgewählten Ursprung in Ihrer Verteilung aus. Das SSLv3 Protokoll ist weniger sicher, daher empfehlen wir, dass du es SSLv3 nur dann auswählst, wenn dein Origin es nicht unterstützt TLSv1 oder später. Der TLSv1 Handshake ist sowohl abwärts- als auch vorwärtskompatibel mit SSLv3, TLSv1 .1 und höher jedoch nicht. Wenn Sie dies wünschen SSLv3, werden CloudFront nur Handshake-Anfragen gesendet SSLv3 .
5. Wählen Sie Änderungen speichern aus.

6. Wiederholen Sie die Schritte 3 bis 5 für jeden weiteren Ursprung, für den Sie HTTPS zwischen CloudFront und Ihrem benutzerdefinierten Ursprung benötigen möchten.
7. Vergewissern Sie sich, dass folgende Punkte erfüllt sind, bevor Sie die aktualisierte Konfiguration in einer Produktionsumgebung verwenden:

   - Das Pfadmuster eines jeden Cache-Verhaltens wird nur auf die Anforderungen angewendet, für die die Viewer HTTPS verwenden sollen.
   - Die Cache-Verhaltensweisen werden in der Reihenfolge aufgeführt, in der Sie sie auswerten CloudFront möchten. Weitere Informationen finden Sie unter Pfadmuster.
   - Die Cache-Verhaltensweisen leiten Anforderungen an die Ursprünge weiter, deren Origin Protocol Policy-Einstellung Sie geändert haben.

Installieren Sie ein SSL/TLS-Zertifikat auf Ihrem benutzerdefinierten Ursprung

Sie können ein SSL-/TLS-Zertifikat von den folgenden Quellen auf Ihrem benutzerdefinierten Ursprungs-Server verwenden:

- Wenn Ihr Ursprung ein Elastic Load Balancing-Load Balancer ist, können Sie ein Zertifikat verwenden, das von AWS Certificate Manager (ACM) bereitgestellt wird. Sie können auch ein Zertifikat verwenden, das von einer vertrauenswürdigen Zertifizierungsstelle signiert und in ACM importiert wurde.
- Für andere Quellen als Elastic Load Balancing Load Balancer müssen Sie ein Zertifikat verwenden, das von einer vertrauenswürdigen Zertifizierungsstelle (CA) eines Drittanbieters signiert wurde, z. B. Comodo oder DigiCert Symantec.

Das vom Ursprungs zurückgegebene Zertifikat muss einen der folgenden Domänennamen enthalten:

- Der Domainname im Feld Origin-Domain des Ursprungs (das `DomainName` Feld in der CloudFront API).
- Den Domänennamen im `Host`-Header, wenn das Cache-Verhalten so konfiguriert ist, dass der `Host`-Header zum Ursprung weiterleitet.

Wenn HTTPS für die Kommunikation mit Ihrem Ursprungsserver CloudFront verwendet wird, CloudFront wird überprüft, ob das Zertifikat von einer vertrauenswürdigen Zertifizierungsstelle ausgestellt wurde. CloudFront unterstützt dieselben Zertifizierungsstellen wie Mozilla. Die aktuelle

Liste finden Sie unter Mozilla-Liste der CA-Zertifikate. Sie können kein selbstsigniertes Zertifikat für die HTTPS-Kommunikation zwischen CloudFront und Ihrem Ursprung verwenden.

> **Important**
>
> Wenn der Ursprungsserver ein abgelaufenes Zertifikat, ein ungültiges Zertifikat oder ein selbstsigniertes Zertifikat zurückgibt oder wenn der Ursprungsserver die Zertifikatskette in der falschen Reihenfolge zurückgibt, CloudFront bricht er die TCP-Verbindung ab, gibt den HTTP-Statuscode 502 (Bad Gateway) an den Viewer zurück und setzt den `X-Cache` Header auf. `Error from cloudfront` Außerdem wird die TCP-Verbindung unterbrochen, wenn die gesamte Zertifikatskette, einschließlich des Zwischenzertifikats CloudFront , nicht vorhanden ist.

Erfordern Sie HTTPS für die Kommunikation zwischen CloudFront und Ihrem Amazon S3 S3-Ursprung

Wenn Ihr Ursprung ein Amazon S3 S3-Bucket ist, CloudFront hängen Ihre Optionen für die Verwendung von HTTPS für die Kommunikation mit davon ab, wie Sie den Bucket verwenden. Wenn Ihr Amazon S3-Bucket als Website-Endpunkt konfiguriert ist, können Sie die Verwendung von HTTPS für die Kommunikation mit Ihrem Ursprung nicht konfigurieren CloudFront , da Amazon S3 in dieser Konfiguration keine HTTPS-Verbindungen unterstützt.

Wenn es sich bei Ihrem Ursprung um einen Amazon S3 S3-Bucket handelt, der HTTPS-Kommunikation unterstützt, CloudFront leitet er Anfragen an S3 weiter. Dabei wird das Protokoll verwendet, mit dem die Zuschauer die Anfragen eingereicht haben. Die Standardeinstellung von Protokoll (nur benutzerdefinierte Ursprünge) ist Match Viewer (An Betrachter anpassen) und kann nicht geändert werden. Wenn Sie jedoch Origin Access Control (OAC) für Ihren Amazon S3-Ursprung aktivieren, hängt die zwischen Amazon S3 CloudFront und Amazon S3 verwendete Kommunikation von Ihren Einstellungen ab. Weitere Informationen finden Sie unter Erstellen Sie eine neue Origin-Zugriffskontrolle.

Wenn Sie HTTPS für die Kommunikation zwischen Amazon S3 CloudFront und Amazon S3 benötigen möchten, müssen Sie den Wert der Viewer-Protokollrichtlinie auf „HTTP zu HTTPS umleiten“ oder „Nur HTTPS“ ändern. Das Verfahren weiter unten in diesem Abschnitt erklärt, wie Sie die Viewer-Protokollrichtlinie mithilfe der CloudFront Konsole ändern. Informationen zur Verwendung der CloudFront API zur Aktualisierung des `ViewerProtocolPolicy` Elements für eine Distribution finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.

Wenn Sie HTTPS mit einem Amazon S3-Bucket verwenden, der die HTTPS-Kommunikation unterstützt, stellt Amazon S3 das SSL/TLS-Zertifikat für Sie bereit.

HTTPS für einen Amazon S3 S3-Ursprung erforderlich

Das folgende Verfahren zeigt Ihnen, wie Sie so konfigurieren CloudFront , dass HTTPS für Ihren Amazon S3 S3-Ursprung erforderlich ist.

So konfigurieren Sie CloudFront , dass HTTPS für Ihren Amazon S3 S3-Ursprung erforderlich ist

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im oberen Bereich der CloudFront Konsole die ID für die Distribution aus, die Sie aktualisieren möchten.
3. Wählen Sie auf der Registerkarte Behaviors das zu aktualisierende Cache-Verhalten und anschließend Edit aus.
4. Geben Sie einen der folgenden Werte für Viewer Protocol Policy an:

   Redirect HTTP to HTTPS

   Zuschauer können beide Protokolle verwenden, HTTP-Anfragen werden jedoch automatisch zu HTTPS-Anfragen umgeleitet. CloudFront gibt den HTTP-Statuscode 301 (Dauerhaft verschoben) zusammen mit der neuen HTTPS-URL zurück. Der Betrachter sendet die Anfrage dann erneut, um die CloudFront HTTPS-URL zu verwenden.

   > **Important**
   >
   > CloudFront leitet`DELETE`,, `OPTIONS` `PATCH``POST`, oder `PUT` Anfragen nicht von HTTP zu HTTPS weiter. Wenn Sie ein Cache-Verhalten so konfigurieren, dass es zu HTTPS umleitet, CloudFront reagiert auf HTTP- `DELETE` `OPTIONS``PATCH`,`POST`,, oder `PUT` Anfragen für dieses Cache-Verhalten mit dem HTTP-Statuscode 403 (Verboten).

   Wenn ein Viewer eine HTTP-Anforderung veranlasst, die an eine HTTPS-Adresse umgeleitet wird, berechnet CloudFront Gebühren für beide Anforderungen. Bei der HTTP-Anfrage fallen die Gebühren nur für die Anfrage und für die Header an, die an den Betrachter CloudFront zurückgesendet werden. Bezüglich der HTTPS-Anforderung fallen Gebühren für

die Anforderung sowie für die vom Ursprung zurückgegebenen Header und das Objekt an, das vom Ursprung zurückgegeben wird.

HTTPS Only

Viewer können auf Ihre Inhalte nur zugreifen, wenn sie HTTPS verwenden. Wenn ein Betrachter statt einer HTTPS-Anfrage eine HTTP-Anfrage sendet, wird der HTTP-Statuscode 403 (Forbidden) CloudFront zurückgegeben und das Objekt nicht zurückgegeben.

5. Wählen Sie Yes, Edit aus.
6. Wiederholen Sie die Schritte 3 bis 5 für jedes weitere Cache-Verhalten, für das Sie HTTPS zwischen Zuschauern und CloudFront CloudFront und zwischen S3 benötigen möchten.
7. Vergewissern Sie sich, dass folgende Punkte erfüllt sind, bevor Sie die aktualisierte Konfiguration in einer Produktionsumgebung verwenden:

   - Das Pfadmuster eines jeden Cache-Verhaltens wird nur auf die Anforderungen angewendet, für die die Viewer HTTPS verwenden sollen.
   - Die Cache-Verhaltensweisen werden in der Reihenfolge aufgeführt, in der Sie sie auswerten CloudFront möchten. Weitere Informationen finden Sie unter Pfadmuster.
   - Die Cache-Verhaltensweisen leiten Anforderungen an die richtigen Ursprünge weiter.

Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront

Wenn Sie HTTPS zwischen Zuschauern und Ihrer CloudFront Distribution benötigen, müssen Sie eine Sicherheitsrichtlinie wählen, die die folgenden Einstellungen festlegt:

- Das SSL-/TLS-Protokoll, das mindestens für die Kommunikation mit Zuschauern CloudFront verwendet wird.
- Die Chiffren, mit denen die CloudFront Kommunikation mit Zuschauern verschlüsselt werden kann.

Um eine Sicherheitsrichtlinie auszuwählen, geben Sie den anwendbaren Wert für Sicherheitsrichtlinie (Mindestversion von SSL/TLS) an. In der folgenden Tabelle sind die Protokolle und Chiffren aufgeführt, die für jede Sicherheitsrichtlinie verwendet CloudFront werden können.

Ein Betrachter muss mindestens eine der unterstützten Verschlüsselungen unterstützen, um eine HTTPS-Verbindung herzustellen. CloudFront CloudFront wählt aus den Verschlüsselungen, die der Betrachter unterstützt, eine Chiffre in der aufgelisteten Reihenfolge aus. Siehe auch OpenSSL-, s2n- und RFC-Verschlüsselungsnamen.

	Sicherheitsrichtlinie						
	SSLv3	TLSv1	TLSv1_26	TLSv1.1_016	TLSv1.2_018	TLSv1.2_019	TLSv1.2_2021
Unterstützte SSL/TLS-Protokolle							
TLSv13.	♦	♦	♦	♦	♦	♦	♦
TLSv12.	♦	♦	♦	♦	♦	♦	♦
TLSv11.	♦	♦	♦	♦			
TLSv1	♦	♦	♦				
SSLv3	♦						
Unterstützte 3.3-Chiffren TLSv1							
TLS_AES_128_GCM_SHA256	♦	♦	♦	♦	♦	♦	♦
TLS AES 256 GCM SHA384	♦	♦	♦	♦	♦	♦	♦
TLS_ 0_ 05_ CHACHA2 POLY13 SHA256	♦	♦	♦	♦	♦	♦	♦
Unterstützte ECDSA-Verschlüsselungen							
ECDHE-ECDSA- -GCM-AES128 SHA256	♦	♦	♦	♦	♦	♦	♦
ECDHE-ECDSA- - AES128 SHA256	♦	♦	♦	♦	♦	♦	
ECDHE-ECDSA- -SHA AES128	♦	♦	♦	♦			

	Sicherheitsrichtlinie						
	SSLv3	TLSv1	TLSv1_26	TLSv1.1_016	TLSv1.2_018	TLSv1.2_019	TLSv1.2_2021
ECDHE-ECDSA- -GCM-AES256 SHA384	♦	♦	♦	♦	♦	♦	♦
ECDHE-ECDSA- 0 CHACHA2 - 05 POLY13	♦	♦	♦	♦	♦	♦	♦
ECDHE-ECDSA- AES256 - SHA384	♦	♦	♦	♦	♦	♦	
ECDHE-ECDSA- -SHA AES256	♦	♦	♦	♦			
Unterstützte RSA-Verschlüsselungen							
ECDHE-RSA- -GCM-AES128 SHA256	♦	♦	♦	♦	♦	♦	♦
ECDHE-RSA- AES128 - SHA256	♦	♦	♦	♦	♦	♦	
ECDHE-RSA-SHA AES128	♦	♦	♦	♦			
ECDHE-RSA- AES256 - GCM- SHA384	♦	♦	♦	♦	♦	♦	♦
ECDHE-RSA- CHACHA2 0- 05 POLY13	♦	♦	♦	♦	♦	♦	♦
ECDHE-RSA AES256 - - SHA384	♦	♦	♦	♦	♦	♦	
ECDHE-RSA-SHA AES256	♦	♦	♦	♦			

	Sicherheitsrichtlinie						
	SSLv3	TLSv1	TLSv1_26	TLSv1.1_016	TLSv1.2_018	TLSv1.2_019	TLSv1.2_2021
AES128-GCM- SHA256	♦	♦	♦	♦	♦		
AES256-GCM- SHA384	♦	♦	♦	♦	♦		
AES128-SHA256	♦	♦	♦	♦	♦		
AES256-SCHA	♦	♦	♦	♦			
AES128-SCHA	♦	♦	♦	♦			
DES-SHA CBC3	♦	♦					
RC4-MD5	♦						

OpenSSL-, s2n- und RFC-Verschlüsselungsnamen

OpenSSL und s2n verwenden für Chiffren andere Namen als die TLS-Standards verwenden (RFC 2246, RFC 4346, RFC 5246, und RFC 8446). Die folgende Tabelle ordnet den OpenSSL-Namen und s2n Namen den RFC-Namen für jedes Verschlüsselungsverfahren zu.

CloudFront Unterstützt für Chiffren mit elliptischen Kurvenalgorithmen für den Schlüsselaustausch die folgenden elliptischen Kurven:

- prime256v1
- X25519
- secp384r1

Weitere Informationen zu den Zertifikatsanforderungen für finden Sie unter. CloudFront Anforderungen für die Verwendung von SSL/TLS Zertifikaten mit CloudFront

OpenSSL- und s2n-Chiffrenname	RFC-Verschlüsselungsname
Unterstützte TLSv1 3.3-Chiffren	

OpenSSL- und s2n-Chiffrenname	RFC-Verschlüsselungsname
TLS_AES_128_GCM_ SHA256	TLS AES 128 GCM SHA256
TLS AES 256 GCM SHA384	TLS AES 256 GCM SHA384
TLS_ 0_ 05_ CHACHA2 POLY13 SHA256	TLS_ CHACHA2 0_ POLY13 05_ SHA256
Unterstützte ECDSA-Verschlüsselungen	
ECDHE-ECDSA- -GCM- AES128 SHA256	TLS_ECDHE_ECDSA_MIT_AES_128_GCM_ SHA256
ECDHE-ECDSA- - AES128 SHA256	TLS_ECDHE_ECDSA_MIT_AES_128_CBC_ SHA256
ECDHE-ECDSA-SHA AES128	TLS_ECDHE_ECDSA_WITH_AES_12 8_CBC_SHA
ECDHE-ECDSA- -GCM- AES256 SHA384	TLS_ECDHE_ECDSA_MIT_AES_256_GCM_ SHA384
ECDHE-ECDSA- 0- 05 CHACHA2 POLY13	TLS_ECDHE_ECDSA_MIT_ CHACHA2 POLY13 0_ 05_ SHA256
ECDHE-ECDSA- - AES256 SHA384	TLS_ECDHE_ECDSA_MIT_AES_256_CBC_ SHA384
ECDHE-ECDSA-SHA AES256	TLS_ECDHE_ECDSA_WITH_AES_25 6_CBC_SHA
Unterstützte RSA-Verschlüsselungen	
ECDHE-RSA- -GCM- AES128 SHA256	TLS_ECDHE_RSA_MIT_AES_128_GCM_ SHA256
ECDHE-RSA- - AES128 SHA256	TLS_ECDHE_RSA_MIT_AES_128_CBC_ SHA256

OpenSSL- und s2n-Chiffrenname	RFC-Verschlüsselungsname
ECDHE-RSA-SHA AES128	TLS_ECDHE_RSA_WITH_AES_128_ CBC_SHA
ECDHE-RSA- AES256 -GCM- SHA384	TLS_ECDHE_RSA_MIT_AES_256_GCM_ SHA384
ECDHE-RSA- 0- 05 CHACHA2 POLY13	CHACHA2TLS_ECDHE_RSA_MIT_ POLY13 0_ 05_ SHA256
ECDHE-RSA- - AES256 SHA384	TLS_ECDHE_RSA_WITH_AES_256_CBC_ SHA384
ECDHE-RSA-SHA AES256	TLS_ECDHE_RSA_WITH_AES_256_ CBC_SHA
AES128-GCM- SHA256	TLS_RSA_MIT_AES_128_GCM_ SHA256
AES256-GCM- SHA384	TLS_RSA_MIT_AES_256_GCM_ SHA384
AES128-SHA256	TLS_RSA_MIT_AES_128_CBC_ SHA256
AES256-SCHA	TLS_RSA_WITH_AES_256_CBC_SHA
AES128-SCHA	TLS_RSA_WITH_AES_128_CBC_SHA
DES-SHA CBC3	TLS_RSA_WITH_3DES_EDE_CBC_SHA
RC4-MD5	TLS_RSA_MIT_ _128_ RC4 MD5

Unterstützte Signaturschemas zwischen Zuschauern und CloudFront

CloudFront unterstützt die folgenden Signaturschemas für Verbindungen zwischen Zuschauern undCloudFront.

	Sicherheitsrichtlinie						
Signaturschemata	SSLv3	TLSv1	TLSv1_2 6	TLSv1.1_ 016	TLSv1.2_ 018	TLSv1.2_ 019	und 2.2_2021 TLSv1
TLS_SIGNATURE_SCHE ME_RSA_PSS_PSS_ SHA256	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_RSA_PSS_PSS_ SHA384	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_RSA_PSS_PSS_ SHA512	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_RSA_PSS_RSAE_ SHA256	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_RSA_PSS_RSAE_ SHA384	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_RSA_PSS_RSAE_ SHA512	♦	♦	♦	♦	♦	♦	♦
PKCS1TLS_SIGNATURE _SCHEME_RSA_ _ SHA256	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_RSA_ _ PKCS1 SHA384	♦	♦	♦	♦	♦	♦	♦

	Sicherheitsrichtlinie						
Signaturschemata	SSLv3	TLSv1	TLSv1_2 6	TLSv1.1_ 016	TLSv1.2_ 018	TLSv1.2_ 019	und 2.2_2021 TLSv1
TLS_SIGNATURE_SCHE ME_RSA_ _ PKCS1 SHA512	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_RSA_ _ PKCS1 SHA224	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE MA_ECDSA_ SHA256	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_ECDSA_ SHA384	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_ECDSA_ SHA512	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_ECDSA_ SHA224	♦	♦	♦	♦	♦	♦	♦
SECP256TLS_SIGNATU RE_SCHEME_ECDSA_ R1_ SHA256	♦	♦	♦	♦	♦	♦	♦
SECP384TLS_SIGNATU RE_SCHEME_ECDSA_ R1_ SHA384	♦	♦	♦	♦	♦	♦	♦
TLS_SIGNATURE_SCHE ME_RSA_ PKCS1 SHA1	♦	♦	♦	♦			
TLS_SIGNATURE_SCHE MA_ECDSA_ SHA1	♦	♦	♦	♦			

Unterstützte Protokolle und Chiffren zwischen und dem Ursprung CloudFront

Wenn Sie sich dafür entscheiden, HTTPS zwischen CloudFront und Ihrem Ursprung vorzuschreiben, können Sie entscheiden, welches SSL/TLS-Protokoll für die sichere Verbindung zulässig ist, und Sie CloudFront können mithilfe einer der in der folgenden Tabelle aufgeführten ECDSA- oder RSA-Chiffren eine Verbindung zum Ursprung herstellen. Ihr Ursprung muss mindestens eine dieser Chiffren unterstützen, um eine HTTPS-Verbindung zu Ihrem Ursprung herzustellen. CloudFront

OpenSSL und s2n verwenden für Chiffren andere Namen als die TLS-Standards verwenden (RFC 2246, RFC 4346, RFC 5246, und RFC 8446). Die folgende Tabelle beinhaltet den OpenSSL-Namen und s2n Namen den RFC-Namen für jedes Verschlüsselungsverfahren.

CloudFront Unterstützt für Chiffren mit elliptischen Kurven für den Schlüsselaustausch die folgenden elliptischen Kurven:

- prime256v1
- secp384r1
- X25519

OpenSSL- und s2n-Chiffrenname	RFC-Verschlüsselungsname
Unterstützte ECDSA-Verschlüsselungen	
ECDHE-ECDSA- -GCM- AES256 SHA384	TLS_ECDHE_ECDSA_MIT_AES_256_GCM_SHA384
ECDHE-ECDSA- - AES256 SHA384	TLS_ECDHE_ECDSA_MIT_AES_256_CBC_SHA384
ECDHE-ECDSA-SHA AES256	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
ECDHE-ECDSA- -GCM- AES128 SHA256	TLS_ECDHE_ECDSA_MIT_AES_128_GCM_SHA256
ECDHE-ECDSA- - AES128 SHA256	TLS_ECDHE_ECDSA_MIT_AES_128_CBC_SHA256

OpenSSL- und s2n-Chiffrenname	RFC-Verschlüsselungsname
ECDHE-ECDSA-SHA AES128	TLS_ECDHE_ECDSA_WITH_AES_12 8_CBC_SHA
Unterstützte RSA-Verschlüsselungen	
ECDHE-RSA- -GCM- AES256 SHA384	TLS_ECDHE_RSA_MIT_AES_256_GCM_ SHA384
ECDHE-RSA- - AES256 SHA384	TLS_ECDHE_RSA_WITH_AES_256_CBC_ SHA384
ECDHE-RSA-SHA AES256	TLS_ECDHE_RSA_WITH_AES_256_ CBC_SHA
ECDHE-RSA- AES128 -GCM- SHA256	TLS_ECDHE_RSA_MIT_AES_128_GCM_ SHA256
ECDHE-RSA- - AES128 SHA256	TLS_ECDHE_RSA_MIT_AES_128_CBC_ SHA256
ECDHE-RSA-SHA AES128	TLS_ECDHE_RSA_WITH_AES_128_ CBC_SHA
AES256-SCHA	TLS_RSA_WITH_AES_256_CBC_SHA
AES128-SCHA	TLS_RSA_WITH_AES_128_CBC_SHA
DES-SHA CBC3	TLS_RSA_WITH_3DES_EDE_CBC_SHA
RC4-MD5	TLS_RSA_MIT_ _128_ RC4 MD5

Unterstützte Signaturschemas zwischen und dem Ursprung CloudFront

CloudFront unterstützt die folgenden Signaturschemas für Verbindungen zwischen CloudFront und dem Ursprung.

- TLS_SIGNATURE_SCHEME_RSA_ _ PKCS1 SHA256
- TLS_SIGNATURE_SCHEME_RSA_ _ PKCS1 SHA384

- TLS_SIGNATURE_SCHEME_RSA_ _ PKCS1 SHA512
- TLS_SIGNATURE_SCHEME_RSA_ _ PKCS1 SHA224
- TLS_SIGNATURE_SCHEMA_ECDSA_ SHA256
- TLS_SIGNATURE_SCHEME_ECDSA_ SHA384
- TLS_SIGNATURE_SCHEME_ECDSA_ SHA512
- TLS_SIGNATURE_SCHEME_ECDSA_ SHA224
- PKCS1TLS_SIGNATURE_SCHEME_RSA_ SHA1
- TLS_SIGNATURE_SCHEMA_ECDSA_ SHA1

Verwenden Sie alternative Domainnamen und HTTPS

Wenn Sie Ihren eigenen Domainnamen URLs für Ihre Dateien verwenden möchten (z. B.`https://www.example.com/image.jpg`) und Sie möchten, dass Ihre Zuschauer HTTPS verwenden, müssen Sie die Schritte in den folgenden Themen ausführen. (Wenn Sie beispielsweise den CloudFront Standard-Vertriebsdomänennamen in Ihrem URLs verwenden`https://d111111abcdef8.cloudfront.net/image.jpg`, folgen Sie stattdessen den Anweisungen im folgenden Thema:Erfordert HTTPS für die Kommunikation zwischen Zuschauern und CloudFront.)

> **Important**
>
> Wenn Sie Ihrer Distribution ein Zertifikat hinzufügen, wird das Zertifikat CloudFront sofort an alle Edge-Standorte weitergegeben. Sobald neue Edge-Standorte verfügbar sind, wird das Zertifikat von CloudFront auch an diese Standorte verteilt. Sie können die Edge-Standorte, an die die CloudFront Zertifikate weitergegeben werden, nicht einschränken.

Themen

- Wählen Sie aus, wie CloudFront HTTPS-Anfragen bearbeitet werden
- Anforderungen für die Verwendung von SSL/TLS Zertifikaten mit CloudFront
- Kontingente für die Verwendung von SSL/TLS Zertifikaten mit CloudFront (HTTPS nur zwischen Zuschauern und CloudFront nur)
- Konfigurieren Sie alternative Domainnamen und HTTPS
- Ermitteln Sie die Größe des öffentlichen Schlüssels in einem SSL/TLS RSA-Zertifikat

- Erhöhen Sie die Kontingente für SSL/TLS-Zertifikate
- SSL/TLS Zertifikate rotieren
- Kehren Sie von einem benutzerdefinierten SSL/TLS-Zertifikat zum Standardzertifikat zurück CloudFront
- Wechseln Sie von einem benutzerdefinierten SSL/TLS-Zertifikat mit dedizierten IP-Adressen zu SNI

Wählen Sie aus, wie CloudFront HTTPS-Anfragen bearbeitet werden

Wenn Sie möchten, dass Ihre Zuschauer HTTPS und alternative Domainnamen für Ihre Dateien verwenden, wählen Sie eine der folgenden Optionen für die Bearbeitung von CloudFront HTTPS-Anfragen:

- Verwenden der Servernamensanzeige (Server Name Indication, SNI) – Empfohlen
- Verwenden einer dedizierten IP-Adresse an jedem Edge-Standort

In diesem Abschnitt wird erläutert, wie beide Optionen funktionieren.

Verwenden Sie SNI, um HTTPS-Anfragen zu bearbeiten (funktioniert für die meisten Kunden)

Die Servernamensanzeige (SNI) ist eine Erweiterung des TLS-Protokolls, die in Browsern und Clients unterstützt wird, die nach 2010 veröffentlicht wurden. Wenn Sie so konfigurieren CloudFront , dass HTTPS-Anfragen mithilfe von SNI bedient werden CloudFront , verknüpfen Sie Ihren alternativen Domainnamen mit einer IP-Adresse für jeden Edge-Standort. Sobald ein Viewer Inhalte von Ihnen durch Senden einer HTTPS-Anforderung abruft, leitet DNS die Anforderung an die IP-Adresse des korrekten Edge-Standorts weiter. Die IP-Adresse Ihres Domainnamens wird während der SSL/TLS Handshake-Verhandlung festgelegt. Die IP-Adresse ist nicht für Ihre Distribution reserviert.

Die SSL/TLS Verhandlung findet zu einem frühen Zeitpunkt des Aufbaus einer HTTPS-Verbindung statt. Wenn nicht sofort festgestellt werden CloudFront kann, für welche Domain die Anfrage bestimmt ist, wird die Verbindung unterbrochen. Sobald ein Viewer, der SNI unterstützt, eine HTTPS-Anforderung zwecks Abrufs Ihrer Inhalte sendet, geschieht Folgendes:

1. Der Betrachter ruft den Domainnamen automatisch aus der Anfrage-URL ab und fügt ihn der SNI-Erweiterung der Hello-Nachricht des TLS-Clients hinzu.

2. Wenn der TLS-Client die Hello -Nachricht CloudFront empfängt, verwendet er den Domainnamen in der SNI-Erweiterung, um die passende CloudFront Distribution zu finden, und sendet das zugehörige TLS-Zertifikat zurück.
3. Der Zuschauer und CloudFront führen die SSL/TLS Verhandlung durch.
4. CloudFront gibt den angeforderten Inhalt an den Betrachter zurück.

Eine aktuelle Liste der Browser, die SNI unterstützen, finden Sie im Wikipedia-Eintrag Server Name Indication.

Wenn Sie SNI nutzen möchten, jedoch einige Ihrer Benutzer Browser verwenden, die SNI nicht unterstützen, haben Sie mehrere Möglichkeiten:

- Konfigurieren Sie CloudFront die Konfiguration für die Bearbeitung von HTTPS-Anfragen mithilfe von dedizierten IP-Adressen anstelle von SNI. Weitere Informationen finden Sie unter Verwenden Sie eine dedizierte IP-Adresse, um HTTPS-Anfragen zu bearbeiten (funktioniert für alle Clients).
- Verwenden Sie das CloudFront SSL/TLS-Zertifikat anstelle eines benutzerdefinierten Zertifikats. Dies erfordert, dass Sie den CloudFront Domainnamen für Ihre Distribution in der URLs für Ihre Dateien verwenden, zum Beispiel. `https://d111111abcdef8.cloudfront.net/logo.png`

  Wenn Sie das CloudFront Standardzertifikat verwenden, müssen die Zuschauer das SSL-Protokoll TLSv1 oder eine neuere Version unterstützen. CloudFront wird SSLv3 mit dem CloudFront Standardzertifikat nicht unterstützt.

  Sie müssen auch das verwendete SSL/TLS Zertifikat von einem benutzerdefinierten Zertifikat auf das CloudFront Standardzertifikat ändern: CloudFront

  - Wenn Sie Ihre Verteilung noch nicht zur Verteilung Ihrer Inhalte genutzt haben, können Sie einfach die Konfiguration ändern. Weitere Informationen finden Sie unter Eine Verteilung aktualisieren.
  - Wenn Sie Ihre Inhalte über Ihre Distribution verteilt haben, müssen Sie eine neue CloudFront Distribution erstellen und die Einstellungen URLs für Ihre Dateien ändern, um die Zeit, in der Ihre Inhalte nicht verfügbar sind, zu verringern oder zu vermeiden. Weitere Informationen finden Sie unter Kehren Sie von einem benutzerdefinierten SSL/TLS-Zertifikat zum Standardzertifikat zurück CloudFront .
- Wenn Sie Einfluss darauf haben, welche Browser Ihre Benutzer verwenden, können Sie veranlassen, dass sie ihre Browser auf eine Version aktualisieren, die SNI unterstützt.
- Verwenden Sie HTTP anstelle von HTTPS.

Verwenden Sie eine dedizierte IP-Adresse, um HTTPS-Anfragen zu bearbeiten (funktioniert für alle Clients)

Die Servernamensanzeige (SNI) ist eine Möglichkeit, eine Anforderung mit einer Domäne zu verknüpfen. Eine weitere Möglichkeit ist die Verwendung einer dedizierten IP-Adresse. Wenn Sie Benutzer haben, die nicht zu einem Browser oder Client wechseln können, der nach 2010 veröffentlicht wurde, können Sie eine dedizierte IP-Adresse für die Bedienung von HTTPS-Anforderungen verwenden. Eine aktuelle Liste der Browser, die SNI unterstützen, finden Sie im Wikipedia-Eintrag Server Name Indication.

> **Important**
>
> Wenn Sie so konfigurieren CloudFront , dass HTTPS-Anfragen über dedizierte IP-Adressen bedient werden, fällt eine zusätzliche monatliche Gebühr an. Die Gebühr beginnt, wenn Sie Ihr SSL/TLS Zertifikat einer Distribution zuordnen und die Verteilung aktivieren. Weitere Informationen zur CloudFront Preisgestaltung finden Sie unter CloudFront Amazon-Preise. Zusätzliche, in diesem Zusammenhang interessante Details finden Sie unter Using the Same Certificate for Multiple CloudFront Distributions.

Wenn Sie so konfigurieren CloudFront , dass HTTPS-Anfragen mithilfe von dedizierten IP-Adressen bearbeitet werden CloudFront, ordnen Sie Ihr Zertifikat an jedem CloudFront Edge-Standort einer dedizierten IP-Adresse zu. Sobald ein Viewer eine HTTPS-Anforderung zwecks Abrufs Ihrer Inhalte sendet, geschieht Folgendes:

1. DNS leitet die Anforderung an die IP-Adresse Ihrer Verteilung am jeweils zuständigen Edge-Standort weiter.
2. Wenn eine Client-Anfrage die SNI-Erweiterung in der `ClientHello` Nachricht enthält, wird nach einer Distribution CloudFront gesucht, die diesem SNI zugeordnet ist.
   - Wenn es eine Übereinstimmung gibt, CloudFront beantwortet die Anfrage mit dem SSL/TLS-Zertifikat.
   - Wenn es keine Übereinstimmung gibt, CloudFront wird stattdessen die IP-Adresse verwendet, um Ihre Distribution zu identifizieren und zu bestimmen, welches SSL/TLS-Zertifikat an den Betrachter zurückgegeben werden soll.
3. Der Betrachter und CloudFront führen die SSL/TLS Verhandlung mithilfe Ihres SSL/TLS-Zertifikats durch.
4. CloudFront gibt den angeforderten Inhalt an den Betrachter zurück.

Diese Methode funktioniert für jede HTTPS-Anforderung, unabhängig von dem Browser oder Viewer anderer Art, den der Benutzer verwendet.

> **Note**
>
> Dedicated IPs sind nicht statisch IPs und können sich im Laufe der Zeit ändern. Die IP-Adresse, die für den Edge-Standort zurückgegeben wird, wird dynamisch aus den IP-Adressbereichen der CloudFront Edge-Serverliste zugewiesen.
> Die IP-Adressbereiche für CloudFront Edge-Server können sich ändern. Um über Änderungen der IP-Adresse informiert zu werden, abonnieren Sie AWS Public IP Address Changes über Amazon SNS.

Beantragen Sie die Erlaubnis zur Verwendung von drei oder mehr dedizierten IP-Zertifikaten SSL/TLS

Wenn Sie die Erlaubnis benötigen, drei oder mehr dedizierte SSL-/TLS-IP-Zertifikate dauerhaft zuzuordnen CloudFront, gehen Sie wie folgt vor. Weitere Informationen zu HTTPS-Anforderungen finden Sie unter Wählen Sie aus, wie CloudFront HTTPS-Anfragen bearbeitet werden.

> **Note**
>
> Dieses Verfahren gilt für die Verwendung von drei oder mehr dedizierten IP-Zertifikaten in Ihren CloudFront Distributionen. Der Standardwert lautet 2. Beachten Sie, dass Sie nicht mehr als ein SSL-Zertifikat an eine Verteilung binden können.
> Sie können einer CloudFront Distribution jeweils nur ein einziges SSL/TLS Zertifikat zuordnen. Diese Zahl steht für die Gesamtzahl der dedizierten IP-SSL-Zertifikate, die Sie in all Ihren CloudFront Distributionen verwenden können.

So beantragen Sie eine Berechtigung zur Verwendung von drei oder mehr Zertifikaten mit einer CloudFront-Verteilung:

1. Besuchen Sie das Support Center und erstellen Sie einen Fall.
2. Geben Sie die Anzahl der Zertifikate an, die Sie benötigen, und beschreiben Sie die Umstände in Ihrer Beantragung. Wir werden Ihr Konto so bald wie möglich aktualisieren.
3. Fahren Sie mit dem nächsten Schritt fort.

Anforderungen für die Verwendung von SSL/TLS Zertifikaten mit CloudFront

Die Anforderungen für SSL/TLS Zertifikate werden in diesem Thema beschrieben. Sie gelten, sofern nicht anders vermerkt, für die beiden folgenden Elemente:

- Zertifikate für die Verwendung von HTTPS zwischen Zuschauern und CloudFront
- Zertifikate für die Verwendung von HTTPS zwischen CloudFront und Ihrer Herkunft

Themen

Zertifikataussteller

Wir empfehlen Ihnen, ein öffentliches Zertifikat zu verwenden, das von AWS Certificate Manager (ACM) ausgestellt wurde. Weitere Informationen zu ACM finden Sie im AWS Certificate Manager - Benutzerhandbuch. Wenn Sie ein ACM-Zertifikat verwenden möchten CloudFront, stellen Sie sicher, dass Sie das Zertifikat in der Region USA Ost (Nord-Virginia) anfordern (`us-east-1`).

CloudFront unterstützt dieselben Zertifizierungsstellen (CAs) wie Mozilla. Wenn Sie ACM also nicht verwenden, verwenden Sie ein Zertifikat, das von einer Zertifizierungsstelle ausgestellt wurde, die in der Mozilla Included CA Certificate List aufgeführt ist. Weitere Informationen zum Abrufen und

Installieren eines Zertifikats finden Sie in der Dokumentation zu Ihrer HTTP-Server-Software und der Dokumentation, die von der Zertifizierungsstelle bereitgestellt wird.

AWS-Region für AWS Certificate Manager

Um ein Zertifikat in AWS Certificate Manager (ACM) zu verwenden, das HTTPS zwischen Zuschauern und erfordertCloudFront, stellen Sie sicher, dass Sie das Zertifikat in der Region USA Ost (Nord-Virginia) anfordern (`us-east-1`).

Wenn Sie HTTPS zwischen CloudFront und Ihrem Ursprung benötigen und einen Load Balancer in Elastic Load Balancing als Ursprung verwenden, können Sie das Zertifikat in einem beliebigen AWS-Region Format anfordern oder importieren.

Zertifikatformat

Das Zertifikat muss das Format X.509 PEM aufweisen. Dies ist das Standardformat bei der Verwendung von AWS Certificate Manager.

Zwischenzertifikate

Wenn Sie eine Drittanbieter-Zertifizierungsstelle (CA) verwenden, müssen alle Zwischenzertifikate in der Zertifikatkette aufgelistet sein, die sich in der Datei `.pem` befinden, beginnend mit einem Zwischenzertifikat für die Zertifizierungsstelle, die das Zertifikat für Ihre Domäne signiert hat. In der Regel finden Sie eine Datei auf der Website der Zertifizierungsstelle, in der die Zwischen- und Stammzertifikate in der richtigen Reihenfolge aufgelistet sind.

> **Important**
>
> Fügen Sie Folgendes nicht hinzu: das Stammzertifikat, Zwischenzertifikate, die sich nicht im Vertrauenspfad befinden, und das Public-Key-Zertifikat Ihrer CA.

Ein Beispiel:

```
-----BEGIN CERTIFICATE-----
Intermediate certificate 2
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Intermediate certificate 1
-----END CERTIFICATE-----
```

Schlüsseltyp

CloudFront unterstützt öffentlich-private RSA- und ECDSA-Schlüsselpaare.

CloudFront unterstützt mithilfe von RSA- und ECDSA-Zertifikaten HTTPS-Verbindungen sowohl zu Viewern als auch zu Ursprüngen. Mit AWS Certificate Manager (ACM) können Sie RSA- oder ECDSA-Zertifikate anfordern und importieren und sie dann Ihrer Distribution zuordnen. CloudFront

Eine Liste der RSA- und ECDSA-Chiffren, die Sie in HTTPS-Verbindungen aushandeln können CloudFront , finden Sie unter und. the section called “Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront” the section called “Unterstützte Protokolle und Chiffren zwischen und dem Ursprung CloudFront ”

Privater Aktivierungsschlüssel

Bei Verwendung eines Zertifikats von einer Drittanbieter-Zertifizierungsstelle sind folgende Punkte zu beachten:

- Der private Schlüssel muss dem öffentlichen Schlüssel des Zertifikats entsprechen.
- Der private Schlüssel muss im PEM-Format vorliegen.
- Der private Schlüssel kann nicht mit einem Passwort verschlüsselt werden.

Wenn AWS Certificate Manager (ACM) das Zertifikat bereitgestellt hat, gibt ACM den privaten Schlüssel nicht frei. Der private Schlüssel wird in ACM gespeichert und kann von AWS Diensten verwendet werden, die in ACM integriert sind.

Berechtigungen

Sie müssen über die Berechtigung verfügen, das Zertifikat zu verwenden und zu importieren. SSL/TLS Wenn Sie AWS Certificate Manager (ACM) verwenden, empfehlen wir, dass Sie AWS Identity and Access Management Berechtigungen verwenden, um den Zugriff auf die Zertifikate einzuschränken. Weitere Informationen finden Sie unter Identity and Access Management im AWS Certificate Manager -Benutzerhandbuch.

Länge des Zertifikatschlüssels

Die Größe des CloudFront unterstützten Zertifikatsschlüssels hängt von der Art des Schlüssels und des Zertifikats ab.

Für RSA-Zertifikate:

- CloudFront unterstützt 1024-Bit-, 2048-Bit-, 3072-Bit- und 4096-Bit-RSA-Schlüssel. Die maximale Schlüssellänge für ein RSA-Zertifikat, das Sie mit verwenden, beträgt 4096 Bit. CloudFront

  Beachten Sie, dass ACM RSA-Zertifikate mit bis zu 2048-Bit-Schlüsseln ausstellt. Um ein 3072-Bit- oder 4096-Bit-RSA-Zertifikat zu verwenden, müssen Sie das Zertifikat extern beziehen und in ACM importieren. Danach steht es Ihnen zur Verwendung zur Verfügung. CloudFront

  Informationen zum Ermitteln der Länge des RSA-Schlüssels finden Sie unter Ermitteln Sie die Größe des öffentlichen Schlüssels in einem SSL/TLS RSA-Zertifikat.

Für ECDSA-Zertifikate:

- CloudFront unterstützt 256-Bit-Schlüssel. Um ein ECDSA-Zertifikat in ACM zu verwenden, das HTTPS zwischen Zuschauern und erfordert, verwenden Sie die elliptische CloudFront Prime256v1-Kurve.

Unterstützte Typen von Zertifikaten

CloudFront unterstützt alle Arten von Zertifikaten, die von einer vertrauenswürdigen Zertifizierungsstelle ausgestellt wurden.

Ablaufdatum des Zertifikats und Zertifikaterneuerung

Wenn Sie Zertifikate verwenden, die Sie von einer Zertifizierungsstelle (CA) eines Drittanbieters erhalten, müssen Sie die Ablaufdaten der Zertifikate überwachen und die Zertifikate, die Sie in AWS Certificate Manager (ACM) importieren oder in den AWS Identity and Access Management Zertifikatsspeicher hochladen, erneuern, bevor sie ablaufen.

> **Important**
>
> Um Probleme mit dem Ablauf von Zertifikaten zu vermeiden, verlängern oder importieren Sie Ihr Zertifikat mindestens 24 Stunden vor dem `NotAfter` Wert Ihres aktuellen Zertifikats erneut. Wenn Ihr Zertifikat innerhalb von 24 Stunden abläuft, fordern Sie ein neues Zertifikat von ACM an oder importieren Sie ein neues Zertifikat in ACM. Ordnen Sie als Nächstes das neue Zertifikat der CloudFront Distribution zu.
> CloudFront verwendet möglicherweise weiterhin das vorherige Zertifikat, während Ihr Zertifikat erneuert oder erneut importiert wird. Dies ist ein asynchroner Vorgang, der bis zu 24 Stunden dauern kann, bis Ihre Änderungen CloudFront angezeigt werden.

Wenn Sie von ACM bereitgestellte Zertifikate verwenden, verwaltet ACM die Zertifikatserneuerungen für Sie. Weitere Informationen finden Sie unter Verwaltete Erneuerung im AWS Certificate Manager - Benutzerhandbuch.

Domainnamen in der CloudFront Distribution und im Zertifikat

Wenn Sie einen benutzerdefinierten Ursprung verwenden, enthält das SSL/TLS Zertifikat für Ihren Ursprung einen Domainnamen im Feld Common Name und möglicherweise mehrere weitere im Feld Subject Alternative Names. (CloudFront unterstützt Platzhalterzeichen in Zertifikatsdomänennamen.)

Einer der Domänennamen im Zertifikat muss mit dem Domänennamen übereinstimmen, den Sie in „Origin Domain Name (Ursprungsdomänenname)“ angeben. Wenn kein Domainname übereinstimmt, wird der HTTP-Statuscode `502 (Bad Gateway)` an den Betrachter CloudFront zurückgegeben.

> **Important**
>
> Wenn Sie einer Distribution einen alternativen Domainnamen hinzufügen, wird CloudFront überprüft, ob der alternative Domainname durch das Zertifikat abgedeckt ist, das Sie angehängt haben. Das Zertifikat muss den alternativen Domänennamen im Feld „Subject Alternate Name (SAN)“ des Zertifikats abdecken. Dies bedeutet, dass das SAN-Feld eine exakte Übereinstimmung mit dem alternativen Domänennamen oder einen Platzhalter auf der gleichen Ebene des alternativen Domänennamens enthalten muss, den Sie hinzufügen. Weitere Informationen finden Sie unter Voraussetzungen für die Verwendung von alternativen Domänennamen.

Minimale SSL/TLS Protokollversion

Wenn Sie dedizierte IP-Adressen verwenden, legen Sie die SSL/TLS Mindestprotokollversion für die Verbindung zwischen Zuschauern fest und CloudFront wählen Sie eine Sicherheitsrichtlinie aus.

Weitere Informationen finden Sie Sicherheitsrichtlinie (Mindestversion von SSL/TLS) im Thema Referenz für alle Verteilungseinstellungen.

Unterstützte HTTP-Versionen

Wenn Sie ein Zertifikat mehreren CloudFront Distributionen zuordnen, müssen alle mit dem Zertifikat verknüpften Distributionen dieselbe Option für Unterstützte HTTP-Versionen verwenden. Sie geben diese Option an, wenn Sie eine CloudFront Verteilung erstellen oder aktualisieren.

Kontingente für die Verwendung von SSL/TLS Zertifikaten mit CloudFront (HTTPS nur zwischen Zuschauern und CloudFront nur)

Beachten Sie die folgenden Kontingente für die Verwendung von SSL/TLS Zertifikaten mitCloudFront. Diese Kontingente gelten nur für SSL/TLS Zertifikate, die Sie mithilfe von AWS Certificate Manager (ACM) bereitstellen, die Sie in ACM importieren oder für die HTTPS-Kommunikation zwischen Zuschauern und in den IAM-Zertifikatsspeicher hochladen. CloudFront

Weitere Informationen finden Sie unter Erhöhen Sie die Kontingente für SSL/TLS-Zertifikate.

Maximale Anzahl von Zertifikaten pro Distribution CloudFront

Sie können jeder CloudFront Distribution maximal ein SSL/TLS Zertifikat zuordnen.

Maximale Anzahl an Zertifikaten, die Sie in ACM importieren oder in den IAM-Zertifikatspeicher hochladen können

Wenn Sie Ihre SSL/TLS Zertifikate von einer Drittanbieter-Zertifizierungsstelle bezogen haben, müssen Sie die Zertifikate an einem der folgenden Orte speichern:

- AWS Certificate Manager – Informationen zum aktuellen Kontingent für die Anzahl der ACM-Zertifikate finden Sie unter Kontingente im AWS Certificate Manager -Benutzerhandbuch. Das angegebene Kontingent umfasst sowohl die Zertifikate, die Sie mithilfe von ACM bereitstellen, als auch die Zertifikate, die Sie in ACM importieren.
- IAM-Zertifikatsspeicher — Informationen zum aktuellen Kontingent (früher als Limit bezeichnet) für die Anzahl der Zertifikate, die Sie für ein AWS Konto in den IAM-Zertifikatsspeicher hochladen können, finden Sie unter IAM- und STS-Grenzwerte im IAM-Benutzerhandbuch. Sie können in der Service Quotas-Konsole ein höheres Kontingent anfordern.

Maximale Anzahl von Zertifikaten pro AWS Konto (nur dedizierte IP-Adressen)

Wenn Sie HTTPS-Anforderungen unter Verwendung dedizierter IP-Adressen bedienen möchten, beachten Sie bitte folgende Hinweise:

- Erlaubt CloudFront Ihnen standardmäßig, zwei Zertifikate mit Ihrem AWS Konto zu verwenden, eines für den täglichen Gebrauch und eines für den Fall, dass Sie Zertifikate für mehrere Distributionen rotieren müssen.
- Wenn Sie mehr als zwei benutzerdefinierte SSL/TLS Zertifikate für Ihr AWS Konto benötigen, können Sie in der Service Quotas Quotas-Konsole ein höheres Kontingent beantragen.

Verwenden Sie dasselbe Zertifikat für CloudFront Distributionen, die mit unterschiedlichen AWS Konten erstellt wurden

Wenn Sie eine Zertifizierungsstelle eines Drittanbieters verwenden und dasselbe Zertifikat mit mehreren CloudFront Distributionen verwenden möchten, die mit unterschiedlichen AWS Konten erstellt wurden, müssen Sie das Zertifikat in ACM importieren oder es für jedes Konto einmal in den IAM-Zertifikatsspeicher hochladen. AWS

Wenn Sie von ACM bereitgestellte Zertifikate verwenden, können Sie nicht so konfigurieren, dass Zertifikate verwendet werden, CloudFront die von einem anderen Konto erstellt wurden. AWS

Verwenden Sie dasselbe Zertifikat für CloudFront und für andere Dienste AWS

Wenn Sie ein Zertifikat von einer vertrauenswürdigen Zertifizierungsstelle wie Comodo oder Symantec erworben haben, können Sie dasselbe Zertifikat für CloudFront und für andere AWS Dienste verwenden. DigiCert Wenn Sie das Zertifikat in ACM importieren, müssen Sie es nur einmal importieren, um es für mehrere AWS -Services verwenden zu können.

Wenn Sie von ACM bereitgestellte Zertifikate verwenden, werden die Zertifikate in ACM gespeichert.

Verwenden Sie dasselbe Zertifikat für mehrere Distributionen CloudFront

Sie können ein und dasselbe Zertifikat für beliebige oder alle CloudFront-Verteilungen nutzen, die Sie zur Bedienung von HTTPS-Anforderungen verwenden. Beachten Sie Folgendes:

- Sie können mit ein und demselben Zertifikat Anforderungen sowohl unter Verwendung dedizierter IP-Adressen als auch unter Verwendung von SNI bedienen.
- Sie können jeweils nur ein Zertifikat mit jeder Verteilung verknüpfen.
- Jede Verteilung muss einen oder mehrere alternative Domänennamen beinhalten, die auch im Feld Common Name (Allgemeiner Name) bzw. im Feld Subject Alternative Names (Alternative Subjektnamen) im Zertifikat angezeigt werden.
- Wenn Sie HTTPS-Anfragen mit dedizierten IP-Adressen bearbeiten und alle Ihre Distributionen mit demselben AWS Konto erstellt haben, können Sie Ihre Kosten erheblich senken, indem Sie für alle Distributionen dasselbe Zertifikat verwenden. CloudFront Gebühren für jedes Zertifikat, nicht für jede Verteilung.

  Nehmen wir beispielsweise an, Sie erstellen drei Distributionen mit demselben AWS Konto und verwenden dasselbe Zertifikat für alle drei Distributionen. In diesem Fall würde Ihnen nur eine Gebühr für die Nutzung dedizierter IP-Adressen in Rechnung gestellt.

Wenn Sie jedoch HTTPS-Anfragen mit dedizierten IP-Adressen bearbeiten und dasselbe Zertifikat verwenden, um CloudFront Verteilungen in verschiedenen AWS Konten zu erstellen, wird jedem Konto die Gebühr für die Verwendung dedizierter IP-Adressen berechnet. Wenn Sie beispielsweise drei Verteilungen mithilfe von drei verschiedenen AWS Konten erstellen und dasselbe Zertifikat für alle drei Verteilungen verwenden, wird jedem Konto die volle Gebühr für die Verwendung dedizierter IP-Adressen berechnet.

Konfigurieren Sie alternative Domainnamen und HTTPS

Um alternative Domainnamen URLs für Ihre Dateien und HTTPS zwischen Zuschauern und zu verwenden CloudFront, führen Sie die entsprechenden Verfahren aus.

Themen

Besorgen Sie sich ein SSL/TLS Zertifikat

Besorgen Sie sich ein SSL/TLS Zertifikat, falls Sie noch keines haben. Weitere Informationen finden Sie in der entsprechenden Dokumentation:

- Informationen zur Verwendung eines von AWS Certificate Manager (ACM) bereitgestellten Zertifikats finden Sie im AWS Certificate Manager Benutzerhandbuch. Fahren Sie anschließend fort mit der unter Aktualisieren Sie Ihre CloudFront Distribution beschriebenen Anleitung.

  > **Note**
  >
  > Wir empfehlen, ACM für die Bereitstellung, Verwaltung und Bereitstellung von SSL/TLS Zertifikaten auf AWS verwalteten Ressourcen zu verwenden. Sie müssen ein ACM-Zertifikat in der Region USA Ost (Nord-Virginia) anfordern.

- Wenn Sie ein Zertifikat von einer Zertifizierungsstelle (CA, Certificate Authority) eines Drittanbieters beziehen möchten, finden Sie entsprechende Informationen zur Vorgehensweise in der Dokumentation der Zertifizierungsstelle. Wenn Sie das Zertifikat haben, fahren Sie mit dem nächsten Schritt fort.

Importieren Sie ein SSL/TLS-Zertifikat

Wenn Sie Ihr Zertifikat von einer Drittanbieter-Zertifizierungsstelle bezogen haben, importieren Sie das Zertifikat in ACM oder laden Sie es in den IAM-Zertifikatspeicher hoch:

ACM (empfohlen)

Mit ACM können Sie Drittanbieter-Zertifikate sowohl programmgesteuert als auch über die ACM-Konsole importieren. Informationen zum Importieren eines Zertifikats in ACM finden Sie unter Importieren von Zertifikaten in AWS Certificate Manager im AWS Certificate Manager - Benutzerhandbuch. Sie müssen das Zertifikat in der Region USA Ost (Nord-Virginia) importieren.

IAM-Zertifikatspeicher

(Nicht empfohlen) Verwenden Sie den folgenden AWS CLI Befehl, um Ihr Drittanbieterzertifikat in den IAM-Zertifikatsspeicher hochzuladen.

```
aws iam upload-server-certificate \
        --server-certificate-name CertificateName \
        --certificate-body file://public_key_certificate_file \
        --private-key file://privatekey.pem \
        --certificate-chain file://certificate_chain_file \
        --path /cloudfront/path/
```

Beachten Sie Folgendes:

- AWS Konto — Sie müssen das Zertifikat mit demselben AWS Konto, mit dem Sie Ihre CloudFront Distribution erstellt haben, in den IAM-Zertifikatsspeicher hochladen.
- --path-Parameter – Wenn Sie Ihr Zertifikat in IAM hochladen, muss der Wert des Parameters `--path` (Zertifikatspfad) mit `/cloudfront/` beginnen, wie z. B. `/cloudfront/production/` oder `/cloudfront/test/`. Zudem muss der Pfad mit einem „/“ (Schrägstrich) enden.
- Vorhandene Zertifikate – Die Werte, die Sie für die Parameter `--server-certificate-name` und `--path` angeben, müssen sich von den entsprechenden Werten bereits vorhandener Zertifikate unterscheiden.
- Verwendung der CloudFront Konsole — Der Wert, den Sie für den `--server-certificate-name` Parameter in angeben AWS CLI, wird beispielsweise in der Liste der SSL-Zertifikate in der CloudFront Konsole angezeigt. `myServerCertificate`
- Verwenden der CloudFront API — Notieren Sie sich die alphanumerische Zeichenfolge, die AWS CLI zurückgegeben wird, `AS1A2M3P4L5E67SIIXR3J` z. B. Dies ist der Wert, den Sie im

Element `IAMCertificateId` angeben müssen. Den ebenfalls von der CLI zurückgegebenen IAM-ARN können Sie ignorieren.

Weitere Informationen zu finden Sie im AWS CLIAWS Command Line Interface Benutzerhandbuch und in der AWS CLI Befehlsreferenz.

Aktualisieren Sie Ihre CloudFront Distribution

Verfahren Sie wie folgt, um Einstellungen für Ihre Verteilung zu aktualisieren:

Um Ihre CloudFront Distribution für alternative Domainnamen zu konfigurieren

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die ID für die Verteilung aus, die Sie aktualisieren möchten.
3. Wählen Sie auf der Registerkarte General die Option Edit aus.
4. Aktualisieren Sie die folgenden Werte:

   Alternativer Domainname (CNAME)

   Wählen Sie Element hinzufügen, um die entsprechenden alternativen Domainnamen hinzuzufügen. Trennen Sie einzelne Domain-Namen mit Kommas oder geben Sie jeden Domain-Namen in einer neuen Zeile ein.

   Benutzerdefiniertes SSL-Zertifikat

   Wählen Sie ein Zertifikat aus der Dropdown-Liste aus.

   Bis zu 100 Zertifikate sind hier aufgelistet. Bei mehr als 100 Zertifikaten geben Sie ggf. einen Zertifikat-ARN in das Feld ein, um das Zertifikat, das Sie hinzufügen möchten, auszuwählen.

   Wenn Sie ein Zertifikat in den IAM-Zertifikatspeicher hochgeladen haben, das Zertifikat in der Liste jedoch nicht angezeigt wird und Sie es durch Eingabe des Namens in dem Feld nicht auswählen können, überprüfen Sie anhand der unter Importieren Sie ein SSL/TLS-Zertifikat beschriebenen Vorgehensweise, ob Sie das Zertifikat korrekt hochgeladen haben.

   > **Important**
   >
   > Nachdem Sie Ihr SSL/TLS Zertifikat mit Ihrer CloudFront Distribution verknüpft haben, löschen Sie das Zertifikat nicht aus ACM oder dem IAM-Zertifikatsspeicher,

bis Sie das Zertifikat aus allen Distributionen entfernt haben und alle Distributionen bereitgestellt sind.

5. Wählen Sie Änderungen speichern aus.
6. Konfigurieren Sie CloudFront es so, dass HTTPS zwischen Zuschauern erforderlich ist und: CloudFront

   a. Wählen Sie auf der Registerkarte Behaviors das zu aktualisierende Cache-Verhalten und anschließend Edit aus.

   b. Geben Sie einen der folgenden Werte für Viewer Protocol Policy an:

      Redirect HTTP to HTTPS

      Betrachter können beide Protokolle verwenden. HTTP-Anforderungen jedoch werden automatisch an HTTPS-Anforderungen umgeleitet. CloudFront gibt den HTTP-Statuscode `301 (Moved Permanently)` zusammen mit der neuen HTTPS-URL zurück. Der Betrachter sendet die Anfrage dann erneut CloudFront unter Verwendung der HTTPS-URL.

      > **Important**
      >
      > CloudFront leitet`DELETE`,, `OPTIONS` `PATCH``POST`, oder `PUT` Anfragen nicht von HTTP zu HTTPS weiter. Wenn Sie ein Cache-Verhalten so konfigurieren, dass es zu HTTPS umleitet`DELETE`, CloudFront reagiert auf HTTP- `OPTIONS``PATCH`,`POST`,, oder `PUT` Anfragen für dieses Cache-Verhalten mit HTTP-Statuscode`403 (Forbidden)`.

      Wenn ein Betrachter eine HTTP-Anfrage stellt, die zu einer HTTPS-Anfrage umgeleitet wird, CloudFront fallen Gebühren für beide Anfragen an. Die Gebühren für die HTTP-Anforderung beziehen sich nur auf die Anforderung und die Header, die von CloudFront an den Viewer zurückgegeben werden. Bezüglich der HTTPS-Anforderung fallen Gebühren für die Anforderung sowie für die vom Ursprung zurückgegebenen Header und die Datei an, die vom Ursprung zurückgegeben wird.

HTTPS Only

Viewer können auf Ihre Inhalte nur zugreifen, wenn sie HTTPS verwenden. Wenn ein Betrachter statt einer HTTPS-Anfrage eine HTTP-Anfrage sendet, CloudFront gibt er den HTTP-Statuscode zurück `403 (Forbidden)` und gibt die Datei nicht zurück.

c. Wählen Sie Yes, Edit aus.

d. Wiederholen Sie Schritt a bis c für jedes weitere Cache-Verhalten, das die Verwendung von HTTPS für die Kommunikation zwischen Viewern und CloudFront erzwingen soll.

7. Vergewissern Sie sich, dass folgende Punkte erfüllt sind, bevor Sie die aktualisierte Konfiguration in einer Produktionsumgebung verwenden:

   - Das Pfadmuster eines jeden Cache-Verhaltens wird nur auf die Anforderungen angewendet, für die die Viewer HTTPS verwenden sollen.
   - Die Cache-Verhaltensweisen sind in der Reihenfolge aufgeführt, in der sie von CloudFront ausgewertet werden sollen. Weitere Informationen finden Sie unter Pfadmuster.
   - Die Cache-Verhaltensweisen leiten Anforderungen an die richtigen Ursprünge weiter.

Ermitteln Sie die Größe des öffentlichen Schlüssels in einem SSL/TLS RSA-Zertifikat

Wenn Sie CloudFront alternative Domainnamen und HTTPS verwenden, beträgt die maximale Größe des öffentlichen Schlüssels in einem SSL/TLS RSA-Zertifikat 4096 Bit. (Dies bezieht sich auf die Schlüsselgröße, nicht auf die Anzahl der Zeichen in dem öffentlichen Schlüssel.) Wenn Sie AWS Certificate Manager für Ihre Zertifikate verwenden, unterstützt ACM zwar größere RSA-Schlüssel, Sie können die größeren Schlüssel jedoch nicht mit verwenden. CloudFront

Sie können die Länge des öffentlichen RSA-Schlüssels durch Ausführen des folgenden OpenSSL-Befehls ermitteln:

```
openssl x509 -in path and filename of SSL/TLS certificate -text -noout
```

Wobei gilt:

- `-in` gibt den Pfad und den Dateinamen Ihres SSL/TLS RSA-Zertifikats an.
- `-text` veranlasst, dass OpenSSL die Länge des öffentlichen RSA-Schlüssels in Bit anzeigt.
- `-noout` verhindert, dass OpenSSL den öffentlichen Schlüssel anzeigt.

Beispielausgabe:

```
Public-Key: (2048 bit)
```

Erhöhen Sie die Kontingente für SSL/TLS-Zertifikate

Es gibt Kontingente für die Anzahl der SSL/TLS Zertifikate, die Sie in AWS Certificate Manager (ACM) importieren oder hochladen AWS Identity and Access Management (IAM) können. Es gibt auch ein Kontingent für die Anzahl der SSL/TLS Zertifikate, die Sie zusammen mit und AWS-Konto bei der Konfiguration CloudFront für die Bearbeitung von HTTPS-Anfragen mithilfe von dedizierten IP-Adressen verwenden können. Sie haben jedoch die Möglichkeit, höhere Kontingente anzufordern.

Themen

Erhöhen Sie das Kontingent für in ACM importierte Zertifikate

Informationen zum Kontingent bezüglich der Anzahl von Zertifikaten, die Sie in ACM importieren können, finden Sie unter Kontingente im AWS Certificate Manager -Benutzerhandbuch.

Verwenden Sie die Service Quotas-Konsole, um ein höheres Kontingent anzufordern. Weitere Informationen finden Sie unter Beantragen einer Kontingenterhöhung im Service-Quotas-Benutzerhandbuch.

Erhöhen Sie das Kontingent für Zertifikate, die auf IAM hochgeladen wurden

Informationen zum Kontingent (früher als Limit bezeichnet) für die Anzahl der Zertifikate, die Sie auf IAM hochladen können, finden Sie unter IAM- und STS-Limits im AM Benutzerhandbuch.

Verwenden Sie die Service Quotas-Konsole, um ein höheres Kontingent anzufordern. Weitere Informationen finden Sie unter Beantragen einer Kontingenterhöhung im Service-Quotas-Benutzerhandbuch.

Erhöhen Sie das Kontingent für Zertifikate, die mit dedizierten IP-Adressen verwendet werden

Informationen zum Kontingent für die Anzahl der SSL-Zertifikate, die Sie für jedes Zertifikat verwenden können, AWS-Konto wenn Sie HTTPS-Anfragen mit dedizierten IP-Adressen bearbeiten, finden Sie unterKontingente für SSL-Zertifikate.

Verwenden Sie die Service Quotas-Konsole, um ein höheres Kontingent anzufordern. Weitere Informationen finden Sie unter Beantragen einer Kontingenterhöhung im Service-Quotas-Benutzerhandbuch.

SSL/TLS Zertifikate rotieren

Wenn Ihre SSL/TLS Zertifikate bald ablaufen, müssen Sie sie rotieren, um die Sicherheit Ihrer Verteilung zu gewährleisten und Serviceunterbrechungen für Ihre Zuschauer zu vermeiden. Sie können sie auf folgende Weise rotieren:

- Bei SSL/TLS Zertifikaten, die von AWS Certificate Manager (ACM) bereitgestellt werden, müssen Sie sie nicht rotieren. ACM verwaltet automatisch die Verlängerungen von Zertifikaten für Sie. Weitere Informationen finden Sie unter Verwaltete Zertifikatserneuerung im AWS Certificate Manager Benutzerhandbuch.
- Wenn Sie eine Zertifizierungsstelle eines Drittanbieters verwenden und die Zertifikate in ACM importiert (empfohlen) oder in den IAM-Zertifikatsspeicher hochgeladen haben, müssen Sie gelegentlich ein Zertifikat durch ein anderes ersetzen.

⚠ Important

- ACM verwaltet keine Zertifikatserneuerungen für Zertifikate, die Sie von Zertifizierungsstellen Dritter erwerben und in ACM importieren.
- Wenn Sie so konfiguriert haben CloudFront , dass HTTPS-Anfragen mithilfe von dedizierten IP-Adressen bearbeitet werden, fallen möglicherweise zusätzliche, anteilige Gebühren für die Verwendung eines oder mehrerer zusätzlicher Zertifikate an, während Sie Zertifikate rotieren. Wir empfehlen Ihnen, Ihre Distributionen zu aktualisieren, um die zusätzlichen Kosten zu minimieren.

Rotieren Sie die Zertifikate SSL/TLS

Gehen Sie wie folgt vor, um Ihre Zertifikate zu rotieren. Viewer können weiterhin auf Ihre Inhalte zugreifen, während Sie Zertifikate rotieren, aber auch nachdem der Vorgang abgeschlossen ist.

So rotieren Sie SSL/TLS-Zertifikate:

1. Überprüfen Sie anhand des Abschnitts Erhöhen Sie die Kontingente für SSL/TLS-Zertifikate, ob Sie eine Berechtigung zur Verwendung mehrerer SSL-Zertifikate benötigen. Wenn dies der Fall ist, beantragen Sie die Berechtigung und warten Sie, bis die Berechtigung erteilt ist, bevor Sie mit Schritt 2 fortfahren.
2. Importieren Sie das neue Zertifikat in ACM oder laden Sie es zu IAM hoch. Weitere Informationen finden Sie unter Import eines SSL/TLS Zertifikats im Amazon CloudFront Developer Guide.
3. (Nur für IAM-Zertifikate) Aktualisieren Sie Ihre Distributionen nacheinander, um das neue Zertifikat zu verwenden. Weitere Informationen finden Sie unter Eine Verteilung aktualisieren.
4. (Optional) Löschen Sie das vorherige Zertifikat aus ACM oder IAM.

> **Important**
>
> Löschen Sie ein SSL/TLS Zertifikat erst, wenn Sie es aus allen Distributionen entfernt haben und der Status der Distributionen, die Sie aktualisiert haben, geändert hat. `Deployed`

Kehren Sie von einem benutzerdefinierten SSL/TLS-Zertifikat zum Standardzertifikat zurück CloudFront

Wenn Sie CloudFront für die Verwendung von HTTPS zwischen Zuschauern und konfiguriert haben und CloudFront Sie für die Verwendung eines benutzerdefinierten SSL/TLS Zertifikats konfiguriert CloudFront haben, können Sie Ihre Konfiguration so ändern, dass das standardmäßige CloudFront SSL/TLS-Zertifikat verwendet wird. Die Vorgehensweise hängt davon ab, ob Sie Ihre Verteilung bereits zur Verteilung Ihrer Inhalte verwendet haben:

- Wenn Sie Ihre Verteilung noch nicht zur Verteilung Ihrer Inhalte genutzt haben, können Sie einfach die Konfiguration ändern. Weitere Informationen finden Sie unter Eine Verteilung aktualisieren.

- Wenn Sie Ihre Inhalte über Ihre Distribution verteilt haben, müssen Sie eine neue CloudFront Distribution erstellen und die Einstellungen URLs für Ihre Dateien ändern, um die Zeit, in der Ihre Inhalte nicht verfügbar sind, zu verringern oder zu vermeiden. In diesem Fall verfahren Sie wie nachfolgend beschrieben.

Kehren Sie zum Standardzertifikat zurück CloudFront

Das folgende Verfahren zeigt Ihnen, wie Sie von einem benutzerdefinierten SSL/TLS Zertifikat zum CloudFront Standardzertifikat zurückkehren.

So kehren Sie zum Standardzertifikat zurück CloudFront

1. Erstellen Sie eine neue CloudFront Distribution mit der gewünschten Konfiguration. Wählen Sie für SSL Certificate die Option Default CloudFront Certificate (*.cloudfront.net) aus.

   Weitere Informationen finden Sie unter Eine Verteilung erstellen.
2. Für Dateien, mit denen Sie verteilen CloudFront, aktualisieren Sie die URLs in Ihrer Anwendung, sodass sie den Domänennamen verwenden, CloudFront der der neuen Distribution zugewiesen wurde. Sie können beispielsweise `https://www.example.com/images/logo.png` in `https://d111111abcdef8.cloudfront.net/images/logo.png` ändern.
3. Löschen Sie entweder die Distribution, die einem benutzerdefinierten SSL/TLS-Zertifikat zugeordnet ist, oder aktualisieren Sie die Verteilung, um den Wert von SSL Certificate in Default CloudFront Certificate (*.cloudfront.net) zu ändern. Weitere Informationen finden Sie unter Eine Verteilung aktualisieren.

   > **Important**
   >
   > Solange Sie diesen Schritt nicht abgeschlossen haben, werden Ihnen AWS weiterhin Gebühren für die Nutzung eines benutzerdefinierten Zertifikats berechnet. SSL/TLS
4. (Optional) Löschen Sie Ihr benutzerdefiniertes SSL/TLS Zertifikat.

   a. Führen Sie den AWS CLI Befehl aus`list-server-certificates`, um die Zertifikat-ID des Zertifikats abzurufen, das Sie löschen möchten. Weitere Informationen finden Sie unter list-server-certificates in der Referenz zum AWS CLI -Befehl.

   b. Führen Sie den AWS CLI Befehl aus`delete-server-certificate`, um das Zertifikat zu löschen. Weitere Informationen finden Sie unter delete-server-certificate in der Referenz zum AWS CLI -Befehl.

Wechseln Sie von einem benutzerdefinierten SSL/TLS-Zertifikat mit dedizierten IP-Adressen zu SNI

Wenn Sie für CloudFront die Verwendung eines benutzerdefinierten SSL/TLS Zertifikats mit dedizierten IP-Adressen konfiguriert haben, können Sie stattdessen auf die Verwendung eines benutzerdefinierten SSL/TLS Zertifikats mit SNI umsteigen und die mit dedizierten IP-Adressen verbundenen Gebühren vermeiden.

> **Important**
>
> Dieses Update Ihrer CloudFront Konfiguration hat keine Auswirkungen auf Zuschauer, die SNI unterstützen. Zuschauer können vor und nach der Änderung sowie während der Übertragung der Änderung an CloudFront Edge-Standorten auf Ihre Inhalte zugreifen. Zuschauer, die SNI nicht unterstützen, können nach der Änderung nicht auf deine Inhalte zugreifen. Weitere Informationen finden Sie unter Wählen Sie aus, wie CloudFront HTTPS-Anfragen bearbeitet werden.

Wechseln Sie von einem benutzerdefinierten Zertifikat zu SNI

Das folgende Verfahren zeigt Ihnen, wie Sie von einem benutzerdefinierten SSL/TLS Zertifikat mit dedizierten IP-Adressen zu SNI wechseln.

So wechseln Sie von einem benutzerdefinierten SSL/TLS Zertifikat mit dedizierten IP-Adressen zu SNI

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die ID der Verteilung aus, die Sie anzeigen oder aktualisieren möchten.
3. Wählen Sie Distribution Settings aus.
4. Wählen Sie auf der Registerkarte General die Option Edit aus.
5. Deaktivieren Sie unter Benutzerdefinierte SSL-Zertifizierung — optional die Option Unterstützung für ältere Clients.
6. Wählen Sie Yes, Edit aus.

Stellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit

Viele Unternehmen, die Inhalte über das Internet verteilen, möchten den Zugriff auf Dokumente, geschäftliche Daten, Medien-Streams oder Inhalte für ausgewählte Benutzer (z. B. Benutzer, die eine Gebühr bezahlt haben) einschränken. Um diesen privaten Inhalt mithilfe von sicher bereitzustellen CloudFront, können Sie Folgendes tun:

- Erfordern Sie, dass Ihre Benutzer mithilfe spezieller CloudFront signierter URLs oder signierter Cookies auf Ihre privaten Inhalte zugreifen.
- Erfordern Sie, dass Ihre Benutzer auf Ihre Inhalte zugreifen CloudFront URLs, indem sie Inhalte verwenden, nicht URLs direkt auf dem Ursprungsserver (z. B. Amazon S3 oder einen privaten HTTP-Server) zugreifen. Eine Anforderung CloudFront URLs ist nicht erforderlich, wir empfehlen es jedoch, um zu verhindern, dass Benutzer die Einschränkungen umgehen, die Sie in signierten URLs oder signierten Cookies angeben.

Weitere Informationen finden Sie unter Beschränken Sie den Zugriff auf Dateien.

Wie werden private Inhalte bereitgestellt

Gehen Sie wie folgt vor, CloudFront um die Bereitstellung privater Inhalte zu konfigurieren:

1. (Optional, aber empfohlen) Erfordern Sie, dass Ihre Benutzer nur über auf Ihre Inhalte zugreifen CloudFront. Die dafür verwendete Methode hängt davon ab, ob Sie Amazon S3 oder benutzerdefinierte Ursprünge verwenden:

   - Amazon S3 – Siehe the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”.
   - Benutzerdefinierter Ursprung – Siehe Beschränken Sie den Zugriff auf Dateien mit benutzerdefinierten Ursprüngen.

   Zu den benutzerdefinierten Ursprüngen gehören Amazon EC2, Amazon S3-Buckets, die als Website-Endpunkte konfiguriert sind, Elastic Load Balancing und Ihre eigenen HTTP-Webserver.
2. Geben Sie die vertrauenswürdigen Schlüsselgruppen oder vertrauenswürdigen Unterzeichner an, die Sie verwenden möchten, um signierte URLs oder signierte Cookies zu erstellen. Es wird empfohlen, vertrauenswürdige Schlüsselgruppen zu verwenden. Weitere Informationen

finden Sie unter Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können.

3. Schreiben Sie Ihre Anwendung so, dass sie auf Anfragen von autorisierten Benutzern entweder mit signierten URLs oder mit `Set-Cookie` Headern reagiert, die signierte Cookies setzen. Befolgen Sie die Schritte auf einem der folgenden Themen:

   - Verwenden Sie signierte URLs
   - Verwenden Sie signierte Cookies

   Wenn Sie sich nicht sicher sind, welche Methode Sie verwenden sollen, lesen Sie nach unter Entscheiden Sie sich dafür, signierte URLs oder signierte Cookies zu verwenden.

Themen

Beschränken Sie den Zugriff auf Dateien

Sie können den Benutzerzugriff auf Ihre privaten Inhalte auf zwei Arten steuern:

- Beschränken Sie den Zugriff auf Dateien in CloudFront Caches.
- Beschränken Sie den Zugriff auf Dateien auf Ihrem Ursprungsserver, indem Sie einen der folgenden Schritte ausführen:
  - Richten Sie eine Ursprungszugriffssteuerung (OAC) für Ihren Amazon-S3-Bucket ein.
  - Konfigurieren Sie benutzerdefinierte Header für einen privaten HTTP-Server (ein benutzerdefinierter Ursprung).

Beschränken Sie den Zugriff auf Dateien in Caches CloudFront

Sie können festlegen CloudFront , dass Benutzer entweder mit signierten URLs oder signierten Cookies auf Ihre Dateien zugreifen müssen. Anschließend entwickeln Sie Ihre Anwendung, um entweder signierte Benutzer zu erstellen und URLs an authentifizierte Benutzer zu verteilen oder um `Set-Cookie` Header zu senden, die signierte Cookies für authentifizierte Benutzer setzen. (Um einigen Benutzern langfristigen Zugriff auf eine kleine Anzahl von Dateien zu gewähren, können Sie signierte URLs Dateien auch manuell erstellen.)

Wenn Sie signierte URLs oder signierte Cookies erstellen, um den Zugriff auf Ihre Dateien zu kontrollieren, können Sie die folgenden Einschränkungen festlegen:

- Ein Enddatum und eine Endzeit, nach welchen die URL nicht mehr gültig ist.
- (Optional) Das Datum und die Zeit, an welchen die URL gültig wird.
- (Optional) Die IP-Adresse oder der IP-Adressbereich der Computer, die für den Zugriff auf Ihre Inhalte verwendet werden können.

Ein Teil einer signierten URL oder eines signierten Cookies wird gehasht und mit dem privaten Schlüssel von einem Schlüsselpaar aus einem privaten und einem öffentlichen Schlüssel signiert. Wenn jemand eine signierte URL oder ein signiertes Cookie verwendet, um auf eine Datei zuzugreifen, werden die signierten und unsignierten Teile der URL oder des Cookies CloudFront verglichen. Wenn sie nicht übereinstimmen, wird die Datei CloudFront nicht bereitgestellt.

Sie müssen RSA- SHA1 zum Signieren URLs oder für Cookies verwenden. CloudFront akzeptiert keine anderen Algorithmen.

Beschränken Sie den Zugriff auf Dateien in Amazon S3 S3-Buckets

Sie können den Inhalt in Ihrem Amazon S3-Bucket optional sichern, sodass Benutzer über die angegebene CloudFront Distribution darauf zugreifen können, aber nicht direkt mit Amazon S3 darauf zugreifen könnenURLs. Dadurch wird verhindert, dass jemand die Amazon S3 S3-URL umgeht CloudFront und verwendet, um Inhalte abzurufen, auf die Sie den Zugriff einschränken möchten. Dieser Schritt ist für die Verwendung von signed nicht erforderlichURLs, wir empfehlen ihn jedoch.

Gehen Sie wie folgt vor CloudFront URLs, damit Benutzer über auf Ihre Inhalte zugreifen können:

- Erteilen Sie einer CloudFront Origin-Zugriffskontrolle die Erlaubnis, die Dateien im S3-Bucket zu lesen.

- Erstellen Sie die Origin-Zugriffskontrolle und verknüpfen Sie sie mit Ihrer CloudFront Distribution.
- Entfernen Sie allen anderen die Erlaubnis, Amazon S3 URLs zum Lesen der Dateien zu verwenden.

Weitere Informationen finden Sie unter the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”.

Beschränken Sie den Zugriff auf Dateien mit benutzerdefinierten Ursprüngen

Wenn Sie einen benutzerdefinierten Ursprungsserver verwenden, können Sie optional benutzerdefinierte Header einrichten, um den Zugriff einzuschränken. CloudFront Damit Ihre Dateien von einem benutzerdefinierten Ursprung abgerufen werden können, müssen die Dateien über eine CloudFront standardmäßige HTTP- (oder HTTPS-) Anfrage zugänglich sein. Durch die Verwendung benutzerdefinierter Header können Sie den Zugriff auf Ihre Inhalte jedoch weiter einschränken, sodass Benutzer nur überCloudFront, nicht direkt darauf zugreifen können. Dieser Schritt ist nicht erforderlich, um signiert zu verwenden URLs, wir empfehlen ihn jedoch.

Um zu verlangen, dass Benutzer über auf Inhalte zugreifen CloudFront, ändern Sie die folgenden Einstellungen in Ihren CloudFront Distributionen:

Angepasste Ursprungs-Header

: Konfigurieren Sie CloudFront es so, dass benutzerdefinierte Header an Ihren Ursprung weitergeleitet werden. Siehe Konfiguriere CloudFront es so, dass benutzerdefinierte Header zu ursprünglichen Anfragen hinzugefügt werden.

Viewer-Protokollrichtlinien

: Konfigurieren Sie Ihre Verteilung so, dass Betrachter für den Zugriff auf Ihre CloudFront HTTPS verwenden müssen. Siehe Viewer-Protokollrichtlinien.

Ursprungsprotokollrichtlinien

: Konfigurieren Sie Ihre Distribution so, dass CloudFront sie dasselbe Protokoll wie die Zuschauer verwenden muss, um Anfragen an den Ursprung weiterzuleiten. Siehe Protokoll (nur benutzerdefinierte Ursprünge).

Nachdem Sie diese Änderungen vorgenommen haben, aktualisieren Sie Ihre Anwendung auf Ihrem benutzerdefinierten Ursprung, sodass nur Anfragen akzeptiert werden, die die benutzerdefinierten Header enthalten, die Sie für das Senden konfiguriert CloudFront haben.

Die Kombination aus Viewer Protocol Policy (Viewer-Protokollrichtlinie) und Origin Protocol Policy (Ursprungs-Protokollrichtlinie) stellt sicher, dass die benutzerdefinierten Header während der Übertragung verschlüsselt werden. Wir empfehlen Ihnen jedoch, regelmäßig wie folgt vorzugehen, um die benutzerdefinierten Header, die an Ihren Ursprung CloudFront weitergeleitet werden, zu rotieren:

1. Aktualisieren Sie Ihre CloudFront Distribution, um damit zu beginnen, einen neuen Header an Ihren benutzerdefinierten Absender weiterzuleiten.
2. Aktualisieren Sie Ihre Anwendung so, dass sie den neuen Header als Bestätigung dafür akzeptiert, dass die Anfrage stammt CloudFront.
3. Wenn Anfragen den Header, den Sie ersetzen, nicht mehr enthalten, aktualisieren Sie Ihre Anwendung so, dass der alte Header nicht mehr als Bestätigung dafür akzeptiert wird, dass die Anfrage stammt CloudFront.

Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können

Themen

Um signierte URLs oder signierte Cookies zu erstellen, benötigen Sie einen Unterzeichner. Ein Unterzeichner ist entweder eine vertrauenswürdige Schlüsselgruppe, in der Sie erstellen CloudFront, oder ein AWS Konto, das ein CloudFront key pair enthält. Wir empfehlen, vertrauenswürdige Schlüsselgruppen mit signierten URLs und signierten Cookies zu verwenden. Weitere Informationen finden Sie unter Wählen Sie zwischen vertrauenswürdigen Schlüsselgruppen (empfohlen) und AWS-Konten.

Der Aussteller erfüllt zwei Zwecke:

- Sobald Sie den Unterzeichner zu Ihrer Distribution hinzufügen, wird verlangt, CloudFront dass Zuschauer signierte URLs oder signierte Cookies verwenden, um auf Ihre Dateien zuzugreifen.

- Wenn Sie signierte URLs oder signierte Cookies erstellen, verwenden Sie den privaten Schlüssel aus dem key pair des Unterzeichners, um einen Teil der URL oder des Cookies zu signieren. Wenn jemand eine eingeschränkte Datei anfordert, CloudFront vergleicht er die Signatur in der URL oder im Cookie mit der unsignierten URL oder dem Cookie, um sicherzustellen, dass es nicht manipuliert wurde. CloudFront überprüft auch, ob die URL oder das Cookie gültig ist, was beispielsweise bedeutet, dass das Ablaufdatum und die Uhrzeit nicht abgelaufen sind.

Wenn Sie einen Unterzeichner angeben, geben Sie indirekt auch die Dateien an, für die signierte URLs oder signierte Cookies erforderlich sind, indem Sie den Unterzeichner zu einem Cache-Verhalten hinzufügen. Wenn Ihre Distribution nur ein Cache-Verhalten hat, müssen Zuschauer signierte URLs oder signierte Cookies verwenden, um auf jede Datei in der Distribution zuzugreifen. Wenn Sie mehrere Cache-Verhaltensweisen erstellen und einigen Cache-Verhalten Unterzeichner hinzufügen und anderen nicht, können Sie festlegen, dass Zuschauer signierte URLs oder signierte Cookies verwenden, um auf einige Dateien zuzugreifen und auf andere nicht.

Gehen Sie wie folgt vor, um die Unterzeichner (die privaten Schlüssel) anzugeben, die signierte URLs oder signierte Cookies erstellen dürfen, und um die Unterzeichner zu Ihrer CloudFront Distribution hinzuzufügen:

1. Entscheiden Sie, ob Sie eine vertrauenswürdige Schlüsselgruppe oder eine AWS-Konto als Unterzeichner verwenden möchten. Es wird empfohlen, eine vertrauenswürdige Schlüsselgruppe zu verwenden. Weitere Informationen finden Sie unter Wählen Sie zwischen vertrauenswürdigen Schlüsselgruppen (empfohlen) und AWS-Konten.
2. Erstellen Sie für den Aussteller, den Sie in Schritt 1 ausgewählt haben, Schlüsselpaar aus öffentlichem und privatem Schlüssel. Weitere Informationen finden Sie unter Erstellen Sie Schlüsselpaare für Ihre Unterzeichner.
3. Wenn Sie.NET oder Java verwenden, um signierte URLs oder signierte Cookies zu erstellen, formatieren Sie den privaten Schlüssel neu. Weitere Informationen finden Sie unter Formatieren Sie den privaten Schlüssel neu (nur .NET und Java).
4. Geben Sie in der Distribution, für die Sie signierte URLs oder signierte Cookies erstellen, den Unterzeichner an. Weitere Informationen finden Sie unter Fügen Sie einer Distribution einen Unterzeichner hinzu.

Wählen Sie zwischen vertrauenswürdigen Schlüsselgruppen (empfohlen) und AWS-Konten

Um signierte URLs oder signierte Cookies verwenden zu können, benötigen Sie einen Unterzeichner. Ein Unterzeichner ist entweder eine vertrauenswürdige Schlüsselgruppe, in der Sie erstellen CloudFront, oder eine, AWS-Konto die ein CloudFront key pair enthält. Es wird empfohlen, vertrauenswürdige Schlüsselgruppen zu verwenden, und zwar aus den folgenden Gründen:

- Bei CloudFront Schlüsselgruppen müssen Sie nicht den Root-Benutzer des AWS Kontos verwenden, um die öffentlichen Schlüssel für CloudFront signierte URLs und signierte Cookies zu verwalten. AWS Bewährte Methoden empfehlen, den Root-Benutzer nicht zu verwenden, wenn dies nicht erforderlich ist.
- Mit CloudFront Schlüsselgruppen können Sie öffentliche Schlüssel, Schlüsselgruppen und vertrauenswürdige Unterzeichner mithilfe der CloudFront API verwalten. Sie können die API verwenden, um die Schlüsselerstellung und die Schlüsselrotation zu automatisieren. Wenn Sie den AWS Root-Benutzer verwenden, müssen Sie den AWS Management Console zur Verwaltung von CloudFront Schlüsselpaaren verwenden, sodass Sie den Vorgang nicht automatisieren können.
- Da Sie Schlüsselgruppen mit der CloudFront API verwalten können, können Sie auch AWS Identity and Access Management (IAM) -Berechtigungsrichtlinien verwenden, um einzuschränken, was verschiedene Benutzer tun dürfen. Beispielsweise können Sie Benutzern erlauben, öffentliche Schlüssel hochzuladen, aber nicht zu löschen. Oder Sie können Benutzern erlauben, öffentliche Schlüssel zu löschen, jedoch nur dann, wenn bestimmte Bedingungen erfüllt sind, z. B. Multi-Factor-Authentication, das Senden der Anforderung aus einem bestimmten Netzwerk oder das Senden der Anforderung innerhalb eines bestimmten Datums- und Uhrzeitbereichs.
- Mit CloudFront Schlüsselgruppen können Sie Ihrer CloudFront Distribution eine höhere Anzahl von öffentlichen Schlüsseln zuordnen, was Ihnen mehr Flexibilität bei der Verwendung und Verwaltung der öffentlichen Schlüssel bietet. Standardmäßig können Sie bis zu vier Schlüsselgruppen einer einzelnen Verteilung zuordnen, und Sie können bis zu fünf öffentliche Schlüssel in einer Schlüsselgruppe haben.

  Wenn Sie den Root-Benutzer für das AWS Konto verwenden, um CloudFront Schlüsselpaare zu verwalten, können Sie nur bis zu zwei aktive CloudFront Schlüsselpaare pro AWS Konto haben.

Erstellen Sie Schlüsselpaare für Ihre Unterzeichner

Jeder Unterzeichner, den Sie zum Erstellen CloudFront signierter URLs oder signierter Cookies verwenden, muss über ein öffentlich-privates key pair verfügen. Der Unterzeichner verwendet seinen

privaten Schlüssel, um die URL oder die Cookies zu signieren, und CloudFront verwendet den öffentlichen Schlüssel, um die Signatur zu verifizieren.

Die Art und Weise, wie Sie ein key pair erstellen, hängt davon ab, ob Sie eine vertrauenswürdige Schlüsselgruppe als Unterzeichner (empfohlen) oder ein CloudFront key pair verwenden. Weitere Informationen finden Sie in den folgenden Abschnitten. Das Schlüsselpaar, das Sie erstellen, muss die folgenden Anforderungen erfüllen:

- Es muss ein SSH-2 RSA-Schlüsselpaar sein.
- Es muss im base64-kodierten PEM-Format vorliegen.
- Es muss ein 2048-Bit-Schlüsselpaar sein.

Um Ihre Anwendungen zu schützen, empfehlen wir Ihnen, Schlüsselpaare regelmäßig zu rotieren. Weitere Informationen finden Sie unter Rotieren von Schlüsselpaaren.

Erstellen eines Schlüsselpaars für eine vertrauenswürdige Schlüsselgruppe (empfohlen)

Führen Sie die folgenden Schritte aus, um ein Schlüsselpaar für eine vertrauenswürdige Schlüsselgruppe zu erstellen:

1. Erstellen Sie das öffentliche-private Schlüsselpaar.
2. Laden Sie den öffentlichen Schlüssel auf CloudFront hoch.
3. Fügen Sie den öffentlichen Schlüssel einer CloudFront Schlüsselgruppe hinzu.

Weitere Informationen finden Sie in den folgenden Verfahren.

Erstellen eines Schlüsselpaares

> **Note**
>
> In den folgenden Schritten wird OpenSSL als Beispiel für eine Möglichkeit verwendet, ein Schlüsselpaar zu erstellen. Es gibt viele andere Möglichkeiten, ein RSA-Schlüsselpaar zu erstellen.

1. Der folgende Beispielbefehl verwendet OpenSSL, um ein RSA-Schlüsselpaar mit einer Länge von 2048 Bit zu generieren und in der Datei mit dem Namen `private_key.pem` zu speichern.

```
openssl genrsa -out private_key.pem 2048
```

2. Die erstellte Datei enthält den öffentlichen und den privaten Schlüssel. Der folgende Beispielbefehl extrahiert den öffentlichen Schlüssel aus der Datei mit dem Namen `private_key.pem`.

   ```
   openssl rsa -pubout -in private_key.pem -out public_key.pem
   ```

   Sie laden den öffentlichen Schlüssel (in der `public_key.pem`-Datei) später im folgenden Verfahren hoch.

Um den öffentlichen Schlüssel hochzuladen CloudFront

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsmenü die Option Public keys (Öffentliche Schlüssel).
3. Wählen Sie Öffentlichen Schlüssel erstellen.
4. Gehen Sie im Fenster Öffentlichen Schlüssel erstellen wie folgt vor:

   a. Geben Sie unter Key name (Schlüsselname) einen Namen ein, um den öffentlichen Schlüssel zu identifizieren.

   b. Fügen Sie unter Key value (Schlüsselwert) den öffentlichen Schlüssel ein. Wenn Sie die Schritte im vorangegangenen Verfahren ausgeführt haben, befindet sich der öffentliche Schlüssel in der Datei mit dem Namen `public_key.pem`. Um den Inhalt des öffentlichen Schlüssels zu kopieren und einzufügen, können Sie Folgendes tun:

      - Verwenden Sie den cat-Befehl in der macOS- oder Linux-Befehlszeile wie folgt:

        ```
        cat public_key.pem
        ```

        Kopieren Sie die Ausgabe dieses Befehls und fügen Sie sie dann in das Feld Key value (Schlüsselwert) ein.

- Öffnen Sie die `public_key.pem` Datei mit einem Klartext-Editor wie Notepad (unter Windows) oder TextEdit (unter macOS). Kopieren Sie den Inhalt der Datei und fügen Sie ihn dann in das Feld key value (Schlüsselwert) ein.

c. (Optional) Fügen Sie unter Comment (Kommentar) einen Kommentar hinzu, um den öffentlichen Schlüssel zu beschreiben.

Wenn Sie fertig sind, wählen Sie Add (Hinzufügen).

5. Notieren Sie die ID des öffentlichen Schlüssels. Sie verwenden ihn später, wenn Sie signierte URLs oder signierte Cookies erstellen, als Wert für das Feld. `Key-Pair-Id`

So fügen Sie den öffentlichen Schlüssel zu einer Schlüsselgruppe hinzu:

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsmenü die Option Key groups (Schlüsselgruppen).
3. Wählen Sie Add key group (Schlüsselgruppe hinzufügen).
4. Führen Sie auf der Seite Create key group (Schlüsselgruppe erstellen) die folgenden Schritte aus:

   a. Geben Sie unter Key group name (Schlüsselgruppenname) einen Namen ein, um die Schlüsselgruppe zu identifizieren.

   b. (Optional) Geben Sie unter Comment (Kommentar) einen Kommentar ein, um die Schlüsselgruppe zu beschreiben.

   c. Wählen Sie für Public keys (Öffentliche Schlüssel) den öffentlichen Schlüssel aus, der der Schlüsselgruppe hinzugefügt werden soll. Wählen Sie dann Add (Hinzufügen) aus. Wiederholen Sie diesen Schritt für jeden öffentlichen Schlüssel, den Sie der Schlüsselgruppe hinzufügen möchten.
5. Wählen Sie Create key group (Schlüsselgruppe erstellen) aus.
6. Notieren Sie den Namen der Schlüsselgruppe. Sie verwenden es später, um die Schlüsselgruppe einem Cache-Verhalten in einer CloudFront Distribution zuzuordnen. (In der CloudFront API verwenden Sie die Schlüsselgruppen-ID, um die Schlüsselgruppe einem Cache-Verhalten zuzuordnen.)

Ein CloudFront key pair erstellen (nicht empfohlen, erfordert den AWS-Konto Root-Benutzer)

> **Important**
>
> Es wird empfohlen, einen öffentlichen Schlüssel für eine vertrauenswürdige Schlüsselgruppe zu erstellen, anstatt diese Schritte auszuführen. Die empfohlene Methode zum Erstellen öffentlicher Schlüssel für signierte URLs und signierte Cookies finden Sie unterErstellen eines Schlüsselpaars für eine vertrauenswürdige Schlüsselgruppe (empfohlen).

Sie können ein CloudFront key pair auf folgende Weise erstellen:

- Erstellen Sie ein key pair in der AWS Management Console und laden Sie den privaten Schlüssel herunter. Weitere Informationen finden Sie im folgenden Verfahren.
- Erstellen Sie mithilfe einer Anwendung wie OpenSSL ein RSA-Schlüsselpaar und laden Sie den öffentlichen Schlüssel auf die AWS Management Console hoch. Weitere Informationen zum Erstellen eines RSA-Schlüsselpaars finden Sie unter Erstellen eines Schlüsselpaars für eine vertrauenswürdige Schlüsselgruppe (empfohlen).

Um CloudFront Schlüsselpaare zu erstellen in AWS Management Console

1. Melden Sie sich AWS Management Console mit den Anmeldeinformationen des Root-Benutzers des AWS Kontos an.

   > **Important**
   >
   > IAM-Benutzer können keine CloudFront Schlüsselpaare erstellen. Sie müssen sich mit Stammbenutzer-Anmeldeinformationen anmelden, um Schlüsselpaare zu erstellen.

2. Wählen Sie Ihren Kontonamen und dann My Security Credentials (Meine Sicherheitsanmeldeinformationen) aus.
3. Wählen Sie CloudFront Schlüsselpaare aus.
4. Vergewissern Sie sich, dass Sie nicht mehr als ein aktives Schlüsselpaar haben. Sie können kein Schlüsselpaar erstellen, wenn Sie bereits zwei aktive Schlüsselpaare haben.
5. Wählen Sie Create a new key pair (Neues Schlüsselpaar erstellen) aus.

> **Note**
>
> Sie können sich auch dafür entscheiden, Ihr eigenes key pair zu erstellen und den öffentlichen Schlüssel hochzuladen. CloudFront Schlüsselpaare unterstützen 1024-, 2048- oder 4096-Bit-Schlüssel.

6. Wählen Sie im Dialogfeld Create Key Pair (Schlüsselpaar erstellen) die Option Download Private Key File (Private Schlüsseldatei herunterladen) und speichern Sie die Datei dann auf Ihrem Computer.

> **Important**
>
> Speichern Sie den privaten Schlüssel für Ihr CloudFront key pair an einem sicheren Ort und legen Sie die Berechtigungen für die Datei fest, sodass nur die gewünschten Administratoren sie lesen können. Wenn jemand Ihren privaten Schlüssel erhält, kann er gültige signierte URLs und signierte Cookies generieren und Ihre Inhalte herunterladen. Sie können den privaten Schlüssel nicht erneut abrufen. Wenn Sie ihn also verlieren oder löschen, müssen Sie ein neues CloudFront key pair erstellen.

7. Notieren Sie die Schlüsselpaar-ID für Ihr Schlüsselpaar. (In der AWS Management Console wird dies als Zugriffsschlüssel-ID bezeichnet.) Sie werden sie verwenden, wenn Sie signierte URLs oder signierte Cookies erstellen.

Formatieren Sie den privaten Schlüssel neu (nur .NET und Java)

Wenn Sie.NET oder Java verwenden, um signierte URLs oder signierte Cookies zu erstellen, können Sie den privaten Schlüssel aus Ihrem key pair nicht im Standard-PEM-Format verwenden, um die Signatur zu erstellen. Führen Sie stattdessen die folgenden Schritte aus:

- .NET Framework – Konvertieren Sie den privaten Schlüssel in das vom .NET Framework verwendete XML-Format. Es sind mehrere Tools verfügbar.
- Java – Konvertieren Sie den privaten Schlüssel in das DER-Format. Eine Möglichkeit, dies zu tun, ist der folgende OpenSSL-Befehl. Im folgenden Befehl ist `private_key.pem` der Name der Datei, die den privaten Schlüssel im PEM-Format enthält, und `private_key.der` der Name der Datei, die den privaten Schlüssel im DER-Format enthält, nachdem Sie den Befehl ausgeführt haben.

```
openssl pkcs8 -topk8 -nocrypt -in private_key.pem -inform PEM -out private_key.der -
outform DER
```

Um sicherzustellen, dass der Encoder korrekt funktioniert, fügen Sie Ihrem Projekt das JAR für die Bouncy Castle-Java-Kryptografie APIs hinzu und fügen Sie dann den Bouncy Castle-Anbieter hinzu.

Fügen Sie einer Distribution einen Unterzeichner hinzu

Ein Unterzeichner ist die vertrauenswürdige Schlüsselgruppe (empfohlen) oder das CloudFront key pair, das signierte URLs und signierte Cookies für eine Distribution erstellen kann. Um signierte URLs oder signierte Cookies mit einer CloudFront Distribution zu verwenden, müssen Sie einen Unterzeichner angeben.

Aussteller sind mit Cache-Verhaltensweisen verknüpft. Auf diese Weise können Sie signierte URLs oder signierte Cookies für einige Dateien und nicht für andere in derselben Distribution verlangen. Für eine Distribution sind signierte URLs oder Cookies nur für Dateien erforderlich, die dem entsprechenden Cache-Verhalten zugeordnet sind.

Ebenso kann ein Unterzeichner nur für Dateien signieren URLs oder Cookies verwenden, die mit den entsprechenden Cache-Verhaltensweisen verknüpft sind. Wenn Sie beispielsweise einen Unterzeichner für ein Cache-Verhalten und einen anderen Unterzeichner für ein anderes Cache-Verhalten haben, kann keiner der Unterzeichner signierte URLs oder Cookies für Dateien erstellen, die mit dem anderen Cache-Verhalten verknüpft sind.

Important

Bevor Sie einen Aussteller zu Ihrer Verteilung hinzufügen, gehen Sie wie folgt vor:

- Definieren Sie die Pfadmuster in den Cache-Verhaltensweisen und die Reihenfolge der Cache-Verhaltensweisen sorgfältig, damit Sie Benutzern keinen unbeabsichtigten Zugriff auf Ihre Inhalte gewähren oder verhindern, dass sie auf Inhalte zugreifen, die für alle verfügbar sein sollen.

  Nehmen wir beispielsweise an, eine Anfrage stimmt mit dem Pfadmuster für zwei Cache-Verhalten überein. Für das erste Cache-Verhalten sind keine signierten URLs oder signierten Cookies erforderlich, für das zweite Cache-Verhalten schon. Benutzer können auf die Dateien zugreifen, ohne signierte URLs oder signierte Cookies zu verwenden, da das Cache-Verhalten CloudFront verarbeitet wird, das mit dem ersten Treffer verknüpft ist.

Weitere Informationen zu Pfadmustern finden Sie unter Pfadmuster.

- Stellen Sie bei einer Distribution, die Sie bereits zum Verteilen von Inhalten verwenden, sicher, dass Sie bereit sind, signierte URLs und signierte Cookies zu generieren, bevor Sie einen Unterzeichner hinzufügen. Wenn Sie einen Unterzeichner hinzufügen, CloudFront lehnt Anfragen ab, die keine gültige signierte URL oder kein signiertes Cookie enthalten.

Sie können Ihrer Distribution Unterzeichner entweder über die CloudFront Konsole oder die API hinzufügen. CloudFront

Console

Die folgenden Schritte zeigen, wie Sie eine vertrauenswürdige Schlüsselgruppe als Aussteller hinzufügen. Sie können auch einen AWS-Konto Unterzeichner als vertrauenswürdigen Unterzeichner hinzufügen, dies wird jedoch nicht empfohlen.

So fügen Sie einen Aussteller mit der Konsole zu einer Verteilung hinzu:

1. Notieren Sie die Schlüsselgruppen-ID der Schlüsselgruppe, die Sie als vertrauenswürdigen Aussteller verwenden möchten. Weitere Informationen finden Sie unter Erstellen eines Schlüsselpaars für eine vertrauenswürdige Schlüsselgruppe (empfohlen).
2. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
3. Wählen Sie die Distribution aus, deren Dateien Sie mit signierten URLs oder signierten Cookies schützen möchten.

   > **Note**
   >
   > Wenn Sie einer neuen Verteilung einen Aussteller hinzufügen möchten, geben Sie dieselben Einstellungen an, die in Schritt 6 beschrieben werden, wenn Sie die Verteilung erstellen.

4. Wählen Sie die Registerkarte Behaviors aus.
5. Wählen Sie das Cache-Verhalten aus, dessen Pfadmuster den Dateien entspricht, die Sie mit signierten URLs oder signierten Cookies schützen möchten, und wählen Sie dann Bearbeiten aus.
6. Führen Sie auf der Seite Edit Behavior (Verhalten bearbeiten) die folgenden Schritte aus:

a. Wählen Sie für „Zuschauerzugriff einschränken“ (signierte URLs oder signierte Cookies verwenden) die Option Ja aus.

b. Wählen Sie in Trusted Key Groups or Trusted Signer (Vertrauenswürdige Schlüsselgruppen oder vertrauenswürdiger Aussteller) die Option Trusted Key Groups (Vertrauenswürdige Schlüsselgruppen) aus.

c. Wählen Sie unter Trusted Key Groups (Vertrauenswürdige Schlüsselgruppen) die hinzuzufügende Schlüsselgruppe aus, und wählen Sie dann Add (Hinzufügen). Wiederholen Sie diesen Vorgang, wenn Sie mehr als eine Schlüsselgruppe hinzufügen möchten.

7. Wählen Sie Yes, Edit (Ja, Bearbeiten), um das Cache-Verhalten zu aktualisieren.

API

Sie können die CloudFront API verwenden, um eine vertrauenswürdige Schlüsselgruppe als Unterzeichner hinzuzufügen. Sie können einen Aussteller zu einer vorhandenen oder zu einer neuen Verteilung hinzufügen. Geben Sie in beiden Fällen die entsprechenden Werte im `TrustedKeyGroups`-Element an.

Sie können auch einen AWS-Konto als vertrauenswürdigen Unterzeichner hinzufügen, dies wird jedoch nicht empfohlen.

Weitere Informationen finden Sie in der Amazon CloudFront API-Referenz zu den folgenden Themen:

- Eine bestehende Distribution aktualisieren — UpdateDistribution
- Eine neue Distribution erstellen — CreateDistribution

Rotieren von Schlüsselpaaren

Wir empfehlen Ihnen, Ihre Schlüsselpaare für signierte URLs und signierte Cookies regelmäßig zu wechseln (zu ändern). Gehen Sie wie folgt vor, um Schlüsselpaare zu wechseln, die Sie verwenden, um signierte URLs oder signierte Cookies zu erstellen, ohne dass sie ungültig werden, URLs oder um Cookies, die noch nicht abgelaufen sind, zu erstellen:

1. Erstellen Sie ein neues Schlüsselpaar, und fügen Sie den öffentlichen Schlüssel zu einer Schlüsselgruppe hinzu. Weitere Informationen finden Sie unter Erstellen eines Schlüsselpaars für eine vertrauenswürdige Schlüsselgruppe (empfohlen).

2. Wenn Sie im vorherigen Schritt eine neue Schlüsselgruppe erstellt haben, fügen Sie die Schlüsselgruppe der Verteilung als Aussteller hinzu.

> **Important**
>
> Entfernen Sie noch keine vorhandenen öffentlichen Schlüssel aus der Schlüsselgruppe oder Schlüsselgruppen aus der Verteilung. Fügen Sie nur die neuen hinzu.

3. Aktualisieren Sie Ihre Anwendung so, dass Signaturen mithilfe der privaten Schlüssel von den neuen Schlüsselpaaren erstellt werden. Vergewissern Sie sich, dass die signierten Cookies URLs oder Cookies, die mit den neuen privaten Schlüsseln signiert wurden, funktionieren.
4. Warten Sie, bis das Ablaufdatum abgelaufen ist URLs oder bis Cookies, die mit dem vorherigen privaten Schlüssel signiert wurden, abgelaufen sind. Entfernen Sie dann den alten öffentlichen Schlüssel aus der Schlüsselgruppe. Wenn Sie in Schritt 2 eine neue Schlüsselgruppe erstellt haben, entfernen Sie die alte Schlüsselgruppe aus Ihrer Verteilung.

Entscheiden Sie sich dafür, signierte URLs oder signierte Cookies zu verwenden

CloudFront signierte URLs und signierte Cookies bieten dieselbe grundlegende Funktionalität: Sie ermöglichen es Ihnen zu kontrollieren, wer auf Ihre Inhalte zugreifen kann. Wenn Sie private Inhalte bereitstellen möchten CloudFront und sich entscheiden möchten, ob Sie signierte URLs oder signierte Cookies verwenden möchten, sollten Sie Folgendes beachten.

Verwenden Sie signierte Dateien URLs in den folgenden Fällen:

- Sie möchten den Zugriff auf einzelne Dateien einschränken, z. B. einen Installations-Download für Ihre Anwendung.
- Ihre Benutzer verwenden einen Client (z. B. einen benutzerdefinierten HTTP-Client), der keine Cookies unterstützt.

Verwenden Sie signierte Cookies in den folgenden Fällen:

- Sie möchten Zugriff auf mehrere beschränkte Dateien bereitstellen, z. B. auf alle Dateien für ein Video im HLS-Format oder alle Dateien im Bereich des Abonnenten einer Website.
- Sie möchten Ihre aktuelle Version nicht ändern URLs.

Wenn Sie derzeit keine URLs signierten Cookies verwenden und Ihr (unsigned) einen der folgenden Parameter für die Abfragezeichenfolge URLs enthält, können Sie URLs weder signierte noch signierte Cookies verwenden:

- `Expires`
- `Policy`
- `Signature`
- `Key-Pair-Id`

CloudFront geht davon aus URLs , dass die Parameter, die einen dieser Abfragezeichenfolgen enthalten URLs, signiert sind, und betrachtet daher signierte Cookies nicht.

Verwenden Sie sowohl signierte als URLs auch signierte Cookies

Signierte Cookies haben URLs Vorrang vor signierten Cookies. Wenn Sie sowohl signierte URLs als auch signierte Cookies verwenden, um den Zugriff auf dieselben Dateien zu kontrollieren, und ein Betrachter eine signierte URL verwendet, um eine Datei anzufordern, CloudFront wird nur anhand der signierten URL bestimmt, ob die Datei an den Betrachter zurückgegeben werden soll.

Verwenden Sie signierte URLs

Eine signierte URL enthält zusätzliche Informationen, wie z. B. Ablaufdatum und -zeit, mit denen Sie den Zugriff auf Ihre Inhalte besser kontrollieren können. Diese zusätzlichen Informationen sind in einer Richtlinienanweisung enthalten, die entweder auf einer vordefinierten oder einer benutzerdefinierten Richtlinie basieren. Die Unterschiede zwischen vordefinierten und benutzerdefinierten Richtlinien sind in den nächsten beiden Abschnitten beschrieben.

> **Note**
>
> Sie können einige signierte URLs mithilfe vordefinierter Richtlinien und einige signierte URLs mithilfe benutzerdefinierter Richtlinien für dieselbe Distribution erstellen.

Themen

- Entscheiden Sie, wie lange Unterschriften gültig URLs sind
- Wann CloudFront überprüft das Ablaufdatum und die Uhrzeit in einer signierten URL
- Beispiel-Code und Drittanbieter-Tools
- Erstellen Sie eine signierte URL mithilfe einer vordefinierten Richtlinie
- Erstellen Sie eine signierte URL mithilfe einer benutzerdefinierten Richtlinie

Entscheiden Sie sich dafür, vordefinierte oder benutzerdefinierte Richtlinien für signierte Richtlinien zu verwenden URLs

Wenn Sie eine signierte URL erstellen, schreiben Sie eine Richtlinienanweisung im JSON-Format, welche die Einschränkungen für die signierte URL festlegt, z. B. wie lange die URL gültig ist. Sie können entweder eine vordefinierte Richtlinie oder eine benutzerdefinierte Richtlinie verwenden. Im Folgenden finden Sie einen Vergleich zwischen vordefinierten und benutzerdefinierten Richtlinien:

Beschreibung	Vordefinierte Richtlinie	Benutzerdefinierte Richtlinie
Sie können die Richtlinienanweisung für mehrere Dateien wiederverwenden. Um die Richtlini enanweisung wiederzuverwenden, müssen Sie Platzhalterzeichen im `Resource`-Objekt verwenden . Weitere Informationen finden Sie unter Werte, die Sie in der Richtlinienanweisung für eine signierte URL angeben, die eine benutzerdefinierte Richtlinie verwendet.)	Nein	Ja
Sie können das Datum und die Zeit festlegen, ab denen Benutzer auf Ihre Inhalte zugreifen können.	Nein	Ja (optional)
Sie können das Datum und die Zeit festlegen, ab denen Benutzer nicht mehr auf Ihre Inhalte zugreifen können.	Ja	Ja
Sie können die IP-Adresse oder den Bereich von IP-Adressen der Benutzer festlegen, die auf Ihre Inhalte zugreifen können.	Nein	Ja (optional)

Beschreibung	Vordefinierte Richtlinie	Benutzerdefinierte Richtlinie
Die signierte URL enthält eine Base64-codierte Version der Richtlinie, was zu einer längeren URL führt.	Nein	Ja

Informationen zum Erstellen signierter Richtlinien URLs mithilfe einer vordefinierten Richtlinie finden Sie unterErstellen Sie eine signierte URL mithilfe einer vordefinierten Richtlinie.

Informationen zum Erstellen signierter Dateien URLs mithilfe einer benutzerdefinierten Richtlinie finden Sie unterErstellen Sie eine signierte URL mithilfe einer benutzerdefinierten Richtlinie.

Wie URLs funktionieren signierte

Hier finden Sie eine Übersicht darüber, wie Sie Amazon S3 für signiert konfigurieren CloudFront URLs und wie CloudFront reagiert, wenn ein Benutzer eine signierte URL verwendet, um eine Datei anzufordern.

1. Geben Sie in Ihrer CloudFront Distribution eine oder mehrere vertrauenswürdige Schlüsselgruppen an, die die öffentlichen Schlüssel enthalten, die zur Überprüfung der URL-Signatur verwendet werden CloudFront können. Sie verwenden die entsprechenden privaten Schlüssel, um die zu signieren URLs.

   Weitere Informationen finden Sie unter Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können.
2. Entwickeln Sie Ihre Anwendung, um zu ermitteln, ob ein Benutzer Zugriff auf Ihre Inhalte haben soll, und URLs um signierte Dateien oder Teile Ihrer Anwendung zu erstellen, auf die Sie den Zugriff einschränken möchten. Weitere Informationen finden Sie unter den folgenden Themen:

   - Erstellen Sie eine signierte URL mithilfe einer vordefinierten Richtlinie
   - Erstellen Sie eine signierte URL mithilfe einer benutzerdefinierten Richtlinie
3. Ein Benutzer fordert eine Datei an, für die Sie eine Signatur benötigen möchten URLs.
4. Ihre Anwendung stellt sicher, dass der Benutzer zum Zugriff auf die Datei berechtigt ist: Er hat sich angemeldet, für den Zugriff auf die Inhalte bezahlt oder andere Anforderungen für den Zugriff erfüllt.
5. Ihre Anwendung erstellt eine signierte URL und gibt diese an den Benutzer zurück.

6. Über die signierte URL kann der Benutzer die Inhalte herunterladen oder streamen.

   Dieser Schritt erfolgt automatisch. Der Benutzer muss in der Regel keine zusätzlichen Schritte ausführen, um auf den Inhalt zuzugreifen. Wenn ein Benutzer beispielsweise in einem Web-Browser auf Ihre Inhalte zugreift, gibt Ihre Anwendung die signierte URL an den Browser zurück. Der Browser verwendet sofort die signierte URL, um auf die Datei im CloudFront Edge-Cache zuzugreifen, ohne dass der Benutzer eingreifen muss.

7. CloudFront verwendet den öffentlichen Schlüssel, um die Signatur zu validieren und zu bestätigen, dass die URL nicht manipuliert wurde. Wenn die Signatur ungültig ist, wird die Anfrage abgelehnt.

   Wenn die Signatur gültig ist, überprüft CloudFront die Richtlinienanweisung in der URL (oder erstellt eine, wenn Sie eine vordefinierte Richtlinie verwenden), um zu bestätigen, dass die Anfrage noch gültig ist. Wenn Sie beispielsweise ein Anfangs- und Enddatum und eine Uhrzeit für die URL angegeben haben, wird CloudFront bestätigt, dass der Benutzer versucht, während des Zeitraums, für den Sie den Zugriff zulassen möchten, auf Ihre Inhalte zuzugreifen.

   Wenn die Anforderung die Anforderungen in der Richtlinienerklärung CloudFront erfüllt, werden die Standardoperationen ausgeführt: Ermittelt, ob sich die Datei bereits im Edge-Cache befindet, leitet die Anforderung gegebenenfalls an den Ursprung weiter und gibt die Datei an den Benutzer zurück.

> **Note**
>
> Wenn eine unsignierte URL Abfragezeichenfolgenparameter enthält, stellen Sie sicher, dass Sie diese in den Teil der URL einschließen, den Sie signieren. Wenn Sie einer signierten URL nach der Erstellung eine Abfragezeichenfolge hinzufügen, gibt die URL einen HTTP 403-Status zurück.

Entscheiden Sie, wie lange Unterschriften gültig URLs sind

Sie können private Inhalte mithilfe einer signierten URL verteilen, die nur für einen kurzen Zeitraum gültig ist – vielleicht nur für ein paar Minuten. Signierte URLs , die für einen so kurzen Zeitraum gültig sind, eignen sich gut für die Verteilung von Inhalten on-the-fly an einen Benutzer zu einem bestimmten Zweck, z. B. zur Verteilung von Leihfilmen oder Musikdownloads an Kunden auf Abruf. Wenn URLs Ihre signierten Dokumente nur für einen kurzen Zeitraum gültig sind, möchten Sie

sie wahrscheinlich automatisch mit einer von Ihnen entwickelten Anwendung generieren. Wenn der Benutzer beginnt, eine Datei herunterzuladen oder eine Mediendatei abzuspielen, CloudFront vergleicht er die Ablaufzeit in der URL mit der aktuellen Uhrzeit, um festzustellen, ob die URL noch gültig ist.

Sie können private Inhalte auch mithilfe einer signierten URL verteilen, die für einen längeren Zeitraum gültig ist – vielleicht für viele Jahre. Signierte URLs , die für einen längeren Zeitraum gültig sind, sind nützlich, um private Inhalte an bekannte Benutzer zu verteilen, z. B. um einen Geschäftsplan an Investoren zu verteilen oder Schulungsmaterial an Mitarbeiter zu verteilen. Sie können eine Anwendung entwickeln, mit der diese längerfristigen signierten Dateien URLs für Sie generiert werden.

Wann CloudFront überprüft das Ablaufdatum und die Uhrzeit in einer signierten URL

CloudFront überprüft das Ablaufdatum und die Uhrzeit in einer signierten URL zum Zeitpunkt der HTTP-Anfrage. Wenn ein Client unmittelbar vor der Ablaufzeit mit dem Download einer großen Datei beginnt, sollte der Download abgeschlossen werden, auch wenn die Ablaufzeit während des Downloads überschritten wird. Wenn die TCP-Verbindung getrennt wird und der Client nach Überschreitung der Ablaufzeit versucht, den Download erneut zu starten, schlägt der Download fehl.

Wenn ein Client Range verwendet GETs , um eine Datei in kleineren Teilen abzurufen, schlägt jede GET-Anforderung fehl, die nach Ablauf der Ablaufzeit erfolgt. Weitere Informationen zu Range GETs finden Sie unterWie CloudFront werden Teilanfragen für ein Objekt (BereichGETs) verarbeitet.

Beispiel-Code und Drittanbieter-Tools

Beispielcode, der den Hash-Teil und den signierten Teil von signed erstellt URLs, finden Sie in den folgenden Themen:

- Erstellen einer URL-Signatur mit Perl
- Erstellen einer URL-Signatur mit PHP
- Erstellen einer URL-Signatur mithilfe von C# und dem .NET Framework
- Erstellen einer URL-Signatur mit Java

Erstellen Sie eine signierte URL mithilfe einer vordefinierten Richtlinie

Führen Sie die folgenden Schritte aus, um eine signierte URL mit einer vordefinierten Richtlinie zu erstellen.

So erstellen Sie eine signierte URL mit einer vordefinierten Richtlinie

1. Wenn Sie.NET oder Java verwenden, um signierte zu erstellen URLs, und wenn Sie den privaten Schlüssel für Ihr key pair nicht vom standardmäßigen .pem-Format in ein mit .NET oder Java kompatibles Format umformatiert haben, tun Sie dies jetzt. Weitere Informationen finden Sie unter Formatieren Sie den privaten Schlüssel neu (nur .NET und Java).
2. Verketten Sie die folgenden Werte. Sie können das Format in diesem Beispiel für eine signierte URL verwenden.

```
https://d111111abcdef8.cloudfront.net/
image.jpg?color=red&size=medium&Expires=1357034400&Signature=nitfHRCrtziwO2HwPfWw~yYDhUF5Ew
j19DzZrvDh6hQ73lDx~-ar3UocvvRQVw6EkC~GdpGQyyOSKQim-
TxAnW7d8F5Kkai9HVx0FIu-5jcQb0UEmatEXAMPLE3ReXySpLSMj0yCd3ZAB4UcBCAqEijkytL6f3fVYNGQI6&Key-
Pair-Id=K2JCJMDEHXQW5F
```

   Entfernen Sie alle Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen). Möglicherweise müssen Sie in der Zeichenfolge im Anwendungscode Escape-Zeichen einfügen. Alle Werte haben den Typ. `String`

   1. ***Base URL for the file***

      Die Basis-URL ist die CloudFront URL, die Sie für den Zugriff auf die Datei verwenden würden, wenn Sie keine signierten verwenden würden URLs, einschließlich Ihrer eigenen Abfragezeichenfolge-Parameter, falls vorhanden. Im vorherigen Beispiel lautet die Basis-URL`https://d111111abcdef8.cloudfront.net/image.jpg`. Weitere Hinweise zum Format von URLs für Distributionen finden Sie unterPassen Sie das URL-Format für Dateien an in CloudFront.

      - Die folgende CloudFront URL bezieht sich auf eine Bilddatei in einer Distribution (unter Verwendung des CloudFront Domainnamens). Beachten Sie, dass `image.jpg` ein `images`-Verzeichnis ist. Der Pfad zur Datei in der URL muss mit dem Pfad zur Datei auf Ihrem HTTP-Server oder in Ihrem Amazon S3 Bucket übereinstimmen.

        `https://d111111abcdef8.cloudfront.net/images/image.jpg`

      - Die folgende CloudFront URL enthält eine Abfragezeichenfolge:

        `https://d111111abcdef8.cloudfront.net/images/image.jpg?size=large`

- Die folgenden Informationen CloudFront URLs beziehen sich auf Bilddateien in einer Distribution. Beide verwenden einen alternativen Domainnamen. Die zweite enthält eine Abfragezeichenfolge:

  `https://www.example.com/images/image.jpg`

  `https://www.example.com/images/image.jpg?color=red`

- Die folgende CloudFront URL bezieht sich auf eine Bilddatei in einer Distribution, die einen alternativen Domainnamen und das HTTPS-Protokoll verwendet:

  `https://www.example.com/images/image.jpg`

2. **?**

   Das ? gibt an, dass Abfrageparameter der Basis-URL folgen. Schließen Sie das ein, ? auch wenn Sie keine Abfrageparameter angeben.

   > **Note**
   >
   > Sie können die folgenden Abfrageparameter in beliebiger Reihenfolge angeben.

3. ***Your query string parameters, if any*&**

   (Optional) Sie können Ihre eigenen Abfragezeichenfolgenparameter eingeben. Fügen Sie dazu zwischen jedem Zeichen ein Und-Zeichen (&) hinzu, z. B. `color=red&size=medium` Sie können Abfragezeichenfolgenparameter in beliebiger Reihenfolge innerhalb der URL angeben.

   > **Important**
   >
   > Ihre Abfragezeichenfolgenparameter können nicht mit `Expires`, `Signature`, oder benannt werden`Key-Pair-Id`.

4. **Expires=*date and time in Unix time format (in seconds) and Coordinated Universal Time (UTC)***

   Das Datum und die Uhrzeit, wann die URL den Zugriff auf die Datei nicht mehr zulassen soll.

Geben Sie das Ablaufdatum und die Ablaufzeit im Unix-Zeitformat (in Sekunden) und in koordinierter Weltzeit (UTC) an. Beispiel: Der 1. Januar 2013 um 10:00 Uhr UTC wird `1357034400` in das Unix-Zeitformat konvertiert, wie im Beispiel zu Beginn dieses Themas gezeigt. Um die Epochenzeit zu verwenden, verwenden Sie eine 32-Bit-Ganzzahl für ein Datum, das nicht später als `2147483647` (19. Januar 2038 um 03:14:07 UTC) liegt. Informationen zu UTC finden Sie unter RFC 3339, Datum und Uhrzeit im Internet: Zeitstempel.

5. **`&Signature=`*`hashed and signed version of the policy statement`***

   Eine gehashte, signierte und Base64-codierte Version der JSON-Richtlinienanweisung. Weitere Informationen finden Sie unter Erstellen Sie eine Signatur für eine signierte URL, die eine vordefinierte Richtlinie verwendet.

6. **`&Key-Pair-Id=`*`public key ID for the CloudFront public key whose corresponding private key you're using to generate the signature`***

   Die ID für einen CloudFront öffentlichen Schlüssel, zum Beispiel`K2JCJMDEHXQW5F`. Die ID des öffentlichen Schlüssels gibt an CloudFront , welcher öffentliche Schlüssel zur Validierung der signierten URL verwendet werden soll. CloudFront vergleicht die Informationen in der Signatur mit den Informationen in der Richtlinienerklärung, um sicherzustellen, dass die URL nicht manipuliert wurde.

   Dieser öffentliche Schlüssel muss zu einer Schlüsselgruppe gehören, die ein vertrauenswürdiger Aussteller in der Verteilung ist. Weitere Informationen finden Sie unter Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können.

Erstellen Sie eine Signatur für eine signierte URL, die eine vordefinierte Richtlinie verwendet

Gehen Sie wie folgt vor, um die Signatur für eine signierte URL zu erstellen, die eine vordefinierte Richtlinie verwendet.

Themen

- Erstellen Sie eine Richtlinienerklärung für eine signierte URL, die eine vordefinierte Richtlinie verwendet
- Erstellen Sie eine Signatur für eine signierte URL, die eine vordefinierte Richtlinie verwendet

Erstellen Sie eine Richtlinienerklärung für eine signierte URL, die eine vordefinierte Richtlinie verwendet

Wenn Sie eine signierte URL mit einer vordefinierten Richtlinie erstellen, ist der `Signature`-Parameter eine gehashte und signierte Version einer Richtlinienanweisung. Bei signierten URLs , die eine vordefinierte Richtlinie verwenden, fügen Sie die Richtlinienerklärung nicht in die URL ein, wie dies bei signierten Richtlinien der Fall ist URLs , die eine benutzerdefinierte Richtlinie verwenden. Zum Erstellen der Richtlinienanweisung führen Sie die folgenden Schritte aus.

So erstellen Sie die Richtlinienanweisung für eine signierte URL, die eine vordefinierte Richtlinie verwendet

1. Erstellen Sie die Richtlinienanweisung unter Verwendung des folgenden JSON-Formats und der UTF-8-Zeichencodierung. Fügen Sie alle Satzzeichen und andere Literalwerte genau wie angegeben ein. Informationen zu den Parametern `Resource` und `DateLessThan` finden Sie unter Werte, die Sie in der Richtlinienanweisung für eine signierte URL angeben, die eine vordefinierte Richtlinie verwendet.

   ```
   {
       "Statement": [
           {
               "Resource": "base URL or stream name",
               "Condition": {
                   "DateLessThan": {
                       "AWS:EpochTime": ending date and time in Unix time format and
    UTC
                   }
               }
           }
       ]
   }
   ```

2. Entfernen Sie alle Leerzeichen (einschließlich Tabulatoren und Zeilenumbrüche) aus der Richtlinienerklärung. Möglicherweise müssen Sie in der Zeichenfolge im Anwendungscode Escape-Zeichen einfügen.

Werte, die Sie in der Richtlinienanweisung für eine signierte URL angeben, die eine vordefinierte Richtlinie verwendet

Beim Erstellen einer Richtlinienanweisung für eine vordefinierte Richtlinie geben Sie die folgenden Werte an.

Ressource

> **Note**
>
> Sie können nur einen Wert für `Resource` angeben.

Die Basis-URL, einschließlich Ihrer Abfragezeichenfolgen, falls vorhanden, jedoch ohne die `Key-Pair-Id` Parameter CloudFront `Expires`Signature, und, zum Beispiel:

```
https://d111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

Beachten Sie Folgendes:

- Protokoll – Der Wert muss mit `http://` oder `https://` beginnen.
- Abfragezeichenfolgeparameter – Wenn Sie über keine Abfragezeichenfolgeparameter verfügen, lassen Sie das Fragezeichen weg.
- Alternative Domänennamen – Wenn Sie einen alternativen Domänennamen (CNAME) in der URL angeben, müssen Sie diesen alternativen Domänennamen angeben, wenn Sie auf Ihrer Webseite oder in Ihrer Anwendung auf die Datei verweisen. Geben Sie nicht die Amazon-S3-URL für das Objekt an.

DateLessThan

Das Ablaufdatum und die Ablaufzeit für die URL im Unix-Zeitformat (in Sekunden) und in koordinierter Weltzeit (UTC). Beispielsweise wird der 1. Januar 2013, 10:00 Uhr UTC in 1357034400 im Unix-Zeitformat umgewandelt.

Dieser Wert muss mit dem Wert des `Expires`-Abfragezeichenfolgeparameters in der signierten URL übereinstimmen. Setzen Sie den Wert nicht in Anführungszeichen.

Weitere Informationen finden Sie unter Wann CloudFront überprüft das Ablaufdatum und die Uhrzeit in einer signierten URL.

Beispiel-Richtlinienanweisung für eine signierte URL, die eine vordefinierte Richtlinie verwendet

Wenn Sie die folgende Beispiel-Richtlinienanweisung in einer signierten URL verwenden, kann ein Benutzer bis zum 1. Januar 2013, 10:00 Uhr UTC, auf die Datei `https://d111111abcdef8.cloudfront.net/horizon.jpg` zugreifen:

```
{
    "Statement": [
        {
            "Resource": "https://d111111abcdef8.cloudfront.net/horizon.jpg?
size=large&license=yes",
            "Condition": {
                "DateLessThan": {
                    "AWS:EpochTime": 1357034400
                }
            }
        }
    ]
}
```

Erstellen Sie eine Signatur für eine signierte URL, die eine vordefinierte Richtlinie verwendet

Um den Wert für den Parameter `Signature` in einer signierten URL zu erstellen, müssen Sie die in Erstellen Sie eine Richtlinienerklärung für eine signierte URL, die eine vordefinierte Richtlinie verwendet erstellte Richtlinienanweisung hashen und signieren.

Weitere Informationen und Beispiele für das Hashing, Signieren und Codieren der Richtlinienanweisung finden Sie unter:

- Linux-Befehle und OpenSSL für Base64-Kodierung und Verschlüsselung
- Code-Beispiele für das Erstellen einer Signatur für eine signierte URL

Option 1: So erstellen Sie eine Signatur mithilfe einer vordefinierten Richtlinie

1. Verwenden Sie die SHA-1-Hash-Funktion und RSA, um die im Verfahren So erstellen Sie die Richtlinienanweisung für eine signierte URL, die eine vordefinierte Richtlinie verwendet erstellte Richtlinienanweisung zu hashen und zu signieren. Verwenden Sie die Version der Richtlinienanweisung, die keine Leerzeichen mehr enthält.

Verwenden Sie für den privaten Schlüssel, der für die Hash-Funktion erforderlich ist, einen privaten Schlüssel, dessen öffentlicher Schlüssel sich in einer aktiven vertrauenswürdigen Schlüsselgruppe für die Verteilung befindet.

> **Note**
>
> Die Methode, die Sie zum Hashen und Signieren der Richtlinienanweisung verwenden, ist abhängig von Ihrer Programmiersprache und Plattform. Einen Beispiel-Code finden Sie unter Code-Beispiele für das Erstellen einer Signatur für eine signierte URL.

2. Entfernen Sie Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen) aus der Hash-Zeichenfolge und der signierten Zeichenfolge.
3. Nehmen Sie eine Base64-Codierung der Zeichenfolge mithilfe von MIME-Base64-Codierung vor. Weitere Informationen finden Sie in Abschnitt 6.8, Base64 Content-Transfer-Encoding in RFC 2045, MIME (Multipurpose Internet Mail Extensions), Teil 1: Format von Internet-Nachrichtentexten.
4. Ersetzen Sie Zeichen, die in einer URL-Abfragezeichenfolge nicht gültig sind, durch gültige Zeichen. In der folgenden Tabelle sind ungültige und gültige Zeichen aufgelistet.

Ersetzen Sie diese ungültigen Zeichen	Durch diese gültigen Zeichen
+	- (Bindestrich)
=	_ (Unterstrich)
/	~ (Tilde)

5. Fügen Sie den resultierenden Wert hinter `&Signature=` zu Ihrer signierten URL hinzu und kehren Sie zu So erstellen Sie eine signierte URL mit einer vordefinierten Richtlinie zurück, um das Verketten der Teile Ihrer signierten URL abzuschließen.

Erstellen Sie eine signierte URL mithilfe einer benutzerdefinierten Richtlinie

Gehen Sie wie folgt vor, um eine signierte URL mithilfe einer benutzerdefinierten Richtlinie zu erstellen.

So erstellen Sie eine signierte URL mit einer benutzerdefinierten Richtlinie

1. Wenn Sie.NET oder Java verwenden, um signierte zu erstellen URLs, und wenn Sie den privaten Schlüssel für Ihr key pair nicht vom standardmäßigen .pem-Format in ein mit .NET oder Java kompatibles Format umformatiert haben, tun Sie dies jetzt. Weitere Informationen finden Sie unter Formatieren Sie den privaten Schlüssel neu (nur .NET und Java).
2. Verketten Sie die folgenden Werte. Sie können das Format in diesem Beispiel für eine signierte URL verwenden.

```
https://d111111abcdef8.cloudfront.net/
image.jpg?color=red&size=medium&Policy=eyANCiAgICEXAMPLEW1lbnQiOiBbeyANCiAgICAgICJSZXNvdXJj
j19DzZrvDh6hQ73lDx~-ar3UocvvRQVw6EkC~GdpGQyyOSKQim-
TxAnW7d8F5Kkai9HVx0FIu-5jcQb0UEmatEXAMPLE3ReXySpLSMj0yCd3ZAB4UcBCAqEijkytL6f3fVYNGQI6&Key-
Pair-Id=K2JCJMDEHXQW5F
```

Entfernen Sie alle Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen). Möglicherweise müssen Sie in der Zeichenfolge im Anwendungscode Escape-Zeichen einfügen. Alle Werte haben den Typ. `String`

1. ***Base URL for the file***

   Die Basis-URL ist die CloudFront URL, die Sie für den Zugriff auf die Datei verwenden würden, wenn Sie keine signierten verwenden würden URLs, einschließlich Ihrer eigenen Abfragezeichenfolge-Parameter, falls vorhanden. Im vorherigen Beispiel lautet die Basis-URL`https://d111111abcdef8.cloudfront.net/image.jpg`. Weitere Hinweise zum Format von URLs für Distributionen finden Sie unterPassen Sie das URL-Format für Dateien an in CloudFront.

   Die folgenden Beispiele zeigen Werte, die Sie für Verteilungen angeben.

   - Die folgende CloudFront URL bezieht sich auf eine Bilddatei in einer Distribution (unter Verwendung des CloudFront Domainnamens). Beachten Sie, dass `image.jpg` ein `images`-Verzeichnis ist. Der Pfad zur Datei in der URL muss mit dem Pfad zur Datei auf Ihrem HTTP-Server oder in Ihrem Amazon S3 Bucket übereinstimmen.

     ```
     https://d111111abcdef8.cloudfront.net/images/image.jpg
     ```

   - Die folgende CloudFront URL enthält eine Abfragezeichenfolge:

     ```
     https://d111111abcdef8.cloudfront.net/images/image.jpg?size=large
     ```

- Die folgenden Informationen CloudFront URLs beziehen sich auf Bilddateien in einer Distribution. Beide verwenden einen alternativen Domänennamen; die zweite enthält eine Abfragezeichenfolge:

  `https://www.example.com/images/image.jpg`

  `https://www.example.com/images/image.jpg?color=red`

- Die folgende CloudFront URL bezieht sich auf eine Bilddatei in einer Distribution, die einen alternativen Domainnamen und das HTTPS-Protokoll verwendet:

  `https://www.example.com/images/image.jpg`

2. **?**

   Das ? gibt an, dass die Parameter der Abfragezeichenfolge der Basis-URL folgen. Schließt das ein, ? auch wenn Sie keine Abfrageparameter angeben.

   > **Note**
   >
   > Sie können die folgenden Abfrageparameter in beliebiger Reihenfolge angeben.

3. ***Your query string parameters, if any*&**

   (Optional) Sie können Ihre eigenen Abfragezeichenfolgenparameter eingeben. Fügen Sie dazu zwischen jedem Zeichen ein Und-Zeichen (&) hinzu, z. B. `color=red&size=medium` Sie können Abfragezeichenfolgenparameter in beliebiger Reihenfolge innerhalb der URL angeben.

   > **Important**
   >
   > Ihre Abfragezeichenfolgenparameter können nicht mit `PolicySignature`, oder benannt werden`Key-Pair-Id`.

   Wenn Sie Ihre eigenen Parameter hinzufügen, fügen Sie & nach jedem Parameter, auch nach dem letzten, ein.

4. **`Policy=`*`base64 encoded version of policy statement`***

   Ihre Richtlinienerklärung im JSON-Format, wobei Leerzeichen entfernt und anschließend base64-codiert wurden. Weitere Informationen finden Sie unter Erstellen Sie eine Richtlinienerklärung für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet.

   Die Richtlinienanweisung steuert den Zugriff, den eine signierte URL einem Benutzer gewährt. Sie enthält die URL der Datei, ein Ablaufdatum und eine Uhrzeit, ein optionales Datum und die Uhrzeit, zu der die URL gültig wird, und eine optionale IP-Adresse oder einen Bereich von IP-Adressen, die auf die Datei zugreifen dürfen.

5. **`&Signature=`*`hashed and signed version of the policy statement`***

   Eine gehashte, signierte und Base64-codierte Version der JSON-Richtlinienanweisung. Weitere Informationen finden Sie unter Erstellen Sie eine Signatur für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet.

6. **`&Key-Pair-Id=`*`public key ID for the CloudFront public key whose corresponding private key you're using to generate the signature`***

   Die ID für einen CloudFront öffentlichen Schlüssel, zum Beispiel`K2JCJMDEHXQW5F`. Die ID des öffentlichen Schlüssels gibt an CloudFront, welcher öffentliche Schlüssel zur Validierung der signierten URL verwendet werden soll. CloudFrontvergleicht die Informationen in der Signatur mit den Informationen in der Richtlinienerklärung, um sicherzustellen, dass die URL nicht manipuliert wurde.

   Dieser öffentliche Schlüssel muss zu einer Schlüsselgruppe gehören, die ein vertrauenswürdiger Aussteller in der Verteilung ist. Weitere Informationen finden Sie unter Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können.

Erstellen Sie eine Richtlinienerklärung für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet

Führen Sie zum Erstellen einer Richtlinienanweisung für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet, die folgenden Schritte aus.

Einige Beispiele für Richtlinienanweisungen, die den Zugriff auf Dateien auf verschiedene Weisen kontrollieren, finden Sie unter the section called “Beispiel-Richtlinienanweisungen für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet”.

So erstellen Sie die Richtlinienanweisung für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet

1. Erstellen Sie die Richtlinienanweisung unter Verwendung des folgenden JSON-Formats. Ersetzen Sie die Symbole für „kleiner als“ (<) und „größer als“ (>) und die darin enthaltenen Beschreibungen durch Ihre eigenen Werte. Weitere Informationen finden Sie unter the section called “Werte, die Sie in der Richtlinienanweisung für eine signierte URL angeben, die eine benutzerdefinierte Richtlinie verwendet”.

```
{
    "Statement": [
        {
            "Resource": "<Optional but recommended: URL of the file>",
            "Condition": {
                "DateLessThan": {
                 "AWS:EpochTime": <Required: ending date and time in Unix time
 format and UTC>
                },
                "DateGreaterThan": {
                 "AWS:EpochTime": <Optional: beginning date and time in Unix time
 format and UTC>
                },
                "IpAddress": {
                 "AWS:SourceIp": "<Optional: IP address>"
                }
            }
        }
    ]
}
```

   Beachten Sie Folgendes:

   - Sie können nur eine Anweisung in die Richtlinie aufnehmen.
   - Verwenden Sie UTF-8-Zeichencodierung.
   - Fügen Sie alle Satzzeichen und Parameternamen genau wie angegeben ein. Abkürzungen für Parameternamen werden nicht akzeptiert.
   - Die Reihenfolge der Parameter im Bereich `Condition` ist unerheblich.

- Informationen zu den Werten für `Resource`, `DateLessThan`, `DateGreaterThan` und `IpAddress` finden Sie unter the section called “Werte, die Sie in der Richtlinienanweisung für eine signierte URL angeben, die eine benutzerdefinierte Richtlinie verwendet”.

2. Entfernen Sie alle Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen) aus der Richtlinienerklärung. Möglicherweise müssen Sie in der Zeichenfolge im Anwendungscode Escape-Zeichen einfügen.
3. Nehmen Sie eine Base64-Codierung der Richtlinienanweisung mithilfe von MIME-Base64-Codierung vor. Weitere Informationen finden Sie in Abschnitt 6.8, Base64 Content-Transfer-Encoding in RFC 2045, MIME (Multipurpose Internet Mail Extensions), Erster Teil: Format von Internet-Nachrichtentexten.
4. Ersetzen Sie Zeichen, die in einer URL-Abfragezeichenfolge nicht gültig sind, durch gültige Zeichen. In der folgenden Tabelle sind ungültige und gültige Zeichen aufgelistet.

Ersetzen Sie diese ungültigen Zeichen	Durch diese gültigen Zeichen
+	- (Bindestrich)
=	_ (Unterstrich)
/	~ (Tilde)

5. Fügen Sie den resultierenden Wert hinter zu Ihrer signierten URL hinz `Policy=`.
6. Erstellen Sie eine Signatur für die signierte URL, indem Sie die Richtlinienanweisung hashen, signieren und eine Base64-Codierung vornehmen. Weitere Informationen finden Sie unter the section called “Erstellen Sie eine Signatur für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet”.

Werte, die Sie in der Richtlinienanweisung für eine signierte URL angeben, die eine benutzerdefinierte Richtlinie verwendet

Beim Erstellen einer Richtlinienanweisung für eine benutzerdefinierte Richtlinie geben Sie die folgenden Werte an.

Ressource

Die URL, einschließlich aller Abfragezeichenfolgen, jedoch ohne die Parameter CloudFront `Policy`, und`Signature`. `Key-Pair-Id` Zum Beispiel:

```
https://d111111abcdef8.cloudfront.net/images/horizon.jpg\?
size=large&license=yes
```

Sie können nur einen URL-Wert für `Resource` angeben.

> **Important**
>
> Sie können den `Resource`-Parameter in einer Richtlinie weglassen. Dies bedeutet jedoch, dass alle, die über die signierte URL verfügen, auf alle Dateien in jeder Verteilung zugreifen können, die mit dem zum Erstellen der signierten URL verwendeten Schlüsselpaar verknüpft ist.

Beachten Sie Folgendes:

- Protokoll – Der Wert muss mit `http://`, `https://` oder `*://` beginnen.
- Abfragezeichenfolgenparameter – Wenn die URL Abfragezeichenfolgenparameter enthält, verwenden Sie einen umgekehrten Schrägstrich (\), um das Fragezeichen (?) zu maskieren, mit dem die Abfragezeichenfolge beginnt. Zum Beispiel:

  ```
  https://d111111abcdef8.cloudfront.net/images/horizon.jpg\?
  size=large&license=yes
  ```
- Platzhalterzeichen – Sie können Platzhalterzeichen in der URL in der Richtlinie verwenden. Die folgenden Platzhalterzeichen werden unterstützt:
  - Sternchen (*) für null oder mehr Zeichen.
  - Fragezeichen (?) für genau ein Zeichen

  Wenn die URL in der Richtlinie mit der URL in der HTTP-Anfrage CloudFront übereinstimmt, wird die URL in der Richtlinie wie folgt in vier Abschnitte unterteilt: Protokoll, Domäne, Pfad und Abfragezeichenfolge:

  ```
  [protocol]://[domain]/[path]\?[query string]
  ```

  Wenn Sie in der URL in der Richtlinie ein Platzhalterzeichen verwenden, erfolgt der Platzhalterabgleich nur innerhalb der Grenzen des Abschnitts, der den Platzhalter enthält. Betrachten Sie etwa diese URL in einer Richtlinie:

  ```
  https://www.example.com/hello*world
  ```

In diesem Beispiel gilt das Sternchen-Platzhalterzeichen (*) nur innerhalb des Pfadabschnitts, entspricht also dem URLs `https://www.example.com/helloworld` und`https://www.example.com/hello-world`, aber nicht der URL. `https://www.example.net/hello?world`

Die folgenden Ausnahmen gelten beim Platzhalterabgleich für die Abschnittsgrenzen:

- Ein nachfolgendes Sternchen im Pfadabschnitt impliziert ein Sternchen im Abschnitt mit der Abfragezeichenfolge. Beispiel: `http://example.com/hello*` ist gleichbedeutend mit `http://example.com/hello*\?*`.
- Ein nachfolgendes Sternchen im Domainabschnitt impliziert ein Sternchen sowohl im Pfad- als auch im Abschnitt mit der Abfragezeichenfolge. Beispiel: `http://example.com*` ist gleichbedeutend mit `http://example.com*/*\?*`.
- Eine URL in der Richtlinie kann den Protokollabschnitt weglassen und im Domainabschnitt mit einem Sternchen beginnen. In diesem Fall wird der Protokollabschnitt implizit auf ein Sternchen gesetzt. Beispielsweise entspricht die URL `*example.com` in einer Richtlinie `*://*example.com/`.
- Ein Sternchen an sich (`"Resource": "*"`) entspricht jeder URL.

Beispielsweise entspricht der Wert: `https://d111111abcdef8.cloudfront.net/*game_download.zip*` in einer Richtlinie allen folgenden Kriterien: URLs

- `https://d111111abcdef8.cloudfront.net/game_download.zip`
- `https://d111111abcdef8.cloudfront.net/example_game_download.zip?license=yes`
- `https://d111111abcdef8.cloudfront.net/test_game_download.zip?license=temp`

- Alternative Domainnamen – Wenn Sie einen alternativen Domainnamen (CNAME) in der URL in der Richtlinie angeben, muss die HTTP-Anfrage diesen alternativen Domainnamen auf Ihrer Webseite oder in Ihrer Anwendung verwenden. Geben Sie nicht die Amazon-S3-URL für die Datei in einer Richtlinie an.

DateLessThan

Das Ablaufdatum und die Ablaufzeit für die URL im Unix-Zeitformat (in Sekunden) und in koordinierter Weltzeit (UTC). Setzen Sie in der Richtlinie den Wert nicht in Anführungszeichen. Weitere Informationen zu UTC finden Sie unter Datum und Uhrzeit im Internet: Zeitstempel.

Beispielsweise wird der 31. Januar 2023, 10:00 Uhr UTC, im Unix-Zeitformat in 1675159200 konvertiert.

Dies ist der einzige erforderliche Parameter in `Condition` diesem Abschnitt. CloudFront benötigt diesen Wert, um zu verhindern, dass Benutzer dauerhaft auf Ihre privaten Inhalte zugreifen können.

Weitere Informationen finden Sie unter the section called “Wann CloudFront überprüft das Ablaufdatum und die Uhrzeit in einer signierten URL”.

DateGreaterThan (Fakultativ)

Ein optionales Datum und eine optionale Zeit für die URL im Unix-Zeitformat (in Sekunden) und in koordinierter Weltzeit (UTC). Benutzer dürfen am oder vor dem angegebenen Datum und der angegebenen Uhrzeit nicht auf die Datei zugreifen. Setzen Sie den Wert nicht in Anführungszeichen.

IpAddress (Fakultativ)

Die IP-Adresse des Clients, der die HTTP-Anfrage stellt. Beachten Sie Folgendes:

- Um allen IP-Adressen den Zugriff auf die Datei zu gewähren, lassen Sie den Parameter `IpAddress` weg.
- Sie können entweder eine IP-Adresse oder einen IP-Adressbereich angeben. Sie können die Richtlinie nicht zum Gewähren von Zugriff verwenden, wenn sich die IP-Adresse des Clients in einem von zwei getrennten Bereichen befindet.
- Um den Zugriff von einer einzigen IP-Adresse zu gewähren, geben Sie Folgendes an:

  `"IPv4 IP address/32"`
- Sie müssen IP-Adressbereiche im IPv4 CIDR-Standardformat angeben (z. B.`192.0.2.0/24`). Weitere Informationen finden Sie unter Classless Inter-domain Routing (CIDR): Internet-Adresszuweisung und Aggregierungsplan.

> **Important**
>
> IP-Adressen im IPv6 Format 2001:0 db 8:85 a3: :8a2e: 0370:7334 werden nicht unterstützt.

Wenn Sie eine benutzerdefinierte Richtlinie verwenden, die Folgendes umfasst: Aktivieren Sie diese Option nicht für die Verteilung. `IpAddress` IPv6 Wenn Sie den Zugriff auf einige

Inhalte anhand der IP-Adresse und der IPv6 Support-Anfragen für andere Inhalte einschränken möchten, können Sie zwei Distributionen erstellen. Weitere Informationen finden Sie unter the section called “Aktivieren IPv6” im Thema the section called “Alle Verteilungseinstellungen”.

Beispiel-Richtlinienanweisungen für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet

Die folgenden Beispiel-Richtlinienanweisungen zeigen, wie der Zugriff auf eine bestimmte Datei, auf alle Dateien in einem Verzeichnis oder auf alle mit einer Schlüsselpaar-ID verknüpften Dateien kontrolliert wird. Die Beispiele zeigen auch, wie der Zugriff von einer einzelnen IP-Adresse oder einem Bereich von IP-Adressen kontrolliert wird und wie Sie verhindern, dass Benutzer die signierte URL nach einem festgelegten Datum und einer festgelegten Zeit verwenden.

Wenn Sie eines dieser Beispiele kopieren und einfügen, entfernen Sie alle Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen), ersetzen Sie die Werte durch Ihre eigenen Werte und fügen Sie nach der schließenden Klammer () ein Zeilenumbruchzeichen ein. }

Weitere Informationen finden Sie unter the section called “Werte, die Sie in der Richtlinienanweisung für eine signierte URL angeben, die eine benutzerdefinierte Richtlinie verwendet”.

Themen

- Beispiel für eine Richtlinienerklärung: Greifen Sie von einem IP-Adressbereich aus auf eine Datei zu
- Beispiel für eine Richtlinienerklärung: Greifen Sie von einem IP-Adressbereich aus auf alle Dateien in einem Verzeichnis zu
- Beispiel für eine Richtlinienanweisung: Greifen Sie von einer IP-Adresse aus auf alle Dateien zu, die mit einer Schlüsselpaar-ID verknüpft sind

Beispiel für eine Richtlinienerklärung: Greifen Sie von einem IP-Adressbereich aus auf eine Datei zu

Das folgende Beispiel für eine Richtlinienanweisung in einer signierten URL legt fest, dass ein Benutzer bis zum 31. Januar 2013, 10:00 Uhr UTC, von IP-Adressen im Bereich `192.0.2.0/24` aus auf die Datei `https://d111111abcdef8.cloudfront.net/game_download.zip` zugreifen kann:

```
{
    "Statement": [
```

```
        {
            "Resource": "https://d111111abcdef8.cloudfront.net/game_download.zip",
            "Condition": {
                "IpAddress": {
                    "AWS:SourceIp": "192.0.2.0/24"
                },
                "DateLessThan": {
                    "AWS:EpochTime": 1675159200
                }
            }
        }
    ]
}
```

Beispiel für eine Richtlinienerklärung: Greifen Sie von einem IP-Adressbereich aus auf alle Dateien in einem Verzeichnis zu

Das folgende Beispiel einer benutzerdefinierten Richtlinie ermöglicht es Ihnen, URLs für jede Datei im `training` Verzeichnis eine Signatur zu erstellen, wie durch das Sternchen-Platzhalterzeichen (*) im `Resource` Parameter angegeben. Benutzer können bis zum 31. Januar 2013, 10:00 Uhr UTC, von IP-Adressen im Bereich `192.0.2.0/24` aus auf die Datei zugreifen:

```
{
    "Statement": [
        {
            "Resource": "https://d111111abcdef8.cloudfront.net/training/*",
            "Condition": {
                "IpAddress": {
                    "AWS:SourceIp": "192.0.2.0/24"
                },
                "DateLessThan": {
                    "AWS:EpochTime": 1675159200
                }
            }
        }
    ]
}
```

Jede signierte URL, mit der Sie diese Richtlinie verwenden, enthält eine URL, die eine bestimmte Datei identifiziert, zum Beispiel:

`https://d111111abcdef8.cloudfront.net/training/orientation.pdf`

Beispiel für eine Richtlinienanweisung: Greifen Sie von einer IP-Adresse aus auf alle Dateien zu, die mit einer Schlüsselpaar-ID verknüpft sind

Mit dem folgenden Beispiel einer benutzerdefinierten Richtlinie können Sie URLs für jede Datei, die mit einer beliebigen Distribution verknüpft ist, signiert erstellen, wie durch das Sternchen-Platzhalterzeichen (`*`) im `Resource` Parameter angezeigt wird. Die signierte URL muss das `https://`-Protokoll verwenden, nicht `http://`. Der Benutzer muss die IP-Adresse verwende `192.0.2.10/32`. (Der Wert `192.0.2.10/32` in CIDR-Notation bezieht sich auf eine einzelne IP-Adresse, `192.0.2.10`.) Die Dateien sind nur vom 31. Januar 2023, 10:00 Uhr UTC, bis zum 2. Februar 2023, 10:00 Uhr UTC, verfügbar:

```
{
    "Statement": [
       {
            "Resource": "https://*",
            "Condition": {
                "IpAddress": {
                    "AWS:SourceIp": "192.0.2.10/32"
                },
                "DateGreaterThan": {
                    "AWS:EpochTime": 1675159200
                },
                "DateLessThan": {
                    "AWS:EpochTime": 1675332000
                }
            }
        }
    ]
}
```

Jede signierte URL, mit der Sie diese Richtlinie verwenden, hat eine URL, die eine bestimmte Datei in einer bestimmten CloudFront Distribution identifiziert, zum Beispiel:

`https://d111111abcdef8.cloudfront.net/training/orientation.pdf`

Die signierte URL enthält auch eine Schlüsselpaar-ID, die mit einer vertrauenswürdigen Schlüsselgruppe in der Verteilung (d111111abcdef8.cloudfront.net) verknüpft werden muss, die Sie in der URL angeben.

Erstellen Sie eine Signatur für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet

Bei der Signatur für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet, handelt es sich um eine gehashte, signierte und Base64-codierte Version der Richtlinienanweisung. Führen Sie die folgenden Schritte aus, um eine Signatur für eine benutzerdefinierte Richtlinie zu erstellen.

Weitere Informationen und Beispiele für das Hashing, Signieren und Codieren der Richtlinienanweisung finden Sie unter:

- Linux-Befehle und OpenSSL für Base64-Kodierung und Verschlüsselung
- Code-Beispiele für das Erstellen einer Signatur für eine signierte URL

Option 1: So erstellen Sie eine Signatur mithilfe einer benutzerdefinierten Richtlinie

1. Verwenden Sie die SHA-1-Hash-Funktion und RSA, um die im Verfahren So erstellen Sie die Richtlinienanweisung für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet erstellte JSON-Richtlinienanweisung zu hashen und zu signieren. Verwenden Sie die Version der Richtlinienanweisung, die keine Leerzeichen mehr enthält, die aber noch nicht Base64-codiert wurde.

   Verwenden Sie für den privaten Schlüssel, der für die Hash-Funktion erforderlich ist, einen privaten Schlüssel, dessen öffentlicher Schlüssel sich in einer aktiven vertrauenswürdigen Schlüsselgruppe für die Verteilung befindet.

   > **Note**
   >
   > Die Methode, die Sie zum Hashen und Signieren der Richtlinienanweisung verwenden, ist abhängig von Ihrer Programmiersprache und Plattform. Einen Beispiel-Code finden Sie unter Code-Beispiele für das Erstellen einer Signatur für eine signierte URL.

2. Entfernen Sie Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen) aus der Hash-Zeichenfolge und der signierten Zeichenfolge.
3. Nehmen Sie eine Base64-Codierung der Zeichenfolge mithilfe von MIME-Base64-Codierung vor. Weitere Informationen finden Sie in Abschnitt 6.8, Base64 Content-Transfer-Encoding in RFC 2045, MIME (Multipurpose Internet Mail Extensions), Teil 1: Format von Internet-Nachrichtentexten.
4. Ersetzen Sie Zeichen, die in einer URL-Abfragezeichenfolge nicht gültig sind, durch gültige Zeichen. In der folgenden Tabelle sind ungültige und gültige Zeichen aufgelistet.

Ersetzen Sie diese ungültigen Zeichen	Durch diese gültigen Zeichen
+	- (Bindestrich)
=	_ (Unterstrich)
/	~ (Tilde)

5. Fügen Sie den resultierenden Wert hinter `&Signature=` zu Ihrer signierten URL hinzu und kehren Sie zu So erstellen Sie eine signierte URL mit einer benutzerdefinierten Richtlinie zurück, um das Verketten der Teile Ihrer signierten URL abzuschließen.

Verwenden Sie signierte Cookies

CloudFront Mit signierten Cookies können Sie kontrollieren, wer auf Ihre Inhalte zugreifen kann, wenn Sie Ihre aktuellen Inhalte nicht ändern möchten URLs oder wenn Sie Zugriff auf mehrere eingeschränkte Dateien gewähren möchten, z. B. auf alle Dateien im Abonnentenbereich einer Website. In diesem Thema werden die Überlegungen bei der Verwendung von signierten Cookies erläutert und es wird beschrieben, wie signierte Cookies mithilfe von vordefinierten und benutzerdefinierten Richtlinien eingerichtet werden.

Themen

- Entscheiden Sie sich dafür, vordefinierte oder benutzerdefinierte Richtlinien für signierte Cookies zu verwenden
- Funktionsweise von signierten Cookies
- Verhindern Sie den Missbrauch signierter Cookies
- Wenn das Ablaufdatum und die Uhrzeit in einem signierten Cookie CloudFront überprüft
- Beispiel-Code und Drittanbieter-Tools
- Legen Sie signierte Cookies mithilfe einer vordefinierten Richtlinie fest
- Legen Sie signierte Cookies mithilfe einer benutzerdefinierten Richtlinie fest
- Erstellen Sie signierte Cookies mit PHP

Entscheiden Sie sich dafür, vordefinierte oder benutzerdefinierte Richtlinien für signierte Cookies zu verwenden

Wenn Sie ein signiertes Cookie erstellen, schreiben Sie eine Richtlinienanweisung im JSON-Format, welche die Einschränkungen für das signierte Cookie festlegt, z. B. wie lange das Cookie gültig ist. Sie können vordefinierte oder benutzerdefinierte Richtlinien verwenden. In der folgenden Tabelle werden vordefinierte und benutzerdefinierte Richtlinien verglichen:

Beschreibung	Vordefinierte Richtlinie	Benutzerdefinierte Richtlinie
Sie können die Richtlinienanweisung für mehrere Dateien wiederverwenden. Um die Richtlini enanweisung wiederzuverwenden, müssen Sie Platzhalterzeichen im `Resource`-Objekt verwenden . Weitere Informationen finden Sie unter Werte, die Sie in der Richtlinienanweisung für eine benutzerd efinierte Richtlinie für signierte Cookies angeben.)	Nein	Ja
Sie können das Datum und die Zeit festlegen, ab denen Benutzer auf Ihre Inhalte zugreifen können.	Nein	Ja (optional)
Sie können das Datum und die Zeit festlegen, ab denen Benutzer nicht mehr auf Ihre Inhalte zugreifen können.	Ja	Ja
Sie können die IP-Adresse oder den Bereich von IP-Adressen der Benutzer festlegen, die auf Ihre Inhalte zugreifen können.	Nein	Ja (optional)

Informationen zum Erstellen von signierten Cookies mit einer vordefinierten Richtlinie finden Sie unter Legen Sie signierte Cookies mithilfe einer vordefinierten Richtlinie fest.

Informationen zum Erstellen von signierten Cookies mit einer benutzerdefinierten Richtlinie finden Sie unter Legen Sie signierte Cookies mithilfe einer benutzerdefinierten Richtlinie fest.

Funktionsweise von signierten Cookies

Hier finden Sie eine Übersicht darüber, wie Sie signierte Cookies konfigurieren CloudFront und wie Sie CloudFront reagieren, wenn ein Benutzer eine Anfrage einreicht, die ein signiertes Cookie enthält.

1. Geben Sie in Ihrer CloudFront Distribution eine oder mehrere vertrauenswürdige Schlüsselgruppen an, die die öffentlichen Schlüssel enthalten, anhand derer die URL-Signatur überprüft werden CloudFront kann. Sie verwenden die entsprechenden privaten Schlüssel, um die zu signieren URLs.

   Weitere Informationen finden Sie unter Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können.
2. Sie entwickeln Ihre Anwendung so, dass ermittelt wird, ob ein Benutzer über Zugriff auf Ihre Inhalte verfügen sollte, und wenn dies der Fall ist, drei `Set-Cookie`-Header an den Viewer zu senden. (Jeder `Set-Cookie` Header kann nur ein Name-Wert-Paar enthalten, und ein CloudFront signiertes Cookie benötigt drei Name-Wert-Paare.) Sie müssen die `Set-Cookie`-Header an den Viewer senden, bevor der Viewer Ihre privaten Inhalte anfragt. Wenn Sie eine kurze Ablaufzeit für das Cookie angeben, möchten Sie vielleicht auch drei weitere `Set-Cookie`-Header als Reaktion auf folgende Anfragen senden, sodass der Benutzer weiterhin Zugriff hat.

   In der Regel weist Ihre CloudFront Distribution mindestens zwei Cache-Verhaltensweisen auf, eines, für das keine Authentifizierung erforderlich ist, und eines, für das eine Authentifizierung erforderlich ist. Die Fehlerseite für den sicheren Teil der Website enthält eine Weiterleitung oder einen Link zu einer Anmeldeseite.

   Wenn Sie Ihre Distribution so konfigurieren, dass Dateien zwischengespeichert werden, die auf Cookies basieren, werden separate Dateien CloudFront nicht zwischengespeichert, die auf den Attributen in signierten Cookies basieren.
3. Ein Benutzer meldet sich auf Ihrer Website an und bezahlt entweder für Inhalte oder erfüllt andere Anforderungen für den Zugriff.
4. Ihre Anwendung gibt die `Set-Cookie`-Header in der Antwort zurück und der Viewer speichert die Name-Wert-Paare.
5. Der Benutzer fordert eine Datei an.

   Der Browser des Benutzers oder ein anderer Viewer ruft die Name-Wert-Paare aus Schritt 4 ab und fügt sie zu der Anfrage in einem `Cookie`-Header hinzu. Dies ist das signierte Cookie.

6. CloudFront verwendet den öffentlichen Schlüssel, um die Signatur im signierten Cookie zu validieren und um zu bestätigen, dass das Cookie nicht manipuliert wurde. Wenn die Signatur ungültig ist, wird die Anfrage abgelehnt.

   Wenn die Signatur im Cookie gültig ist, überprüft CloudFront die Richtlinienerklärung im Cookie (oder erstellt eine, wenn Sie eine vordefinierte Richtlinie verwenden), um zu bestätigen, dass die Anfrage noch gültig ist. Wenn Sie beispielsweise ein Anfangs- und Enddatum und eine Uhrzeit für das Cookie angegeben haben, wird CloudFront bestätigt, dass der Benutzer versucht, während des Zeitraums, für den Sie den Zugriff zulassen möchten, auf Ihre Inhalte zuzugreifen.

   Wenn die Anfrage die Anforderungen der Richtlinienerklärung CloudFront erfüllt, wird Ihr Inhalt genauso bereitgestellt wie bei Inhalten, für die keine Einschränkungen gelten: Es wird festgestellt, ob sich die Datei bereits im Edge-Cache befindet, leitet die Anfrage gegebenenfalls an den Ursprung weiter und gibt die Datei an den Benutzer zurück.

Verhindern Sie den Missbrauch signierter Cookies

Wenn Sie den `Domain`-Parameter in einem `Set-Cookie`-Header angeben, geben Sie einen möglichst genauen Wert ein, um das Potenzial für den Zugriff durch einen Benutzer mit demselben Stammdomänennamen zu verringern. Beispielsweise ist app.example.com gegenüber example.com vorzuziehen – insbesondere, wenn Sie example.com nicht kontrollieren. Dadurch können Sie verhindern, dass ein Benutzer von www.example.com aus auf Ihre Inhalte zugreift.

Gehen Sie wie folgt vor, um diese Art von Angriff zu verhindern:

- Schließen Sie die Cookie-Attribute `Expires` und `Max-Age` aus, damit der `Set-Cookie`-Header ein Sitzungs-Cookie erstellt. Sitzungs-Cookies werden automatisch gelöscht, wenn der Benutzer den Browser schließt. Dies verringert das Risiko, dass ein Benutzer unbefugten Zugriff auf Ihre Inhalte erhält.
- Fügen Sie das Attribut `Secure` ein, damit das Cookie verschlüsselt wird, wenn ein Viewer es in eine Anfrage einfügt.
- Verwenden Sie wenn möglich eine benutzerdefinierte Richtlinie und fügen Sie die IP-Adresse des Viewers ein.
- Geben Sie auf der Grundlage davon, wie lange Sie Benutzern Zugriff auf Ihre Inhalte gewähren möchten, im Attribut `CloudFront-Expires` eine möglichst kurze, vernünftige Ablaufzeit an.

Wenn das Ablaufdatum und die Uhrzeit in einem signierten Cookie CloudFront überprüft

Um festzustellen, ob ein signiertes Cookie noch gültig ist, werden das Ablaufdatum und die Uhrzeit im Cookie zum Zeitpunkt der HTTP-Anfrage CloudFront überprüft. Wenn ein Client unmittelbar vor der Ablaufzeit mit dem Download einer großen Datei beginnt, sollte der Download abgeschlossen werden, auch wenn die Ablaufzeit während des Downloads überschritten wird. Wenn die TCP-Verbindung getrennt wird und der Client nach Überschreitung der Ablaufzeit versucht, den Download erneut zu starten, schlägt der Download fehl.

Wenn ein Client Range verwendet GETs , um eine Datei in kleineren Teilen abzurufen, schlägt jede GET-Anforderung fehl, die nach Ablauf der Ablaufzeit erfolgt. Weitere Informationen zu Range GETs finden Sie unterWie CloudFront werden Teilanfragen für ein Objekt (BereichGETs) verarbeitet.

Beispiel-Code und Drittanbieter-Tools

Der Beispielcode für private Inhalte zeigt nur, wie die Signatur für signierte Inhalte erstellt wird URLs. Das Erstellen einer Signatur für ein signiertes Cookie ist jedoch sehr ähnlich, deshalb ist ein Großteil des Beispiel-Codes auch hier von Bedeutung. Weitere Informationen finden Sie unter den folgenden Themen:

- Erstellen einer URL-Signatur mit Perl
- Erstellen einer URL-Signatur mit PHP
- Erstellen einer URL-Signatur mithilfe von C# und dem .NET Framework
- Erstellen einer URL-Signatur mit Java

Legen Sie signierte Cookies mithilfe einer vordefinierten Richtlinie fest

Um ein signiertes Cookie mit einer vordefinierten Richtlinie einzurichten, führen Sie die folgenden Schritte aus. Zum Erstellen einer Signatur vgl. Erstellen Sie eine Signatur für ein signiertes Cookie, das eine vordefinierte Richtlinie verwendet.

So richten Sie ein signiertes Cookies mit einer vordefinierten Richtlinie ein

1. Wenn Sie .NET oder Java verwenden, um signierte Cookies zu erstellen, und den privaten Schlüssel für Ihr Schlüsselpaar noch nicht vom PEM-Standardformat in ein mit .NET oder Java kompatibles Format neu formatiert haben, holen Sie diesen Schritt jetzt nach. Weitere Informationen finden Sie unter Formatieren Sie den privaten Schlüssel neu (nur .NET und Java).

2. Programmieren Sie Ihre Anwendung so, dass drei `Set-Cookie`-Header an genehmigte Viewer gesendet werden. Sie benötigen drei `Set-Cookie`-Header, weil jeder `Set-Cookie`-Header nur ein Name-Wert-Paar enthalten kann und ein CloudFront -signiertes Cookie drei Name-Wert-Paare erfordert. Die Name-Wert-Paare sind: `CloudFront-Expires`, `CloudFront-Signature` und `CloudFront-Key-Pair-Id`. Die Werte müssen auf dem Viewer vorhanden sein, bevor ein Benutzer die erste Anfrage für eine Datei stellt, bei der der Zugriff kontrolliert werden soll.

> **Note**
>
> Im Allgemeinen empfehlen wir, die Attribute `Expires` und `Max-Age` auszuschließen. Der Ausschluss der Attribute bewirkt, dass der Browser das Cookie löscht, wenn der Benutzer den Browser schließt. Dies verringert das Risiko, dass ein Benutzer unbefugten Zugriff auf Ihre Inhalte erhält. Weitere Informationen finden Sie unter Verhindern Sie den Missbrauch signierter Cookies.

Bei den Namen der Cookie-Attribute muss die Groß- und Kleinschreibung beachtet werden.

Zeilenumbrüche werden nur hinzugefügt, damit die Attribute besser lesbar sind.

```
Set-Cookie:
CloudFront-Expires=date and time in Unix time format (in seconds) and Coordinated
 Universal Time (UTC);
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly

Set-Cookie:
CloudFront-Signature=hashed and signed version of the policy statement;
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly

Set-Cookie:
CloudFront-Key-Pair-Id=public key ID for the CloudFront public key whose
 corresponding private key you're using to generate the signature;
Domain=optional domain name;
```

```
Path=/optional directory path;
Secure;
HttpOnly
```

(Optional) **`Domain`**

Der Domänenname für die angeforderte Datei. Wenn Sie kein `Domain`-Attribut angeben, ist der Standardwert der Domänenname in der URL und dieser gilt nur für die angegebenen Domänennamen, nicht für Unterdomänen. Wenn Sie ein `Domain`-Attribut angeben, gilt dieses auch für Unterdomänen. Ein vorangestellter Punkt im Domänennamen (z. B. `Domain=.example.com`) ist optional. Wenn Sie ein `Domain`-Attribut angeben, müssen darüber hinaus der Domänenname in der URL und der Wert des `Domain`-Attributs übereinstimmen.

Sie können den Domainnamen angeben, der Ihrer Distribution CloudFront zugewiesen wurde, z. B. d111111abcdef8.cloudfront.net, aber Sie können nicht *.cloudfront.net für den Domainnamen angeben.

Wenn Sie einen alternativen Domainnamen wie example.com in verwenden möchten, müssen Sie den alternativen Domainnamen zu Ihrer Distribution hinzufügen URLs, unabhängig davon, ob Sie das Attribut angeben. `Domain` Weitere Informationen finden Sie unter Alternative Domainnamen (CNAMEs) im Thema Referenz für alle Verteilungseinstellungen.

(Optional) **`Path`**

Der Pfad für die angeforderte Datei. Wenn Sie kein `Path`-Attribut angeben, ist der Standardwert der Pfad in der URL.

`Secure`

Macht es erforderlich, dass der Viewer Cookies vor dem Senden einer Anfrage verschlüsselt. Wir empfehlen, den `Set-Cookie` Header über eine HTTPS-Verbindung zu senden, um sicherzustellen, dass die Cookie-Attribute vor man-in-the-middle Angriffen geschützt sind.

`HttpOnly`

Definiert, wie der Browser (sofern unterstützt) mit dem Cookie-Wert interagiert. Mit`HttpOnly`, auf die Cookie-Werte kann nicht zugegriffen werden. JavaScript Diese Vorsichtsmaßnahme kann dazu beitragen, Cross-Site Scripting (XSS) -Angriffe abzuwehren. Weitere Informationen finden Sie unter HTTP-Cookies verwenden.

`CloudFront-Expires`

Geben Sie das Ablaufdatum und die Ablaufzeit im Unix-Zeitformat (in Sekunden) und in koordinierter Weltzeit (UTC) an. Beispielsweise wird der 1. Januar 2013, 10:00 Uhr UTC in 1357034400 im Unix-Zeitformat umgewandelt. Um die Epochen-Uhrzeit zu verwenden, geben Sie eine 32-Bit-Ganzzahl für ein Datum an, das nicht nach 2147483647 (19. Januar 2038 um 03:14:07 UTC) liegt. Weitere Informationen zu UTC finden Sie unter RFC 3339, Datum und Uhrzeit im Internet: Zeitstempel, https://tools.ietf.org/html/rfc3339.

`CloudFront-Signature`

Eine gehashte, signierte und Base64-codierte Version einer JSON-Richtlinienanweisung. Weitere Informationen finden Sie unter Erstellen Sie eine Signatur für ein signiertes Cookie, das eine vordefinierte Richtlinie verwendet.

`CloudFront-Key-Pair-Id`

Die ID für einen CloudFront öffentlichen Schlüssel, zum Beispiel`K2JCJMDEHXQW5F`. Die ID des öffentlichen Schlüssels gibt an CloudFront , welcher öffentliche Schlüssel zur Validierung der signierten URL verwendet werden soll. CloudFront vergleicht die Informationen in der Signatur mit den Informationen in der Richtlinienerklärung, um sicherzustellen, dass die URL nicht manipuliert wurde.

Dieser öffentliche Schlüssel muss zu einer Schlüsselgruppe gehören, die ein vertrauenswürdiger Aussteller in der Verteilung ist. Weitere Informationen finden Sie unter Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können.

Das folgende Beispiel zeigt `Set-Cookie` Header für ein signiertes Cookie, wenn Sie den Domainnamen verwenden, der Ihrer Distribution zugeordnet ist, in Ihren URLs Dateien:

```
Set-Cookie: CloudFront-Expires=1426500000; Domain=d111111abcdef8.cloudfront.net; Path=/
images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=yXrSIgyQoeE4FBI4eMKF6ho~CA8_;
 Domain=d111111abcdef8.cloudfront.net; Path=/images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JCJMDEHXQW5F;
 Domain=d111111abcdef8.cloudfront.net; Path=/images/*; Secure; HttpOnly
```

Das folgende Beispiel zeigt `Set-Cookie` Header für ein signiertes Cookie, wenn Sie den alternativen Domainnamen example.org in Ihren Dateien verwenden: URLs

```
Set-Cookie: CloudFront-Expires=1426500000; Domain=example.org; Path=/images/*; Secure;
 HttpOnly
Set-Cookie: CloudFront-Signature=yXrSIgyQoeE4FBI4eMKF6ho~CA8_; Domain=example.org;
 Path=/images/*; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JCJMDEHXQW5F; Domain=example.org; Path=/images/*;
 Secure; HttpOnly
```

Wenn Sie einen alternativen Domainnamen wie example.com in verwenden möchten URLs, müssen Sie den alternativen Domainnamen zu Ihrer Distribution hinzufügen, unabhängig davon, ob Sie das Attribut angeben. `Domain` Weitere Informationen finden Sie unter Alternative Domainnamen (CNAMEs) im Thema Referenz für alle Verteilungseinstellungen.

Erstellen Sie eine Signatur für ein signiertes Cookie, das eine vordefinierte Richtlinie verwendet

Gehen Sie wie folgt vor, um die Signatur für ein signiertes Cookie zu erstellen, das eine vordefinierte Richtlinie verwendet.

Themen

- Erstellen Sie eine Richtlinienerklärung für ein signiertes Cookie, das eine vordefinierte Richtlinie verwendet
- Unterzeichnen Sie die Richtlinienerklärung, um eine Signatur für ein signiertes Cookie zu erstellen, das eine vordefinierte Richtlinie verwendet

Erstellen Sie eine Richtlinienerklärung für ein signiertes Cookie, das eine vordefinierte Richtlinie verwendet

Wenn Sie ein signiertes Cookie einrichten, das eine vordefinierte Richtlinie verwendet, ist das `CloudFront-Signature`-Attribut eine gehashte und signierte Version einer Richtlinienanweisung. Bei signierten Cookies, die eine vordefinierte Richtlinie verwenden, fügen Sie die Richtlinienanweisung nicht in den `Set-Cookie`-Header ein, wie Sie es bei signierten Cookies tun, die eine benutzerdefinierte Richtlinie verwenden. Führen Sie die folgenden Schritte aus, um die Richtlinienanweisung zu erstellen.

So erstellen Sie eine Richtlinienanweisung für ein signiertes Cookie, das eine vordefinierte Richtlinie verwendet

1. Erstellen Sie die Richtlinienanweisung unter Verwendung des folgenden JSON-Formats und der UTF-8-Zeichencodierung. Fügen Sie alle Satzzeichen und andere Literalwerte genau wie angegeben ein. Informationen zu den Parametern `Resource` und `DateLessThan` finden Sie

unter Werte, die Sie in der Richtlinienanweisung für eine vordefinierte Richtlinie für signierte Cookies angeben.

```
{
    "Statement": [
        {
            "Resource": "base URL or stream name",
            "Condition": {
                "DateLessThan": {
                    "AWS:EpochTime": ending date and time in Unix time format and
 UTC
                }
            }
        }
    ]
}
```

2. Entfernen Sie alle Leerzeichen (einschließlich Tabulatoren und Zeilenumbrüche) aus der Richtlinienerklärung. Möglicherweise müssen Sie in der Zeichenfolge im Anwendungscode Escape-Zeichen einfügen.

Werte, die Sie in der Richtlinienanweisung für eine vordefinierte Richtlinie für signierte Cookies angeben

Beim Erstellen einer Richtlinienanweisung für eine vordefinierte Richtlinie geben Sie die folgenden Werte an:

Ressource

Die Basis-URL einschließlich Ihrer Abfragezeichenfolgen, sofern vorhanden, zum Beispiel:

```
https://d111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

Sie können nur einen Wert für `Resource` angeben.

Beachten Sie Folgendes:

- Protokoll – Der Wert muss mit `http://` oder `https://` beginnen.
- Abfragezeichenfolgeparameter – Wenn Sie über keine Abfragezeichenfolgeparameter verfügen, lassen Sie das Fragezeichen weg.

- Alternative Domänennamen – Wenn Sie einen alternativen Domänennamen (CNAME) in der URL angeben, müssen Sie diesen alternativen Domänennamen angeben, wenn Sie auf Ihrer Webseite oder in Ihrer Anwendung auf die Datei verweisen. Geben Sie nicht die Amazon S3-URL für Datei an.

DateLessThan

Das Ablaufdatum und die Ablaufzeit für die URL im Unix-Zeitformat (in Sekunden) und in koordinierter Weltzeit (UTC). Setzen Sie den Wert nicht in Anführungszeichen.

Beispielsweise wird der 16. März 2015, 10:00 Uhr UTC in 1426500000 im Unix-Zeitformat umgewandelt.

Dieser Wert muss mit dem Wert des `CloudFront-Expires`-Attributs im `Set-Cookie`-Header übereinstimmen. Setzen Sie den Wert nicht in Anführungszeichen.

Weitere Informationen finden Sie unter Wenn das Ablaufdatum und die Uhrzeit in einem signierten Cookie CloudFront überprüft.

Beispiel-Richtlinienanweisung für eine vordefinierte Richtlinie

Wenn Sie die folgende Beispiel-Richtlinienanweisung in einem signierten Cookie verwenden, kann ein Benutzer bis zum 16. März 2015, 10:00 Uhr UTC, auf die Datei `https://d111111abcdef8.cloudfront.net/horizon.jpg` zugreifen:

```
{
    "Statement": [
        {
            "Resource": "https://d111111abcdef8.cloudfront.net/horizon.jpg?
size=large&license=yes",
            "Condition": {
                "DateLessThan": {
                    "AWS:EpochTime": 1426500000
                }
            }
        }
    ]
}
```

Unterzeichnen Sie die Richtlinienerklärung, um eine Signatur für ein signiertes Cookie zu erstellen, das eine vordefinierte Richtlinie verwendet

Um den Wert für das `CloudFront-Signature`-Attribut in einem `Set-Cookie`-Header zu erstellen, müssen Sie die in So erstellen Sie eine Richtlinienanweisung für ein signiertes Cookie, das eine vordefinierte Richtlinie verwendet erstellte Richtlinienanweisung hashen und signieren.

Weitere Informationen und Beispiele für das Hashing, Signieren und Codieren der Richtlinienanweisung finden Sie in den folgenden Themen:

- Linux-Befehle und OpenSSL für Base64-Kodierung und Verschlüsselung
- Code-Beispiele für das Erstellen einer Signatur für eine signierte URL

So erstellen Sie eine Signatur für ein signiertes Cookie mit einer vordefinierten Richtlinie

1. Verwenden Sie die SHA-1-Hash-Funktion und RSA, um die im Verfahren So erstellen Sie eine Richtlinienanweisung für ein signiertes Cookie, das eine vordefinierte Richtlinie verwendet erstellte Richtlinienanweisung zu hashen und zu signieren. Verwenden Sie die Version der Richtlinienerklärung, die keine Leerzeichen mehr enthält.

   Verwenden Sie für den privaten Schlüssel, der für die Hash-Funktion erforderlich ist, einen privaten Schlüssel, dessen öffentlicher Schlüssel sich in einer aktiven vertrauenswürdigen Schlüsselgruppe für die Verteilung befindet.

   > **Note**
   >
   > Die Methode, die Sie zum Hashen und Signieren der Richtlinienanweisung verwenden, ist abhängig von Ihrer Programmiersprache und Plattform. Einen Beispiel-Code finden Sie unter Code-Beispiele für das Erstellen einer Signatur für eine signierte URL.

2. Entfernen Sie Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen) aus der Hash-Zeichenfolge und der signierten Zeichenfolge.
3. Nehmen Sie eine Base64-Codierung der Zeichenfolge mithilfe von MIME-Base64-Codierung vor. Weitere Informationen finden Sie in Abschnitt 6.8, Base64 Content-Transfer-Encoding in RFC 2045, MIME (Multipurpose Internet Mail Extensions), Teil 1: Format von Internet-Nachrichtentexten.
4. Ersetzen Sie Zeichen, die in einer URL-Abfragezeichenfolge nicht gültig sind, durch gültige Zeichen. In der folgenden Tabelle sind ungültige und gültige Zeichen aufgelistet.

Ersetzen Sie diese ungültigen Zeichen	Durch diese gültigen Zeichen
+	- (Bindestrich)
=	_ (Unterstrich)
/	~ (Tilde)

5. Fügen Sie den resultierenden Wert im `Set-Cookie`-Header für das `CloudFront-Signature`-Name-Wert-Paar ein. Kehren Sie anschließend zu So richten Sie ein signiertes Cookies mit einer vordefinierten Richtlinie ein zurück und fügen Sie den `Set-Cookie`-Header für `CloudFront-Key-Pair-Id` hinzu.

Legen Sie signierte Cookies mithilfe einer benutzerdefinierten Richtlinie fest

Führen Sie die folgenden Schritte aus, um ein signiertes Cookie einzurichten, das eine benutzerdefinierte Richtlinie verwendet:

So richten Sie ein signiertes Cookies mit einer benutzerdefinierten Richtlinie ein

1. Wenn Sie.NET oder Java verwenden, um signierte zu erstellen URLs, und wenn Sie den privaten Schlüssel für Ihr key pair nicht vom standardmäßigen .pem-Format in ein mit .NET oder Java kompatibles Format umformatiert haben, tun Sie dies jetzt. Weitere Informationen finden Sie unter Formatieren Sie den privaten Schlüssel neu (nur .NET und Java).
2. Programmieren Sie Ihre Anwendung so, dass drei `Set-Cookie`-Header an genehmigte Viewer gesendet werden. Sie benötigen drei `Set-Cookie` Header, da jeder `Set-Cookie` Header nur ein Name-Wert-Paar enthalten kann und ein CloudFront signiertes Cookie drei Name-Wert-Paare benötigt. Die Name-Wert-Paare sind: `CloudFront-Policy`, `CloudFront-Signature` und `CloudFront-Key-Pair-Id`. Die Werte müssen auf dem Viewer vorhanden sein, bevor ein Benutzer die erste Anfrage für eine Datei stellt, bei der der Zugriff kontrolliert werden soll.

> **Note**
>
> Im Allgemeinen empfehlen wir, die Attribute `Expires` und `Max-Age` auszuschließen. Dies bewirkt, dass der Browser das Cookie löscht, wenn der Benutzer den Browser schließt. Dies verringert das Risiko, dass ein Benutzer unbefugten Zugriff auf Ihre Inhalte

erhält. Weitere Informationen finden Sie unter Verhindern Sie den Missbrauch signierter Cookies.

Bei den Namen der Cookie-Attribute muss die Groß- und Kleinschreibung beachtet werden.

Zeilenumbrüche werden nur hinzugefügt, damit die Attribute besser lesbar sind.

```
Set-Cookie:
CloudFront-Policy=base64 encoded version of the policy statement;
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly


Set-Cookie:
CloudFront-Signature=hashed and signed version of the policy statement;
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly

Set-Cookie:
CloudFront-Key-Pair-Id=public key ID for the CloudFront public key whose
 corresponding private key you're using to generate the signature;
Domain=optional domain name;
Path=/optional directory path;
Secure;
HttpOnly
```

(Optional) **`Domain`**

Der Domänenname für die angeforderte Datei. Wenn Sie kein `Domain`-Attribut angeben, ist der Standardwert der Domänenname in der URL und dieser gilt nur für die angegebenen Domänennamen, nicht für Unterdomänen. Wenn Sie ein `Domain`-Attribut angeben, gilt dieses auch für Unterdomänen. Ein vorangestellter Punkt im Domänennamen (z. B. `Domain=.example.com`) ist optional. Wenn Sie ein `Domain`-Attribut angeben, müssen darüber hinaus der Domänenname in der URL und der Wert des `Domain`-Attributs übereinstimmen.

Sie können den Domainnamen angeben, der Ihrer Distribution CloudFront zugewiesen wurde, z. B. d111111abcdef8.cloudfront.net, aber Sie können nicht *.cloudfront.net für den Domainnamen angeben.

Wenn Sie einen alternativen Domainnamen wie example.com in verwenden möchten, müssen Sie den alternativen Domainnamen zu Ihrer Distribution hinzufügen URLs, unabhängig davon, ob Sie das Attribut angeben. `Domain` Weitere Informationen finden Sie unter Alternative Domainnamen (CNAMEs) im Thema Referenz für alle Verteilungseinstellungen.

(Optional) **`Path`**

Der Pfad für die angeforderte Datei. Wenn Sie kein `Path`-Attribut angeben, ist der Standardwert der Pfad in der URL.

`Secure`

Macht es erforderlich, dass der Viewer Cookies vor dem Senden einer Anfrage verschlüsselt. Wir empfehlen, den `Set-Cookie` Header über eine HTTPS-Verbindung zu senden, um sicherzustellen, dass die Cookie-Attribute vor man-in-the-middle Angriffen geschützt sind.

`HttpOnly`

Macht es erforderlich, dass der Viewer das Cookie nur in HTTP- oder HTTPS-Anfragen sendet.

`CloudFront-Policy`

Ihre Richtlinienerklärung im JSON-Format, wobei Leerzeichen entfernt und anschließend base64-codiert sind. Weitere Informationen finden Sie unter Erstellen Sie eine Signatur für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet.

Die Richtlinienanweisung steuert den Zugriff, den ein signiertes Cookie einem Benutzer gewährt. Sie enthält die Dateien, auf die der Benutzer zugreifen kann, ein Ablaufdatum und eine Ablaufuhrzeit, ein optionales Datum und die Uhrzeit, zu der die URL gültig wird, und eine optionale IP-Adresse oder einen Bereich von IP-Adressen, die auf die Datei zugreifen dürfen.

`CloudFront-Signature`

Eine gehashte, signierte und Base64-codierte Version der JSON-Richtlinienanweisung. Weitere Informationen finden Sie unter Erstellen Sie eine Signatur für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet.

`CloudFront-Key-Pair-Id`

Die ID für einen CloudFront öffentlichen Schlüssel, zum Beispiel. `K2JCJMDEHXQW5F` Die ID des öffentlichen Schlüssels gibt an CloudFront, welcher öffentliche Schlüssel zur Validierung der signierten URL verwendet werden soll. CloudFrontvergleicht die Informationen in der Signatur mit den Informationen in der Richtlinienerklärung, um sicherzustellen, dass die URL nicht manipuliert wurde.

Dieser öffentliche Schlüssel muss zu einer Schlüsselgruppe gehören, die ein vertrauenswürdiger Aussteller in der Verteilung ist. Weitere Informationen finden Sie unter Geben Sie Unterzeichner an, die signierte URLs und signierte Cookies erstellen können.

`Set-Cookie`Beispiel-Header für benutzerdefinierte Richtlinien

Sehen Sie sich die folgenden Beispiele für `Set-Cookie` Header-Paare an.

Wenn Sie einen alternativen Domainnamen wie example.org in verwenden möchten URLs, müssen Sie den alternativen Domainnamen zu Ihrer Distribution hinzufügen, unabhängig davon, ob Sie das `Domain` Attribut angeben. Weitere Informationen finden Sie unter Alternative Domainnamen (CNAMEs) im Thema Referenz für alle Verteilungseinstellungen.

Example Beispiel 1

Sie können die `Set-Cookie` Header für ein signiertes Cookie verwenden, wenn Sie den Domainnamen verwenden, der Ihrer Distribution zugeordnet ist, in Ihren URLs Dateien.

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZW1lbnQiOlt7IlJlc291cmNlIjoiaHR0cDovL2QxMTExMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dh
 Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_;
 Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JCJMDEHXQW5F;
 Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
```

Example Beispiel 2

Sie können die `Set-Cookie` Header für ein signiertes Cookie verwenden, wenn Sie URLs für Ihre Dateien einen alternativen Domainnamen (example.org) verwenden.

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZW1lbnQiOlt7IlJlc291cmNlIjoiaHR0cDovL2QxMTExMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dh
 Domain=example.org; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_; Domain=example.org;
 Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JCJMDEHXQW5F; Domain=example.org; Path=/; Secure;
 HttpOnly
```

Example Beispiel 3

Sie können die `Set-Cookie` Header-Paare für eine signierte Anfrage verwenden, wenn Sie den Domainnamen verwenden, der Ihrer Distribution zugeordnet ist, in der URLs für Ihre Dateien.

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZW1lbnQiOlt7IlJlc291cmNlIjoiaHR0cDovL2QxMTExMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dh
 Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_;
 Domain=d111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JCJMDEHXQW5F;
 Domain=dd111111abcdef8.cloudfront.net; Path=/; Secure; HttpOnly
```

Example Beispiel 4

Sie können die `Set-Cookie` Header-Paare für eine signierte Anfrage verwenden, wenn Sie einen alternativen Domainnamen (example.org) verwenden, der URLs Ihrer Distribution in Ihren Dateien zugeordnet ist.

```
Set-Cookie: CloudFront-
Policy=eyJTdGF0ZW1lbnQiOlt7IlJlc291cmNlIjoiaHR0cDovL2QxMTExMTFhYmNkZWY4LmNsb3VkZnJvbnQubmV0L2dh
 Domain=example.org; Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Signature=dtKhpJ3aUYxqDIwepczPiDb9NXQ_; Domain=example.org;
 Path=/; Secure; HttpOnly
Set-Cookie: CloudFront-Key-Pair-Id=K2JCJMDEHXQW5F; Domain=example.org; Path=/; Secure;
 HttpOnly
```

Erstellen Sie eine Richtlinienerklärung für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet

Zum Erstellen einer Richtlinienanweisung für eine benutzerdefinierte Richtlinie führen Sie die folgenden Schritte aus. Einige Beispiel-Richtlinienanweisungen, die den Zugriff auf Dateien auf

verschiedene Weisen kontrollieren, finden Sie unter Beispiel-Richtlinienanweisungen für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet.

So erstellen Sie die Richtlinienanweisung für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet

1. Erstellen Sie die Richtlinienanweisung unter Verwendung des folgenden JSON-Formats.

```
{
    "Statement": [
        {
            "Resource": "URL of the file",
            "Condition": {
                "DateLessThan": {
                    "AWS:EpochTime":required ending date and time in Unix time
 format and UTC
                },
                "DateGreaterThan": {
                    "AWS:EpochTime":optional beginning date and time in Unix time
 format and UTC
                },
                "IpAddress": {
                    "AWS:SourceIp": "optional IP address"
                }
            }
        }
    ]
}
```

Beachten Sie Folgendes:

- Sie können nur eine Anweisung einschließen.
- Verwenden Sie UTF-8-Zeichencodierung.
- Fügen Sie alle Satzzeichen und Parameternamen genau wie angegeben ein. Abkürzungen für Parameternamen werden nicht akzeptiert.
- Die Reihenfolge der Parameter im Bereich `Condition` ist unerheblich.
- Informationen zu den Werten für `Resource`, `DateLessThan`, `DateGreaterThan` und `IpAddress` finden Sie unter Werte, die Sie in der Richtlinienanweisung für eine benutzerdefinierte Richtlinie für signierte Cookies angeben.

2. Entfernen Sie alle Leerzeichen (einschließlich Tabulatoren und Zeilenumbrüche) aus der Richtlinienerklärung. Möglicherweise müssen Sie in der Zeichenfolge im Anwendungscode Escape-Zeichen einfügen.
3. Nehmen Sie eine Base64-Codierung der Richtlinienanweisung mithilfe von MIME-Base64-Codierung vor. Weitere Informationen finden Sie in Abschnitt 6.8, Base64 Content-Transfer-Encoding in RFC 2045, MIME (Multipurpose Internet Mail Extensions), Erster Teil: Format von Internet-Nachrichtentexten.
4. Ersetzen Sie Zeichen, die in einer URL-Abfragezeichenfolge nicht gültig sind, durch gültige Zeichen. In der folgenden Tabelle sind ungültige und gültige Zeichen aufgelistet.

Ersetzen Sie diese ungültigen Zeichen	Durch diese gültigen Zeichen
+	- (Bindestrich)
=	_ (Unterstrich)
/	~ (Tilde)

5. Fügen Sie den resultierenden Wert im `Set-Cookie`-Header hinter `CloudFront-Policy=` ein.
6. Erstellen Sie eine Signatur für den `Set-Cookie`-Header für `CloudFront-Signature`, indem Sie die Richtlinienanweisung hashen, signieren und eine Base64-Codierung vornehmen. Weitere Informationen finden Sie unter Erstellen Sie eine Signatur für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet.

Werte, die Sie in der Richtlinienanweisung für eine benutzerdefinierte Richtlinie für signierte Cookies angeben

Beim Erstellen einer Richtlinienanweisung für eine benutzerdefinierte Richtlinie geben Sie die folgenden Werte an.

Ressource

Die Basis-URL einschließlich Ihrer Abfragezeichenfolgen, sofern vorhanden:

```
https://d111111abcdef8.cloudfront.net/images/horizon.jpg?
size=large&license=yes
```

> **Important**
>
> Wenn Sie den `Resource`-Parameter weglassen, können Benutzer auf alle Dateien zugreifen, die mit einer Verteilung verknüpft sind, die mit dem zum Erstellen der signierten URL verwendeten Schlüsselpaar verknüpft ist.

Sie können nur einen Wert für `Resource` angeben.

Beachten Sie Folgendes:

- Protokoll – Der Wert muss mit `http://` oder `https://` beginnen.
- Abfragezeichenfolgeparameter – Wenn Sie über keine Abfragezeichenfolgeparameter verfügen, lassen Sie das Fragezeichen weg.
- Platzhalter – Sie können das Platzhalterzeichen, das null oder mehr Zeichen (*), oder das Platzhalterzeichen, das genau einem Zeichen (?) irgendwo in der Zeichenfolge entspricht, verwenden. Beispielsweise würde der Wert:

  ```
  https://d111111abcdef8.cloudfront.net/*game_download.zip*
  ```

  u. a. die folgenden Dateien umfassen:

  - `https://d111111abcdef8.cloudfront.net/game_download.zip`
  - `https://d111111abcdef8.cloudfront.net/example_game_download.zip?license=yes`
  - `https://d111111abcdef8.cloudfront.net/test_game_download.zip?license=temp`
- Alternative Domänennamen – Wenn Sie einen alternativen Domänennamen (CNAME) in der URL angeben, müssen Sie diesen alternativen Domänennamen angeben, wenn Sie auf Ihrer Webseite oder in Ihrer Anwendung auf die Datei verweisen. Geben Sie nicht die Amazon S3-URL für Datei an.

DateLessThan

Das Ablaufdatum und die Ablaufzeit für die URL im Unix-Zeitformat (in Sekunden) und in koordinierter Weltzeit (UTC). Setzen Sie den Wert nicht in Anführungszeichen.

Beispielsweise wird der 16. März 2015, 10:00 Uhr UTC in 1426500000 im Unix-Zeitformat umgewandelt.

Weitere Informationen finden Sie unter Wenn das Ablaufdatum und die Uhrzeit in einem signierten Cookie CloudFront überprüft.

DateGreaterThan (Fakultativ)

Ein optionales Datum und eine optionale Zeit für die URL im Unix-Zeitformat (in Sekunden) und in koordinierter Weltzeit (UTC). Benutzer dürfen am oder vor dem angegebenen Datum und der angegebenen Uhrzeit nicht auf die Datei zugreifen. Setzen Sie den Wert nicht in Anführungszeichen.

IpAddress (Fakultativ)

Die IP-Adresse des Clients, der die GET-Anfrage stellt. Beachten Sie Folgendes:

- Um allen IP-Adressen den Zugriff auf die Datei zu gewähren, lassen Sie den Parameter `IpAddress` weg.
- Sie können entweder eine IP-Adresse oder einen IP-Adressbereich angeben. Sie können die Richtlinie beispielsweise nicht so einrichten, dass Zugriff gewährt wird, wenn die IP-Adresse des Clients sich in einem von zwei getrennten Bereichen befindet.
- Um den Zugriff von einer einzigen IP-Adresse zu gewähren, geben Sie Folgendes an:

  `"`*`IPv4 IP address`*`/32"`
- Sie müssen IP-Adressbereiche im IPv4 CIDR-Standardformat angeben (z. B.`192.0.2.0/24`). Weitere Informationen erhalten Sie unter RFC 4632, Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan, https://tools.ietf.org/html/rfc4632.

> **Important**
>
> IP-Adressen im IPv6 Format 2001:0 db 8:85 a3: :8a2e: 0370:7334 werden nicht unterstützt.

Wenn Sie eine benutzerdefinierte Richtlinie verwenden, die Folgendes umfasst: Aktivieren Sie diese Option nicht für die Verteilung. `IpAddress` IPv6 Wenn Sie den Zugriff auf einige Inhalte anhand der IP-Adresse und der IPv6 Support-Anfragen für andere Inhalte einschränken möchten, können Sie zwei Distributionen erstellen. Weitere Informationen finden Sie unter Aktivieren IPv6 im Thema Referenz für alle Verteilungseinstellungen.

Beispiel-Richtlinienanweisungen für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet

Die folgenden Beispiel-Richtlinienanweisungen zeigen, wie der Zugriff auf eine bestimmte Datei, auf alle Dateien in einem Verzeichnis oder auf alle mit einer Schlüsselpaar-ID verknüpften Dateien kontrolliert wird. Die Beispiele zeigen auch, wie der Zugriff von einer einzelnen IP-Adresse oder einem Bereich von IP-Adressen kontrolliert wird und wie Sie verhindern, dass Benutzer das signierte Cookie nach einem festgelegten Datum und einer festgelegten Zeit verwenden.

Wenn Sie eines dieser Beispiele kopieren und einfügen, entfernen Sie alle Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen), ersetzen Sie die Werte durch Ihre eigenen Werte und fügen Sie nach der schließenden Klammer (}) ein Zeilenumbruchzeichen ein.

Weitere Informationen finden Sie unter Werte, die Sie in der Richtlinienanweisung für eine benutzerdefinierte Richtlinie für signierte Cookies angeben.

Themen

- Beispiel für eine Richtlinienerklärung: Greifen Sie von einem IP-Adressbereich aus auf eine Datei zu
- Beispiel für eine Richtlinienerklärung: Greifen Sie von einem IP-Adressbereich aus auf alle Dateien in einem Verzeichnis zu
- Beispiel für eine Richtlinienanweisung: Greifen Sie von einer IP-Adresse aus auf alle Dateien zu, die mit einer Schlüsselpaar-ID verknüpft sind

Beispiel für eine Richtlinienerklärung: Greifen Sie von einem IP-Adressbereich aus auf eine Datei zu

Die folgende Beispiel-Richtlinienanweisung in einem signierten Cookie legt fest, dass ein Benutzer bis zum 1. Januar 2023, 10:00 Uhr UTC, von IP-Adressen im Bereich `https://d111111abcdef8.cloudfront.net/game_download.zip` auf die Datei `192.0.2.0/24` zugreifen kann:

```
{
    "Statement": [
        {
            "Resource": "https://d111111abcdef8.cloudfront.net/game_download.zip",
            "Condition": {
                "IpAddress": {
                    "AWS:SourceIp": "192.0.2.0/24"
                },
```

```
                "DateLessThan": {
                    "AWS:EpochTime": 1357034400
                }
            }
        }
    ]
}
```

Beispiel für eine Richtlinienerklärung: Greifen Sie von einem IP-Adressbereich aus auf alle Dateien in einem Verzeichnis zu

Mit der folgenden Beispiel-Richtlinienanweisung können Sie signierte Cookies für jede Datei im `training`-Verzeichnis erstellen wie durch das Platzhalterzeichen * im `Resource`-Parameter verdeutlicht. Benutzer können bis zum 1. Januar 2013, 10:00 Uhr UTC, von IP-Adressen im Bereich `192.0.2.0/24` auf die Datei zugreifen:

```
{
    "Statement": [
        {
            "Resource": "https://d111111abcdef8.cloudfront.net/training/*",
            "Condition": {
                "IpAddress": {
                    "AWS:SourceIp": "192.0.2.0/24"
                },
                "DateLessThan": {
                    "AWS:EpochTime": 1357034400
                }
            }
        }
    ]
}
```

Jedes signierte Cookie, in dem Sie diese Richtlinie verwenden, enthält eine Basis-URL, die eine bestimmte Datei kennzeichnet, zum Beispiel:

`https://d111111abcdef8.cloudfront.net/training/orientation.pdf`

Beispiel für eine Richtlinienanweisung: Greifen Sie von einer IP-Adresse aus auf alle Dateien zu, die mit einer Schlüsselpaar-ID verknüpft sind

Mit der folgenden Beispiel-Richtlinienanweisung können Sie signierte Cookies für jede mit einer beliebigen Verteilung verknüpfte Datei einrichten wie durch das Platzhalterzeichen * im `Resource`-

Parameter verdeutlicht. Der Benutzer muss die IP-Adresse verwende `192.0.2.10/32`. (Der Wert `192.0.2.10/32` in CIDR-Notation bezieht sich auf eine einzelne IP-Adresse, `192.0.2.10`.) Die Dateien sind nur vom 1. Januar 2013, 10:00 Uhr UTC, bis zum 2. Januar 2013, 10:00 Uhr UTC, verfügbar:

```
{
    "Statement": [
        {
            "Resource": "https://*",
            "Condition": {
                "IpAddress": {
                    "AWS:SourceIp": "192.0.2.10/32"
                },
                "DateGreaterThan": {
                    "AWS:EpochTime": 1357034400
                },
                "DateLessThan": {
                    "AWS:EpochTime": 1357120800
                }
            }
        }
    ]
}
```

Jedes signierte Cookie, in dem Sie diese Richtlinie verwenden, enthält eine Basis-URL, die eine bestimmte Datei in einer bestimmten CloudFront Distribution identifiziert, zum Beispiel:

`https://d111111abcdef8.cloudfront.net/training/orientation.pdf`

Das signierte Cookie enthält auch eine Schlüsselpaar-ID, die mit einem vertrauenswürdigen Aussteller in der Verteilung (d111111abcdef8.cloudfront.net) verknüpft werden muss, die Sie in der Basis-URL angeben.

Erstellen Sie eine Signatur für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet

Bei der Signatur für ein signiertes Cookie, das eine benutzerdefinierte Richtlinie verwendet, handelt es sich um eine gehashte, signierte und Base64-codierte Version der Richtlinienanweisung.

Weitere Informationen und Beispiele für das Hashing, Signieren und Codieren der Richtlinienanweisung finden Sie unter:

- Linux-Befehle und OpenSSL für Base64-Kodierung und Verschlüsselung

- Code-Beispiele für das Erstellen einer Signatur für eine signierte URL

So erstellen Sie eine Signatur für ein signiertes Cookie mithilfe einer benutzerdefinierten Richtlinie

1. Verwenden Sie die SHA-1-Hash-Funktion und RSA, um die im Verfahren So erstellen Sie die Richtlinienanweisung für eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet erstellte JSON-Richtlinienanweisung zu hashen und zu signieren. Verwenden Sie die Version der Richtlinienanweisung, die keine Leerzeichen mehr enthält, die aber noch nicht Base64-codiert wurde.

   Verwenden Sie für den privaten Schlüssel, der für die Hash-Funktion erforderlich ist, einen privaten Schlüssel, dessen öffentlicher Schlüssel sich in einer aktiven vertrauenswürdigen Schlüsselgruppe für die Verteilung befindet.

   > **Note**
   >
   > Die Methode, die Sie zum Hashen und Signieren der Richtlinienanweisung verwenden, ist abhängig von Ihrer Programmiersprache und Plattform. Einen Beispiel-Code finden Sie unter Code-Beispiele für das Erstellen einer Signatur für eine signierte URL.

2. Entfernen Sie Leerzeichen (einschließlich Tabulatoren und Zeilenumbruchzeichen) aus der Hash-Zeichenfolge und der signierten Zeichenfolge.
3. Nehmen Sie eine Base64-Codierung der Zeichenfolge mithilfe von MIME-Base64-Codierung vor. Weitere Informationen finden Sie in Abschnitt 6.8, Base64 Content-Transfer-Encoding in RFC 2045, MIME (Multipurpose Internet Mail Extensions), Teil 1: Format von Internet-Nachrichtentexten.
4. Ersetzen Sie Zeichen, die in einer URL-Abfragezeichenfolge nicht gültig sind, durch gültige Zeichen. In der folgenden Tabelle sind ungültige und gültige Zeichen aufgelistet.

Ersetzen Sie diese ungültigen Zeichen	Durch diese gültigen Zeichen
+	- (Bindestrich)
=	_ (Unterstrich)

Ersetzen Sie diese ungültigen Zeichen	Durch diese gültigen Zeichen
/	~ (Tilde)

5. Fügen Sie den resultierenden Wert im `Set-Cookie`-Header für das `CloudFront-Signature=`-Name-Wert-Paar ein und kehren Sie zu So richten Sie ein signiertes Cookies mit einer benutzerdefinierten Richtlinie ein zurück, um den `Set-Cookie`-Header für `CloudFront-Key-Pair-Id` hinzuzufügen.

Erstellen Sie signierte Cookies mit PHP

Das folgende Codebeispiel ähnelt dem Beispiel insofern, als es einen Link zu einem Video erstellt. Erstellen einer URL-Signatur mit PHP Anstatt jedoch die URL im Code zu signieren, signiert dieses Beispiel die Cookies mit der `create_signed_cookies()` Funktion. Der clientseitige Player verwendet die Cookies, um jede Anfrage an die Distribution zu authentifizieren. CloudFront

Dieser Ansatz ist nützlich, um Inhalte wie HTTP Live Streaming (HLS) oder Dynamic Adaptive Streaming over HTTP (DASH) zu streamen, bei denen der Client mehrere Anfragen stellen muss, um das Manifest, die Segmente und die zugehörigen Wiedergabe-Assets abzurufen. Mithilfe signierter Cookies kann der Client jede Anfrage authentifizieren, ohne für jedes Segment eine neue signierte URL generieren zu müssen.

> **Note**
>
> - Das Erstellen einer URL-Signatur ist nur ein Teil des Prozesses der Bereitstellung privater Inhalte mithilfe signierter Cookies. Weitere Informationen finden Sie unter Verwenden Sie signierte Cookies.

Themen

- Erstellen Sie die RSA SHA-1-Signatur
- Erstellen Sie die signierten Cookies
- Vollständiger Code

In den folgenden Abschnitten wird das Codebeispiel in einzelne Teile unterteilt. Das vollständige Codebeispiel finden Sie unten.

Erstellen Sie die RSA SHA-1-Signatur

Dieses Codebeispiel macht Folgendes:

1. Die Funktion `rsa_sha1_sign` hasht und signiert die Grundsatzerklärung. Die erforderlichen Argumente sind eine Richtlinienanweisung und der private Schlüssel, der einem öffentlichen Schlüssel entspricht, der sich in einer vertrauenswürdigen Schlüsselgruppe für Ihre Verteilung befindet.
2. Als Nächstes erstellt die Funktion `url_safe_base64_encode` eine URL-sichere Version der Signatur.

```
function rsa_sha1_sign($policy, $private_key_filename) {
    $signature = "";
    $fp = fopen($private_key_filename, "r");
    $priv_key = fread($fp, 8192);
    fclose($fp);
    $pkeyid = openssl_get_privatekey($priv_key);
    openssl_sign($policy, $signature, $pkeyid);
    openssl_free_key($pkeyid);
    return $signature;
}

function url_safe_base64_encode($value) {
    $encoded = base64_encode($value);
    return str_replace(
        array('+', '=', '/'),
        array('-', '_', '~'),
        $encoded);
}
```

Erstellen Sie die signierten Cookies

Der folgende Code konstruiert und erstellt die signierten Cookies unter Verwendung der folgenden Cookie-Attribute: `CloudFront-Expires`CloudFront-Signature, und`CloudFront-Key-Pair-Id`. Der Code verwendet eine benutzerdefinierte Richtlinie.

```
function create_signed_cookies($resource, $private_key_filename, $key_pair_id,
 $expires, $client_ip = null) {
   $policy = array(
       'Statement' => array(
           array(
```

```
                'Resource' => $resource,
                'Condition' => array(
                    'DateLessThan' => array('AWS:EpochTime' => $expires)
                )
            )
        )
    );

    if ($client_ip) {
        $policy['Statement'][0]['Condition']['IpAddress'] = array('AWS:SourceIp' =>
 $client_ip . '/32');
    }

    $policy = json_encode($policy);
    $encoded_policy = url_safe_base64_encode($policy);
    $signature = rsa_sha1_sign($policy, $private_key_filename);
    $encoded_signature = url_safe_base64_encode($signature);

    return array(
        'CloudFront-Policy' => $encoded_policy,
        'CloudFront-Signature' => $encoded_signature,
        'CloudFront-Key-Pair-Id' => $key_pair_id
    );
}
```

Weitere Informationen finden Sie unter Legen Sie signierte Cookies mithilfe einer benutzerdefinierten Richtlinie fest.

Vollständiger Code

Der folgende Beispielcode bietet eine vollständige Demonstration der Erstellung CloudFront signierter Cookies mit PHP. Sie können das vollständige Beispiel aus der Datei demo-php.zip herunterladen.

Im folgenden Beispiel können Sie das `$policy Condition` Element so ändern, dass IPv4 sowohl IPv6 Adressbereiche als auch Adressbereiche zulässig sind. Ein Beispiel finden Sie unter IPv6 Adressen in IAM-Richtlinien verwenden im Amazon Simple Storage Service-Benutzerhandbuch.

```
<?php

function rsa_sha1_sign($policy, $private_key_filename) {
    $signature = "";
    $fp = fopen($private_key_filename, "r");
    $priv_key = fread($fp, 8192);
```

```
    fclose($fp);
    $pkeyid = openssl_get_privatekey($priv_key);
    openssl_sign($policy, $signature, $pkeyid);
    openssl_free_key($pkeyid);
    return $signature;
}

function url_safe_base64_encode($value) {
    $encoded = base64_encode($value);
    return str_replace(
        array('+', '=', '/'),
        array('-', '_', '~'),
        $encoded);
}

function create_signed_cookies($resource, $private_key_filename, $key_pair_id,
 $expires, $client_ip = null) {
    $policy = array(
        'Statement' => array(
            array(
                'Resource' => $resource,
                'Condition' => array(
                    'DateLessThan' => array('AWS:EpochTime' => $expires)
                )
            )
        )
    );

    if ($client_ip) {
        $policy['Statement'][0]['Condition']['IpAddress'] = array('AWS:SourceIp' =>
 $client_ip . '/32');
    }

    $policy = json_encode($policy);
    $encoded_policy = url_safe_base64_encode($policy);
    $signature = rsa_sha1_sign($policy, $private_key_filename);
    $encoded_signature = url_safe_base64_encode($signature);

    return array(
        'CloudFront-Policy' => $encoded_policy,
        'CloudFront-Signature' => $encoded_signature,
        'CloudFront-Key-Pair-Id' => $key_pair_id
    );
}
```

```
$private_key_filename = '/home/test/secure/example-priv-key.pem';
$key_pair_id = 'K2JCJMDEHXQW5F';
$base_url = 'https://d1234.cloudfront.net';

$expires = time() + 3600; // 1 hour from now

// Get the viewer real IP from the x-forward-for header as $_SERVER['REMOTE_ADDR']
 will return viewer facing IP. An alternative option is to use CloudFront-Viewer-
Address header. Note that this header is a trusted CloudFront immutable header. Example
 format: IP:PORT ("CloudFront-Viewer-Address": "1.2.3.4:12345")
$client_ip = $_SERVER['HTTP_X_FORWARDED_FOR'];


// For HLS manifest and segments (using wildcard)
$hls_resource = $base_url . '/sign/*';
$signed_cookies = create_signed_cookies($hls_resource, $private_key_filename,
 $key_pair_id, $expires, $client_ip);

// Set the cookies
$cookie_domain = parse_url($base_url, PHP_URL_HOST);
foreach ($signed_cookies as $name => $value) {
    setcookie($name, $value, $expires, '/', $cookie_domain, true, true);
}

?>

<!DOCTYPE html>
<html>
<head>
    <title>CloudFront Signed HLS Stream with Cookies</title>
</head>
<body>
    <h1>Amazon CloudFront Signed HLS Stream with Cookies</h1>
    <h2>Expires at <?php echo gmdate('Y-m-d H:i:s T', $expires); ?> only viewable by IP
 <?php echo $client_ip; ?></h2>

    <div id='hls-video'>
        <video id="video" width="640" height="360" controls></video>
    </div>

    <script src="https://cdn.jsdelivr.net/npm/hls.js@latest"></script>
```

```
    <script>
        var video = document.getElementById('video');
        var manifestUrl = '<?php echo $base_url; ?>/sign/manifest.m3u8';

        if (Hls.isSupported()) {
            var hls = new Hls();
            hls.loadSource(manifestUrl);
            hls.attachMedia(video);
        }
        else if (video.canPlayType('application/vnd.apple.mpegurl')) {
            video.src = manifestUrl;
        }
    </script>
</body>
</html>
```

Anstatt signierte Cookies zu verwenden, können Sie signierte URLs Cookies verwenden. Weitere Informationen finden Sie unter Erstellen einer URL-Signatur mit PHP.

Linux-Befehle und OpenSSL für Base64-Kodierung und Verschlüsselung

Sie können den folgenden Linux-Befehlszeilenbefehl und OpenSSL verwenden, um die Richtlinienanweisung zu hashen und zu signieren, die Signatur in Base64 zu codieren und Zeichen, die in URL-Abfragezeichenfolgeparametern nicht gültig sind, durch gültige zu ersetzen.

Informationen zu OpenSSL finden Sie unter https://www.openssl.org.

```
cat policy | tr -d "\n" | tr -d " \t\n\r" | openssl sha1 -sign private_key.pem |
 openssl base64 -A | tr -- '+=/' '-_~'
```

Beim vorhergehenden Befehl:

- `cat`liest die Datei `policy`
- `tr -d "\n" | tr -d " \t\n\r"`entfernt die Leerzeichen und das Zeilenumbruchzeichen, die von hinzugefügt wurden `cat`
- OpenSSL hasht die Datei mit SHA-1 und signiert sie mit RSA und der privaten Schlüsseldatei `private_key.pem`
- OpenSSL Base64-kodiert die gehashte und signierte Richtlinienerklärung
- `tr`ersetzt Zeichen, die in URL-Abfragezeichenfolgenparametern nicht gültig sind, durch gültige Zeichen

Weitere Codebeispiele, die das Erstellen einer Signatur veranschaulichen, finden Sie unterCode-Beispiele für das Erstellen einer Signatur für eine signierte URL.

Code-Beispiele für das Erstellen einer Signatur für eine signierte URL

Dieser Abschnitt enthält Anwendungsbeispiele zum Herunterladen, die zeigen, wie Signaturen für signiert erstellt URLs werden. Beispiele sind in Perl, PHP, C# und Java verfügbar. Sie können jedes der Beispiele verwenden, um signierte zu erstellen URLs. Das Perl-Skript wird auf Linux-/macOS-Plattformen ausgeführt. Das PHP-Beispiel funktioniert auf allen Servern, auf denen PHP ausgeführt wird. Das C#-Beispiel verwendet das .NET Framework.

Beispielcode in JavaScript (Node.js) finden Sie unter Creating Amazon CloudFront Signed URLs in Node.js im AWS Developer Blog.

Beispielcode in Python finden Sie unter Generate a signed URL for Amazon CloudFront in der AWS SDK for Python (Boto3) API-Referenz und diesen Beispielcode im GitHub Boto3-Repository.

Themen

- Erstellen einer URL-Signatur mit Perl
- Erstellen einer URL-Signatur mit PHP
- Erstellen einer URL-Signatur mithilfe von C# und dem .NET Framework
- Erstellen einer URL-Signatur mit Java

Erstellen einer URL-Signatur mit Perl

Dieser Abschnitt enthält ein Perl-Skript für Linux/Mac Plattformen, mit dem Sie die Signatur für private Inhalte erstellen können. Um die Signatur zu erstellen, führen Sie das Skript mit Befehlszeilenargumenten aus, die die CloudFront URL, den Pfad zum privaten Schlüssel des Unterzeichners, die Schlüssel-ID und ein Ablaufdatum für die URL angeben. Das Tool kann auch signiert dekodieren. URLs

> **Note**
>
> Das Erstellen einer URL-Signatur ist nur ein Teil der Bereitstellung privater Inhalte über eine signierte URL. Weitere Informationen zu diesem end-to-end Prozess finden Sie unterVerwenden Sie signierte URLs.

Themen

Quelle für das Perl-Skript zum Erstellen einer signierten URL

Der folgende Perl-Quellcode kann verwendet werden, um eine signierte URL für CloudFront zu erstellen. Kommentare im Code beinhalten Informationen zu den Befehlszeilen-Switches und den Features des Tools.

```
#!/usr/bin/perl -w

# Copyright 2008 Amazon Technologies, Inc.  Licensed under the Apache License, Version
 2.0 (the "License");
# you may not use this file except in compliance with the License. You may obtain a
 copy of the License at:
#
# https://aws.amazon.com/apache2.0
#
# This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 KIND, either express or implied.
# See the License for the specific language governing permissions and limitations under
 the License.

=head1 cfsign.pl

cfsign.pl - A tool to generate and verify Amazon CloudFront signed URLs

=head1 SYNOPSIS

This script uses an existing RSA key pair to sign and verify Amazon CloudFront signed
 URLs

View the script source for details as to which CPAN packages are required beforehand.

For help, try:

cfsign.pl --help

URL signing examples:

cfsign.pl --action encode --url https://images.my-website.com/gallery1.zip --policy
 sample_policy.json --private-key privkey.pem --key-pair-id mykey
```

```
cfsign.pl --action encode --url https://images.my-website.com/gallery1.zip --expires
 1257439868 --private-key privkey.pem --key-pair-id mykey

URL decode example:

cfsign.pl --action decode --url "http//mydist.cloudfront.net/?Signature=AGO-
PgxkYo99MkJFHvjfGXjG1QDEXeaDb4Qtzmy85wqyJjK7eKojQWa4BCRcow__&Policy=eyJTdGF0ZW1lbnQiOlt7IlJlc29
Pair-Id=mykey"


To generate an RSA key pair, you can use openssl and the following commands:

# Generate a 2048 bit key pair
openssl genrsa -out private-key.pem 2048
openssl rsa -in private-key.pem -pubout -out public-key.pem


=head1 OPTIONS

=over 8

=item B<--help>

Print a help message and exits.

=item B<--action> [action]

The action to execute.  action can be one of:

  encode - Generate a signed URL (using a canned policy or a user policy)
  decode - Decode a signed URL

=item B<--url>

The URL to en/decode

=item B<--stream>

The stream to en/decode

=item B<--private-key>

The path to your private key.
```

```
=item B<--key-pair-id>

The key pair identifier.

=item B<--policy>

The CloudFront policy document.

=item B<--expires>

The Unix epoch time when the URL is to expire. If both this option and
the --policy option are specified, --policy will be used. Otherwise, this
option alone will use a canned policy.

=back

=cut

use strict;
use warnings;

# you might need to use CPAN to get these modules.
# run perl -MCPAN -e "install <module>" to get them.
# The openssl command line will also need to be in your $PATH.
use File::Temp qw/tempfile/;
use File::Slurp;
use Getopt::Long;
use IPC::Open2;
use MIME::Base64 qw(encode_base64 decode_base64);
use Pod::Usage;
use URI;

my $CANNED_POLICY
    = '{"Statement":[{"Resource":"<RESOURCE>","Condition":{"DateLessThan":
{"AWS:EpochTime":<EXPIRES>}}}]}';

my $POLICY_PARAM      = "Policy";
my $EXPIRES_PARAM     = "Expires";
my $SIGNATURE_PARAM   = "Signature";
my $KEY_PAIR_ID_PARAM = "Key-Pair-Id";

my $verbose = 0;
my $policy_filename = "";
```

```
my $expires_epoch = 0;
my $action = "";
my $help = 0;
my $key_pair_id = "";
my $url = "";
my $stream = "";
my $private_key_filename = "";

my $result = GetOptions("action=s"      => \$action,
                        "policy=s"      => \$policy_filename,
                        "expires=i"     => \$expires_epoch,
                        "private-key=s" => \$private_key_filename,
                        "key-pair-id=s" => \$key_pair_id,
                        "verbose"       => \$verbose,
                        "help"          => \$help,
                        "url=s"         => \$url,
                        "stream=s"      => \$stream,
                    );

if ($help or !$result) {
    pod2usage(1);
    exit;
}

if ($url eq "" and $stream eq "") {
    print STDERR "Must include a stream or a URL to encode or decode with the --stream
 or --url option\n";
    exit;
}

if ($url ne "" and $stream ne "") {
    print STDERR "Only one of --url and --stream may be specified\n";
    exit;
}

if ($url ne "" and !is_url_valid($url)) {
    exit;
}

if ($stream ne "") {
    exit unless is_stream_valid($stream);

    # The signing mechanism is identical, so from here on just pretend we're
    # dealing with a URL
```

```
    $url = $stream;
}

if ($action eq "encode") {
    # The encode action will generate a private content URL given a base URL,
    # a policy file (or an expires timestamp) and a key pair id parameter
    my $private_key;
    my $public_key;
    my $public_key_file;

    my $policy;
    if ($policy_filename eq "") {
        if ($expires_epoch == 0) {
            print STDERR "Must include policy filename with --policy argument or an
 expires" .
                         "time using --expires\n";
        }

        $policy = $CANNED_POLICY;
        $policy =~ s/<EXPIRES>/$expires_epoch/g;
        $policy =~ s/<RESOURCE>/$url/g;
    } else {
        if (! -e $policy_filename) {
            print STDERR "Policy file $policy_filename does not exist\n";
            exit;
        }
        $expires_epoch = 0; # ignore if set
        $policy = read_file($policy_filename);
    }

    if ($private_key_filename eq "") {
        print STDERR "You must specific the path to your private key file with --
private-key\n";
        exit;
    }

    if (! -e $private_key_filename) {
        print STDERR "Private key file $private_key_filename does not exist\n";
        exit;
    }

    if ($key_pair_id eq "") {
        print STDERR "You must specify a key pair id with --key-pair-id\n";
        exit;
```

```
    }

    my $encoded_policy = url_safe_base64_encode($policy);
    my $signature = rsa_sha1_sign($policy, $private_key_filename);
    my $encoded_signature = url_safe_base64_encode($signature);

    my $generated_url = create_url($url, $encoded_policy, $encoded_signature,
 $key_pair_id, $expires_epoch);


    if ($stream ne "") {
        print "Encoded stream (for use within a swf):\n" . $generated_url . "\n";
        print "Encoded and escaped stream (for use on a webpage):\n" .
 escape_url_for_webpage($generated_url) . "\n";
    } else {
        print "Encoded URL:\n" . $generated_url . "\n";
    }
} elsif ($action eq "decode") {
    my $decoded = decode_url($url);
    if (!$decoded) {
        print STDERR "Improperly formed URL\n";
        exit;
    }

    print_decoded_url($decoded);
} else {
    # No action specified, print help.  But only if this is run as a program (caller
 will be empty)
    pod2usage(1) unless caller();
}

# Decode a private content URL into its component parts
sub decode_url {
    my $url = shift;

    if ($url =~ /(.*)\?(.*)/) {
        my $base_url = $1;
        my $params = $2;

        my @unparsed_params = split(/&/, $params);
        my %params = ();
        foreach my $param (@unparsed_params) {
            my ($key, $val) = split(/=/, $param);
            $params{$key} = $val;
```

```
        }

        my $encoded_signature = "";
        if (exists $params{$SIGNATURE_PARAM}) {
            $encoded_signature = $params{"Signature"};
        } else {
            print STDERR "Missing Signature URL parameter\n";
            return 0;
        }

        my $encoded_policy = "";
        if (exists $params{$POLICY_PARAM}) {
            $encoded_policy = $params{$POLICY_PARAM};
        } else {
            if (!exists $params{$EXPIRES_PARAM}) {
                print STDERR "Either the Policy or Expires URL parameter needs to be
specified\n";
                return 0;
            }

            my $expires = $params{$EXPIRES_PARAM};

            my $policy = $CANNED_POLICY;
            $policy =~ s/<EXPIRES>/$expires/g;

            my $url_without_cf_params = $url;
            $url_without_cf_params =~ s/$SIGNATURE_PARAM=[^&]*&?//g;
            $url_without_cf_params =~ s/$POLICY_PARAM=[^&]*&?//g;
            $url_without_cf_params =~ s/$EXPIRES_PARAM=[^&]*&?//g;
            $url_without_cf_params =~ s/$KEY_PAIR_ID_PARAM=[^&]*&?//g;

            if ($url_without_cf_params =~ /(.*)\?$/) {
                $url_without_cf_params = $1;
            }

            $policy =~ s/<RESOURCE>/$url_without_cf_params/g;

            $encoded_policy = url_safe_base64_encode($policy);
        }

        my $key = "";
        if (exists $params{$KEY_PAIR_ID_PARAM}) {
            $key = $params{$KEY_PAIR_ID_PARAM};
        } else {
```

```
            print STDERR "Missing $KEY_PAIR_ID_PARAM parameter\n";
            return 0;
        }

        my $policy = url_safe_base64_decode($encoded_policy);

        my %ret = ();
        $ret{"base_url"} = $base_url;
        $ret{"policy"} = $policy;
        $ret{"key"} = $key;

        return \%ret;
    } else {
        return 0;
    }
}

# Print a decoded URL out
sub print_decoded_url {
    my $decoded = shift;

    print "Base URL: \n" . $decoded->{"base_url"} . "\n";
    print "Policy: \n" . $decoded->{"policy"} . "\n";
    print "Key: \n" . $decoded->{"key"} . "\n";
}

# Encode a string with base 64 encoding and replace some invalid URL characters
sub url_safe_base64_encode {
    my ($value) = @_;

    my $result = encode_base64($value);
    $result =~ tr|+=/|-_~|;

    return $result;
}

# Decode a string with base 64 encoding.  URL-decode the string first
# followed by reversing any special character ("+=/") translation.
sub url_safe_base64_decode {
    my ($value) = @_;

    $value =~ s/%([0-9A-Fa-f]{2})/chr(hex($1))/eg;
    $value =~ tr|-_~|+=/|;
```

```
    my $result = decode_base64($value);

    return $result;
}

# Create a private content URL
sub create_url {
    my ($path, $policy, $signature, $key_pair_id, $expires) = @_;

    my $result;
    my $separator = $path =~ /\?/ ? '&' : '?';
    if ($expires) {
        $result = "$path$separator$EXPIRES_PARAM=$expires&$SIGNATURE_PARAM=$signature&
$KEY_PAIR_ID_PARAM=$key_pair_id";
    } else {
        $result = "$path$separator$POLICY_PARAM=$policy&$SIGNATURE_PARAM=$signature&
$KEY_PAIR_ID_PARAM=$key_pair_id";
    }
    $result =~ s/\n//g;

    return $result;
}

# Sign a document with given private key file.
# The first argument is the document to sign
# The second argument is the name of the private key file
sub rsa_sha1_sign {
    my ($to_sign, $pvkFile) = @_;
    print "openssl sha1 -sign $pvkFile $to_sign\n";

    return write_to_program($pvkFile, $to_sign);
}

# Helper function to write data to a program
sub write_to_program {
my ($keyfile, $data) = @_;
unlink "temp_policy.dat" if (-e "temp_policy.dat");
unlink "temp_sign.dat" if (-e "temp_sign.dat");

write_file("temp_policy.dat", $data);

system("openssl dgst -sha1 -sign \"$keyfile\" -out temp_sign.dat temp_policy.dat");

my $output = read_file("temp_sign.dat");
```

```
    return $output;
}

# Read a file into a string and return the string
sub read_file {
    my ($file) = @_;

    open(INFILE, "<$file") or die("Failed to open $file: $!");
    my $str = join('', <INFILE>);
    close INFILE;

    return $str;
}

sub is_url_valid {
    my ($url) = @_;

    # HTTP distributions start with http[s]:// and are the correct thing to sign
    if ($url =~ /^https?:\/\//) {
        return 1;
    } else {
        print STDERR "CloudFront requires absolute URLs for HTTP distributions\n";
        return 0;
    }
}

sub is_stream_valid {
    my ($stream) = @_;

    if ($stream =~ /^rtmp:\/\// or $stream =~ /^\/?cfx\/st/) {
        print STDERR "Streaming distributions require that only the stream name is
 signed.\n";
        print STDERR "The stream name is everything after, but not including, cfx/st/
\n";
        return 0;
    } else {
        return 1;
    }
}

# flash requires that the query parameters in the stream name are url
# encoded when passed in through javascript, etc.  This sub handles the minimal
# required url encoding.
```

```
sub escape_url_for_webpage {
    my ($url) = @_;

    $url =~ s/\?/%3F/g;
    $url =~ s/=/%3D/g;
    $url =~ s/&/%26/g;

    return $url;
}

1;
```

Erstellen einer URL-Signatur mit PHP

Jeder Webserver, auf dem PHP ausgeführt wird, kann diesen PHP-Beispielcode verwenden, um Richtlinienerklärungen und Signaturen für private CloudFront Distributionen zu erstellen. Das vollständige Beispiel erstellt eine funktionierende Webseite mit signierten URL-Links, die einen Videostream per CloudFront Streaming abspielen. Sie können das vollständige Beispiel aus der Datei demo-php.zip herunterladen.

> **Hinweise**
>
> - Das Erstellen einer URL-Signatur ist nur ein Teil der Bereitstellung privater Inhalte über eine signierte URL. Weitere Informationen zum gesamten Prozess finden Sie unter Verwenden Sie signierte URLs.
> - Sie können auch signierte Dateien erstellen, URLs indem Sie die `UrlSigner` Klasse in der verwenden AWS SDK für PHP. Weitere Informationen finden Sie unter Klasse UrlSigner in der AWS SDK für PHP API-Referenz.

Themen

In den folgenden Abschnitten wird das Codebeispiel in einzelne Teile unterteilt. Die finden Sie Beispiel für vollständigen Code unten.

Erstellen Sie die RSA SHA-1-Signatur

Dieses Codebeispiel macht Folgendes:

- Die Funktion `rsa_sha1_sign` hasht und signiert die Grundsatzerklärung. Die erforderlichen Argumente sind eine Richtlinienanweisung und der private Schlüssel, der einem öffentlichen Schlüssel entspricht, der sich in einer vertrauenswürdigen Schlüsselgruppe für Ihre Verteilung befindet.
- Als Nächstes erstellt die Funktion `url_safe_base64_encode` eine URL-sichere Version der Signatur.

```
function rsa_sha1_sign($policy, $private_key_filename) {
    $signature = "";

    // load the private key
    $fp = fopen($private_key_filename, "r");
    $priv_key = fread($fp, 8192);
    fclose($fp);
    $pkeyid = openssl_get_privatekey($priv_key);

    // compute signature
    openssl_sign($policy, $signature, $pkeyid);

    // free the key from memory
    openssl_free_key($pkeyid);

    return $signature;
}

function url_safe_base64_encode($value) {
    $encoded = base64_encode($value);
    // replace unsafe characters +, = and / with
    // the safe characters -, _ and ~
    return str_replace(
        array('+', '=', '/'),
        array('-', '_', '~'),
        $encoded);
}
```

Der folgende Codeausschnitt verwendet die Funktionen `get_canned_policy_stream_name()` und `get_custom_policy_stream_name()` erstellt eine vorgefertigte und benutzerdefinierte Richtlinie. CloudFront verwendet die Richtlinien, um die URL für das Streamen des Videos zu erstellen, einschließlich der Angabe der Ablaufzeit.

Anschließend können Sie anhand einer vorgefertigten Richtlinie oder einer benutzerdefinierten Richtlinie festlegen, wie der Zugriff auf Ihre Inhalte verwaltet werden soll. Weitere Informationen darüber, welche Option Sie wählen sollten, finden Sie im Entscheiden Sie sich dafür, vordefinierte oder benutzerdefinierte Richtlinien für signierte Richtlinien zu verwenden URLs Abschnitt.

Erstellen Sie eine vordefinierte Richtlinie

Der folgende Beispielcode erstellt eine vordefinierte Richtlinienanweisung für die Signatur.

> **Note**
>
> Die `$expires` Variable ist ein date/time Stempel, der eine Ganzzahl sein muss, keine Zeichenfolge.

```
function get_canned_policy_stream_name($video_path, $private_key_filename,
 $key_pair_id, $expires) {
    // this policy is well known by CloudFront, but you still need to sign it, since it
 contains your parameters
    $canned_policy = '{"Statement":[{"Resource":"' . $video_path . '","Condition":
{"DateLessThan":{"AWS:EpochTime":'. $expires . '}}}]}';
    // the policy contains characters that cannot be part of a URL, so we base64 encode
 it
    $encoded_policy = url_safe_base64_encode($canned_policy);
    // sign the original policy, not the encoded version
    $signature = rsa_sha1_sign($canned_policy, $private_key_filename);
    // make the signature safe to be included in a URL
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, null, $encoded_signature,
 $key_pair_id, $expires);
    // URL-encode the query string characters
    return $stream_name;
}
```

Weitere Informationen zu vordefinierten Richtlinien finden Sie unter Erstellen Sie eine signierte URL mithilfe einer vordefinierten Richtlinie.

Erstellen einer benutzerdefinierten Richtlinie

Der folgende Beispielcode erstellt eine benutzerdefinierte Richtlinienanweisung für die Signatur.

```
function get_custom_policy_stream_name($video_path, $private_key_filename,
 $key_pair_id, $policy) {
    // the policy contains characters that cannot be part of a URL, so we base64 encode
 it
    $encoded_policy = url_safe_base64_encode($policy);
    // sign the original policy, not the encoded version
    $signature = rsa_sha1_sign($policy, $private_key_filename);
    // make the signature safe to be included in a URL
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, $encoded_policy, $encoded_signature,
 $key_pair_id, null);
    // URL-encode the query string characters
    return $stream_name;
}
```

Weitere Informationen zu benutzerdefinierten Richtlinien finden Sie unter Erstellen Sie eine signierte URL mithilfe einer benutzerdefinierten Richtlinie.

Beispiel für vollständigen Code

Der folgende Beispielcode bietet eine vollständige Demonstration der Erstellung CloudFront signierter Dateien URLs mit PHP. Sie können das vollständige Beispiel aus der Datei demo-php.zip herunterladen.

Im folgenden Beispiel können Sie das `$policy` `Condition` Element so ändern, dass IPv4 sowohl IPv6 Adressbereiche als auch Adressbereiche zulässig sind. Ein Beispiel finden Sie unter IPv6Adressen in IAM-Richtlinien verwenden im Amazon Simple Storage Service-Benutzerhandbuch.

```
<?php

function rsa_sha1_sign($policy, $private_key_filename) {
    $signature = "";

    // load the private key
```

```
    $fp = fopen($private_key_filename, "r");
    $priv_key = fread($fp, 8192);
    fclose($fp);
    $pkeyid = openssl_get_privatekey($priv_key);

    // compute signature
    openssl_sign($policy, $signature, $pkeyid);

    // free the key from memory
    openssl_free_key($pkeyid);

    return $signature;
}

function url_safe_base64_encode($value) {
    $encoded = base64_encode($value);
    // replace unsafe characters +, = and / with the safe characters -, _ and ~
    return str_replace(
        array('+', '=', '/'),
        array('-', '_', '~'),
        $encoded);
}

function create_stream_name($stream, $policy, $signature, $key_pair_id, $expires) {
    $result = $stream;
    // if the stream already contains query parameters, attach the new query parameters
 to the end
    // otherwise, add the query parameters
    $separator = strpos($stream, '?') == FALSE ? '?' : '&';
    // the presence of an expires time means we're using a canned policy
    if($expires) {
        $result .= $separator . "Expires=" . $expires . "&Signature=" . $signature .
 "&Key-Pair-Id=" . $key_pair_id;
    }
    // not using a canned policy, include the policy itself in the stream name
    else {
        $result .= $separator . "Policy=" . $policy . "&Signature=" . $signature .
 "&Key-Pair-Id=" . $key_pair_id;
    }

    // new lines would break us, so remove them
    return str_replace('\n', '', $result);
}
```

```
function get_canned_policy_stream_name($video_path, $private_key_filename,
 $key_pair_id, $expires) {
    // this policy is well known by CloudFront, but you still need to sign it, since it
 contains your parameters
    $canned_policy = '{"Statement":[{"Resource":"' . $video_path . '","Condition":
{"DateLessThan":{"AWS:EpochTime":'. $expires . '}}}]}';
    // the policy contains characters that cannot be part of a URL, so we base64 encode
 it
    $encoded_policy = url_safe_base64_encode($canned_policy);
    // sign the original policy, not the encoded version
    $signature = rsa_sha1_sign($canned_policy, $private_key_filename);
    // make the signature safe to be included in a URL
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, null, $encoded_signature,
 $key_pair_id, $expires);
    // URL-encode the query string characters
    return $stream_name;
}

function get_custom_policy_stream_name($video_path, $private_key_filename,
 $key_pair_id, $policy) {
    // the policy contains characters that cannot be part of a URL, so we base64 encode
 it
    $encoded_policy = url_safe_base64_encode($policy);
    // sign the original policy, not the encoded version
    $signature = rsa_sha1_sign($policy, $private_key_filename);
    // make the signature safe to be included in a URL
    $encoded_signature = url_safe_base64_encode($signature);

    // combine the above into a stream name
    $stream_name = create_stream_name($video_path, $encoded_policy, $encoded_signature,
 $key_pair_id, null);
    // URL-encode the query string characters
    return $stream_name;
}


// Path to your private key.  Be very careful that this file is not accessible
// from the web!

$private_key_filename = '/home/test/secure/example-priv-key.pem';
```

```
$key_pair_id = 'K2JCJMDEHXQW5F';

// Make sure you have "Restrict viewer access" enabled on this path behaviour and using
 the above Trusted key groups (recommended).
$video_path = 'https://example.com/secure/example.mp4';

$expires = time() + 300; // 5 min from now
$canned_policy_stream_name = get_canned_policy_stream_name($video_path,
 $private_key_filename, $key_pair_id, $expires);

// Get the viewer real IP from the x-forward-for header as $_SERVER['REMOTE_ADDR']
 will return viewer facing IP. An alternative option is to use CloudFront-Viewer-
Address header. Note that this header is a trusted CloudFront immutable header. Example
 format: IP:PORT ("CloudFront-Viewer-Address": "1.2.3.4:12345")
$client_ip = $_SERVER['HTTP_X_FORWARDED_FOR'];
$policy =
'{'.
    '"Statement":['.
        '{'.
            '"Resource":"'. $video_path . '",'.
            '"Condition":{'.
                '"IpAddress":{"AWS:SourceIp":"' . $client_ip . '/32"},'.
                '"DateLessThan":{"AWS:EpochTime":' . $expires . '}'.
            '}'.
        '}'.
    ']' .
    '}';
$custom_policy_stream_name = get_custom_policy_stream_name($video_path,
 $private_key_filename, $key_pair_id, $policy);

?>

<html>

<head>
    <title>CloudFront</title>
</head>

<body>
    <h1>Amazon CloudFront</h1>
    <h2>Canned Policy</h2>
    <h3>Expires at <?php echo gmdate('Y-m-d H:i:s T', $expires); ?></h3>
    <br />
```

```
    <div id='canned'>The canned policy video will be here: <br>

        <video width="640" height="360" autoplay muted controls>
        <source src="<?php echo $canned_policy_stream_name; ?>" type="video/mp4">
        Your browser does not support the video tag.
        </video>
    </div>

    <h2>Custom Policy</h2>
    <h3>Expires at <?php echo gmdate('Y-m-d H:i:s T', $expires); ?> only viewable by IP
 <?php echo $client_ip; ?></h3>
    <div id='custom'>The custom policy video will be here: <br>

         <video width="640" height="360" autoplay muted controls>
         <source src="<?php echo $custom_policy_stream_name; ?>" type="video/mp4">
         Your browser does not support the video tag.
        </video>
    </div>

</body>

</html>
```

Weitere Beispiele für URL-Signaturen finden Sie in den folgenden Themen:

- Erstellen einer URL-Signatur mit Perl
- Erstellen einer URL-Signatur mithilfe von C# und dem .NET Framework
- Erstellen einer URL-Signatur mit Java

Anstatt signierte Cookies zu verwenden, URLs um die Signatur zu erstellen, können Sie signierte Cookies verwenden. Weitere Informationen finden Sie unter Erstellen Sie signierte Cookies mit PHP.

Erstellen einer URL-Signatur mithilfe von C# und dem .NET Framework

Die C#-Beispiele in diesem Abschnitt implementieren eine Beispielanwendung, die zeigt, wie Signaturen für CloudFront private Distributionen mithilfe von vorgefertigten und benutzerdefinierten Richtlinienanweisungen erstellt werden. Die Beispiele enthalten Dienstprogrammfunktionen auf der Grundlage des AWS SDK für .NET, das in .NET-Anwendungen nützlich sein kann.

Sie können signierte URLs und signierte Cookies auch mithilfe von erstellen. SDK für .NET Schlagen Sie unter den folgenden Themen in der API-Referenz für SDK für .NET nach:

- Signiert URLs — AmazonCloudFrontUrlSigner
- Signierte Cookies — AmazonCloudFrontCookieSigner

Um den Code herunterzuladen, navigieren Sie zu Signaturcode in C #.

> **Hinweise**
>
> - Die `AmazonCloudFrontCookieSigner` Klassen `AmazonCloudFrontUrlSigner` und die Kurse wurden in ein separates Paket umgezogen. Weitere Informationen zu ihrer Verwendung finden Sie unter CookieSigner und UrlSigner im AWS SDK für .NET (V4) Developer Guide.
> - Das Erstellen einer URL-Signatur ist nur ein Teil der Bereitstellung privater Inhalte über eine signierte URL. Weitere Informationen finden Sie unter Verwenden Sie signierte URLs. Weitere Informationen zur Verwendung signierter Cookies finden Sie unterVerwenden Sie signierte Cookies.

Verwenden Sie einen RSA-Schlüssel im.NET Framework

Um einen RSA-Schlüssel im.NET Framework zu verwenden, müssen Sie die AWS bereitgestellte PEM-Datei in das XML-Format konvertieren, das das.NET Framework verwendet.

Nach der Konvertierung weist die Datei mit dem privaten RSA-Schlüssel das folgende Format auf:

Example : Privater RSA-Schlüssel im XML .NET Framework-Format

```
<RSAKeyValue>
  <Modulus>
    wO5IvYCP5UcoCKDo1dcspoMehWBZcyfs9QEzGi6Oe5y+ewGr1oW+vB2GPB
    ANBiVPcUHTFWhwaIBd3oglmF0lGQljP/jOfmXHUK2kUUnLnJp+oOBL2NiuFtqcW6h/L5lIpD8Yq+NRHg
    Ty4zDsyr2880MvXv88yEFURCkqEXAMPLE=
  </Modulus>
  <Exponent>AQAB</Exponent>
  <P>
    5bmKDaTz
    npENGVqz4Cea8XPH+sxt+2VaAwYnsarVUoSBeVt8WLloVuZGG9IZYmH5KteXEu7fZveYd9UEXAMPLE==
  </P>
  <Q>
    1v9l/WN1a1N3rOK4VGoCokx7kR2SyTMSbZgF9IWJNOugR/WZw7HTnjipO3c9dy1Ms9pUKwUF4
```

```
    6d7049EXAMPLE==
  </Q>
  <DP>
    RgrSKuLWXMyBH+/l1Dx/I4tXuAJIrlPyo+VmiOc7b5NzHptkSHEPfR9s1
    OK0VqjknclqCJ3Ig86OMEtEXAMPLE==
  </DP>
  <DQ>
    pjPjvSFw+RoaTu0pgCA/jwW/FGyfN6iim1RFbkT4
    z49DZb2IM885f3vf35eLTaEYRYUHQgZtChNEV0TEXAMPLE==
  </DQ>
  <InverseQ>
    nkvOJTg5QtGNgWb9i
    cVtzrL/1pFEOHbJXwEJdU99N+7sMK+1066DL/HSBUCD63qD4USpnf0myc24in0EXAMPLE==</InverseQ>
  <D>
      Bc7mp7XYHynuPZxChjWNJZIq+A73gm0ASDv6At7F8Vi9r0xUlQe/v0AQS3ycN8QlyR4XMbzMLYk
      3yjxFDXo4ZKQtOGzLGteCU2srANiLv26/imXA8FVidZftTAtLviWQZBVPTeYIA69ATUYPEq0a5u5wjGy
      UOij9OWyuEXAMPLE=
   </D>
</RSAKeyValue>
```

Signaturmethode für vordefinierte Richtlinien in C#

Der folgende C#-Code erstellt eine signierte URL, die eine vordefinierte Richtlinie verwendet, indem die folgenden Schritte ausgeführt werden:

- Es wird eine Richtlinienanweisung erstellt.
- Hasht die Richtlinienanweisung mithilfe von RSA und des privaten Schlüssels SHA1, dessen entsprechender öffentlicher Schlüssel sich in einer vertrauenswürdigen Schlüsselgruppe befindet, und signiert das Ergebnis mit Hilfe von RSA.
- Die gehashte und signierte Richtlinienanweisung wird in Base64 codiert. Dabei werden Sonderzeichen ersetzt, damit die Zeichenfolge sicher als URL-Anfrageparameter verwendet werden kann.
- Verkettet die Werte.

Ein Beispiel für die vollständige Implementierung finden Sie unter Signaturcode in C#.

> **Note**
>
> Das `keyId` wird zurückgegeben, wenn Sie einen öffentlichen Schlüssel in hochladen. CloudFront Weitere Informationen finden Sie unter

6

&Key-Pair-Id.

Example : In C# festgelegte Methode zum Signieren von Richtlinien

```
public static string ToUrlSafeBase64String(byte[] bytes)
{
    return System.Convert.ToBase64String(bytes)
        .Replace('+', '-')
        .Replace('=', '_')
        .Replace('/', '~');
}

public static string CreateCannedPrivateURL(string urlString,
    string durationUnits, string durationNumber, string pathToPolicyStmnt,
    string pathToPrivateKey, string keyId)
{
    // args[] 0-thisMethod, 1-resourceUrl, 2-seconds-minutes-hours-days
    // to expiration, 3-numberOfPreviousUnits, 4-pathToPolicyStmnt,
    // 5-pathToPrivateKey, 6-keyId

    TimeSpan timeSpanInterval = GetDuration(durationUnits, durationNumber);

    // Create the policy statement.
    string strPolicy = CreatePolicyStatement(pathToPolicyStmnt,
        urlString,
        DateTime.Now,
        DateTime.Now.Add(timeSpanInterval),
        "0.0.0.0/0");
    if ("Error!" == strPolicy) return "Invalid time frame." +
        "Start time cannot be greater than end time.";

    // Copy the expiration time defined by policy statement.
    string strExpiration = CopyExpirationTimeFromPolicy(strPolicy);

    // Read the policy into a byte buffer.
    byte[] bufferPolicy = Encoding.ASCII.GetBytes(strPolicy);

    // Initialize the SHA1CryptoServiceProvider object and hash the policy data.
    using (SHA1CryptoServiceProvider
        cryptoSHA1 = new SHA1CryptoServiceProvider())
    {
```

```
        bufferPolicy = cryptoSHA1.ComputeHash(bufferPolicy);

        // Initialize the RSACryptoServiceProvider object.
        RSACryptoServiceProvider providerRSA = new RSACryptoServiceProvider();
        XmlDocument xmlPrivateKey = new XmlDocument();

        // Load your private key, which you created by converting your
        // .pem file to the XML format that the .NET framework uses.
        // Several tools are available.
        xmlPrivateKey.Load(pathToPrivateKey);

        // Format the RSACryptoServiceProvider providerRSA and
        // create the signature.
        providerRSA.FromXmlString(xmlPrivateKey.InnerXml);
        RSAPKCS1SignatureFormatter rsaFormatter =
            new RSAPKCS1SignatureFormatter(providerRSA);
        rsaFormatter.SetHashAlgorithm("SHA1");
        byte[] signedPolicyHash = rsaFormatter.CreateSignature(bufferPolicy);

        // Convert the signed policy to URL-safe base64 encoding and
        // replace unsafe characters + = / with the safe characters - _ ~
        string strSignedPolicy = ToUrlSafeBase64String(signedPolicyHash);

        // Concatenate the URL, the timestamp, the signature,
        // and the key pair ID to form the signed URL.
        return urlString +
            "?Expires=" +
            strExpiration +
            "&Signature=" +
            strSignedPolicy +
            "&Key-Pair-Id=" +
            keyId;
    }
}
```

Signaturmethode für benutzerdefinierte Richtlinien in C#

Der folgende C#-Code erstellt eine signierte URL, die eine benutzerdefinierte Richtlinie verwendet, indem die folgenden Schritte ausgeführt werden:

1. Es wird eine Richtlinienanweisung erstellt.
2. Die Richtlinienanweisung wird in Base64 codiert. Dabei werden Sonderzeichen ersetzt, damit die Zeichenfolge sicher als URL-Anfrageparameter verwendet werden kann.

3. Hasht die Richtlinienanweisung mithilfe von RSA und des privaten Schlüssels SHA1, dessen entsprechender öffentlicher Schlüssel sich in einer vertrauenswürdigen Schlüsselgruppe befindet, und verschlüsselt das Ergebnis mithilfe von RSA.
4. Die gehashte Richtlinienanweisung wird in Base64 codiert. Dabei werden Sonderzeichen ersetzt, damit die Zeichenfolge sicher als URL-Anfrageparameter verwendet werden kann.
5. Verkettet die Werte.

Ein Beispiel für die vollständige Implementierung finden Sie unter Signaturcode in C#.

> **Note**
>
> Das `keyId` wird zurückgegeben, wenn Sie einen öffentlichen Schlüssel in hochladen. CloudFront Weitere Informationen finden Sie unter
>
> 6
>
> &Key-Pair-Id.

Example : Methode zum Signieren benutzerdefinierter Richtlinien in C#

```
public static string ToUrlSafeBase64String(byte[] bytes)
{
    return System.Convert.ToBase64String(bytes)
        .Replace('+', '-')
        .Replace('=', '_')
        .Replace('/', '~');
}

public static string CreateCustomPrivateURL(string urlString,
    string durationUnits, string durationNumber, string startIntervalFromNow,
    string ipaddress, string pathToPolicyStmnt, string pathToPrivateKey,
    string keyId)
{
    // args[] 0-thisMethod, 1-resourceUrl, 2-seconds-minutes-hours-days
    // to expiration, 3-numberOfPreviousUnits, 4-starttimeFromNow,
    // 5-ip_address, 6-pathToPolicyStmt, 7-pathToPrivateKey, 8-keyId

    TimeSpan timeSpanInterval = GetDuration(durationUnits, durationNumber);
    TimeSpan timeSpanToStart = GetDurationByUnits(durationUnits,
        startIntervalFromNow);
    if (null == timeSpanToStart)
```

```
        return "Invalid duration units." +
            "Valid options: seconds, minutes, hours, or days";

    string strPolicy = CreatePolicyStatement(
        pathToPolicyStmnt, urlString, DateTime.Now.Add(timeSpanToStart),
        DateTime.Now.Add(timeSpanInterval), ipaddress);

    // Read the policy into a byte buffer.
    byte[] bufferPolicy = Encoding.ASCII.GetBytes(strPolicy);

    // Convert the policy statement to URL-safe base64 encoding and
    // replace unsafe characters + = / with the safe characters - _ ~

    string urlSafePolicy = ToUrlSafeBase64String(bufferPolicy);

    // Initialize the SHA1CryptoServiceProvider object and hash the policy data.
    byte[] bufferPolicyHash;
    using (SHA1CryptoServiceProvider cryptoSHA1 =
        new SHA1CryptoServiceProvider())
    {
        bufferPolicyHash = cryptoSHA1.ComputeHash(bufferPolicy);

        // Initialize the RSACryptoServiceProvider object.
        RSACryptoServiceProvider providerRSA = new RSACryptoServiceProvider();
        XmlDocument xmlPrivateKey = new XmlDocument();

        // Load your private key, which you created by converting your
        // .pem file to the XML format that the .NET framework uses.
        // Several tools are available.
        xmlPrivateKey.Load(pathToPrivateKey);

        // Format the RSACryptoServiceProvider providerRSA
        // and create the signature.
        providerRSA.FromXmlString(xmlPrivateKey.InnerXml);
        RSAPKCS1SignatureFormatter RSAFormatter =
            new RSAPKCS1SignatureFormatter(providerRSA);
        RSAFormatter.SetHashAlgorithm("SHA1");
        byte[] signedHash = RSAFormatter.CreateSignature(bufferPolicyHash);

        // Convert the signed policy to URL-safe base64 encoding and
        // replace unsafe characters + = / with the safe characters - _ ~
        string strSignedPolicy = ToUrlSafeBase64String(signedHash);

        return urlString +
```

```
            "?Policy=" +
            urlSafePolicy +
            "&Signature=" +
            strSignedPolicy +
            "&Key-Pair-Id=" +
            keyId;
    }
}
```

Dienstprogrammmethoden für die Generierung von Signaturen

Mit den folgenden Methoden lassen sich die Richtlinienanweisung aus einer Datei abrufen und Zeitintervalle für die Generierung von Signaturen parsen.

Example : Hilfsmethoden für die Signaturgenerierung

```
public static string CreatePolicyStatement(string policyStmnt,
   string resourceUrl,
   DateTime startTime,
   DateTime endTime,
   string ipAddress)

{
   // Create the policy statement.
   FileStream streamPolicy = new FileStream(policyStmnt, FileMode.Open,
 FileAccess.Read);
   using (StreamReader reader = new StreamReader(streamPolicy))
   {
      string strPolicy = reader.ReadToEnd();

      TimeSpan startTimeSpanFromNow = (startTime - DateTime.Now);
      TimeSpan endTimeSpanFromNow = (endTime - DateTime.Now);
      TimeSpan intervalStart =
         (DateTime.UtcNow.Add(startTimeSpanFromNow)) -
         new DateTime(1970, 1, 1, 0, 0, 0, DateTimeKind.Utc);
      TimeSpan intervalEnd =
         (DateTime.UtcNow.Add(endTimeSpanFromNow)) -
         new DateTime(1970, 1, 1, 0, 0, 0, DateTimeKind.Utc);

      int startTimestamp = (int)intervalStart.TotalSeconds; // START_TIME
      int endTimestamp = (int)intervalEnd.TotalSeconds;  // END_TIME

      if (startTimestamp > endTimestamp)
         return "Error!";
```

```
      // Replace variables in the policy statement.
      strPolicy = strPolicy.Replace("RESOURCE", resourceUrl);
      strPolicy = strPolicy.Replace("START_TIME", startTimestamp.ToString());
      strPolicy = strPolicy.Replace("END_TIME", endTimestamp.ToString());
      strPolicy = strPolicy.Replace("IP_ADDRESS", ipAddress);
      strPolicy = strPolicy.Replace("EXPIRES", endTimestamp.ToString());
      return strPolicy;
   }
}

public static TimeSpan GetDuration(string units, string numUnits)
{
   TimeSpan timeSpanInterval = new TimeSpan();
   switch (units)
   {
      case "seconds":
         timeSpanInterval = new TimeSpan(0, 0, 0, int.Parse(numUnits));
         break;
      case "minutes":
         timeSpanInterval = new TimeSpan(0, 0, int.Parse(numUnits), 0);
         break;
      case "hours":
         timeSpanInterval = new TimeSpan(0, int.Parse(numUnits), 0 ,0);
         break;
      case "days":
         timeSpanInterval = new TimeSpan(int.Parse(numUnits),0 ,0 ,0);
         break;
      default:
         Console.WriteLine("Invalid time units;" +
            "use seconds, minutes, hours, or days");
         break;
   }
   return timeSpanInterval;
}

private static TimeSpan GetDurationByUnits(string durationUnits,
   string startIntervalFromNow)
{
   switch (durationUnits)
   {
      case "seconds":
         return new TimeSpan(0, 0, int.Parse(startIntervalFromNow));
      case "minutes":
```

```
            return new TimeSpan(0, int.Parse(startIntervalFromNow), 0);
         case "hours":
            return new TimeSpan(int.Parse(startIntervalFromNow), 0, 0);
         case "days":
            return new TimeSpan(int.Parse(startIntervalFromNow), 0, 0, 0);
         default:
            return new TimeSpan(0, 0, 0, 0);
      }
   }

   public static string CopyExpirationTimeFromPolicy(string policyStatement)
   {
      int startExpiration = policyStatement.IndexOf("EpochTime");
      string strExpirationRough = policyStatement.Substring(startExpiration +
         "EpochTime".Length);
      char[] digits = { '0', '1', '2', '3', '4', '5', '6', '7', '8', '9' };

      List<char> listDigits = new List<char>(digits);
      StringBuilder buildExpiration = new StringBuilder(20);

      foreach (char c in strExpirationRough)
      {
         if (listDigits.Contains(c))
            buildExpiration.Append(c);
      }
      return buildExpiration.ToString();
   }
```

Weitere Informationen finden Sie auch unter

- Erstellen einer URL-Signatur mit Perl
- Erstellen einer URL-Signatur mit PHP
- Erstellen einer URL-Signatur mit Java

Erstellen einer URL-Signatur mit Java

Zusätzlich zum folgenden Codebeispiel können Sie die `CloudFrontUrlSigner`Utility-Klasse in der AWS SDK für Java (Version 1) verwenden, um CloudFront signierte zu erstellenURLs.

Weitere Beispiele finden Sie unter Erstellen von signierten Cookies URLs und Cookies mithilfe eines AWS SDK in der Codebibliothek für AWS SDK-Codebeispiele.

> **Note**
>
> Das Erstellen einer signierten URL ist nur ein Teil des Prozesses der Bereitstellung privater Inhalte mit CloudFront. Weitere Informationen zum gesamten Prozess finden Sie unter Verwenden Sie signierte URLs.

Das folgende Beispiel zeigt, wie eine CloudFront signierte URL erstellt wird.

Example Java-Richtlinie und Verschlüsselungsmethoden für Signaturen

```
package org.example;

import java.time.Instant;
import java.time.temporal.ChronoUnit;
import software.amazon.awssdk.services.cloudfront.CloudFrontUtilities;
import software.amazon.awssdk.services.cloudfront.model.CannedSignerRequest;
import software.amazon.awssdk.services.cloudfront.url.SignedUrl;

public class Main {

    public static void main(String[] args) throws Exception {
        CloudFrontUtilities cloudFrontUtilities = CloudFrontUtilities.create();
        Instant expirationDate = Instant.now().plus(7, ChronoUnit.DAYS);
        String resourceUrl = "https://a1b2c3d4e5f6g7.cloudfront.net";
        String keyPairId = "K1UA3WV15I7JSD";
        CannedSignerRequest cannedRequest = CannedSignerRequest.builder()
                .resourceUrl(resourceUrl)
                .privateKey(new java.io.File("/path/to/private_key.pem").toPath())
                .keyPairId(keyPairId)
                .expirationDate(expirationDate)
                .build();
        SignedUrl signedUrl =
 cloudFrontUtilities.getSignedUrlWithCannedPolicy(cannedRequest);
        String url = signedUrl.url();
        System.out.println(url);

    }
}
```

Weitere Informationen finden Sie auch unter:

- Erstellen einer URL-Signatur mit Perl
- Erstellen einer URL-Signatur mit PHP
- Erstellen einer URL-Signatur mithilfe von C# und dem .NET Framework

Beschränken Sie den Zugriff auf einen AWS Ursprung

Sie können einige AWS Ursprünge so konfigurieren CloudFront , dass sie die folgenden Vorteile bieten:

- Schränkt den Zugriff auf den AWS Ursprung ein, sodass er nicht öffentlich zugänglich ist
- Stellt sicher, dass Zuschauer (Nutzer) nur über die angegebene CloudFront Distribution auf die Inhalte in der AWS Originalversion zugreifen können. Dadurch wird verhindert, dass sie direkt aus dem Bucket oder über eine unbeabsichtigte Verteilung auf die Inhalte zugreifen CloudFront

Stellen Sie dazu so ein, CloudFront dass authentifizierte Anfragen an Ihren AWS Ursprung gesendet werden, und konfigurieren Sie den AWS Ursprung so, dass nur authentifizierte Anfragen von abgerufen werden können. CloudFront Weitere Informationen zu kompatiblen Herkunftstypen finden Sie in den folgenden Themen. AWS

Themen

- Beschränken Sie den Zugriff auf einen AWS Elemental MediaPackage v2-Ursprung
- Beschränken Sie den Zugriff auf einen AWS Elemental MediaStore Ursprung
- Beschränken Sie den Zugriff auf den URL-Ursprung einer AWS Lambda Funktion
- Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung
- Beschränken Sie den Zugriff mit VPC-Ursprüngen

Beschränken Sie den Zugriff auf einen AWS Elemental MediaPackage v2-Ursprung

CloudFront bietet Origin Access Control (OAC) zur Beschränkung des Zugriffs auf einen MediaPackage v2-Ursprung.

> **Note**
>
> CloudFront OAC unterstützt nur v2. MediaPackage MediaPackage v1 wird nicht unterstützt.

Themen

Ein neues OAC erstellen

Führen Sie die in den folgenden Themen beschriebenen Schritte aus, um ein neues OAC in einzurichten. CloudFront

Themen

Voraussetzungen

Bevor Sie OAC erstellen und einrichten, müssen Sie über eine CloudFront Distribution mit MediaPackage v2-Ursprung verfügen. Weitere Informationen finden Sie unter Verwenden Sie einen MediaStore Container oder einen MediaPackage Channel.

Erteilen Sie CloudFront die Erlaubnis, auf den MediaPackage v2-Ursprung zuzugreifen

Bevor Sie ein OAC erstellen oder es in einer CloudFront Distribution einrichten, stellen Sie sicher, dass es über die Zugriffsrechte für den MediaPackage v2-Ursprung CloudFront verfügt. Tun Sie dies, nachdem Sie eine CloudFront Distribution erstellt haben, aber bevor Sie das OAC dem MediaPackage v2-Ursprung in der Distributionskonfiguration hinzufügen.

Verwenden Sie eine IAM-Richtlinie, um dem CloudFront Dienstprinzipal (`cloudfront.amazonaws.com`) den Zugriff auf den Ursprung zu ermöglichen. Das `Condition` Element in der Richtlinie ermöglicht den CloudFront Zugriff auf den MediaPackage v2-Ursprung nur, wenn die Anfrage im Namen der CloudFront Distribution erfolgt, die den MediaPackage v2-Ursprung

enthält. Dies ist die Distribution mit dem MediaPackage v2-Ursprung, zu der Sie OAC hinzufügen möchten.

Example : IAM-Richtlinie, die nur Lesezugriff für eine CloudFront Distribution mit aktiviertem OAC ermöglicht

Die folgende Richtlinie ermöglicht der CloudFront Distribution (*E1PDK09ESKHJWT*) den Zugriff auf den v2-Ursprung. MediaPackage Der Ursprung ist der für das `Resource` Element angegebene ARN.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCloudFrontServicePrincipal",
            "Effect": "Allow",
            "Principal": {"Service": "cloudfront.amazonaws.com"},
            "Action": "mediapackagev2:GetObject",
            "Resource": "arn:aws:mediapackagev2:us-
east-1:123456789012:channelGroup/channel-group-name/channel/channel-name/
originEndpoint/origin_endpoint_name",
            "Condition": {
                "StringEquals": {"AWS:SourceArn":
 "arn:aws:cloudfront::123456789012:distribution/E1PDK09ESKHJWT"}
            }
        }
    ]
}
```

Hinweise

- Wenn Sie die MQAR-Funktion und Origin Access Control (OAC) aktiviert haben, fügen Sie die `mediapackagev2:GetHeadObject` Aktion der IAM-Richtlinie hinzu. MQAR benötigt diese Berechtigung, um `HEAD` Anfragen an den v2-Ursprung zu senden. MediaPackage Weitere Informationen zu MQAR finden Sie unter. Resilienz im Hinblick auf Medienqualität
- Wenn Sie eine Distribution erstellen, die keine Berechtigungen für Ihren MediaPackage v2-Ursprung hat, können Sie in der CloudFront Konsole „Richtlinie kopieren“ und dann „Endpunktberechtigungen aktualisieren“ auswählen. Anschließend können Sie die kopierte

Berechtigung an den Endpunkt anhängen. Weitere Informationen finden Sie im AWS Elemental MediaPackage Benutzerhandbuch unter Felder für Endpunktrichtlinien.

Das OAC erstellen

Um ein OAC zu erstellen, können Sie die AWS Management Console, AWS CloudFormation AWS CLI, oder die CloudFront API verwenden.

Console

Um ein OAC zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Origin access (Ursprungszugriff) aus.
3. Wählen Sie Create control setting (Kontrolleinstellung erstellen) aus.
4. Gehen Sie im Formular Neues OAC erstellen wie folgt vor:
   a. Geben Sie einen Namen und (optional) eine Beschreibung für das OAC ein.
   b. Für das Signierverhalten empfehlen wir, die Standardeinstellung beizubehalten (Anfragen signieren (empfohlen)). Weitere Informationen finden Sie unter the section called “Erweiterte Einstellungen für die Ursprungszugriffssteuerung”.
5. Wählen Sie als Origin-Typ MediaPackage V2 aus.
6. Wählen Sie Erstellen aus.

> **Tip**
> Nachdem Sie das OAC erstellt haben, notieren Sie sich den Namen. Sie benötigen diesen im folgenden Verfahren.

Um ein OAC zu einem MediaPackage v2-Ursprung in einer Distribution hinzuzufügen

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie eine Distribution mit einem MediaPackage V2-Ursprung aus, zu der Sie das OAC hinzufügen möchten, und wählen Sie dann den Tab Origins.

3. Wählen Sie den MediaPackage v2-Ursprung aus, dem Sie das OAC hinzufügen möchten, und wählen Sie dann Bearbeiten.
4. Wählen Sie HTTPS only (Nur HTTPS) für Protocol (Protokoll) Ihres Ursprungs aus.
5. Wähle aus dem Drop-down-Menü für die Origin-Zugriffskontrolle den OAC-Namen aus, den du verwenden möchtest.
6. Wählen Sie Änderungen speichern aus.

Die Distribution beginnt mit der Bereitstellung an allen CloudFront Edge-Standorten. Wenn ein Edge-Standort die neue Konfiguration empfängt, signiert er alle Anfragen, die er an den MediaPackage v2-Ursprung sendet.

CloudFormation

Verwenden Sie den `AWS::CloudFront::OriginAccessControl` Ressourcentyp AWS CloudFormation, um ein OAC mit zu erstellen. Das folgende Beispiel zeigt die AWS CloudFormation Vorlagensyntax im YAML-Format für die Erstellung eines OAC.

```
Type: AWS::CloudFront::OriginAccessControl
Properties:
  OriginAccessControlConfig:
      Description: An optional description for the origin access control
      Name: ExampleOAC
      OriginAccessControlOriginType: mediapackagev2
      SigningBehavior: always
      SigningProtocol: sigv4
```

Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter AWS::CloudFront::OriginAccessSteuerung.

CLI

Verwenden Sie den aws cloudfront create-origin-access-control Befehl, um eine Origin-Zugriffskontrolle mit dem AWS Command Line Interface (AWS CLI) zu erstellen. Sie können eine Eingabedatei verwenden, um die Eingabeparameter für den Befehl bereitzustellen, anstatt jeden einzelnen Parameter als Befehlszeileneingabe anzugeben.

So erstellen Sie eine Ursprungszugriffssteuerung (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl zum Erstellen einer Datei mit dem Namen `origin-access-control.yaml`. Diese Datei enthält alle Eingabeparameter für den Befehl create-origin-access-control.

```
aws cloudfront create-origin-access-control --generate-cli-skeleton yaml-input >
 origin-access-control.yaml
```

2. Öffnen Sie die Datei `origin-access-control.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, um einen Namen für die OAC und eine Beschreibung (optional) hinzuzufügen, und ändern Sie `SigningBehavior` zu `always`. Speichern Sie dann die Datei.

   Weitere Informationen zu anderen OAC-Einstellungen finden Sie unter the section called “Erweiterte Einstellungen für die Ursprungszugriffssteuerung”.

3. Verwenden Sie den folgenden Befehl, um die Ursprungszugriffssteuerung mit Eingabeparametern aus der Datei `origin-access-control.yaml` zu erstellen.

```
aws cloudfront create-origin-access-control --cli-input-yaml file://origin-
access-control.yaml
```

   Notieren Sie den `Id`-Wert in der Befehlsausgabe. Sie benötigen ihn, um das OAC zu einem MediaPackage v2-Ursprung in einer CloudFront Distribution hinzuzufügen.

Um ein OAC an einen MediaPackage v2-Ursprung in einer vorhandenen Distribution anzuhängen (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um die Verteilungskonfiguration für die CloudFront Distribution zu speichern, zu der Sie das OAC hinzufügen möchten. Die Distribution muss einen MediaPackage v2-Ursprung haben.

```
aws cloudfront get-distribution-config --id <CloudFront distribution ID> --
output yaml > dist-config.yaml
```

2. Öffnen Sie die Datei mit dem Namen `dist-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und nehmen Sie die folgenden Änderungen vor:

   - Fügen Sie im Objekt `Origins` die ID der OAC dem Feld namens `OriginAccessControlId` hinzu.
   - Entfernen Sie den Wert aus dem Feld namens `OriginAccessIdentity`, sofern vorhanden.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

   Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Verteilung zu aktualisieren und die Ursprungszugriffssteuerung zu verwenden.

   ```
   aws cloudfront update-distribution --id <CloudFront distribution ID> --cli-
   input-yaml file://dist-config.yaml
   ```

Die Verteilung beginnt mit der Bereitstellung an allen CloudFront Edge-Standorten. Wenn ein Edge-Standort die neue Konfiguration empfängt, signiert er alle Anfragen, die er an den MediaPackage v2-Ursprung sendet.

API

Um ein OAC mit der CloudFront API zu erstellen, verwenden Sie CreateOriginAccessControl. Weitere Informationen zu den Feldern, die Sie in diesem API-Aufruf angeben, finden Sie in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Nachdem Sie ein OAC erstellt haben, können Sie es mithilfe eines der folgenden API-Aufrufe an einen MediaPackage v2-Ursprung in einer Distribution anhängen:

- Um es an eine bestehende Distribution anzuhängen, verwenden Sie UpdateDistribution.
- Um es an eine neue Distribution anzuhängen, verwenden Sie CreateDistribution.

Geben Sie für diese beiden API-Aufrufe die OAC-ID in das `OriginAccessControlId` Feld innerhalb eines Ursprungs ein. Weitere Informationen zu den anderen Feldern, die Sie in diesen

API-Aufrufen angeben, finden Sie in Referenz für alle Verteilungseinstellungen und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Erweiterte Einstellungen für die Ursprungszugriffssteuerung

Die CloudFront OAC-Funktion umfasst erweiterte Einstellungen, die nur für bestimmte Anwendungsfälle vorgesehen sind. Verwenden Sie die empfohlenen Einstellungen, sofern Sie die erweiterten Einstellungen nicht speziell benötigen.

OAC enthält eine Einstellung mit dem Namen Signaturverhalten (in der Konsole) oder `SigningBehavior` (in der API, CLI und AWS CloudFormation). Diese Einstellung bietet die folgenden Optionen:

Ursprungsanforderungen immer signieren (empfohlene Einstellung)

Wir empfehlen die Verwendung dieser Einstellung mit der Bezeichnung Sign requests (recommended) (Anforderungen signieren (empfohlen)) in der Konsole bzw. `always` in der API, CLI und AWS CloudFormation. Mit dieser Einstellung werden CloudFront immer alle Anfragen signiert, die an den MediaPackage v2-Ursprung gesendet werden.

Ursprungsanforderungen nie signieren

Diese Einstellung heißt Do not sign requests (Anforderungen nicht signieren) in der Konsole bzw. `never` in der API, CLI und AWS CloudFormation. Verwenden Sie diese Einstellung, um OAC für alle Ursprünge in allen Distributionen zu deaktivieren, die dieses OAC verwenden. Dies kann Zeit und Mühe sparen, verglichen mit dem Entfernen eines OAC nacheinander aus allen Origins und Distributionen, die es verwenden. Mit dieser Einstellung signiert CloudFront es keine Anfragen, die es an den MediaPackage v2-Ursprung sendet.

> **Warning**
>
> Um diese Einstellung verwenden zu können, muss der MediaPackage v2-Ursprung öffentlich zugänglich sein. Wenn Sie diese Einstellung mit einem MediaPackage v2-Ursprung verwenden, der nicht öffentlich zugänglich ist, CloudFront können Sie nicht auf den Ursprung zugreifen. Der MediaPackage v2-Ursprung gibt Fehler zurück CloudFront und CloudFront leitet diese Fehler an die Zuschauer weiter. Weitere Informationen finden Sie im AWS Elemental MediaPackage Benutzerhandbuch MediaPackage im Beispiel für eine MediaPackage v2-Richtlinie für Richtlinien und Berechtigungen.

Viewer (Client)-**`Authorization`**-Header nicht überschreiben

Diese Einstellung heißt Do not override authorization header (Autorisierungsheader nicht überschreiben) in der Konsole bzw. `no-override` in der API, CLI und AWS CloudFormation. Verwenden Sie diese Einstellung, wenn Sie Originalanfragen nur signieren CloudFront möchten, wenn die entsprechende Viewer-Anfrage keinen `Authorization` Header enthält. Mit dieser Einstellung wird der `Authorization` Header der Viewer-Anfrage weitergegeben, CloudFront wenn eine vorhanden ist, signiert aber die ursprüngliche Anfrage (fügt einen eigenen `Authorization` Header hinzu), wenn die Viewer-Anfrage keinen `Authorization` Header enthält.

> **Warning**
>
> Um den `Authorization` Header aus der Viewer-Anfrage weiterzugeben, müssen Sie den `Authorization` Header zu einer Cache-Richtlinie für alle Cache-Verhaltensweisen hinzufügen, die MediaPackage v2-Ursprünge verwenden, die mit dieser ursprünglichen Zugriffskontrolle verknüpft sind.

Beschränken Sie den Zugriff auf einen AWS Elemental MediaStore Ursprung

CloudFront bietet Origin Access Control (OAC), um den Zugriff auf einen Ursprung einzuschränken. AWS Elemental MediaStore

Themen

- Erstellen Sie eine neue Origin-Zugriffskontrolle
- Erweiterte Einstellungen für die Ursprungszugriffssteuerung

Erstellen Sie eine neue Origin-Zugriffskontrolle

Führen Sie die in den folgenden Themen beschriebenen Schritte aus, um eine neue Origin-Zugriffskontrolle in einzurichten CloudFront.

Themen

- Voraussetzungen
- Erteilen Sie die CloudFront Erlaubnis, auf den MediaStore Ursprung zuzugreifen

- Erstellen Sie die Origin-Zugriffskontrolle

Voraussetzungen

Bevor Sie die Origin-Zugriffskontrolle erstellen und einrichten, müssen Sie über eine CloudFront Distribution mit einem MediaStore Ursprung verfügen.

Erteilen Sie die CloudFront Erlaubnis, auf den MediaStore Ursprung zuzugreifen

Bevor du eine Zugriffskontrolle für den Ursprung erstellst oder sie in einer CloudFront Distribution einrichtest, vergewissere dich, dass diese CloudFront Person über Zugriffsberechtigungen für den MediaStore Ursprung verfügt. Tun Sie dies, nachdem Sie eine CloudFront Distribution erstellt haben, aber bevor Sie das OAC zum MediaStore Ursprung in der Distributionskonfiguration hinzufügen.

Verwenden Sie eine MediaStore Container-Richtlinie, um dem CloudFront Dienstprinzipal (`cloudfront.amazonaws.com`) den Zugriff auf den Ursprung zu ermöglichen. Verwenden Sie ein `Condition` Element in der Richtlinie, um nur dann Zugriff auf den MediaStore Container CloudFront zu gewähren, wenn die Anfrage im Namen der CloudFront Distribution erfolgt, die den MediaStore Ursprung enthält. Dies ist die Distribution mit dem MediaStore Ursprung, zu der Sie OAC hinzufügen möchten.

Im Folgenden finden Sie Beispiele für MediaStore Container-Richtlinien, die es einer CloudFront Distribution ermöglichen, auf einen MediaStore Ursprung zuzugreifen.

Example MediaStore Container-Richtlinie, die schreibgeschützten Zugriff für eine CloudFront Distribution mit aktiviertem OAC ermöglicht

JSON

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "AllowCloudFrontServicePrincipalReadOnly",
                "Effect": "Allow",
                "Principal": {
                  "Service": "cloudfront.amazonaws.com"
                },
                "Action": [
                  "mediastore:GetObject"
```

```
                ],
                "Resource":
 "arn:aws:mediastore:<region>:111122223333:container/<container name>/*",
                "Condition": {
                    "StringEquals": {
                      "AWS:SourceArn":
 "arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
                    },
                    "Bool": {
                      "aws:SecureTransport": "true"
                    }
                }
            }
        ]
}
```

Example MediaStore Container-Richtlinie, die Lese- und Schreibzugriff für eine CloudFront Distribution mit aktiviertem OAC ermöglicht

JSON

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "AllowCloudFrontServicePrincipalReadWrite",
                "Effect": "Allow",
                "Principal": {
                  "Service": "cloudfront.amazonaws.com"
                },
                "Action": [
                  "mediastore:GetObject",
                  "mediastore:PutObject"
                ],
                "Resource":
 "arn:aws:mediastore:<region>:111122223333:container/<container name>/*",
                "Condition": {
                    "StringEquals": {
                      "AWS:SourceArn":
 "arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
                    },
```

```
                    "Bool": {
                      "aws:SecureTransport": "true"
                    }
                }
            }
        ]
}
```

> **Note**
>
> Um Schreibzugriff zu ermöglichen, müssen Sie die zulässigen HTTP-Methoden so konfigurieren, dass sie PUT in die Verhaltenseinstellungen Ihrer CloudFront Distribution aufgenommen werden.

Erstellen Sie die Origin-Zugriffskontrolle

Um ein OAC zu erstellen, können Sie die AWS Management Console, AWS CloudFormation AWS CLI, oder die CloudFront API verwenden.

Console

So erstellen Sie eine Ursprungszugriffssteuerung

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Origin access (Ursprungszugriff) aus.
3. Wählen Sie Create control setting (Kontrolleinstellung erstellen) aus.
4. Gehen Sie im Formular Create control setting (Kontrolleinstellung erstellen) wie folgt vor:
   a. Geben Sie im Bereich Details einen Namen (Name) und (optional) eine Beschreibung (Description) für die Ursprungszugriffssteuerung ein.
   b. Es empfiehlt sich, im Bereich Settings (Einstellungen) die Standardeinstellung (Sign requests (recommended)) (Anforderungen signieren (empfohlen)) zu belassen. Weitere Informationen finden Sie unter the section called “Erweiterte Einstellungen für die Ursprungszugriffssteuerung”.
5. Wählen Sie MediaStore aus der Drop-down-Liste „Origin-Typ“.
6. Wählen Sie Erstellen aus.

Nachdem die OAC erstellt wurde, notieren Sie sich den Namen. Sie benötigen diesen im folgenden Verfahren.

Um einem Ursprung in einer Distribution eine MediaStore Ursprungszugriffskontrolle hinzuzufügen

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie eine Distribution mit einem MediaStore Ursprung aus, zu dem Sie das OAC hinzufügen möchten, und wählen Sie dann den Tab Origins.
3. Wählen Sie den MediaStore Ursprung aus, dem Sie das OAC hinzufügen möchten, und klicken Sie dann auf Bearbeiten.
4. Wählen Sie HTTPS only (Nur HTTPS) für Protocol (Protokoll) Ihres Ursprungs aus.
5. Wählen Sie im Dropdown-Menü Origin access control (Ursprungszugriffssteuerung) die OAC aus, die Sie verwenden möchten.
6. Wählen Sie Änderungen speichern aus.

Die Verteilung beginnt mit der Bereitstellung an allen CloudFront Edge-Standorten. Wenn ein Edge-Standort die neue Konfiguration empfängt, signiert er alle Anfragen, die er an den MediaStore Bucket-Ursprung sendet.

CloudFormation

Verwenden Sie den `AWS::CloudFront::OriginAccessControl` Ressourcentyp AWS CloudFormation, um eine Origin-Zugriffskontrolle (OAC) zu erstellen. Das folgende Beispiel zeigt die AWS CloudFormation Vorlagensyntax im YAML-Format für die Erstellung einer Origin-Zugriffskontrolle.

```
Type: AWS::CloudFront::OriginAccessControl
Properties:
  OriginAccessControlConfig:
      Description: An optional description for the origin access control
      Name: ExampleOAC
      OriginAccessControlOriginType: mediastore
      SigningBehavior: always
      SigningProtocol: sigv4
```

Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter AWS::CloudFront::OriginAccessSteuerung.

CLI

Verwenden Sie den aws cloudfront create-origin-access-control Befehl, um eine Origin-Zugriffskontrolle mit dem AWS Command Line Interface (AWS CLI) zu erstellen. Sie können eine Eingabedatei verwenden, um die Eingabeparameter für den Befehl bereitzustellen, anstatt jeden einzelnen Parameter als Befehlszeileneingabe anzugeben.

So erstellen Sie eine Ursprungszugriffssteuerung (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl zum Erstellen einer Datei mit dem Namen `origin-access-control.yaml`. Diese Datei enthält alle Eingabeparameter für den Befehl create-origin-access-control.

```
aws cloudfront create-origin-access-control --generate-cli-skeleton yaml-input >
 origin-access-control.yaml
```

2. Öffnen Sie die Datei `origin-access-control.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, um einen Namen für die OAC und eine Beschreibung (optional) hinzuzufügen, und ändern Sie `SigningBehavior` zu `always`. Speichern Sie dann die Datei.

   Weitere Informationen zu anderen OAC-Einstellungen finden Sie unter the section called “Erweiterte Einstellungen für die Ursprungszugriffssteuerung”.

3. Verwenden Sie den folgenden Befehl, um die Ursprungszugriffssteuerung mit Eingabeparametern aus der Datei `origin-access-control.yaml` zu erstellen.

```
aws cloudfront create-origin-access-control --cli-input-yaml file://origin-
access-control.yaml
```

   Notieren Sie den `Id`-Wert in der Befehlsausgabe. Sie benötigen ihn, um das OAC zu einem MediaStore Ursprung in einer CloudFront Distribution hinzuzufügen.

Um ein OAC an einen MediaStore Ursprung in einer vorhandenen Distribution anzuhängen (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um die Verteilungskonfiguration für die CloudFront Distribution zu speichern, zu der Sie das OAC hinzufügen möchten. Die Verteilung muss einen MediaStore Ursprung haben.

```
aws cloudfront get-distribution-config --id <CloudFront distribution ID> --
output yaml > dist-config.yaml
```

2. Öffnen Sie die Datei mit dem Namen `dist-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und nehmen Sie die folgenden Änderungen vor:

   - Fügen Sie im Objekt `Origins` die ID der OAC dem Feld namens `OriginAccessControlId` hinzu.
   - Entfernen Sie den Wert aus dem Feld namens `OriginAccessIdentity`, sofern vorhanden.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

   Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Verteilung zu aktualisieren und die Ursprungszugriffssteuerung zu verwenden.

```
aws cloudfront update-distribution --id <CloudFront distribution ID> --cli-
input-yaml file://dist-config.yaml
```

Die Verteilung beginnt mit der Bereitstellung an allen CloudFront Edge-Standorten. Wenn ein Edge-Standort die neue Konfiguration empfängt, signiert er alle Anfragen, die er an den MediaStore Ursprung sendet.

API

Um mit der CloudFront API eine Origin-Zugriffskontrolle zu erstellen, verwenden Sie CreateOriginAccessControl. Weitere Informationen zu den Feldern, die Sie in diesem API-Aufruf

angeben, finden Sie in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Nachdem Sie eine Origin-Zugriffskontrolle erstellt haben, können Sie sie mithilfe eines der folgenden API-Aufrufe an einen MediaStore Ursprung in einer Distribution anhängen:

- Um sie an eine bestehende Distribution anzuhängen, verwenden Sie UpdateDistribution.
- Um es an eine neue Distribution anzuhängen, verwenden Sie CreateDistribution.

Geben Sie für beide API-Aufrufe die ID der Ursprungszugriffssteuerung im Feld `OriginAccessControlId` innerhalb eines Ursprungs an. Weitere Informationen zu den anderen Feldern, die Sie in diesen API-Aufrufen angeben, finden Sie unter Referenz für alle Verteilungseinstellungen und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Erweiterte Einstellungen für die Ursprungszugriffssteuerung

Die CloudFront Origin-Zugriffskontrollfunktion umfasst erweiterte Einstellungen, die nur für bestimmte Anwendungsfälle vorgesehen sind. Verwenden Sie die empfohlenen Einstellungen, sofern Sie die erweiterten Einstellungen nicht speziell benötigen.

Die Origin-Zugriffskontrolle enthält eine Einstellung namens Signaturverhalten (in der Konsole) oder `SigningBehavior` (in der API, CLI und AWS CloudFormation). Diese Einstellung bietet die folgenden Optionen:

Ursprungsanforderungen immer signieren (empfohlene Einstellung)

Wir empfehlen die Verwendung dieser Einstellung mit der Bezeichnung Sign requests (recommended) (Anforderungen signieren (empfohlen)) in der Konsole bzw. `always` in der API, CLI und AWS CloudFormation. Mit dieser Einstellung signiert es CloudFront immer alle Anfragen, die es an den MediaStore Ursprung sendet.

Ursprungsanforderungen nie signieren

Diese Einstellung heißt Do not sign requests (Anforderungen nicht signieren) in der Konsole bzw. `never` in der API, CLI und AWS CloudFormation. Verwenden Sie diese Einstellung, um die Ursprungszugriffssteuerung für alle Ursprünge in allen Verteilungen zu deaktivieren, die diese Ursprungszugriffssteuerung verwenden. Auf diese Weise lässt sich im Vergleich zum einzelnen Entfernen einer Ursprungszugriffssteuerung aus allen Ursprüngen und Verteilungen, die

diese verwenden, Zeit und Aufwand sparen. Mit dieser Einstellung signiert es CloudFront keine Anfragen, die es an den MediaStore Ursprung sendet.

> **Warning**
>
> Um diese Einstellung verwenden zu können, muss der MediaStore Ursprung öffentlich zugänglich sein. Wenn Sie diese Einstellung mit einem MediaStore Ursprung verwenden, der nicht öffentlich zugänglich ist, CloudFront können Sie nicht auf den Ursprung zugreifen. Der MediaStore Ursprung gibt Fehler zurück CloudFront und CloudFront leitet diese Fehler an die Zuschauer weiter. Weitere Informationen finden Sie in der MediaStore Beispiel-Container-Richtlinie für öffentlichen Lesezugriff über HTTPS.

Viewer (Client)-**`Authorization`**-Header nicht überschreiben

Diese Einstellung heißt Do not override authorization header (Autorisierungsheader nicht überschreiben) in der Konsole bzw. `no-override` in der API, CLI und AWS CloudFormation. Verwenden Sie diese Einstellung, wenn Sie ursprüngliche Anfragen nur signieren CloudFront möchten, wenn die entsprechende Viewer-Anfrage keinen `Authorization` Header enthält. Mit dieser Einstellung wird der `Authorization` Header der Viewer-Anfrage weitergegeben, CloudFront wenn eine vorhanden ist, signiert aber die ursprüngliche Anfrage (fügt einen eigenen `Authorization` Header hinzu), wenn die Viewer-Anfrage keinen `Authorization` Header enthält.

> **Warning**
>
> Um den `Authorization` Header aus der Viewer-Anfrage weiterzugeben, müssen Sie den `Authorization` Header zu einer Cache-Richtlinie für alle Cache-Verhaltensweisen hinzufügen, die MediaStore Ursprünge verwenden, die mit dieser ursprünglichen Zugriffskontrolle verknüpft sind.

Beschränken Sie den Zugriff auf den URL-Ursprung einer AWS Lambda Funktion

CloudFront bietet Origin Access Control (OAC), um den Zugriff auf den URL-Ursprung einer Lambda-Funktion einzuschränken.

Themen

Erstellen Sie ein neues OAC

Führen Sie die in den folgenden Themen beschriebenen Schritte aus, um ein neues OAC in einzurichten. CloudFront

> **Important**
>
> Wenn Sie mit Ihrer Lambda-Funktions-URL `POST` Methoden verwenden`PUT`, müssen Ihre Benutzer den SHA256 des Hauptteils berechnen und den Payload-Hashwert des Anforderungstexts in den `x-amz-content-sha256` Header aufnehmen, wenn sie die Anfrage an senden. CloudFront Lambda unterstützt keine unsignierten Payloads.

Themen

Voraussetzungen

Bevor Sie OAC erstellen und einrichten, benötigen Sie eine CloudFront Distribution mit einer Lambda-Funktions-URL als Ursprung. Um OAC zu verwenden, müssen Sie den Wert für den Parameter angeben`AWS_IAM`. `AuthType` Weitere Informationen finden Sie unter Verwenden Sie eine Lambda-Funktions-URL.

Erteilen Sie die CloudFront Erlaubnis, auf die URL der Lambda-Funktion zuzugreifen

Bevor Sie ein OAC erstellen oder es in einer CloudFront Distribution einrichten, stellen Sie sicher, dass es über die Zugriffsberechtigung für die Lambda-Funktions-URL CloudFront verfügt. Tun Sie dies, nachdem Sie eine CloudFront Distribution erstellt haben, aber bevor Sie das OAC zur Lambda-Funktions-URL in der Verteilungskonfiguration hinzufügen.

> **Note**
>
> Um die IAM-Richtlinie für die Lambda-Funktions-URL zu aktualisieren, müssen Sie die AWS Command Line Interface ()AWS CLI verwenden. Die Bearbeitung der IAM-Richtlinie in der Lambda-Konsole wird derzeit nicht unterstützt.

Der folgende AWS CLI Befehl gewährt dem CloudFront Service Principal (`cloudfront.amazonaws.com`) Zugriff auf Ihre Lambda-Funktions-URL. Das `Condition` Element in der Richtlinie ermöglicht den CloudFront Zugriff auf Lambda nur, wenn die Anfrage im Namen der CloudFront Distribution erfolgt, die die URL der Lambda-Funktion enthält. Dies ist die Distribution mit dem URL-Ursprung der Lambda-Funktion, zu der Sie OAC hinzufügen möchten.

Example : AWS CLI Befehl zum Aktualisieren einer Richtlinie, um für eine Distribution mit aktiviertem OAC nur Lesezugriff zu CloudFront gewähren

Der folgende AWS CLI Befehl ermöglicht der CloudFront Distribution (*E1PDK09ESKHJWT*) den Zugriff auf Ihr Lambda*FUNCTION_URL_NAME*.

```
aws lambda add-permission \
--statement-id "AllowCloudFrontServicePrincipal" \
--action "lambda:InvokeFunctionUrl" \
--principal "cloudfront.amazonaws.com" \
--source-arn "arn:aws:cloudfront::123456789012:distribution/E1PDK09ESKHJWT" \
--function-name FUNCTION_URL_NAME
```

> **Note**
>
> Wenn Sie eine Distribution erstellen und diese keine Berechtigung für Ihre Lambda-Funktions-URL hat, können Sie in der CloudFront Konsole den Befehl CLI kopieren auswählen und diesen Befehl dann über Ihr Befehlszeilenterminal eingeben. Weitere Informationen finden Sie AWS-Services im AWS Lambda Entwicklerhandbuch unter Gewähren von Funktionszugriff auf.

Erstellen Sie das OAC

Um ein OAC zu erstellen, können Sie die AWS Management Console, AWS CloudFormation AWS CLI, oder die CloudFront API verwenden.

Console

Um ein OAC zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Origin access (Ursprungszugriff) aus.
3. Wählen Sie Create control setting (Kontrolleinstellung erstellen) aus.
4. Gehen Sie im Formular Neues OAC erstellen wie folgt vor:
   a. Geben Sie einen Namen und (optional) eine Beschreibung für das OAC ein.
   b. Für das Signierverhalten empfehlen wir, die Standardeinstellung beizubehalten (Anfragen signieren (empfohlen)). Weitere Informationen finden Sie unter the section called “Erweiterte Einstellungen für die Ursprungszugriffssteuerung”.
5. Wählen Sie als Origin-Typ Lambda aus.
6. Wählen Sie Erstellen aus.

> **Tip**
>
> Nachdem Sie das OAC erstellt haben, notieren Sie sich den Namen. Sie benötigen diesen im folgenden Verfahren.

So fügen Sie einer Lambda-Funktions-URL in einer Distribution eine Origin-Zugriffskontrolle hinzu

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie eine Distribution mit einer Lambda-Funktions-URL aus, zu der Sie das OAC hinzufügen möchten, und wählen Sie dann die Registerkarte Origins.
3. Wählen Sie die URL der Lambda-Funktion aus, zu der Sie das OAC hinzufügen möchten, und wählen Sie dann Bearbeiten aus.
4. Wählen Sie HTTPS only (Nur HTTPS) für Protocol (Protokoll) Ihres Ursprungs aus.
5. Wählen Sie aus dem Drop-down-Menü für die Origin-Zugriffskontrolle den OAC-Namen aus, den Sie verwenden möchten.
6. Wählen Sie Änderungen speichern aus.

Die Distribution beginnt mit der Bereitstellung an allen CloudFront Edge-Standorten. Wenn ein Edge-Standort die neue Konfiguration empfängt, signiert er alle Anfragen, die er an die URL der Lambda-Funktion sendet.

CloudFormation

Verwenden Sie den `AWS::CloudFront::OriginAccessControl` Ressourcentyp AWS CloudFormation, um ein OAC mit zu erstellen. Das folgende Beispiel zeigt die AWS CloudFormation Vorlagensyntax im YAML-Format für die Erstellung eines OAC.

```
Type: AWS::CloudFront::OriginAccessControl
Properties:
  OriginAccessControlConfig:
      Description: An optional description for the origin access control
      Name: ExampleOAC
      OriginAccessControlOriginType: lambda
      SigningBehavior: always
      SigningProtocol: sigv4
```

Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter AWS::CloudFront::OriginAccessSteuerung.

CLI

Verwenden Sie den aws cloudfront create-origin-access-control Befehl, um eine Origin-Zugriffskontrolle mit dem AWS Command Line Interface (AWS CLI) zu erstellen. Sie können eine Eingabedatei verwenden, um die Eingabeparameter für den Befehl bereitzustellen, anstatt jeden einzelnen Parameter als Befehlszeileneingabe anzugeben.

So erstellen Sie eine Ursprungszugriffssteuerung (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl zum Erstellen einer Datei mit dem Namen `origin-access-control.yaml`. Diese Datei enthält alle Eingabeparameter für den Befehl create-origin-access-control.

   ```
   aws cloudfront create-origin-access-control --generate-cli-skeleton yaml-input >
    origin-access-control.yaml
   ```

2. Öffnen Sie die Datei `origin-access-control.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, um einen Namen für die OAC und eine Beschreibung (optional)

hinzuzufügen, und ändern Sie `SigningBehavior` zu `always`. Speichern Sie dann die Datei.

Weitere Informationen zu anderen OAC-Einstellungen finden Sie unter the section called "Erweiterte Einstellungen für die Ursprungszugriffssteuerung".

3. Verwenden Sie den folgenden Befehl, um die Ursprungszugriffssteuerung mit Eingabeparametern aus der Datei `origin-access-control.yaml` zu erstellen.

```
aws cloudfront create-origin-access-control --cli-input-yaml file://origin-
access-control.yaml
```

Notieren Sie den `Id`-Wert in der Befehlsausgabe. Sie benötigen es, um das OAC zu einer Lambda-Funktions-URL in einer CloudFront Distribution hinzuzufügen.

Um ein OAC an eine Lambda-Funktions-URL in einer vorhandenen Distribution anzuhängen (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um die Verteilungskonfiguration für die CloudFront Distribution zu speichern, zu der Sie das OAC hinzufügen möchten. Die Distribution muss eine Lambda-Funktions-URL als Ursprung haben.

```
aws cloudfront get-distribution-config --id <CloudFront distribution ID> --
output yaml > dist-config.yaml
```

2. Öffnen Sie die Datei mit dem Namen `dist-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und nehmen Sie die folgenden Änderungen vor:

   - Fügen Sie im Objekt `Origins` die ID der OAC dem Feld namens `OriginAccessControlId` hinzu.
   - Entfernen Sie den Wert aus dem Feld namens `OriginAccessIdentity`, sofern vorhanden.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

   Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Verteilung zu aktualisieren und die Ursprungszugriffssteuerung zu verwenden.

```
aws cloudfront update-distribution --id <CloudFront distribution ID> --cli-
input-yaml file://dist-config.yaml
```

Die Verteilung beginnt mit der Bereitstellung an allen CloudFront Edge-Standorten. Wenn ein Edge-Standort die neue Konfiguration empfängt, signiert er alle Anfragen, die er an die URL der Lambda-Funktion sendet.

API

Um ein OAC mit der CloudFront API zu erstellen, verwenden Sie. CreateOriginAccessControl Weitere Informationen zu den Feldern, die Sie in diesem API-Aufruf angeben, finden Sie in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Nachdem Sie ein OAC erstellt haben, können Sie es mit einem der folgenden API-Aufrufe an eine Lambda-Funktions-URL in einer Distribution anhängen:

- Um es an eine bestehende Distribution anzuhängen, verwenden Sie. UpdateDistribution
- Um es an eine neue Distribution anzuhängen, verwenden Sie CreateDistribution.

Geben Sie für diese beiden API-Aufrufe die OAC-ID in das `OriginAccessControlId` Feld innerhalb eines Ursprungs ein. Weitere Informationen zu den anderen Feldern, die Sie in diesen API-Aufrufen angeben, finden Sie in und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Erweiterte Einstellungen für die Ursprungszugriffssteuerung

Die CloudFront OAC-Funktion umfasst erweiterte Einstellungen, die nur für bestimmte Anwendungsfälle vorgesehen sind. Verwenden Sie die empfohlenen Einstellungen, sofern Sie die erweiterten Einstellungen nicht speziell benötigen.

OAC enthält eine Einstellung mit dem Namen Signaturverhalten (in der Konsole) oder `SigningBehavior` (in der API, CLI und AWS CloudFormation). Diese Einstellung bietet die folgenden Optionen:

Ursprungsanforderungen immer signieren (empfohlene Einstellung)

Wir empfehlen die Verwendung dieser Einstellung mit der Bezeichnung Sign requests (recommended) (Anforderungen signieren (empfohlen)) in der Konsole bzw. `always` in der API, CLI und AWS CloudFormation. Mit dieser Einstellung signiert es CloudFront immer alle Anfragen, die es an die URL der Lambda-Funktion sendet.

Ursprungsanforderungen nie signieren

Diese Einstellung heißt Do not sign requests (Anforderungen nicht signieren) in der Konsole bzw. `never` in der API, CLI und AWS CloudFormation. Verwenden Sie diese Einstellung, um OAC für alle Ursprünge in allen Distributionen zu deaktivieren, die dieses OAC verwenden. Dies kann Zeit und Mühe sparen, verglichen mit dem Entfernen eines OAC nacheinander aus allen Origins und Distributionen, die es verwenden. Signiert mit dieser Einstellung CloudFront keine Anfragen, die an die URL der Lambda-Funktion gesendet werden.

> **Warning**
>
> Um diese Einstellung verwenden zu können, muss die URL der Lambda-Funktion öffentlich zugänglich sein. Wenn Sie diese Einstellung mit einer Lambda-Funktions-URL verwenden, die nicht öffentlich zugänglich ist, CloudFront können Sie nicht auf den Ursprung zugreifen. Die URL der Lambda-Funktion gibt Fehler zurück CloudFront und CloudFront leitet diese Fehler an die Betrachter weiter. Weitere Informationen finden Sie unter Sicherheits- und Authentifizierungsmodell für die Lambda-Funktion URLs im AWS Lambda Benutzerhandbuch.

Viewer (Client)-**`Authorization`**-Header nicht überschreiben

Diese Einstellung heißt Do not override authorization header (Autorisierungsheader nicht überschreiben) in der Konsole bzw. `no-override` in der API, CLI und AWS CloudFormation. Verwenden Sie diese Einstellung, wenn Sie ursprüngliche Anfragen nur signieren CloudFront möchten, wenn die entsprechende Viewer-Anfrage keinen Header enthält. `Authorization` Mit dieser Einstellung wird der `Authorization` Header der Viewer-Anfrage weitergegeben, CloudFront wenn eine vorhanden ist, signiert aber die ursprüngliche Anfrage (fügt einen eigenen `Authorization` Header hinzu), wenn die Viewer-Anfrage keinen `Authorization` Header enthält.

⚠ Warning

- Wenn Sie diese Einstellung verwenden, müssen Sie die Signature Version 4-Signatur für die Lambda-Funktions-URL anstelle des Namens oder des CNAME Ihrer CloudFront Distribution angeben. Wenn der `Authorization` Header von der Viewer-Anfrage an die URL der Lambda-Funktion weitergeleitet wird, validiert Lambda die Signatur anhand des Hosts der CloudFront Lambda-URL-Domain. Wenn die Signatur nicht auf der Lambda-URL-Domain basiert, stimmt der Host in der Signatur nicht mit dem Host überein, der vom Lambda-URL-Ursprung verwendet wird. Das bedeutet, dass die Anfrage fehlschlagen wird, was zu einem Fehler bei der Signaturvalidierung führt.
- Um den `Authorization` Header aus der Viewer-Anfrage weiterzugeben, müssen Sie den `Authorization` Header zu einer Cache-Richtlinie für alle Cache-Verhaltensweisen hinzufügen, die die Lambda-Funktion verwenden, die mit dieser Origin-Zugriffskontrolle URLs verknüpft ist.

Beispiel für einen Vorlagencode

Wenn Ihr CloudFront Ursprung eine Lambda-Funktions-URL ist, die mit einem OAC verknüpft ist, können Sie das folgende Python-Skript verwenden, um Dateien mit der Methode in die Lambda-Funktion hochzuladen. `POST`

Bei diesem Code wird davon ausgegangen, dass Sie das OAC so konfiguriert haben, dass das standardmäßige Signaturverhalten auf Anfragen immer signieren eingestellt ist, und dass Sie die Einstellung Autorisierungsheader nicht überschreiben nicht ausgewählt haben.

Diese Konfiguration ermöglicht es dem OAC, die SigV4-Autorisierung mit Lambda mithilfe des Lambda-Hostnamens korrekt zu verwalten. Die Payload wird mithilfe von SigV4 aus der IAM-Identität signiert, die für die Lambda-Funktions-URL autorisiert ist, die als Typ bezeichnet wird. `IAM_AUTH`

Die Vorlage zeigt, wie signierte Payload-Hashwerte im x-amz-content-sha256 Header für `POST` Anfragen von der Clientseite behandelt werden. Diese Vorlage wurde speziell für die Verwaltung von Formulardaten-Nutzlasten entwickelt. Die Vorlage ermöglicht sichere Datei-Uploads an eine Lambda-Funktions-URL und verwendet AWS Authentifizierungsmechanismen CloudFront, um sicherzustellen, dass nur autorisierte Anfragen auf die Lambda-Funktion zugreifen können.

Der Code umfasst die folgenden Funktionen:

- Erfüllt die Anforderung, den Payload-Hash in den Header aufzunehmen x-amz-content-sha256
- Verwendet die SigV4-Authentifizierung für sicheren Zugriff AWS-Service
- Unterstützt Datei-Uploads mithilfe mehrteiliger Formulardaten
- Beinhaltet die Fehlerbehandlung für Anforderungsausnahmen

```
import boto3
from botocore.auth import SigV4Auth
from botocore.awsrequest import AWSRequest
import requests
import hashlib
import os


def calculate_body_hash(body):
    return hashlib.sha256(body).hexdigest()


def sign_request(request, credentials, region, service):
    sigv4 = SigV4Auth(credentials, service, region)
    sigv4.add_auth(request)


def upload_file_to_lambda(cloudfront_url, file_path, region):
    # AWS credentials
    session = boto3.Session()
    credentials = session.get_credentials()

    # Prepare the multipart form-data
    boundary = "------------------------boundary"

    # Read file content
    with open(file_path, 'rb') as file:
        file_content = file.read()

    # Get the filename from the path
    filename = os.path.basename(file_path)
```

```
    # Prepare the multipart body
    body = (
        f'--{boundary}\r\n'
        f'Content-Disposition: form-data; name="file"; filename="{filename}"\r\n'
        f'Content-Type: application/octet-stream\r\n\r\n'
    ).encode('utf-8')
    body += file_content
    body += f'\r\n--{boundary}--\r\n'.encode('utf-8')

    # Calculate SHA256 hash of the entire body
    body_hash = calculate_body_hash(body)

    # Prepare headers
    headers = {
        'Content-Type': f'multipart/form-data; boundary={boundary}',
        'x-amz-content-sha256': body_hash
    }

    # Create the request
    request = AWSRequest(
        method='POST',
        url=cloudfront_url,
        data=body,
        headers=headers
    )

    # Sign the request
    sign_request(request, credentials, region, 'lambda')

    # Get the signed headers
    signed_headers = dict(request.headers)

    # Print request headers before sending
    print("Request Headers:")
    for header, value in signed_headers.items():
        print(f"{header}: {value}")

    try:
        # Send POST request with signed headers
        response = requests.post(
            cloudfront_url,
            data=body,
            headers=signed_headers
```

```
        )

        # Print response status and content
        print(f"\nStatus code: {response.status_code}")
        print("Response:", response.text)

        # Print response headers
        print("\nResponse Headers:")
        for header, value in response.headers.items():
            print(f"{header}: {value}")

    except requests.exceptions.RequestException as e:
        print(f"An error occurred: {e}")


# Usage
cloudfront_url = "https://d111111abcdef8.cloudfront.net"
file_path = r"filepath"
region = "us-east-1"  # example: "us-west-2"

upload_file_to_lambda(cloudfront_url, file_path, region)
```

Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung

CloudFront bietet zwei Möglichkeiten, authentifizierte Anfragen an einen Amazon S3 S3-Ursprung zu senden: Origin Access Control (OAC) und Origin Access Identity (OAI). OAC hilft Ihnen, Ihre Ursprünge zu sichern, z. B. Amazon S3.

Wir empfehlen, stattdessen OAC zu verwenden, da es die folgenden Funktionen unterstützt:

- Alle Amazon S3 S3-Buckets insgesamt AWS-Regionen, einschließlich der Opt-in-Regionen, die nach Dezember 2022 eingeführt wurden
- Serverseitige Verschlüsselung mit AWS KMS (SSE-KMS) von Amazon S3
- Dynamische Anforderungen (`PUT` und `DELETE`) an Amazon S3

OAI unterstützt diese Funktionen nicht oder erfordert in diesen Szenarien zusätzliche Behelfslösungen. Wenn Sie OAI bereits verwenden und migrieren möchten, finden Sie weitere Informationen unter. the section called “Migration von der Ursprungszugriffsidentität (OAI) zur Ursprungszugriffssteuerung (OAC)”

Hinweise

- Wenn Sie CloudFront OAC mit Amazon S3-Bucket-Ursprüngen verwenden, müssen Sie Amazon S3 Object Ownership auf Bucket owner enforced setzen, die Standardeinstellung für neue Amazon S3 S3-Buckets. Verwenden Sie bei Bedarf die bevorzugte Einstellung des Bucket-Besitzers ACLs, um die Kontrolle über Objekte zu behalten, die über hochgeladen werden. CloudFront
- Wenn Ihr Ursprung ein Amazon S3 S3-Bucket ist, der als Website-Endpunkt konfiguriert ist, müssen Sie ihn CloudFront als benutzerdefinierten Ursprung einrichten. Das bedeutet, dass Sie OAC (oder OAI) nicht verwenden können. OAC unterstützt keine Origin-Umleitung mithilfe von Lambda @Edge.

In den folgenden Themen wird die Verwendung von OAC mit einem Amazon-S3-Ursprung beschrieben.

Topics

- the section called “Erstellen Sie eine neue Origin-Zugriffskontrolle”
- the section called “Löschen Sie eine Distribution, bei der ein OAC an einen S3-Bucket angehängt ist”
- the section called “Migration von der Ursprungszugriffsidentität (OAI) zur Ursprungszugriffssteuerung (OAC)”
- the section called “Erweiterte Einstellungen für die Ursprungszugriffssteuerung”

Erstellen Sie eine neue Origin-Zugriffskontrolle

Führen Sie die in den folgenden Themen beschriebenen Schritte aus, um eine neue Origin-Zugriffskontrolle in einzurichten CloudFront.

Themen

- Voraussetzungen
- Erteilen Sie die CloudFront Erlaubnis, auf den S3-Bucket zuzugreifen
- Erstellen Sie die Origin-Zugriffskontrolle

Voraussetzungen

Bevor Sie Origin Access Control (OAC) erstellen und einrichten, benötigen Sie eine CloudFront Distribution mit einem Amazon S3 S3-Bucket-Ursprung. Dieser Ursprung muss ein regulärer S3-Bucket sein, kein Bucket, der als Website-Endpunkt konfiguriert wurde. Weitere Informationen zum Einrichten einer CloudFront Distribution mit einem S3-Bucket-Ursprung finden Sie unterthe section called “Beginnen Sie mit einer Standarddistribution”.

> **Important**
>
> Wenn Sie OAC verwenden, um Ihren Amazon S3-Ursprung zu sichern, erfolgt die Kommunikation zwischen Amazon S3 CloudFront und Amazon S3 immer über HTTPS, aber nur, wenn Sie sich dafür entscheiden, Anfragen immer zu signieren. Sie müssen in der Konsole die Option Anfragen signieren (empfohlen) auswählen oder `always` in der CloudFront API angeben, AWS CLI, oder CloudFormation.
> Wenn Sie stattdessen entweder die Option Anfragen nicht signieren oder Autorisierungsheader nicht überschreiben wählen, CloudFront wird das Verbindungsprotokoll verwendet, das Sie in den folgenden Richtlinien angegeben haben:
>
> - Viewer-Protokollrichtlinie
> - Origin-Protokollrichtlinie (nur benutzerdefinierte Ursprünge)
>
> Wenn Sie beispielsweise „Autorisierungsheader nicht überschreiben“ wählen und HTTPS zwischen CloudFront und Ihrem Amazon S3 S3-Ursprung verwenden möchten, verwenden Sie „HTTP zu HTTPS umleiten“ oder „Nur HTTPS“ für die Viewer-Protokollrichtlinie.

Erteilen Sie die CloudFront Erlaubnis, auf den S3-Bucket zuzugreifen

Bevor Sie eine Origin-Zugriffskontrolle (OAC) erstellen oder sie in einer CloudFront Distribution einrichten, stellen Sie sicher, dass diese Person über die Zugriffsberechtigung für den S3-Bucket-Ursprung CloudFront verfügt. Tun Sie dies, nachdem Sie eine CloudFront Distribution erstellt haben, aber bevor Sie das OAC zum S3-Ursprung in der Distributionskonfiguration hinzufügen.

Verwenden Sie eine S3-Bucket-Richtlinie, um dem CloudFront Dienstprinzipal (`cloudfront.amazonaws.com`) den Zugriff auf den Bucket zu ermöglichen. Verwenden Sie ein `Condition` Element in der Richtlinie, CloudFront um nur dann auf den Bucket zuzugreifen, wenn

die Anfrage im Namen der CloudFront Distribution erfolgt, die den S3-Ursprung enthält. Dies ist die Distribution mit dem S3-Ursprung, zu der Sie OAC hinzufügen möchten.

Informationen zum Hinzufügen oder Ändern einer Bucket-Richtlinie finden Sie unter Hinzufügen einer Bucket-Richtlinie mit der Amazon-S3-Konsole im Amazon-S3-Benutzerhandbuch.

Im Folgenden finden Sie Beispiele für S3-Bucket-Richtlinien, die einer CloudFront Distribution mit OAC-aktiviertem Zugriff auf einen S3-Ursprung ermöglichen.

Example S3-Bucket-Richtlinie, die nur Lesezugriff für eine CloudFront Distribution mit aktiviertem OAC ermöglicht

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCloudFrontServicePrincipalReadOnly",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
      "Condition": {
        "StringEquals": {
          "AWS:SourceArn":
 "arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
        }
      }
    }
  ]
}
```

Example S3-Bucket-Richtlinie, die Lese- und Schreibzugriff für eine CloudFront Distribution mit aktiviertem OAC ermöglicht

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCloudFrontServicePrincipalReadWrite",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
      "Condition": {
        "StringEquals": {
          "AWS:SourceArn":
 "arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
        }
      }
    }
  ]
}
```

SSE-KMS

Wenn die Objekte im S3-Bucket-Ursprung mit serverseitiger Verschlüsselung mit AWS Key Management Service (SSE-KMS) verschlüsselt sind, müssen Sie sicherstellen, dass die CloudFront Distribution über die Berechtigung zur Verwendung des Schlüssels verfügt. AWS KMS Um der CloudFront Distribution die Erlaubnis zur Verwendung des KMS-Schlüssels zu erteilen, fügen Sie der KMS-Schlüsselrichtlinie eine Erklärung hinzu. Weitere Informationen zum Ändern einer Schlüsselrichtlinie finden Sie unter Ändern einer Schlüsselrichtlinie im AWS Key Management Service -Entwicklerhandbuch.

Example Erklärung zur KMS-Schlüsselrichtlinie

Das folgende Beispiel zeigt eine AWS KMS Richtlinienanweisung, die der CloudFront Distribution mit OAC den Zugriff auf einen KMS-Schlüssel für SSE-KMS ermöglicht.

```
{
    "Sid": "AllowCloudFrontServicePrincipalSSE-KMS",
    "Effect": "Allow",
    "Principal": {
        "Service": [
            "cloudfront.amazonaws.com"
        ]
     },
    "Action": [
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:GenerateDataKey*"
    ],
    "Resource": "*",
    "Condition": {
            "StringEquals": {
                "AWS:SourceArn":
 "arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
            }
        }
}
```

Erstellen Sie die Origin-Zugriffskontrolle

Um eine Origin-Zugriffskontrolle (OAC) zu erstellen, können Sie die AWS Management Console, AWS CloudFormation AWS CLI, oder die CloudFront API verwenden.

Console

So erstellen Sie eine Ursprungszugriffssteuerung

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Origin access (Ursprungszugriff) aus.
3. Wählen Sie Create control setting (Kontrolleinstellung erstellen) aus.
4. Gehen Sie im Formular Create control setting (Kontrolleinstellung erstellen) wie folgt vor:

a. Geben Sie im Bereich Details einen Namen (Name) und (optional) eine Beschreibung (Description) für die Ursprungszugriffssteuerung ein.

b. Es empfiehlt sich, im Bereich Settings (Einstellungen) die Standardeinstellung (Sign requests (recommended)) (Anforderungen signieren (empfohlen)) zu belassen. Weitere Informationen finden Sie unter the section called “Erweiterte Einstellungen für die Ursprungszugriffssteuerung”.

5. Wählen Sie „S3“ aus der Dropdown-Liste Origin type (Ursprungstyp) aus.
6. Wählen Sie Erstellen aus.

   Nachdem die OAC erstellt wurde, notieren Sie sich den Namen. Sie benötigen diesen im folgenden Verfahren.

So fügen Sie eine Ursprungszugriffssteuerung einem S3-Ursprung in einer Verteilung hinzu

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie eine Verteilung mit einem S3-Ursprung aus, der Sie die OAC hinzufügen möchten, und wählen Sie dann den Tab Origins (Ursprünge) aus.
3. Wählen Sie den S3-Ursprung aus, dem Sie die OAC hinzufügen möchten, und wählen Sie dann Edit (Bearbeiten) aus.
4. Wähle für Origin-Zugriff die Einstellungen für die Origin-Zugriffskontrolle (empfohlen).
5. Wählen Sie im Dropdown-Menü Origin access control (Ursprungszugriffssteuerung) die OAC aus, die Sie verwenden möchten.
6. Wählen Sie Änderungen speichern aus.

Die Distribution beginnt mit der Bereitstellung an allen CloudFront Edge-Standorten. Wenn ein Edge-Standort die neue Konfiguration erhält, signiert er alle Anforderungen, die er an den S3-Bucket-Ursprung sendet.

CloudFormation

Verwenden Sie den `AWS::CloudFront::OriginAccessControl` Ressourcentyp AWS CloudFormation, um eine Origin Access Control (OAC) mit zu erstellen. Das folgende Beispiel zeigt die AWS CloudFormation Vorlagensyntax im YAML-Format für die Erstellung einer Origin-Zugriffskontrolle.

```
Type: AWS::CloudFront::OriginAccessControl
Properties:
  OriginAccessControlConfig:
      Description: An optional description for the origin access control
      Name: ExampleOAC
      OriginAccessControlOriginType: s3
      SigningBehavior: always
      SigningProtocol: sigv4
```

Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter AWS::CloudFront::OriginAccessSteuerung.

CLI

Verwenden Sie den aws cloudfront create-origin-access-control Befehl, um eine Origin-Zugriffskontrolle mit dem AWS Command Line Interface (AWS CLI) zu erstellen. Sie können eine Eingabedatei verwenden, um die Eingabeparameter für den Befehl bereitzustellen, anstatt jeden einzelnen Parameter als Befehlszeileneingabe anzugeben.

So erstellen Sie eine Ursprungszugriffssteuerung (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl zum Erstellen einer Datei mit dem Namen `origin-access-control.yaml`. Diese Datei enthält alle Eingabeparameter für den Befehl create-origin-access-control.

   ```
   aws cloudfront create-origin-access-control --generate-cli-skeleton yaml-input >
    origin-access-control.yaml
   ```

2. Öffnen Sie die Datei `origin-access-control.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, um einen Namen für die OAC und eine Beschreibung (optional) hinzuzufügen, und ändern Sie `SigningBehavior` zu `always`. Speichern Sie dann die Datei.

   Weitere Informationen zu anderen OAC-Einstellungen finden Sie unter the section called “Erweiterte Einstellungen für die Ursprungszugriffssteuerung”.

3. Verwenden Sie den folgenden Befehl, um die Ursprungszugriffssteuerung mit Eingabeparametern aus der Datei `origin-access-control.yaml` zu erstellen.

```
aws cloudfront create-origin-access-control --cli-input-yaml file://origin-
access-control.yaml
```

Notieren Sie den `Id`-Wert in der Befehlsausgabe. Sie benötigen ihn, um das OAC zu einem S3-Bucket-Ursprung in einer CloudFront Distribution hinzuzufügen.

So fügen Sie eine OAC einem S3-Bucket-Ursprung in einer vorhandenen Verteilung an (CLI mit Eingabedatei)

1. Verwenden Sie den folgenden Befehl, um die Verteilungskonfiguration für die CloudFront Distribution zu speichern, zu der Sie das OAC hinzufügen möchten. Die Verteilung muss über einen S3-Bucket-Ursprung verfügen.

   ```
   aws cloudfront get-distribution-config --id <CloudFront distribution ID> --
   output yaml > dist-config.yaml
   ```

2. Öffnen Sie die Datei mit dem Namen `dist-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei und nehmen Sie die folgenden Änderungen vor:

   - Fügen Sie im Objekt `Origins` die ID der OAC dem Feld namens `OriginAccessControlId` hinzu.
   - Entfernen Sie den Wert aus dem Feld namens `OriginAccessIdentity`, sofern vorhanden.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

   Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Verteilung zu aktualisieren und die Ursprungszugriffssteuerung zu verwenden.

   ```
   aws cloudfront update-distribution --id <CloudFront distribution ID> --cli-
   input-yaml file://dist-config.yaml
   ```

Die Verteilung beginnt mit der Bereitstellung an allen CloudFront Edge-Standorten. Wenn ein Edge-Standort die neue Konfiguration erhält, signiert er alle Anforderungen, die er an den S3-Bucket-Ursprung sendet.

API

Um eine Origin-Zugriffskontrolle mit der CloudFront API zu erstellen, verwenden Sie CreateOriginAccessControl. Weitere Informationen zu den Feldern, die Sie in diesem API-Aufruf angeben, finden Sie in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Nachdem Sie eine Ursprungszugriffssteuerung erstellt haben, können Sie diese mit einem der folgenden API-Aufrufe an einen S3-Bucket-Ursprung in einer Verteilung anfügen:

- Um es an eine bestehende Distribution anzuhängen, verwenden Sie UpdateDistribution.
- Um es an eine neue Distribution anzuhängen, verwenden Sie CreateDistribution.

Geben Sie für beide API-Aufrufe die ID der Ursprungszugriffssteuerung im Feld `OriginAccessControlId` innerhalb eines Ursprungs an. Weitere Informationen zu den anderen Feldern, die Sie in diesen API-Aufrufen angeben, finden Sie unter Referenz für alle Verteilungseinstellungen und in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Löschen Sie eine Distribution, bei der ein OAC an einen S3-Bucket angehängt ist

Wenn Sie eine Distribution löschen müssen, deren OAC an einen S3-Bucket angehängt ist, sollten Sie die Distribution löschen, bevor Sie den S3-Bucket-Ursprung löschen. Sie können auch die Region in den Namen der Ursprungs-Domain aufnehmen. Wenn dies nicht möglich ist, können Sie das OAC aus der Distribution entfernen, indem Sie vor dem Löschen auf „Öffentlich“ wechseln. Weitere Informationen finden Sie unter Löschen einer -Verteilung.

Migration von der Ursprungszugriffsidentität (OAI) zur Ursprungszugriffssteuerung (OAC)

Um von einer älteren Origin-Zugriffsidentität (OAI) zu einer Origin-Zugriffskontrolle (OAC) zu migrieren, aktualisieren Sie zunächst den S3-Bucket-Ursprung, sodass sowohl die OAI als auch die Distribution mit aktiviertem OAC auf den Inhalt des Buckets zugreifen können. Dadurch wird sichergestellt, dass während der Umstellung CloudFront nie der Zugriff auf den Bucket verloren

geht. Damit sowohl OAI als auch die Distribution mit aktiviertem OAC auf einen S3-Bucket zugreifen können, aktualisieren Sie die Bucket-Richtlinie so, dass sie zwei Anweisungen enthält, eine für jede Art von Principal.

Das folgende Beispiel für eine S3-Bucket-Richtlinie ermöglicht sowohl einer OAI als auch einer Distribution mit aktiviertem OAC den Zugriff auf einen S3-Ursprung.

Example S3-Bucket-Richtlinie, die nur Lesezugriff für eine OAI und eine Distribution mit aktiviertem OAC ermöglicht CloudFront

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCloudFrontServicePrincipalReadOnly",
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudfront.amazonaws.com"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::<S3 bucket name>/*",
            "Condition": {
                "StringEquals": {
                    "AWS:SourceArn":
 "arn:aws:cloudfront::111122223333:distribution/<CloudFront distribution ID>"
                }
            }
        },
        {
            "Sid": "AllowLegacyOAIReadOnly",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access
 Identity <origin access identity ID>"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::<S3 bucket name>/*"
        }
    ]
}
```

Nachdem Sie die Bucket-Richtlinie des S3-Ursprungs aktualisiert haben, um den Zugriff sowohl für OAI als auch für OAC zu ermöglichen, können Sie die Verteilungskonfiguration so aktualisieren, dass OAC anstelle von OAI verwendet wird. Weitere Informationen finden Sie unter the section called “Erstellen Sie eine neue Origin-Zugriffskontrolle”.

Nachdem die Verteilung vollständig bereitgestellt wurde, können Sie die Anweisung in der Bucket-Richtlinie, die den Zugriff für die OAI ermöglicht, entfernen. Weitere Informationen finden Sie unter the section called “Erteilen Sie die CloudFront Erlaubnis, auf den S3-Bucket zuzugreifen”.

Erweiterte Einstellungen für die Ursprungszugriffssteuerung

Die CloudFront Origin-Zugriffskontrollfunktion umfasst erweiterte Einstellungen, die nur für bestimmte Anwendungsfälle vorgesehen sind. Verwenden Sie die empfohlenen Einstellungen, sofern Sie die erweiterten Einstellungen nicht speziell benötigen.

Die Origin-Zugriffskontrolle enthält eine Einstellung namens Signaturverhalten (in der Konsole) oder `SigningBehavior` (in der API, CLI und AWS CloudFormation). Diese Einstellung bietet die folgenden Optionen:

Ursprungsanforderungen immer signieren (empfohlene Einstellung)

Wir empfehlen die Verwendung dieser Einstellung mit der Bezeichnung Sign requests (recommended) (Anforderungen signieren (empfohlen)) in der Konsole bzw. `always` in der API, CLI und AWS CloudFormation. Mit dieser Einstellung signiert es CloudFront immer alle Anfragen, die es an den S3-Bucket-Ursprung sendet.

Ursprungsanforderungen nie signieren

Diese Einstellung heißt Do not sign requests (Anforderungen nicht signieren) in der Konsole bzw. `never` in der API, CLI und AWS CloudFormation. Verwenden Sie diese Einstellung, um die Ursprungszugriffssteuerung für alle Ursprünge in allen Verteilungen zu deaktivieren, die diese Ursprungszugriffssteuerung verwenden. Auf diese Weise lässt sich im Vergleich zum einzelnen Entfernen einer Ursprungszugriffssteuerung aus allen Ursprüngen und Verteilungen, die diese verwenden, Zeit und Aufwand sparen. Mit dieser Einstellung signiert es CloudFront keine Anfragen, die es an den S3-Bucket-Ursprung sendet.

> **Warning**
>
> Um diese Einstellung verwenden zu können, muss der S3-Bucket-Ursprung öffentlich zugänglich sein. Wenn Sie diese Einstellung mit einem S3-Bucket-Ursprung verwenden,

der nicht öffentlich zugänglich ist, CloudFront können Sie nicht auf den Ursprung zugreifen. Der S3-Bucket-Ursprung gibt Fehler zurück CloudFront und CloudFront leitet diese Fehler an die Zuschauer weiter.

Viewer (Client)-**`Authorization`**-Header nicht überschreiben

Diese Einstellung heißt Do not override authorization header (Autorisierungsheader nicht überschreiben) in der Konsole bzw. `no-override` in der API, CLI und AWS CloudFormation. Verwenden Sie diese Einstellung, wenn Sie ursprüngliche Anfragen nur signieren CloudFront möchten, wenn die entsprechende Viewer-Anfrage keinen `Authorization` Header enthält. Mit dieser Einstellung wird der `Authorization` Header der Viewer-Anfrage weitergegeben, CloudFront wenn eine vorhanden ist, signiert aber die ursprüngliche Anfrage (fügt einen eigenen `Authorization` Header hinzu), wenn die Viewer-Anfrage keinen `Authorization` Header enthält.

> **Warning**
>
> Um den `Authorization`-Header aus der Viewer-Anforderung zu übergeben, müssen Sie den `Authorization`-Header zu einer Cache-Richtlinie für alle Cache-Verhaltensweisen hinzufügen, die S3-Bucket-Ursprünge verwenden, die dieser Ursprungszugriffssteuerung zugeordnet sind.

Verwenden Sie eine ursprüngliche Zugriffsidentität (veraltet, nicht empfohlen)

Übersicht über das die Ursprungszugriffsidentität

CloudFront Origin Access Identity (OAI) bietet ähnliche Funktionen wie Origin Access Control (OAC), funktioniert aber nicht für alle Szenarien. Insbesondere unterstützt OAI Folgendes nicht:

- Amazon S3 S3-Buckets in allen Regionen AWS-Regionen, einschließlich optionaler Regionen
- Serverseitige Verschlüsselung mit AWS KMS (SSE-KMS) von Amazon S3
- Dynamische Anforderungen (`PUT`, `POST` oder `DELETE`) an Amazon S3
- Nach Januar 2023 neu AWS-Regionen auf den Markt gebracht

> **Tip**
>
> Wir empfehlen, stattdessen OAC zu verwenden. Informationen zum Einrichten von OAC finden Sie unter. Erstellen Sie eine neue Origin-Zugriffskontrolle Weitere Informationen zur Migration von OAI zu OAC finden Sie unter the section called “Migration von der Ursprungszugriffsidentität (OAI) zur Ursprungszugriffssteuerung (OAC)”.

Erteilen Sie einer Origin-Zugriffsidentität die Erlaubnis, Dateien im Amazon S3 S3-Bucket zu lesen

Wenn Sie mit der CloudFront Konsole eine OAI erstellen oder einer Distribution hinzufügen, können Sie die Amazon S3 S3-Bucket-Richtlinie automatisch aktualisieren, um der OAI Zugriff auf Ihren Bucket zu gewähren. Alternativ können Sie die Bucket-Richtlinie manuell erstellen oder aktualisieren. Unabhängig davon, welche Methode Sie verwenden, sollten Sie die Berechtigungen dennoch überprüfen, um Folgendes sicherzustellen:

- Ihre CloudFront OAI kann im Namen von Zuschauern, die sie anfordern, auf Dateien im Bucket zugreifen. CloudFront
- Zuschauer können Amazon S3 nicht verwenden URLs , um außerhalb von auf Ihre Dateien zuzugreifen CloudFront.

> **Important**
>
> Wenn Sie so konfigurieren CloudFront , dass alle CloudFront unterstützten HTTP-Methoden akzeptiert und weitergeleitet werden, stellen Sie sicher, dass Sie Ihrer CloudFront OAI die gewünschten Berechtigungen erteilen. Wenn Sie beispielsweise so konfigurieren CloudFront , dass Anfragen, die diese `DELETE` Methode verwenden, angenommen und weitergeleitet werden, konfigurieren Sie Ihre Bucket-Richtlinie so, dass `DELETE` Anfragen entsprechend behandelt werden, sodass Zuschauer nur Dateien löschen können, die Sie möchten.

Verwenden Sie Amazon S3 S3-Bucket-Richtlinien

Sie können einer CloudFront OAI Zugriff auf Dateien in einem Amazon S3 S3-Bucket gewähren, indem Sie die Bucket-Richtlinie auf folgende Weise erstellen oder aktualisieren:

- Verwenden des Tabs Permissions (Berechtigungen) des Amazon-S3-Buckets in der Amazon-S3-Konsole.

- Verwendung PutBucketPolicyin der Amazon S3 S3-API.
- Verwenden der CloudFront-Konsole Wenn Sie Ihren Origin-Einstellungen in der CloudFront Konsole eine OAI hinzufügen, können Sie Ja, die Bucket-Richtlinie aktualisieren wählen, CloudFront um mitzuteilen, dass die Bucket-Richtlinie in Ihrem Namen aktualisiert werden soll.

Wenn Sie die Bucket-Richtlinie manuell aktualisieren, stellen Sie Folgendes sicher:

- Geben Sie die korrekte OAI als das `Principal` in der Richtlinie an.
- Erteilen Sie der OAI die Berechtigungen, die es für den Zugriff auf Objekte im Auftrag von Viewern benötigt.

Weitere Informationen finden Sie in den folgenden Abschnitten.

Eine OAI als den **`Principal`** in einer Bucket-Richtlinie angeben

Um eine OAI als `Principal` in einer Amazon-S3-Bucket-Richtlinie anzugeben, verwenden Sie den Amazon-Ressourcennamen (ARN) der OAI, der die ID der OAI enthält. Zum Beispiel:

```
"Principal": {
    "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity <origin
 access identity ID>"
}
```

Finde die OAI-ID in der CloudFront Konsole unter Sicherheit, Origin-Zugriff, Identitäten (veraltet). Alternativ können Sie sie ListCloudFrontOriginAccessIdentitiesin der CloudFront API verwenden.

Erteilen von Berechtigungen für eine OAI

Um der OAI die Berechtigungen für den Zugriff auf Objekte in Ihrem Amazon-S3-Bucket zu erteilen, verwenden Sie Aktionen in der Richtlinie, die sich auf bestimmte Amazon-S3-API-Operationen beziehen. Die `s3:GetObject`-Aktion ermöglicht es der OAI beispielsweise, Objekte im Bucket zu lesen. Weitere Informationen finden Sie in den Beispielen im folgenden Abschnitt oder unter Amazon S3-Aktionen im Benutzerhandbuch zu Amazon Simple Storage Service.

Beispiele für Amazon-S3-Bucket-Richtlinien

Die folgenden Beispiele zeigen Amazon S3 S3-Bucket-Richtlinien, die es CloudFront OAI ermöglichen, auf einen S3-Bucket zuzugreifen.

Suchen Sie die OAI-ID in der CloudFront Konsole unter Sicherheit, Origin-Zugriff, Identitäten (veraltet). Alternativ können Sie sie ListCloudFrontOriginAccessIdentitiesin der CloudFront API verwenden.

Example Amazon-S3-Bucket-Richtlinie, die der OAI Lesezugriff gewährt

Im folgenden Beispiel kann die OAI Objekte im angegebenen Bucket (`s3:GetObject`) lesen.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForCloudFrontPrivateContent",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access
 Identity <origin access identity ID>"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::<S3 bucket name>/*"
        }
    ]
}
```

Example Amazon-S3-Bucket-Richtlinie, die der OAI Lese- und Schreibzugriff gewährt

Im folgenden Beispiel kann die OAI Objekte im angegebenen Bucket (`s3:GetObject` und `s3:PutObject`) lesen und schreiben. Auf diese Weise können Zuschauer Dateien in Ihren Amazon S3 S3-Bucket hochladen CloudFront.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForCloudFrontPrivateContent",
    "Statement": [
        {
            "Effect": "Allow",
```

```
            "Principal": {
                "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access
 Identity <origin access identity ID>"
            },
            "Action": [
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Resource": "arn:aws:s3:::<S3 bucket name>/*"
        }
    ]
}
```

Amazon S3 S3-Objekt verwenden ACLs (nicht empfohlen)

> **Important**
>
> Es wird empfohlen, Amazon-S3-Bucket-Richtlinien zu verwenden, um einer OAI Zugriff auf einen S3-Bucket zu gewähren. Sie können Zugriffskontrolllisten (ACLs) wie in diesem Abschnitt beschrieben verwenden, wir empfehlen dies jedoch nicht.
> Amazon S3 empfiehlt, S3 Object Ownership auf Bucket Owner Enforced zu setzen, was bedeutet, dass diese für den Bucket und die darin enthaltenen Objekte deaktiviert ACLs sind. Wenn Sie diese Einstellung für Objekteigentümerschaft anwenden, müssen Sie Bucket-Richtlinien verwenden, um Zugriff auf die OAI zu gewähren (siehe vorheriger Abschnitt).
> Der folgende Abschnitt bezieht sich nur auf ältere Anwendungsfälle, die dies erfordern ACLs.

Sie können einer CloudFront OAI Zugriff auf Dateien in einem Amazon S3 S3-Bucket gewähren, indem Sie die ACL der Datei auf folgende Weise erstellen oder aktualisieren:

- Verwenden des Tabs Permissions (Berechtigungen) des Amazon-S3-Objekts in der Amazon-S3-Konsole.
- Verwendung PutObjectAclin der Amazon S3 S3-API.

Wenn Sie einer OAI über eine ACL Zugriff gewähren, müssen Sie die OAI mit seiner kanonischen Amazon-S3-Benutzer-ID angeben. In der CloudFront Konsole finden Sie diese ID unter Sicherheit, Origin-Zugriff, Identitäten (veraltet). Wenn Sie die CloudFront API verwenden, verwenden Sie den

Wert des `S3CanonicalUserId` Elements, das bei der Erstellung der OAI zurückgegeben wurde, oder rufen Sie die CloudFront API ListCloudFrontOriginAccessIdentitiesauf.

Verwenden Sie eine Ursprungszugriffsidentität in Amazon S3 S3-Regionen, die nur die Authentifizierung mit Signaturversion 4 unterstützen

In neueren Amazon-S3-Regionen ist es erforderlich, Signature Version 4 für authentifizierte Anforderungen zu verwenden. (Informationen zu den Signaturversionen, die in jeder Amazon-S3-Region unterstützt werden, finden Sie unter Amazon-Simple-Storage-Service-Endpunkte und -Kontingente in der Allgemeine AWS-Referenz.) Wenn Sie eine Ursprungszugriffsidentität verwenden und sich Ihr Bucket in einer der Regionen befindet, die Signature Version 4 erfordert, beachten Sie Folgendes:

- `DELETE`-, `GET`-, `HEAD`-, `OPTIONS`- und `PATCH`-Anfragen werden ohne Einschränkungen unterstützt.
- `POST`-Anfragen werden nicht unterstützt.

Beschränken Sie den Zugriff mit VPC-Ursprüngen

Sie können CloudFront damit Inhalte aus Anwendungen bereitstellen, die in Ihren privaten VPC-Subnetzen (Virtual Private Cloud) gehostet werden. Sie können Application Load Balancers (ALBs), Network Load Balancers (NLBs) und EC2 Instances in privaten Subnetzen als VPC-Ursprünge verwenden.

Hier sind einige Gründe, warum Sie VPC Origins verwenden sollten:

- Sicherheit — VPC Origins wurde entwickelt, um den Sicherheitsstatus Ihrer Anwendung zu verbessern, indem Ihre Load Balancer und EC2 Instances in privaten Subnetzen platziert werden, wodurch CloudFront der zentrale Einstiegspunkt entsteht. Benutzeranfragen werden über eine private, sichere Verbindung CloudFront zu den VPC-Ursprüngen weitergeleitet und bieten so zusätzliche Sicherheit für Ihre Anwendungen.
- Verwaltung — VPC-Ursprünge reduzieren den Betriebsaufwand, der für eine sichere Konnektivität zwischen den CloudFront Ursprüngen erforderlich ist. Sie können Ihre Ursprünge in private Subnetze ohne öffentlichen Zugriff verschieben, und Sie müssen keine Zugriffskontrolllisten (ACLs) oder andere Mechanismen implementieren, um den Zugriff auf Ihre Ursprünge einzuschränken. Auf diese Weise müssen Sie nicht in undifferenzierte Entwicklungsarbeit investieren, um Ihre Webanwendungen damit zu schützen. CloudFront
- Skalierbarkeit und Leistung — VPC Origins hilft Ihnen dabei, Ihre Webanwendungen zu sichern, sodass Sie Zeit haben, sich auf das Wachstum Ihrer kritischen Geschäftsanwendungen

zu konzentrieren und gleichzeitig die Sicherheit zu verbessern und gleichzeitig eine hohe Leistung und globale Skalierbarkeit aufrechtzuerhalten. CloudFront VPC Origins optimiert das Sicherheitsmanagement und reduziert die betriebliche Komplexität, sodass Sie es CloudFront als zentrale Anlaufstelle für Ihre Anwendungen verwenden können.

Voraussetzungen

Bevor Sie einen VPC-Ursprung für Ihre CloudFront Distribution erstellen, müssen Sie die folgenden Schritte ausführen:

- Erstellen Sie eine virtuelle private Cloud (VPC) auf Amazon VPC.
  - Ihre VPC muss sich in derselben Distribution befinden AWS-Konto wie Ihre CloudFront Distribution.
  - Ihre VPC muss sich in einer der Regionen befinden AWS-Regionen , die für VPC-Ursprünge unterstützt werden. Weitere Informationen finden Sie unter Wird AWS-Regionen für VPC-Ursprünge unterstützt.
  - Das mit Ihren VPC-Subnetzen ACLs verknüpfte Netzwerk gilt für ausgehenden (ausgehenden) Datenverkehr, wenn die Beibehaltung der Client-IP-Adresse auf Ihrem VPC-Ursprung aktiviert ist. Damit der Datenverkehr jedoch über Ihren VPC-Ursprung verlassen kann, müssen Sie die ACL sowohl als eingehende als auch als ausgehende Regel konfigurieren.

    Um beispielsweise TCP- und UDP-Clients, die einen kurzlebigen Quellport verwenden, über Ihren VPC-Ursprung eine Verbindung zu Ihrem Endpunkt herzustellen, verknüpfen Sie das Subnetz Ihres Endpunkts mit einer Netzwerk-ACL, die ausgehenden Datenverkehr für einen kurzlebigen TCP- oder UDP-Port (Portbereich 1024-65535, Ziel 0.0.0.0/0) zulässt. Erstellen Sie außerdem eine passende Regel für eingehende Nachrichten (Portbereich 1024-65535, Quelle 0.0.0.0/0).

  Informationen zum Erstellen einer VPC finden Sie unter Erstellen einer VPC plus andere VPC-Ressourcen im Amazon VPC-Benutzerhandbuch.
- Nehmen Sie Folgendes in Ihre VPC auf:
  - Internet-Gateway — Sie müssen der VPC, die über Ihre VPC-Ursprungsressourcen verfügt, ein Internet-Gateway hinzufügen. Das Internet-Gateway muss angeben, dass die VPC Datenverkehr aus dem Internet empfangen kann. Das Internet-Gateway wird nicht für die Weiterleitung von Datenverkehr zu Ursprüngen innerhalb des Subnetzes verwendet, und Sie müssen die Routing-Richtlinien nicht aktualisieren.

- Privates Subnetz mit mindestens einer verfügbaren IPv4 Adresse — CloudFront leitet zu Ihrem Subnetz über ein vom Service verwaltetes elastic network interface (ENI) weiter, das CloudFront erstellt wird, nachdem Sie Ihre VPC-Ursprungsressource mit definiert haben. CloudFront Sie müssen mindestens eine verfügbare IPv4 Adresse in Ihrem privaten Subnetz haben, damit der ENI-Erstellungsprozess erfolgreich sein kann. Die IPv4 Adresse kann privat sein und es fallen keine zusätzlichen Kosten an.

  > **Note**
  >
  > IPv6-Nur Subnetze werden nicht unterstützt.

- Starten Sie im privaten Subnetz einen Application Load Balancer, einen Network Load Balancer oder eine EC2 Instance, die Sie als Ursprung verwenden möchten.
  - Die Ressource, die Sie starten, muss vollständig bereitgestellt sein und sich im Status Aktiv befinden, bevor Sie sie für einen VPC-Ursprung verwenden können.
  - Gateway Load Balancer, Dual-Stack Network Load Balancer und Network Load Balancer mit TLS-Listenern können nicht als Ursprünge hinzugefügt werden.
  - Um als VPC-Ursprung verwendet zu werden, muss einem Network Load Balancer eine Sicherheitsgruppe zugeordnet sein.
  - Aktualisieren Sie Ihre Sicherheitsgruppen für die privaten VPC-Ursprünge, um die Liste der CloudFront verwalteten Präfixe ausdrücklich zuzulassen. Weitere Informationen finden Sie unter Verwenden Sie die Liste der CloudFront verwalteten Präfixe.
    - Nachdem der VPC-Ursprung erstellt wurde, kann die Sicherheitsgruppe weiter eingeschränkt werden, sodass nur Datenverkehr von Ihren VPC-Ursprüngen zugelassen wird. Aktualisieren Sie dazu die zulässige Datenverkehrsquelle von der Liste der verwalteten Präfixe auf die CloudFront Sicherheitsgruppe.

    > **Note**
    >
    > WebSockets, gRPC-Verkehr, Origin-Request- und Origin-Response-Trigger mit eingeschaltetem Lambda @Edge CloudFront werden für VPC-Ursprünge nicht unterstützt. Weitere Informationen finden Sie Arbeiten Sie mit Anfragen und Antworten in der Lambda @Edge -Dokumentation.

Erstellen Sie einen VPC-Ursprung (neue Distribution)

Das folgende Verfahren zeigt Ihnen, wie Sie einen VPC-Ursprung für Ihre neue CloudFront Distribution in der CloudFront Konsole erstellen. Alternativ können Sie die CreateDistributionAPI-Operationen CreateVpcOriginund mit dem SDK AWS CLI oder einem AWS SDK verwenden.

So erstellen Sie einen VPC-Ursprung für eine neue Distribution CloudFront

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie VPC-Ursprünge, VPC-Ursprung erstellen.
3. Füllen Sie die erforderlichen Felder aus. Wählen Sie für Origin ARN den ARN Ihres Application Load Balancer, Network Load Balancer oder EC2 Ihrer Instance aus. Wenn Sie den ARN nicht sehen, können Sie Ihren spezifischen Ressourcen-ARN kopieren und ihn stattdessen hier einfügen.
4. Wählen Sie Create VPC origin aus.
5. Warten Sie, bis sich Ihr VPC-Ursprungsstatus auf Bereitgestellt ändert. Dies kann bis zu 15 Minuten dauern.
6. Wählen Sie Verteilungen, Verteilung erstellen.
7. Wählen Sie für Origin-Domain Ihre VPC-Ursprungsressource aus der Drop-down-Liste aus.

   Wenn es sich bei VPC VPC-Ursprung um eine EC2 Instance handelt, kopieren Sie den privaten IP-DNS-Namen der Instance und fügen Sie ihn in das Feld Origin-Domain ein.
8. Beenden Sie die Erstellung Ihrer Distribution. Weitere Informationen finden Sie unter Erstellen Sie eine CloudFront Distribution in der Konsole.

Erstellen Sie einen VPC-Ursprung (bestehende Distribution)

Das folgende Verfahren zeigt Ihnen, wie Sie in der CloudFront Konsole einen VPC-Ursprung für Ihre bestehende CloudFront Distribution erstellen, um die kontinuierliche Verfügbarkeit Ihrer Anwendungen sicherzustellen. Alternativ können Sie die Operationen CreateVpcOriginund die UpdateDistributionWithStagingConfigAPI mit dem AWS CLI oder einem AWS SDK verwenden.

Optional können Sie Ihren VPC-Ursprung zu Ihrer bestehenden Distribution hinzufügen, ohne eine Staging-Verteilung zu erstellen.

Um einen VPC-Ursprung für Ihre bestehende CloudFront Distribution zu erstellen

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie VPC-Ursprünge, VPC-Ursprung erstellen.
3. Füllen Sie die erforderlichen Felder aus. Wählen Sie für Origin ARN den ARN Ihres Application Load Balancer, Network Load Balancer oder EC2 Ihrer Instance aus. Wenn Sie den ARN nicht sehen, können Sie Ihren spezifischen Ressourcen-ARN kopieren und ihn stattdessen hier einfügen.
4. Wählen Sie Create VPC origin aus.
5. Warten Sie, bis sich Ihr VPC-Ursprungsstatus auf Bereitgestellt ändert. Dies kann bis zu 15 Minuten dauern.
6. Rufen Sie im Navigationsbereich Distributions auf.
7. Wählen Sie die ID Ihrer Distribution.
8. Wählen Sie auf der Registerkarte Allgemein unter Kontinuierliche Bereitstellung die Option Staging-Verteilung erstellen aus. Weitere Informationen finden Sie unter Verwenden Sie CloudFront Continuous Deployment, um CDN-Konfigurationsänderungen sicher zu testen.
9. Folgen Sie den Schritten im Assistenten zum Erstellen einer Staging-Verteilung, um eine Staging-Verteilung zu erstellen. Schließen Sie die folgenden Schritte ein:

   - Wählen Sie für Origins die Option Create origin aus.
   - Wählen Sie für Origin-Domain Ihre VPC-Ursprungsressource aus dem Drop-down-Menü aus.

     Wenn es sich bei VPC VPC-Ursprung um eine EC2 Instance handelt, kopieren Sie den privaten IP-DNS-Namen der Instance und fügen Sie ihn in das Feld Origin-Domain ein.
   - Wählen Sie Create Origin (Ursprung erstellen) aus.
10. Testen Sie in Ihrer Staging-Distribution den VPC-Ursprung.
11. Machen Sie die Konfiguration der Staging-Verteilung zu Ihrer primären Distribution. Weitere Informationen finden Sie unter Werben Sie für eine Konfiguration der Staging-Verteilung.
12. Entfernen Sie den öffentlichen Zugriff auf Ihren VPC-Ursprung, indem Sie das Subnetz privat machen. Danach ist der VPC-Ursprung nicht mehr über das Internet auffindbar, hat aber CloudFront weiterhin privaten Zugriff darauf. Weitere Informationen finden Sie unter Zuordnen oder Trennen eines Subnetzes zu einer Routing-Tabelle im Amazon VPC-Benutzerhandbuch.

Einen VPC-Ursprung aktualisieren

Das folgende Verfahren zeigt Ihnen, wie Sie einen VPC-Ursprung für Ihre CloudFront Distribution in der CloudFront Konsole aktualisieren. Alternativ können Sie die UpdateVpcOriginAPI-Operationen UpdateDistributionund mit dem SDK AWS CLI oder einem AWS SDK verwenden.

Um einen vorhandenen VPC-Ursprung für Ihre CloudFront Distribution zu aktualisieren

1. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Rufen Sie im Navigationsbereich Distributions auf.
3. Wählen Sie die ID Ihrer Distribution.
4. Wählen Sie die Registerkarte Behaviors aus.
5. Stellen Sie sicher, dass der VPC-Ursprung nicht der Standardursprung für Ihr Cache-Verhalten ist.
6. Wählen Sie den Tab Ursprünge aus.
7. Wählen Sie den VPC-Ursprung aus, den Sie aktualisieren möchten, und wählen Sie Löschen. Dadurch wird der VPC-Ursprung von Ihrer Distribution getrennt. Wiederholen Sie die Schritte 2-7, um den VPC-Ursprung von allen anderen Distributionen zu trennen.
8. Wählen Sie VPC-Ursprünge.
9. Wählen Sie den VPC-Ursprung aus und klicken Sie auf Bearbeiten.
10. Nehmen Sie Ihre Aktualisierungen vor und wählen Sie VPC-Ursprung aktualisieren.
11. Warten Sie, bis sich Ihr VPC-Ursprungsstatus auf Bereitgestellt ändert. Dies kann bis zu 15 Minuten dauern.
12. Rufen Sie im Navigationsbereich Distributions auf.
13. Wählen Sie die ID Ihrer Distribution.
14. Wählen Sie den Tab Ursprünge aus.
15. Wählen Sie Create Origin (Ursprung erstellen) aus.
16. Wählen Sie für Origin-Domain Ihre VPC-Ursprungsressource aus dem Drop-down-Menü aus.

    Wenn es sich bei VPC VPC-Ursprung um eine EC2 Instance handelt, kopieren Sie den privaten IP-DNS-Namen der Instance und fügen Sie ihn in das Feld Origin-Domain ein.
17. Wählen Sie Create Origin (Ursprung erstellen) aus. Dadurch wird der VPC-Ursprung wieder mit Ihrer Distribution verknüpft. Wiederholen Sie die Schritte 12-17, um den aktualisierten VPC-Ursprung allen anderen Distributionen zuzuordnen.

Wird AWS-Regionen für VPC-Ursprünge unterstützt

VPC-Ursprünge werden derzeit in der folgenden Werbung AWS-Regionen unterstützt. Auf Ausnahmen in der Availability Zone (AZ) wird hingewiesen.

Name der Region	Region
USA Ost (Ohio)	`us-east-2`
USA Ost (Nord-Virginia)	`us-east-1 (except AZ use1-az3)`
USA West (Nordkalifornien)	`us-west-1 (except AZ usw1-az2)`
USA West (Oregon)	`us-west-2`
Africa (Cape Town)	`af-south-1`
Asia Pacific (Hong Kong)	`ap-east-1`
Asien-Pazifik (Mumbai)	`ap-south-1`
Asien-Pazifik (Hyderabad)	`ap-south-2`
Asien-Pazifik (Jakarta)	`ap-southeast-3`
Asien-Pazifik (Melbourne)	`ap-southeast-4`
Asien-Pazifik (Osaka)	`ap-northeast-3`
Asien-Pazifik (Singapur)	`ap-southeast-1`
Asien-Pazifik (Sydney)	`ap-southeast-2`
Asien-Pazifik (Tokio)	`ap-northeast-1 (except AZ apne1-az3)`
Asien-Pazifik (Seoul)	`ap-northeast-2 (except AZ apne2-az1)`
Kanada (Zentral)	`ca-central-1 (except AZ cac1-az3)`

Name der Region	Region
Kanada West (Calgary)	`ca-west-1`
Europe (Frankfurt)	`eu-central-1`
Europa (Irland)	`eu-west-1`
Europa (London)	`eu-west-2`
Europa (Milan)	`eu-south-1`
Europa (Paris)	`eu-west-3`
Europa (Spain)	`eu-south-2`
Europa (Stockholm)	`eu-north-1`
Europa (Zürich)	`eu-central-2`
Israel (Tel Aviv)	`il-central-1`
Naher Osten (Bahrain)	`me-south-1`
Naher Osten (VAE)	`me-central-1`
Südamerika (São Paulo)	`sa-east-1`

Beschränken Sie den Zugriff auf Application Load Balancers

Sie können sowohl interne als auch mit dem Internet verbundene Application Load Balancer mit Amazon verwenden. CloudFront Sie können interne Application Load Balancer in privaten Subnetzen verwenden, CloudFront indem Sie VPC-Ursprünge verwenden. CloudFront VPC-Ursprünge ermöglichen es Ihnen, Inhalte von Anwendungen bereitzustellen, die in privaten VPC-Subnetzen gehostet werden, ohne sie dem öffentlichen Internet zugänglich zu machen. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff mit VPC-Ursprüngen.

Wenn Sie einen mit dem Internet verbundenen Application Load Balancer mit verwenden CloudFront, können Sie die folgenden Sicherheitsmaßnahmen verwenden, um zu verhindern, dass Benutzer direkt auf einen Application Load Balancer zugreifen, und den Zugriff nur über zulassen. CloudFront

1. Konfigurieren CloudFront Sie so, dass ein benutzerdefinierter HTTP-Header zu Anfragen hinzugefügt wird, die an den Application Load Balancer gesendet werden.
2. Konfigurieren Sie den Application Load Balancer so, dass nur Anforderungen weitergeleitet werden, die den benutzerdefinierten HTTP-Header enthalten.
3. Erfordern Sie HTTPS, um die Sicherheit dieser Lösung zu verbessern.

CloudFront kann auch dazu beitragen, die Latenz zu reduzieren und sogar einige Distributed-Denial-of-Service (DDoS) -Angriffe zu absorbieren.

Wenn Ihr Anwendungsfall einen doppelten Zugriff auf Webanwendungen CloudFront sowohl von beiden als auch von Application Load Balancer direkt über das Internet erfordert, sollten Sie Ihre Webanwendung APIs wie folgt aufteilen:

- APIs das muss durchgehen. CloudFront In diesem Fall sollten Sie erwägen, einen separaten privaten Application Load Balancer als Ursprung zu verwenden.
- APIs die Zugriff über Application Load Balancer erfordern. In diesem Fall umgehen CloudFront Sie.

Alternativ können für eine Webanwendung oder andere Inhalte, die von einem mit dem Internet verbundenen Application Load Balancer in Elastic Load Balancing bereitgestellt CloudFront werden, Objekte zwischengespeichert und direkt Benutzern (Viewern) bereitgestellt werden, wodurch die Belastung Ihres Application Load Balancer reduziert wird. Ein mit dem Internet verbundener Load Balancer hat einen öffentlich auflösbaren DNS-Namen und leitet Anfragen von Clients über das Internet an Ziele weiter.

Weitere Informationen finden Sie unter den folgenden Themen. Nachdem Sie diese Schritte abgeschlossen haben, können Benutzer nur über auf Ihren Application Load Balancer zugreifen CloudFront.

Themen

Konfigurieren Sie so CloudFront , dass Anfragen ein benutzerdefinierter HTTP-Header hinzugefügt wird

Sie können so konfigurieren CloudFront , dass den Anfragen, die es an Ihren Ursprung sendet (in diesem Fall ein Application Load Balancer), ein benutzerdefinierter HTTP-Header hinzugefügt wird.

> **Important**
>
> Dieser Anwendungsfall beruht darauf, den Namen und den Wert des benutzerdefinierten Headers geheim zu halten. Wenn der Header-Name und der Wert nicht geheim sind, könnten andere HTTP-Clients sie möglicherweise in Anfragen aufnehmen, die sie direkt an den Application Load Balancer senden. Dies kann dazu führen, dass sich der Application Load Balancer so verhält, als kämen die Anfragen von CloudFront einem Ort, an dem dies nicht der Fall war. Um dies zu verhindern, halten Sie den Namen und den Wert des benutzerdefinierten Headers geheim.

Sie können mit der CloudFront Konsole oder der CloudFront API so konfigurieren CloudFront , dass zu ursprünglichen Anfragen ein benutzerdefinierter HTTP-Header hinzugefügt wird. AWS CloudFormation

Um einen benutzerdefinierten HTTP-Header (CloudFront Konsole) hinzuzufügen

Verwende in der CloudFront Konsole die Einstellung Benutzerdefinierte Origin-Header in den Origin-Einstellungen. Geben Sie den Header-Namen und seinen Wert ein.

> **Note**
>
> Verwenden Sie in der Produktion zufällig generierte Header-Namen und -Werte. Behandeln Sie Header-Namen und -Werte wie Benutzernamen und Passwörter als sichere Anmeldeinformationen.

Sie können die Einstellung Benutzerdefinierte Origin-Header bearbeiten, wenn Sie einen Ursprung für eine bestehende CloudFront Distribution erstellen oder bearbeiten und wenn Sie eine neue Distribution erstellen. Weitere Informationen erhalten Sie unter Eine Verteilung aktualisieren und Eine Verteilung erstellen.

So fügen Sie einen benutzerdefinierten HTTP-Header hinzu (AWS CloudFormation)

Verwenden Sie in einer AWS CloudFormation Vorlage die `OriginCustomHeaders` Eigenschaft, wie im folgenden Beispiel gezeigt.

> **Note**
>
> Der Header-Name und der Wert in diesem Beispiel dienen nur zur Demonstration. Verwenden Sie in der Produktion zufällig generierte Werte. Behandeln Sie den Namen und den Wert des Headers als sichere Berechtigung, ähnlich einem Benutzernamen und einem Passwort.

```
AWSTemplateFormatVersion: '2010-09-09'
Resources:
  TestDistribution:
    Type: 'AWS::CloudFront::Distribution'
    Properties:
      DistributionConfig:
        Origins:
          - DomainName: app-load-balancer.example.com
            Id: Example-ALB
            CustomOriginConfig:
              OriginProtocolPolicy: https-only
              OriginSSLProtocols:
                - TLSv1.2
            OriginCustomHeaders:
               - HeaderName: X-Custom-Header
                 HeaderValue: random-value-1234567890
        Enabled: 'true'
        DefaultCacheBehavior:
          TargetOriginId: Example-ALB
          ViewerProtocolPolicy: allow-all
          CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6
        PriceClass: PriceClass_All
        ViewerCertificate:
          CloudFrontDefaultCertificate: 'true'
```

Weitere Informationen finden Sie unter Origin und OriginCustomHeaderEigenschaften im AWS CloudFormation Benutzerhandbuch.

Um einen benutzerdefinierten HTTP-Header (CloudFront API) hinzuzufügen

Verwenden Sie in der CloudFront API das `CustomHeaders` Objekt darin`Origin`. Weitere Informationen finden Sie unter CreateDistributionund UpdateDistributionin der Amazon CloudFront API-Referenz sowie in der Dokumentation für Ihr SDK oder einen anderen API-Client.

Es gibt einige Header-Namen, die Sie nicht als benutzerdefinierte Ursprungs-Header angeben können. Weitere Informationen finden Sie unter Benutzerdefinierte Header, die nicht zu CloudFront ursprünglichen Anfragen hinzugefügt werden können.

Konfigurieren Sie einen Application Load Balancer so, dass er nur Anfragen weiterleitet, die einen bestimmten Header enthalten

Nachdem Sie konfiguriert haben CloudFront , dass den Anfragen, die er an Ihren Application Load Balancer sendet, ein benutzerdefinierter HTTP-Header hinzugefügt wird (siehe vorheriger Abschnitt), können Sie den Load Balancer so konfigurieren, dass er nur Anfragen weiterleitet, die diesen benutzerdefinierten Header enthalten. Dazu fügen Sie eine neue Regel hinzu und ändern die Standardregel im Listener Ihres Load Balancers.

Voraussetzungen

Um die folgenden Verfahren verwenden zu können, benötigen Sie einen Application Load Balancer mit mindestens einem Listener. Wenn Sie noch keinen erstellt haben, lesen Sie Create an Application Load Balancer im Benutzerhandbuch für Application Load Balancer.

Die folgenden Verfahren ändern einen HTTPS-Listener. Sie können den gleichen Prozess verwenden, um einen HTTP-Listener zu ändern.

So aktualisieren Sie die Regeln in einem Application Load Balancer-Listener

1. Fügen Sie eine neue Regel hinzu. Folgen Sie den Anweisungen unter Regel hinzufügen mit den folgenden Änderungen:

   - Fügen Sie die Regel dem Load Balancer hinzu, der der Ursprung Ihrer CloudFront Distribution ist.
   - Wählen Sie für Bedingung hinzufügen die Option Http-Header aus. Geben Sie den Namen und den Wert des HTTP-Headers an, den Sie als benutzerdefinierten Origin-Header hinzugefügt haben CloudFront.

- Wählen Sie für Aktion hinzufügen die Option Weiterleiten an aus. Wählen Sie die Zielgruppe aus, in die Sie Anfragen weiterleiten möchten.

2. Bearbeiten Sie die Standardregel im Listener Ihres Load Balancers. Folgen Sie den Anweisungen unter Regel bearbeiten mit den folgenden Änderungen:

   - Bearbeiten Sie die Standardregel des Load Balancers, der der Ursprung Ihrer CloudFront Distribution ist.
   - Löschen Sie die Standardaktion, und wählen Sie dann für Aktion hinzufügen die Option Feste Antwort zurückgeben aus.
   - Geben Sie für den Antwortcode **`403`** ein.
   - Geben Sie für den Antworttext **`Access denied`** ein.

Nachdem Sie diese Schritte abgeschlossen haben, hat Ihr Load Balancer-Listener zwei Regeln. Eine Regel leitet Anfragen weiter, die den HTTP-Header enthalten (Anfragen, die von kommen). CloudFront Die andere Regel sendet eine feste Antwort auf alle anderen Anfragen (Anfragen, die nicht von kommen CloudFront).

Sie können überprüfen, ob die Lösung funktioniert, indem Sie eine Anfrage an Ihre CloudFront Distribution und eine an Ihren Application Load Balancer senden. Die Anfrage zur CloudFront Rückgabe Ihrer Webanwendung oder Ihres Inhalts und die direkt an Ihren Application Load Balancer gesendete Anfrage geben eine `403` Antwort mit der Klartextnachricht `Access denied` zurück.

(Optional) Verbesserung der Sicherheit dieser Lösung

Um die Sicherheit dieser Lösung zu verbessern, können Sie Ihre CloudFront Distribution so konfigurieren, dass beim Senden von Anfragen an Ihren Application Load Balancer immer HTTPS verwendet wird. Denken Sie daran, dass diese Lösung nur funktioniert, wenn Sie den benutzerdefinierten Header-Namen und den Wert geheim halten. Die Verwendung von HTTPS kann dazu beitragen, dass ein Spion den Namen und den Wert des Headers entdeckt. Wir empfehlen außerdem, den Namen und den Wert des Headers regelmäßig zu wechseln.

Verwenden von HTTPS für Origin-Anfragen

Um die Verwendung von HTTPS für ursprüngliche Anfragen CloudFront zu konfigurieren, setzen Sie die Einstellung Origin Protocol Policy auf Nur HTTPS. Diese Einstellung ist in der CloudFront Konsole und in der CloudFront API verfügbar. AWS CloudFormation Weitere Informationen finden Sie unter Protokoll (nur benutzerdefinierte Ursprünge).

Folgendes gilt auch, wenn Sie CloudFront die Verwendung von HTTPS für ursprüngliche Anfragen konfigurieren:

- Sie müssen die Konfiguration so konfigurieren CloudFront , dass der `Host` Header mit der ursprünglichen Anforderungsrichtlinie an den Ursprung weitergeleitet wird. Sie können die Richtlinie für AllViewer verwaltete Ursprungsanfragen verwenden.
- Stellen Sie sicher, dass Ihr Application Load Balancer über einen HTTPS-Listener verfügt (wie im vorherigen Abschnitt gezeigt). Weitere Informationen finden Sie unter Einen HTTPS-Listener erstellen im Benutzerhandbuch für Application Load Balancers. Für die Verwendung eines HTTPS-Listeners benötigen Sie ein SSL/TLS-Zertifikat, das dem Domainnamen entspricht, der an Ihren Application Load Balancer weitergeleitet wird.
- SSL/TLS-Zertifikate für CloudFront können nur im in (ACM) angefordert (oder importiert) werden. `us-east-1` AWS-Region AWS Certificate Manager Da es CloudFront sich um einen globalen Dienst handelt, verteilt er das Zertifikat automatisch von der `us-east-1` Region an alle Regionen, die mit Ihrer Distribution verknüpft sind. CloudFront
  - Wenn Sie beispielsweise einen Application Load Balancer (ALB) in der `ap-southeast-2` Region haben, müssen Sie SSL/TLS-Zertifikate sowohl in der `ap-southeast-2` Region (für die Verwendung von HTTPS zwischen CloudFront und dem ALB-Ursprung) als auch in der `us-east-1` Region (für die Verwendung von HTTPS zwischen Zuschauern und) konfigurieren. CloudFront Beide Zertifikate sollten mit dem Domainnamen übereinstimmen, der an Ihren Application Load Balancer weitergeleitet wird. Weitere Informationen finden Sie unter AWS-Region für AWS Certificate Manager.
- Wenn die Endbenutzer (auch Viewer oder Clients genannt) Ihrer Webanwendung HTTPS verwenden können, können Sie auch so konfigurieren, dass HTTPS-Verbindungen von CloudFront den Endbenutzern bevorzugt (oder sogar erforderlich) werden. Verwenden Sie dazu die Einstellung Betrachter-Protokollrichtlinie. Sie können es so einstellen, dass Endbenutzer von HTTP auf HTTPS umgeleitet oder Anfragen, die HTTP verwenden, abgelehnt werden. Diese Einstellung ist in der CloudFront Konsole und in der CloudFront API verfügbar. AWS CloudFormation Weitere Informationen finden Sie unter Viewer-Protokollrichtlinien.

Wechseln des Header-Namens und des Werts

Zusätzlich zur Verwendung von HTTPS empfehlen wir auch, den Header-Namen und -Wert regelmäßig zu ändern. Befolgen Sie hierfür die folgenden Schritte:

1. Konfigurieren CloudFront Sie so, dass ein zusätzlicher benutzerdefinierter HTTP-Header zu Anfragen hinzugefügt wird, die an den Application Load Balancer gesendet werden.
2. Aktualisieren Sie die Application Load Balancer-Listener-Regel, um Anforderungen weiterzuleiten, die diesen zusätzlichen benutzerdefinierten HTTP-Header enthalten.
3. Konfigurieren CloudFront Sie so, dass der ursprüngliche benutzerdefinierte HTTP-Header nicht mehr zu Anfragen hinzugefügt wird, die er an den Application Load Balancer sendet.
4. Aktualisieren Sie die Application Load Balancer-Listener-Regel, um die Weiterleitung von Anforderungen zu beenden, die den ursprünglichen benutzerdefinierten HTTP-Header enthalten.

Weitere Informationen zum Ausführen dieser Schritte finden Sie in den vorherigen Abschnitten.

(Optional) Beschränken Sie den Zugriff auf den Ursprung, indem Sie die AWS Präfixliste -managed für verwenden CloudFront

Um den Zugriff auf Ihren Application Load Balancer weiter einzuschränken, können Sie die dem Application Load Balancer zugeordnete Sicherheitsgruppe so konfigurieren, dass sie nur Datenverkehr akzeptiert, CloudFront wenn der Dienst eine AWS-verwaltete Präfixliste verwendet. Dadurch wird verhindert, dass Datenverkehr, der nicht CloudFront stammt, Ihren Application Load Balancer auf der Netzwerkschicht (Schicht 3) oder Transportschicht (Schicht 4) erreicht.

Weitere Informationen finden Sie im CloudFront Blogbeitrag Beschränken Sie den Zugriff auf Ihre Ursprünge mithilfe der AWS-verwalteten Präfixliste für Amazon.

Beschränken Sie die geografische Verteilung Ihrer Inhalte

Sie können geografische Beschränkungen verwenden, die manchmal auch als Geoblocking bezeichnet werden, um zu verhindern, dass Benutzer an bestimmten geografischen Standorten auf Inhalte zugreifen, die Sie über eine CloudFront Amazon-Distribution vertreiben. Es gibt zwei Möglichkeiten zum Verwenden der geografischen Einschränkung:

- Verwenden Sie die Funktion für CloudFront geografische Einschränkungen. Verwenden Sie diese Option, um den Zugriff auf alle mit einer Verteilung verknüpften Dateien zu beschränken und um den Zugriff auf der Länderebene einzuschränken.
- Verwenden Sie den Geolokalisierungsdienst eines Drittanbieters. Verwenden Sie diese Option, um den Zugriff auf einen Teil der mit einer Verteilung verknüpften Dateien zu beschränken oder um den Zugriff auf eine feinere Granularität als die Länderebene einzuschränken.

Themen

Verwenden Sie CloudFront geografische Einschränkungen

Wenn ein Benutzer Ihre Inhalte anfordert, werden die angeforderten Inhalte in der CloudFront Regel unabhängig davon bereitgestellt, wo sich der Benutzer befindet. Wenn Sie verhindern möchten, dass Nutzer in bestimmten Ländern auf Ihre Inhalte zugreifen, können Sie die Funktion für CloudFront geografische Einschränkungen verwenden, um eine der folgenden Aktionen durchzuführen:

- Gewähren Sie Ihren Benutzern den Zugriff auf Ihre Inhalte nur, wenn diese sich in einem der zugelassenen Länder auf Ihrer Zulassungsliste befinden.
- Verhindern Sie, dass Benutzer auf Ihre Inhalte zugreifen, wenn sie sich in einem der gesperrten Länder auf Ihrer Deny-Liste befinden.

Wenn eine Anfrage beispielsweise aus einem Land kommt, in dem Sie nicht berechtigt sind, Ihre Inhalte zu verteilen, können Sie die Anfrage mithilfe CloudFront geografischer Beschränkungen blockieren.

> **Note**
>
> CloudFront bestimmt den Standort Ihrer Nutzer mithilfe einer Drittanbieter-Datenbank. Die Genauigkeit der Zuweisung zwischen IP-Adressen und Ländern variiert je nach Region. Gemäß kürzlich erfolgten Tests beträgt die allgemeine Genauigkeit 99,8 %. Wenn der Standort eines Benutzers nicht ermittelt werden CloudFront kann, CloudFront wird der Inhalt bereitgestellt, den der Benutzer angefordert hat.

So funktioniert die geografische Einschränkung:

1. Nehmen wir an, Sie haben nur Rechte zum Verteilen Ihrer Inhalte in Liechtenstein. Sie aktualisieren Ihre CloudFront Distribution, um eine Zulassungsliste hinzuzufügen, die nur Liechtenstein enthält. (Alternativ können Sie eine Deny-Liste mit allen Ländern außer Liechtenstein hinzufügen.)

2. Ein Benutzer in Monaco fordert Ihre Inhalte an, und DNS leitet die Anfrage an einen CloudFront Edge-Standort in Mailand, Italien, weiter.
3. Der Edge-Standort in Mailand sucht nach Ihrer Verteilung und ermittelt, dass der Benutzer in Monaco nicht zum Herunterladen Ihrer Inhalte berechtigt ist.
4. CloudFront gibt dem Benutzer einen HTTP-Statuscode `403 (Forbidden)` zurück.

Sie können optional konfigurieren CloudFront , dass dem Benutzer eine benutzerdefinierte Fehlermeldung zurückgegeben wird, und Sie können angeben, wie lange die Fehlerantwort für die angeforderte Datei zwischengespeichert werden soll CloudFront . Der Standardwert liegt bei 10 Sekunden. Weitere Informationen finden Sie unter Erstellen Sie eine benutzerdefinierte Fehlerseite für bestimmte HTTP-Statuscodes.

Geografische Einschränkungen gelten für eine gesamte Verteilung. Wenn Sie eine Einschränkung auf einen Teil Ihres Inhalts und eine andere Einschränkung (oder keine Beschränkung) auf einen anderen Teil Ihres Inhalts anwenden müssen, müssen Sie separate CloudFront Distributionen erstellen oder einen Geolocation-Dienst eines Drittanbieters verwenden.

Wenn Sie CloudFront Standardprotokolle (Zugriffsprotokolle) aktivieren, können Sie die CloudFront abgelehnten Anfragen identifizieren, indem Sie nach den Protokolleinträgen suchen, in denen der Wert von `sc-status` (der HTTP-Statuscode) steht. `403` Wenn Sie jedoch nur die Standardprotokolle verwenden, können Sie eine Anfrage, die aufgrund des Standorts des Benutzers CloudFront abgelehnt wurde, nicht von einer Anfrage unterscheiden, die CloudFront abgelehnt wurde, weil der Benutzer aus einem anderen Grund nicht berechtigt war, auf die Datei zuzugreifen. Wenn Sie über einen Geolokalisierungsdienst eines Drittanbieters wie Digital Element oder verfügen MaxMind, können Sie den Standort von Anfragen anhand der IP-Adresse in der Spalte `c-ip` (Client-IP) in den Zugriffsprotokollen ermitteln. Weitere Informationen zu CloudFront Standardprotokollen finden Sie unter. Standardprotokollierung (Zugriffsprotokolle)

Im folgenden Verfahren wird erklärt, wie Sie mithilfe der CloudFront Konsole geografische Einschränkungen zu einer vorhandenen Distribution hinzufügen können. Informationen zur Verwendung der Konsole zum Erstellen von Verteilungen finden Sie unter Eine Verteilung erstellen.

So fügen Sie Ihrer CloudFront Webdistribution (Konsole) geografische Einschränkungen hinzu

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich die Option Distributionen und dann die Option, die Sie aktualisieren möchten.

3. Wählen Sie die Registerkarte Sicherheit und anschließend Geografische Einschränkungen aus.
4. Wählen Sie Edit (Bearbeiten) aus.
5. Wählen Sie Allow list (Genehmigungsliste) aus, um eine Liste der zulässigen Länder zu erstellen, oder Block list (Blockliste), um eine Liste der gesperrten Länder zu erstellen.
6. Fügen Sie die gewünschten Länder zur Liste hinzu und wählen Sie dann Save changes (Änderungen speichern) aus.

Verwenden Sie einen Geolocation-Dienst eines Drittanbieters

Mit der Funktion für CloudFront geografische Beschränkungen steuern Sie die Verteilung Ihrer Inhalte auf Landesebene für alle Dateien, die Sie mit einer bestimmten Webdistribution verteilen. Wenn Sie einen Anwendungsfall für geografische Einschränkungen haben, bei dem die Beschränkungen nicht an Landesgrenzen gebunden sind, oder wenn Sie den Zugriff nur auf einige der Dateien beschränken möchten, die Sie über eine bestimmte Distribution bereitstellen, können Sie die Verbindung CloudFront mit einem Geolokalisierungsdienst eines Drittanbieters kombinieren. Dies ermöglicht Ihnen die Kontrolle des Zugriffs auf Ihre Inhalte nicht nur auf der Grundlage des Landes, sondern auch auf der Grundlage der Stadt, der Postleitzahl oder sogar der Längen- und Breitengrade.

Wenn Sie einen Geolokalisierungsdienst eines Drittanbieters verwenden, empfehlen wir IhnenURLs, CloudFront signiert zu verwenden. Damit können Sie ein Ablaufdatum und eine Uhrzeit angeben, nach deren Ablauf die URL nicht mehr gültig ist. Darüber hinaus empfehlen wir Ihnen, einen Amazon S3 S3-Bucket als Quelle zu verwenden, da Sie dann eine CloudFront Origin-Zugriffskontrolle verwenden können, um zu verhindern, dass Benutzer direkt vom Ursprung aus auf Ihre Inhalte zugreifen. Weitere Informationen zur signierten Zugriffskontrolle URLs und zur Herkunftszugriffskontrolle finden Sie unterStellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit.

In den folgenden Schritten wird erläutert, wie Sie den Zugriff auf Ihre Dateien mithilfe eines Geolokalisierungsservices eines Drittanbieters kontrollieren können.

So verwenden Sie einen Geolocation-Dienst eines Drittanbieters, um den Zugriff auf Dateien in einer Distribution einzuschränken CloudFront

1. Legen Sie ein Konto bei einem Geolokalisierungsdienst an.
2. Laden Sie Ihre Inhalte in einen Amazon-S3-Bucket hoch.

3. Konfigurieren Sie Amazon CloudFront und Amazon S3 für die Bereitstellung privater Inhalte. Weitere Informationen finden Sie unter Stellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit.
4. Schreiben Sie Ihre Webanwendung so, dass folgende Aufgaben ausgeführt werden:
   - Senden der IP-Adresse für jede Benutzeranfrage an den Geolokalisierungsdienst.
   - Bewerten Sie den Rückgabewert des Geolocation-Service, um festzustellen, ob sich der Benutzer an einem Ort befindet, an dem Sie Ihre CloudFront Inhalte verteilen möchten.
   - Wenn Sie Ihre Inhalte an den Standort des Benutzers verteilen möchten, generieren Sie eine signierte URL für Ihre CloudFront Inhalte. Wenn Sie keinen Inhalt an diesen Speicherort verteilen möchten, geben Sie dem Benutzer den HTTP-Statuscode `403 (Forbidden)` zurück. Alternativ können Sie so konfigurieren CloudFront , dass eine benutzerdefinierte Fehlermeldung zurückgegeben wird. Weitere Informationen finden Sie unter the section called “Erstellen Sie eine benutzerdefinierte Fehlerseite für bestimmte HTTP-Statuscodes”.

   Weitere Informationen finden Sie in der Dokumentation für den Geolokationsdienst, den Sie verwenden.

Sie können die IP-Adressen der Besucher Ihrer Website mithilfe einer Webserver-Variablen abrufen. Beachten Sie folgende Einschränkungen:

- Wenn Ihr Webserver nicht über einen Load Balancer mit dem Internet verbunden ist, können Sie eine Webserver-Variable verwenden, um die IP-Remote-Adresse abzurufen. Diese IP-Adresse ist jedoch nicht immer die IP-Adresse des Benutzers. Es kann sich dabei auch um die IP-Adresse eines Proxy-Servers handeln, je nachdem, wie der Benutzer mit dem Internet verbunden ist.
- Wenn Ihr Webserver über einen Load Balancer mit dem Internet verbunden ist, enthält eine Webserver-Variable möglicherweise die IP-Adresse des Load Balancers, nicht die IP-Adresse des Benutzers. Bei dieser Konfiguration empfehlen wir, die letzte IP-Adresse im `X-Forwarded-For`-HTTP-Header zu verwenden. Dieser Header enthält in der Regel mehr als eine IP-Adresse, von denen die meisten für Proxys oder Load Balancer gelten. Die letzte IP-Adresse in der Liste wird am wahrscheinlichsten mit dem geografischen Standort des Benutzers verknüpft.

Wenn Ihr Webserver nicht mit einem Load Balancer verbunden ist, empfehlen wir, anstelle des `X-Forwarded-For`-Headers Webserver-Variablen zu verwenden, um Spoofing von IP-Adressen zu vermeiden.

Vertrauliche Daten durch Verschlüsselung auf Feldebene schützen

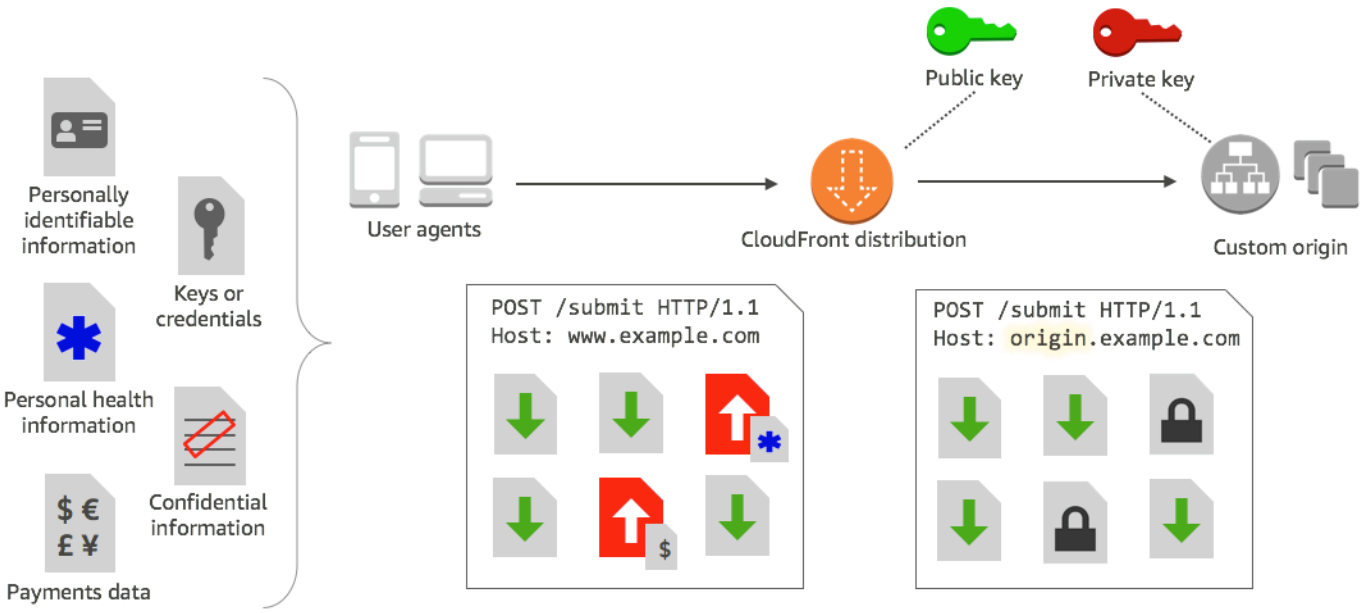
Mit Amazon CloudFront können Sie sichere end-to-end Verbindungen zu Originalservern mithilfe von HTTPS erzwingen. Die Verschlüsselung auf Feldebene fügt zusammen mit HTTPS eine zusätzliche Sicherheitsebene hinzu, mit der Sie bestimmte Daten während der gesamten Systemverarbeitung so schützen können, dass nur bestimmte Anwendungen sie sehen können.

Die Verschlüsselung auf Feldebene ermöglicht es Ihren Benutzern, vertrauliche Informationen in sicherer Weise auf Ihre Webserver hochzuladen. Die vertraulichen Informationen, die von Ihren Benutzern bereitgestellt werden, werden am Rand, in der Nähe des Benutzers, verschlüsselt und bleiben über den gesamten Anwendungs-Stack hinweg verschlüsselt. Diese Verschlüsselung stellt sicher, dass nur Anwendungen, die die Daten benötigen - und über die Anmeldeinformationen zum Entschlüsseln verfügen - dies tun können.

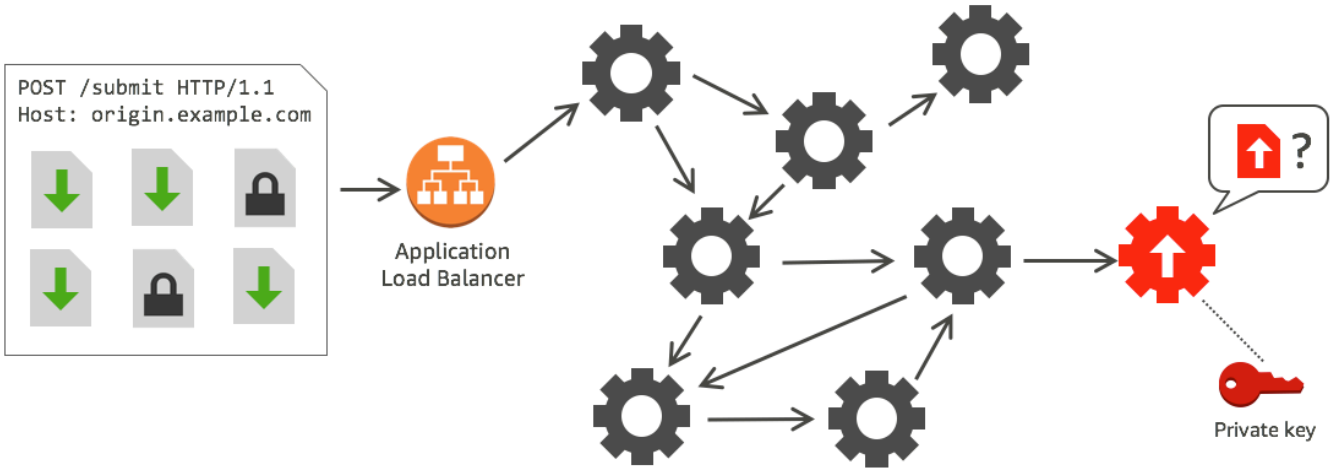
Um die Verschlüsselung auf Feldebene zu verwenden, geben Sie bei der Konfiguration Ihrer CloudFront Distribution die Felder in POST-Anfragen an, die Sie verschlüsseln möchten, sowie den öffentlichen Schlüssel, mit dem sie verschlüsselt werden sollen. Sie können bis zu 10 Datenfelder in einer Anfrage verschlüsseln. (Sie können nicht alle Daten in einer Anfrage mit Verschlüsselung auf Feldebene verschlüsseln; Sie müssen einzelne Felder angeben, die verschlüsselt werden sollen.)

Wenn die HTTPS-Anfrage mit Verschlüsselung auf Feldebene an den Ursprung weitergeleitet und durch Ihre Ursprungsanwendung bzw. Ihr Ursprungssubsystem geleitet wird, sind die vertraulichen Daten immer noch verschlüsselt, wodurch das Risiko einer Datenschutzverletzung oder eines versehentlichen Datenverlustes der vertraulichen Daten verringert wird. Komponenten, die aus geschäftlichen Gründen Zugriff auf die sensiblen Daten benötigen, wie z. B. ein Zahlungssystem für eine Kreditnummer, können mit dem entsprechenden privaten Schlüssel entschlüsseln und auf die Daten zugreifen.

> **Note**
>
> Beachten Sie, dass der Ursprung für die Verschlüsselung auf Feldebene die Chunked-Codierung unterstützen muss.



CloudFront Bei der Verschlüsselung auf Feldebene wird eine asymmetrische Verschlüsselung verwendet, die auch als Verschlüsselung mit öffentlichen Schlüsseln bezeichnet wird. Sie geben einen öffentlichen Schlüssel an CloudFront, und alle vertraulichen Daten, die Sie angeben, werden automatisch verschlüsselt. Der Schlüssel, den Sie angeben, CloudFront kann nicht zum Entschlüsseln der verschlüsselten Werte verwendet werden. Das kann nur mit Ihrem privaten Schlüssel geschehen.



Themen

- Überblick über die Verschlüsselung auf Feldebene

- Richten Sie eine Verschlüsselung auf Feldebene ein
- Entschlüsseln Sie Datenfelder an Ihrem Ursprung

Überblick über die Verschlüsselung auf Feldebene

Die folgenden Schritte geben einen Überblick über die Einrichtung der Verschlüsselung auf Feldebene. Spezifische Schritte finden Sie unter Richten Sie eine Verschlüsselung auf Feldebene ein.

1. Holen Sie sich ein Schlüsselpaar aus öffentlichem und privatem Schlüssel. Sie müssen den öffentlichen Schlüssel abrufen und hinzufügen, bevor Sie mit der Einrichtung der Verschlüsselung auf Feldebene in CloudFront beginnen.
2. Erstellen Sie ein Verschlüsselungsprofil auf Feldebene. Verschlüsselungsprofile auf Feldebene, in denen Sie erstellen CloudFront, definieren die Felder, die verschlüsselt werden sollen.
3. Erstellen Sie eine Verschlüsselungskonfiguration auf Feldebene. Eine Konfiguration spezifiziert die zu verwendenden Profile, basierend auf dem Inhaltstyp der Anfrage oder einem Abfrageargument, um bestimmte Datenfelder zu verschlüsseln. Sie können auch die Verhaltensoptionen für die Anforderungsweiterleitung auswählen, die Sie für verschiedene Szenarien benötigen. Sie können beispielsweise das Verhalten festlegen, wenn der durch das Abfrageargument in einer Anforderungs-URL angegebene Profilname in nicht existiert. CloudFront
4. Verknüpfen zu einem Cache-Verhalten. Verknüpfen Sie die Konfiguration mit einem Cache-Verhalten für eine Verteilung, um anzugeben, wann CloudFront Daten verschlüsseln soll.

Richten Sie eine Verschlüsselung auf Feldebene ein

Führen Sie die folgenden Schritte aus, um mit der Verschlüsselung auf Feldebene zu beginnen. Weitere Informationen zu Kontingenten (früher als Limits bezeichnet) für die Verschlüsselung auf Feldebene finden Sie unter Kontingente.

- Schritt 1: Erstellen eines RSA-Schlüsselpaars
- Schritt 2: Fügen Sie Ihren öffentlichen Schlüssel hinzu CloudFront
- Schritt 3: Erstellen eines Profils für die Verschlüsselung auf Feldebene
- Schritt 4: Erstellen einer Konfiguration
- Schritt 5: Hinzufügen einer Konfiguration zu einem Cache-Verhalten

Schritt 1: Erstellen eines RSA-Schlüsselpaars

Um zu beginnen, müssen Sie ein RSA-Schlüsselpaar erstellen, das einen öffentlichen Schlüssel und einen privaten Schlüssel enthält. Der öffentliche Schlüssel ermöglicht CloudFront die Verschlüsselung von Daten, und der private Schlüssel ermöglicht es Komponenten an Ihrem Ursprung, die verschlüsselten Felder zu entschlüsseln. Sie können OpenSSL oder ein anderes Tool verwenden, um ein Schlüsselpaar zu erstellen. Die Schlüsselgröße muss 2048 Bit betragen.

Wenn Sie beispielsweise OpenSSL verwenden, können Sie mithilfe des folgenden Befehls ein Schlüsselpaar mit einer Länge von 2048 Bits erstellen und in der Datei `private_key.pem` speichern:

```
openssl genrsa -out private_key.pem 2048
```

Die erstellte Datei enthält den öffentlichen und den privaten Schlüssel. Um den öffentlichen Schlüssel aus dieser Datei zu extrahieren, führen Sie den folgenden Befehl aus:

```
openssl rsa -pubout -in private_key.pem -out public_key.pem
```

Die Datei mit dem öffentlichen Schlüssel (`public_key.pem`) enthält den codierten Schlüsselwert, den Sie im folgenden Schritt einfügen.

Schritt 2: Fügen Sie Ihren öffentlichen Schlüssel hinzu CloudFront

Nachdem Sie Ihr RSA-Schlüsselpaar erhalten haben, fügen Sie Ihren öffentlichen Schlüssel zu CloudFront hinzu.

Um Ihren öffentlichen Schlüssel zu CloudFront (Konsole) hinzuzufügen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Public key aus.
3. Wählen Sie Add public key (Öffentlichen Schlüssel hinzufügen) aus.
4. Geben Sie unter Key name einen eindeutigen Namen für den Schlüssel ein. Der Name darf keine Leerzeichen enthalten und darf nur alphanumerische Zeichen, Unterstriche (_) und Bindestriche (-) enthalten. Die maximale Anzahl der Zeichen beträgt 128.

5. Fügen Sie unter Key value (Schlüsselwert) den codierten Schlüsselwert für den öffentlichen Schlüssel ein, einschließlich der Zeilen `-----BEGIN PUBLIC KEY-----` und `-----END PUBLIC KEY-----`.
6. In Comment fügen Sie einen optionalen Kommentar hinzu. Beispielsweise können Sie das Ablaufdatum für den öffentlichen Schlüssel angeben.
7. Wählen Sie Hinzufügen aus.

Sie können weitere Schlüssel zur Verwendung hinzufügen, CloudFront indem Sie die Schritte des Verfahrens wiederholen.

Schritt 3: Erstellen eines Profils für die Verschlüsselung auf Feldebene

Nachdem Sie mindestens einen öffentlichen Schlüssel hinzugefügt haben CloudFront, erstellen Sie ein Profil, das angibt, CloudFront welche Felder verschlüsselt werden sollen.

So erstellen Sie ein Profil für die Verschlüsselung auf Feldebene (Konsole)

1. Wählen Sie im Navigationsbereich die Option Field-level encryption aus.
2. Wählen Sie Create profile (Profil erstellen) aus.
3. Füllen Sie die folgenden Felder aus:

   Profilname

   Geben Sie einen eindeutigen Namen für das Profil ein. Der Name darf keine Leerzeichen enthalten und darf nur alphanumerische Zeichen, Unterstriche (_) und Bindestriche (-) enthalten. Die maximale Anzahl der Zeichen beträgt 128.

   Public key name

   Wählen Sie in der Dropdownliste den Namen eines öffentlichen Schlüssels aus, den Sie CloudFront in Schritt 2 hinzugefügt haben. CloudFront verwendet den Schlüssel, um die Felder zu verschlüsseln, die Sie in diesem Profil angeben.

   Provider name

   Geben Sie einen Satz ein, um den Schlüssel zu identifizieren (z. B. den Anbieter, bei dem Sie das Schlüsselpaar erhalten haben). Diese Informationen werden zusammen mit dem privaten Schlüssel benötigt, wenn Anwendungen Datenfelder entschlüsseln. Der Name des Anbieters darf keine Leerzeichen enthalten und darf nur alphanumerische Zeichen, Doppelpunkte (:), Unterstriche (_) und Bindestriche (-) enthalten. Die maximale Anzahl der Zeichen beträgt 128.

Feldnamensmuster für die Zuordnung

Geben Sie die Namen der Datenfelder oder Muster ein, die Datenfeldnamen in der Anfrage identifizieren, die Sie mit CloudFront verschlüsseln möchten. Wählen Sie die Option +, um alle Felder hinzuzufügen, die Sie mit diesem Schlüssel verschlüsseln möchten.

Für das Feldnamenmuster können Sie den gesamten Namen des Datenfeldes eingeben DateOfBirth, z. B. oder nur den ersten Teil des Namens mit einem Platzhalterzeichen (*), z. B. CreditCard *. Das Feldnamensmuster darf neben dem optionalen Platzhalterzeichen (*) nur aus alphanumerischen Zeichen, eckigen Klammern ([und]), Punkten (.), Unterstrichen (_) und Bindestrichen (-) bestehen.

Stellen Sie sicher, dass Sie keine überschneidenden Zeichen für unterschiedliche Feldnamensmuster verwenden. Wenn Sie beispielsweise ein Feldnamensmuster von ABC* haben, können Sie kein weiteres Feldnamensmuster hinzufügen, das AB* ist. Dazu kommt, dass bei Feldnamen zwischen Groß- und Kleinschreibung unterschieden wird und die maximale Anzahl der Zeichen 128 beträgt.

Kommentar

(Optional) Geben Sie einen Kommentar zu diesem Profil ein. Die maximale Anzahl der Zeichen beträgt 128.

4. Nachdem Sie die Felder ausgefüllt haben, wählen Sie Create profile (Profil erstellen) aus.
5. Wenn Sie weitere Profile hinzufügen möchten, wählen Sie Add profile aus.

Schritt 4: Erstellen einer Konfiguration

Nachdem Sie ein oder mehrere Verschlüsselungsprofile auf Feldebene erstellt haben, erstellen Sie eine Konfiguration, die den Inhaltstyp der Anforderung angibt, die die zu verschlüsselnden Daten enthält, das für die Verschlüsselung zu verwendende Profil und andere Optionen, die angeben, wie Sie mit der Verschlüsselung umgehen CloudFront möchten.

Wenn die Daten beispielsweise nicht verschlüsselt CloudFront werden können, können Sie in den folgenden Szenarien angeben, ob eine Anfrage blockiert oder an Ihren Ursprung weitergeleitet werden CloudFront soll:

- Wenn der Inhaltstyp einer Anfrage nicht in einer Konfiguration enthalten ist — Wenn Sie einer Konfiguration keinen Inhaltstyp hinzugefügt haben, können Sie angeben, ob die Anfrage mit

diesem Inhaltstyp an den Ursprung weitergeleitet werden CloudFront soll, ohne Datenfelder zu verschlüsseln, oder ob die Anfrage blockiert und ein Fehler zurückgegeben werden soll.

> **Note**
>
> Wenn Sie einer Konfiguration einen Inhaltstyp hinzufügen, aber kein Profil angegeben haben, das mit diesem Typ verwendet werden soll, werden Anfragen mit diesem Inhaltstyp CloudFront immer an den Ursprung weitergeleitet.

- Wenn der in einem Abfrageargument angegebene Profilname unbekannt ist — Wenn Sie das `fle-profile` Abfrageargument mit einem Profilnamen angeben, der für Ihre Distribution nicht existiert, können Sie angeben, ob die Anfrage an den Ursprung gesendet werden CloudFront soll, ohne Datenfelder zu verschlüsseln, oder ob die Anfrage blockiert und ein Fehler zurückgegeben werden soll.

In einer Konfiguration können Sie auch angeben, ob die Bereitstellung eines Profils als Abfrageargument in einer URL ein Profil überschreibt, das Sie dem Inhaltstyp für diese Abfrage zugeordnet haben. CloudFront Verwendet standardmäßig das Profil, das Sie einem Inhaltstyp zugeordnet haben, sofern Sie einen angeben. Auf diese Weise können Sie ein Profil haben, das standardmäßig verwendet wird, aber für bestimmte Anforderungen entscheiden, dass Sie ein anderes Profil erzwingen möchten.

So können Sie z. B. (in Ihrer Konfiguration) **`SampleProfile`** als das zu verwendende Abfrageargumentprofil festlegen. Dann könnten Sie `https://d1234.cloudfront.net?fle-profile=SampleProfile` anstelle des Profils`https://d1234.cloudfront.net`, das Sie **`SampleProfile`** für den Inhaltstyp der Anfrage eingerichtet haben, die URL CloudFront verwenden, um diese Anfrage verwenden zu können.

Sie können bis zu 10 Konfigurationen für ein einzelnes Konto erstellen und dann eine der Konfigurationen dem Cache-Verhalten einer beliebigen Verteilung für das Konto zuordnen.

So erstellen Sie eine Konfiguration für die Verschlüsselung auf Feldebene (Konsole)

1. Wählen Sie auf der Seite Verschlüsselung auf Feldebene die Option Create configuration (Konfiguration erstellen) aus.

   Hinweis: Wenn Sie nicht mindestens ein Profil erstellt haben, wird die Option zum Erstellen einer Konfiguration nicht angezeigt.

2. Füllen Sie die folgenden Felder aus, um das zu verwendende Profil anzugeben. (Einige Felder können nicht geändert werden.)

   Inhaltstyp (kann nicht geändert werden)

   Der Inhaltstyp ist auf `application/x-www-form-urlencoded` festgelegt und nicht änderbar.

   Default profile ID (optional)

   Wählen Sie in der Dropdown-Liste das Profil aus, das Sie dem Inhaltstyp im Feld Content type zuordnen möchten.

   Content-Format (kann nicht geändert werden)

   Das Content-Format ist auf `URLencoded` festgelegt und nicht änderbar.

3. Wenn Sie das CloudFront Standardverhalten für die folgenden Optionen ändern möchten, aktivieren Sie das entsprechende Kontrollkästchen.

   Weiterleiten der Anfrage an den Ursprung, wenn der Inhaltstyp der Anfrage nicht konfiguriert ist.

   Aktivieren Sie das Kontrollkästchen, wenn Sie zulassen möchten, dass die Anfrage an Ihren Ursprung geht, wenn Sie kein Profil für den Inhaltstyp der Anfrage angegeben haben.

   Überschreiben des Profils für einen Inhaltstyp mit einem bereitgestellten Abfrageargument

   Aktivieren Sie das Kontrollkästchen, wenn Sie zulassen möchten, dass ein in einem Abfrageargument angegebenes Profil das Profil überschreibt, das Sie für einen Inhaltstyp angegeben haben.

4. Wenn Sie das Kontrollkästchen aktivieren, damit ein Abfrageargument das Standardprofil überschreiben kann, müssen Sie die folgenden zusätzlichen Felder für die Konfiguration ausfüllen. Sie können bis zu fünf dieser Abfrageargumentzuordnungen für die Verwendung mit Abfragen erstellen.

   Query argument

   Geben Sie den Wert ein, den Sie URLs für das `fle-profile` Abfrageargument einschließen möchten. Dieser Wert zeigt CloudFront , dass die Profil-ID (die Sie im nächsten Feld angeben), die mit diesem Abfrageargument verknüpft ist, für die Verschlüsselung auf Feldebene für diese Abfrage verwendet werden soll.

Die maximale Anzahl der Zeichen beträgt 128. Der Wert darf keine Leerzeichen enthalten und darf nur alphanumerische Zeichen oder die folgenden Zeichen enthalten: Bindestrich (-), Punkt (.), Unterstrich (_), Stern (*), Pluszeichen (+), Prozent (%).

Profile ID

Wählen Sie in der Dropdown-Liste das Profil aus, das Sie mit dem Wert verknüpfen möchten, den Sie für Query argument eingegeben haben.

Weiterleiten der Anfrage an den Ursprung, wenn das in einem Abfrageargument angegebene Profil nicht existiert.

Aktivieren Sie das Kontrollkästchen, wenn Sie zulassen möchten, dass die Anfrage zu Ihrem Ursprung geht, wenn das in einem Abfrageargument angegebene Profil nicht in CloudFront definiert ist.

Schritt 5: Hinzufügen einer Konfiguration zu einem Cache-Verhalten

Um eine Verschlüsselung auf Feldebene zu verwenden, verknüpfen Sie eine Konfiguration mit einem Cache-Verhalten für eine Verteilung, indem Sie die Konfigurations-ID als Wert für Ihre Verteilung hinzufügen.

Important

Um eine Verschlüsselungskonfiguration auf Feldebene mit einem Cacheverhalten zu verknüpfen, muss die Verteilung so konfiguriert sein, dass sie immer HTTPS verwendet und HTTP `POST`- und `PUT`-Anforderungen von Viewern akzeptiert. D.h., eine der folgenden Bedingungen muss erfüllt sein:

- Die Viewer Protocol Policy (Viewer-Protokollrichtlinie) des Cacheverhaltens muss Redirect HTTP to HTTPS (HTTP zu HTTPS umleiten) oder HTTPS only (Nur HTTPS) sein. (In AWS CloudFormation oder in der CloudFront API, `ViewerProtocolPolicy` muss auf `redirect-to-https` oder gesetzt sein`https-only`.)
- Die Allowed HTTP Methods (Erlaubte HTTP-Methoden) des Cache-Verhaltens müssen auf GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE gesetzt sein. (In AWS CloudFormation oder in der CloudFront API `AllowedMethods` muss auf`GET`,,,`HEAD`,`OPTIONS`, `PUT` `POST``PATCH`, gesetzt sein`DELETE`. Diese können in beliebiger Reihenfolge angegeben werden.)

- Die Origin Protocol Policy (Ursprungsprotokollrichtlinie) der Ursprungseinstellungen muss auf Match Viewer (Übereinstimmung mit Viewer) oder HTTPS Only (Nur HTTPS) festgelegt sein. (In AWS CloudFormation oder in der CloudFront API, `OriginProtocolPolicy` muss auf `match-viewer` oder gesetzt werden`https-only`.)

Weitere Informationen finden Sie unter Referenz für alle Verteilungseinstellungen.

Entschlüsseln Sie Datenfelder an Ihrem Ursprung

CloudFront verschlüsselt Datenfelder mit dem. AWS Encryption SDK Die Daten bleiben während des gesamten Anwendungs-Stacks verschlüsselt und können nur von Anwendungen abgerufen werden, die über die Anmeldeinformationen verfügen, um sie zu entschlüsseln.

Nach der Verschlüsselung wird der Verschlüsselungstext base64-kodiert. Wenn Ihre Anwendungen den Text am Ursprung entschlüsseln, müssen sie zuerst den Verschlüsselungstext entschlüsseln und dann das AWS Encryption SDK verwenden, um die Daten zu entschlüsseln.

Das folgende Codebeispiel veranschaulicht, wie Anwendungen Daten an Ihrem Ursprung entschlüsseln können. Beachten Sie Folgendes:

- Um das Beispiel zu vereinfachen, lädt dieses Beispiel öffentliche und private Schlüssel (im DER-Format) aus Dateien im Arbeitsverzeichnis. In der Praxis würden Sie den privaten Schlüssel an einem sicheren Offline-Platz, wie z. B. einem Offline-Hardware-Sicherheitsmodul, aufbewahren und den öffentlichen Schlüssel an Ihr Entwicklungsteam verteilen.
- CloudFront verwendet beim Verschlüsseln der Daten spezifische Informationen, und für die Entschlüsselung sollte am Ursprung derselbe Satz von Parametern verwendet werden. Zu den Parametern, CloudFront die bei der Initialisierung verwendet werden, MasterKey gehören:
  - PROVIDER_NAME: Sie haben diesen Wert beim Anlegen eines Verschlüsselungsprofils auf Feldebene angegeben. Verwenden Sie hier denselben Wert.
  - KEY_NAME: Sie haben beim Hochladen einen Namen für Ihren öffentlichen Schlüssel erstellt und dann den Schlüsselnamen im Profil angegeben. CloudFront Verwenden Sie hier denselben Wert.
  - ALGORITHMUS: `RSA/ECB/OAEPWithSHA-256AndMGF1Padding` Wird als Algorithmus für die Verschlüsselung CloudFront verwendet. Sie müssen also denselben Algorithmus verwenden, um die Daten zu entschlüsseln.

- Wenn Sie das folgende Beispielprogramm mit dem Verschlüsselungstext als Eingabe ausführen, werden die entschlüsselten Daten in Ihrer Konsole ausgegeben. Weitere Informationen finden Sie im Java-Beispielcode im AWS Encryption SDK.

Beispiel-Code

```
import java.nio.file.Files;
import java.nio.file.Paths;
import java.security.KeyFactory;
import java.security.PrivateKey;
import java.security.PublicKey;
import java.security.spec.PKCS8EncodedKeySpec;
import java.security.spec.X509EncodedKeySpec;

import org.apache.commons.codec.binary.Base64;

import com.amazonaws.encryptionsdk.AwsCrypto;
import com.amazonaws.encryptionsdk.CryptoResult;
import com.amazonaws.encryptionsdk.jce.JceMasterKey;

/**
 * Sample example of decrypting data that has been encrypted by CloudFront field-level
 encryption.
 */
public class DecryptExample {

    private static final String PRIVATE_KEY_FILENAME = "private_key.der";
    private static final String PUBLIC_KEY_FILENAME = "public_key.der";
    private static PublicKey publicKey;
    private static PrivateKey privateKey;

    // CloudFront uses the following values to encrypt data, and your origin must use
 same values to decrypt it.
    // In your own code, for PROVIDER_NAME, use the provider name that you specified
 when you created your field-level
    // encryption profile. This sample uses 'DEMO' for the value.
    private static final String PROVIDER_NAME = "DEMO";
    // In your own code, use the key name that you specified when you added your public
 key to CloudFront. This sample
    // uses 'DEMOKEY' for the key name.
    private static final String KEY_NAME = "DEMOKEY";
```

```
    // CloudFront uses this algorithm when encrypting data.
    private static final String ALGORITHM = "RSA/ECB/OAEPWithSHA-256AndMGF1Padding";

    public static void main(final String[] args) throws Exception {

        final String dataToDecrypt = args[0];

        // This sample uses files to get public and private keys.
        // In practice, you should distribute the public key and save the private key
in secure storage.
        populateKeyPair();

        System.out.println(decrypt(debase64(dataToDecrypt)));
    }

    private static String decrypt(final byte[] bytesToDecrypt) throws Exception {
        // You can decrypt the stream only by using the private key.

        // 1. Instantiate the SDK
        final AwsCrypto crypto = new AwsCrypto();

        // 2. Instantiate a JCE master key
        final JceMasterKey masterKey = JceMasterKey.getInstance(
                publicKey,
                privateKey,
                PROVIDER_NAME,
                KEY_NAME,
                ALGORITHM);

        // 3. Decrypt the data
        final CryptoResult <byte[], ? > result = crypto.decryptData(masterKey,
bytesToDecrypt);
        return new String(result.getResult());
    }

    // Function to decode base64 cipher text.
    private static byte[] debase64(final String value) {
        return Base64.decodeBase64(value.getBytes());
    }

    private static void populateKeyPair() throws Exception {
        final byte[] PublicKeyBytes =
Files.readAllBytes(Paths.get(PUBLIC_KEY_FILENAME));
```

```
        final byte[] privateKeyBytes =
 Files.readAllBytes(Paths.get(PRIVATE_KEY_FILENAME));
        publicKey = KeyFactory.getInstance("RSA").generatePublic(new
 X509EncodedKeySpec(PublicKeyBytes));
        privateKey = KeyFactory.getInstance("RSA").generatePrivate(new
 PKCS8EncodedKeySpec(privateKeyBytes));
    }
}
```

Video-on-Demand und Live-Streaming-Video mit CloudFront

Sie können CloudFront es für die Bereitstellung von Video-on-Demand (VOD) oder Live-Streaming-Video verwenden, indem Sie einen beliebigen HTTP-Ursprung verwenden. Eine Möglichkeit, Video-Workflows in der Cloud einzurichten, ist die Verwendung CloudFront zusammen mit AWS Media Services.

Themen

Über das Streamen von Videos

Sie müssen einen Encoder verwenden, um Videoinhalte zu verpacken, bevor CloudFront Sie die Inhalte verteilen können. Beim Paketerstellungsprozess werden Segmente erstellt, die Ihre Audio-, Video- und Untertitelinhalte enthalten. Es werden dabei auch Manifestdateien erzeugt, die in einer bestimmten Reihenfolge beschreiben, welche Segmente abgespielt werden sollen und wann. Gängige Paketformate sind MPEG DASH, Apple HLS, Microsoft Smooth Streaming und CMAF.

VOD-Streaming

Beim VOD-Streaming werden Ihre Videoinhalte auf einem Server gespeichert und die Zuschauer können sie jederzeit ansehen. Zum Erstellen einer Komponente (eines Assets), die von den Betrachtern gestreamt werden kann, verwenden Sie einen Encoder, z. B. AWS Elemental MediaConvert, um Ihre Mediendateien zu formatieren und zu verpacken.

Nachdem Ihr Video in die richtigen Formate verpackt wurde, können Sie es auf einem Server oder in einem Amazon S3 S3-Bucket speichern und es dann auf Wunsch der Zuschauer bereitstellen. CloudFront

Live-Video-Streaming

Beim Live-Video-Streaming werden Ihre Videoinhalte in Echtzeit gestreamt, wenn Live-Events stattfinden, oder als Rund-um-die-Uhr-Live-Kanal eingerichtet. Um Live-Ausgaben für die Übertragung und Streaming-Übertragung zu erstellen, verwenden Sie einen Encoder AWS

Elemental MediaLive, um das Video zu komprimieren und für die Wiedergabe auf Geräten zu formatieren.

Nachdem Ihr Video codiert wurde, können Sie es in verschiedenen Übertragungsformaten speichern AWS Elemental MediaStore oder es mithilfe von. AWS Elemental MediaPackage Verwenden Sie eine dieser Quellen, um eine CloudFront Distribution für die Bereitstellung der Inhalte einzurichten. Spezielle Schritte und Anleitungen zur Erstellung von Verteilungen, die zusammen mit diesen Services verwendet werden können, finden Sie unter Stellen Sie das Video bereit, indem Sie AWS Elemental MediaStore es als Quelle verwenden und Bereitstellen Sie Live-Videos, formatiert mit AWS Elemental MediaPackage.

Wowza und Unified Streaming bieten auch Tools, mit denen Sie Videos streamen können. CloudFront Weitere Informationen zur Verwendung von Wowza mit CloudFront finden Sie unter Bringen Sie Ihre Wowza Streaming Engine-Lizenz für CloudFront Live-HTTP-Streaming auf der Wowza-Dokumentationswebsite. Informationen zur Verwendung von Unified Streaming mit CloudFront für VOD-Streaming finden Sie CloudFrontauf der Unified Streaming-Dokumentationswebsite.

Stellen Sie Video-on-Demand bereit mit CloudFront

Verwenden Sie die folgenden Dienste, um Video-on-Demand-Streaming (VOD) bereitzustellen: CloudFront

- Amazon S3, um den Inhalt in seinem Originalformat zu speichern und das transkodierte Video zu speichern,
- Ein Encoder (z. B. AWS Elemental MediaConvert) zur Transcodierung des Videos in Streaming-Formate.
- CloudFront um das transkodierte Video den Zuschauern zur Verfügung zu stellen. Informationen zu Microsoft Smooth Streaming finden Sie unter Video-on-Demand für Microsoft Smooth Streaming konfigurieren.

Um eine VOD-Lösung zu erstellen mit CloudFront

1. Laden Sie Ihre Inhalte in einen Amazon-S3-Bucket hoch. Weitere Informationen zum Arbeiten mit Amazon S3 finden Sie im Benutzerhandbuch zu Amazon Simple Storage Service.
2. Transkodieren Sie Ihre Inhalte mithilfe eines MediaConvert Jobs. Über den Auftrag wird Ihr Video in die Formate konvertiert, die von den Abspielgeräten, die Ihre Viewer verwenden, benötigt

werden. Sie können den Auftrag auch dazu verwenden, um Komponenten (Assets) zu erstellen, die hinsichtlich der Auflösung und Bitrate variieren. Diese Ressourcen werden für das Streaming mit adaptiver Bitrate (ABR) verwendet, bei dem die Anzeigequalität an die verfügbare Bandbreite des Betrachters angepasst wird. MediaConvert speichert das transkodierte Video in einem S3-Bucket.

3. Stellen Sie Ihre konvertierten Inhalte mithilfe einer CloudFront Distribution bereit. Viewer können die Inhalte jederzeit auf jedem Gerät ansehen.

Video-on-Demand für Microsoft Smooth Streaming konfigurieren

Sie haben die folgenden Optionen, CloudFront um Video-on-Demand-Inhalte (VOD) zu verteilen, die Sie in das Microsoft Smooth Streaming-Format transkodiert haben:

- Geben Sie einen Webserver, auf dem Microsoft IIS ausgeführt wird und der Smooth Streaming unterstützt, als Ursprung für Ihre Verteilung an.
- Aktivieren Sie Smooth Streaming im Cache-Verhalten einer CloudFront Distribution. Da Sie in einer Verteilung mehrere Cache-Verhaltensweisen verwenden können, können Sie eine Verteilung sowohl für Smooth Streaming-Mediendateien als auch für andere Inhalte verwenden.

> **Important**
>
> Wenn Sie einen Webserver, auf dem Microsoft IIS ausgeführt wird, als Ihren Ursprung angeben, aktivieren Sie Smooth Streaming nicht im Cache-Verhalten Ihrer CloudFront Distribution. CloudFront kann keinen Microsoft IIS-Server als Ursprung verwenden, wenn Sie Smooth Streaming als Cache-Verhalten aktivieren.

Wenn Sie Smooth Streaming in einem Cache-Verhalten aktivieren (d. h. Sie verfügen nicht über einen Server mit Microsoft IIS), beachten Sie Folgendes:

- Sie können weiterhin andere Inhalte mit demselben Cache-Verhalten verteilen, wenn die Inhalte mit dem Wert von Path Pattern für dieses Cache-Verhalten übereinstimmen.
- CloudFront kann entweder einen Amazon S3 S3-Bucket oder einen benutzerdefinierten Ursprung für Smooth Streaming-Mediendateien verwenden. CloudFront kann keinen Microsoft IIS-Server als Ursprung verwenden, wenn Sie Smooth Streaming für das Cache-Verhalten aktivieren.

- Sie können die Gültigkeit von Mediendateien im Smooth Streaming-Format nicht aufheben. Wenn Sie Dateien aktualisieren möchten, bevor sie ablaufen, müssen Sie diese umbenennen. Weitere Informationen finden Sie unter Inhalte hinzufügen, entfernen oder ersetzen, die CloudFront verbreitet werden.

Informationen zu Smooth Streaming-Clients finden Sie unter Smooth Streaming auf der Microsoft-Dokumentationswebsite.

Wird verwendet CloudFront , um Smooth Streaming-Dateien zu verteilen, wenn ein Microsoft IIS-Webserver nicht der Ursprung ist

1. Transkodieren Sie Ihre Mediendateien in das fragmentierte Smooth Streaming-Format MP4.
2. Führen Sie eine der folgenden Aktionen aus:
   - Wenn Sie die CloudFront Konsole verwenden: Wenn Sie eine Distribution erstellen oder aktualisieren, aktivieren Sie Smooth Streaming in einem oder mehreren Cache-Verhalten der Distribution.
   - Wenn Sie die CloudFront API verwenden: Fügen Sie das `SmoothStreaming` Element dem `DistributionConfig` komplexen Typ für eines oder mehrere Cache-Verhalten der Distribution hinzu.
3. Laden Sie die Smooth Streaming-Dateien auf den Ursprung hoch.
4. Erstellen Sie entweder eine `clientaccesspolicy.xml`- oder eine `crossdomainpolicy.xml`-Datei und fügen Sie diese zu einem Speicherort hinzu, der am Stamm Ihrer Verteilung zugänglich ist, z. B. `https://d111111abcdef8.cloudfront.net/clientaccesspolicy.xml`. Folgendes ist eine Beispielrichtlinie:

```
<?xml version="1.0" encoding="utf-8"?>
<access-policy>
<cross-domain-access>
<policy>
<allow-from http-request-headers="*">
<domain uri="*"/>
</allow-from>
<grant-to>
<resource path="/" include-subpaths="true"/>
</grant-to>
</policy>
</cross-domain-access>
```

```
</access-policy>
```

Weitere Informationen finden Sie unter Making a Service Available Across Domain Boundaries auf der Microsoft Developer Network-Website.

5. Geben Sie in Ihrer Anwendung (z.B. einem Media-Player) für Links die URL für die Mediendatei im folgenden Format an:

   ```
   https://d111111abcdef8.cloudfront.net/video/presentation.ism/Manifest
   ```

Stellen Sie Videostreaming mit CloudFront und AWS Media Services bereit

Informationen zur Verwendung von AWS Media Services mit CloudFront zur Bereitstellung von Live-Inhalten für ein globales Publikum finden Sie in der folgenden Anleitung.

Verwenden Sie AWS Elemental MediaLive, um Live-Video-Streams in Echtzeit zu codieren. Um einen großen Videostream zu kodieren, MediaLive komprimieren Sie ihn in kleinere Versionen (Kodierungen), die an Ihre Zuschauer verteilt werden können.

Nachdem Sie einen Live-Videostream komprimiert haben, können Sie eine der folgenden beiden Hauptoptionen verwenden, um den Inhalt vorzubereiten und bereitzustellen:

- Konvertieren Sie Ihre Inhalte in die erforderlichen Formate und stellen Sie sie dann bereit. Wenn Sie Inhalte in mehreren Formaten benötigen, verwenden Sie diese Option, AWS Elemental MediaPackageum die Inhalte für verschiedene Gerätetypen zu verpacken. Wenn Sie die Inhalte verpacken, können Sie auch zusätzliche Funktionen implementieren und das Digital Rights Management (digitale Rechteverwaltung, DRM) hinzufügen, um eine unbefugte Verwendung Ihrer Inhalte zu verhindern. step-by-stepAnweisungen CloudFront zur Bereitstellung MediaPackage formatierter Inhalte finden Sie unterBereitstellen Sie Live-Videos, formatiert mit AWS Elemental MediaPackage.
- Speichern und Bereitstellen Ihrer Inhalte mit skalierbarem Ursprung — Wenn Inhalte in den Formaten MediaLive codiert sind, die von allen Geräten, die Ihre Zuschauer verwenden, erforderlich sind, verwenden Sie einen hochgradig skalierbaren Ursprung, z. B. AWS Elemental MediaStoreum den Inhalt bereitzustellen. step-by-stepAnweisungen zur Bereitstellung von Inhalten CloudFront , die in einem MediaStore Container gespeichert sind, finden Sie unterStellen Sie das Video bereit, indem Sie AWS Elemental MediaStore es als Quelle verwenden.

Nachdem Sie Ihren Ursprung eingerichtet haben, indem Sie eine dieser Optionen auswählen, können Sie Live-Streaming-Videos mithilfe von CloudFront an Betrachter verteilen.

> **Tip**
>
> Sie können sich über eine AWS Lösung informieren, die automatisch Dienste bereitstellt, um ein hochverfügbares Echtzeit-Anzeigeerlebnis zu schaffen. Die Schritte zur automatischen Bereitstellung dieser Lösung können Sie unter Live-Streaming – automatische Bereitstellung einsehen.

Themen

Stellen Sie das Video bereit, indem Sie AWS Elemental MediaStore es als Quelle verwenden

Wenn Sie ein Video in einem AWS Elemental MediaStoreContainer gespeichert haben, können Sie eine CloudFront Distribution erstellen, um den Inhalt bereitzustellen.

Zu Beginn gewähren Sie CloudFront Zugriff auf Ihren MediaStore Container. Anschließend erstellen Sie eine CloudFront Distribution und konfigurieren sie so, dass sie verwendet werden kann MediaStore.

Um Inhalte aus einem AWS Elemental MediaStore Container bereitzustellen

1. Folgen Sie den Anweisungen unter Amazon den CloudFront Zugriff auf Ihren AWS Elemental MediaStore Container ermöglichen und kehren Sie dann zu diesen Schritten zurück, um Ihre Distribution zu erstellen.
2. Verwenden Sie die folgenden Einstellungen, um eine Verteilung zu erstellen:

    a. Ursprungsdomain — Der Datenendpunkt, der Ihrem MediaStore Container zugewiesen ist. Wählen Sie aus der Drop-down-Liste den MediaStore Container für Ihr Live-Video aus.

    b. Herkunftspfad — Die Ordnerstruktur im MediaStore Container, in dem Ihre Objekte gespeichert sind. Weitere Informationen finden Sie unter the section called “Ursprungspfad”.

c. Benutzerdefinierten Header hinzufügen — Fügen Sie Header-Namen und Werte hinzu CloudFront , wenn Sie benutzerdefinierte Header hinzufügen möchten, wenn Anfragen an Ihren Ursprung weitergeleitet werden.

d. Viewer-Protokollrichtlinie — Wählen Sie „HTTP zu HTTPS umleiten“. Weitere Informationen finden Sie unter the section called “Viewer-Protokollrichtlinien”.

e. Cache-Richtlinie und Origin-Anforderungsrichtlinie

- Wählen Sie für Cache policy (Cache-Richtlinie) die Option Create policy (Richtlinie erstellen) aus und erstellen Sie dann eine Cache-Richtlinie, die Ihren Caching-Anforderungen und der Segmentdauer entspricht. Aktualisieren Sie nach dem Erstellen der Richtlinie die Liste der Cache-Richtlinien und wählen Sie die Richtlinie aus, die Sie gerade erstellt haben.
- Wählen Sie für Origin-Anforderungsrichtlinie die Option CORS- CustomOrigin aus der Dropdownliste aus.

Für die anderen Einstellungen können Sie bestimmte Werte basierend auf anderen technischen Anforderungen oder den Anforderungen Ihres Unternehmens festlegen. Eine Liste aller Optionen für Verteilungen und Informationen über ihre Einstellungen finden Sie unter the section called “Alle Verteilungseinstellungen”.

3. Geben Sie für Links in Ihrer Anwendung (z. B. einem Media Player) den Namen der Mediendatei in demselben Format an, das Sie für andere Objekte verwenden, die Sie bei der Verteilung verwenden. CloudFront

Bereitstellen Sie Live-Videos, formatiert mit AWS Elemental MediaPackage

Wenn Sie einen Livestream mithilfe von formatiert haben AWS Elemental MediaPackage, können Sie eine CloudFront Verteilung erstellen und das Cache-Verhalten für die Bereitstellung des Livestreams konfigurieren. Beim folgenden Vorgang wird davon ausgegangen, dass Sie bereits einen Kanal erstellt und Endpunkte für Ihr Live-Video hinzugefügt haben. MediaPackage

Gehen Sie folgendermaßen vor, um MediaPackage manuell eine CloudFront Distribution für zu erstellen:

Schritt 1: Erstellen und konfigurieren Sie eine CloudFront Distribution

Gehen Sie wie folgt vor, um eine CloudFront Verteilung für den Live-Videokanal einzurichten, mit dem Sie erstellt haben MediaPackage.

So erstellen Sie eine Verteilung für Ihren Live-Video-Kanal

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie Create distribution (Verteilung erstellen) aus.
3. Wählen Sie die Einstellungen für die Verteilung aus, einschließlich der folgenden:

   Ursprungsdomäne

   Der Ursprung, an dem sich Ihr MediaPackage Live-Videokanal und Ihre Endpunkte befinden. Wählen Sie das Textfeld und dann aus der Drop-down-Liste die MediaPackage Ursprungsdomain für Ihr Live-Video aus. Sie können eine Domäne zu mehreren Ursprungsendpunkten zuordnen.

   Wenn Sie Ihre Ursprungsdomäne mit einem anderen AWS -Konto erstellt haben, geben Sie den Ursprung-URL-Wert in das Feld ein. Der Ursprung muss eine HTTPS-URL sein.

   Beispiel: bei einem HLS-Endpunkt wie `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.m3u8` ist die Ursprungsdomäne `3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com`.

   Weitere Informationen finden Sie unter the section called “Ursprungsdomäne”.

   Ursprungspfad

   Der Pfad zum MediaPackage Endpunkt, von dem aus der Inhalt bereitgestellt wird.

   Weitere Informationen über die Funktionsweise eines Ursprungspfads finden Sie unter the section called “Ursprungspfad”.

> **Important**
>
> Der Platzhalterpfad * ist für die Weiterleitung an einer beliebigen Stelle in der CloudFront Distribution erforderlich. Um zu verhindern, dass Anfragen, die keinem expliziten Pfad entsprechen, an den echten Ursprung weitergeleitet werden, erstellen Sie einen „Dummy“ -Ursprung für diesen Platzhalterpfad.

Example : Erstellen eines „Dummy“-Ursprungs

Im folgenden Beispiel werden die Endpunkte `abc123` und `def456` zum „echten“ Ursprung weitergeleitet, aber Anfragen nach Videoinhalten eines anderen Endpunkts werden ohne die richtige Subdomäne an `mediapackage.us-west-2.amazonaws.com` weitergeleitet. Das führt zu einem `404`-HTTP-Fehler.

MediaPackage Endpunkte:

```
https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/
index.m3u8
https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/def456/
index.m3u8
```

CloudFront Ursprung A:

```
Domain: 3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com
Path: None
```

CloudFront Herkunft B:

```
Domain: mediapackage.us-west-2.amazonaws.com
Path: None
```

CloudFront Verhalten des Caches:

```
1. Path: /out/v1/abc123/* forward to Origin A
2. Path: /out/v1/def456/* forward to Origin A
3. Path: * forward to Origin B
```

Für die anderen Verteilungseinstellungen können Sie bestimmte Werte basierend auf anderen technischen Anforderungen oder den Anforderungen Ihres Unternehmens festlegen. Eine Liste aller Optionen für Verteilungen und Informationen über ihre Einstellungen finden Sie unter the section called “Alle Verteilungseinstellungen”.

Wenn Sie die Auswahl der anderen Verteilungseinstellungen abgeschlossen haben, wählen Sie Create Distribution (Verteilung erstellen) aus.

4. Wählen Sie die gerade erstellte Verteilung aus und klicken Sie dann auf Behaviors (Verhaltensweisen).
5. Wählen Sie das Cache-Standardverhalten und anschließend Edit (Bearbeiten) aus. Geben Sie die korrekten Einstellungen für das Cache-Verhalten für den Kanal an, den Sie für den Ursprung auswählen. Später fügen Sie einen oder mehrere zusätzliche Ursprünge hinzu und bearbeiten deren Einstellungen für das Cache-Verhalten.
6. Gehe zur Seite mit den CloudFront Distributionen.
7. Warten Sie, bis der Wert in der Spalte Letzte Änderung für Ihre Distribution von Bereitstellen auf Datum und Uhrzeit geändert wurde, was darauf hinweist, dass CloudFront Ihre Distribution erstellt wurde.

Schritt 2: Fügen Sie Origins für die Domains Ihrer MediaPackage Endgeräte hinzu

Wiederhole die Schritte hier, um jeden deiner MediaPackage Kanalendpunkte zu deiner Distribution hinzuzufügen. Denke dabei daran, dass du einen „Dummy“ -Ursprung erstellen musst.

So fügen Sie andere Endpunkte als Ursprünge hinzu

1. Wähle auf der CloudFront Konsole die Distribution aus, die du für deinen Kanal erstellt hast.
2. Klicken Sie auf Origins (Ursprünge) und wählen Sie Create origin (Ursprung erstellen) aus.
3. Wähle für Origin-Domain in der Drop-down-Liste einen MediaPackage Endpunkt für deinen Kanal aus.
4. Für die anderen Einstellungen legen Sie die Werte basierend auf anderen technischen Anforderungen oder den Anforderungen Ihres Unternehmens fest. Weitere Informationen finden Sie unter the section called “Ursprungseinstellungen”.
5. Wählen Sie Create Origin (Ursprung erstellen) aus.

Schritt 3: Konfigurieren der Cache-Verhaltensweisen für alle Endpunkte

Für jeden Endpunkt müssen Sie Cache-Verhaltensweisen konfigurieren, um Pfadmuster hinzuzufügen, die Anfragen korrekt weiterleiten. Die Pfadmuster, die Sie angeben, hängen vom bereitgestellten Videoformat ab. Das folgende Verfahren umfasst die Pfadmuster-Informationen, die für Apple HLS-, CMAF-, DASH- und Microsoft Smooth Streaming-Formate zu verwenden sind.

Sie richten in der Regel zwei Cache-Verhaltensweisen für jeden Endpunkt ein:

- Das übergeordnete Manifest, bei dem es sich um den Index für Ihre Dateien handelt.

- Die Segmente, die Dateien der Videoinhalte darstellen.

So erstellen Sie ein Cache-Verhalten für einen Endpunkt

1. Wähle auf der CloudFront Konsole die Distribution aus, die du für deinen Kanal erstellt hast.
2. Wählen Sie Behaviors (Verhaltensweisen) und anschließend die Option Create behavior (Verhalten erstellen) aus.
3. Verwenden Sie für das Pfadmuster eine bestimmte MediaPackage `OriginEndpoint` GUID als Pfadpräfix.

   Pfadmuster

   Erstellen Sie für einen HLS-Endpunkt wie `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.m3u8` die folgenden zwei Cache-Verhaltensweisen:

   - Verwenden Sie für übergeordnete und untergeordnete Manifeste `/out/v1/abc123/*.m3u8`.
   - Für die Inhaltssegmente verwenden Sie `/out/v1/abc123/*.ts`.

   Erstellen Sie für einen CMAF-Endpunkt wie `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.m3u8` die folgenden zwei Cache-Verhaltensweisen:

   - Verwenden Sie für übergeordnete und untergeordnete Manifeste `/out/v1/abc123/*.m3u8`.
   - Für die Inhaltssegmente verwenden Sie `/out/v1/abc123/*.mp4`.

   Erstellen Sie für einen DASH-Endpunkt wie `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.mpd` die folgenden zwei Cache-Verhaltensweisen:

   - Verwenden Sie für das übergeordnete Manifest `/out/v1/abc123/*.mpd`.
   - Für die Inhaltssegmente verwenden Sie `/out/v1/abc123/*.mp4`.

   Für einen Microsoft-Smooth-Streaming-Endpunkt wie `https://3ae97e9482b0d011.mediapackage.us-west-2.amazonaws.com/out/v1/abc123/index.ism` wird nur ein Manifest bereitgestellt, sodass Sie nur ein Cache-Verhalten erstellen: `out/v1/abc123/index.ism/*`.

4. Geben Sie für jedes Cache-Verhalten Werte für die folgenden Einstellungen an:

   Viewer-Protokollrichtlinien

   Wählen Sie Redirect HTTP to HTTPS (HTTP an HTTPS umleiten) aus.

   Cache-Richtlinien und Ursprungsanforderungsrichtlinie

   Wählen Sie für Cache policy (Cache-Richtlinie) die Option Create policy (Richtlinie erstellen) aus. Geben Sie für Ihre neue Cache-Richtlinie die folgenden Einstellungen an:

   Mindest-TTL

   Legen Sie diese Einstellung auf 5 Sekunden oder weniger fest, um zu verhindern, dass veralteter Inhalt bereitgestellt wird.

   Abfragezeichenfolgen

   Wählen Sie für Query strings (Abfragezeichenfolgen) (in Cache key settings (Cache-Schlüssel-Einstellungen)) die Option Include specified query strings (Angegebene Abfragezeichenfolgen einschließen) aus. Fügen Sie unter Allow (Erlauben) die folgenden Werte hinzu, indem Sie sie eingeben und anschließend Add item (Element hinzufügen) auswählen:

   - Fügen Sie `m` als Abfragezeichenfolge einen Parameter hinzu, CloudFront den Sie als Grundlage für das Caching verwenden möchten. Die MediaPackage Antwort enthält immer das Tag`?m=###`, um die geänderte Uhrzeit des Endpunkts zu erfassen. Wenn bereits Inhalt mit einem anderen Wert für dieses Tag zwischengespeichert ist, CloudFront wird ein neues Manifest angefordert, anstatt die zwischengespeicherte Version bereitzustellen.
   - Wenn Sie die zeitversetzte Anzeigefunktion in verwenden MediaPackage, geben Sie `start` und `end` als zusätzliche Abfragezeichenfolge-Parameter für das Cache-Verhalten für Manifestanfragen (`*.m3u8`*`.mpd`, und) an. `index.ism/*` Auf diese Weise werden Inhalte bereitgestellt, die spezifisch für den angeforderten Zeitraum in der Manifest-Anfrage sind. Weitere Informationen über die Start- und Endanfrageparameter für die Time-Shift-Anzeige und Formatierung von Inhalten finden Sie unter Time-Shift-Anzeige im AWS Elemental MediaPackage Benutzerhandbuch.
   - Wenn Sie die Manifestfilterfunktion in verwenden MediaPackage, geben Sie `aws.manifestfilter` als zusätzlichen Abfragezeichenfolge-Parameter für die Cache-Richtlinie an, die Sie mit dem Cache-Verhalten für Manifestanforderungen (`*.m3u8`*`.mpd`, und`index.ism/*`) verwenden. Dadurch wird Ihre Distribution

so konfiguriert, dass die `aws.manifestfilter` Abfragezeichenfolge an Ihren MediaPackage Ursprung weitergeleitet wird. Dies ist erforderlich, damit die Manifestfilterfunktion funktioniert. Weitere Informationen finden Sie unter Manifestfilterung im AWS Elemental MediaPackage Benutzerhandbuch.

- Wenn Sie HLS mit niedriger Latenz (LL-HLS) verwenden, geben Sie `_HLS_msn` und `_HLS_part` als zusätzliche Abfragezeichenfolgenparameter für die Cache-Richtlinie an, die Sie zusammen mit dem Cache-Verhalten für Manifestanfragen verwenden (`*.m3u8`). Dadurch wird Ihre Distribution so konfiguriert, dass sie die Zeichenketten `_HLS_msn` und die `_HLS_part` Abfragezeichenfolgen an Ihren MediaPackage Ursprung weiterleitet. Dies ist erforderlich, damit die Funktion zum Blockieren von Playlisten durch LL-HLS funktioniert.

5. Wählen Sie Erstellen aus.
6. Nachdem Sie die Cache-Richtlinie erstellt haben, kehren Sie zum Workflow zur Erstellung des Cacheverhaltens zurück. Aktualisieren Sie die Liste der Cache-Richtlinien und wählen Sie die Richtlinie aus, die Sie gerade erstellt haben.
7. Wählen Sie Create behavior (Verhalten erstellen) aus.
8. Wenn es sich bei Ihrem Endpunkt nicht um einen Microsoft-Smooth-Streaming–Endpunkt handelt, wiederholen Sie diese Schritte, um ein zweites Cache-Verhalten zu erstellen.

Schritt 4: Aktivieren Sie die headerbasierte CDN-Autorisierung MediaPackage

Wir empfehlen, die headerbasierte MediaPackage CDN-Autorisierung zwischen Endpunkten und der Distribution zu aktivieren. MediaPackage CloudFront Weitere Informationen finden Sie unter CDN-Autorisierung aktivieren im MediaPackage Benutzerhandbuch.AWS Elemental MediaPackage

Schritt 5: CloudFront Zur Bereitstellung des Live-Stream-Kanals verwenden

Nachdem Sie die Distribution erstellt, die Ursprünge hinzugefügt, das Cache-Verhalten erstellt und die Header-basierte CDN-Autorisierung aktiviert haben, können Sie den Livestream-Kanal mithilfe von bereitstellen. CloudFront CloudFront leitet Anfragen von Zuschauern auf der Grundlage der Einstellungen, die Sie für das Cache-Verhalten konfiguriert haben, an die richtigen MediaPackage Endpunkte weiter.

Geben Sie für Links in Ihrer Anwendung (z. B. einem Media Player) die URL für die Mediendatei im Standardformat für CloudFront URLs an. Weitere Informationen finden Sie unter the section called “Datei anpassen URLs”.

video-on-demandInhalte bereitstellen mit AWS Elemental MediaPackage

Wenn Sie Ihre video-on-demand (VOD-) Inhalte von einer Quelle AWS Elemental MediaPackage stammen, können Sie eine CloudFront Verteilung erstellen und optimiertes Cache-Verhalten konfigurieren, um die VOD-Inhalte Zuschauern bereitzustellen. Beim folgenden Prozess wird davon ausgegangen, dass Sie bereits eine Verpackungsgruppe mit einer Verpackungskonfiguration erstellt und ein Asset mit aufgenommen haben. MediaPackage

Gehen Sie folgendermaßen vor, um MediaPackage manuell eine CloudFront Distribution für zu erstellen:

Schritt 1: Erstellen und konfigurieren Sie eine CloudFront Distribution

Gehen Sie wie folgt vor, um eine CloudFront Verteilung für die Paketgruppe einzurichten, die Sie mit erstellt haben MediaPackage.

Um eine Distribution für Ihre VOD-Inhalte zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie Create distribution (Verteilung erstellen) aus.
3. Wählen Sie die Einstellungen für die Verteilung aus, einschließlich der folgenden:

   Ursprungsdomäne

   Der Ursprung Ihrer MediaPackage Verpackungsgruppe. Geben Sie den Wert der Herkunfts-URL in das Textfeld ein. Der Ursprung muss eine HTTPS-URL sein.

   Beispiel: bei einem HLS-Endpunkt wie `https://3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com/out/v1/abc123/def456/ghi789/index.m3u8` ist die Ursprungsdomäne `3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com`.

   Weitere Informationen finden Sie unter the section called “Ursprungsdomäne”.

   Ursprungspfad

   Der Pfad, von dem aus der Inhalt bereitgestellt wird.

Weitere Informationen über die Funktionsweise eines Ursprungspfads finden Sie unter the section called “Ursprungspfad”.

> **Important**
>
> Der Platzhalterpfad * ist für die Weiterleitung an einer beliebigen Stelle in der CloudFront Distribution erforderlich. Um zu verhindern, dass Anfragen, die keinem expliziten Pfad entsprechen, an den echten Ursprung weitergeleitet werden, erstellen Sie einen „Dummy“ -Ursprung für diesen Platzhalterpfad.

Example : Erstellen eines „Dummy“-Ursprungs

Im folgenden Beispiel werden die Paketierungskonfigurationen `def456` und die `321xyz` Route zum „echten“ Ursprung, aber Anfragen für andere Videoinhalte werden `mediapackage-vod.us-west-2.amazonaws.com` ohne die richtige Subdomain weitergeleitet, was zu einem `404` HTTP-Fehler führt.

MediaPackage Inhalt URLs für ein einzelnes Asset für eine Verpackungsgruppe mit zwei Verpackungskonfigurationen:

```
https://3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com/out/v1/
abc123/def456/ghi789/index.m3u8
https://3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com/out/v1/
abc123/321xyz/654uvw/index.m3u8
```

CloudFront Herkunft A:

```
Domain: 3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com
Path: None
```

CloudFront Herkunft B:

```
Domain: mediapackage-vod.us-west-2.amazonaws.com
Path: None
```

CloudFront Verhalten des Caches:

```
1. Path: /out/v1/*/def456/* forward to Origin A
2. Path: /out/v1/*/321xyz/* forward to Origin A
3. Path: * forward to Origin B
```

Für die anderen Verteilungseinstellungen können Sie bestimmte Werte basierend auf anderen technischen Anforderungen oder den Anforderungen Ihres Unternehmens festlegen. Eine Liste aller Optionen für Verteilungen und Informationen über ihre Einstellungen finden Sie unter the section called “Alle Verteilungseinstellungen”.

Wenn Sie die Auswahl der anderen Verteilungseinstellungen abgeschlossen haben, wählen Sie Create Distribution (Verteilung erstellen) aus.

4. Wählen Sie die gerade erstellte Verteilung aus und klicken Sie dann auf Behaviors (Verhaltensweisen).
5. Wählen Sie das Cache-Standardverhalten und anschließend Edit (Bearbeiten) aus. Geben Sie die richtigen Einstellungen für das Cache-Verhalten für die Verpackungskonfiguration an, die Sie für den Ursprung ausgewählt haben. Später fügen Sie einen oder mehrere zusätzliche Ursprünge hinzu und bearbeiten die Einstellungen für das Cache-Verhalten für sie.
6. Gehen Sie zur Seite „CloudFront Verteilungen“.
7. Warten Sie, bis der Wert in der Spalte Letzte Änderung für Ihre Distribution von Bereitstellen auf Datum und Uhrzeit geändert wurde, was darauf hinweist, dass CloudFront Ihre Distribution erstellt wurde.

Schritt 2: Fügen Sie Origins für die Domains Ihrer MediaPackage Verpackungsgruppen hinzu

Wiederholen Sie die Schritte hier, um jede Ihrer MediaPackage Verpackungsgruppen zu Ihrer Distribution hinzuzufügen. Beachten Sie dabei, dass Sie einen „Dummy“ -Ursprung erstellen müssen.

Um weitere Verpackungsgruppen als Herkunft hinzuzufügen

1. Wählen Sie auf der CloudFront Konsole die Distribution aus, die Sie für Ihren Kanal erstellt haben.
2. Klicken Sie auf Origins (Ursprünge) und wählen Sie Create origin (Ursprung erstellen) aus.
3. Geben Sie für Origin-Domain die URL für die MediaPackage Verpackungsgruppe ein.
4. Für die anderen Einstellungen legen Sie die Werte basierend auf anderen technischen Anforderungen oder den Anforderungen Ihres Unternehmens fest. Weitere Informationen finden Sie unter the section called “Ursprungseinstellungen”.

5. Wählen Sie Create Origin (Ursprung erstellen) aus.

Schritt 3: Konfigurieren Sie das Cache-Verhalten für alle Verpackungskonfigurationen

Für jede Verpackungskonfiguration müssen Sie das Cache-Verhalten konfigurieren, um Pfadmuster hinzuzufügen, die Anfragen korrekt weiterleiten. Die Pfadmuster, die Sie angeben, hängen vom bereitgestellten Videoformat ab. Das folgende Verfahren umfasst die Pfadmuster-Informationen, die für Apple HLS-, CMAF-, DASH- und Microsoft Smooth Streaming-Formate zu verwenden sind.

In der Regel richten Sie mehrere Cache-Verhaltensweisen für jede Verpackungskonfiguration ein:

- Das übergeordnete Manifest, bei dem es sich um den Index für Ihre Dateien handelt.
- Die Segmente, die Dateien der Videoinhalte darstellen. Ein Format kann je nach Konfiguration mehr als eine Erweiterung für Inhalte verwenden. Für jede Erweiterung ist ein Cache-Verhalten erforderlich.

Um ein Cache-Verhalten für eine Verpackungskonfiguration zu erstellen

1. Wählen Sie auf der CloudFront Konsole die Distribution aus, die Sie für Ihren Kanal erstellt haben.
2. Wählen Sie Behaviors (Verhaltensweisen) und anschließend die Option Create behavior (Verhalten erstellen) aus.
3. Verwenden Sie für Path Pattern eine bestimmte GUID für die MediaPackage VOD-Paketkonfiguration als Pfadpräfix. Dies ist die zweite GUID in einem MediaPackage VOD-Pfad.

   Pfadmuster

   Erstellen Sie für HLS-Inhalte wie `https://3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com/out/v1/abc123/def456/ghi789/index.m3u8` die folgenden Cache-Verhaltensweisen:

   - Verwenden Sie für übergeordnete und untergeordnete Manifeste `/out/v1/*/def456/*.m3u8`.
   - Verwenden Sie für die Inhaltssegmente diese Option `/out/v1/*/def456/*.ts` und wiederholen Sie den Vorgang für alle benötigten Segmenterweiterungen.

Erstellen Sie für CMAF-Inhalte wie `https://3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com/out/v1/abc123/def456/ghi789/index.m3u8` die folgenden Cache-Verhaltensweisen:

- Verwenden Sie für übergeordnete und untergeordnete Manifeste `/out/v1/*/def456/*.m3u8`.
- Verwenden Sie für die Inhaltssegmente `/out/v1/*/def456/*.mp4` und wiederholen Sie den Vorgang für alle benötigten Segmenterweiterungen.

Erstellen Sie für DASH-Inhalte wie `https://3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com/out/v1/abc123/def456/ghi789/index.mpd` die folgenden Cache-Verhaltensweisen:

- Verwenden Sie für das übergeordnete Manifest `/out/v1/*/def456/*.mpd`.
- Für die Inhaltssegmente verwenden Sie `/out/v1/*/def456/*.mp4`.

Für einen Microsoft-Smooth-Streaming-Endpunkt wie `https://3ae97e9482b0d011.egress.mediapackage-vod.us-west-2.amazonaws.com/out/v1/abc123/def456/ghi789/index.ism/Manifest` wird nur ein Manifest bereitgestellt, sodass Sie nur ein Cache-Verhalten erstellen: `out/v1/*/def456/*/index.ism/*`.

4. Geben Sie für jedes Cache-Verhalten Werte für die folgenden Einstellungen an:

   Viewer-Protokollrichtlinien

   Wählen Sie Redirect HTTP to HTTPS (HTTP an HTTPS umleiten) aus.

   Cache-Richtlinien und Ursprungsanforderungsrichtlinie

   Wählen Sie für Cache policy (Cache-Richtlinie) die Option Create policy (Richtlinie erstellen) aus. Geben Sie für Ihre neue Cache-Richtlinie die folgenden Einstellungen an:

   Mindest-TTL

   Legen Sie diese Einstellung auf 5 Sekunden oder weniger fest, um zu verhindern, dass veralteter Inhalt bereitgestellt wird.

Abfragezeichenfolgen

Wählen Sie für Query strings (Abfragezeichenfolgen) (in Cache key settings (Cache-Schlüssel-Einstellungen)) die Option Include specified query strings (Angegebene Abfragezeichenfolgen einschließen) aus. Fügen Sie unter Allow (Erlauben) die folgenden Werte hinzu, indem Sie sie eingeben und anschließend Add item (Element hinzufügen) auswählen:

- Wenn Sie die Manifestfilterfunktion in verwenden MediaPackage, geben Sie `aws.manifestfilter` als zusätzlichen Abfragezeichenfolge-Parameter für die Cache-Richtlinie an, die Sie mit dem Cache-Verhalten für Manifestanfragen verwenden (`*.m3u8*.mpd`, und`index.ism/*`). Dadurch wird Ihre Distribution so konfiguriert, dass die `aws.manifestfilter` Abfragezeichenfolge an Ihren MediaPackage Ursprung weitergeleitet wird. Dies ist erforderlich, damit die Manifestfilterfunktion funktioniert. Weitere Informationen finden Sie unter Manifestfilterung im AWS Elemental MediaPackage Benutzerhandbuch.

5. Wählen Sie Erstellen aus.
6. Nachdem Sie die Cache-Richtlinie erstellt haben, kehren Sie zum Workflow zur Erstellung des Cacheverhaltens zurück. Aktualisieren Sie die Liste der Cache-Richtlinien und wählen Sie die Richtlinie aus, die Sie gerade erstellt haben.
7. Wählen Sie Create behavior (Verhalten erstellen) aus.
8. Wenn es sich bei Ihrem Endpunkt nicht um einen Microsoft-Smooth-Streaming–Endpunkt handelt, wiederholen Sie diese Schritte, um ein zweites Cache-Verhalten zu erstellen.

Schritt 4: Aktivieren Sie die MediaPackage headerbasierte CDN-Autorisierung

Wir empfehlen, die headerbasierte MediaPackage CDN-Autorisierung zwischen MediaPackage VOD-Inhalten und der Distribution zu aktivieren. CloudFront Weitere Informationen finden Sie im MediaPackage Benutzerhandbuch unter CDN-Autorisierung aktivieren.AWS Elemental MediaPackage

Schritt 5: CloudFront Zur Bereitstellung des VOD-Inhalts verwenden

Nachdem Sie die Distribution erstellt, die Ursprünge hinzugefügt, das Cache-Verhalten erstellt und die headerbasierte CDN-Autorisierung aktiviert haben, können Sie den VOD-Inhalt mithilfe von bereitstellen. CloudFront CloudFront leitet Anfragen von Zuschauern auf der Grundlage der Einstellungen, die Sie für das Cache-Verhalten konfiguriert haben, an den richtigen MediaPackage VOD-Inhalt weiter.

Geben Sie für Links in Ihrer Anwendung (z. B. einem Media Player) die URL für die Mediendatei im Standardformat für CloudFront URLs an. Weitere Informationen finden Sie unter the section called “Datei anpassen URLs”.

Resilienz im Hinblick auf Medienqualität

Media Quality-Aware Resiliency (MQAR) ist eine integrierte Funktion zwischen Amazon CloudFront und Media Services.AWS MQAR ermöglicht eine automatisierte regionsübergreifende Herkunftsauswahl auf der Grundlage des Media Quality Confidence Score (MQCS). MQCS wird auf der AWS Elemental MediaLive Grundlage von Parametern synthetisiert, die sich auf die Medienqualität auswirken, die von den Zuschauern wahrgenommen wird. Sie können AWS Media Services so konfigurieren CloudFront , dass Ihr Live-Event-Streaming mit hoher Ausfallsicherheit bereitgestellt wird, indem Sie mehrere Optionen verwenden, die Sie in den Failover-Kriterien der CloudFront Ursprungsgruppe angeben können.

Wenn Sie die MQAR-Funktion für Ihre Distribution aktivieren, autorisieren Sie, automatisch CloudFront die Quelle auszuwählen, die den höchsten Qualitätsfaktor hat.

Der Qualitätsfaktor steht für Qualitätsprobleme beim Medienstreaming, die Sie ursprünglich hatten, wie z. B. schwarze Frames, eingefrorene oder fallengelassene Frames oder wiederholte Frames. Wenn Ihre AWS Elemental MediaPackage v2-Quellen beispielsweise in zwei verschiedenen AWS-Regionen Versionen bereitgestellt werden und einer einen höheren Wert für die Medienqualität meldet als der andere, CloudFront wird automatisch zu dem Ursprung gewechselt, der den höheren Wert meldet.

Gehen Sie wie folgt vor, CloudFront um dies zu erreichen:

1. CloudFront leitet eine `GET` Anfrage an den primären MediaPackage Ursprung weiter und initiiert gleichzeitig auch eine `HEAD` Anfrage an den sekundären MediaPackage Ursprung. CloudFront erhält den Medienqualitätsfaktor in den Antwort-Headern der einzelnen Absender.
2. Als Nächstes CloudFront verfolgt er die Punktzahl für jeden Absender und verwendet diese Informationen, um den Ursprung mit der höheren Punktzahl zu bestimmen, wenn eine neue Anfrage eingeht.

Die Bewertung der Medienqualität für Ihre Herkunft kann sich in Echtzeit ändern. CloudFront ermittelt dies anhand der MQCS-Änderungen und wechselt zwischen den Quellen, um sicherzustellen, dass die Zuschauer die Inhalte mit höherer Medienqualität sehen. Weitere Informationen finden Sie

unter Nutzung von Medienqualitätswerten MediaPackage im AWS Elemental MediaPackage V2-Benutzerhandbuch.

MQAR hilft dabei, so früh wie möglich CloudFront festzustellen, ob ein Problem vorliegt, das sich möglicherweise auf Kunden auswirken könnte. Beispielsweise können Probleme wie Netzwerkverbindung, Videoverarbeitung, Audioverlust oder -ausfälle sowie Probleme mit der Encodergeschwindigkeit den Qualitätsfaktor für Ihre Zuschauer beeinträchtigen.

MQAR ermöglicht einen nahtlosen Wechsel zwischen den Quellen, sodass Sie einen stabilen, regionsübergreifenden Workflow für die end-to-end Medienbereitstellung einrichten und AWS Ihren Zuschauern qualitativ hochwertige Inhalte bereitstellen können.

> **Note**
>
> Derzeit unterstützt diese Funktion nur MediaPackage v2-Ursprünge.

Gehen Sie wie folgt vor, um diese Funktion für Ihre Distribution zu aktivieren:

1. Falls Sie dies noch nicht getan haben, erstellen Sie Ihre MediaPackage v2-Ursprünge und aktivieren Sie diese Funktion in Ihrer Endpunktkonfiguration. Erstellen Sie für eine regionsübergreifende Bereitstellung einen sekundären Kanal in einem anderen Kanal AWS-Region mit denselben Einstellungen. Weitere Informationen finden Sie unter den folgenden Themen im AWS Elemental MediaPackage V2-Benutzerhandbuch:
   - Erstellen Sie einen Kanal und einen Endpunkt
   - Aktivieren Sie den Medienqualitätsfaktor
2. Um Ihre MediaPackage v2-Ursprünge für zu verwenden CloudFront, erstellen oder aktualisieren Sie eine CloudFront Distribution. Siehe Eine Verteilung erstellen und Eine Verteilung aktualisieren.
3. Erstellen Sie eine Ursprungsgruppe und wählen Sie Ihre beiden Ursprünge als primären und sekundären Ursprung aus. Aktivieren Sie in Ihrer Herkunftsgruppe die Option Medienqualitätsfaktor. Weitere Informationen finden Sie unter Erstellen Sie eine Ursprungsgruppe.
4. Wählen Sie in Ihrem Cache-Verhalten für Ihre Distribution die von Ihnen erstellte Ursprungsgruppe aus. Wir empfehlen, dass das Verhalten des Caches dem Kanalpfadmuster entspricht.

Wenn CloudFront festgestellt wird, dass beide MediaPackage v2-Ursprünge dieselbe Bewertung haben, wird die Anfrage an den primären Ursprung weitergeleitet, der in der Ursprungsgruppe aufgeführt ist. Wenn der ursprünglich ausgewählte Absender mit einem Fehlercode antwortet, der

den Failover-Kriterien entspricht, die Sie in Ihrer Ursprungsgruppe angegeben haben, CloudFront versucht er, die Anfrage erneut an den alternativen Ursprung in Ihrer Ursprungsgruppe zu senden, unabhängig von dessen Bewertung für die Medienqualität.

Hinweise

- CloudFront verfolgt den Qualitätsfaktor für jedes Cache-Verhalten, das eine für die Medienqualitätsbewertung aktivierte Ursprungsgruppe verwendet. Wenn dieselbe Ursprungsgruppe für mehrere Kanäle verwendet wird, die einen Medienqualitätsfaktor ausgeben, erstellen Sie ein separates Cache-Verhalten für das Pfadmuster jedes Kanals, um eine Vermischung der Punktzahlen zu vermeiden. Weitere Informationen zu Quoten für Ursprungsgruppen finden Sie unterAllgemeine Kontingente für Verteilungen.
- Derzeit ist MQAR nicht verfügbar, wenn Sie eine Lambda @Edge -Funktion in Triggern verwenden, die auf den Ursprung gerichtet sind (Origin-Anfrage und Origin-Antwort), die mit dem Cache-Verhalten Ihrer Distribution verknüpft sind. Weitere Informationen finden Sie unter Einstellungen für das Cache-Verhalten.
- Wenn Sie die MQAR-Funktion und Origin Access Control (OAC) aktiviert haben, fügen Sie die Aktion der `mediapackagev2:GetHeadObject` IAM-Richtlinie hinzu. MQAR benötigt diese Berechtigung, um `HEAD` Anfragen an den v2-Ursprung zu senden. MediaPackage Weitere Informationen zu OAC finden Sie unter. Beschränken Sie den Zugriff auf einen AWS Elemental MediaPackage v2-Ursprung

MQAR-Protokollfelder

CloudFront stellt die folgenden Felder in Echtzeitprotokollen bereit, die den Qualitätsfaktor und den ausgewählten Ursprung widerspiegeln. Sie können diese Felder in Ihren CloudFront Echtzeitprotokollen aktivieren:

- `r-host`
- `sr-reason`
- `x-edge-mqcs`

Weitere Informationen finden Sie unter Felder 65-67.

Personalisieren Sie am Rand mit Funktionen

Mit Amazon können Sie Ihren eigenen Code schreiben CloudFront, um anzupassen, wie Ihre CloudFront Distributionen HTTP-Anfragen und -Antworten verarbeiten. Der Code wird in der Nähe Ihrer Zuschauer (Benutzer) ausgeführt, um die Latenz zu minimieren, und Sie müssen keine Server oder andere Infrastruktur verwalten. Sie können Code schreiben, um die Anfragen und Antworten zu bearbeiten, die durchfließen CloudFront, grundlegende Authentifizierung und Autorisierung durchzuführen, HTTP-Antworten am Edge zu generieren und vieles mehr.

Der Code, den Sie schreiben und an Ihre CloudFront Distribution anhängen, wird als Edge-Funktion bezeichnet. CloudFront bietet zwei Möglichkeiten, Edge-Funktionen zu schreiben und zu verwalten:

CloudFront Funktionen

: Sie können einfache Funktionen JavaScript für umfangreiche, latenzempfindliche CDN-Anpassungen einschreiben. Die Runtime-Umgebung von CloudFront Functions bietet Startzeiten im Submillisekundenbereich, kann sofort skaliert werden, um Millionen von Anfragen pro Sekunde zu verarbeiten, und ist äußerst sicher. CloudFront Functions ist eine native Funktion von CloudFront, was bedeutet, dass Sie Ihren Code vollständig darin erstellen, testen und bereitstellen können. CloudFront

Lambda@Edge

: Lambda @Edge ist eine Erweiterung von AWS Lambda, die leistungsstarkes und flexibles Computing für komplexe Funktionen und vollständige Anwendungslogik bietet, die Ihren Zuschauern näher kommt und zudem äußerst sicher ist. Lambda@Edge-Funktionen laufen in Node.js- oder Python-Laufzeitumgebungen. Sie veröffentlichen sie in einer Single AWS-Region, aber wenn Sie die Funktion einer CloudFront Distribution zuordnen, repliziert Lambda @Edge Ihren Code automatisch auf der ganzen Welt.

Wenn Sie AWS WAF auf laufen CloudFront, können Sie AWS WAF eingefügte Header sowohl für CloudFront Functions als auch für Lambda @Edge verwenden. Dies funktioniert für Anfragen und Antworten von Zuschauern und Absendern.

Themen

- Personalisieren Sie am Rand mit Lambda @Edge
- Einschränkungen für Edge-Funktionen

Unterschiede zwischen CloudFront Functions und Lambda @Edge

CloudFront Functions und Lambda @Edge bieten beide die Möglichkeit, Code als Reaktion auf CloudFront Ereignisse auszuführen.

CloudFront Functions ist ideal für einfache Funktionen mit kurzer Laufzeit für die folgenden Anwendungsfälle:

- Normalisierung von Cache-Schlüsseln — Transformieren Sie HTTP-Anforderungsattribute (Header, Abfragezeichenfolgen, Cookies und sogar den URL-Pfad), um einen optimalen Cache-Schlüssel zu erstellen, der Ihre Cache-Trefferquote verbessern kann.
- Header-Manipulation — Fügen Sie HTTP-Header in die Anfrage oder Antwort ein, ändern oder löschen Sie sie. Beispielsweise können Sie jeder Anfrage einen `True-Client-IP`-Header hinzufügen.
- URL-Weiterleitungen oder -Umschreibungen — Leiten Sie Besucher auf der Grundlage der Informationen in der Anfrage auf andere Seiten weiter oder schreiben Sie alle Anfragen von einem Pfad in einen anderen um.
- Autorisierung anfordern — Überprüfen Sie Hash-Autorisierungstoken wie JSON-Webtoken (JWT), indem Sie die Autorisierungsheader oder andere Metadaten der Anfrage überprüfen.

Informationen zu den ersten Schritten mit CloudFront Funktionen finden Sie unter. Personalisieren Sie am Rand mit CloudFront Funktionen

Lambda @Edge ist ideal für die folgenden Anwendungsfälle:

- Funktionen, deren Ausführung mehrere Millisekunden oder länger dauert
- Funktionen, die eine einstellbare CPU oder einen einstellbaren Arbeitsspeicher erfordern
- Funktionen, die von Bibliotheken von Drittanbietern abhängen (einschließlich des AWS SDK für die Integration mit anderen AWS-Services)
- Funktionen, die Netzwerkzugriff benötigen, um externe Dienste für die Verarbeitung nutzen zu können
- Funktionen, die Dateisystemzugriff oder Zugriff auf den Hauptteil von HTTP-Anfragen erfordern

Die ersten Schritte mit Lambda@Edge finden Sie unter Personalisieren Sie am Rand mit Lambda @Edge.

Um Ihnen bei der Auswahl der Option für Ihren Anwendungsfall zu helfen, verwenden Sie die folgende Tabelle, um die Unterschiede zwischen CloudFront Functions und Lambda @Edge zu verstehen. Hinweise zu den Unterschieden, die für die Hilfsmethoden zur Modifikation von Origin gelten, finden Sie unterWählen Sie zwischen CloudFront Functions und Lambda @Edge.

	CloudFront Funktionen	Lambda@Edge
Programmiersprachen	JavaScript (ECMAScript 5.1-konform)	Node.js und Python
Ereignisquellen	• Betrachteranfrage • Betrachterantwort	• Betrachteranfrage • Betrachterantwort • Ursprungsanfrage • Ursprungsantwort
Unterstützt Amazon CloudFront KeyValueStore	Ja CloudFront KeyValueStore unterstützt nur JavaScript Runtime 2.0	Nein
Skalieren	Bis zu Millionen von Anfragen pro Sekunde	Bis zu 10.000 Anforderungen pro Sekunde pro Region
Dauer der Funktion	Unterhalb einer Millisekunde	Bis zu 5 Sekunden (Betrachteranfrage und Betrachterantwort) Bis zu 30 Sekunden (Ursprungsanfrage und Ursprungsantwort)
Maximale Größe des Funktionsspeichers	2 MB	128 MB (Zuschaueranfrage und Zuschauerantwort)

	CloudFront Funktionen	Lambda@Edge
		10.240 MB (10 GB) (ursprüngliche Anfrage und ursprüngliche Antwort) Weitere Informati onen finden Sie unter Kontingente für Lambda@Edge.
Maximale Größe des Funktionscodes und der enthaltenen Bibliotheken	10 KB	50 MB (Zuschaue ranfrage und Zuschauer antwort) 50 MB (Ursprungsanfrage und Ursprungsantwort)
Netzwerkzugriff	Nein	Ja
Zugriff auf das Dateisystem	Nein	Ja
Zugang zum Anfragetext	Nein	Ja
Zugriff auf Geolokations- und Gerätedat en	Ja	Nein (Zuschaueranfrage und Zuschauerantwort) Ja (ursprüngliche Anfrage und ursprüngliche Antwort)
Kann komplett darin bauen und testen CloudFront	Ja	Nein
Funktionsprotokollierung und Metriken	Ja	Ja

Personalisieren Sie am Rand mit CloudFront Funktionen

Mit CloudFront Functions können Sie einfache Funktionen JavaScript für umfangreiche, latenzempfindliche CDN-Anpassungen schreiben. Ihre Funktionen können die Anfragen und Antworten bearbeiten, die durchfließen CloudFront, grundlegende Authentifizierung und Autorisierung durchführen, HTTP-Antworten am Edge generieren und vieles mehr. Die Runtime-Umgebung von CloudFront Functions bietet Startzeiten unter einer Millisekunde, kann sofort skaliert werden, um Millionen von Anfragen pro Sekunde zu verarbeiten, und ist äußerst sicher. CloudFront Functions ist eine native Funktion von CloudFront, was bedeutet, dass Sie Ihren Code vollständig darin erstellen, testen und bereitstellen können. CloudFront

Wenn Sie eine CloudFront Funktion einer CloudFront Verteilung zuordnen, CloudFront fängt es Anfragen und Antworten an CloudFront Edge-Standorten ab und leitet sie an Ihre Funktion weiter. Sie können CloudFront Funktionen aufrufen, wenn die folgenden Ereignisse eintreten:

- Wann CloudFront erhält er eine Anfrage von einem Zuschauer (Zuschaueranfrage)
- Before CloudFront gibt die Antwort an den Betrachter zurück (Zuschauerantwort)

Weitere Informationen zu CloudFront Funktionen finden Sie in den folgenden Themen:

Themen

Tutorial: Erstellen Sie eine einfache Funktion mit CloudFront Funktionen

Dieses Tutorial zeigt Ihnen, wie Sie mit CloudFront Functions beginnen können. Sie können eine einfache Funktion erstellen, die den Betrachter zu einer anderen URL weiterleitet und die auch einen benutzerdefinierten Antwortheader zurückgibt.

Inhalt

Voraussetzungen

Um CloudFront Functions verwenden zu können, benötigen Sie eine CloudFront Distribution. Falls Sie keines haben, finden Sie weitere Informationen unter Beginnen Sie mit einer CloudFront Standarddistribution.

Erstellen der Funktion

Sie können die CloudFront Konsole verwenden, um eine einfache Funktion zu erstellen, die den Betrachter zu einer anderen URL weiterleitet und außerdem einen benutzerdefinierten Antwortheader zurückgibt.

Um eine CloudFront Funktion zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Funktionen und dann Funktion erstellen aus.
3. Geben Sie auf der Seite Funktion erstellen in das Feld Name einen Funktionsnamen ein, z. *`MyFunctionName`* B.
4. (Optional) Geben Sie unter Beschreibung eine Beschreibung für die Funktion ein, z. **`Simple test function`** B.
5. Behalten Sie für Runtime die ausgewählte JavaScript Standardversion bei.
6. Wählen Sie Funktion erstellen aus.
7. Kopieren Sie den folgenden Funktionscode. Dieser Funktionscode leitet den Betrachter auf eine andere URL um und gibt auch einen benutzerdefinierten Antwortheader zurück.

```
function handler(event) {
    // NOTE: This example function is for a viewer request event trigger.
    // Choose viewer request for event trigger when you associate this function
 with a distribution.
    var response = {
        statusCode: 302,
        statusDescription: 'Found',
        headers: {
            'cloudfront-functions': { value: 'generated-by-CloudFront-Functions' },
            'location': { value: 'https://aws.amazon.com/cloudfront/' }
        }
    };
    return response;
}
```

8. Fügen Sie für Funktionscode den Code in den Code-Editor ein, um den Standardcode zu ersetzen.
9. Wählen Sie Änderungen speichern.
10. (Optional) Sie können die Funktion testen, bevor Sie sie veröffentlichen. In diesem Tutorial wird nicht beschrieben, wie eine Funktion getestet wird. Weitere Informationen finden Sie unter Funktionen testen.
11. Wählen Sie die Registerkarte Veröffentlichen und dann die Funktion Veröffentlichen. Sie müssen die Funktion veröffentlichen, bevor Sie sie Ihrer CloudFront Distribution zuordnen können.
12. Als Nächstes können Sie die Funktion einem Verteilungs- oder Cache-Verhalten zuordnen. Wählen Sie auf der *MyFunctionName* Seite die Registerkarte Veröffentlichen aus.

> **Warning**
>
> Wählen Sie in den folgenden Schritten eine Verteilung oder ein Cache-Verhalten aus, das zum Testen verwendet wird. Ordnen Sie diese Testfunktion nicht einem Verteilungs- oder Cache-Verhalten zu, das in der Produktion verwendet wird.

13. Wählen Sie Add association.
14. Wählen Sie im Dialogfeld Zuordnen ein Verteilungs- und/oder ein Cache-Verhalten aus. Behalten Sie für Ereignistyp den Standardwert bei.
15. Wählen Sie Add association.

    In der Tabelle Zugeordnete Verteilungen wird die zugeordnete Verteilung angezeigt.

16. Warten Sie ein paar Minuten, bis die zugehörige Verteilung die Bereitstellung abgeschlossen hat. Um den Status der Verteilung zu überprüfen, wählen Sie die Verteilung in der Tabelle Zugeordnete Verteilungen aus und klicken Sie dann auf Verteilung anzeigen.

    Wenn der Status der Verteilung Bereitgestellt lautet, können Sie überprüfen, ob die Funktion funktioniert.

Überprüfen Sie die Funktion

Nachdem Sie die Funktion bereitgestellt haben, können Sie überprüfen, ob sie für Ihre Distribution funktioniert.

Um die Funktion zu überprüfen

1. Navigieren Sie in Ihrem Webbrowser zum Domainnamen Ihrer Distribution (z. B.`https://d111111abcdef8.cloudfront.net`).

    Die Funktion gibt eine Umleitung an den Browser zurück, sodass der Browser automatisch an weitergeleitet wir `https://aws.amazon.com/cloudfront/`.
2. In einem Befehlszeilenfenster können Sie beispielsweise ein Tool verwenden, curl um eine Anfrage an den Domainnamen Ihrer Distribution zu senden.

    ```
    curl -v https://d111111abcdef8.cloudfront.net/
    ```

    In der Antwort sehen Sie die Umleitungsantwort (`302 Found`) und die benutzerdefinierten Antwort-Header, die die Funktion hinzugefügt hat. Ihre Antwort könnte wie das folgende Beispiel aussehen.

    Example

    ```
    curl -v https://d111111abcdef8.cloudfront.net/
    > GET / HTTP/1.1
    > Host: d111111abcdef8.cloudfront.net
    > User-Agent: curl/7.64.1
    > Accept: */*
    >
    < HTTP/1.1 302 Found
    < Server: CloudFront
    < Date: Tue, 16 Mar 2021 18:50:48 GMT
    < Content-Length: 0
    ```

```
< Connection: keep-alive
< Location: https://aws.amazon.com/cloudfront/
< Cloudfront-Functions: generated-by-CloudFront-Functions
< X-Cache: FunctionGeneratedResponse from cloudfront
< Via: 1.1 3035b31bddaf14eded329f8d22cf188c.cloudfront.net (CloudFront)
< X-Amz-Cf-Pop: PHX50-C2
< X-Amz-Cf-Id: ULZdIz6j43uGBlXyob_JctF9x7CCbwpNniiMlmNbmwzH1YWP9FsEHg==
```

Tutorial: Erstellen Sie eine CloudFront Funktion, die Schlüsselwerte enthält

Dieses Tutorial zeigt Ihnen, wie Sie Schlüsselwerte in eine CloudFront Funktion einbeziehen. Schlüsselwerte sind Teil eines Schlüssel-Wert-Paares. Sie nehmen den Namen (aus dem Schlüssel-Wert-Paar) in den Funktionscode auf. Wenn die Funktion ausgeführt wird, wird der Name CloudFront durch den Wert ersetzt.

Schlüssel-Wert-Paare sind Variablen, die in einem Schlüsselwertspeicher gespeichert werden. Wenn Sie in Ihrer Funktion einen Schlüssel verwenden (anstelle von hartkodierten Werten), ist Ihre Funktion flexibler. Sie können den Wert des Schlüssels ändern, ohne Codeänderungen vornehmen zu müssen. Schlüssel-Wert-Paare können auch die Größe Ihrer Funktion reduzieren. Weitere Informationen finden Sie unter ???.

Inhalt

Voraussetzungen

Wenn Sie mit CloudFront Funktionen und dem Schlüsselwertspeicher noch nicht vertraut sind, empfehlen wir Ihnen, das Tutorial unter zu befolgen. the section called “Tutorial: Erstelle eine einfache CloudFront Funktion”

Nachdem Sie dieses Tutorial abgeschlossen haben, können Sie diesem Tutorial folgen, um die von Ihnen erstellte Funktion zu erweitern. Für dieses Tutorial empfehlen wir, dass Sie zuerst den Schlüsselwertspeicher erstellen.

Erstellen Sie den Schlüsselwertspeicher

Erstellen Sie zunächst den Schlüsselwertspeicher, der für Ihre Funktion verwendet werden soll.

Um den Schlüsselwertspeicher zu erstellen

1. Planen Sie die Schlüssel-Wert-Paare, die Sie in die Funktion aufnehmen möchten. Notieren Sie sich die Schlüsselnamen. Die Schlüssel-Wert-Paare, die Sie in einer Funktion verwenden möchten, müssen sich in einem einzigen Schlüsselwertspeicher befinden.
2. Entscheiden Sie sich, in welcher Reihenfolge Sie vorgehen möchten. Es gibt zwei Vorgehensweisen:
   - Erstellen Sie einen Schlüsselwertspeicher und fügen Sie dem Speicher Schlüssel-Wert-Paare hinzu. Dann die Funktion erstellen (oder ändern) und die Schlüsselnamen integrieren.
   - Oder: Die Funktion erstellen (oder ändern) und die zu verwendenden Schlüsselnamen integrieren. Erstellen Sie dann einen Schlüsselwertspeicher und fügen Sie die Schlüssel-Wert-Paare hinzu.
3. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
4. Wählen Sie im Navigationsbereich Funktionen und dann die KeyValueStoresRegisterkarte aus.
5. Wählen Sie Erstellen KeyValueStore und geben Sie die folgenden Felder ein:
   - Geben Sie einen Namen und (optional) eine Beschreibung für den Shop ein.
   - Lassen Sie den S3-URI leer. In diesem Tutorial geben Sie die Schlüssel-Wert-Paare manuell ein.
6. Wählen Sie Create (Erstellen) aus. Die Seite mit den Details für den neuen Schlüsselwertspeicher wird angezeigt. Diese Seite enthält den Abschnitt Schlüssel-Wert-Paare, der derzeit leer ist.

Fügen Sie Schlüssel-Wert-Paare zum Schlüssel-Wert-Speicher hinzu

Fügen Sie als Nächstes manuell eine Liste von Schlüssel-Wert-Paaren zum Schlüssel-Wert-Speicher hinzu, den Sie zuvor erstellt haben.

Um Schlüssel-Wert-Paare zum Schlüssel-Wert-Speicher hinzuzufügen

1. Wählen Sie im Abschnitt Schlüssel-Wert-Paare die Option Schlüssel-Wert-Paare hinzufügen aus.

2. Wählen Sie Paar hinzufügen und geben Sie dann einen Schlüssel und einen Wert ein. Klicken Sie auf das Häkchen, um Ihre Änderungen zu bestätigen, und wiederholen Sie diesen Schritt, um weitere hinzuzufügen.
3. Wenn Sie fertig sind, wählen Sie Änderungen speichern, um die Schlüssel-Wert-Paare im Schlüsselwertspeicher zu speichern. Wählen Sie im Bestätigungsdialogfeld die Option Fertig aus.

Sie haben jetzt einen Schlüsselwertspeicher, der eine Gruppe von Schlüssel-Wert-Paaren enthält.

Ordnen Sie den Schlüsselwertspeicher der Funktion zu

Sie haben jetzt den Schlüsselwertspeicher erstellt. Weiterhin haben Sie eine Funktion erstellt oder geändert, die die Schlüsselnamen aus dem Schlüsselwertspeicher enthält. Sie können jetzt den Schlüsselwertspeicher und die Funktion verknüpfen. Sie erstellen diese Zuordnung innerhalb der Funktion.

Um den Schlüsselwertspeicher mit der Funktion zu verknüpfen

1. Wählen Sie im Navigationsbereich Funktionen aus. Die Registerkarte Funktionen wird standardmäßig oben angezeigt.
2. Wählen Sie den Funktionsnamen und wählen Sie im KeyValueStore Abschnitt Zugeordnet die Option Vorhandenes zuordnen aus KeyValueStore.
3. Wählen Sie den Schlüsselwertspeicher aus und klicken Sie auf Zuordnen KeyValueStore.

> **Note**
>
> Sie können jeder Funktion nur einen Schlüsselwertspeicher zuordnen.

Testen und veröffentlichen Sie den Funktionscode

Nachdem Sie den Schlüsselwertspeicher mit Ihrer Funktion verknüpft haben, können Sie den Funktionscode testen und veröffentlichen. Sie sollten den Funktionscode jedes Mal testen, wenn Sie ihn ändern, auch wenn Sie Folgendes tun:

- Der Funktion einen Schlüsselwertspeicher zuordnen.

- Ändern Sie die Funktion und ihren Schlüsselwertspeicher so, dass sie ein neues Schlüssel-Wert-Paar enthalten.
- Ändern Sie den Wert eines Schlüssel-Wert-Paares.

Um den Funktionscode zu testen und zu veröffentlichen

1. Weitere Informationen zum Testen einer Funktion finden Sie unter the section called “Funktionen testen”. Stellen Sie sicher, dass Sie die Funktion in der `DEVELOPMENT`-Phase testen.
2. Veröffentlichen Sie die Funktion, wenn Sie bereit sind, die Funktion (mit den neuen oder überarbeiteten Schlüssel-Wert-Paaren) in einer `LIVE` Umgebung zu verwenden.

   Beim Veröffentlichen wird die Version der Funktion von der `DEVELOPMENT` Bühne in die Live-Phase CloudFront kopiert. Die Funktion hat den neuen Code und ist dem Schlüsselwertspeicher zugeordnet. (Die Zuordnung muss in der Live-Phase nicht erneut ausgeführt werden.)

   Weitere Informationen zum Veröffentlichen einer Funktion finden Sie unter the section called “Funktionen veröffentlichen”.

Funktionscode schreiben

Sie können CloudFront Functions verwenden, um einfache Funktionen JavaScript für umfangreiche, latenzempfindliche CDN-Anpassungen zu schreiben. Ihr Funktionscode kann die Anfragen und Antworten bearbeiten, die durchfließen CloudFront, grundlegende Authentifizierung und Autorisierung durchführen, HTTP-Antworten am Edge generieren und vieles mehr.

Informationen zum Schreiben von Funktionscode für CloudFront Funktionen finden Sie in den folgenden Themen. Codebeispiele finden Sie unter CloudFront Funktionen, Beispiele für CloudFront und das amazon-cloudfront-functions Repository auf GitHub.

Themen

- Ermitteln Sie den Zweck der Funktion
- CloudFront Funktionen, Ereignisstruktur
- JavaScript Laufzeitfunktionen für CloudFront Funktionen
- Hilfsmethoden für Schlüsselwertspeicher
- Hilfsmethoden für die Änderung des Ursprungs
- Hilfsmethoden für CloudFront SaaS Manager-Eigenschaften

- Verwendung von async und await

Ermitteln Sie den Zweck der Funktion

Bevor Sie Ihren Funktionscode schreiben, bestimmen Sie den Zweck Ihrer Funktion. Die meisten CloudFront Funktionen in Functions haben einen der folgenden Zwecke.

Themen

- Ändern der HTTP-Anforderung in einem Viewer-Anforderungsereignistyp
- Generieren einer HTTP-Antwort in einem Viewer-Anforderungsereignistyp
- Ändern der HTTP-Antwort in einem Viewer-Antwortereignistyp
- Ähnliche Informationen

Unabhängig vom Zweck Ihrer Funktion ist `handler` der Einstiegspunkt für jede Funktion. Es benötigt ein einzelnes aufgerufenes Argument`event`, das an die Funktion von übergeben wird CloudFront. `event` ist ein JSON-Objekt, das eine Darstellung der HTTP-Anfrage enthält (und der Antwort, wenn Ihre Funktion die HTTP-Antwort ändert).

Ändern der HTTP-Anforderung in einem Viewer-Anforderungsereignistyp

Ihre Funktion kann die HTTP-Anfrage ändern, die vom Viewer (Client) CloudFront empfangen wird, und die geänderte Anfrage CloudFront zur weiteren Verarbeitung an sie zurücksenden. Beispielsweise könnte Ihr Funktionscode den Cache-Schlüssel normalisieren oder Anforderungs-Header ändern.

Nachdem Sie eine Funktion erstellt und veröffentlicht haben, die die HTTP-Anfrage ändert, stellen Sie sicher, dass Sie eine Zuordnung für den Ereignistyp der Viewer-Anforderung hinzufügen. Weitere Informationen finden Sie unter Erstellen der Funktion. Dadurch wird die Funktion jedes Mal ausgeführt, wenn CloudFront eine Anfrage von einem Viewer eingeht, bevor überprüft wird, ob sich das angeforderte Objekt im CloudFront Cache befindet.

Example Beispiel

Der folgende Pseudocode zeigt die Struktur einer Funktion, die die HTTP-Anfrage ändert.

```
function handler(event) {
    var request = event.request;

    // Modify the request object here.
```

```
    return request;
}
```

Die Funktion gibt das geänderte `request` Objekt an zurück CloudFront. CloudFrontsetzt die Verarbeitung der zurückgegebenen Anfrage fort, indem sie den CloudFront Cache auf einen Cache-Treffer überprüft und die Anfrage gegebenenfalls an den Ursprung sendet.

Generieren einer HTTP-Antwort in einem Viewer-Anforderungsereignistyp

Ihre Funktion kann am Edge eine HTTP-Antwort generieren und sie direkt an den Viewer (Client) zurückgeben, ohne nach einer zwischengespeicherten Antwort zu suchen oder sie weiter zu verarbeiten. CloudFront Beispielsweise könnte Ihr Funktionscode die Anfrage an eine neue URL umleiten oder nach Autorisierung suchen und eine `401`- oder `403`-Antwort auf nicht autorisierte Anfragen zurückgeben.

Wenn Sie eine Funktion erstellen, die eine HTTP-Antwort generiert, achten Sie darauf, den Ereignistyp der Betrachteranfrage auszuwählen. Das bedeutet, dass die Funktion jedes Mal ausgeführt wird, wenn sie eine Anfrage von einem Viewer CloudFront erhält, bevor CloudFront die Anfrage weiter verarbeitet wird.

Example Beispiel

Der folgende Pseudocode zeigt die Struktur einer Funktion, die eine HTTP-Antwort generiert.

```
function handler(event) {
    var request = event.request;

    var response = ...; // Create the response object here,
                        // using the request properties if needed.

    return response;
}
```

Die Funktion gibt ein `response` Objekt an zurück CloudFront, das CloudFront sofort zum Viewer zurückkehrt, ohne den CloudFront Cache zu überprüfen oder eine Anfrage an den Ursprung zu senden.

Ändern der HTTP-Antwort in einem Viewer-Antwortereignistyp

Ihre Funktion kann die HTTP-Antwort ändern, bevor sie CloudFront an den Viewer (Client) gesendet wird, unabhängig davon, ob die Antwort aus dem CloudFront Cache oder vom Ursprung stammt.

Beispielsweise könnte Ihr Funktionscode Antwortheader, Statuscodes oder Textinhalte hinzufügen oder ändern.

Wenn Sie eine Funktion erstellen, die die HTTP-Antwort ändert, achten Sie darauf, den Ereignistyp Betrachterantwort zu wählen. Das bedeutet, dass die Funktion ausgeführt wird, bevor sie eine Antwort an den Viewer CloudFront zurückgibt, unabhängig davon, ob die Antwort aus dem CloudFront Cache oder vom Ursprung stammt.

Example Beispiel

Der folgende Pseudocode zeigt die Struktur einer Funktion, die die HTTP-Antwort ändert.

```
function handler(event) {
    var request = event.request;
    var response = event.response;

    // Modify the response object here,
    // using the request properties if needed.

    return response;
}
```

Die Funktion gibt das geänderte `response` Objekt an zurück CloudFront, das CloudFront sofort zum Viewer zurückkehrt.

Ähnliche Informationen

Weitere Informationen zum Arbeiten mit CloudFront Funktionen finden Sie in den folgenden Themen:

- Ereignisstruktur
- JavaScript Laufzeitfunktionen
- CloudFront Funktionen, Beispiele
- Einschränkungen für Edge-Funktionen

CloudFront Funktionen, Ereignisstruktur

CloudFront Functions übergibt ein `event` Objekt als Eingabe an Ihren Funktionscode, wenn die Funktion ausgeführt wird. Wenn Sie eine Funktion testen, erstellen Sie das `event`-Objekt und übergeben es an Ihre Funktion. Wenn Sie ein `event`-Objekt zum Testen einer Funktion erstellen,

können Sie die Felder `distributionDomainName`, `distributionId` und `requestId` im `context`-Objekt weglassen. Stellen Sie sicher, dass die Namen der Header in Kleinbuchstaben geschrieben sind, was bei dem `event` Objekt, das CloudFront Functions in der Produktion an Ihre Funktion weitergibt, immer der Fall ist.

Im Folgenden wird ein Überblick über die Struktur dieses Ereignisobjekts gegeben.

```
{
    "version": "1.0",
    "context": {
        <context object>
    },
    "viewer": {
        <viewer object>
    },
    "request": {
        <request object>
    },
    "response": {
        <response object>
    }
}
```

Weitere Informationen finden Sie unter den folgenden Themen:

Themen

- Feld Version
- Context-Objekt
- Betrachterobjekt
- Objekt anfordern
- Antwortobjekt
- Statuscode und Text
- Struktur von Abfragezeichenfolge, Header und Cookie
- Beispiel für Antwortobjekt
- Beispiel für Ereignisobjekt

Feld Version

Das `version` Feld enthält eine Zeichenfolge, die die Version des CloudFront Functions-Ereignisobjekts angibt. Die aktuelle Version ist `1.0`.

Context-Objekt

Das `context`-Objekt enthält kontextbezogene Informationen über das Ereignis. Er enthält folgende Felder:

`distributionDomainName`

Der CloudFront Domänenname (z. B. d111111abcdef8.cloudfront.net) der Standarddistribution, die dem Ereignis zugeordnet ist.

Das `distributionDomainName` Feld wird nur angezeigt, wenn Ihre Funktion für Standardverteilungen aufgerufen wird.

`endpoint`

Der CloudFront Domänenname (z. B. d111111abcdef8.cloudfront.net) der Verbindungsgruppe, die dem Ereignis zugeordnet ist.

Das `endpoint` Feld wird nur angezeigt, wenn Ihre Funktion für Mehrmandantenverteilungen aufgerufen wird.

`distributionId`

Die ID der Verteilung (z. B. EXAMPLE), EDFDVBD6 die dem Ereignis zugeordnet ist.

`eventType`

Der Ereignistyp, entweder `viewer-request` oder `viewer-response`.

`requestId`

Eine Zeichenfolge, die eine CloudFront Anfrage (und die zugehörige Antwort) eindeutig identifiziert.

Betrachterobjekt

Das `viewer`-Objekt enthält ein `ip`-Feld, dessen Wert die IP-Adresse des Betrachters (Clients) ist, der die Anfrage gesendet hat. Wenn der Betrachter einen HTTP-Proxy oder einen Load Balancer

verwendet hat, um die Anfrage zu senden, entspricht der Wert der IP-Adresse des Proxys bzw. des Load Balancers.

Objekt anfordern

Das `request` Objekt enthält eine Darstellung einer viewer-to-CloudFront HTTP-Anfrage. In dem `event` Objekt, das an Ihre Funktion übergeben wurde, stellt das `request` Objekt die tatsächliche Anfrage dar, die vom Viewer CloudFront empfangen wurde.

Wenn Ihr Funktionscode ein `request` Objekt an zurückgibt CloudFront, muss es dieselbe Struktur verwenden.

Das `request`-Objekt enthält die folgenden Felder:

`method`

Die HTTP-Methode der Anforderung. Wenn Ihr Funktionscode a zurückgibt`request`, kann er dieses Feld nicht ändern. Dies ist das einzige schreibgeschützte Feld im `request`-Objekt.

`uri`

Der relative Pfad des angeforderten Objekts.

> **Note**
>
> Wenn Ihre Funktion den `uri` Wert ändert, gilt Folgendes:
>
> - Der neue `uri`-Wert muss mit einem Schrägstrich (/) beginnen (`/`).
> - Wenn eine Funktion den `uri`-Wert ändert, ändert sie auch das Objekt, das die Betrachter anfordert.
> - Wenn eine Funktion den `uri`-Wert ändert, wird weder das Cache-Verhalten für die Anforderung noch der Ursprung geändert, an den die Ursprungsanfrage weitergeleitet wird.

`querystring`

Ein Objekt, das die Abfragezeichenkette in der Anfrage darstellt. Wenn die Anfrage keine Abfragezeichenfolge enthält, enthält das `request`-Objekt immer noch ein leeres `querystring`-Objekt.

Das `querystring`-Objekt enthält ein Feld für jeden Abfragezeichenfolgenparameter in der Anforderung.

`headers`

Ein Objekt, das den HTTP-Header in der Anforderung darstellt. Wenn die Anfrage `Cookie`-Header enthält, sind diese Header nicht Teil des `headers`-Objekts. Cookies werden im `cookies`-Objekt separat dargestellt.

Das `headers`-Objekt enthält ein Feld für jeden Header in der Anfrage. Header-Namen werden im Event-Objekt in ASCII-Kleinbuchstaben umgewandelt, und Header-Namen müssen ASCII-Kleinbuchstaben sein, wenn sie durch Ihren Funktionscode hinzugefügt werden. Wenn CloudFront Functions das Ereignisobjekt wieder in eine HTTP-Anfrage konvertiert, wird der erste Buchstabe jedes Worts in Header-Namen groß geschrieben, sofern es sich um einen ASCII-Buchstaben handelt. CloudFront Functions wendet keine Änderungen auf Nicht-ASCII-Symbole in Header-Namen an. Wird zum Beispiel in `TÈst-header` die Funktion integriert`tÈst-header`. Das Nicht-ASCII-Symbol `È` ist unverändert.

Wörter werden durch einen Bindestrich (`-`) getrennt. Wenn Ihr Funktionscode beispielsweise einen Header mit dem Name`example-header-name`, CloudFront konvertiert diesen `Example-Header-Name` in der HTTP-Anfrage in.

`cookies`

Ein Objekt, das die Cookies in der Anfrage darstellt (`Cookie`-Header).

Das `cookies`-Objekt enthält ein Feld für jeden Cookie in der Anfrage.

Weitere Hinweise zur Struktur von Abfragezeichenfolgen, Headern und Cookies finden Sie unter Struktur von Abfragezeichenfolge, Header und Cookie.

Ein `event`-Beispielobjekt finden Sie unter Beispiel für Ereignisobjekt.

Antwortobjekt

Das `response` Objekt enthält eine Darstellung einer CloudFront-to-viewer HTTP-Antwort. In dem `event` Objekt, das an Ihre Funktion übergeben wurde, stellt CloudFront das `response` Objekt die tatsächliche Antwort auf eine Viewer-Anfrage dar.

Wenn Ihr Funktionscode ein `response`-Objekt zurückgibt, muss er dieselbe Struktur verwenden.

Das `response`-Objekt enthält die folgenden Felder:

`statusCode`

Den HTTP-Statuscode der Antwort. Dieser Wert ist eine Ganzzahl, keine Zeichenfolge.

Ihre Funktion kann den `statusCode` generieren oder ändern.

`statusDescription`

Die HTTP-Statusbeschreibung der Antwort. Wenn Ihr Funktionscode eine Antwort generiert, ist dieses Feld optional.

`headers`

Ein Objekt, das die HTTP-Header in der Antwort darstellt. Wenn die Antwort `Set-Cookie` Header enthält, sind diese Header nicht Teil des `headers`-Objekts. Cookies werden im `cookies`-Objekt separat dargestellt.

Das `headers`-Objekt enthält ein Feld für jeden Header in der Antwort. Headernamen werden im Ereignisobjekt in Kleinbuchstaben konvertiert und Headernamen müssen in Kleinbuchstaben stehen, wenn sie vom Funktionscode hinzugefügt werden. Wenn CloudFront Functions das Ereignisobjekt wieder in eine HTTP-Antwort konvertiert, wird der erste Buchstabe jedes Worts in Header-Namen groß geschrieben. Wörter werden durch einen Bindestrich (`-`) getrennt. Wenn Ihr Funktionscode beispielsweise einen Header mit dem Namen`example-header-name`, CloudFront konvertiert diesen `Example-Header-Name` in die HTTP-Antwort.

`cookies`

Ein Objekt, das die Cookies in der Antwort darstellt (`Set-Cookie`-Header).

Das `cookies`-Objekt enthält ein Feld für jedes Cookie in der Antwort.

`body`

Das Hinzufügen des `body`-Felds ist optional. Es wird im `response`-Objekt nur dann vorhanden sein, wenn Sie es in Ihrer Funktion angeben. Ihre Funktion hat keinen Zugriff auf den Originaltext, der vom CloudFront Cache oder Origin zurückgegeben wurde. Wenn Sie das `body` Feld in Ihrer Viewer-Antwortfunktion nicht angeben, wird der ursprüngliche Text, der vom CloudFront Cache oder Origin zurückgegeben wurde, an den Viewer zurückgegeben.

Wenn Sie einen benutzerdefinierten Text CloudFront an den Viewer zurückgeben möchten, geben Sie den Textinhalt im `data` Feld und die Textkodierung im `encoding` Feld an. Sie können die Codierung als Klartext (`"encoding": "text"`) oder als Base64-codierten Inhalt (`"encoding": "base64"`) angeben.

Als Abkürzung können Sie den Textinhalt auch direkt im Feld `body` angeben (`"body": "<specify the body content here>"`). Wenn Sie dies tun, lassen Sie die `encoding` Felder `data` und weg. CloudFront behandelt den Hauptteil in diesem Fall als einfachen Text.

`encoding`

Die Codierung für den `body`-Inhalt (Feld `data`). Die einzigen gültigen Codierungen sind `text` und `base64`.

Wenn Sie `encoding` als angeben`base64`, der Hauptteil aber nicht gültig ist, wird Base64 CloudFront zurückgegeben.

`data`

Der `body`-Inhalt.

Weitere Informationen zu geänderten Statuscodes und Textinhalten finden Sie unter Statuscode und Text.

Weitere Informationen zur Struktur von Headern und Cookies finden Sie unter Struktur von Abfragezeichenfolge, Header und Cookie.

Ein `response`-Beispielobjekt finden Sie unter Beispiel für Antwortobjekt.

Statuscode und Text

Mit CloudFront Funktionen können Sie den Statuscode der Viewer-Antwort aktualisieren, den gesamten Antworttext durch einen neuen ersetzen oder den Antworttext entfernen. Zu den häufigsten Szenarien für die Aktualisierung der Antwort des Betrachters nach der Auswertung von Aspekten der Antwort aus dem CloudFront Cache oder dem Ursprung gehören die folgenden:

- Ändern des Status, um einen HTTP-200-Statuscode festzulegen, und Erstellen statischer Textinhalte für die Rückgabe an den Viewer.
- Ändern des Status, um einen HTTP-301- oder -302-Statuscode festzulegen, der den Benutzer auf eine andere Website umleitet.
- Entscheiden, ob der Text der Viewer-Antwort weitergeleitet oder verworfen werden soll.

> **Note**
>
> Wenn der Ursprung einen HTTP-Fehler von 400 und höher zurückgibt, wird die CloudFront Funktion nicht ausgeführt. Weitere Informationen finden Sie unter Einschränkungen für alle Edge-Funktionen.

Wenn Sie mit der HTTP-Antwort arbeiten, hat CloudFront Functions keinen Zugriff auf den Antworttext. Sie können den Textinhalt ersetzen, indem Sie ihn auf den gewünschten Wert setzen, oder den Text entfernen, indem Sie den Wert auf leer setzen. Wenn Sie das Textfeld in Ihrer Funktion nicht aktualisieren, wird der ursprüngliche Text, der vom CloudFront Cache oder Origin zurückgegeben wurde, an den Viewer zurückgegeben.

> **Tip**
>
> Wenn Sie CloudFront Funktionen verwenden, um einen Hauptteil zu ersetzen, achten Sie darauf, die entsprechenden Überschriften, z. B. `content-encodingcontent-type`, oder`content-length`, am neuen Hauptinhalt auszurichten.
> Wenn beispielsweise der CloudFront Ursprung oder der Cache zurückgegeben wird, die Funktion „Antwort des Betrachters“ `content-encoding: gzip` jedoch einen Textkörper festlegt, der aus reinem Text besteht, muss die Funktion auch die `content-type` Überschriften `content-encoding` und entsprechend ändern.

Wenn Ihre CloudFront Funktion so konfiguriert ist, dass sie einen HTTP-Fehler von 400 oder höher zurückgibt, wird Ihrem Viewer keine benutzerdefinierte Fehlerseite angezeigt, die Sie für denselben Statuscode angegeben haben.

Struktur von Abfragezeichenfolge, Header und Cookie

Abfragezeichenfolgen, Header und Cookies haben dieselbe Struktur. Abfragezeichenfolgen können in Anforderungen vorkommen. Header erscheinen in Anforderungen und Antworten. Cookies erscheinen in Anforderungen und Antworten.

Jede Abfragezeichenfolge, jeder Header oder jedes Cookie ist ein eindeutiges Feld innerhalb des übergeordneten `querystring`-, `headers`- oder `cookies`-Objekts. Der Feldname ist der Name der Abfragezeichenfolge, des Headers oder des Cookies. Jedes Feld enthält eine `value`-Eigenschaft mit dem Wert der Abfragezeichenfolge, des Headers oder des Cookies.

Inhalt

Werte oder Objekte von Abfragezeichenfolgen

Eine Funktion kann zusätzlich zum Objekt den Wert einer Abfragezeichenfolge zurückgeben. Der Wert der Abfragezeichenfolge kann verwendet werden, um die Parameter der Abfragezeichenfolge in beliebiger benutzerdefinierter Reihenfolge anzuordnen.

Example Beispiel

Verwenden Sie Code wie den folgenden, um eine Abfragezeichenfolge in Ihrem Funktionscode zu ändern.

```
var request = event.request;
request.querystring =
 'ID=42&Exp=1619740800&TTL=1440&NoValue=&querymv=val1&querymv=val2,val3';
```

Besondere Überlegungen für Header

Nur für Header werden Headernamen im Ereignisobjekt in Kleinbuchstaben konvertiert und Headernamen müssen Kleinbuchstaben aufweisen, wenn sie vom Funktionscode hinzugefügt werden. Wenn CloudFront Functions das Ereignisobjekt wieder in eine HTTP-Anfrage oder -Antwort konvertiert, wird der erste Buchstabe jedes Worts in Header-Namen groß geschrieben. Wörter werden durch einen Bindestrich (`-`) getrennt. Wenn Ihr Funktionscode beispielsweise einen Header mit dem Name`example-header-name`, CloudFront konvertiert diesen `Example-Header-Name` in die HTTP-Anfrage oder -Antwort.

Example Beispiel

Betrachten Sie den folgenden `Host` Header in einer HTTP-Anfrage.

```
Host: video.example.com
```

Dieser Header wird im `request`-Objekt wie folgt dargestellt:

```
"headers": {
    "host": {
        "value": "video.example.com"
    }
}
```

Um auf den `Host`-Header in Ihrem Funktionscode zuzugreifen, verwenden Sie Code wie den folgenden:

```
var request = event.request;
var host = request.headers.host.value;
```

Um einen Header in Ihrem Funktionscode hinzuzufügen oder zu ändern, verwenden Sie Code wie den folgenden (dieser Code fügt einen Header mit Namen `X-Custom-Header` und Wert `example value` hinzu):

```
var request = event.request;
request.headers['x-custom-header'] = {value: 'example value'};
```

Doppelte Abfragezeichenfolgen, Header und Cookies (**`multiValue`**-Array)

Eine HTTP-Anfrage oder Antwort kann mehr als eine Abfragezeichenfolge, einen Header oder ein Cookie mit demselben Namen enthalten. In diesem Fall sind die doppelten Abfragezeichenfolgen, -Header oder -Cookies in einem Feld im `request`- oder `response`-Objekt zusammengefasst, aber dieses Feld enthält eine zusätzliche Eigenschaft namens `multiValue`. Die `multiValue`-Eigenschaft enthält ein Array mit den Werten der doppelten Abfragezeichenfolgen, Header oder Cookies.

Example Beispiel

Stellen Sie sich eine HTTP-Anfrage mit den folgenden `Accept` Headern vor.

```
Accept: application/json
Accept: application/xml
Accept: text/html
```

Diese Header werden im Objekt wie folgt dargestellt. `request`

```
"headers": {
```

```
    "accept": {
        "value": "application/json",
        "multiValue": [
            {
                "value": "application/json"
            },
            {
                "value": "application/xml"
            },
            {
                "value": "text/html"
            }
        ]
    }
}
```

> **Note**
>
> Der erste Header-Wert (in diesem Fall`application/json`) wird sowohl in den `multiValue` Eigenschaften als `value` auch wiederholt. Auf diese Weise können Sie auf alle Werte zugreifen, indem Sie das `multiValue`-Array durchlaufen.

Wenn Ihr Funktionscode eine Abfragezeichenfolge, einen Header oder ein Cookie mit einem `multiValue` Array ändert, verwendet CloudFront Functions die folgenden Regeln, um die Änderungen anzuwenden:

1. Wenn das `multiValue`-Array existiert und Änderungen hat, wird diese Änderung angewendet. Das erste Element in der `value`-Eigenschaft wird ignoriert.
2. Andernfalls wird jede Änderung der `value`-Eigenschaft angewendet, und nachfolgende Werte (falls vorhanden) bleiben unverändert.

Die `multiValue`-Eigenschaft wird nur verwendet, wenn die HTTP-Anfrage oder Antwort doppelte Abfragezeichenfolgen, Header oder Cookies mit demselben Namen enthält, wie im vorherigen Beispiel gezeigt. Wenn jedoch mehrere Werte in einer einzelnen Abfragezeichenfolge, einem Header oder einem Cookie vorhanden sind, wird die `multiValue`-Eigenschaft nicht verwendet.

Example Beispiel

Stellen Sie sich eine Anfrage mit einem `Accept` Header vor, der drei Werte enthält.

```
Accept: application/json, application/xml, text/html
```

Dieser Header wird im `request` Objekt wie folgt dargestellt.

```
"headers": {
    "accept": {
        "value": "application/json, application/xml, text/html"
    }
}
```

Cookie-Attribute

In einem `Set-Cookie`-Header in einer HTTP-Antwort enthält der Header das Name-Wert-Paar für das Cookie und optional eine Reihe von durch Semikolons getrennten Attributen.

Example Beispiel

```
Set-Cookie: cookie1=val1; Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr 2021
 07:28:00 GMT
```

In dem `response`-Objekt werden diese Attribute in der `attributes`-Eigenschaft des Cookie-Feldes dargestellt. Der vorangehende `Set-Cookie`-Header wird beispielsweise wie folgt dargestellt:

```
"cookie1": {
    "value": "val1",
    "attributes": "Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr 2021
 07:28:00 GMT"
}
```

Beispiel für Antwortobjekt

Das folgende Beispiel zeigt ein `response`-Objekt – die Ausgabe einer Viewer-Antwortfunktion –, in dem der Text durch eine Viewer-Antwortfunktion ersetzt wurde.

```
{
  "response": {
    "statusCode": 200,
    "statusDescription": "OK",
    "headers": {
      "date": {
```

```
        "value": "Mon, 04 Apr 2021 18:57:56 GMT"
      },
      "server": {
        "value": "gunicorn/19.9.0"
      },
      "access-control-allow-origin": {
        "value": "*"
      },
      "access-control-allow-credentials": {
        "value": "true"
      },
      "content-type": {
        "value": "text/html"
      },
      "content-length": {
        "value": "86"
      }
    },
    "cookies": {
      "ID": {
        "value": "id1234",
        "attributes": "Expires=Wed, 05 Apr 2021 07:28:00 GMT"
      },
      "Cookie1": {
        "value": "val1",
        "attributes": "Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr 2021
07:28:00 GMT",
        "multiValue": [
          {
            "value": "val1",
            "attributes": "Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr 2021
07:28:00 GMT"
          },
          {
            "value": "val2",
            "attributes": "Path=/cat; Domain=example.com; Expires=Wed, 10 Jan 2021
07:28:00 GMT"
          }
        ]
      }
    },

    // Adding the body field is optional and it will not be present in the response
object
```

```
    // unless you specify it in your function.
    // Your function does not have access to the original body returned by the
 CloudFront
    // cache or origin.
    // If you don't specify the body field in your viewer response function, the
 original
    // body returned by the CloudFront cache or origin is returned to viewer.

     "body": {
      "encoding": "text",
      "data": "<!DOCTYPE html><html><body><p>Here is your custom content.</p></body></
html>"
    }
  }
}
```

Beispiel für Ereignisobjekt

Das folgende Beispiel zeigt ein vollständiges `event`-Objekt: Dies ist ein Beispielaufruf für eine Standardverteilung und nicht für eine Mehrmandantenverteilung. Bei Verteilungen mit mehreren Mandanten wird das `endpoint` Feld anstelle von verwendet. Der Wert von `distributionDomainName` `endpoint` ist der CloudFront Domänenname (z. B. d111111abcdef8.cloudfront.net) der Verbindungsgruppe, die dem Ereignis zugeordnet ist.

> **Note**
>
> Das `event`-Objekt ist die Eingabe für Ihre Funktion. Ihre Funktion gibt nur das `request`- oder `response`-Objekt zurück, nicht das vollständige `event`-Objekt.

```
{
    "version": "1.0",
    "context": {
        "distributionDomainName": "d111111abcdef8.cloudfront.net",
        "distributionId": "EDFDVBD6EXAMPLE",
        "eventType": "viewer-response",
        "requestId": "EXAMPLEntjQpEXAMPLE_SG5Z-EXAMPLEPmPfEXAMPLEu3EqEXAMPLE=="
    },
    "viewer": {"ip": "198.51.100.11"},
    "request": {
        "method": "GET",
        "uri": "/media/index.mpd",
```

```
        "querystring": {
            "ID": {"value": "42"},
            "Exp": {"value": "1619740800"},
            "TTL": {"value": "1440"},
            "NoValue": {"value": ""},
            "querymv": {
                "value": "val1",
                "multiValue": [
                    {"value": "val1"},
                    {"value": "val2,val3"}
                ]
            }
        },
        "headers": {
            "host": {"value": "video.example.com"},
            "user-agent": {"value": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:83.0)
 Gecko/20100101 Firefox/83.0"},
            "accept": {
                "value": "application/json",
                "multiValue": [
                    {"value": "application/json"},
                    {"value": "application/xml"},
                    {"value": "text/html"}
                ]
            },
            "accept-language": {"value": "en-GB,en;q=0.5"},
            "accept-encoding": {"value": "gzip, deflate, br"},
            "origin": {"value": "https://website.example.com"},
            "referer": {"value": "https://website.example.com/videos/12345678?
action=play"},
            "cloudfront-viewer-country": {"value": "GB"}
        },
        "cookies": {
            "Cookie1": {"value": "value1"},
            "Cookie2": {"value": "value2"},
            "cookie_consent": {"value": "true"},
            "cookiemv": {
                "value": "value3",
                "multiValue": [
                    {"value": "value3"},
                    {"value": "value4"}
                ]
            }
        }
```

```
    },
    "response": {
        "statusCode": 200,
        "statusDescription": "OK",
        "headers": {
            "date": {"value": "Mon, 04 Apr 2021 18:57:56 GMT"},
            "server": {"value": "gunicorn/19.9.0"},
            "access-control-allow-origin": {"value": "*"},
            "access-control-allow-credentials": {"value": "true"},
            "content-type": {"value": "application/json"},
            "content-length": {"value": "701"}
        },
        "cookies": {
            "ID": {
                "value": "id1234",
                "attributes": "Expires=Wed, 05 Apr 2021 07:28:00 GMT"
            },
            "Cookie1": {
                "value": "val1",
                "attributes": "Secure; Path=/; Domain=example.com; Expires=Wed, 05 Apr
 2021 07:28:00 GMT",
                "multiValue": [
                    {
                        "value": "val1",
                        "attributes": "Secure; Path=/; Domain=example.com; Expires=Wed,
 05 Apr 2021 07:28:00 GMT"
                    },
                    {
                        "value": "val2",
                        "attributes": "Path=/cat; Domain=example.com; Expires=Wed, 10
 Jan 2021 07:28:00 GMT"
                    }
                ]
            }
        }
    }
}
```

JavaScript Laufzeitfunktionen für CloudFront Funktionen

Die JavaScript Runtime-Umgebung von CloudFront Functions ist kompatibel mit ECMAScript (ES) Version 5.1 und unterstützt auch einige Funktionen der ES-Versionen 6 bis 12.

Für die meisten up-to-date Funktionen empfehlen wir die Verwendung von JavaScript Runtime 2.0.

Die JavaScript Runtime 2.0-Features weisen im Vergleich zu 1.0 die folgenden Änderungen auf:

- Methoden für das Puffermodul sind verfügbar
- Die folgenden nicht standardmäßigen Prototyp-Methoden für Zeichenfolgen sind nicht verfügbar:
  - `String.prototype.bytesFrom()`
  - `String.prototype.fromBytes()`
  - `String.prototype.fromUTF8()`
  - `String.prototype.toBytes()`
  - `String.prototype.toUTF8()`
- Das kryptografische Modul weist die folgenden Änderungen auf:
  - `hash.digest()`— Der Rückgabetyp wird auf geändert, `Buffer` wenn keine Kodierung angegeben ist
  - `hmac.digest()`— Der Rückgabetyp wird auf geändert, `Buffer` wenn keine Kodierung angegeben wurde
- Weitere Informationen zu zusätzlichen neuen Funktionen finden Sie unterJavaScript Runtime 2.0-Funktionen für CloudFront Funktionen.

Themen

- JavaScript Runtime 1.0-Funktionen für CloudFront Funktionen
- JavaScript Runtime 2.0-Funktionen für CloudFront Funktionen

JavaScript Runtime 1.0-Funktionen für CloudFront Funktionen

Die JavaScript Runtime-Umgebung von CloudFront Functions ist kompatibel mit ECMAScript (ES) Version 5.1 und unterstützt auch einige Funktionen der ES-Versionen 6 bis 9. Sie enthält auch einige nicht standardmäßige Methoden, die nicht Teil der ES-Spezifikationen sind.

In den folgenden Themen werden alle unterstützten Sprachfunktionen aufgeführt.

Themen

- Kernfunktionen
- Primitive Objekte
- Integrierte Objekte

- Fehlertypen
- Globale
- Integrierten Module
- Eingeschränkte Funktionen

Kernfunktionen

Die folgenden Kern-Features von ES werden unterstützt.

Typen

Alle ES 5.1-Typen werden unterstützt. Dies umfasst boolesche Werte, Zahlen, Zeichenfolgen, Objekte, Arrays, Funktionen, Funktionskonstruktoren und reguläre Ausdrücke.

Operatoren

Alle ES 5.1-Operatoren werden unterstützt.

Der Potenzierungsoperator (`**`) wird unterstützt.

Anweisungen

> **Note**
>
> `const`- und `let`-Anweisungen werden nicht unterstützt.

Die folgenden ES 5.1-Anweisungen werden unterstützt:

- `break`
- `catch`
- `continue`
- `do-while`
- `else`
- `finally`
- `for`
- `for-in`
- `if`

- `return`
- `switch`
- `throw`
- `try`
- `var`
- `while`
- Bezeichnete Anweisungen

Literale

ES 6-Vorlagenliterale werden unterstützt: mehrzeilige Zeichenfolgen, Interpolation von Ausdrücken und Verschachtelungsvorlagen.

Funktionen

Alle Features von ES 5.1 werden unterstützt.

ES 6-Pfeilfunktionen werden unterstützt, und die ES 6 Rest-Parametersyntax wird unterstützt.

Unicode

Quelltext und Zeichenfolgenliterale können Unicode-codierte Zeichen enthalten. Unicode-Code-Punkt-Escape-Sequenzen von sechs Zeichen (z. B. `\uXXXX`) werden ebenfalls unterstützt.

Strikter Modus

Funktionen arbeiten standardmäßig im strikten Modus, sodass Sie Ihrem Funktionscode keine `use strict`-Anweisung hinzufügen müssen. Dies können nicht geändert werden.

Primitive Objekte

Die folgenden primitiven Objekte von ES werden unterstützt.

Objekt

Die folgenden ES 5.1-Methoden für Objekte werden unterstützt:

- `create` (ohne Eigenschaftenliste)
- `defineProperties`
- `defineProperty`
- `freeze`

- `getOwnPropertyDescriptor`
- `getOwnPropertyNames`
- `getPrototypeOf`
- `hasOwnProperty`
- `isExtensible`
- `isFrozen`
- `prototype.isPrototypeOf`
- `isSealed`
- `keys`
- `preventExtensions`
- `prototype.propertyIsEnumerable`
- `seal`
- `prototype.toString`
- `prototype.valueOf`

Die folgenden ES 6-Methoden für Objekte werden unterstützt:

- `assign`
- `is`
- `prototype.setPrototypeOf`

Die folgenden ES 8-Methoden für Objekte werden unterstützt:

- `entries`
- `values`

Zeichenfolge

Die folgenden ES 5.1-Methoden für Zeichenfolgen werden unterstützt:

- `fromCharCode`
- `prototype.charAt`
- `prototype.concat`
- `prototype.indexOf`
- `prototype.lastIndexOf`
- `prototype.match`

- `prototype.replace`
- `prototype.search`
- `prototype.slice`
- `prototype.split`
- `prototype.substr`
- `prototype.substring`
- `prototype.toLowerCase`
- `prototype.trim`
- `prototype.toUpperCase`

Die folgenden ES 6-Methoden für Zeichenfolgen werden unterstützt:

- `fromCodePoint`
- `prototype.codePointAt`
- `prototype.endsWith`
- `prototype.includes`
- `prototype.repeat`
- `prototype.startsWith`

Die folgenden ES 8-Methoden für Zeichenfolgen werden unterstützt:

- `prototype.padStart`
- `prototype.padEnd`

Die folgenden ES 9-Methoden für Zeichenfolgen werden unterstützt:

- `prototype.trimStart`
- `prototype.trimEnd`

Die folgenden nicht standardmäßigen Methoden für Zeichenfolgen werden unterstützt:

- `prototype.bytesFrom(`*`array | string, encoding`*`)`

  Erstellt eine Bytezeichenfolge aus einem Array von Oktetten oder einer codierten Zeichenfolge. Die Optionen für die Zeichenfolgencodierung sind `hex`, `base64` und `base64url`.

- `prototype.fromBytes(`*`start`*`[, `*`end`*`])`

  Erstellt eine Unicode-Zeichenfolge aus einer Bytezeichenfolge, in der jedes Byte durch den entsprechenden Unicode-Codepunkt ersetzt wird.

- `prototype.fromUTF8(start[, end])`

  Erstellt eine Unicode-Zeichenfolge aus einer UTF-8-codierten Bytezeichenfolge. Wenn die Codierung falsch ist, wird `null` zurückgegeben.

- `prototype.toBytes(start[, end])`

  Erstellt eine Bytezeichenfolge aus einer Unicode-Zeichenfolge. Alle Zeichen müssen im Bereich von [0,255] liegen. Wenn nicht, wird `null` zurückgegeben.

- `prototype.toUTF8(start[, end])`

  Erstellt eine UTF-8-codierte Bytezeichenfolge aus einer Unicode-Zeichenfolge.

Nummer

Alle ES 5.1-Methoden für Zahlen werden unterstützt.

Die folgenden ES 6-Methoden für Zahlen werden unterstützt:

- `isFinite`
- `isInteger`
- `isNaN`
- `isSafeInteger`
- `parseFloat`
- `parseInt`
- `prototype.toExponential`
- `prototype.toFixed`
- `prototype.toPrecision`
- `EPSILON`
- `MAX_SAFE_INTEGER`
- `MAX_VALUE`
- `MIN_SAFE_INTEGER`
- `MIN_VALUE`
- `NEGATIVE_INFINITY`
- `NaN`
- `POSITIVE_INFINITY`

Integrierte Objekte

Die folgenden integrierten Objekte von ES werden unterstützt.

Math-Knoten

Alle Mathematikmethoden von ES 5.1 werden unterstützt.

> **Note**
>
> In der Runtime-Umgebung von CloudFront Functions verwendet die `Math.random()` Implementierung OpenBSD`arc4random`, das mit dem Zeitstempel versehen ist, wann die Funktion ausgeführt wird.

Die folgenden ES 6-Mathematikmethoden werden unterstützt:

- `acosh`
- `asinh`
- `atanh`
- `cbrt`
- `clz32`
- `cosh`
- `expm1`
- `fround`
- `hypot`
- `imul`
- `log10`
- `log1p`
- `log2`
- `sign`
- `sinh`
- `tanh`
- `trunc`
- `E`

- `LN10`
- `LN2`
- `LOG10E`
- `LOG2E`
- `PI`
- `SQRT1_2`
- `SQRT2`

Datum

Alle `Date`-Features von ES 5.1 werden unterstützt.

> **Note**
>
> Aus Sicherheitsgründen gibt `Date` immer den gleichen Wert – die Startzeit der Funktion – während der Lebensdauer einer einzelnen Funktionsausführung zurück. Weitere Informationen finden Sie unter Eingeschränkte Funktionen.

Funktion

Die Methoden `apply`, `bind` und `call` werden unterstützt.

Funktionskonstruktoren werden nicht unterstützt.

Reguläre Ausdrücke

Alle Features für reguläre Ausdrücke von ES 5.1 werden unterstützt. Die Sprache für reguläre Ausdrücke ist Perl-kompatibel. ES 9 benannte Aufnahmegruppen werden unterstützt.

JSON

Alle Funktionen von ES 5.1 JSON werden unterstützt, einschließlich `parse` und `stringify`.

Array

Die folgenden ES 5.1-Methoden für Arrays werden unterstützt:

- `isArray`
- `prototype.concat`
- `prototype.every`

- `prototype.filter`
- `prototype.forEach`
- `prototype.indexOf`
- `prototype.join`
- `prototype.lastIndexOf`
- `prototype.map`
- `prototype.pop`
- `prototype.push`
- `prototype.reduce`
- `prototype.reduceRight`
- `prototype.reverse`
- `prototype.shift`
- `prototype.slice`
- `prototype.some`
- `prototype.sort`
- `prototype.splice`
- `prototype.unshift`

Die folgenden ES 6-Methoden für Arrays werden unterstützt:

- `of`
- `prototype.copyWithin`
- `prototype.fill`
- `prototype.find`
- `prototype.findIndex`

Die folgenden ES 7-Methoden für Arrays werden unterstützt:

- `prototype.includes`

Eingegebene Arrays

Die folgenden von ES 6 eingegebenen Arrays werden unterstützt:

- `Int8Array`
- `Uint8Array`

- `Uint8ClampedArray`
- `Int16Array`
- `Uint16Array`
- `Int32Array`
- `Uint32Array`
- `Float32Array`
- `Float64Array`
- `prototype.copyWithin`
- `prototype.fill`
- `prototype.join`
- `prototype.set`
- `prototype.slice`
- `prototype.subarray`
- `prototype.toString`

ArrayBuffer

Die folgenden Methoden für `ArrayBuffer` werden unterstützt:

- `prototype.isView`
- `prototype.slice`

Promise

Die folgenden Methoden für Versprechen werden unterstützt:

- `reject`
- `resolve`
- `prototype.catch`
- `prototype.finally`
- `prototype.then`

Crypto

Das kryptografische Modul bietet standardmäßige Hashing- und Hash-basierte HMAC-Helfer (Message Authentication Code). Sie können das Modul mit `require('crypto')` laden. Das Modul stellt die folgenden Methoden bereit, die sich genau wie ihre Gegenstücke von Node.js verhalten:

- `createHash(algorithm)`
- `hash.update(data)`
- `hash.digest([encoding])`
- `createHmac(algorithm, secret key)`
- `hmac.update(data)`
- `hmac.digest([encoding])`

Weitere Informationen finden Sie unter Krypto (Hash und HMAC) im Abschnitt über integrierte Module.

Konsole

Dies ist ein Hilfsobjekt zum Debuggen. Es unterstützt nur die `log()`-Methode, um Protokollnachrichten aufzuzeichnen.

> **Note**
>
> CloudFront Functions unterstützt keine Kommasyntax, wie zum Beispiel. `console.log('a', 'b')` Verwenden Sie stattdessen das `console.log('a' + ' ' + 'b')` Format.

Fehlertypen

Die folgenden Fehlerobjekte werden unterstützt:

- `Error`
- `EvalError`
- `InternalError`
- `MemoryError`
- `RangeError`
- `ReferenceError`
- `SyntaxError`
- `TypeError`
- `URIError`

Globale

Das `globalThis`-Objekt wird unterstützt.

Die folgenden globalen Funktionen von ES 5.1 werden unterstützt:

- `decodeURI`
- `decodeURIComponent`
- `encodeURI`
- `encodeURIComponent`
- `isFinite`
- `isNaN`
- `parseFloat`
- `parseInt`

Die folgenden globalen Konstanten werden unterstützt:

- `NaN`
- `Infinity`
- `undefined`

Integrierten Module

Die folgenden integrierten Module werden unterstützt.

Module

- Krypto (Hash und HMAC)
- Abfragezeichenfolge

Krypto (Hash und HMAC)

Das kryptographische Modul (`crypto`) bietet standardmäßige Hashing- und HMAC-Helfer (Hash-basierter Nachrichtenauthentifizierungscode). Sie können das Modul mit `require('crypto')` laden. Das Modul bietet die folgenden Methoden, die sich genau wie ihre Gegenstücke von Node.js verhalten.

Hashing-Methoden

`crypto.createHash(algorithm)`

Erstellt und gibt ein Hash-Objekt zurück, mit dem Sie Hash-Digests mit dem angegebenen Algorithmus generieren können: `md5`, `sha1` oder `sha256`.

`hash.update(data)`

Aktualisiert den Hash-Inhalt mit dem angegebenen `data`.

`hash.digest([encoding])`

Berechnet den Digest aller mit `hash.update()` übergebenen Daten. Die Codierung kann `hex`, `base64` oder `base64url` sein.

HMAC-Methoden

`crypto.createHmac(algorithm, secret key)`

Erstellt und gibt ein HMAC-Objekt zurück, das das angegebene `algorithm` und `secret key` verwendet. Der Algorithmus kann `md5`, `sha1` oder `sha256` sein.

`hmac.update(data)`

Aktualisiert den HMAC-Inhalt mit den angegebenen `data`.

`hmac.digest([encoding])`

Berechnet den Digest aller mit `hmac.update()` übergebenen Daten. Die Codierung kann `hex`, `base64` oder `base64url` sein.

Abfragezeichenfolge

> **Note**
>
> Das CloudFront Functions-Ereignisobjekt analysiert automatisch URL-Abfragezeichenfolgen für Sie. Das bedeutet, dass Sie dieses Modul in den meisten Fällen nicht verwenden müssen.

Das Modul für Abfragezeichenfolgen (`querystring`) bietet Methoden zum Analysieren und Formatieren von URL-Abfragezeichenfolgen. Sie können das Modul mit `require('querystring')` laden. Das Modul bietet die folgenden Methoden.

`querystring.escape(`*`string`*`)`

URL-kodiert die angegebene `string` und gibt eine entflohene Abfragezeichenfolge zurück. Die Methode wird von `querystring.stringify()` verwendet und sollte nicht direkt verwendet werden.

`querystring.parse(`*`string`*`[, `*`separator`*`[, `*`equal`*`[, `*`options`*`]]])`

Analysiert eine Abfragezeichenfolge (`string`) und gibt ein Objekt zurück.

Der `separator`-Parameter ist eine Teilzeichenfolge zum Abgrenzen von Schlüssel- und Wertepaaren in der Abfragezeichenfolge. Standardmäßig ist dies &.

Der `equal`-Parameter ist eine Teilzeichenfolge zum Abgrenzen von Schlüsseln und Werten in der Abfragezeichenfolge. Standardmäßig ist dies =.

Der `options`-Parameter ist ein Objekt mit den folgenden Schlüsseln:

`decodeURIComponent` *`function`*

Eine Funktion zum Entschlüsseln von prozentkodierten Zeichen in der Abfragezeichenfolge. Standardmäßig ist dies `querystring.unescape()`.

`maxKeys` *`number`*

Die maximale Anzahl der Schlüssel zum Parsen. Standardmäßig ist dies `1000`. Verwenden Sie den Wert `0`, um die Beschränkungen für das Zählen von Schlüsseln aufzuheben.

Standardmäßig wird davon ausgegangen, dass prozentcodierte Zeichen innerhalb der Abfragezeichenfolge die UTF-8-Codierung verwenden. Ungültige UTF-8-Sequenzen werden durch das Ersatzzeichen `U+FFFD` ersetzt.

Zum Beispiel für die folgende Abfragezeichenfolge:

```
'name=value&abc=xyz&abc=123'
```

Der Rückgabewert von `querystring.parse()` ist:

```
{
name: 'value',
abc: ['xyz', '123']
}
```

`querystring.decode()` ist ein Alias für `querystring.parse()`.

`querystring.stringify(`*`object`*`[, `*`separator`*`[, `*`equal`*`[, `*`options`*`]]])`

Serialisiert ein `object` und gibt eine Abfragezeichenfolge zurück.

Der `separator`-Parameter ist eine Teilzeichenfolge zum Abgrenzen von Schlüssel- und Wertepaaren in der Abfragezeichenfolge. Standardmäßig ist dies &.

Der `equal`-Parameter ist eine Teilzeichenfolge zum Abgrenzen von Schlüsseln und Werten in der Abfragezeichenfolge. Standardmäßig ist dies =.

Der `options`-Parameter ist ein Objekt mit den folgenden Schlüsseln:

`encodeURIComponent` *`function`*

Die Funktion, die zum Konvertieren von URL-unsicheren Zeichen in die prozentuale Kodierung in der Abfragezeichenfolge verwendet wird. Standardmäßig ist dies `querystring.escape()`.

Standardmäßig werden Zeichen, die eine prozentuale Kodierung innerhalb der Abfragezeichenfolge erfordern, als UTF-8 codiert. Um eine andere Codierung zu verwenden, geben Sie die Option `encodeURIComponent` an.

Zum Beispiel für den folgenden Code:

```
querystring.stringify({ name: 'value', abc: ['xyz', '123'], anotherName: '' });
```

Der Rückgabewert ist:

```
'name=value&abc=xyz&abc=123&anotherName='
```

`querystring.encode()` ist ein Alias für `querystring.stringify()`.

`querystring.unescape(`*`string`*`)`

Dekodiert die prozentualen Zeichen der URL in der angegebenen `string` und gibt eine nicht entdeckene Abfragezeichenfolge zurück. Diese Methode wird von `querystring.parse()` verwendet und sollte nicht direkt verwendet werden.

Eingeschränkte Funktionen

Die folgenden JavaScript Sprachfunktionen werden entweder nicht unterstützt oder sind aus Sicherheitsgründen eingeschränkt.

Dynamische Codeauswertung

Die dynamische Codeauswertung wird nicht unterstützt. Sowohl `eval()`- als auch `Function`-Konstruktoren geben einen Fehler aus, wenn sie versucht werden. Zum Beispiel gibt `const sum = new Function('a', 'b', 'return a + b')` einen Fehler aus.

Timer

Die Funktionen `setTimeout()`, `setImmediate()` und `clearTimeout()` werden nicht unterstützt. Es gibt keine Bestimmung, innerhalb einer Funktionsausführung zu verschieben oder zu ergeben. Ihre Funktion muss synchron bis zum Abschluss ausgeführt werden.

Datum und Zeitstempel

Aus Sicherheitsgründen besteht kein Zugang zu hochauflösenden Timern. Alle `Date`-Methoden zum Abfragen der aktuellen Uhrzeit geben während der Lebensdauer einer einzelnen Funktionsausführung immer den gleichen Wert zurück. Der zurückgegebene Zeitstempel ist die Zeit, zu der die Funktion gestartet wurde. Folglich können Sie die verstrichene Zeit in Ihrer Funktion nicht messen.

Zugriff auf das Dateisystem

Es gibt keinen Zugriff auf das Dateisystem. Zum Beispiel gibt es kein `fs`-Modul für den Dateisystemzugriff wie in Node.js.

Zugriff verarbeiten

Es gibt keinen Prozesszugriff. Beispielsweise gibt es kein `process` globales Objekt für die Verarbeitung von Informationszugriffen wie in Node.js.

Umgebungsvariablen

Es gibt keinen Zugriff auf Umgebungsvariablen.

Stattdessen können Sie CloudFront KeyValueStore damit einen zentralen Datenspeicher mit Schlüssel-Wert-Paaren für Ihre Funktionen erstellen. CloudFront CloudFront KeyValueStore ermöglicht dynamische Aktualisierungen Ihrer Konfigurationsdaten, ohne dass Codeänderungen vorgenommen werden müssen. Sie müssen JavaScript Runtime 2.0 verwenden, um es zu verwenden CloudFront KeyValueStore. Weitere Informationen finden Sie unter Amazon CloudFront KeyValueStore.

Netzwerkzugriff

Es gibt keine Unterstützung für Netzwerkaufrufe. Zum Beispiel werden XHR, HTTP(S) und Socket nicht unterstützt.

JavaScript Runtime 2.0-Funktionen für CloudFront Funktionen

Die JavaScript Runtime-Umgebung von CloudFront Functions ist kompatibel mit ECMAScript (ES) Version 5.1 und unterstützt auch einige Funktionen der ES-Versionen 6 bis 12. Sie enthält auch einige nicht standardmäßige Methoden, die nicht Teil der ES-Spezifikationen sind. In den folgenden Themen werden alle unterstützten Features dieser Laufzeit aufgeführt.

Themen

Kern-Features

Die folgenden Kern-Features von ES werden unterstützt.

Typen

Alle ES 5.1-Typen werden unterstützt. Dies umfasst boolesche Werte, Zahlen, Zeichenfolgen, Objekte, Arrays, Funktionen und reguläre Ausdrücke.

Operatoren

Alle ES 5.1-Operatoren werden unterstützt.

Der Potenzierungsoperator (`**`) wird unterstützt.

Anweisungen

Die folgenden ES 5.1-Anweisungen werden unterstützt:

- `break`
- `catch`
- `continue`
- `do-while`

- `else`
- `finally`
- `for`
- `for-in`
- `if`
- `label`
- `return`
- `switch`
- `throw`
- `try`
- `var`
- `while`

Die folgenden ES 6-Anweisungen werden unterstützt:

- `const`
- `let`

Die folgenden ES 8-Anweisungen werden unterstützt:

- `async`
- `await`

> **Note**
>
> `async`, `awaitconst`, und `let` werden in JavaScript Runtime 2.0 unterstützt. `await`kann nur innerhalb von `async` Funktionen verwendet werden. `async`Argumente und Schließungen werden nicht unterstützt.

Literale

ES 6-Vorlagenliterale werden unterstützt: mehrzeilige Zeichenfolgen, Interpolation von Ausdrücken und Verschachtelungsvorlagen.

Funktionen

Alle Features von ES 5.1 werden unterstützt.

ES 6-Pfeilfunktionen werden unterstützt, und die ES 6 Rest-Parametersyntax wird unterstützt.

Unicode

Quelltext und Zeichenfolgenliterale können Unicode-codierte Zeichen enthalten. Unicode-Code-Punkt-Escape-Sequenzen von sechs Zeichen (z. B. `\uXXXX`) werden ebenfalls unterstützt.

Strikter Modus

Funktionen arbeiten standardmäßig im strikten Modus, sodass Sie Ihrem Funktionscode keine `use strict`-Anweisung hinzufügen müssen. Dies können nicht geändert werden.

Primitive Objekte

Die folgenden primitiven Objekte von ES werden unterstützt.

Objekt

Die folgenden ES 5.1-Methoden für Objekte werden unterstützt:

- `Object.create()` (ohne Eigenschaftenliste)
- `Object.defineProperties()`
- `Object.defineProperty()`
- `Object.freeze()`
- `Object.getOwnPropertyDescriptor()`
- `Object.getOwnPropertyDescriptors()`
- `Object.getOwnPropertyNames()`
- `Object.getPrototypeOf()`
- `Object.isExtensible()`
- `Object.isFrozen()`
- `Object.isSealed()`
- `Object.keys()`
- `Object.preventExtensions()`
- `Object.seal()`

Die folgenden ES 6-Methoden für Objekte werden unterstützt:

- `Object.assign()`

Die folgenden ES 8-Methoden für Objekte werden unterstützt:

- `Object.entries()`
- `Object.values()`

Die folgenden ES-5.1-Prototyp-Methoden für Objekte werden unterstützt:

- `Object.prototype.hasOwnProperty()`
- `Object.prototype.isPrototypeOf()`
- `Object.prototype.propertyIsEnumerable()`
- `Object.prototype.toString()`
- `Object.prototype.valueOf()`

Die folgenden ES 6-Prototyp-Methoden für Objekte werden unterstützt:

- `Object.prototype.is()`
- `Object.prototype.setPrototypeOf()`

String

Die folgenden ES 5.1-Methoden für Zeichenfolgen werden unterstützt:

- `String.fromCharCode()`

Die folgenden ES 6-Methoden für Zeichenfolgen werden unterstützt:

- `String.fromCodePoint()`

Die folgenden ES-5.1-Prototyp-Methoden für Zeichenfolgen werden unterstützt:

- `String.prototype.charAt()`
- `String.prototype.concat()`
- `String.prototype.indexOf()`
- `String.prototype.lastIndexOf()`
- `String.prototype.match()`
- `String.prototype.replace()`
- `String.prototype.search()`
- `String.prototype.slice()`
- `String.prototype.split()`
- `String.prototype.substr()`
- `String.prototype.substring()`

- `String.prototype.toLowerCase()`
- `String.prototype.trim()`
- `String.prototype.toUpperCase()`

Die folgenden ES 6-Prototyp-Methoden für Zeichenfolgen werden unterstützt:

- `String.prototype.codePointAt()`
- `String.prototype.endsWith()`
- `String.prototype.includes()`
- `String.prototype.repeat()`
- `String.prototype.startsWith()`

Die folgenden ES 8-Prototyp-Methoden für Zeichenfolgen werden unterstützt:

- `String.prototype.padStart()`
- `String.prototype.padEnd()`

Die folgenden ES 9-Prototyp-Methoden für Zeichenfolgen werden unterstützt:

- `String.prototype.trimStart()`
- `String.prototype.trimEnd()`

Die folgenden ES 12-Prototyp-Methoden für Zeichenfolgen werden unterstützt:

- `String.prototype.replaceAll()`

> **Note**
>
> `String.prototype.replaceAll()`ist neu in JavaScript Runtime 2.0.

Anzahl

ALLE ES 5-Zahlen werden unterstützt.

Die folgenden ES 6-Eigenschaften für Zahlen werden unterstützt:

- `Number.EPSILON`
- `Number.MAX_SAFE_INTEGER`
- `Number.MIN_SAFE_INTEGER`
- `Number.MAX_VALUE`
- `Number.MIN_VALUE`

- `Number.NaN`
- `Number.NEGATIVE_INFINITY`
- `Number.POSITIVE_INFINITY`

Die folgenden ES 6-Methoden für Zahlen werden unterstützt:

- `Number.isFinite()`
- `Number.isInteger()`
- `Number.isNaN()`
- `Number.isSafeInteger()`
- `Number.parseInt()`
- `Number.parseFloat()`

Die folgenden ES 5.1-Prototyp-Methoden für Zahlen werden unterstützt:

- `Number.prototype.toExponential()`
- `Number.prototype.toFixed()`
- `Number.prototype.toPrecision()`

Numerische ES 12-Trennzeichen werden unterstützt.

> **Note**
>
> Numerische ES12-Trennzeichen sind neu in JavaScript Runtime 2.0.

Integrierte Objekte

Die folgenden integrierten Objekte von ES werden unterstützt.

Math-Knoten

Alle Mathematikmethoden von ES 5.1 werden unterstützt.

> **Note**
>
> In der Runtime-Umgebung von CloudFront Functions verwendet die `Math.random()` Implementierung OpenBSD`arc4random`, das mit dem Zeitstempel versehen ist, wann die Funktion ausgeführt wird.

Die folgenden ES 6-Mathematikeigenschaften werden unterstützt:

- `Math.E`
- `Math.LN10`
- `Math.LN2`
- `Math.LOG10E`
- `Math.LOG2E`
- `Math.PI`
- `Math.SQRT1_2`
- `Math.SQRT2`

Die folgenden ES 6-Mathematikmethoden werden unterstützt:

- `Math.abs()`
- `Math.acos()`
- `Math.acosh()`
- `Math.asin()`
- `Math.asinh()`
- `Math.atan()`
- `Math.atan2()`
- `Math.atanh()`
- `Math.cbrt()`
- `Math.ceil()`
- `Math.clz32()`
- `Math.cos()`
- `Math.cosh()`
- `Math.exp()`
- `Math.expm1()`
- `Math.floor()`
- `Math.fround()`
- `Math.hypot()`
- `Math.imul()`

- `Math.log()`
- `Math.log1p()`
- `Math.log2()`
- `Math.log10()`
- `Math.max()`
- `Math.min()`
- `Math.pow()`
- `Math.random()`
- `Math.round()`
- `Math.sign()`
- `Math.sinh()`
- `Math.sin()`
- `Math.sqrt()`
- `Math.tan()`
- `Math.tanh()`
- `Math.trunc()`

Datum

Alle `Date`-Features von ES 5.1 werden unterstützt.

> **Note**
>
> Aus Sicherheitsgründen gibt `Date` immer den gleichen Wert – die Startzeit der Funktion – während der Lebensdauer einer einzelnen Funktionsausführung zurück. Weitere Informationen finden Sie unter Eingeschränkte Funktionen.

Funktion

Die folgenden ES-5.1-Prototyp-Methoden werden unterstützt:

- `Function.prototype.apply()`
- `Function.prototype.bind()`

- `Function.prototype.call()`

Funktionskonstruktoren werden nicht unterstützt.

Reguläre Ausdrücke

Alle Features für reguläre Ausdrücke von ES 5.1 werden unterstützt. Die Sprache für reguläre Ausdrücke ist Perl-kompatibel.

Die folgenden ES-5.1-Prototyp-Zugriffseigenschaften werden unterstützt:

- `RegExp.prototype.global`
- `RegExp.prototype.ignoreCase`
- `RegExp.protoype.multiline`
- `RegExp.protoype.source`
- `RegExp.prototype.sticky`
- `RegExp.prototype.flags`

> **Note**
>
> `RegExp.prototype.sticky`und `RegExp.prototype.flags` sind neu in JavaScript Runtime 2.0.

Die folgenden ES-5.1-Prototyp-Methoden werden unterstützt:

- `RegExp.prototype.exec()`
- `RegExp.prototype.test()`
- `RegExp.prototype.toString()`
- `RegExp.prototype[@@replace]()`
- `RegExp.prototype[@@split]()`

> **Note**
>
> `RegExp.prototype[@@split]()`ist neu in JavaScript Runtime 2.0.

Die folgenden ES-5.1-Instance-Eigenschaften werden unterstützt:

- `lastIndex`

ES 9 benannte Aufnahmegruppen werden unterstützt.

JSON

Die folgenden ES 5.1-Mathematikmethoden werden unterstützt:

- `JSON.parse()`
- `JSON.stringify()`

Array

Die folgenden ES 5.1-Methoden für Arrays werden unterstützt:

- `Array.isArray()`

Die folgenden ES 6-Methoden für Arrays werden unterstützt:

- `Array.of()`

Die folgenden ES-5.1-Prototyp-Methoden werden unterstützt:

- `Array.prototype.concat()`
- `Array.prototype.every()`
- `Array.prototype.filter()`
- `Array.prototype.forEach()`
- `Array.prototype.indexOf()`
- `Array.prototype.join()`
- `Array.prototype.lastIndexOf()`
- `Array.prototype.map()`
- `Array.prototype.pop()`
- `Array.prototype.push()`
- `Array.prototype.reduce()`
- `Array.prototype.reduceRight()`
- `Array.prototype.reverse()`
- `Array.prototype.shift()`
- `Array.prototype.slice()`
- `Array.prototype.some()`
- `Array.prototype.sort()`

- `Array.prototype.splice()`
- `Array.prototype.unshift()`

Die folgenden ES 6-Prototyp-Methoden werden unterstützt:

- `Array.prototype.copyWithin()`
- `Array.prototype.fill()`
- `Array.prototype.find()`
- `Array.prototype.findIndex()`

Die folgenden ES 7-Prototyp-Methoden werden unterstützt:

- `Array.prototype.includes()`

Eingegebene Arrays

Die folgenden von ES 6 eingegebenen Array-Konstruktoren werden unterstützt:

- `Float32Array`
- `Float64Array`
- `Int8Array`
- `Int16Array`
- `Int32Array`
- `Uint8Array`
- `Uint8ClampedArray`
- `Uint16Array`
- `Uint32Array`

Die folgenden ES-6-Methoden werden unterstützt:

- `TypedArray.from()`
- `TypedArray.of()`

> **Note**
>
> `TypedArray.from()`und `TypedArray.of()` sind neu in JavaScript Runtime 2.0.

Die folgenden ES 6-Prototyp-Methoden werden unterstützt:

- `TypedArray.prototype.copyWithin()`
- `TypedArray.prototype.every()`
- `TypedArray.prototype.fill()`
- `TypedArray.prototype.filter()`
- `TypedArray.prototype.find()`
- `TypedArray.prototype.findIndex()`
- `TypedArray.prototype.forEach()`
- `TypedArray.prototype.includes()`
- `TypedArray.prototype.indexOf()`
- `TypedArray.prototype.join()`
- `TypedArray.prototype.lastIndexOf()`
- `TypedArray.prototype.map()`
- `TypedArray.prototype.reduce()`
- `TypedArray.prototype.reduceRight()`
- `TypedArray.prototype.reverse()`
- `TypedArray.prototype.some()`
- `TypedArray.prototype.set()`
- `TypedArray.prototype.slice()`
- `TypedArray.prototype.sort()`
- `TypedArray.prototype.subarray()`
- `TypedArray.prototype.toString()`

> **Note**
>
> `TypedArray.prototype.every()`,`TypedArray.prototype.fill()`,`TypedArray.proto`
> `TypedArray.prototype.reduceRight()``TypedArray.prototype.reverse()`,
> und `TypedArray.prototype.some()` sind neu in JavaScript Runtime 2.0.

ArrayBuffer

Die folgenden ES 6-Methoden ArrayBuffer werden unterstützt:

- `isView()`

Die folgenden ES 6-Prototypmethoden ArrayBuffer werden unterstützt:

- `ArrayBuffer.prototype.slice()`

Promise

Die folgenden ES 6-Methoden für Versprechen werden unterstützt:

- `Promise.all()`
- `Promise.allSettled()`
- `Promise.any()`
- `Promise.reject()`
- `Promise.resolve()`
- `Promise.race()`

> **Note**
>
> `Promise.all()`, `Promise.allSettled()Promise.any()`, und `Promise.race()` sind neu in JavaScript Runtime 2.0.

Die folgenden ES 6-Prototyp-Methoden für Versprechen werden unterstützt:

- `Promise.prototype.catch()`
- `Promise.prototype.finally()`
- `Promise.prototype.then()`

DataView

Die folgenden ES 6-Prototyp-Methoden werden unterstützt:

- `DataView.prototype.getFloat32()`
- `DataView.prototype.getFloat64()`
- `DataView.prototype.getInt16()`
- `DataView.prototype.getInt32()`
- `DataView.prototype.getInt8()`
- `DataView.prototype.getUint16()`
- `DataView.prototype.getUint32()`
- `DataView.prototype.getUint8()`

- `DataView.prototype.setFloat32()`
- `DataView.prototype.setFloat64()`
- `DataView.prototype.setInt16()`
- `DataView.prototype.setInt32()`
- `DataView.prototype.setInt8()`
- `DataView.prototype.setUint16()`
- `DataView.prototype.setUint32()`
- `DataView.prototype.setUint8()`

> **Note**
>
> Alle Prototypmethoden von Dataview ES 6 sind neu in JavaScript Runtime 2.0.

Symbol

Die folgenden ES-6-Methoden werden unterstützt:

- `Symbol.for()`
- `Symbol.keyfor()`

> **Note**
>
> Alle Symbol ES 6-Methoden sind neu in JavaScript Runtime 2.0.

Textdecoder

Die folgenden Prototyp-Methoden werden unterstützt:

- `TextDecoder.prototype.decode()`

Die folgenden Prototyp-Zugriffseigenschaften werden unterstützt:

- `TextDecoder.prototype.encoding`
- `TextDecoder.prototype.fatal`
- `TextDecoder.prototype.ignoreBOM`

Text-Encoder

Die folgenden Prototyp-Methoden werden unterstützt:

- `TextEncoder.prototype.encode()`
- `TextEncoder.prototype.encodeInto()`

Fehlertypen

Die folgenden Fehlerobjekte werden unterstützt:

- `Error`
- `EvalError`
- `InternalError`
- `RangeError`
- `ReferenceError`
- `SyntaxError`
- `TypeError`
- `URIError`

Globale

Das `globalThis`-Objekt wird unterstützt.

Die folgenden globalen Funktionen von ES 5.1 werden unterstützt:

- `decodeURI()`
- `decodeURIComponent()`
- `encodeURI()`
- `encodeURIComponent()`
- `isFinite()`
- `isNaN()`
- `parseFloat()`
- `parseInt()`

Die folgenden globalen Funktionen von ES 6 werden unterstützt:

- `atob()`

- `btoa()`

> **Note**
>
> `atob()`und `btoa()` sind neu in JavaScript Runtime 2.0.

Die folgenden globalen Konstanten werden unterstützt:

- `NaN`
- `Infinity`
- `undefined`
- `arguments`

Integrierten Module

Die folgenden integrierten Module werden unterstützt.

Module

- Buffer
- Abfragezeichenfolge
- Crypto

Buffer

Das Modul bietet die folgenden Methoden:

- `Buffer.alloc(size[, fill[, encoding]])`

  Weisen Sie einen `Buffer` zu.

  - `size`: Puffergröße. Geben Sie eine Ganzzahl ein.
  - `fill`: Optional. Geben Sie eine Zeichenfolge, `Buffer`, Uint8Array oder eine Ganzzahl ein. Der Standardwert ist `0`.
  - `encoding`: Optional. Wenn `fill` eine Zeichenfolge ist, geben Sie eine der folgenden Optionen ein: `utf8`, `hex`, `base64`, `base64url`. Der Standardwert ist `utf8`.
- `Buffer.allocUnsafe(size)`

Weisen Sie einen nicht initialisierten `Buffer` zu.

- `size`: Geben Sie eine Ganzzahl ein.

- `Buffer.byteLength(value[, encoding])`

  Gibt die Länge eines Werts in Byte zurück.

  - `value`: Eine Zeichenfolge,, `Buffer` TypedArray, Dataview oder Arraybuffer.
  - `encoding`: Optional. Wenn `value` eine Zeichenfolge ist, geben Sie eine der folgenden Optionen ein: `utf8`, `hex`, `base64`, `base64url`. Der Standardwert ist `utf8`.

- `Buffer.compare(buffer1, buffer2)`

  Vergleichen Sie zwei `Buffer`, um Arrays besser sortieren zu können. Gibt `0` zurück, wenn sie identisch sind, `-1`, wenn `buffer1` an erster Stelle steht, oder `1`, wenn `buffer2` an erster Stelle steht.

  - `buffer1`: Geben Sie einen `Buffer` ein.
  - `buffer2`: Geben Sie einen anderen `Buffer` ein.

- `Buffer.concat(list[, totalLength])`

  Verketten Sie mehrere `Buffer`. Gibt `0` zurück, wenn keiner vorhanden ist. Gibt bis zu `totalLength` zurück.

  - `list`: Geben Sie eine Liste von `Buffern` ein. Beachten Sie, dass dies auf `totalLength` gekürzt wird.
  - `totalLength`: Optional. Geben Sie eine Ganzzahl ohne Vorzeichen ein. Wenn das Feld leer ist, wird die Summe der `Buffer`-Instances in der Liste verwendet.

- `Buffer.from(array)`

  Erstellen Sie einen `Buffer` aus einem Array.

  - `array`: Geben Sie ein Byte-Array von `0` bis `255` ein.

- `Buffer.from(arrayBuffer, byteOffset[, length]))`

  Erstellen Sie eine Ansicht von `arrayBuffer`, beginnend beim Versatz `byteOffset` mit der Länge `length`.

  - `arrayBuffer`: Geben Sie ein `Buffer`-Array ein.
  - `byteOffset`: Geben Sie eine Ganzzahl ein.
  - `length`: Optional. Geben Sie eine Ganzzahl ein.

- `Buffer.from(buffer)`

  Erstellen Sie eine Kopie des `Buffers`.

  - `buffer`: Geben Sie einen `Buffer` ein.
- `Buffer.from(object[, offsetOrEncoding[, length]])`

  Erstellen Sie einen `Buffer` aus einem Objekt. Gibt `Buffer.from(object.valueOf(), offsetOrEncoding, length)` zurück, wenn `valueOf()` nicht dem Objekt entspricht.

  - `object`: Geben Sie ein Objekt ein.
  - `offsetOrEncoding`: Optional. Geben Sie eine Ganzzahl oder eine Kodierungszeichenfolge ein.
  - `length`: Optional. Geben Sie eine Ganzzahl ein.
- `Buffer.from(string[, encoding])`

  Erstellen Sie einen `Buffer` aus einer Zeichenfolge.

  - `string`: Geben Sie eine Zeichenfolge ein.
  - `encoding`: Optional. Machen Sie eine der folgenden Eingaben: `utf8`, `hex`, `base64`, `base64url`. Der Standardwert ist `utf8`.
- `Buffer.isBuffer(object)`

  Prüfen Sie, ob `object` ein Puffer ist. Gibt `true` oder `false` zurück.

  - `object`: Geben Sie ein Objekt ein.
- `Buffer.isEncoding(encoding)`

  Prüfen Sie, ob `encoding` unterstützt wird. Gibt `true` oder `false` zurück.

  - `encoding`: Optional. Machen Sie eine der folgenden Eingaben: `utf8`, `hex`, `base64`, `base64url`. Der Standardwert ist `utf8`.

Das Modul bietet die folgenden Puffer-Prototyp-Methoden:

- `Buffer.prototype.compare(target[, targetStart[, targetEnd[, sourceStart[, sourceEnd]]]])`

  Vergleichen Sie `Buffer` mit dem Ziel. Gibt `0` zurück, wenn sie identisch sind, `1`, wenn `buffer` an erster Stelle steht, oder `-1`, wenn `target` an erster Stelle steht.

  - `target`: Geben Sie einen `Buffer` ein.

  - `targetStart`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
  - `targetEnd`: Optional. Geben Sie eine Ganzzahl ein. Die Standardeinstellung ist die `target`-Länge.
  - `sourceStart`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
  - `sourceEnd`: Optional. Geben Sie eine Ganzzahl ein. Die Standardeinstellung ist die `Buffer`-Länge.
- `Buffer.prototype.copy(target[, targetStart[, sourceStart[, sourceEnd]]])`

  Kopieren Sie den Puffer nach `target`.
  - `target`: Geben Sie einen `Buffer` oder `Uint8Array` ein.
  - `targetStart`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
  - `sourceStart`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
  - `sourceEnd`: Optional. Geben Sie eine Ganzzahl ein. Die Standardeinstellung ist die `Buffer`-Länge.
- `Buffer.prototype.equals(otherBuffer)`

  Vergleichen Sie `Buffer` mit `otherBuffer`. Gibt `true` oder `false` zurück.
  - `otherBuffer`: Geben Sie eine Zeichenfolge ein.
- `Buffer.prototype.fill(value[, offset[, end][, encoding])`

  Geben Sie für den `Buffer` den Wert `value` ein.
  - `value`: Geben Sie eine Zeichenfolge, einen `Buffer` oder eine Ganzzahl ein.
  - `offset`: Optional. Geben Sie eine Ganzzahl ein.
  - `end`: Optional. Geben Sie eine Ganzzahl ein.
  - `encoding`: Optional. Machen Sie eine der folgenden Eingaben: `utf8`, `hex`, `base64`, `base64url`. Der Standardwert ist `utf8`.
- `Buffer.prototype.includes(value[, byteOffset][, encoding])`

  Suchen Sie nach `value` im `Buffer`. Gibt `true` oder `false` zurück.
  - `value`: Geben Sie eine Zeichenfolge, einen `Buffer`, ein `Uint8Array` oder eine Ganzzahl ein.
  - `byteOffset`: Optional. Geben Sie eine Ganzzahl ein.
  - `encoding`: Optional. Machen Sie eine der folgenden Eingaben: `utf8`, `hex`, `base64`, `base64url`. Der Standardwert ist `utf8`.

- `Buffer.prototype.indexOf(value[, byteOffset][, encoding])`

  Suchen Sie nach dem ersten `value` im `Buffer`. Gibt `index` zurück, wenn er gefunden wurde, oder gibt `-1` zurück, wenn er nicht gefunden wurde.

  - `value`: Geben Sie eine Zeichenfolge, einen `Buffer`, ein Unit8Array oder eine Ganzzahl von 0 bis 255 ein.
  - `byteOffset`: Optional. Geben Sie eine Ganzzahl ein.
  - `encoding`: Optional. Geben Sie eine der folgenden Optionen ein, wenn `value` eine Zeichenfolge ist: `utf8`, `hex`, `base64`, `base64url`. Der Standardwert ist `utf8`.
- `Buffer.prototype.lastIndexOf(value[, byteOffset][, encoding])`

  Suchen Sie nach dem letzten `value` im `Buffer`. Gibt `index` zurück, wenn er gefunden wurde, oder gibt `-1` zurück, wenn er nicht gefunden wurde.

  - `value`: Geben Sie eine Zeichenfolge, einen `Buffer`, ein Unit8Array oder eine Ganzzahl von 0 bis 255 ein.
  - `byteOffset`: Optional. Geben Sie eine Ganzzahl ein.
  - `encoding`: Optional. Geben Sie eine der folgenden Optionen ein, wenn `value` eine Zeichenfolge ist: `utf8`, `hex`, `base64`, `base64url`. Der Standardwert ist `utf8`.
- `Buffer.prototype.readInt8(offset)`

  Lesen Sie `Int8` beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readIntBE(offset, byteLength)`

  Lesen Sie `Int` als Big-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
  - `byteLength`: Optional. Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.readInt16BE(offset)`

  Lesen Sie `Int16` als Big-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readInt32BE(offset)`

  Lesen Sie `Int32` als Big-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.

- `Buffer.prototype.readIntLE(offset, byteLength)`

  Lesen Sie `Int` als Little-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.readInt16LE(offset)`

  Lesen Sie `Int16` als Little-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readInt32LE(offset)`

  Lesen Sie `Int32` als Little-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readUInt8(offset)`

  Lesen Sie `UInt8` beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readUIntBE(offset, byteLength)`

  Lesen Sie `UInt` als Big-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.readUInt16BE(offset)`

  Lesen Sie `UInt16` als Big-Endian beim `offset` vom `Buffer`.

- • `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readUInt32BE(offset)`

  Lesen Sie `UInt32` als Big-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readUIntLE(offset, byteLength)`

  Lesen Sie `UInt` als Little-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.

- `Buffer.prototype.readUInt16LE(offset)`

  Lesen Sie `UInt16` als Little-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readUInt32LE(offset)`

  Lesen Sie `UInt32` als Little-Endian beim `offset` vom `Buffer`.

  - `offset`: Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readDoubleBE([offset])`

  Lesen Sie einen 64-Bit-Double-Wert als Big-Endian beim `offset` vom `Buffer`.

  - `offset`: Optional. Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readDoubleLE([offset])`

  Lesen Sie einen 64-Bit-Double-Wert als Little-Endian beim `offset` vom `Buffer`.

  - `offset`: Optional. Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readFloatBE([offset])`

  Lesen Sie einen 32-Bit-Float-Wert als Big-Endian beim `offset` vom `Buffer`.

  - `offset`: Optional. Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.readFloatLE([offset])`

  Lesen Sie einen 32-Bit-Float-Wert als Little-Endian beim `offset` vom `Buffer`.

  - `offset`: Optional. Geben Sie eine Ganzzahl ein.
- `Buffer.prototype.subarray([start[, end]])`

  Gibt eine Kopie vom `Buffer` zurück, der versetzt und mit einem neuen `start` und `end` zugeschnitten wurde.

  - `start`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
  - `end`: Optional. Geben Sie eine Ganzzahl ein. Die Standardeinstellung ist die Pufferlänge.
- `Buffer.prototype.swap16()`

  Tauschen Sie die Byte-Reihenfolge des `Buffer`-Arrays aus und behandeln Sie es wie ein Array von 16-Bit-Zahlen. Die `Buffer`-Länge muss durch 2 teilbar sein, sonst erhalten Sie eine Fehlermeldung.
- `Buffer.prototype.swap32()`

Tauschen Sie die Byte-Reihenfolge des `Buffer`-Arrays aus und behandeln Sie es wie ein Array von 32-Bit-Zahlen. Die `Buffer`-Länge muss durch 4 teilbar sein, sonst erhalten Sie eine Fehlermeldung.

- `Buffer.prototype.swap64()`

  Tauschen Sie die Byte-Reihenfolge des `Buffer`-Arrays aus und behandeln Sie es wie ein Array von 64-Bit-Zahlen. Die `Buffer`-Länge muss durch 8 teilbar sein, sonst erhalten Sie eine Fehlermeldung.

- `Buffer.prototype.toJSON()`

  Gibt `Buffer` als JSON zurück.

- `Buffer.prototype.toString([encoding[, start[, end]]])`

  Konvertieren Sie den `Buffer` von `start` bis `end` in eine kodierte Zeichenfolge.

  - `encoding`: Optional. Machen Sie eine der folgenden Eingaben: `utf8`, `hex`, `base64` oder `base64url`. Der Standardwert ist `utf8`.
  - `start`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
  - `end`: Optional. Geben Sie eine Ganzzahl ein. Die Standardeinstellung ist die Pufferlänge.

- `Buffer.prototype.write(string[, offset[, length]][, encoding])`

  Schreiben Sie die codierte `string` in den `Buffer`, wenn genügend Platz vorhanden ist, oder die gekürzte `string`, wenn nicht genügend Platz vorhanden ist.

  - `string`: Geben Sie eine Zeichenfolge ein.
  - `offset`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
  - `length`: Optional. Geben Sie eine Ganzzahl ein. Die Standardeinstellung ist die Länge der Zeichenfolge.
  - `encoding`: Optional. Geben Sie optional eine der folgenden Optionen ein: `utf8`, `hex`, `base64` oder `base64url`. Der Standardwert ist `utf8`.

- `Buffer.prototype.writeInt8(value, offset, byteLength)`

  Schreiben Sie den `Int8-value` der `byteLength` beim `offset` des `Buffers`.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Geben Sie eine Ganzzahl ein
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.

- `Buffer.prototype.writeIntBE(value, offset, byteLength)`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Big-Endian.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Geben Sie eine Ganzzahl ein
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeInt16BE(value, offset, byteLength)`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Big-Endian.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Geben Sie eine Ganzzahl ein
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeInt32BE(value, offset, byteLength)`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Big-Endian.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Geben Sie eine Ganzzahl ein
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeIntLE(offset, byteLength)`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Little-Endian.

  - `offset`: Geben Sie eine Ganzzahl ein.
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeInt16LE(offset, byteLength)`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Little-Endian.

  - `offset`: Geben Sie eine Ganzzahl ein.
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeInt32LE(offset, byteLength)`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Little-Endian.

  - `offset`: Geben Sie eine Ganzzahl ein.
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeUInt8(value, offset, byteLength)`

Schreiben Sie den UInt8-value der byteLength beim offset des Buffers.

- value: Geben Sie eine Ganzzahl ein.
- offset: Geben Sie eine Ganzzahl ein
- byteLength: Geben Sie eine Ganzzahl von 1 bis 6 ein.

- `Buffer.prototype.writeUIntBE(value, offset, byteLength)`

  Schreiben Sie den value beim offset des Buffers mit Big-Endian.

  - value: Geben Sie eine Ganzzahl ein.
  - offset: Geben Sie eine Ganzzahl ein
  - byteLength: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeUInt16BE(value, offset, byteLength)`

  Schreiben Sie den value beim offset des Buffers mit Big-Endian.

  - value: Geben Sie eine Ganzzahl ein.
  - offset: Geben Sie eine Ganzzahl ein
  - byteLength: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeUInt32BE(value, offset, byteLength)`

  Schreiben Sie den value beim offset des Buffers mit Big-Endian.

  - value: Geben Sie eine Ganzzahl ein.
  - offset: Geben Sie eine Ganzzahl ein
  - byteLength: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeUIntLE(value, offset, byteLength)`

  Schreiben Sie den value beim offset des Buffers mit Little-Endian.

  - value: Geben Sie eine Ganzzahl ein.
  - offset: Geben Sie eine Ganzzahl ein
  - byteLength: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeUInt16LE(value, offset, byteLength)`

  Schreiben Sie den value beim offset des Buffers mit Little-Endian.

  - value: Geben Sie eine Ganzzahl ein.
  - offset: Geben Sie eine Ganzzahl ein

  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeUInt32LE(value, offset, byteLength)`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Little-Endian.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Geben Sie eine Ganzzahl ein
  - `byteLength`: Geben Sie eine Ganzzahl von 1 bis 6 ein.
- `Buffer.prototype.writeDoubleBE(value, [offset])`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Big-Endian.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
- `Buffer.prototype.writeDoubleLE(value, [offset])`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Little-Endian.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
- `Buffer.prototype.writeFloatBE(value, [offset])`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Big-Endian.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.
- `Buffer.prototype.writeFloatLE(value, [offset])`

  Schreiben Sie den `value` beim `offset` des `Buffers` mit Little-Endian.

  - `value`: Geben Sie eine Ganzzahl ein.
  - `offset`: Optional. Geben Sie eine Ganzzahl ein. Standard = 0.

Die folgenden Instanzmethoden werden unterstützt:

- `buffer[index]`

  Rufen Sie Oktett (Byte) beim `index` im `Buffer` ab oder legen Sie es fest.

  - Rufen Sie eine Zahl von `0` bis `255` ab. Oder legen Sie eine Zahl von `0` bis `255` fest.

Die folgenden Instanzeigenschaften werden unterstützt:

- `buffer`

  Rufen Sie das `ArrayBuffer`-Objekt für den Puffer ab.

- `byteOffset`

  Rufen Sie das `byteOffset` vom `Arraybuffer`-Objekt des Puffers ab.

- `length`

  Rufen Sie die Byteanzahl des Puffers ab.

> **Note**
>
> Alle Methoden des Buffer-Moduls sind neu in Runtime 2.0. JavaScript

Abfragezeichenfolge

> **Note**
>
> Das CloudFront Functions-Ereignisobjekt analysiert automatisch URL-Abfragezeichenfolgen für Sie. Das bedeutet, dass Sie dieses Modul in den meisten Fällen nicht verwenden müssen.

Das Modul für Abfragezeichenfolgen (`querystring`) bietet Methoden zum Analysieren und Formatieren von URL-Abfragezeichenfolgen. Sie können das Modul mit `require('querystring')` laden. Das Modul bietet die folgenden Methoden.

`querystring.escape(`*`string`*`)`

URL-kodiert die angegebene `string` und gibt eine entflohene Abfragezeichenfolge zurück. Die Methode wird von `querystring.stringify()` verwendet und sollte nicht direkt verwendet werden.

`querystring.parse(`*`string`*`[, `*`separator`*`[, `*`equal`*`[, `*`options`*`]]])`

Analysiert eine Abfragezeichenfolge (`string`) und gibt ein Objekt zurück.

Der `separator`-Parameter ist eine Teilzeichenfolge zum Abgrenzen von Schlüssel- und Wertepaaren in der Abfragezeichenfolge. Standardmäßig ist dies &.

Der `equal`-Parameter ist eine Teilzeichenfolge zum Abgrenzen von Schlüsseln und Werten in der Abfragezeichenfolge. Standardmäßig ist dies =.

Der `options`-Parameter ist ein Objekt mit den folgenden Schlüsseln:

`decodeURIComponent` *`function`*

Eine Funktion zum Entschlüsseln von prozentkodierten Zeichen in der Abfragezeichenfolge. Standardmäßig ist dies `querystring.unescape()`.

`maxKeys` *`number`*

Die maximale Anzahl der Schlüssel zum Parsen. Standardmäßig ist dies `1000`. Verwenden Sie den Wert `0`, um die Beschränkungen für das Zählen von Schlüsseln aufzuheben.

Standardmäßig wird davon ausgegangen, dass prozentcodierte Zeichen innerhalb der Abfragezeichenfolge die UTF-8-Codierung verwenden. Ungültige UTF-8-Sequenzen werden durch das Ersatzzeichen `U+FFFD` ersetzt.

Zum Beispiel für die folgende Abfragezeichenfolge:

```
'name=value&abc=xyz&abc=123'
```

Der Rückgabewert von `querystring.parse()` ist:

```
{
name: 'value',
abc: ['xyz', '123']
}
```

`querystring.decode()` ist ein Alias für `querystring.parse()`.

`querystring.stringify(`*`object`*`[, `*`separator`*`[, `*`equal`*`[, `*`options`*`]]])`

Serialisiert ein `object` und gibt eine Abfragezeichenfolge zurück.

Der `separator`-Parameter ist eine Teilzeichenfolge zum Abgrenzen von Schlüssel- und Wertepaaren in der Abfragezeichenfolge. Standardmäßig ist dies &.

Der `equal`-Parameter ist eine Teilzeichenfolge zum Abgrenzen von Schlüsseln und Werten in der Abfragezeichenfolge. Standardmäßig ist dies =.

Der `options`-Parameter ist ein Objekt mit den folgenden Schlüsseln:

`encodeURIComponent` *`function`*

Die Funktion, die zum Konvertieren von URL-unsicheren Zeichen in die prozentuale Kodierung in der Abfragezeichenfolge verwendet wird. Standardmäßig ist dies `querystring.escape()`.

Standardmäßig werden Zeichen, die eine prozentuale Kodierung innerhalb der Abfragezeichenfolge erfordern, als UTF-8 codiert. Um eine andere Codierung zu verwenden, geben Sie die Option `encodeURIComponent` an.

Zum Beispiel für den folgenden Code:

```
querystring.stringify({ name: 'value', abc: ['xyz', '123'], anotherName: '' });
```

Der Rückgabewert ist:

```
'name=value&abc=xyz&abc=123&anotherName='
```

`querystring.encode()` ist ein Alias für `querystring.stringify()`.

`querystring.unescape(`*`string`*`)`

Dekodiert die prozentualen Zeichen der URL in der angegebenen `string` und gibt eine nicht entdeckene Abfragezeichenfolge zurück. Diese Methode wird von `querystring.parse()` verwendet und sollte nicht direkt verwendet werden.

Crypto

Das kryptographische Modul (`crypto`) bietet standardmäßige Hashing- und HMAC-Helfer (Hash-basierter Nachrichtenauthentifizierungscode). Sie können das Modul mit `require('crypto')` laden.

Hashing-Methoden

`crypto.createHash(`*`algorithm`*`)`

Erstellt und gibt ein Hash-Objekt zurück, mit dem Sie Hash-Digests mit dem angegebenen Algorithmus generieren können: `md5`, `sha1` oder `sha256`.

`hash.update(data)`

Aktualisiert den Hash-Inhalt mit dem angegebenen `data`.

`hash.digest([encoding])`

Berechnet den Digest aller mit `hash.update()` übergebenen Daten. Die Codierung kann `hex`, `base64` oder `base64url` sein.

HMAC-Methoden

`crypto.createHmac(algorithm, secret key)`

Erstellt und gibt ein HMAC-Objekt zurück, das das angegebene `algorithm` und `secret key` verwendet. Der Algorithmus kann `md5`, `sha1` oder `sha256` sein.

`hmac.update(data)`

Aktualisiert den HMAC-Inhalt mit den angegebenen `data`.

`hmac.digest([encoding])`

Berechnet den Digest aller mit `hmac.update()` übergebenen Daten. Die Codierung kann `hex`, `base64` oder `base64url` sein.

Eingeschränkte Features

Die folgenden JavaScript Sprachfunktionen werden entweder nicht unterstützt oder sind aus Sicherheitsgründen eingeschränkt.

Dynamische Codeauswertung

Die dynamische Codeauswertung wird nicht unterstützt. Sowohl `eval()`- als auch `Function`-Konstruktoren geben einen Fehler aus, wenn sie versucht werden. Zum Beispiel gibt `const sum = new Function('a', 'b', 'return a + b')` einen Fehler aus.

Timer

Die Funktionen `setTimeout()`, `setImmediate()` und `clearTimeout()` werden nicht unterstützt. Es gibt keine Bestimmung, innerhalb einer Funktionsausführung zu verschieben oder zu ergeben. Ihre Funktion muss synchron bis zum Abschluss ausgeführt werden.

Datum und Zeitstempel

Aus Sicherheitsgründen besteht kein Zugang zu hochauflösenden Timern. Alle `Date`-Methoden zum Abfragen der aktuellen Uhrzeit geben während der Lebensdauer einer einzelnen Funktionsausführung immer den gleichen Wert zurück. Der zurückgegebene Zeitstempel ist die Zeit, zu der die Funktion gestartet wurde. Folglich können Sie die verstrichene Zeit in Ihrer Funktion nicht messen.

Zugriff auf das Dateisystem

Es gibt keinen Zugriff auf das Dateisystem. Zum Beispiel gibt es kein `fs`-Modul für den Dateisystemzugriff wie in Node.js.

Zugriff verarbeiten

Es gibt keinen Prozesszugriff. Beispielsweise gibt es kein `process` globales Objekt für die Verarbeitung von Informationszugriffen wie in Node.js.

Umgebungsvariablen

Es gibt keinen Zugriff auf Umgebungsvariablen. Stattdessen können Sie verwenden, CloudFront KeyValueStore um einen zentralen Datenspeicher mit Schlüssel-Wert-Paaren für Ihre Funktionen zu erstellen. CloudFront CloudFront KeyValueStore ermöglicht dynamische Aktualisierungen Ihrer Konfigurationsdaten, ohne dass Codeänderungen vorgenommen werden müssen. Weitere Informationen finden Sie unter Amazon CloudFront KeyValueStore.

Netzwerkzugriff

Es gibt keine Unterstützung für Netzwerkaufrufe. Zum Beispiel werden XHR, HTTP(S) und Socket nicht unterstützt.

Hilfsmethoden für Schlüsselwertspeicher

> **Note**
>
> Aufrufe von Hilfsmethoden zum Speichern von Schlüsselwerten aus CloudFront Functions lösen kein AWS CloudTrail Datenereignis aus. Diese Ereignisse werden nicht im CloudTrail Ereignisverlauf protokolliert. Weitere Informationen finden Sie unter Protokollieren Amazon CloudFront Amazon-API-Aufrufen mit AWS CloudTrail.

Dieser Abschnitt gilt, wenn Sie den CloudFront Key Value Store verwenden, um Schlüsselwerte in die von Ihnen erstellte Funktion aufzunehmen. CloudFront Functions verfügt über ein Modul, das drei Hilfsmethoden zum Lesen von Werten aus dem Schlüsselwertspeicher bereitstellt.

Um dieses Modul im Funktionscode zu verwenden, stellen Sie sicher, dass Sie der Funktion einen Schlüsselwertspeicher zugeordnet haben.

Fügen Sie als Nächstes die folgenden Anweisungen in die ersten Zeilen des Funktionscodes ein:

```
import cf from 'cloudfront';
const kvsHandle = cf.kvs();
```

`get()`-Methode

Verwenden Sie diese Methode, um den Schlüsselwert für den von Ihnen angegebenen Schlüsselnamen zurückzugeben.

Anforderung

```
get("key", options);
```

- `key`: Der Name des Schlüssels, dessen Wert abgerufen werden muss
- `options`: Es gibt eine Option,`format`. Diese stellt sicher, dass die Funktion die Daten korrekt analysiert. Mögliche Werte:
  - `string`: (Standard) UTF8 codiert
  - `json`
  - `bytes`: Roher Binärdatenpuffer

Beispiel anfordern

```
const value = await kvsHandle.get("myFunctionKey", { format: "string"});
```

Antwort

Die Antwort ist eine`promise`, die zu einem Wert in dem von using `options` angeforderten Format aufgelöst wird. Standardmäßig wird der Wert als Zeichenfolge zurückgegeben.

Fehlerbehandlung

Die `get()` Methode gibt einen Fehler zurück, wenn der von Ihnen angeforderte Schlüssel nicht im zugehörigen Schlüsselwertspeicher vorhanden ist. Um diesen Anwendungsfall zu verwalten, können Sie Ihrem Code einen `try catch` UND-Block hinzufügen.

> **Warning**
>
> Die Verwendung von Promise-Kombinatoren (zum Beispiel,`Promise.all`,`Promise.any`,) und Promise-Chain-Methoden (zum Beispiel `then` und`catch`) kann eine hohe Auslastung des Funktionsspeichers erfordern. Wenn Ihre Funktion das maximale Funktionsspeicherkontingent überschreitet, kann sie nicht ausgeführt werden. Um diesen Fehler zu vermeiden, empfehlen wir, die `await` Syntax sequentiell oder in Schleifen zu verwenden, um mehrere Werte anzufordern.
> Beispiel
>
> ```
> var value1 = await kvs.get('key1');
> var value2 = await kvs.get('key2');
> ```
>
> Derzeit verbessert die Verwendung von Promise-Kombinatoren zum Abrufen mehrerer Werte die Leistung nicht, wie im folgenden Beispiel.
>
> ```
> var values = await Promise.all([kvs.get('key1'), kvs.get('key2'),]);
> ```

`exists()`-Methode

Verwenden Sie diese Methode, um festzustellen, ob der Schlüssel im Schlüsselwertspeicher vorhanden ist oder nicht.

Anforderung

```
exists("key");
```

Beispiel anfordern

```
const exist = await kvsHandle.exists("myFunctionkey");
```

Antwort

Die Antwort ist a`promise`, die einen booleschen Wert (`true`oder`false`) zurückgibt. Dieser Wert gibt an, ob der Schlüssel im Schlüsselwertspeicher vorhanden ist oder nicht.

`meta()`-Methode

Verwenden Sie diese Methode, um Metadaten über den Schlüsselwertspeicher zurückzugeben.

Anforderung

```
meta();
```

Beispiel anfordern

```
const meta = await kvsHandle.meta();
```

Antwort

Die Antwort ist ein `promise`, das in ein Objekt mit den folgenden Eigenschaften aufgelöst wird:

- `creationDateTime`: Erstellungsdatum und -uhrzeit des Schlüsselwertspeichers im ISO 8601-Format.
- `lastUpdatedDateTime`: Datum und Uhrzeit der letzten Synchronisierung des Schlüsselwertspeichers mit der Quelle im ISO 8601-Format. Der Wert beinhaltet nicht die Ausbreitungszeit bis zum Edge.
- `keyCount`: Die Gesamtzahl der Schlüssel im KVS nach der letzten Synchronisierung mit der Quelle.

Beispiel für eine Antwort

```
{keyCount:3,creationDateTime:2023-11-30T23:07:55.765Z,lastUpdatedDateTime:2023-12-15T03:57:52.4
```

Hilfsmethoden für die Änderung des Ursprungs

Dieser Abschnitt gilt, wenn Sie den für die Anfrage verwendeten Ursprung in Ihrem CloudFront Functions-Code dynamisch aktualisieren oder ändern. Sie können den Ursprung nur auf Anfrage des Betrachters aktualisieren. Dies CloudFront gilt nur für Funktionen. CloudFront Functions verfügt über ein Modul, das Hilfsmethoden zur dynamischen Aktualisierung oder Änderung des Ursprungs bereitstellt.

Um dieses Modul zu verwenden, erstellen Sie eine CloudFront Funktion mit JavaScript Runtime 2.0 und fügen Sie die folgende Anweisung in die erste Zeile des Funktionscodes ein:

```
import cf from 'cloudfront';
```

Weitere Informationen finden Sie unter JavaScript Runtime 2.0-Funktionen für CloudFront Funktionen.

> **Note**
>
> Auf den Seiten Test-API und Testkonsole wird nicht getestet, ob eine Änderung des Ursprungs stattgefunden hat. Durch das Testen wird jedoch sichergestellt, dass der Funktionscode fehlerfrei ausgeführt wird.

Wählen Sie zwischen CloudFront Functions und Lambda @Edge

Sie können Ihre Ursprünge aktualisieren, indem Sie entweder CloudFront Functions oder Lambda @Edge verwenden.

Wenn Sie CloudFront Functions verwenden, um Ursprünge zu aktualisieren, verwenden Sie den Viewer-Request-Event-Trigger, was bedeutet, dass diese Logik bei jeder Anfrage ausgeführt wird, wenn diese Funktion verwendet wird. Bei der Verwendung von Lambda @Edge befinden sich die Funktionen zur Aktualisierung des Ursprungs auf dem Trigger des Origin-Request-Ereignisses, was bedeutet, dass diese Logik nur bei Cache-Fehlschlägen ausgeführt wird.

Ihre Wahl hängt weitgehend von Ihrer Arbeitslast und der bestehenden Nutzung von CloudFront Functions und Lambda @Edge in Ihren Distributionen ab. Die folgenden Überlegungen können Ihnen bei der Entscheidung helfen, ob Sie CloudFront Functions oder Lambda @Edge verwenden möchten, um Ihre Ursprünge zu aktualisieren.

CloudFront Functions ist in den folgenden Situationen am nützlichsten:

- Wenn Ihre Anfragen dynamisch sind (was bedeutet, dass sie nicht zwischengespeichert werden können) und immer an den Ursprung weitergeleitet werden. CloudFront Functions bietet eine bessere Leistung und niedrigere Gesamtkosten.
- Wenn Sie bereits über eine CloudFront Viewer-Anforderungsfunktion verfügen, die bei jeder Anfrage ausgeführt wird, können Sie die ursprüngliche Aktualisierungslogik zur vorhandenen Funktion hinzufügen.

Informationen zur Verwendung von CloudFront Funktionen zur Aktualisierung von Ursprüngen finden Sie in den folgenden Themen in den Hilfsmethoden.

Lambda @Edge ist in den folgenden Situationen am nützlichsten:

- Wenn Sie Inhalte haben, die in hohem Maße zwischengespeichert werden können, kann Lambda @Edge kostengünstiger sein, da es nur bei Cache-Fehlern ausgeführt wird, während CloudFront Functions bei jeder Anfrage ausgeführt wird.
- Wenn Sie bereits über eine Lambda @Edge -Funktion für Ursprungsanfragen verfügen, können Sie die Origin-Aktualisierungslogik zur vorhandenen Funktion hinzufügen.
- Wenn Ihre Origin-Aktualisierungslogik das Abrufen von Daten aus Datenquellen von Drittanbietern wie Amazon DynamoDB oder Amazon S3 erfordert.

Weitere Informationen zu Lambda @Edge finden Sie unterPersonalisieren Sie am Rand mit Lambda @Edge.

updateRequestOrigin() -Methode

Verwenden Sie die `updateRequestOrigin()` Methode, um die Ursprungseinstellungen für eine Anfrage zu aktualisieren. Sie können diese Methode verwenden, um bestehende Ursprungseigenschaften für Ursprünge zu aktualisieren, die bereits in Ihrer Distribution definiert sind, oder um einen neuen Ursprung für die Anfrage zu definieren. Geben Sie dazu die Eigenschaften an, die Sie ändern möchten.

> **Important**
>
> Alle Einstellungen, die Sie nicht in der angeben`updateRequestOrigin()`, erben dieselben Einstellungen aus der Konfiguration des vorhandenen Ursprungs.

Der von der `updateRequestOrigin()` Methode festgelegte Ursprung kann ein beliebiger HTTP-Endpunkt sein und muss kein vorhandener Ursprung in Ihrer CloudFront Distribution sein.

> **Hinweise**
>
> - Wenn Sie einen Ursprung aktualisieren, der Teil einer Ursprungsgruppe ist, wird nur der primäre Ursprung der Ursprungsgruppe aktualisiert. Der sekundäre Ursprung bleibt

unverändert. Jeder Antwortcode des geänderten Ursprungs, der den Failover-Kriterien entspricht, löst einen Failover zum sekundären Ursprung aus.

- Wenn Sie den Quelltyp ändern und OAC aktiviert haben, stellen Sie sicher, dass der Eingangstyp mit dem neuen Originstyp `originAccessControlConfig` übereinstimmt.
- Sie können die `updateRequestOrigin()` Methode nicht verwenden, um VPC-Ursprünge zu aktualisieren. Die Anfrage wird fehlschlagen.

Anforderung

```
updateRequestOrigin({origin properties})
```

Die `origin properties` kann Folgendes enthalten:

domainName (optional)

Der Domänenname des Ursprungs. Wenn dies nicht angegeben wird, wird stattdessen der Domainname des zugewiesenen Ursprungs verwendet.

Für benutzerdefinierte Ursprünge

Geben Sie einen DNS-Domainnamen an, z. `www.example.com` B. Der Domainname darf keinen Doppelpunkt (:) enthalten und darf keine IP-Adresse sein. Der Domänenname kann bis zu 253 Zeichen lang sein.

Für S3-Ursprünge

Geben Sie den DNS-Domainnamen des Amazon S3 S3-Buckets an, z. `amzn-s3-demo-bucket.s3.eu-west-1.amazonaws.com` B. Der Name kann bis zu 128 Zeichen lang sein und muss in Kleinbuchstaben geschrieben werden.

originPath (optional)

Der Verzeichnispfad auf dem Ursprung, aus dem die Anforderung den Inhalt abrufen soll. Der Pfad sollte mit einem Schrägstrich (/) beginnen, aber nicht mit einem enden. Zum Beispiel sollte er nicht mit `example-path/` enden. Wenn dies nicht angegeben ist, wird der Quellpfad vom zugewiesenen Ursprung verwendet.

Für benutzerdefinierte Ursprünge

Der Pfad sollte URL-kodiert sein und eine maximale Länge von 255 Zeichen haben.

customHeaders (optional)

Sie können benutzerdefinierte Header in die Anforderung aufnehmen, indem Sie für jeden benutzerdefinierten Header einen Header-Namen und einen -Wert angeben. Das Format unterscheidet sich von dem der Anfrage- und Antwortheader in der Ereignisstruktur. Verwenden Sie die folgende Syntax für Schlüssel-Wert-Paare:

```
{"key1": "value1", "key2": "value2", ...}
```

Sie können keine unzulässigen Header hinzufügen, und ein Header mit demselben Namen darf nicht auch in der eingehenden Anfrage enthalten sein. `headers` Der Header-Name muss in Ihrem Funktionscode in Kleinbuchstaben geschrieben werden. Wenn CloudFront Functions das Ereignisobjekt wieder in eine HTTP-Anforderung konvertiert, wird der erste Buchstabe jedes Worts in Header-Namen groß geschrieben und die Wörter werden durch einen Bindestrich getrennt.

Wenn der Funktionscode beispielsweise einen Header mit dem Name`example-header-name`, CloudFront konvertiert diesen `Example-Header-Name` in der HTTP-Anfrage in einen Header hinzufügt. Weitere Informationen erhalten Sie unter Benutzerdefinierte Header, die nicht zu CloudFront ursprünglichen Anfragen hinzugefügt werden können und Einschränkungen für Edge-Funktionen.

Wenn dies nicht angegeben wird, werden alle benutzerdefinierten Header aus dem zugewiesenen Ursprung verwendet.

Verbindungsversuche (optional)

Die Häufigkeit, mit der CloudFront versucht wird, eine Verbindung zum Ursprung herzustellen. Das Minimum ist 1 und das Maximum ist 3. Wenn dies nicht angegeben ist, werden die Verbindungsversuche vom zugewiesenen Ursprung verwendet.

OriginShield (optional)

Dadurch wird CloudFront Origin Shield aktiviert oder aktualisiert. Die Verwendung von Origin Shield kann dazu beitragen, die Belastung Ihres Ursprungs zu reduzieren. Weitere Informationen finden Sie unter Verwenden Sie Amazon CloudFront Origin Shield. Wenn dies nicht angegeben ist, werden die Origin Shield-Einstellungen des zugewiesenen Ursprungs verwendet.

aktiviert (erforderlich)

Boolescher Ausdruck zur Aktivierung oder Deaktivierung von Origin Shield. Akzeptiert einen `true` Oder-Wert`false`.

Region (erforderlich, wenn aktiviert)

Das AWS-Region für Origin Shield. Geben Sie die AWS-Region an, die die niedrigste Latenz zu Ihrem Ursprung aufweist. Verwenden Sie den Regionalcode, nicht den Namen der Region. Verwenden Sie dies beispielsweise, `us-east-2` um die Region USA Ost (Ohio) anzugeben.

Wenn du CloudFront Origin Shield aktivierst, musst du das AWS-Region dafür angeben. Eine Liste der verfügbaren Regionen AWS-Regionen und Hilfe bei der Auswahl der besten Region für deine Herkunft findest du unterWähle die AWS Region für Origin Shield.

originAccessControlConfig (optional)

Die eindeutige Kennung einer Origin Access Control (OAC) für diesen Ursprung. Dies wird nur verwendet, wenn der Ursprung ein CloudFront OAC unterstützt, z. B. Amazon S3, Lambda-Funktion URLs und MediaStore MediaPackage V2. Wenn dies nicht angegeben ist, werden die OAC-Einstellungen des zugewiesenen Ursprungs verwendet.

Dies unterstützt die ältere Origin Access Identity (OAI) nicht. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff auf einen AWS Ursprung.

aktiviert (erforderlich)

Boolescher Ausdruck zum Aktivieren oder Deaktivieren von OAC. Akzeptiert einen Oder-Wert`true`. `false`

SigningBehavior (erforderlich, wenn aktiviert)

Gibt an, welche Anfragen CloudFront signiert werden (fügt Authentifizierungsinformationen hinzu). Geben Sie `always` für den häufigsten Anwendungsfall an. Weitere Informationen finden Sie unter Erweiterte Einstellungen für die Ursprungszugriffssteuerung.

Folgende Werte sind in diesem Feld möglich:

- `always`— CloudFront signiert alle ursprünglichen Anfragen und überschreibt dabei den `Authorization` Header der Viewer-Anfrage, falls vorhanden.
- `never`— signiert CloudFront keine ursprünglichen Anfragen. Dieser Wert deaktiviert die Ursprungszugriffskontrolle für den Ursprung.
- `no-override`— Wenn die Viewer-Anfrage den `Authorization` Header nicht enthält, wird die ursprüngliche Anfrage CloudFront signiert. Wenn die Viewer-Anfrage den `Authorization` Header enthält, wird die ursprüngliche Anfrage CloudFront nicht signiert und stattdessen der `Authorization` Header aus der Viewer-Anfrage weitergegeben.

> **Warning**
>
> Um den `Authorization` Header aus der Viewer-Anfrage weiterzugeben, müssen Sie ihn zu einer Quellanforderungsrichtlinie für alle Cache-Verhaltensweisen hinzufügen, die Ursprünge verwenden, die mit dieser ursprünglichen Zugriffskontrolle verknüpft sind. Weitere Informationen finden Sie unter Kontrollieren Sie Herkunftsanfragen mit einer Richtlinie.

SigningProtocol (erforderlich, wenn aktiviert)

Das Signaturprotokoll des OAC, das festlegt, wie Anfragen CloudFront signiert (authentifiziert) werden. Der einzige gültige Wert ist `sigv4`.

OriginType (erforderlich, wenn aktiviert)

Der Typ des Ursprungs für dieses OAC. Gültige Werte sind: `s3`, `mediapackagev2`, `mediastore` und `lambda`.

Timeouts (optional)

Timeouts, die Sie angeben können, wie lange versucht CloudFront werden soll, auf die Antwort von Origins zu warten oder Daten zu senden. Wenn dies nicht angegeben ist, werden die Timeout-Einstellungen des zugewiesenen Ursprungs verwendet.

> **Note**
>
> Sofern nicht anders angegeben, unterstützen diese Timeouts sowohl benutzerdefinierte Ursprünge als auch Amazon S3 S3-Ursprünge.

readTimeout (optional)

Das `readTimeout` gilt für die beiden folgenden Werte:

- Wie lange (in Sekunden) auf eine Antwort gewartet wird, nachdem eine Anfrage an den Ursprung weitergeleitet CloudFront wurde.
- Wie lange (in Sekunden) nach dem Empfang eines Antwortpakets vom Ursprung und vor dem Empfang des nächsten Pakets CloudFront gewartet wird.

Das minimale Timeout beträgt 1 Sekunde und das Maximum 120 Sekunden. Weitere Informationen finden Sie unter Timeout bei der Antwort.

responseCompletionTimeout (Optional)

Die Zeit (in Sekunden), in der eine Anfrage vom CloudFront Ursprung geöffnet bleiben und auf eine Antwort warten kann. Wenn bis zu diesem Zeitpunkt noch keine vollständige Antwort vom Ursprung eingegangen ist, wird die Verbindung CloudFront beendet.

Der Wert für `responseCompletionTimeout` muss gleich oder größer als der Wert für sein`readTimeout`. Weitere Informationen finden Sie unter Timeout bei Abschluss der Antwort.

keepAliveTimeout (Optional)

Dieses Timeout gilt nur für benutzerdefinierte Ursprünge, nicht für Amazon S3 S3-Ursprünge. (Bei S3-Ursprungskonfigurationen werden diese Einstellungen ignoriert.)

Das `keepAliveTimeout` gibt an, wie lange versucht CloudFront werden soll, die Verbindung zum Ursprung aufrechtzuerhalten, nachdem das letzte Paket der Antwort empfangen wurde. Das minimale Timeout beträgt 1 Sekunde und das Maximum 120 Sekunden. Weitere Informationen finden Sie unter Keep-Alive-Timeout (nur benutzerdefinierte und VPC-Ursprünge).

ConnectionTimeout (optional)

Die Anzahl der Sekunden, die CloudFront gewartet wird, wenn versucht wird, eine Verbindung zum Ursprung herzustellen. Das minimale Timeout beträgt 1 Sekunde und das Maximum 10 Sekunden. Weitere Informationen finden Sie unter Verbindungstimeout.

customOriginConfig (Optional)

Wird verwendet`customOriginConfig`, um Verbindungseinstellungen für Ursprünge anzugeben, die kein Amazon S3 S3-Bucket sind. Es gibt eine Ausnahme: Sie können diese Einstellungen angeben, wenn der S3-Bucket mit statischem Website-Hosting konfiguriert ist. (Bei anderen Arten von S3-Bucket-Konfigurationen werden diese Einstellungen ignoriert.) Wenn `customOriginConfig` nicht angegeben, werden die Einstellungen des zugewiesenen Ursprungs verwendet.

Port (erforderlich)

Der HTTP-Port, der für die Verbindung zum Ursprung CloudFront verwendet wird. Geben Sie den HTTP-Port an, den der Ursprung überwacht.

Protokoll (erforderlich)

Gibt das Protokoll (HTTP oder HTTPS) an, das für die Verbindung zum Ursprung CloudFront verwendet wird. Gültige Werte sind:

- `http`— verwendet CloudFront immer HTTP, um eine Verbindung zum Ursprung herzustellen
- `https`— verwendet CloudFront immer HTTPS, um eine Verbindung zum Ursprung herzustellen

sslProtocols (erforderlich)

Eine Liste, die das SSL/TLS Mindestprotokoll angibt, das CloudFront verwendet wird, wenn Sie über HTTPS eine Verbindung zu Ihrem Ursprung herstellen. Gültige Werte sind: `SSLv3`, `TLSv1`, `TLSv1.1` und `TLSv1.2`. Weitere Informationen finden Sie unter Mindest-SSL-Protokoll für Ursprung.

Example — Aktualisierung auf den Ursprung der Amazon S3 S3-Anfrage

Das folgende Beispiel ändert den Ursprung der Viewer-Anfrage in einen S3-Bucket, aktiviert OAC und setzt benutzerdefinierte Header zurück, die an den Ursprung gesendet wurden.

```
cf.updateRequestOrigin({
    "domainName" : "amzn-s3-demo-bucket-in-us-east-1.s3.us-east-1.amazonaws.com",
    "originAccessControlConfig": {
        "enabled": true,
        "signingBehavior": "always",
        "signingProtocol": "sigv4",
        "originType": "s3"
    },
    // Empty object resets any header configured on the assigned origin
    "customHeaders": {}
});
```

Example — Aktualisierung der Herkunft der Application Load Balancer Balancer-Anforderung

Das folgende Beispiel ändert den Ursprung der Viewer-Anfrage in einen Application Load Balancer Balancer-Ursprung und legt einen benutzerdefinierten Header und Timeouts fest.

```
cf.updateRequestOrigin({
    "domainName" : "example-1234567890.us-east-1.elb.amazonaws.com",
```

```
    "timeouts": {
        "readTimeout": 30,
        "connectionTimeout": 5
    },
    "customHeaders": {
        "x-stage": "production",
        "x-region": "us-east-1"
    }
});
```

Example — Update auf Origin mit aktiviertem Origin Shield

Im folgenden Beispiel ist Origin Shield für den Ursprung in der Distribution aktiviert. Der Funktionscode aktualisiert nur den Domainnamen, der für den Ursprung verwendet wurde, und lässt alle anderen optionalen Parameter weg. In diesem Fall wird Origin Shield weiterhin mit dem geänderten Ursprungs-Domainnamen verwendet, da die Origin Shield-Parameter nicht aktualisiert wurden.

```
cf.updateRequestOrigin({
    "domainName" : "www.example.com"
});
```

selectRequestOriginById() Methode

Verwenden Sie diese Option`selectRequestOriginById()`, um einen vorhandenen Ursprung zu aktualisieren, indem Sie einen anderen Ursprung auswählen, der bereits in Ihrer Distribution konfiguriert ist. Diese Methode verwendet dieselben Einstellungen, die durch den aktualisierten Ursprung definiert sind.

Diese Methode akzeptiert nur Ursprünge, die bereits in derselben Distribution definiert sind, die beim Ausführen der Funktion verwendet wurde. Ursprünge werden durch die Quell-ID referenziert. Dabei handelt es sich um den Ursprungsnamen, den Sie bei der Einrichtung des Ursprungs definiert haben.

Wenn Sie in Ihrer Distribution einen VPC-Ursprung konfiguriert haben, können Sie mit dieser Methode Ihren Ursprung auf Ihren VPC-Ursprung aktualisieren. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff mit VPC-Ursprüngen.

Anforderung

```
selectRequestOriginById(origin_id)
```

Im vorherigen Beispiel `origin_id` handelt es sich um eine Zeichenfolge, die auf den Ursprungsnamen eines Ursprungs in der Distribution verweist, auf der die Funktion ausgeführt wird.

Example — Wählen Sie den Ursprung der Amazon S3 S3-Anfrage

Im folgenden Beispiel wird der angegebene Ursprung `amzn-s3-demo-bucket-in-us-east-1` aus der Liste der Ursprünge ausgewählt, die mit der Verteilung verknüpft sind, und die Konfigurationseinstellungen des `amzn-s3-demo-bucket-in-us-east-1` Ursprungs werden auf die Anfrage angewendet.

```
cf.selectRequestOriginById("amzn-s3-demo-bucket-in-us-east-1");
```

Example — Wählen Sie den Ursprung der Application Load Balancer Balancer-Anforderung

Im folgenden Beispiel wird ein Application Load Balancer Balancer-Ursprung ausgewählt, der `myALB-prod` aus der Liste der mit der Verteilung verknüpften Ursprünge benannt ist, und wendet die Konfigurationseinstellungen von `myALB-prod` auf die Anforderung an.

```
cf.selectRequestOriginById("myALB-prod");
```

createRequestOriginMethode Group ()

Wird verwendet`createRequestOriginGroup()`, um zwei Ursprünge zu definieren, die als Ursprungsgruppe für Failover in Szenarien verwendet werden sollen, die eine hohe Verfügbarkeit erfordern.

Eine Ursprungsgruppe umfasst zwei Ursprünge (einen primären und einen sekundären) und ein von Ihnen festgelegtes Failover-Kriterium. Sie erstellen eine Ursprungsgruppe, um das Origin-Failover in zu unterstützen. CloudFront Wenn Sie mit dieser Methode eine Ursprungsgruppe erstellen oder aktualisieren, können Sie die Ursprungsgruppe anstelle eines einzelnen Ursprungs angeben. CloudFront führt unter Verwendung der Failover-Kriterien ein Failover vom primären Ursprung zum sekundären Ursprung durch.

Wenn Sie in Ihrer Distribution einen VPC-Ursprung konfiguriert haben, können Sie diese Methode verwenden, um eine Ursprungsgruppe mithilfe eines VPC-Ursprungs zu erstellen. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff mit VPC-Ursprüngen.

Anforderung

```
createRequestOriginGroup({origin_group_properties})
```

Im vorherigen Beispiel `origin_group_properties` kann sie Folgendes enthalten:

OriginIDs (erforderlich)

Array von`origin_ids`, wobei das eine Zeichenfolge `origin_id` ist, die auf den Ursprungsnamen eines Ursprungs in der Distribution verweist, auf der die Funktion ausgeführt wird. Sie müssen zwei Ursprünge als Teil des Arrays angeben. Der erste Ursprung in der Liste ist der primäre Ursprung und der zweite Ursprung dient als zweiter Ursprung für Failover-Zwecke.

Auswahlkriterien (optional)

Wählen Sie aus, ob Sie die ursprünglichen `default` Failover-Kriterien oder die `media-quality-score` basierte Failoverlogik verwenden möchten. Gültige Werte sind:

- `default`verwendet die Failover-Kriterien, basierend auf den Statuscodes, die in der angegeben sind. `failoverCriteria` Wenn Sie die Funktion `selectionCriteria` nicht angeben, `default` wird verwendet.
- `media-quality-score`wird verwendet, wenn die medienbewusste Routing-Funktion verwendet wird.

Failover-Kriterien (erforderlich)

Eine Reihe von Statuscodes, die, wenn sie vom primären Ursprung zurückgegeben werden, einen Failover CloudFront zum sekundären Ursprung auslösen. Wenn Sie eine bestehende Ursprungsgruppe überschreiben, überschreibt dieses Array alle Failover-Statuscodes, die in der ursprünglichen Konfiguration der Ursprungsgruppe festgelegt sind.

Wenn Sie diese Option verwenden `media-quality-scoreselectionCriteria`, CloudFront wird versucht, Anfragen auf der Grundlage der Medienqualitätsbewertung weiterzuleiten. Wenn der ausgewählte Ursprung einen in diesem Array festgelegten Fehlercode zurückgibt, CloudFront wird ein Failover zum anderen Ursprung durchgeführt.

Example — Gruppe mit Ursprung der Anfrage erstellen

Das folgende Beispiel erstellt eine Ursprungsgruppe für eine Anfrage, die den Ursprung verwendet IDs. Diese Ursprünge IDs stammen aus der Konfiguration der Ursprungsgruppe für die Distribution, die zur Ausführung dieser Funktion verwendet wurde.

```
cf.createRequestOriginGroup({
    originIds: ["us-east-1-s3-origin", "us-west-2-s3-origin"],
    failoverCriteria: {
```

```
        statusCodes: [500, 502, 503, 504]
    }
});
```

Hilfsmethoden für CloudFront SaaS Manager-Eigenschaften

Verwenden Sie die folgenden Hilfsfunktionen für CloudFront SaaS Manager, um Werte für Ihre Multi-Tenant-Verteilungen in der von Ihnen erstellten Funktion abzurufen. Um die Beispiele auf dieser Seite verwenden zu können, müssen Sie zuerst eine CloudFront Funktion mithilfe von JavaScript Runtime 2.0 erstellen. Weitere Informationen finden Sie unter JavaScript Runtime 2.0-Funktionen für CloudFront Funktionen.

Themen

- Verbindungsgruppen
- Mandanten für die Verteilung

Verbindungsgruppen

Die Verbindungsgruppe, die Ihren Verteilungsmandanten zugeordnet ist, hat einen Domänennamen.

Verwenden Sie das `endpoint` Feld des `context` Unterobjekts des Ereignisobjekts, um diesen Wert abzurufen.

Anforderung

```
const value = event.context.endpoint;
```

Antwort

Die Antwort ist ein`string`, die den Domänennamen der Verbindungsgruppe enthält, z. B. d111111abcdef8.cloudfront.net. Das `endpoint` Feld wird nur angezeigt, wenn Ihre Funktion für eine Mehrmandantenverteilung mit einer zugehörigen Verbindungsgruppe aufgerufen wird. Weitere Informationen finden Sie unter Context-Objekt.

Mandanten für die Verteilung

CloudFront Functions verfügt über ein Modul, das den Zugriff auf bestimmte Werte für Distributionsmandanten ermöglicht.

Um dieses Modul zu verwenden, fügen Sie die folgende Anweisung in die erste Zeile Ihres Funktionscodes ein:

```
import cf from 'cloudfront';
```

Sie können die folgenden Beispiele nur in der `handler` Funktion verwenden, entweder direkt oder über eine beliebige Funktion mit verschachtelten Aufrufen.

`distributionTenant.id` field

Verwenden Sie dieses Feld, um den Wert der Distribution-Mandanten-ID abzurufen.

Anforderung

```
const value = cf.distributionTenant.id;
```

Antwort

Die Antwort ist ein`string`, die die Verteilungsmandanten-ID enthält, z. `dt_1a2b3c4d5e6f7` B.

Fehlerbehandlung

Wenn Ihre Funktion für eine Standardverteilung aufgerufen wird, wird bei Angabe des `distributionTenant.id` Felds der `distributionTenant module is not available` Typfehler zurückgegeben. Um diesen Anwendungsfall zu behandeln, können Sie Ihrem Code einen `try catch` UND-Block hinzufügen.

`distributionTenant.parameters.get()`-Methode

Verwenden Sie diese Methode, um den Wert für die von Ihnen angegebenen Parameternamen für den Verteilungsmandanten zurückzugeben.

```
distributionTenant.parameters.get("key");
```

`key`: Der Name des Parameters des Verteilungsmandanten, für den Sie den Wert abrufen möchten.

Anfrage

```
const value = distributionTenant.parameters.get("key");
```

Antwort

Die Antwort ist einestring, die den Wert für den Verteilungsmandantenparameter enthält. Wenn Ihr Schlüsselname beispielsweise lautetTenantPath, könnte der Wert für diesen Parameter lautentenant1.

Fehlerbehandlung

Möglicherweise erhalten Sie die folgenden Fehler:

- Wenn Ihre Funktion für eine Standardverteilung aufgerufen wird, gibt die `distributionTenant.parameters.get()` Methode den `distributionTenant module is not available` Typfehler zurück.
- Der `DistributionTenantParameterKeyNotFound` Fehler wird zurückgegeben, wenn der von Ihnen angegebene Parameter für den Verteilungsmandanten nicht existiert.

Um diese Anwendungsfälle zu verwalten, können Sie Ihrem Code einen `catch` Block `try` und einen Block hinzufügen.

Verwendung von async und await

CloudFront Funktionen, die in JavaScript Runtime Functions 2.0 ausgeführt werden, stellen eine `async await` Syntax für den Umgang mit `Promise` Objekten bereit. Promises stellen verzögerte Ergebnisse dar, auf die über das `await`-Schlüsselwort in Funktionen zugegriffen werden kann, die als `async` gekennzeichnet sind. Verschiedene neue WebCrypto Funktionen verwenden Promises.

Weitere Informationen über `Promise`-Objekte finden Sie unter Promise.

> **Note**
>
> Sie müssen JavaScript Runtime 2.0 für die folgenden Codebeispiele verwenden. `await`kann nur innerhalb von `async` Funktionen verwendet werden. `async`Argumente und Schließungen werden nicht unterstützt.

```
async function answer() {
    return 42;
}

// Note: async, await can be used only inside an async function. async arguments and
 closures are not supported.
```

```
async function handler(event) {
    // var answer_value = answer(); // returns Promise, not a 42 value
    let answer_value = await answer(); // resolves Promise, 42
    console.log("Answer"+answer_value);
    event.request.headers['answer'] = { value : ""+answer_value };
    return event.request;
}
```

Der folgende JavaScript Beispielcode zeigt, wie Versprechen mit der `then` Chain-Methode angezeigt werden. Sie können `catch` verwenden, um Fehler anzuzeigen.

> **Warning**
>
> Die Verwendung von Promise-Kombinatoren (zum Beispiel,`Promise.all`,`Promise.any`,) und Promise-Chain-Methoden (zum Beispiel `then` und`catch`) kann eine hohe Auslastung des Funktionsspeichers erfordern. Wenn Ihre Funktion das maximale Funktionsspeicherkontingent überschreitet, kann sie nicht ausgeführt werden. Um diesen Fehler zu vermeiden, empfehlen wir, die `await` Syntax anstelle von `promise` Methoden zu verwenden.

```
async function answer() {
    return 42;
}

async function squared_answer() {
   return answer().then(value => value * value)
}
// Note: async, await can be used only inside an async function. async arguments and
 closures are not supported.
async function handler(event) {
    // var answer_value = answer(); // returns Promise, not a 42 value
    let answer_value = await squared_answer(); // resolves Promise, 42
    console.log("Answer"+answer_value);
    event.request.headers['answer'] = { value : ""+answer_value };
    return event.request;
}
```

Funktionen erstellen

Sie erstellen eine Funktion in zwei Schritten:

1. Erstellen Sie den Funktionscode als JavaScript. Sie können das Standardbeispiel von der CloudFront Konsole aus verwenden oder ein eigenes schreiben. Weitere Informationen finden Sie unter den folgenden Themen:
   - Funktionscode schreiben
   - the section called “Ereignisstruktur”
   - CloudFront Funktionen, Beispiele für CloudFront
2. Verwenden Sie CloudFront es, um die Funktion zu erstellen und Ihren Code einzufügen. Der Code ist innerhalb der Funktion vorhanden (nicht als Referenz).

Console

Eine Funktion erstellen

1. Melden Sie sich bei der CloudFront Konsole unter an https://console.aws.amazon.com/cloudfront/v4/home#/functions und wählen Sie die Seite Funktionen aus.
2. Wählen Sie Funktion erstellen aus.
3. Geben Sie einen Funktionsnamen ein, der innerhalb von eindeutig ist AWS-Konto, wählen Sie die JavaScript Version aus und klicken Sie dann auf Weiter. Die Seite mit den Details der neuen Funktion wird angezeigt.

   > **Note**
   >
   > Um Schlüssel-Wert-Paare in der Funktion zu verwenden, müssen Sie JavaScript Runtime 2.0 wählen.

4. Wählen Sie im Abschnitt Funktionscode die Registerkarte Build und geben Sie Ihren Funktionscode ein. Der auf der Registerkarte Erstellen angezeigte Beispielcode veranschaulicht die grundlegende Syntax für den Funktionscode.
5. Wählen Sie Änderungen speichern.
6. Wenn der Funktionscode Schlüssel-Wert-Paare verwendet, müssen Sie einen Schlüsselwertspeicher zuordnen.

   Sie können den Schlüsselwertspeicher zuordnen, wenn Sie die Funktion zum ersten Mal erstellen. Sie können ihn auch später zuordnen, indem Sie die Funktion aktualisieren.

   Gehen Sie wie folgt vor, um jetzt einen Schlüsselwertspeicher zuzuordnen:

- Gehen Sie zum KeyValueStore Bereich Assoziieren und wählen Sie Vorhandenes zuordnen aus KeyValueStore.
- Wählen Sie den Schlüsselwertspeicher aus, der die Schlüssel-Wert-Paare in der Funktion enthält, und wählen Sie dann Assoziieren aus. KeyValueStore

CloudFront ordnet den Speicher sofort der Funktion zu. Sie müssen die Funktion nicht speichern.

CLI

Wenn Sie die CLI verwenden, erstellen Sie normalerweise zuerst den Funktionscode in einer Datei und dann die Funktion mit der AWS CLI.

Eine Funktion erstellen

1. Erstellen Sie den Funktionscode in einer Datei und speichern Sie ihn in einem Verzeichnis, mit dem Ihr Computer eine Verbindung herstellen kann.
2. Führen Sie den Befehl wie im Beispiel veranschaulicht aus. In diesem Beispiel wird die `fileb://` Notation verwendet, um die Datei zu übergeben. Es sind Zeilenumbrüche enthalten, um den Befehl lesbarer zu machen.

```
aws cloudfront create-function \
    --name MaxAge \
    --function-config '{"Comment":"Max Age 2 years","Runtime":"cloudfront-
js-2.0","KeyValueStoreAssociations":{"Quantity":1,"Items":
[{"KeyValueStoreARN":"arn:aws:cloudfront::111122223333:key-value-store/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"}]}}' \
    --function-code fileb://function-max-age-v1.js
```

Hinweise

- `Runtime`— Die Version von JavaScript. Um Schlüssel-Wert-Paare in der Funktion zu verwenden, müssen Sie Version 2.0 angeben.
- `KeyValueStoreAssociations`— Wenn Ihre Funktion Schlüssel-Wert-Paare verwendet, können Sie den Schlüsselwertspeicher zuordnen, wenn Sie die Funktion zum ersten Mal erstellen. Oder Sie können ihn später zuordnen, indem

Sie. `update-function` Die `Quantity` ist immer 1, weil jeder Funktion nur ein Schlüsselwertspeicher zugeordnet werden kann.

Wenn der Befehl erfolgreich ausgeführt wurde, wird die Ausgabe folgendermaßen angezeigt.

```
ETag: ETVABCEXAMPLE
FunctionSummary:
  FunctionConfig:
    Comment: Max Age 2 years
    Runtime: cloudfront-js-2.0
    KeyValueStoreAssociations= \
      {Quantity=1, \
      Items=[{KeyValueStoreARN='arn:aws:cloudfront::111122223333:key-value-
store/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'}]} \
  FunctionMetadata:
    CreatedTime: '2021-04-18T20:38:56.915000+00:00'
    FunctionARN: arn:aws:cloudfront::111122223333:function/MaxAge
    LastModifiedTime: '2023-11-19T20:38:56.915000+00:00'
    Stage: DEVELOPMENT
  Name: MaxAge
  Status: UNPUBLISHED
Location: https://cloudfront.amazonaws.com/2020-05-31/function/
arn:aws:cloudfront:::function/MaxAge
```

Die meisten Informationen werden aus der Anfrage wiederholt. Weitere Informationen werden von hinzugefügt CloudFront.

Hinweise

- `ETag`— Dieser Wert ändert sich jedes Mal, wenn Sie den Schlüsselwertspeicher ändern. Sie verwenden diesen Wert und den Funktionsnamen, um in future auf die Funktion zu verweisen. Stellen Sie sicher, dass Sie immer den aktuellen verwenden`ETag`.
- `FunctionARN`— Der ARN für Ihre CloudFront Funktion.
- 111122223333 — Das. AWS-Konto
- `Stage`— Die Phase der Funktion (oder). `LIVE` `DEVELOPMENT`
- `Status`— Der Status der Funktion (`PUBLISHED`oder`UNPUBLISHED`).

Nachdem Sie die Funktion erstellt haben, wird sie der `DEVELOPMENT` Bühne hinzugefügt. Wir empfehlen Ihnen, Ihre Funktion zu testen, bevor Sie sie veröffentlichen. Nachdem Sie Ihre Funktion veröffentlicht haben, wechselt sie zur `LIVE` Phase.

Funktionen testen

Bevor Sie die Funktion in der Live-Phase (Produktion) bereitstellen, können Sie Ihre Funktion testen, um sicherzustellen, dass sie wie vorgesehen funktioniert. Um eine Funktion zu testen, geben Sie ein Ereignisobjekt an, das eine HTTP-Anfrage oder -Antwort darstellt, die Ihre CloudFront Distribution in der Produktion erhalten könnte.

CloudFront Functions macht Folgendes:

1. Führt die Funktion aus, wobei das bereitgestellte Ereignisobjekt als Eingabe verwendet wird.
2. Gibt das Ergebnis der Funktion (das geänderte Ereignisobjekt) zusammen mit allen Funktionsprotokollen oder Fehlermeldungen und der Rechenauslastung der Funktion zurück. Weitere Informationen zur Computing-Nutzung finden Sie unter the section called “Verstehen Sie die Computernutzung”.

> **Note**
>
> Wenn Sie eine Funktion testen, überprüft sie CloudFront nur anhand von Fehlern bei der Funktionsausführung. CloudFrontüberprüft nicht, ob die Anfrage nach der Veröffentlichung erfolgreich bearbeitet wird. Wenn Ihre Funktion beispielsweise einen erforderlichen Header löscht, ist der Test erfolgreich, da kein Problem mit dem Code vorliegt. Wenn Sie die Funktion jedoch veröffentlichen und sie einer Verteilung zuordnen, schlägt die Funktion fehl, wenn eine Anfrage gestellt wurde. CloudFront

Inhalt

Einrichten des Ereignisobjekts

Bevor Sie eine Funktion testen, müssen Sie das Ereignisobjekt einrichten, mit dem Sie sie testen können. Es gibt mehrere Möglichkeiten, dies zu tun.

Option 1: Einrichten eines Ereignisobjekts, ohne es zu speichern

Sie können ein Ereignisobjekt im Visual Editor in der CloudFront Konsole einrichten, ohne es zu speichern.

Sie können dieses Ereignisobjekt verwenden, um die Funktion von der CloudFront Konsole aus zu testen, auch wenn es nicht gespeichert ist.

Option 2: Erstellen eines Ereignisobjekts im visuellen Editor

Sie können ein Ereignisobjekt im Visual Editor in der CloudFront Konsole einrichten und es nicht speichern. Sie können für jede Funktion 10 Ereignisobjekte erstellen, so dass Sie beispielsweise verschiedene mögliche Eingaben testen können.

Wenn Sie das Ereignisobjekt auf diese Weise erstellen, können Sie das Ereignisobjekt verwenden, um die Funktion in der CloudFront Konsole zu testen. Sie können es nicht verwenden, um die Funktion mit einer AWS API oder einem SDK zu testen.

Option 3: Erstellen eines Ereignisobjekts mit einem Texteditor

Sie können einen Texteditor verwenden, um ein Ereignisobjekt im JSON-Format zu erstellen. Informationen zur Struktur eines Ereignisobjekts finden Sie unter Ereignisstruktur.

Sie können dieses Ereignisobjekt verwenden, um die Funktion mit der CLI zu testen. Sie können es jedoch nicht verwenden, um die Funktion in der CloudFront Konsole zu testen.

Um ein Event-Objekt zu erstellen (Option 1 oder 2)

1. Melden Sie sich bei der CloudFront Konsole unter an https://console.aws.amazon.com/cloudfront/v4/home#/functions und wählen Sie die Seite Funktionen aus.

   Wählen Sie die Funktion aus, die Sie testen möchten.
2. Wählen Sie auf der Seite der Funktionsdetails die Registerkarte Test aus.
3. Wählen Sie für Ereignistyp eine der folgenden Optionen aus:

- Wenn die Funktion eine HTTP-Anfrage ändert oder basierend auf der Anfrage eine Antwort generiert, wählen Sie Betrachteranfrage aus. Der Abschnitt „Anfrage“ wird angezeigt.
- Wählen Sie Antwort des Betrachters aus. Die Abschnitte „Anfrage“ und „Antwort“ werden angezeigt.

4. Füllen Sie die Felder aus, die in das Ereignis aufgenommen werden sollen. Sie können „JSON bearbeiten“ wählen, um die unformatierte JSON-Datei anzuzeigen.
5. (Optional) Um das Ereignis zu speichern, wählen Sie Speichern aus, geben Sie im Feld Testereignis speichern einen Namen ein und wählen Sie dann Speichern aus.

   Sie können auch „JSON bearbeiten“ wählen und das unformatierte JSON kopieren und es in Ihrer eigenen Datei außerhalb von speichern CloudFront.

Um ein Event-Objekt zu erstellen (Option 3)

Erstellen Sie das Ereignisobjekt mit einem Texteditor. Speichern Sie die Datei in einem Verzeichnis, mit dem Ihr Computer eine Verbindung herstellen kann.

Stellen Sie sicher, dass Sie die folgenden Richtlinien befolgen:

- Ignorieren Sie die Felder `distributionDomainName`, `distributionId` und `requestId`.
- Die Namen von Headern, Cookies und Abfragezeichenfolgen müssen in Kleinbuchstaben geschrieben werden.

Eine Möglichkeit, ein Ereignisobjekt auf diese Weise zu erstellen, besteht darin, mit dem visuellen Editor ein Beispiel zu erstellen. Sie können sicher sein, dass das Beispiel korrekt formatiert ist. Sie können dann den unformatierten JSON-Code kopieren, in einen Text-Editor einfügen und die Datei speichern.

Weitere Hinweise zur Struktur eines Ereignisses finden Sie unter. Ereignisstruktur

Testen der Funktion

Sie können eine Funktion in der CloudFront Konsole oder mit der AWS Command Line Interface (AWS CLI) testen.

Console

So testen Sie die -Funktion

1. Melden Sie sich bei der CloudFront Konsole unter an https://console.aws.amazon.com/cloudfront/v4/home#/functions und wählen Sie die Seite Funktionen aus.
2. Wählen Sie die Funktion aus, die Sie testen möchten.
3. Wählen Sie die Registerkarte Test.
4. Stellen Sie sicher, dass das korrekte Ereignis angezeigt wird. Um vom aktuell angezeigten Ereignis zu wechseln, wählen Sie im Feld Testereignis auswählen ein anderes Ereignis aus.
5. Wählen Sie Testfunktion. Die Konsole zeigt die Ausgabe der Funktion, einschließlich Funktionsprotokollen und Rechenauslastung.

CLI

Sie können eine Funktion mit dem aws cloudfront test-function Befehl testen.

So testen Sie die -Funktion

1. Öffnen Sie ein Befehlszeilenfenster.
2. Führen Sie den folgenden Befehl aus demselben Verzeichnis aus, das die angegebene Datei enthält.

   In diesem Beispiel wird die `fileb://` Notation verwendet, um die Ereignisobjektdatei zu übergeben. Es sind Zeilenumbrüche enthalten, um den Befehl lesbarer zu machen.

```
aws cloudfront test-function \
    --name MaxAge \
    --if-match ETVABCEXAMPLE \
    --event-object fileb://event-maxage-test01.json \
    --stage DEVELOPMENT
```

   Hinweise

   - Sie verweisen mit ihrem Namen und ETag (im `if-match` Parameter) auf die Funktion. Sie verweisen auf das Ereignisobjekt anhand seines Speicherorts in Ihrem Dateisystem.

- Die Phase kann `DEVELOPMENT` oder `LIVE` sein.

Wenn der Befehl erfolgreich ausgeführt wurde, wird die Ausgabe folgendermaßen angezeigt.

```
TestResult:
  ComputeUtilization: '21'
  FunctionErrorMessage: ''
  FunctionExecutionLogs: []
  FunctionOutput: '{"response":{"headers":{"cloudfront-functions":
{"value":"generated-by-CloudFront-Functions"},"location":{"value":"https://
aws.amazon.com/cloudfront/"}},"statusDescription":"Found","cookies":
{},"statusCode":302}}'
  FunctionSummary:
    FunctionConfig:
      Comment: MaxAge function
      Runtime: cloudfront-js-2.0
      KeyValueStoreAssociations= \
      {Quantity=1, \
      Items=[{KeyValueStoreARN='arn:aws:cloudfront::111122223333:key-value-
store/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'}]} \
    FunctionMetadata:
      CreatedTime: '2021-04-18T20:38:56.915000+00:00'
      FunctionARN: arn:aws:cloudfront::111122223333:function/MaxAge
      LastModifiedTime: '2023-17-20T10:38:57.057000+00:00'
      Stage: DEVELOPMENT
    Name: MaxAge
    Status: UNPUBLISHED
```

Hinweise

- `FunctionExecutionLogs` enthält eine Liste von Protokollzeilen, die die Funktion in `console.log()`-Anweisungen geschrieben hat (falls vorhanden).
- `ComputeUtilization`enthält Informationen zur Ausführung Ihrer Funktion. Siehe the section called “Verstehen Sie die Computernutzung”.
- `FunctionOutput` enthält das von der Funktion zurückgegebene Ereignisobjekt.

Verstehen Sie die Computernutzung

Rechenauslastung ist die Zeit, die die Ausführung der Funktion in Anspruch genommen hat, als Prozentsatz der maximal zulässigen Zeit. Zum Beispiel bedeutet ein Wert von 35, dass die Funktion in 35 % der maximal zulässigen Zeit abgeschlossen wurde.

Wenn eine Funktion kontinuierlich die maximal zulässige Zeit überschreitet, wird die Funktion CloudFront gedrosselt. Die folgende Liste zeigt, mit welcher Wahrscheinlichkeit eine Funktion basierend auf dem Wert der Rechenauslastung gedrosselt wird.

Rechenauslastungswert:

- 1 – 50 – Die Funktion liegt deutlich unter der maximal zulässigen Zeit und sollte ohne Drosselung ausgeführt werden.
- 51 – 70 – Die Funktion nähert sich der maximal zulässigen Zeit. Erwägen Sie die Optimierung des Funktionscodes.
- 71 — 100 — Die Funktion kommt der maximal zulässigen Zeit sehr nahe oder überschreitet sie. CloudFront wird diese Funktion wahrscheinlich drosseln, wenn Sie sie einer Verteilung zuordnen.

Funktionen aktualisieren

Sie können eine Funktion jederzeit aktualisieren. Die Änderungen werden nur an der Version der Funktion vorgenommen, die sich in der `DEVELOPMENT`-Phase befindet. Um die Updates von der `DEVELOPMENT` Stage nach zu kopieren`LIVE`, müssen Sie die Funktion veröffentlichen.

Sie können den Code einer Funktion in der CloudFront Konsole oder mit der AWS Command Line Interface (AWS CLI) aktualisieren.

Console

Um den Funktionscode zu aktualisieren

1. Melden Sie sich bei der CloudFront Konsole unter an https://console.aws.amazon.com/cloudfront/v4/home#/functions und wählen Sie die Seite Funktionen aus.

   Wählen Sie die zu aktualisierende Funktion aus.
2. Wählen Sie Bearbeiten und nehmen Sie die folgenden Änderungen vor:

   - Aktualisieren Sie alle Felder im Bereich „Details“.

- Ändern oder entfernen Sie den zugehörigen Schlüsselwertspeicher. Weitere Hinweise zu Schlüsselwertspeichern finden Sie unter the section called “ CloudFront KeyValueStore”.

- Ändern Sie den Funktionscode. Wählen Sie die Registerkarte Erstellen, nehmen Sie die Änderungen vor und wählen Sie dann Änderungen speichern, um die Änderungen am Code zu speichern.

CLI

So aktualisieren Sie den Funktionscode:

1. Öffnen Sie ein Befehlszeilenfenster.
2. Führen Sie den folgenden Befehl aus.

   In diesem Beispiel wird die `fileb://` Notation verwendet, um die Datei zu übergeben. Es sind Zeilenumbrüche enthalten, um den Befehl lesbarer zu machen.

```
aws cloudfront update-function \
    --name MaxAge \
    --function-config '{"Comment":"Max Age 2 years","Runtime":"cloudfront-
js-2.0","KeyValueStoreAssociations":{"Quantity":1,"Items":
[{"KeyValueStoreARN":"arn:aws:cloudfront::111122223333:key-value-store/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"}]}}' \
    --function-code fileb://function-max-age-v1.js \
    --if-match ETVABCEXAMPLE
```

> **Hinweise**
>
> - Sie können die Funktion sowohl anhand ihres Namens als auch ETag (im `if-match` Parameter) identifizieren. Stellen Sie sicher, dass Sie den aktuellen verwenden ETag. Sie können diesen Wert aus der DescribeFunctionAPI-Operation abrufen.
> - Sie müssen den `function-code` angeben, auch wenn Sie ihn nicht ändern möchten.

- Seien Sie vorsichtig mit der `function-config`. Sie sollten alles übergeben, was Sie in der Konfiguration beibehalten möchten. Gehen Sie insbesondere mit dem Schlüsselwertspeicher folgendermaßen vor:
  - Um die bestehende Schlüsselwertspeicherzuordnung (falls vorhanden) beizubehalten, geben Sie den Namen des vorhandenen Speichers an.
  - Um die Zuordnung zu ändern, geben Sie den Namen des neuen Schlüsselwertspeichers an.
  - Um die Zuordnung zu entfernen, lassen Sie den `KeyValueStoreAssociations` Parameter weg.

Wenn der Befehl erfolgreich ausgeführt wurde, wird die Ausgabe folgendermaßen angezeigt.

```
ETag: ETVXYZEXAMPLE
FunctionSummary:
  FunctionConfig:
    Comment: Max Age 2 years \
    Runtime: cloudfront-js-2.0 \
    KeyValueStoreAssociations= \
      {Quantity=1, \
      Items=[{KeyValueStoreARN='arn:aws:cloudfront::111122223333:key-value-
store/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'}]} \
  FunctionMetadata: \
    CreatedTime: '2021-04-18T20:38:56.915000+00:00' \
    FunctionARN: arn:aws:cloudfront::111122223333:function/MaxAge \
    LastModifiedTime: '2023-12-19T23:41:15.389000+00:00' \
    Stage: DEVELOPMENT \
  Name: MaxAge \
  Status: UNPUBLISHED
```

Die meisten Informationen werden aus der Anfrage wiederholt. Weitere Informationen werden von CloudFront hinzugefügt.

Hinweise

- `ETag`— Dieser Wert ändert sich jedes Mal, wenn Sie den Schlüsselwertspeicher ändern.
- `FunctionARN`— Der ARN für Ihre CloudFront Funktion.

- `Stage`— Die Bühne für die Funktion (`LIVE`oder`DEVELOPMENT`).
- `Status`— Der Status der Funktion (`PUBLISHED`oder`UNPUBLISHED`).

Funktionen veröffentlichen

Wenn Sie Ihre Funktion veröffentlichen, wird die Funktion von der `DEVELOPMENT` Bühne auf die `LIVE` Bühne kopiert.

Wenn Cache-Verhalten nicht mit der Funktion verknüpft sind, können Sie sie durch das Veröffentlichen mit einem Cache-Verhalten verknüpfen. Sie können Cache-Verhalten nur Funktionen zuordnen, die sich in der `LIVE`-Phase befinden.

Important

- Wir empfehlen Ihnen, die Funktion vor dem Veröffentlichen zu testen.
- Nachdem Sie die Funktion veröffentlicht haben, beginnen alle Cache-Verhaltensweisen, die mit dieser Funktion verknüpft sind, automatisch, die neu veröffentlichte Kopie zu verwenden, sobald die Bereitstellung der Distributionen abgeschlossen ist.

Sie können eine Funktion in der CloudFront Konsole oder mit dem AWS CLI veröffentlichen.

Console

Um eine Funktion zu veröffentlichen

1. Melden Sie sich bei der CloudFront Konsole unter an https://console.aws.amazon.com/cloudfront/v4/home#/functions und wählen Sie die Seite Funktionen aus.
2. Wählen Sie die zu aktualisierende Funktion aus.
3. Wählen Sie den Tab „Veröffentlichen“ und dann „Veröffentlichen“. Wenn Ihre Funktion bereits mit einem oder mehreren Cache-Verhalten verknüpft ist, wählen Sie „Veröffentlichen und aktualisieren“.
4. (Optional) Um die Verteilungen anzuzeigen, die mit der Funktion verknüpft sind, wählen Sie Assoziierte CloudFront Verteilungen aus, um diesen Abschnitt zu erweitern.

Bei erfolgreicher Ausführung erscheint oben auf der Seite ein Banner mit der Aufschrift Erfolgreich ***Function name***veröffentlicht. Sie können auch den Tab Build und dann Live auswählen, um die Live-Version des Funktionscodes anzuzeigen.

CLI

Um eine Funktion zu veröffentlichen

1. Öffnen Sie ein Befehlszeilenfenster.
2. Führen Sie den Befehl aws cloudfront publish-function aus. Im Beispiel werden Zeilenumbrüche bereitgestellt, um das Beispiel besser lesbar zu machen.

```
aws cloudfront publish-function \
    --name MaxAge \
    --if-match ETVXYZEXAMPLE
```

Wenn der Befehl erfolgreich ausgeführt wurde, wird die Ausgabe folgendermaßen angezeigt.

```
FunctionSummary:
  FunctionConfig:
    Comment: Max Age 2 years
    Runtime: cloudfront-js-2.0
  FunctionMetadata:
    CreatedTime: '2021-04-18T21:24:21.314000+00:00'
    FunctionARN: arn:aws:cloudfront::111122223333:function/ExampleFunction
    LastModifiedTime: '2023-12-19T23:41:15.389000+00:00'
    Stage: LIVE
  Name: MaxAge
  Status: UNASSOCIATED
```

Funktionen mit Verteilungen verknüpfen

Um eine Funktion mit einer Verteilung zu verwenden, verknüpfen Sie die Funktion mit einem oder mehreren Cache-Verhalten in der Verteilung. Sie können eine Funktion mit verschiedenen Cache-Verhaltensweisen in mehreren Verteilungen verknüpfen.

Sie können eine Funktion den folgenden Verhaltensweisen zuordnen:

- Ein vorhandenes Cache-Verhalten
- Ein neues Cache-Verhalten in einer vorhandenen Distribution

- Ein neues Cache-Verhalten in einer neuen Distribution

Wenn Sie eine Funktion mit einem Cache-Verhalten verknüpfen, müssen Sie einen Ereignistyp auswählen. Der Ereignistyp bestimmt, wann die Funktion CloudFront ausgeführt wird.

Sie können die folgenden Ereignistypen wählen:

- Viewer-Anfrage — Die Funktion wird ausgeführt, wenn sie eine Anfrage von einem Viewer CloudFront erhält.
- Antwort des Betrachters — Die Funktion wird ausgeführt, bevor eine Antwort an den Betrachter CloudFront zurückgegeben wird.

Sie können in Functions keine Ereignistypen verwenden, die auf den Ursprung gerichtet sind (ursprüngliche Anfrage und ursprüngliche Antwort). CloudFront Stattdessen können Sie Lambda @Edge verwenden. Weitere Informationen finden Sie unter CloudFront Ereignisse, die eine Lambda @Edge -Funktion auslösen können.

> **Note**
>
> Bevor Sie eine Funktion verknüpfen, müssen Sie sie in der `LIVE`-Phase veröffentlichen.

Sie können eine Funktion mit einer Distribution in der CloudFront Konsole oder mit AWS Command Line Interface (AWS CLI) verknüpfen. Das folgende Verfahren zeigt, wie eine Funktion mit einem vorhandenen Cache-Verhalten verknüpft wird.

Console

Um eine Funktion einem vorhandenen Cache-Verhalten zuzuordnen

1. Melden Sie sich bei der CloudFront Konsole unter an https://console.aws.amazon.com/cloudfront/v4/home#/functions und wählen Sie die Seite Funktionen aus.
2. Wählen Sie die Funktion aus, die Sie zuordnen möchten.
3. Wählen Sie auf der Funktionsseite die Registerkarte Veröffentlichen.
4. Wählen Sie die Funktion „Veröffentlichen“.
5. Wählen Sie Add association. Wählen Sie im daraufhin angezeigten Dialogfeld eine Verteilung, einen Ereignistyp und/oder ein Cache-Verhalten aus.

Wählen Sie für den Ereignistyp aus, wann diese Funktion ausgeführt werden soll:

- Viewer-Anfrage — Führen Sie die Funktion jedes Mal aus, CloudFront wenn eine Anfrage eingeht.
- Antwort des Betrachters — Führen Sie die Funktion jedes Mal aus, wenn eine Antwort CloudFront zurückgegeben wird.

6. Um die Konfiguration zu speichern, wählen Sie Zuordnung hinzufügen.

CloudFront ordnet die Verteilung der Funktion zu. Warten Sie ein paar Minuten, bis die zugehörige Verteilung die Bereitstellung abgeschlossen hat. Sie können auf der Seite mit den Funktionsdetails die Option Verteilung anzeigen wählen, um den Fortschritt zu überprüfen.

CLI

Um eine Funktion einem vorhandenen Cache-Verhalten zuzuordnen

1. Öffnen Sie ein Befehlszeilenfenster.
2. Geben Sie den folgenden Befehl ein, um die Verteilungskonfiguration für die Distribution zu speichern, deren Cache-Verhalten Sie einer Funktion zuordnen möchten. Dieser Befehl speichert die Verteilungskonfiguration in einer Datei mit dem Namen `dist-config.yaml`. Um diesen Befehl zu verwenden, gehen Sie folgendermaßen vor:

   - Ersetzen Sie *`DistributionID`* durch die ID der Verteilung.
   - Führen Sie den Befehl in einer Zeile aus. Im Beispiel werden Zeilenumbrüche bereitgestellt, um das Beispiel besser lesbar zu machen.

```
aws cloudfront get-distribution-config \
    --id DistributionID \
    --output yaml > dist-config.yaml
```

   Wenn der Befehl erfolgreich ist, wird keine Ausgabe AWS CLI zurückgegeben.

3. Öffnen Sie die Datei mit dem Namen`dist-config.yaml`, die Sie erstellt haben. Bearbeiten Sie die Datei, indem Sie die folgenden Änderungen vornehmen.

   a. Benennen Sie das `ETag`-Feld in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

b. Suchen Sie im Cache-Verhalten das Objekt mit dem Namen `FunctionAssociations`. Aktualisieren Sie dieses Objekt, um eine Funktionszuordnung hinzuzufügen. Die YAML-Syntax für eine Funktionszuordnung sieht wie im folgenden Beispiel aus.

   - Das folgende Beispiel zeigt den Ereignistyp Betrachteranfrage (Auslöser). Um den Ereignistyp Betrachterantwort zu verwenden, ersetzen Sie `viewer-request` durch `viewer-response`.
   - Ersetzen Sie *`arn:aws:cloudfront::111122223333:function/ExampleFunction`* durch den Amazon-Ressourcennamen (ARN) der Funktion, die Sie diesem Cache-Verhalten zuordnen. Um die Funktion ARN zu erhalten, können Sie den Befehl aws cloudfront list-functions verwenden.

```
FunctionAssociations:
  Items:
    - EventType: viewer-request
      FunctionARN: arn:aws:cloudfront::111122223333:function/ExampleFunction
  Quantity: 1
```

c. Nachdem Sie diese Änderungen vorgenommen haben, speichern Sie die Datei.

4. Verwenden Sie den folgenden Befehl, um die Verteilung zu aktualisieren und die Funktionszuordnung hinzuzufügen. Um diesen Befehl zu verwenden, gehen Sie folgendermaßen vor:

   - Ersetzen Sie *`DistributionID`* durch die ID der Verteilung.
   - Führen Sie den Befehl in einer Zeile aus. Im Beispiel werden Zeilenumbrüche bereitgestellt, um das Beispiel besser lesbar zu machen.

```
aws cloudfront update-distribution \
    --id DistributionID \
    --cli-input-yaml file://dist-config.yaml
```

Wenn der Befehl erfolgreich ist, sehen Sie eine Ausgabe wie die folgende, die die Verteilung beschreibt, die gerade mit der Funktionszuordnung aktualisiert wurde. Die folgende Beispielausgabe wird zur besseren Lesbarkeit abgeschnitten.

```
Distribution:
  ARN: arn:aws:cloudfront::111122223333:distribution/EBEDLT3BGRBBW
```

```
  ... truncated ...
  DistributionConfig:
    ... truncated ...
    DefaultCacheBehavior:
      ... truncated ...
      FunctionAssociations:
        Items:
        - EventType: viewer-request
          FunctionARN: arn:aws:cloudfront::111122223333:function/ExampleFunction
        Quantity: 1
      ... truncated ...
  DomainName: d111111abcdef8.cloudfront.net
  Id: EDFDVBD6EXAMPLE
  LastModifiedTime: '2021-04-19T22:39:09.158000+00:00'
  Status: InProgress
ETag: E2VJGGQEG1JT8S
```

Der `Status` der Verteilung ändert sich in `InProgress` während der erneuten Bereitstellung der Verteilung. Wenn die neue Verteilungskonfiguration einen CloudFront Edge-Standort erreicht, verwendet dieser Edge-Standort die zugehörige Funktion. Wenn die Verteilung vollständig bereitgestellt ist, werden die `Status` Änderungen wieder zu geändert`Deployed`. Dies weist darauf hin, dass die zugehörige CloudFront Funktion an allen CloudFront Edge-Standorten weltweit verfügbar ist. Dies dauert in der Regel einige Minuten.

Amazon CloudFront KeyValueStore

CloudFront KeyValueStore ist ein sicherer, globaler Schlüsselwert-Datenspeicher mit niedriger Latenz, der Lesezugriff von CloudFront Functions aus ermöglicht und so eine erweiterte, anpassbare Logik an den Edge-Standorten ermöglicht. CloudFront

Mit CloudFront KeyValueStore können Sie unabhängig voneinander Aktualisierungen am Funktionscode und an den mit einer Funktion verknüpften Daten vornehmen. Diese Trennung vereinfacht den Funktionscode und erleichtert die Aktualisierung von Daten, ohne dass Codeänderungen vorgenommen werden müssen.

> **Note**
>
> Um diese CloudFront Funktion verwenden zu können CloudFront KeyValueStore, muss sie JavaScript Runtime 2.0 verwenden.

Im Folgenden finden Sie das allgemeine Verfahren für die Verwendung von Schlüssel-Wert-Paaren:

- Erstellen Sie Schlüsselwertspeicher und füllen Sie sie mit einer Reihe von Schlüssel-Wert-Paaren. Sie können Ihre Key Value Stores zu einem Amazon S3 S3-Bucket hinzufügen oder sie manuell eingeben.
- Ordnen Sie die Key-Value-Stores Ihrer CloudFront Funktion zu.
- Verwenden Sie in Ihrem Funktionscode den Namen des Schlüssels, um entweder den mit dem Schlüssel verknüpften Wert abzurufen oder um zu bestimmen, ob ein Schlüssel existiert. Weitere Hinweise zur Verwendung von Schlüssel-Wert-Paaren im Funktionscode und zu Hilfsmethoden finden Sie unter. the section called “Hilfsmethoden für Schlüsselwertspeicher”

Anwendungsfälle

Sie können Schlüssel-Wert-Paare für die folgenden Beispiele verwenden:

- URL-Umschreibungen oder Weiterleitungen — Das Schlüssel-Wert-Paar kann die umgeschriebene URL oder die Weiterleitung enthalten. URLs URLs
- A/B-Tests und Feature-Flags — Sie können eine Funktion zur Durchführung von Experimenten erstellen, indem Sie einer bestimmten Version Ihrer Website einen bestimmten Prozentsatz des Traffics zuweisen.
- Zugriffsautorisierung — Sie können eine Zugriffskontrolle implementieren, um Anfragen auf der Grundlage der von Ihnen definierten Kriterien und der in einem Schlüsselwertspeicher gespeicherten Daten zuzulassen oder abzulehnen.

Unterstützte Formate für Werte

Sie können den Wert in einem Schlüssel-Wert-Paar in einem der folgenden Formate speichern:

- String
- Bytekodierte Zeichenfolge
- JSON

Sicherheit

Die CloudFront Funktion und alle ihre Schlüsselwerte speichern Daten werden sicher behandelt, und zwar wie folgt:

- CloudFront verschlüsselt alle Schlüsselwertspeicher im Ruhezustand und während der Übertragung (beim Lesen oder Schreiben in die Schlüsselwertspeicher), wenn Sie die CloudFront KeyValueStoreAPI-Operationen aufrufen.
- Wenn die Funktion ausgeführt wird, CloudFront entschlüsselt jedes Schlüssel-Wert-Paar im Speicher an den Edge-Standorten. CloudFront

Informationen zu den ersten Schritten finden Sie in den folgenden Themen. CloudFront KeyValueStore

Themen

- Arbeiten Sie mit dem Schlüsselwertspeicher
- Arbeiten Sie mit Schlüsselwertdaten

- Weitere Informationen zu den ersten Schritten finden Sie im CloudFront KeyValueStore AWS Blogbeitrag Introducing Amazon. CloudFront KeyValueStore

Arbeiten Sie mit dem Schlüsselwertspeicher

Sie müssen einen Schlüsselwertspeicher für die Schlüssel-Wert-Paare erstellen, die Sie in CloudFront Functions verwenden möchten.

Nachdem Sie die Schlüsselwertspeicher erstellt und Schlüssel-Wert-Paare hinzugefügt haben, können Sie die Schlüsselwerte in Ihrem Funktionscode verwenden. CloudFront

Informationen zu den ersten Schritten finden Sie in den folgenden Themen:

Themen

- Erstellen Sie einen Schlüsselwertspeicher
- Ordnen Sie einer Funktion einen Schlüsselwertspeicher zu
- Aktualisieren Sie einen Schlüsselwertspeicher
- Rufen Sie einen Verweis auf einen Schlüsselwertspeicher ab
- Löschen Sie einen Schlüsselwertspeicher
- Dateiformat für Schlüssel-Wert-Paare

> **Note**
>
> Die JavaScript Runtime 2.0 enthält einige Hilfsmethoden für die Arbeit mit Schlüsselwerten im Funktionscode. Weitere Informationen finden Sie unter the section called “Hilfsmethoden für Schlüsselwertspeicher”.

Erstellen Sie einen Schlüsselwertspeicher

Sie können einen Schlüsselwertspeicher und seine Schlüssel-Wert-Paare gleichzeitig erstellen. Sie können jetzt auch einen leeren Schlüsselwertspeicher erstellen und die Schlüssel-Wert-Paare später hinzufügen.

> **Note**
>
> Wenn Sie Ihre Datenquelle aus einem Amazon S3 S3-Bucket angeben, müssen Sie über die `s3:GetBucketLocation` Berechtigungen `s3:GetObject` und für diesen Bucket verfügen. Wenn Sie nicht über diese Berechtigungen verfügen, CloudFront kann Ihr Key Value Store nicht erfolgreich erstellt werden.

Entscheiden Sie, ob Sie beim Erstellen des Schlüsselwertspeichers gleichzeitig Schlüssel-Wert-Paare hinzufügen möchten. Sie können Ihre Schlüssel-Wert-Paare mithilfe der CloudFront Konsole, CloudFront der API oder importieren. AWS SDKs Sie können Ihre Datei mit Schlüssel-Wert-Paaren jedoch nur importieren, wenn Sie den Schlüssel-Wert-Speicher zum ersten Mal erstellen.

Informationen zum Erstellen einer Datei mit Schlüssel-Wert-Paaren finden Sie unter. Dateiformat für Schlüssel-Wert-Paare

Console

So erstellen Sie einen Schlüsselwertspeicher

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Seite Funktionen in der CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home#/functions.
2. Wählen Sie die KeyValueStoresRegisterkarte und dann Erstellen aus KeyValueStore.
3. Geben Sie für den Schlüsselwertspeicher einen Namen und optional eine Beschreibung ein.

4. Vollständige S3-URI:

   - Wenn Sie eine Datei mit Schlüssel-Wert-Paaren haben, geben Sie den Pfad zu dem Amazon S3 S3-Bucket ein, in dem Sie die Datei gespeichert haben.
   - Lassen Sie dieses Feld leer, wenn Sie die Schlüssel-Wert-Paare manuell eingeben möchten.

5. Wählen Sie Create (Erstellen) aus. Der Schlüsselwertspeicher ist jetzt vorhanden.

   Die Seite mit den Details für den neuen Schlüsselwertspeicher wird angezeigt. Die Informationen auf der Seite umfassen die ID und den ARN des Schlüsselwertspeichers.

   - Die ID ist eine zufällige Zeichenfolge, die für Sie einzigartig ist AWS-Konto.
   - Der ARN hat folgende Syntax:

     ```
     AWS-Konto:key-value-store/the key value stores ID
     ```

6. Sehen Sie sich den Abschnitt Schlüssel-Wert-Paare an. Wenn Sie eine Datei importiert haben, werden in diesem Abschnitt einige Schlüssel-Wert-Paare angezeigt. Sie haben die folgenden Möglichkeiten:

   - Wenn Sie eine Datei importiert haben, können Sie weitere Werte auch manuell hinzufügen.
   - Wenn Sie keine Datei aus einem Amazon S3 S3-Bucket importiert haben und jetzt Schlüssel-Wert-Paare hinzufügen möchten, können Sie den nächsten Schritt ausführen.
   - Sie können diesen Schritt überspringen und die Schlüssel-Wert-Paare später hinzufügen.

7. So fügen Sie die Paare jetzt hinzu:

   a. Wählen Sie Schlüssel-Wert-Paare hinzufügen aus.

   b. Wählen Sie Paar hinzufügen und geben Sie einen Namen und einen Wert ein. Wiederholen Sie diesen Schritt, um weitere Paare hinzuzufügen.

   c. Wenn Sie fertig sind, wählen Sie Änderungen speichern, um alle Schlüssel-Wert-Paare im Schlüsselwertspeicher zu speichern. Wählen Sie im daraufhin angezeigten Dialogfeld „Fertig“ aus.

8. Um den Schlüsselwertspeicher jetzt einer Funktion zuzuordnen, füllen Sie den Abschnitt Zugeordnete Funktionen aus. Weitere Informationen finden Sie unter ??? oder ???.

   Sie können die Funktion auch später zuordnen, entweder von dieser Detailseite des Schlüsselwertspeichers oder von der Detailseite der Funktion aus.

AWS CLI

Um einen Schlüsselwertspeicher zu erstellen

- Führen Sie den folgenden Befehl aus, um einen Schlüsselwertspeicher zu erstellen und die Schlüssel-Wert-Paare aus einem Amazon S3 S3-Bucket zu importieren.

```
aws cloudfront create-key-value-store \
    --name=keyvaluestore1 \
    --comment="This is my key value store file" \
    --import-source=SourceType=S3,SourceARN=arn:aws:s3:::amzn-s3-demo-
bucket1/kvs-input.json
```

Antwort

```
{
    "ETag": "ETVABCEXAMPLE",
    "Location": "https://cloudfront.amazonaws.com/2020-05-31/key-value-store/
arn:aws:cloudfront::123456789012:key-value-store/8aa76c93-3198-462c-aaf6-
example",
    "KeyValueStore": {
        "Name": "keyvaluestore1",
        "Id": "8aa76c93-3198-462c-aaf6-example",
        "Comment": "This is my key value store file",
        "ARN": "arn:aws:cloudfront::123456789012:key-value-
store/8aa76c93-3198-462c-aaf6-example",
        "Status": "PROVISIONING",
        "LastModifiedTime": "2024-08-06T22:19:10.813000+00:00"
    }
}
```

API

Um einen Key-Value-Store zu erstellen

1. Verwenden der CloudFront CreateKeyValueStoreBetrieb. Für diese Operation sind mehrere Parameter erforderlich:
   - A `name` des Schlüsselwertspeichers.
   - Ein `comment`-Parameter mit einem Kommentar.

- Ein `import-source` Parameter, mit dem Sie Schlüssel-Wert-Paare aus einer Datei importieren können, die in einem Amazon S3 S3-Bucket gespeichert ist. Sie können nur dann aus einer Datei importieren, wenn Sie den Key-Value-Store zum ersten Mal erstellen. Hinweise zur Dateistruktur finden Sie unterthe section called “Dateiformat für Schlüssel-Wert-Paare”.

Die Operationsantwort enthält die folgenden Informationen:

- Die in der Anforderung übergebenen Werte, einschließlich des von Ihnen zugewiesenen Namens.
- Daten wie die Erstellungszeit.
- Ein `ETag` (z. B.`ETVABCEXAMPLE`), der ARN, der den Namen des Schlüsselwertspeichers enthält (z. B.`arn:aws:cloudfront::123456789012:key-value-store/keyvaluestore1`).

  Sie werden eine Kombination aus dem`ETag`, dem ARN und dem Namen verwenden, um programmgesteuert mit dem Schlüsselwertspeicher zu arbeiten.

Status des Speichers von Schlüsselwerten

Wenn Sie einen Schlüsselwertspeicher erstellen, kann der Datenspeicher die folgenden Statuswerte haben.

Wert	Beschreibung
Bereitstellung	Der Schlüsselwertspeicher wurde erstellt und verarbeitet CloudFront derzeit die von Ihnen angegebene Datenquelle.
Bereit	Der Schlüsselwertspeicher wurde erstellt und die von Ihnen angegebene Datenquelle wurde CloudFront erfolgreich verarbeitet.
Der Import ist fehlgeschlagen	CloudFront konnte die von Ihnen angegebene Datenquelle nicht verarbeiten. Dieser Status kann angezeigt werden, wenn Ihr Dateiformat nicht gültig ist oder wenn es die Größenbeschränkung überschreitet. Weitere Informationen finden Sie unter Dateiformat für Schlüssel-Wert-Paare.

Ordnen Sie einer Funktion einen Schlüsselwertspeicher zu

Nachdem Sie Ihren Schlüsselwertspeicher erstellt haben, können Sie Ihre Funktion aktualisieren, um sie Ihrem Schlüsselwertspeicher zuzuordnen. Sie müssen diese Zuordnung vornehmen, um die Schlüssel-Wert-Paare aus diesem Speicher in dieser Funktion verwenden zu können. Die folgenden Regeln gelten:

- Eine Funktion kann nur einen Schlüsselwertspeicher haben
- Sie können denselben Schlüsselwertspeicher mehreren Funktionen zuordnen

Console

Um einen Schlüsselwertspeicher mit einer Funktion zu verknüpfen

1. Melden Sie sich bei der CloudFront Konsole unter an https://console.aws.amazon.com/cloudfront/v4/home#/functions und wählen Sie die Seite Funktionen aus.
2. Wählen Sie den Namen der Funktion.
3. Gehen Sie zum KeyValueStore Bereich Associate und wählen Sie Associate existing aus KeyValueStore.
4. Wählen Sie den Schlüsselwertspeicher aus, der die Schlüssel-Wert-Paare in der Funktion enthält, und wählen Sie dann Assoziieren aus. KeyValueStore

   CloudFront ordnet den Speicher sofort der Funktion zu. Sie müssen die Funktion nicht speichern.
5. Um einen anderen Schlüsselwertspeicher anzugeben, wählen Sie Zugeordnet aktualisieren KeyValueStore, wählen Sie einen anderen Namen für den Schlüsselwertspeicher aus und klicken Sie dann auf Zuordnen KeyValueStore.

Weitere Informationen finden Sie unter the section called “Funktionen aktualisieren”.

AWS CLI

Um einen Schlüsselwertspeicher mit einer Funktion zu verknüpfen

- Führen Sie den folgenden Befehl aus, um die *MaxAge* Funktion zu aktualisieren und eine Schlüsselwertspeicher-Ressource zuzuordnen.

```
aws cloudfront update-function \
```

```
    --name MaxAge \
    --function-config '{"Comment":"Max Age 2 years","Runtime":"cloudfront-
js-2.0","KeyValueStoreAssociations":{"Quantity":1,"Items":
[{"KeyValueStoreARN":"arn:aws:cloudfront::123456789012:key-value-
store/8aa76c93-3198-462c-aaf6-example"}]}}' \
    --function-code fileb://function-max-age-v1.js \
    --if-match ETVABCEXAMPLE
```

- Um einen Schlüsselwertspeicher mit einer Funktion zu verknüpfen, geben Sie den `KeyValueStoreAssociations` Parameter und den Schlüsselwertspeicher ARN an.
- Um die Zuordnung zu ändern, geben Sie einen anderen Schlüsselwertspeicher-ARN an.
- Um die Zuordnung zu entfernen, entfernen Sie den `KeyValueStoreAssociations` Parameter.

Weitere Informationen finden Sie unter the section called “Funktionen aktualisieren”.

API

Um einen Schlüsselwertspeicher mit einer Funktion zu verknüpfen

- Verwenden Sie die API-Operation UpdateFunction. Weitere Informationen finden Sie unter the section called “Funktionen aktualisieren”.

Hinweise

- Wenn Sie einen Schlüsselwertspeicher ändern, ohne die Schlüssel-Wert-Paare zu ändern, oder wenn Sie nur die Schlüssel-Wert-Paare im Schlüsselwertspeicher ändern, müssen Sie den Schlüssel-Wert-Speicher nicht erneut zuordnen. Sie müssen die Funktion auch nicht erneut veröffentlichen.

  Wir empfehlen jedoch, die Funktion zu testen, um sicherzustellen, dass sie erwartungsgemäß funktioniert. Weitere Informationen finden Sie unter Funktionen testen.
- Sie können sich alle Funktionen ansehen, die bestimmte Schlüsselwertspeicher verwenden. Wählen Sie in der CloudFront Konsole die Detailseite für den Schlüsselwertspeicher aus.

Aktualisieren Sie einen Schlüsselwertspeicher

Wenn Sie einen Schlüsselwertspeicher aktualisieren, können Sie die Schlüssel-Wert-Paare oder die Zuordnung zwischen dem Schlüsselwertspeicher und der Funktion ändern.

Console

Um einen Schlüsselwertspeicher zu aktualisieren

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Seite Funktionen in der CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home#/functions.
2. Wählen Sie die Registerkarte KeyValueStores aus.
3. Wählen Sie den Schlüsselwertspeicher aus, den Sie aktualisieren möchten.
   - Um die Schlüssel-Wert-Paare zu aktualisieren, wählen Sie im Abschnitt Schlüssel-Wert-Paare die Option Bearbeiten aus. Sie können beliebige Schlüssel-Wert-Paare hinzufügen oder löschen. Sie können auch den Wert für ein vorhandenes Schlüssel-Wert-Paar ändern. Wenn Sie fertig sind, wählen Sie Änderungen speichern aus.
   - Um die Zuordnung für diesen Schlüsselwertspeicher zu aktualisieren, wählen Sie Gehe zu Funktionen. Weitere Informationen finden Sie unter the section called “Ordnen Sie einer Funktion einen Schlüsselwertspeicher zu”.

AWS CLI

Um einen Schlüsselwertspeicher zu aktualisieren

1. Ändern Sie die Schlüssel-Wert-Paare — Sie können weitere Schlüssel-Wert-Paare hinzufügen, ein oder mehrere Schlüssel-Wert-Paare löschen und den Wert eines vorhandenen Schlüssel-Wert-Paares ändern. Weitere Informationen finden Sie unter Arbeiten Sie mit Schlüsselwertdaten.
2. Ändern Sie die Funktionszuordnung für den Schlüsselwertspeicher — Informationen zum Aktualisieren der Funktion und der Zuordnung für den Schlüsselwertspeicher finden Sie unter. Ordnen Sie einer Funktion einen Schlüsselwertspeicher zu

> **Tip**
>
> Sie benötigen den ARN des Schlüsselwertspeichers. Weitere Informationen finden Sie unter the section called “Rufen Sie einen Verweis auf einen Schlüsselwertspeicher ab”.

API

So aktualisieren Sie einen Schlüsselwertspeicher

1. Ändern Sie die Schlüssel-Wert-Paare — Sie können weitere Schlüssel-Wert-Paare hinzufügen, ein oder mehrere Schlüssel-Wert-Paare löschen und den Wert eines vorhandenen Schlüssel-Wert-Paares ändern. Weitere Informationen finden Sie unter Arbeiten Sie mit Schlüsselwertdaten.
2. Ändern Sie die Funktionszuordnung für den Schlüsselwertspeicher — Verwenden Sie die API-Operation, um die Funktionszuordnung für den Schlüsselwertspeicher zu aktualisieren. UpdateFunction Weitere Informationen finden Sie unter the section called “Funktionen aktualisieren”.

> **Tip**
>
> Sie benötigen den ARN des Schlüsselwertspeichers. Weitere Informationen finden Sie unter the section called “Rufen Sie einen Verweis auf einen Schlüsselwertspeicher ab”.

Rufen Sie einen Verweis auf einen Schlüsselwertspeicher ab

Um programmgesteuert mit den Schlüsselwertspeichern arbeiten zu können, benötigen Sie den `ETag` und den Namen des Schlüsselwertspeichers.

Um beide Werte abzurufen, können Sie die AWS Command Line Interface (AWS CLI) oder die CloudFront API verwenden.

AWS CLI

Um den Schlüsselwert abzurufen, speichern Sie die Referenz

1. Um eine Liste von Schlüsselwertspeichern zurückzugeben, führen Sie den folgenden Befehl aus: Suchen Sie den Namen des Schlüsselwertspeichers, den Sie ändern möchten.

```
aws cloudfront list-key-value-stores
```

2. Suchen Sie in der Antwort den Namen des gewünschten Schlüsselwertspeichers.

   Antwort

```
{
    "KeyValueStoreList": {
        "Items": [
            {
                "Name": "keyvaluestore3",
                "Id": "37435e19-c205-4271-9e5c-example3",
                "ARN": "arn:aws:cloudfront::123456789012:key-value-
store/37435e19-c205-4271-9e5c-example3",
                "Status": "READY",
                "LastModifiedTime": "2024-05-08T14:50:18.876000+00:00"
            },
            {
                "Name": "keyvaluestore2",
                "Id": "47970d59-6408-474d-b850-example2",
                "ARN": "arn:aws:cloudfront::123456789012:key-value-
store/47970d59-6408-474d-b850-example2",
                "Status": "READY",
                "LastModifiedTime": "2024-05-30T21:06:22.113000+00:00"
            },
            {
                "Name": "keyvaluestore1",
                "Id": "8aa76c93-3198-462c-aaf6-example",
                "ARN": "arn:aws:cloudfront::123456789012:key-value-
store/8aa76c93-3198-462c-aaf6-example",
                "Status": "READY",
                "LastModifiedTime": "2024-08-06T22:19:30.510000+00:00"
            }
        ]
    }
```

```
}
```

3. Führen Sie den folgenden Befehl aus, um den `ETag` für den angegebenen Schlüsselwertspeicher zurückzugeben.

```
aws cloudfront describe-key-value-store \
    --name=keyvaluestore1
```

Antwort

```
{
    "ETag": "E3UN6WX5RRO2AG",
    "KeyValueStore": {
        "Name": "keyvaluestore1",
        "Id": "8aa76c93-3198-462c-aaf6-example",
        "Comment": "This is an example KVS",
        "ARN": "arn:aws:cloudfront::123456789012:key-value-
store/8aa76c93-3198-462c-aaf6-example",
        "Status": "READY",
        "LastModifiedTime": "2024-08-06T22:19:30.510000+00:00"
    }
}
```

API

Um die Referenz zum Schlüsselwertspeicher abzurufen

1. Verwenden der CloudFront ListKeyValueStoresAPI-Vorgang zur Rückgabe einer Liste von Schlüsselwertspeichern. Suchen Sie den Namen des Schlüsselwertspeichers, den Sie ändern möchten.
2. Verwenden der CloudFront DescribeKeyValueStoreAPI-Vorgang und geben Sie den Namen des Schlüsselwertspeichers an, den Sie im vorherigen Schritt zurückgegeben haben.

Die Antwort enthält eine UUID, den ARN des Schlüsselwertspeichers und den `ETag` des Schlüsselwertspeichers.

- Ein`ETag`, wie `E3UN6WX5RRO2AG`
- Die UUID ist 128 Bit lang, wie `8aa76c93-3198-462c-aaf6-example`

- Der ARN enthält die AWS-Konto Zahl, die Konstante `key-value-store` und die UUID, wie im folgenden Beispiel:

  ```
  arn:aws:cloudfront::123456789012:key-value-store/8aa76c93-3198-462c-aaf6-example
  ```

Weitere Hinweise zu diesem `DescribeKeyValueStore` Vorgang finden Sie unter. the section called “Über den CloudFront KeyValueStore”

Löschen Sie einen Schlüsselwertspeicher

Sie können Ihren Key Value Store mithilfe der CloudFront Amazon-Konsole oder API löschen.

Console

Um einen Key Value Store zu löschen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Seite Funktionen in der CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home#/functions.
2. Wählen Sie den Namen der Funktion.
3. Überprüfen Sie im KeyValueStore Abschnitt Zugeordnet, ob der Funktion ein Schlüsselwertspeicher zugeordnet ist. Ist dies der Fall, entfernen Sie die Zuordnung, indem Sie „Zuordnung trennen“ KeyValueStore und anschließend „Zuordnung entfernen“ wählen.
4. Wählen Sie im Navigationsbereich die Seite Funktionen und dann die KeyValueStoresRegisterkarte aus.
5. Wählen Sie den Schlüsselwertspeicher aus, den Sie löschen möchten, und klicken Sie dann auf Löschen.

AWS CLI

Um einen Schlüsselwertspeicher zu löschen

1. Ruft den `ETag` und den Namen der Schlüsselwertspeicher ab. Weitere Informationen finden Sie unter the section called “Rufen Sie einen Verweis auf einen Schlüsselwertspeicher ab”.
2. Überprüfen Sie, ob der Schlüsselwertspeicher einer Funktion zugeordnet ist. Wenn dies der Fall ist, entfernen Sie die Zuordnung. Weitere Informationen zu diesen beiden Schritten finden Sie unter ???.

3. Wenn Sie den Namen und den Schlüsselwertspeicher kennen und `ETag` dieser nicht mehr mit einer Funktion verknüpft ist, können Sie ihn löschen.

   Führen Sie den folgenden Befehl aus, um den angegebenen Schlüsselwertspeicher zu löschen.

   ```
   aws cloudfront delete-key-value-store \
       --name=keyvaluestore1 \
       --if-match=E3UN6WX5RRO2AG
   ```

API

Um einen Schlüsselwertspeicher zu löschen

1. Ruft den `ETag` und den Namen der Schlüsselwertspeicher ab. Weitere Informationen finden Sie unter the section called “Rufen Sie einen Verweis auf einen Schlüsselwertspeicher ab”.
2. Überprüfen Sie, ob der Schlüsselwertspeicher einer Funktion zugeordnet ist. Wenn dies der Fall ist, entfernen Sie die Zuordnung. Weitere Informationen zu diesen beiden Schritten finden Sie unter ???.
3. Um den Schlüsselwertspeicher zu löschen, verwenden Sie CloudFront DeleteKeyValueStoreAPI-Vorgang.

Dateiformat für Schlüssel-Wert-Paare

Wenn Sie eine UTF-8-kodierte Datei erstellen, verwenden Sie das folgende JSON-Format:

```
{
  "data":[
    {
      "key":"key1",
      "value":"value"
    },
    {
      "key":"key2",
      "value":"value"
    }
  ]
}
```

Ihre Datei darf keine doppelten Schlüssel enthalten. Wenn Sie in Ihrem Amazon S3 S3-Bucket eine ungültige Datei angegeben haben, können Sie die Datei aktualisieren, um alle Duplikate zu entfernen, und dann erneut versuchen, Ihren Key Value Store zu erstellen.

Weitere Informationen finden Sie unter Erstellen Sie einen Schlüsselwertspeicher.

> **Note**
>
> Die Datei für Ihre Datenquelle und ihre Schlüssel-Wert-Paare haben die folgenden Beschränkungen:
>
> - Dateigröße – 5 MB
> - Schlüsselgröße – 512 Zeichen
> - Wertgröße – 1 024 Zeichen

Arbeiten Sie mit Schlüsselwertdaten

In diesem Thema wird beschrieben, wie Schlüssel-Wert-Paare zu einem vorhandenen Schlüssel-Wert-Speicher hinzugefügt werden. Informationen zum Einbeziehen von Schlüssel-Wert-Paaren bei der ersten Erstellung der Schlüssel-Wert-Speicher finden Sie unter. the section called “Erstellen Sie einen Schlüsselwertspeicher”

Themen

- Arbeiten Sie mit Schlüssel-Wert-Paaren (Konsole)
- Über den CloudFront KeyValueStore
- Arbeiten Sie mit Schlüssel-Wert-Paaren ()AWS CLI
- Arbeiten Sie mit Schlüssel-Wert-Paaren (API)

Arbeiten Sie mit Schlüssel-Wert-Paaren (Konsole)

Sie können die CloudFront Konsole verwenden, um mit Ihren Schlüssel-Wert-Paaren zu arbeiten.

Um mit Schlüssel-Wert-Paaren zu arbeiten

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Seite Funktionen in der CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home#/functions
2. Wählen Sie die Registerkarte KeyValueStores aus.

3. Wählen Sie den Schlüsselwertspeicher aus, den Sie bearbeiten möchten.
4. Wählen Sie im Abschnitt Schlüssel-Wert-Paare die Option Bearbeiten aus.
5. Sie können ein Schlüssel-Wert-Paar hinzufügen, ein Schlüssel-Wert-Paar löschen oder den Wert für ein vorhandenes Schlüssel-Wert-Paar ändern.
6. Wenn Sie fertig sind, wählen Sie Änderungen speichern aus.

Über den CloudFront KeyValueStore

> **Tip**
>
> Die CloudFront KeyValueStore API ist ein globaler Dienst, der Signature Version 4A (Sigv4A) zur Authentifizierung verwendet. Für die Verwendung temporärer Anmeldeinformationen mit SigV4a sind Sitzungstoken der Version 2 erforderlich. Weitere Informationen finden Sie unter Verwenden temporärer Anmeldeinformationen mit der CloudFront KeyValueStore API.

Wenn Sie die AWS Command Line Interface (AWS CLI) oder Ihren eigenen Code verwenden, um die CloudFront KeyValueStore API aufzurufen, lesen Sie die folgenden Abschnitte.

Wenn Sie mit einem Schlüsselwertspeicher und seinen Schlüssel-Wert-Paaren arbeiten, hängt der Service, den Sie aufrufen, von Ihrem Anwendungsfall ab:

- Verwenden Sie den Dienst, um mit Schlüssel-Wert-Paaren in einem vorhandenen Schlüssel-Wert-Speicher zu arbeiten. CloudFront KeyValueStore
- Verwenden Sie den Dienst, um Schlüsselwertpaare in den Schlüsselwertspeicher aufzunehmen, wenn Sie den Schlüsselwertspeicher zum ersten Mal erstellen. CloudFront

Sowohl die CloudFront API als auch die CloudFront KeyValueStore API haben einen `DescribeKeyValueStore` Vorgang. Sie rufen sie aus verschiedenen Gründen an. Informationen zu den Unterschieden finden Sie in der folgenden Tabelle.

	CloudFront DescribeKeyValueStore API	CloudFront KeyValueStore DescribeKeyValueStore API
Daten über den Schlüsselwertspeicher	Gibt Daten zurück, z. B. den Status und das	Gibt Daten über den Inhalt der Speicherr

	CloudFront DescribeKeyValueStore API	CloudFront KeyValueStore DescribeKeyValueStore API
	Datum, an dem der Schlüsselwertspeicher selbst zuletzt geändert wurde.	essource zurück — die Schlüssel-Wert-Paare im Speicher und die Größe des Inhalts.
Daten, die den Schlüsselwertspeicher identifizieren	Gibt an`ETag`, die UUID und den ARN des Schlüsselwertspeichers zurück.	Gibt ein `ETag` und den ARN des Schlüssel wertspeichers zurück.

Hinweise

- Jeder DescribeKeyValueStore Operation gibt einen anderen zurück`ETag`. Sie `ETags` sind nicht austauschbar.
- Wenn Sie einen API-Vorgang aufrufen, um eine Aktion abzuschließen, müssen Sie das `ETag` von der entsprechenden API aus angeben. In dem CloudFront KeyValueStore DeleteKeyVorgang geben Sie beispielsweise den an`ETag`, den Sie von der CloudFront KeyValueStore DescribeKeyValueStoreOperation.
- Wenn Sie Ihre CloudFront Funktionen mithilfe von aufrufen CloudFront KeyValueStore, werden die Werte im Schlüsselwertspeicher während des Aufrufs der Funktion nicht aktualisiert oder geändert. Aktualisierungen werden zwischen den Aufrufen einer Funktion verarbeitet.

Arbeiten Sie mit Schlüssel-Wert-Paaren ()AWS CLI

Sie können die folgenden AWS Command Line Interface Befehle für ausführen. CloudFront KeyValueStore

Inhalt

- Beschreiben Sie einen Schlüsselwertspeicher
- Erstellen Sie ein Schlüssel-Wert-Paar
- Löschen Sie ein Schlüssel-Wert-Paar
- Aktualisieren Sie Schlüssel-Wert-Paare

Schlüssel-Wert-Paare auflisten

Führen Sie den folgenden Befehl aus, um Schlüssel-Wert-Paare in Ihrem Schlüssel-Wert-Speicher aufzulisten.

```
aws cloudfront-keyvaluestore list-keys \
    --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example
```

Antwort

```
{
    "Items": [
        {
            "Key": "key1",
            "Value": "value1"
        }
    ]
}
```

Rufen Sie Schlüssel-Wert-Paare ab

Führen Sie den folgenden Befehl aus, um ein Schlüssel-Wert-Paar in Ihrem Schlüsselwertspeicher abzurufen.

```
aws cloudfront-keyvaluestore get-key \
    --key=key1 \
    --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example
```

Antwort

```
{
    "Key": "key1",
    "Value": "value1",
```

```
    "ItemCount": 1,
    "TotalSizeInBytes": 11
}
```

Beschreiben Sie einen Schlüsselwertspeicher

Führen Sie den folgenden Befehl aus, um einen Schlüsselwertspeicher zu beschreiben.

```
aws cloudfront-keyvaluestore describe-key-value-store \
    --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example
```

Antwort

```
{
    "ETag": "KV1F83G8C2ARO7P",
    "ItemCount": 1,
    "TotalSizeInBytes": 11,
    "KvsARN": "arn:aws:cloudfront::123456789012:key-value-store/37435e19-
c205-4271-9e5c-example",
    "Created": "2024-05-08T07:48:45.381000-07:00",
    "LastModified": "2024-08-05T13:50:58.843000-07:00",
    "Status": "READY"
}
```

Erstellen Sie ein Schlüssel-Wert-Paar

Führen Sie den folgenden Befehl aus, um ein Schlüssel-Wert-Paar in Ihrem Schlüssel-Wert-Speicher zu erstellen.

```
aws cloudfront-keyvaluestore put-key \
    --if-match=KV1PA6795UKMFR9 \
    --key=key2 \
    --value=value2 \
    --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example
```

Antwort

```
{
    "ETag": "KV13V1IB3VIYZZH",
    "ItemCount": 3,
```

```
    "TotalSizeInBytes": 31
}
```

Löschen Sie ein Schlüssel-Wert-Paar

Führen Sie den folgenden Befehl aus, um ein Schlüssel-Wert-Paar zu löschen.

```
aws cloudfront-keyvaluestore delete-key \
    --if-match=KV13V1IB3VIYZZH \
    --key=key1 \
    --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example
```

Ausgabe

```
{
    "ETag": "KV1VC38T7YXB528",
    "ItemCount": 2,
    "TotalSizeInBytes": 22
}
```

Aktualisieren Sie Schlüssel-Wert-Paare

Sie können den `update-keys` Befehl verwenden, um mehr als ein Schlüssel-Wert-Paar zu aktualisieren. Um beispielsweise ein vorhandenes Schlüssel-Wert-Paar zu löschen und ein anderes zu erstellen, führen Sie den folgenden Befehl aus.

```
aws cloudfront-keyvaluestore update-keys \
    --if-match=KV2EUQ1WTGCTBG2 \
    --kvs-arn=arn:aws:cloudfront::123456789012:key-value-store/37435e19-c205-4271-9e5c-
example \
    --deletes '[{"Key":"key2"}]' \
    --puts '[{"Key":"key3","Value":"value3"}]'
```

Antwort

```
{
    "ETag": "KV3AEGXETSR30VB",
    "ItemCount": 3,
    "TotalSizeInBytes": 28
}
```

Arbeiten Sie mit Schlüssel-Wert-Paaren (API)

Folgen Sie diesem Abschnitt, um programmatisch mit Ihren Schlüssel-Wert-Paaren zu arbeiten.

Inhalt

Holen Sie sich einen Verweis auf einen Schlüsselwertspeicher

Wenn Sie die CloudFront KeyValueStore API verwenden, um einen Schreibvorgang aufzurufen, müssen Sie den ARN und den `ETag` des Schlüsselwertspeichers angeben. Gehen Sie wie folgt vor, um diese Daten abzurufen:

Um einen Verweis auf einen Schlüsselwertspeicher abzurufen

1. Verwenden der CloudFront ListKeyValueStoresAPI-Operation zum Abrufen einer Liste von Schlüsselwertspeichern. Suchen Sie den Schlüsselwertspeicher, den Sie ändern möchten.
2. Verwenden der CloudFrontKeyValueStore DescribeKeyValueStore API-Betrieb und geben Sie den Schlüsselwertspeicher aus dem vorherigen Schritt an.

   Die Antwort umfasst den ARN und den `ETag` des Schlüsselwertspeichers.

   - Der ARN umfasst die AWS-Konto Zahl, die Konstante `key-value-store` und die UUID, wie im folgenden Beispiel:

     ```
     arn:aws:cloudfront::123456789012:key-value-store/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
     ```
   - Und `ETag` das sieht wie das folgende Beispiel aus:

     ```
     ETVABCEXAMPLE2
     ```

Ändern Sie Schlüssel-Wert-Paare in einem Schlüsselwertspeicher

Sie können den Schlüsselwertspeicher angeben, der das Schlüssel-Wert-Paar enthält, das Sie aktualisieren möchten.

Sehen Sie sich die folgenden CloudFront KeyValueStore API-Operationen an:

- CloudFrontKeyValueStore DeleteKey— Löscht ein Schlüssel-Wert-Paar
- CloudFrontKeyValueStore GetKey— Gibt ein Schlüssel-Wert-Paar zurück
- CloudFrontKeyValueStore ListKeys— Gibt eine Liste von Schlüssel-Wert-Paaren zurück
- CloudFrontKeyValueStore PutKey— Sie können die folgenden Aufgaben ausführen:
  - Erstellen Sie ein Schlüssel-Wert-Paar in einem Schlüsselwertspeicher, indem Sie einen neuen Schlüsselnamen und einen neuen Schlüsselwert angeben.
  - Legen Sie einen anderen Wert in einem vorhandenen Schlüssel-Wert-Paar fest, indem Sie einen vorhandenen Schlüsselnamen und einen neuen Schlüsselwert angeben.
- CloudFrontKeyValueStore UpdateKeys— Sie können eine oder mehrere der folgenden Aktionen in einem all-or-nothing Vorgang ausführen:
  - Löschen Sie ein oder mehrere Schlüssel-Wert-Paare
  - Erstellen Sie ein oder mehrere neue Schlüssel-Wert-Paare
  - Legen Sie einen anderen Wert in einem oder mehreren vorhandenen Schlüssel-Wert-Paaren fest

Beispielcode für CloudFront KeyValueStore

Example

Der folgende Code zeigt Ihnen, wie Sie den `DescribeKeyValueStore` API-Vorgang für einen Schlüsselwertspeicher aufrufen.

```
const {
  CloudFrontKeyValueStoreClient,
  DescribeKeyValueStoreCommand,
} = require("@aws-sdk/client-cloudfront-keyvaluestore");

require("@aws-sdk/signature-v4-crt");

(async () => {
  try {
    const client = new CloudFrontKeyValueStoreClient({
      region: "us-east-1"
    });
    const input = {
      KvsARN: "arn:aws:cloudfront::123456789012:key-value-store/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
    };
    const command = new DescribeKeyValueStoreCommand(input);
```

```
    const response = await client.send(command);
  } catch (e) {
    console.log(e);
  }
})();
```

Personalisieren Sie am Rand mit Lambda @Edge

Lambda @Edge ist eine Erweiterung von. AWS Lambda Lambda @Edge ist ein Rechenservice, mit dem Sie Funktionen ausführen können, mit denen Sie die von Amazon bereitgestellten Inhalte anpassen können CloudFront . Sie können Node.js- oder Python-Funktionen in der Lambda-Konsole in einer AWS-Region, USA Ost (Nord-Virginia), erstellen.

Nachdem Sie die Funktion erstellt haben, können Sie mithilfe der Lambda-Konsole oder der CloudFront Lambda-Konsole Trigger hinzufügen, sodass die Funktionen AWS an Orten ausgeführt werden, die sich näher am Betrachter befinden, ohne Server bereitzustellen oder zu verwalten. Optional können Sie Lambda- und CloudFront API-Operationen verwenden, um Ihre Funktionen und Trigger programmgesteuert einzurichten.

Lambda@Edge skaliert automatisch zwischen einigen wenigen Anforderungen pro Tag und ein paar Tausend Anforderungen pro Sekunde. Die Verarbeitung von Anfragen an AWS Orten, die sich näher am Betrachter befinden, anstatt auf Originalservern, reduziert die Latenz erheblich und verbessert die Benutzererfahrung.

> **Note**
>
> Lambda @Edge wird bei gRPC-Anfragen nicht unterstützt. Weitere Informationen finden Sie unter. gRPC mit CloudFront Distributionen verwenden

Themen

- Testen und Debuggen von Lambda @Edge -Funktionen
- Lambda @Edge -Funktionen und Replikate löschen
- Lambda@Edge-Ereignisstruktur
- Arbeiten Sie mit Anfragen und Antworten
- Beispielfunktionen für Lambda@Edge

So funktioniert Lambda @Edge mit Anfragen und Antworten

Wenn Sie eine CloudFront Verteilung mit einer Lambda @Edge -Funktion verknüpfen, CloudFront fängt sie Anfragen und Antworten an CloudFront Edge-Standorten ab. Sie können Lambda-Funktionen ausführen, wenn die folgenden CloudFront Ereignisse eintreten:

- Wann CloudFront erhält er eine Anfrage von einem Zuschauer (Viewer-Anfrage)
- Bevor CloudFront eine Anfrage an den Ursprung weitergeleitet wird (ursprüngliche Anfrage)
- Wann CloudFront erhält er eine Antwort vom Ursprung (ursprüngliche Antwort)
- Before CloudFront gibt die Antwort an den Zuschauer zurück (Antwort des Betrachters)

Wenn Sie verwenden AWS WAF, wird die Lambda @Edge Viewer-Anfrage ausgeführt, nachdem alle AWS WAF Regeln angewendet wurden.

Weitere Informationen erhalten Sie unter Arbeiten Sie mit Anfragen und Antworten und Lambda@Edge-Ereignisstruktur.

Möglichkeiten, Lambda @Edge zu verwenden

Es gibt viele Verwendungsmöglichkeiten für die Lambda @Edge -Verarbeitung mit Ihrer CloudFront Amazon-Distribution, wie z. B. die folgenden Beispiele:

- Eine Lambda-Funktion kann Cookies untersuchen und neu schreiben, URLs sodass Benutzern verschiedene Versionen einer Website zum A/B Testen angezeigt werden.
- CloudFront kann den Zuschauern je nach verwendetem Gerät unterschiedliche Objekte zurückgeben, indem sie den `User-Agent` Header überprüft, der Informationen zu den Geräten enthält. CloudFront Kann beispielsweise je nach Bildschirmgröße des Geräts unterschiedliche Bilder zurückgeben. In ähnlicher Weise könnte die Funktion den Wert des `Referer` Headers berücksichtigen und veranlassen CloudFront , dass die Bilder an Bots mit der niedrigsten verfügbaren Auflösung zurückgegeben werden.

- Sie können Cookies auch auf andere Kriterien überprüfen. Wenn Sie beispielsweise auf einer Einzelhandels-Website, die Kleidung verkauft, Cookies verwenden, um anzugeben, welche Farbe ein Benutzer für eine Jacke ausgewählt hat, kann eine Lambda-Funktion die Anfrage so ändern, dass das Bild einer Jacke in der ausgewählten Farbe CloudFront zurückgegeben wird.
- Eine Lambda-Funktion kann HTTP-Antworten generieren, wenn CloudFront Viewer-Anfragen oder Origin-Request-Ereignisse auftreten.
- Eine Funktion kann Header oder Autorisierungstoken überprüfen und einen Header einfügen, um den Zugriff auf Ihre Inhalte zu kontrollieren, bevor die Anfrage an Ihren Ursprung CloudFront weitergeleitet wird.
- Eine Lambda-Funktion kann auch Netzwerkaufrufe an externe Ressourcen durchführen, um die Anmeldeinformationen von Benutzern zu bestätigen oder weitere Inhalte zum Anpassen einer Antwort abzurufen.

Weitere Informationen, einschließlich Beispielcode, finden Sie unter. Beispielfunktionen für Lambda@Edge

Weitere Informationen zum Einrichten von Lambda @Edge in der Konsole finden Sie unterTutorial: Erstellen Sie eine grundlegende Lambda @Edge -Funktion (Konsole).

Erste Schritte mit Lambda @Edge -Funktionen (Konsole)

Mit Lambda @Edge können Sie CloudFront Trigger verwenden, um eine Lambda-Funktion aufzurufen. Wenn Sie eine CloudFront Verteilung mit einer Lambda-Funktion verknüpfen, CloudFront fängt sie Anfragen und Antworten an CloudFront Edge-Standorten ab und führt die Funktion aus. Lambda-Funktionen können die Sicherheit verbessern oder Informationen in der Nähe Ihrer Zuschauer anpassen, um die Leistung zu verbessern.

Die folgende Liste bietet einen grundlegenden Überblick darüber, wie Lambda-Funktionen mit CloudFront erstellt und verwendet werden.

Überblick: Lambda-Funktionen erstellen und verwenden mit CloudFront

1. Erstellen Sie eine Lambda-Funktion in der Region USA Ost (Nord-Virginia).
2. Speichern und veröffentlichen Sie eine nummerierte Version der Funktion.

   Wenn Sie die Funktion ändern möchten, müssen Sie die $LATEST-Version der Funktion in der Region USA Ost (Nord-Virginia) bearbeiten. Bevor Sie sie dann einrichten, um damit zu arbeiten CloudFront, veröffentlichen Sie eine neue nummerierte Version.

3. Ordnen Sie der Funktion ein CloudFront Verteilungs- und Cache-Verhalten zu. Geben Sie dann ein oder mehrere CloudFront Ereignisse (Trigger) an, die die Ausführung der Funktion veranlassen. Sie können beispielsweise einen Trigger erstellen, damit die Funktion ausgeführt wird, wenn sie eine Anfrage von einem Viewer CloudFront erhält.
4. Wenn Sie einen Trigger erstellen, erstellt Lambda Repliken der Funktion an AWS Standorten auf der ganzen Welt.

> **Tip**
>
> Weitere Informationen finden Sie unter Funktionen erstellen und aktualisieren, Ereignisstruktur und Hinzufügen von CloudFront Triggern. Weitere Ideen und Codebeispiele finden Sie auch in Beispielfunktionen für Lambda@Edge.

Ein step-by-step Tutorial finden Sie unter dem folgenden Thema:

Themen

- Tutorial: Erstellen Sie eine grundlegende Lambda @Edge -Funktion (Konsole)

Tutorial: Erstellen Sie eine grundlegende Lambda @Edge -Funktion (Konsole)

Dieses Tutorial zeigt Ihnen, wie Sie mit Lambda @Edge beginnen können, indem Sie eine Beispielfunktion für Node.js erstellen und konfigurieren, die in CloudFront ausgeführt wird. In diesem Beispiel werden einer Antwort beim CloudFront Abrufen einer Datei HTTP-Sicherheitsheader hinzugefügt. (Dies kann die Sicherheit und den Datenschutz einer Website verbessern.)

Für dieses Tutorial benötigen Sie keine eigene Website. Wenn Sie sich jedoch dafür entscheiden, Ihre eigene Lambda @Edge -Lösung zu erstellen, folgen Sie ähnlichen Schritten und wählen aus denselben Optionen.

Themen

- Schritt 1: Registrieren für AWS-Konto
- Schritt 2: Erstellen einer CloudFront -Verteilung
- Schritt 3: Erstellen Ihrer Funktion
- Schritt 4: Fügen Sie einen CloudFront Trigger hinzu, um die Funktion auszuführen
- Schritt 5: Überprüfen, ob die Funktion funktioniert

- Schritt 6: Beheben von Problemen
- Schritt 7: Bereinigen der Ressourcen für Ihr Beispiel
- Ähnliche Informationen

Schritt 1: Registrieren für AWS-Konto

Falls Sie dies noch nicht getan haben, melden Sie sich für eine an AWS-Konto. Weitere Informationen finden Sie unter Melden Sie sich für eine an AWS-Konto.

Schritt 2: Erstellen einer CloudFront -Verteilung

Bevor Sie die Lambda @Edge -Beispielfunktion erstellen, müssen Sie über eine CloudFront Umgebung verfügen, mit der Sie arbeiten können und die einen Ursprung enthält, von dem aus Inhalte bereitgestellt werden können.

In diesem Beispiel erstellen Sie eine CloudFront Distribution, die einen Amazon S3 S3-Bucket als Ursprung für die Verteilung verwendet. Wenn Sie bereits über eine Umgebung verfügen, die Sie benutzen können, können Sie diesen Schritt überspringen.

Um eine CloudFront Distribution mit einem Amazon S3 S3-Ursprung zu erstellen

1. Erstellen Sie einen Amazon S3-Bucket mit einer oder zwei Dateien, z. B. Image-Dateien, als Beispielinhalt. Für Hilfe folgen Sie den Schritten unter Hochladen Ihrer Inhalte zu Amazon S3. Stellen Sie sicher, dass Sie Berechtigungen erteilen, um öffentliche Lesezugriff auf die Objekte in Ihrem Bucket zu gewähren.
2. Erstellen Sie eine CloudFront Distribution und fügen Sie Ihren S3-Bucket als Ursprung hinzu, indem Sie die Schritte CloudFront unter Web-Distribution erstellen befolgen. Wenn Sie bereits eine Verteilung haben, können Sie stattdessen den Bucket als Ursprung für diese Verteilung hinzufügen.

> **Tip**
>
> Notieren Sie sich die ID Ihrer Verteilung. Wenn Sie später in diesem Tutorial einen CloudFront Trigger für Ihre Funktion hinzufügen, müssen Sie die ID für Ihre Distribution in einer Dropdownliste auswählen — zum Beispiel. `E653W22221KDDL`

Schritt 3: Erstellen Ihrer Funktion

In diesem Schritt erstellen Sie eine Lambda-Funktion aus einer Blueprint-Vorlage in der Lambda-Konsole. Die Funktion fügt Code hinzu, um Sicherheitsheader in Ihrer CloudFront-Verteilung zu aktualisieren.

Eine Lambda-Funktion erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Lambda Konsole unter. https://console.aws.amazon.com/lambda/

   > **Important**
   >
   > Vergewissern Sie sich, dass Sie sich in der Region US-East-1 (Nord-Virginia) AWS-Region (US-East-1) befinden. Sie müssen sich in dieser Region befinden, um Lambda@Edge-Funktionen erstellen zu können.

2. Wählen Sie Funktion erstellen.
3. Wählen Sie auf der Seite Funktion erstellen die Option Blueprint verwenden aus, und filtern Sie dann nach den CloudFront Blueprints, indem Sie sie in das Suchfeld eingeben. **`cloudfront`**

   > **Note**
   >
   > CloudFront Blueprints sind nur in der Region US-East-1 (Nord-Virginia) (us-east-1) verfügbar.

4. Wählen Sie den Blueprint „HTTP-Antwort-Header ändern“ als Vorlage für Ihre Funktion aus.
5. Geben Sie folgende Informationen zu Ihrer Funktion ein:

   - Funktionsname — Geben Sie einen Namen für Ihre Funktion ein.
   - Ausführungsrolle — Wählen Sie aus, wie Sie die Berechtigungen für Ihre Funktion festlegen möchten. Um die empfohlene Standardvorlage für Lambda @Edge -Berechtigungsrichtlinien zu verwenden, wählen Sie Neue Rolle aus AWS Richtlinienvorlagen erstellen.
   - Rollenname — Geben Sie einen Namen für die Rolle ein, die durch die Richtlinienvorlage erstellt wird.
   - Richtlinienvorlagen — Lambda fügt automatisch die Richtlinienvorlage Basic Lambda @Edge -Berechtigungen hinzu, da Sie einen CloudFront Blueprint als Grundlage für Ihre Funktion ausgewählt haben. Diese Richtlinienvorlage fügt Ausführungsrollenberechtigungen hinzu,

mit denen CloudFront Sie Ihre Lambda-Funktion CloudFront an Standorten auf der ganzen Welt für Sie ausführen können. Weitere Informationen finden Sie unter Richten Sie IAM-Berechtigungen und -Rollen für Lambda @Edge ein.

6. Wählen Sie unten auf der Seite die Option Funktion erstellen aus.
7. Wählen Sie im daraufhin angezeigten Bereich Deploy to Lambda @Edge die Option Abbrechen aus. (Für dieses Tutorial müssen Sie den Funktionscode ändern, bevor Sie die Funktion in Lambda @Edge bereitstellen.)
8. Scrollen Sie auf der Seite nach unten zum Abschnitt Codequelle.
9. Ersetzen Sie den Vorlagencode durch eine Funktion, die die Sicherheits-Header ändert, die Ihren Ursprung zurückgibt. Beispielsweise könnten Sie Code wie den folgenden verwenden:

```
'use strict';
export const handler = (event, context, callback) => {

    //Get contents of response
    const response = event.Records[0].cf.response;
    const headers = response.headers;

    //Set new headers
    headers['strict-transport-security'] = [{key: 'Strict-Transport-Security',
 value: 'max-age= 63072000; includeSubdomains; preload'}];
    headers['content-security-policy'] = [{key: 'Content-Security-Policy', value:
 "default-src 'none'; img-src 'self'; script-src 'self'; style-src 'self'; object-
src 'none'"}];
    headers['x-content-type-options'] = [{key: 'X-Content-Type-Options', value:
 'nosniff'}];
    headers['x-frame-options'] = [{key: 'X-Frame-Options', value: 'DENY'}];
    headers['x-xss-protection'] = [{key: 'X-XSS-Protection', value: '1;
 mode=block'}];
    headers['referrer-policy'] = [{key: 'Referrer-Policy', value: 'same-origin'}];

    //Return modified response
    callback(null, response);
};
```

10. Wählen Sie Datei, Speichern, um den aktualisierten Code zu speichern.
11. Wählen Sie Bereitstellen.

Fahren Sie mit dem nächsten Abschnitt fort, um einen CloudFront Trigger zum Ausführen der Funktion hinzuzufügen.

Schritt 4: Fügen Sie einen CloudFront Trigger hinzu, um die Funktion auszuführen

Da Sie nun über eine Lambda-Funktion zum Aktualisieren von Sicherheitsheadern verfügen, konfigurieren Sie den CloudFront Trigger so, dass Ihre Funktion so ausgeführt wird, dass die Header zu jeder Antwort hinzugefügt werden, die vom Ursprung Ihrer Distribution CloudFront empfangen wird.

Um den CloudFront Trigger für Ihre Funktion zu konfigurieren

1. Wählen Sie in der Lambda-Konsole auf der Funktionsübersichtsseite für Ihre Funktion die Option Auslöser hinzufügen aus.
2. Wählen Sie für Trigger-Konfiguration die Option CloudFront.
3. Wählen Sie Deploy to Lambda @Edge.
4. Geben Sie im Bereich Deploy to Lambda @Edge unter Configure CloudFront trigger die folgenden Informationen ein:

   - Distribution — Die CloudFront Distribution-ID, die Ihrer Funktion zugeordnet werden soll. Wählen Sie in der Dropdownliste die Vertriebs-ID aus.
   - Cache-Verhalten — Das Cache-Verhalten, das mit dem Trigger verwendet werden soll. Behalten Sie in diesem Beispiel den Wert * bei, der das Standard-Cache-Verhalten Ihrer Verteilung bezeichnet. Weitere Informationen finden Sie unter Einstellungen für das Cache-Verhalten im Thema Referenz für alle Verteilungseinstellungen.
   - CloudFront event — Der Trigger, der angibt, wann Ihre Funktion ausgeführt wird. Wir möchten, dass die Security-Header-Funktion immer dann ausgeführt wird, wenn eine Antwort vom Ursprung CloudFront zurückgegeben wird. Wählen Sie in der Drop-down-Liste Origin-Antwort aus. Weitere Informationen finden Sie unter Trigger für eine Lambda @Edge -Funktion hinzufügen.
5. Aktivieren Sie das Kontrollkästchen Bereitstellung auf Lambda @Edge bestätigen.
6. Wählen Sie Deploy, um den Trigger hinzuzufügen und die Funktion AWS an Standorten weltweit zu replizieren.
7. Warten Sie, bis die Funktion repliziert wurde. Dies dauert in der Regel mehrere Minuten.

   Sie können überprüfen, ob die Replikation abgeschlossen ist, indem Sie die CloudFront-Konsole öffnen und sich Ihre Verteilung ansehen. Warten Sie, bis sich der Verteilungsstatus von

Bereitstellen auf Datum und Uhrzeit ändert. Dies bedeutet, dass Ihre Funktion repliziert wurde. Zur Überprüfung der Funktionstätigkeit befolgen Sie die Schritte im nächsten Abschnitt.

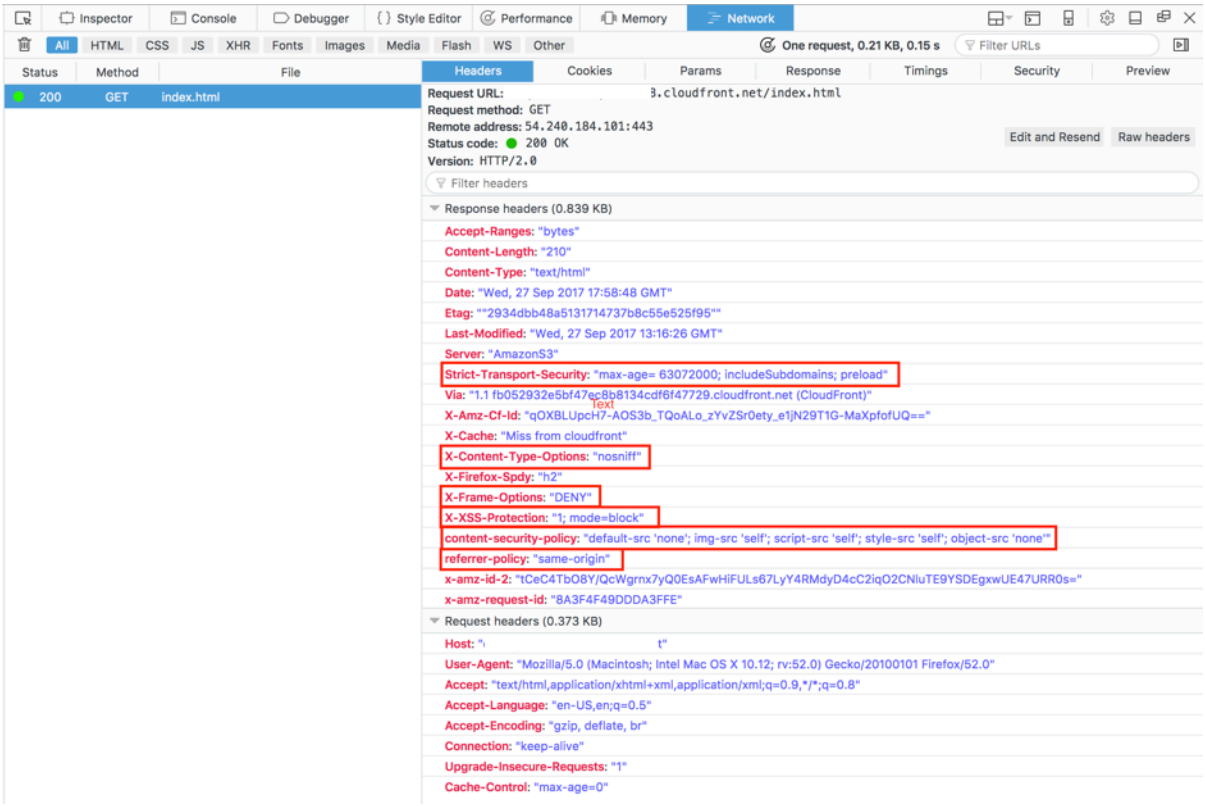
Schritt 5: Überprüfen, ob die Funktion funktioniert

Nachdem Sie Ihre Lambda-Funktion erstellt und einen Trigger konfiguriert haben, um sie für eine CloudFront Distribution auszuführen, überprüfen Sie, ob die Funktion das leistet, was Sie von ihr erwarten. In diesem Beispiel überprüfen wir die von CloudFront zurückgegebenen HTTP-Header, um sicherzustellen, dass die Sicherheits-Header hinzugefügt werden.

Überprüfen, ob Ihre Lambda@Edge-Funktion Sicherheitsheader hinzufügt

1. Geben Sie in einem Browser die URL für eine Datei in Ihrem S3-Bucket ein.
   Sie können beispielsweise eine URL wie die folgende Verwenden: `https://d111111abcdef8.cloudfront.net/image.jpg`.

   Weitere Informationen zum CloudFront Domainnamen, der in der Datei-URL verwendet werden soll, finden Sie unter. Passen Sie das URL-Format für Dateien an in CloudFront
2. Öffnen Sie die Web-Developer-Symbolleiste Ihres Browsers. Öffnen Sie beispielsweise in Ihrem Browserfenster in Chrome das Kontextmenü (Rechtsklick) und wählen Sie Inspect (Untersuchen) aus.
3. Wählen Sie die Registerkarte Network (Netzwerk) aus.
4. Laden Sie die Seite, um Ihr Image anzuzeigen, und wählen Sie dann auf der linken Seite eine HTTP-Anforderung. Sie sehen die HTTP-Header in einem separaten Fenster.
5. Sehen Sie sich die Liste der HTTP-Header an, um zu überprüfen, ob die erwarteten Sicherheitsheader in der Liste enthalten sind. Beispielsweise könnten Sie Header wie im folgenden Screenshot gezeigt sehen.



Wenn die Sicherheitsheader in Ihrem Header-Liste enthalten sind, dann ist das hervorragend! Sie haben erfolgreich Ihre erste Lambda@Edge-Funktion erstellt. Wenn CloudFront Rückgabefehler auftreten oder andere Probleme auftreten, fahren Sie mit dem nächsten Schritt fort, um die Probleme zu beheben.

Schritt 6: Beheben von Problemen

Wenn Fehler CloudFront zurückgegeben werden oder die Sicherheitsheader nicht wie erwartet hinzugefügt werden, können Sie die Ausführung Ihrer Funktion anhand von CloudWatch Protokollen untersuchen. Stellen Sie sicher, dass Sie die Protokolle verwenden, die an dem AWS Ort gespeichert sind, der dem Ort, an dem die Funktion ausgeführt wird, am nächsten liegt.

Wenn Sie sich die Datei beispielsweise von London aus ansehen, versuchen Sie, die Region in der CloudWatch Konsole auf Europa (London) zu ändern.

Überprüfen von CloudWatch -Protokollen für Ihre Lambda@Edge-Funktion

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudWatch Konsole unter https://console.aws.amazon.com/cloudwatch/.

2. Ändern Sie Region auf den Speicherort, der angezeigt wird, wenn Sie die Datei in Ihrem Browser anzeigen. Hier wird die Funktion ausgeführt.
3. Klicken Sie im linken Bereich auf Logs (Protokolle), um die Protokolle für Ihre Verteilung anzuzeigen.

Weitere Informationen finden Sie unter Überwachen Sie CloudFront Metriken mit Amazon CloudWatch.

Schritt 7: Bereinigen der Ressourcen für Ihr Beispiel

Wenn Sie einen Amazon S3 S3-Bucket und eine CloudFront Distribution nur für dieses Tutorial erstellt haben, löschen Sie die AWS Ressourcen, die Sie zugewiesen haben, sodass keine Gebühren mehr anfallen. Nachdem Sie Ihre AWS Ressourcen gelöscht haben, sind alle Inhalte, die Sie hinzugefügt haben, nicht mehr verfügbar.

Aufgaben

- Löschen des S3-Buckets
- Löschen Sie die Lambda-Funktion
- Löschen Sie die Distribution CloudFront

Löschen des S3-Buckets

Bevor Sie Ihren Amazon S3-Bucket löschen, stellen Sie sicher, dass die Protokollierung für den Bucket deaktiviert ist. Andernfalls werden AWS weiterhin Logs in Ihren Bucket geschrieben, während Sie ihn löschen.

Deaktivieren Sie die Protokollierung für einen Bucket wie folgt:

1. Öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie den Bucket aus, und wählen Sie dann Properties (Eigenschaften).
3. Wählen Sie unter Properties (Eigenschaften) Logging (Protokollierung) aus.
4. Deaktivieren Sie das Kontrollkästchen Enabled (Aktiviert).
5. Wählen Sie Save (Speichern) aus.

Jetzt können Sie Ihren Bucket löschen. Weitere Informationen erhalten Sie unter Löschen eines Buckets im Amazon Simple Storage Service-Konsolenbenutzerhandbuch.

Löschen Sie die Lambda-Funktion

Anweisungen zum Löschen der Lambda-Funktionsassoziation und optional der Funktion selbst finden Sie unterLambda @Edge -Funktionen und Replikate löschen.

Löschen Sie die Distribution CloudFront

Bevor Sie eine CloudFront Distribution löschen, müssen Sie sie deaktivieren. Eine deaktivierte Verteilung funktioniert nicht mehr und es fallen keine weiteren Kosten für sie an. Sie können eine deaktivierte Verteilung jederzeit wieder aktivieren. Nachdem Sie eine deaktivierte Verteilung gelöscht haben, ist sie nicht länger verfügbar.

Um eine CloudFront Distribution zu deaktivieren und zu löschen

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Klicken Sie mit der rechten Maustaste auf die Verteilung, die Sie deaktivieren möchten, und anschließend auf Disable (Deaktivieren).
3. Wenn Sie zur Bestätigung aufgefordert werden, wählen Sie Yes, Disable (Ja, deaktivieren).
4. Wählen Sie die deaktivierte Verteilung aus, und klicken Sie dann auf Delete (Löschen).
5. Wenn Sie zur Bestätigung aufgefordert werden, wählen Sie Yes, Delete.

Ähnliche Informationen

Nun, da Sie eine grundlegende Vorstellung davon haben, wie Lambda@Edge-Funktionen funktionieren, können Sie hier weitere Informationen erhalten:

- Beispielfunktionen für Lambda@Edge
- Bewährte Methoden für das Lambda @Edge -Design
- Reduzierung der Latenz und Verlagerung von Rechenleistung an den Edge mit Lambda @Edge

Richten Sie IAM-Berechtigungen und -Rollen für Lambda @Edge ein

Um Lambda @Edge zu konfigurieren, benötigen Sie die folgenden IAM-Berechtigungen und -Rollen für: AWS Lambda

- IAM-Berechtigungen — Mit diesen Berechtigungen können Sie Ihre Lambda-Funktion erstellen und sie Ihrer CloudFront Distribution zuordnen.

- Eine Lambda-Funktionsausführungsrolle (IAM-Rolle) — Die Lambda-Serviceprinzipale übernehmen diese Rolle, um Ihre Funktion auszuführen.
- Dienstgebundene Rollen für Lambda @Edge — Die dienstverknüpften Rollen ermöglichen es bestimmten, Lambda-Funktionen in Protokolldateien AWS-Services zu replizieren AWS-Regionen und deren Verwendung zu ermöglichen CloudWatch. CloudFront

IAM-Berechtigungen sind erforderlich, um Lambda @Edge -Funktionen mit Distributionen zu verknüpfen CloudFront

Zusätzlich zu den IAM-Berechtigungen, die Sie für Lambda benötigen, benötigen Sie die folgenden Berechtigungen, um Lambda-Funktionen Distributionen zuzuordnen: CloudFront

- `lambda:GetFunction`— Erteilt die Erlaubnis, Konfigurationsinformationen für Ihre Lambda-Funktion und eine vorsignierte URL zum Herunterladen einer `.zip` Datei abzurufen, die die Funktion enthält.
- `lambda:EnableReplication*`— Erteilt die Berechtigung für die Ressourcenrichtlinie, sodass der Lambda-Replikationsdienst den Funktionscode und die Konfiguration abrufen kann.
- `lambda:DisableReplication*`— Erteilt der Ressourcenrichtlinie die Berechtigung, sodass der Lambda-Replikationsdienst die Funktion löschen kann.

> **Important**
>
> Sie müssen das Sternchen (`*`) am Ende der Aktionen `lambda:EnableReplication*` und `lambda:DisableReplication*` hinzufügen.

- Geben Sie für die Ressource den ARN der Funktionsversion an, die Sie ausführen möchten, wenn ein CloudFront Ereignis eintritt, wie im folgenden Beispiel:

  `arn:aws:lambda:us-east-1:123456789012:function:TestFunction:2`
- `iam:CreateServiceLinkedRole`— Erteilt die Erlaubnis, eine serviceverknüpfte Rolle zu erstellen, in der Lambda @Edge Lambda-Funktionen repliziert. CloudFront Nachdem Sie Lambda @Edge zum ersten Mal konfiguriert haben, wird die serviceverknüpfte Rolle automatisch für Sie erstellt. Sie müssen diese Berechtigung nicht zu anderen Distributionen hinzufügen, die Lambda @Edge verwenden.
- `cloudfront:UpdateDistribution`oder `cloudfront:CreateDistribution` — Erteilt die Erlaubnis, eine Distribution zu aktualisieren oder zu erstellen.

Weitere Informationen finden Sie unter den folgenden Themen:

- Identity and Access Management für Amazon CloudFront
- Zugriffsberechtigungen für Lambda-Ressourcen im AWS Lambda Developer Guide

Funktionsausführungsrolle für Service-Prinzipale

Sie müssen eine IAM-Rolle erstellen, die die Dienstprinzipale `lambda.amazonaws.com` und die `edgelambda.amazonaws.com` Dienstprinzipale bei der Ausführung Ihrer Funktion übernehmen können.

> **Tip**
>
> Wenn Sie Ihre Funktion in der Lambda-Konsole erstellen, können Sie wählen, ob Sie mithilfe einer AWS Richtlinienvorlage eine neue Ausführungsrolle erstellen möchten. Dieser Schritt fügt automatisch die erforderlichen Lambda @Edge -Berechtigungen hinzu, um Ihre Funktion auszuführen. Siehe Schritt 5 im Tutorial: Eine einfache Lambda @Edge -Funktion erstellen.

Weitere Informationen zum manuellen Erstellen einer IAM-Rolle finden Sie unter Rollen erstellen und Richtlinien anhängen (Konsole) im IAM-Benutzerhandbuch.

Example Beispiel: Vertrauensrichtlinie für Rollen

Sie können diese Rolle auf der Registerkarte Trust Relationship in der IAM-Konsole hinzufügen. Fügen Sie diese Richtlinie nicht auf der Registerkarte „Berechtigungen“ hinzu.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": {
            "Service": [
               "lambda.amazonaws.com",
               "edgelambda.amazonaws.com"
            ]
         },
```

```
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Weitere Informationen zu den Berechtigungen, die Sie der Ausführungsrolle gewähren müssen, finden Sie unter Lambda-Ressourcenzugriffsberechtigungen im AWS Lambda Entwicklerhandbuch.

Hinweise

- Standardmäßig werden Daten in CloudWatch Logs geschrieben, wenn ein CloudFront Ereignis eine Lambda-Funktion auslöst. Wenn Sie diese Protokolle verwenden möchten, benötigt die Ausführungsrolle die Berechtigung, Daten in CloudWatch Logs zu schreiben. Sie können das Vordefinierte verwendenAWSLambdaBasicExecutionRole, um der Ausführungsrolle die Berechtigung zu erteilen.

  Weitere Informationen zu CloudWatch Protokollen finden Sie unterthe section called “Protokolle für Edge-Funktionen”.
- Wenn Ihr Lambda-Funktionscode auf andere AWS Ressourcen zugreift, z. B. auf das Lesen eines Objekts aus einem S3-Bucket, benötigt die Ausführungsrolle die Erlaubnis, diese Aktion auszuführen.

Serviceverknüpfte Rollen für Lambda@Edge

Lambda @Edge verwendet dienstverknüpfte IAM-Rollen. Eine serviceverknüpfte Rolle ist ein spezieller Typ von IAM-Rolle, der direkt mit einem Service verknüpft ist. Serviceverknüpfte Rollen werden vom Service vorab definiert und beinhalten alle Berechtigungen, die dieser zum Aufrufen anderer AWS -Services in Ihrem Namen benötigt.

Lambda @Edge verwendet die folgenden dienstverknüpften IAM-Rollen:

- AWSServiceRoleForLambdaReplicator – Lambda@Edge verwendet diese Rolle, um es Lambda@Edge zu ermöglichen, Funktionen in AWS-Regionen zu replizieren.

  Wenn Sie zum ersten Mal einen Lambda @Edge -Trigger hinzufügen CloudFront, AWSServiceRoleForLambdaReplicator wird automatisch eine Rolle mit dem Namen erstellt, damit Lambda @Edge Funktionen replizieren kann. AWS-Regionen Diese Rolle

ist erforderlich, um Lambda @Edge -Funktionen verwenden zu können. Der ARN für die AWSServiceRoleForLambdaReplicator Rolle sieht wie das folgende Beispiel aus:

```
arn:aws:iam::123456789012:role/aws-service-role/
replicator.lambda.amazonaws.com/AWSServiceRoleForLambdaReplicator
```

- AWSServiceRoleForCloudFrontLogger— CloudFront verwendet diese Rolle, um Protokolldateien in die Datenbank zu übertragen CloudWatch. Sie können Protokolldateien verwenden, um Lambda @Edge -Validierungsfehler zu debuggen.

  Die AWSServiceRoleForCloudFrontLogger Rolle wird automatisch erstellt, wenn Sie eine Lambda @Edge -Funktionsassoziation hinzufügen, an die Lambda @Edge -Fehlerprotokolldateien übertragen werden können CloudFront . CloudWatch Der ARN für die AWSServiceRoleForCloudFrontLogger-Rolle sieht so aus:

  ```
  arn:aws:iam::account_number:role/aws-service-role/
  logger.cloudfront.amazonaws.com/AWSServiceRoleForCloudFrontLogger
  ```

Eine serviceverknüpfte Rolle vereinfacht das Einrichten und Verwenden von Lambda@Edge, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. Lambda@Edge definiert die Berechtigungen seiner servicegebundenen Rollen. Nur Lambda@Edge kann die Rollen übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Sie können die Berechtigungsrichtlinie keiner anderen IAM-Entität zuordnen.

Sie müssen alle zugehörigen Ressourcen CloudFront oder Lambda @Edge -Ressourcen entfernen, bevor Sie eine serviceverknüpfte Rolle löschen können. Dies trägt zum Schutz Ihrer Lambda @Edge -Ressourcen bei, sodass Sie keine dienstbezogene Rolle entfernen, die weiterhin für den Zugriff auf aktive Ressourcen erforderlich ist.

Weitere Informationen zu serviceverknüpften Rollen finden Sie unter Dienstbezogene Rollen für CloudFront.

Serviceverknüpfte Rollenberechtigungen für Lambda@Edge

Lambda@Edge verwendet zwei servicegebundene Rollen. Diese heißen AWSServiceRoleForLambdaReplicator und AWSServiceRoleForCloudFrontLogger. In den folgenden Abschnitten werden die Berechtigungen für jede dieser Rollen beschrieben.

Inhalt

- Dienstbezogene Rollenberechtigungen für Logger CloudFront

Serviceverknüpfte Rollenberechtigungen für Lambda Replicator

Diese serviceverknüpfte Rolle ermöglicht Lambda das Replizieren von Lambda@Edge-Funktionen zu AWS-Regionen.

Die serviceverknüpfte Rolle AWSServiceRoleForLambdaReplicator vertraut dem Service `replicator.lambda.amazonaws.com`, sodass dieser die Rolle annehmen kann.

Die Rollenberechtigungsrichtlinie erlaubt Lambda@Edge die Durchführung der folgenden Aktionen für die angegebenen Ressourcen:

- `lambda:CreateFunction` auf `arn:aws:lambda:*:*:function:*`
- `lambda:DeleteFunction` auf `arn:aws:lambda:*:*:function:*`
- `lambda:DisableReplication` auf `arn:aws:lambda:*:*:function:*`
- `iam:PassRole` auf `all AWS resources`
- `cloudfront:ListDistributionsByLambdaFunction` auf `all AWS resources`

Dienstbezogene Rollenberechtigungen für Logger CloudFront

Diese dienstbezogene Rolle ermöglicht das CloudFront Pushen von Protokolldateien, CloudWatch sodass Sie Lambda @Edge -Validierungsfehler debuggen können.

Die serviceverknüpfte Rolle AWSServiceRoleForCloudFrontLogger vertraut dem Service `logger.cloudfront.amazonaws.com`, sodass dieser die Rolle annehmen kann.

Die Rollenberechtigungsrichtlinie ermöglicht es Lambda @Edge, die folgenden Aktionen für die angegebene `arn:aws:logs:*:*:log-group:/aws/cloudfront/*` Ressource durchzuführen:

- `logs:CreateLogGroup`
- `logs:CreateLogStream`
- `logs:PutLogEvents`

Sie müssen Berechtigungen konfigurieren, damit eine IAM-Entität (z. B. ein Benutzer, eine Gruppe oder eine Rolle) die mit dem Lambda@Edge-Service verknüpften Rollen löschen kann. Weitere Informationen finden Sie unter serviceverknüpfte Rollenberechtigung im IAM-Benutzerhandbuch.

Serviceverknüpfte Rollen für Lambda@Edge erstellen

Servicegebundene Rollen für Lambda@Edge werden in der Regel nicht manuell erstellt. In den folgenden Szenarien legt der Service die Rollen für Sie automatisch an:

- Wenn Sie zum ersten Mal einen Trigger erstellen, erstellt der Dienst die AWSServiceRoleForLambdaReplicator Rolle (sofern sie nicht bereits vorhanden ist). Diese Rolle ermöglicht es Lambda, Lambda @Edge -Funktionen auf zu replizieren. AWS-Regionen

  Wenn Sie die serviceverknüpfte Rolle löschen, wird die Rolle erneut erstellt, wenn Sie einen neuen Auslöser für Lambda@Edge in einer Verteilung hinzufügen.
- Wenn Sie eine CloudFront Distribution aktualisieren oder erstellen, die über eine Lambda @Edge -Zuordnung verfügt, erstellt der Dienst die AWSServiceRoleForCloudFrontLogger Rolle (sofern die Rolle noch nicht vorhanden ist). Diese Rolle ermöglicht es CloudFront , Ihre Protokolldateien per Push zu CloudWatch übertragen.

  Wenn Sie die dienstverknüpfte Rolle löschen, wird die Rolle erneut erstellt, wenn Sie eine CloudFront Distribution aktualisieren oder erstellen, die über eine Lambda @Edge -Zuordnung verfügt.

Um diese dienstbezogenen Rollen manuell zu erstellen, können Sie die folgenden Befehle AWS Command Line Interface ()AWS CLI ausführen:

So erstellen Sie die AWSServiceRoleForLambdaReplicator-Rolle

- Führen Sie den folgenden Befehl aus.

```
aws iam create-service-linked-role --aws-service-name
 replicator.lambda.amazonaws.com
```

So erstellen Sie die AWSServiceRoleForCloudFrontLogger-Rolle

- Führen Sie den folgenden Befehl aus.

```
aws iam create-service-linked-role --aws-service-name
 logger.cloudfront.amazonaws.com
```

Bearbeiten von serviceverknüpften Lambda@Edge-Rollen

Lambda @Edge erlaubt Ihnen nicht, die Rollen AWSServiceRoleForLambdaReplicator oder die AWSServiceRoleForCloudFrontLogger dienstbezogenen Rollen zu bearbeiten. Nachdem der Dienst eine dienstverknüpfte Rolle erstellt hat, können Sie den Namen der Rolle nicht mehr ändern, da verschiedene Entitäten möglicherweise auf die Rolle verweisen. Sie können mithilfe von IAM jedoch die Beschreibung der Rolle bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Unterstützt AWS-Regionen für dienstverknüpfte Lambda @Edge -Rollen

CloudFront unterstützt die Verwendung von dienstverknüpften Rollen für Lambda @Edge im Folgenden: AWS-Regionen

- USA Ost (Nord-Virginia) – `us-east-1`
- USA Ost (Ohio) – `us-east-2`
- USA West (Nordkalifornien) – `us-west-1`
- USA West (Oregon) – `us-west-2`
- Asia Pacific (Mumbai) – `ap-south-1`
- Asien-Pazifik (Seoul) – `ap-northeast-2`
- Asia Pacific (Singapore) – `ap-southeast-1`
- Asien-Pazifik (Sydney) – `ap-southeast-2`
- Asien-Pazifik (Tokio) – `ap-northeast-1`
- Europa (Frankfurt) – `eu-central-1`
- Europa (Ireland) – `eu-west-1`
- Europa (London) – `eu-west-2`
- South America (São Paulo) – `sa-east-1`

Schreiben und erstellen Sie eine Lambda @Edge -Funktion

Um Lambda @Edge zu verwenden, schreiben Sie den Code für Ihre AWS Lambda Funktion. Informationen zum Schreiben von Lambda @Edge -Funktionen finden Sie in den folgenden Ressourcen:

- Lambda@Edge-Ereignisstruktur— Verstehen Sie die mit Lambda @Edge zu verwendende Eventstruktur.

- Beispielfunktionen für Lambda@Edge— Beispielfunktionen, wie das A/B Testen und Generieren einer HTTP-Umleitung.

Das Programmiermodell für die Verwendung von Node.js oder Python mit Lambda @Edge entspricht der Verwendung von Lambda in einem. AWS-Region Weitere Informationen finden Sie unter Erstellen von Lambda-Funktionen mit Node.js oder Erstellen von Lambda-Funktionen mit Python im AWS Lambda Entwicklerhandbuch.

Fügen Sie in Ihre Lambda @Edge -Funktion den `callback` Parameter ein und geben Sie das entsprechende Objekt für Anfrage- oder Antwortereignisse zurück:

- Request events (Anfrageereignisse) – Schließen Sie das `cf.request`-Objekt in die Antwort ein.

  Wenn Sie eine Antwort generieren, schließen Sie das Objekt `cf.response` in die Antwort ein. Weitere Informationen finden Sie unter Generieren Sie HTTP-Antworten in Anforderungsauslösern.
- Response events (Antwortereignisse): Schließen Sie das `cf.response`-Objekt in die Antwort ein.

Nachdem Sie Ihren eigenen Code geschrieben oder eines der Beispiele verwendet haben, erstellen Sie die Funktion in Lambda. Informationen zum Erstellen einer Funktion oder zum Bearbeiten einer vorhandenen Funktion finden Sie in den folgenden Themen:

Themen

- Erstellen Sie eine Lambda @Edge -Funktion
- Eine Lambda-Funktion bearbeiten

Nachdem Sie die Funktion in Lambda erstellt haben, richten Sie Lambda so ein, dass die Funktion auf der Grundlage bestimmter CloudFront Ereignisse ausgeführt wird, die als Trigger bezeichnet werden. Weitere Informationen finden Sie unter Trigger für eine Lambda @Edge -Funktion hinzufügen.

Erstellen Sie eine Lambda @Edge -Funktion

Gehen Sie wie folgt vor AWS Lambda , um die Ausführung von Lambda-Funktionen einzurichten, die auf CloudFront Ereignissen basieren.

So erstellen Sie eine Lambda@Edge-Funktion

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Lambda Konsole unter https://console.aws.amazon.com/lambda/.
2. Wenn Sie bereits über eine oder mehrere Lambda-Funktionen verfügen, wählen Sie Create function.

   Wenn Sie nicht über Funktionen verfügen, wählen Sie Get Started Now.
3. Wählen Sie oben auf der Seite in der Liste „Region“ die Option US Ost (Nord-Virginia) aus.
4. Erstellen Sie eine Funktion mit Ihrem eigenen Code oder erstellen Sie eine Funktion, die mit einer CloudFront -Vorlage beginnt.

   - Um eine Funktion mit Ihrem eigenen Code zu erstellen, wählen Sie Author from scratch.
   - Um eine Liste mit Blueprints für anzuzeigen CloudFront, geben Sie Cloudfront in das Filterfeld ein und wählen Sie dann Enter.

     Wenn Sie eine dieser Vorlagen verwenden möchten, wählen Sie den Namen der entsprechenden Vorlage.
5. Geben Sie im Abschnitt Basic information folgende Werte ein:

   a. Name — Geben Sie einen Namen für Ihre Funktion ein.

   b. Rolle — Um schnell loszulegen, wählen Sie Neue Rolle aus Vorlage (n) erstellen. Sie können auch „Vorhandene Rolle auswählen“ oder „Benutzerdefinierte Rolle erstellen“ wählen und dann den Anweisungen folgen, um die Informationen für diesen Abschnitt zu vervollständigen.

   c. Rollenname — Geben Sie einen Namen für die Rolle ein.

   d. Richtlinienvorlagen — Wählen Sie Basic Edge Lambda-Berechtigungen aus.
6. Wenn Sie in Schritt 4 Author from scratch gewählt haben, fahren Sie mit Schritt 7 fort.

   Wenn Sie in Schritt 4 einen Blueprint ausgewählt haben, können Sie im Abschnitt Cloudfront einen Trigger erstellen, der diese Funktion einem Cache in einer CloudFront Distribution und einem Ereignis zuordnet. CloudFront Wir empfehlen, an dieser Stelle Remove zu wählen, damit es bei der Erstellung der Funktionen keinen Auslöser gibt. Sie können Auslöser zu einem späteren Zeitpunkt hinzufügen.

> **Tip**
>
> Wir empfehlen, dass Sie die Funktion testen und debuggen, bevor Sie Trigger hinzufügen. Wenn Sie jetzt einen Trigger hinzufügen, wird die Funktion ausgeführt, sobald Sie die Funktion erstellt haben. Die Replikation an AWS Standorte auf der ganzen Welt ist abgeschlossen und die entsprechende Distribution wird bereitgestellt.

7. Wählen Sie Funktion erstellen.

   Lambda erstellt zwei Versionen Ihrer Funktion: $LATEST und Version 1. Sie können nur die Version $LATEST bearbeiten, die Konsole zeit jedoch zunächst Version 1 an.

8. Um die Funktion zu bearbeiten, wählen Sie Version 1 oben auf der Seite, unter dem ARN für die Funktion. Wählen Sie anschließend auf der Registerkarte Versions die Option $LATEST. (Wenn Sie die Funktion verlassen haben und anschließend zurückgekehrt sind, lautet die Bezeichnung der Schaltfläche Qualifiers.)
9. Wählen Sie auf der Registerkarte Configuration den geeigneten Wert für Code entry type. Folgen Sie dann den Eingabeaufforderungen, um Ihren Code zu bearbeiten oder hochzuladen.
10. Wählen Sie den Wert für Runtime (Laufzeit) basierend auf dem Code der Funktion.
11. Fügen Sie im Bereich Tags geeignete Tags hinzu.
12. Wählen Sie Actions und dann Publish new version.
13. Geben Sie eine Beschreibung für die neue Version der Funktion ein.
14. Wählen Sie Publish.
15. Testen und debuggen Sie die Funktion. Weitere Informationen zum Testen in der Lambda-Konsole finden Sie unter Aufrufen einer Lambda-Funktion mithilfe der Konsole im Entwicklerhandbuch.AWS Lambda
16. Wenn Sie bereit sind, die Funktion bei CloudFront Ereignissen ausführen zu lassen, veröffentlichen Sie eine weitere Version und bearbeiten Sie die Funktion, um Auslöser hinzuzufügen. Weitere Informationen finden Sie unter Trigger für eine Lambda @Edge -Funktion hinzufügen.

Eine Lambda-Funktion bearbeiten

Nachdem Sie eine Lambda @Edge -Funktion erstellt haben, können Sie sie mit der Lambda-Konsole bearbeiten.

Hinweise

- Die Originalversion ist mit $LATEST gekennzeichnet.
- Sie können nur die $LATEST-Version bearbeiten.
- Jedes Mal, wenn Sie die $LATEST-Version bearbeiten, müssen Sie eine neue nummerierte Version veröffentlichen.
- Sie können keine Auslöser für $LATEST erstellen.
- Wenn Sie eine neue Version einer Funktion veröffentlichen, kopiert Lambda nicht automatisch Auslöser von der vorherigen Version zur neuen Version. Sie müssen die Auslöser für die neue Version reproduzieren.
- Wenn Sie einer Funktion einen Trigger für ein CloudFront Ereignis hinzufügen und es bereits einen Trigger für dieselbe Verteilung, dasselbe Cache-Verhalten und dasselbe Ereignis für eine frühere Version derselben Funktion gibt, löscht Lambda den Trigger aus der früheren Version.
- Nachdem Sie Aktualisierungen an einer CloudFront Verteilung vorgenommen haben, z. B. Trigger hinzugefügt haben, müssen Sie warten, bis die Änderungen an den Edge-Standorten wirksam werden, bevor die Funktionen, die Sie in den Triggern angegeben haben, funktionieren.

So bearbeiten Sie eine Lambda-Funktion

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Lambda Konsole unter https://console.aws.amazon.com/lambda/.
2. Wählen Sie oben auf der Seite in der Liste „Region“ die Option US Ost (Nord-Virginia) aus.
3. Wählen Sie in der Liste der Funktionen den Namen der Funktion aus.

   Die Konsole zeigt standardmäßig die $LATEST-Version an. Sie können frühere Versionen anzeigen (wählen Sie Qualifiers), aber Sie können nur $LATEST bearbeiten.
4. Wählen Sie auf der Registerkarte Code für Code entry type (Code-Eingabetyp) die Bearbeitung des Codes im Browser, laden Sie eine .zip-Datei hoch oder laden Sie eine Datei aus Amazon S3 hoch.
5. Wählen Sie entweder Save oder Save and test.
6. Wählen Sie Actions und Publish new version.

7. Geben Sie im Dialogfeld Publish new version from $LATEST eine Beschreibung der neuen Version ein. Diese Beschreibung wird in der Liste der Versionen zusammen mit einer automatisch generierten Versionsnummer angezeigt.
8. Wählen Sie Publish.

   Die neue Version wird automatisch die aktuelle Version. Die Versionsnummer wird in der Version in der oberen linken Ecke der Seite angezeigt.

   > **Note**
   >
   > Wenn Sie noch keine Auslöser für Ihre Funktion hinzugefügt haben, finden Sie weitere Informationen unter. Trigger für eine Lambda @Edge -Funktion hinzufügen

9. Wählen Sie die Registerkarte Triggers.
10. Wählen Sie Add trigger.
11. Wählen Sie im Dialogfeld Add trigger das Feld mit Punkten und dann CloudFront.

    > **Note**
    >
    > Wenn Sie bereits einen oder mehrere Auslöser für eine Funktion erstellt haben, CloudFront ist dies der Standarddienst.

12. Geben Sie die folgenden Werte an, um anzugeben, wann die Lambda-Funktion ausgeführt werden soll.

    a. Verteilungs-ID — Wählen Sie die ID der Distribution aus, zu der Sie den Trigger hinzufügen möchten.

    b. Cache-Verhalten — Wählen Sie das Cache-Verhalten, das die Objekte angibt, für die Sie die Funktion ausführen möchten.

    c. CloudFront Ereignis — Wählen Sie das CloudFront Ereignis aus, das die Ausführung der Funktion veranlasst.

    d. Triggern und replizieren aktivieren — Aktivieren Sie dieses Kontrollkästchen, damit Lambda die Funktion global repliziert. AWS-Regionen

13. Wählen Sie Absenden aus.
14. Um weitere Auslöser für diese Funktion hinzuzufügen, wiederholen Sie die Schritte 10 bis 13.

Weitere Informationen zum Testen und Debuggen der Funktion in der Lambda-Konsole finden Sie unter Aufrufen einer Lambda-Funktion mithilfe der Konsole im Entwicklerhandbuch.AWS Lambda

Wenn Sie bereit sind, die Funktion bei CloudFront Ereignissen ausführen zu lassen, veröffentlichen Sie eine andere Version und bearbeiten Sie die Funktion, um Auslöser hinzuzufügen. Weitere Informationen finden Sie unter Trigger für eine Lambda @Edge -Funktion hinzufügen.

Trigger für eine Lambda @Edge -Funktion hinzufügen

Ein Lambda @Edge -Trigger ist eine Kombination aus einer CloudFront Verteilung, einem Cache-Verhalten und einem Ereignis, das die Ausführung einer Funktion bewirkt. Sie können beispielsweise einen Trigger erstellen, der bewirkt, dass die Funktion ausgeführt wird, wenn von einem Viewer eine Anfrage für ein bestimmtes Cache-Verhalten CloudFront empfangen wird, das Sie für Ihre Distribution eingerichtet haben. Sie können einen oder mehrere CloudFront Trigger angeben.

> **Tip**
>
> Wenn Sie eine CloudFront Verteilung erstellen, geben Sie Einstellungen an, die festlegen, CloudFront wie auf unterschiedliche Anfragen reagiert werden soll. Die Standardeinstellungen werden als Standard-Cache-Verhalten für die Verteilung bezeichnet. Sie können zusätzliche Cache-Verhaltensweisen einrichten, die definieren, wie unter bestimmten Umständen CloudFront reagiert wird, z. B. wenn eine Anfrage für einen bestimmten Dateityp eingeht. Weitere Informationen finden Sie unter Einstellungen für das Cache-Verhalten.

Wenn Sie zum ersten Mal eine Lambda-Funktion erstellen, können Sie nur einen Trigger angeben. Sie können derselben Funktion später weitere Trigger hinzufügen, indem Sie die Lambda-Konsole verwenden oder die Verteilung in der CloudFront Konsole bearbeiten.

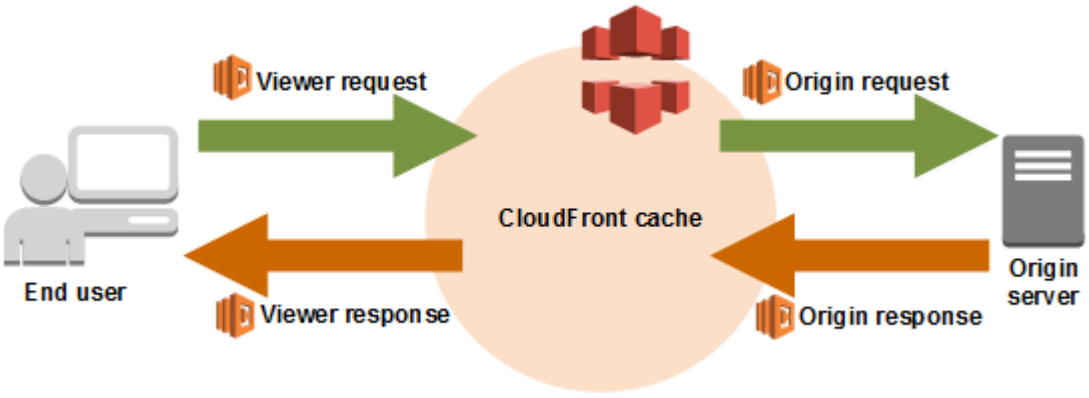
- Die Lambda-Konsole funktioniert gut, wenn Sie einer Funktion für dieselbe CloudFront Distribution weitere Trigger hinzufügen möchten.
- Die CloudFront Konsole kann besser sein, wenn Sie Trigger für mehrere Distributionen hinzufügen möchten, da es einfacher ist, die Distribution zu finden, die Sie aktualisieren möchten. Sie können auch andere CloudFront Einstellungen gleichzeitig aktualisieren.

Themen

- Wählen Sie das Ereignis aus, das die Funktion auslösen soll
- Trigger zu einer Lambda @Edge -Funktion hinzufügen

CloudFront Ereignisse, die eine Lambda @Edge -Funktion auslösen können

Für jedes Cache-Verhalten in einer CloudFront Amazon-Distribution können Sie bis zu vier Trigger (Assoziationen) hinzufügen, die bewirken, dass eine Lambda-Funktion ausgeführt wird, wenn bestimmte CloudFront Ereignisse eintreten. CloudFront Trigger können auf einem von vier CloudFront Ereignissen basieren, wie in der folgenden Abbildung dargestellt.



Die folgenden CloudFront Ereignisse können verwendet werden, um Lambda @Edge -Funktionen auszulösen:

Viewer-Anforderung

Die Funktion wird ausgeführt, wenn sie eine Anfrage von einem Viewer CloudFront erhält, bevor sie überprüft, ob sich das angeforderte Objekt im CloudFront Cache befindet.

Die Funktion wird in den folgenden Fällen nicht ausgeführt:

- Beim Abrufen einer benutzerdefinierten Fehlerseite.
- Wenn eine HTTP-Anfrage CloudFront automatisch an HTTPS umgeleitet wird (wenn der Wert von „HTTP zu HTTPS umleiten“ Viewer-Protokollrichtlinien lautet).

Ursprungsanfrage

Die Funktion wird nur ausgeführt, wenn eine Anfrage an Ihren Ursprung CloudFront weitergeleitet wird. Wenn sich das angeforderte Objekt im CloudFront Cache befindet, wird die Funktion nicht ausgeführt.

Ursprungsantwort

Die Funktion wird ausgeführt, nachdem CloudFront sie eine Antwort vom Ursprung erhalten hat und bevor das Objekt in der Antwort zwischengespeichert wird. Beachten Sie, dass die Funktion auch dann ausgeführt wird, wenn ein Fehler vom Ursprung zurückgegeben wird.

Die Funktion wird in den folgenden Fällen nicht ausgeführt:

- Wenn sich die angeforderte Datei im CloudFront Cache befindet und nicht abgelaufen ist.
- Wenn die Antwort von einer Funktion generiert wird, die von einem Ursprungs-Anforderungsereignis ausgelöst wurde.

Viewer-Antwort

Diese Funktion wird ausgeführt, bevor die angeforderte Datei an den Viewer zurückgegeben wird. Beachten Sie, dass die Funktion unabhängig davon ausgeführt wird, ob sich die Datei bereits im CloudFront Cache befindet.

Die Funktion wird in den folgenden Fällen nicht ausgeführt:

- Wenn der Ursprung den HTTP-Statuscode „400“ oder höher zurückgibt.
- Wenn eine benutzerdefinierte Fehlerseite zurückgesendet wird.
- Wenn die Antwort von einer Funktion generiert wird, die von einem Viewer-Anforderungsereignis ausgelöst wurde.
- Wenn eine HTTP-Anfrage CloudFront automatisch an HTTPS umgeleitet wird (wenn der Wert von „HTTP zu HTTPS umleiten“ Viewer-Protokollrichtlinien lautet).

Wenn Sie einem Cache-Verhalten mehrere Auslöser hinzufügen, können diese jeweils dieselbe oder verschiedene Funktionen für jeden Auslöser ausführen. Sie können Funktionen auch in mehreren Verteilungen zuweisen.

> **Note**
>
> Wenn ein CloudFront Ereignis die Ausführung einer Lambda-Funktion auslöst, muss die Funktion beendet werden, bevor sie fortgesetzt CloudFront werden kann.
> Wenn beispielsweise eine Lambda-Funktion durch ein CloudFront Viewer-Anforderungsereignis ausgelöst CloudFront wird, wird keine Antwort an den Betrachter zurückgegeben oder die Anfrage an den Ursprung weitergeleitet, bis die Lambda-Funktion vollständig ausgeführt wurde.

Das bedeutet, dass jede Anfrage, die eine Lambda-Funktion auslöst, die Latenz für die Anfrage erhöht, sodass Sie möchten, dass die Funktion so schnell wie möglich ausgeführt wird.

Wählen Sie das Ereignis aus, das die Funktion auslösen soll

Wenn Sie entscheiden, welches CloudFront Ereignis Sie verwenden möchten, um eine Lambda-Funktion auszulösen, sollten Sie Folgendes berücksichtigen:

Ich CloudFront möchte Objekte zwischenspeichern, die durch eine Lambda-Funktion geändert wurden

Um ein Objekt zwischenzuspeichern, das durch eine Lambda-Funktion geändert wurde, CloudFront sodass es bei der nächsten Anforderung vom Edge-Standort aus bedient werden kann, verwenden Sie die Origin-Anfrage oder das Origin-Response-Ereignis.

Dadurch verringert sich die Verarbeitungslast für den Ursprung, die Latenz für nachfolgende Anforderungen wird verringert und die Kosten für den Aufruf von Lambda@Edge für nachfolgende Anforderungen reduziert.

Wenn Sie beispielsweise Header für Objekte hinzufügen, entfernen oder ändern möchten, die vom Ursprung zurückgegeben werden, und Sie das Ergebnis zwischenspeichern CloudFront möchten, verwenden Sie das Origin-Response-Ereignis.

Ich möchte, dass die Funktion für jede Anfrage ausgeführt wird

Um die Funktion für jede Anfrage auszuführen, die für die Verteilung CloudFront eingeht, verwenden Sie die Viewer-Anforderung - oder Viewer-Antwort-Ereignisse.

Origin-Request- und Origin-Response-Ereignisse treten nur auf, wenn ein angefordertes Objekt nicht an einem Edge-Standort zwischengespeichert ist und eine Anfrage an den Ursprung CloudFront weiterleitet.

Ich möchte, dass die Funktion den Cache-Schlüssel ändert

Um einen Wert zu ändern, den Sie als Grundlage für das Caching verwenden, verwenden Sie das Viewer-Anforderungsereignis.

Wenn beispielsweise eine Funktion eine URL so ändert, dass Abkürzungen für die Sprachversionen in den Pfad eingebunden werden (zum Beispiel, weil der Benutzer seine Sprache aus einer Dropdown-Liste ausgewählt hat), verwenden Sie das Viewer-Anfrageereignis:

- URL in der Viewer-Anfrage — index.html https://example.com/en/
- URL, wenn die Anfrage von einer IP-Adresse in Deutschland kommt — https://example.com/de/index.html

Sie können Viewer-Anfrageereignisse auch verwenden, wenn die Zwischenspeicherung auf der Basis von Cookies oder Anfrageereignissen erfolgt.

> **Note**
>
> Wenn die Funktion Cookies oder Header ändert, konfigurieren Sie CloudFront sie so, dass der entsprechende Teil der Anfrage an den Ursprung weitergeleitet wird. Weitere Informationen finden Sie unter den folgenden Themen:
>
> - Auf Cookies basierender Inhalt zwischenspeichern
> - Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern

Die Funktion wirkt sich auf die Antwort vom Ursprung aus

Um die Anfrage so zu ändern, dass sich dies auf die Antwort vom Ursprung auswirkt, verwenden Sie das Ereignis der ursprünglichen Anfrage.

In der Regel werden die meisten Ereignisse mit Zuschaueranfragen nicht an den Absender weitergeleitet. CloudFront reagiert auf eine Anfrage mit einem Objekt, das sich bereits im Edge-Cache befindet. Wenn die Funktion die Anfrage auf der Grundlage eines ursprünglichen Anforderungsereignisses ändert, CloudFront speichert sie die Antwort auf die geänderte ursprüngliche Anfrage im Cache.

Trigger zu einer Lambda @Edge -Funktion hinzufügen

Sie können die AWS Lambda Konsole oder die CloudFront Amazon-Konsole verwenden, um Ihrer Lambda @Edge -Funktion einen Trigger hinzuzufügen.

> **Important**
>
> Sie können Trigger nur für nummerierte Versionen Ihrer Funktion erstellen (nicht für $LATEST).

Lambda console

Um Trigger für CloudFront Ereignisse zu einer Lambda @Edge -Funktion hinzuzufügen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Lambda Konsole unter https://console.aws.amazon.com/lambda/.
2. Wählen Sie oben auf der Seite in der Liste „Region“ die Option US Ost (Nord-Virginia) aus.
3. Wählen Sie auf der Seite Functions den Namen der Funktion, für die Sie Auslöser hinzufügen möchten.
4. Wählen Sie auf der Seite mit der Funktionsübersicht den Tab Versionen aus.
5. Wählen Sie die Version, der Sie Auslöser hinzufügen möchten.

   Nach der Wahl einer Version ändert sich der Name der Schaltfläche in Version: $LATEST oder Version: Versionsnummer.
6. Wählen Sie die Registerkarte Triggers.
7. Wählen Sie Add trigger.
8. Wählen Sie für die Trigger-Konfiguration die Option Quelle auswählen**`cloudfront`**, geben Sie die Eingabetaste ein und wählen Sie dann CloudFront.

   > **Note**
   >
   > Wenn Sie bereits einen oder mehrere Auslöser erstellt haben, CloudFront ist dies der Standarddienst.
9. Geben Sie die folgenden Werte an, um anzugeben, wann die Lambda-Funktion ausgeführt werden soll.

   a. Verteilung — Wählen Sie die Distribution aus, zu der Sie den Trigger hinzufügen möchten.

   b. Cache-Verhalten — Wählen Sie das Cache-Verhalten, das die Objekte angibt, für die Sie die Funktion ausführen möchten.

   > **Note**
   >
   > Wenn Sie * für das Zwischenspeicher-Verhalten angeben, stellt die Lambda-Funktion das Standard-Zwischenspeicher-Verhalten bereit.

c. CloudFront Ereignis — Wählen Sie das CloudFront Ereignis aus, das die Ausführung der Funktion veranlasst.

d. Hauptteil einbeziehen — Aktivieren Sie dieses Kontrollkästchen, wenn Sie in Ihrer Funktion auf den Anforderungstext zugreifen möchten.

e. Bestätigen Sie die Bereitstellung auf Lambda @Edge — Aktivieren Sie dieses Kontrollkästchen, damit die Funktion AWS Lambda global repliziert wird. AWS-Regionen

10. Wählen Sie Hinzufügen aus.

    Die Funktion beginnt mit der Verarbeitung von Anfragen für die angegebenen CloudFront Ereignisse, wenn die aktualisierte CloudFront Distribution bereitgestellt wird. Um zu ermitteln, ob eine Verteilung bereitgestellt ist, wählen Sie im Navigationsbereich die Option Distributions. Wenn eine Verteilung bereitgestellt wird, ändert sich der Wert der Spalte Status für die Verteilung von Bereitstellen auf Datum und Uhrzeit der Bereitstellung.

CloudFront console

Um Trigger für CloudFront Ereignisse zu einer Lambda @Edge -Funktion hinzuzufügen

1. Rufen Sie den ARN der Lambda-Funktion ab, der Sie Auslöser hinzufügen möchten:

   a. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Lambda Konsole unter https://console.aws.amazon.com/lambda/.

   b. Wählen Sie in der Liste der Regionen oben auf der Seite die Option USA Ost (Nord-Virginia).

   c. Wählen Sie in der Liste der Funktionen den Namen der Funktion, der Sie Auslöser hinzufügen möchten.

   d. Wählen Sie auf der Seite mit der Funktionsübersicht die Registerkarte Versionen und dann die nummerierte Version aus, zu der Sie Trigger hinzufügen möchten.

   e. Wählen Sie die Schaltfläche ARN kopieren, um den ARN in Ihre Zwischenablage zu kopieren. Der ARN für die Lambda-Funktion sieht ungefähr so aus:

      ```
      arn:aws:lambda:us-east-1:123456789012:function:TestFunction:2
      ```

      Die Nummer am Ende (2 in diesem Beispiel) ist die Versionsnummer der Funktion.

2. Öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home

3. Wählen Sie in der Liste der Verteilungen die ID der Verteilung aus, der Sie Auslöser hinzufügen möchten.
4. Wählen Sie die Registerkarte Behaviors aus.
5. Wählen Sie das Cache-Verhalten aus, dem Sie Trigger hinzufügen möchten, und klicken Sie dann auf Bearbeiten.
6. Wählen Sie für Funktionszuordnungen in der Liste Funktionstyp die Option Lambda @Edge aus, wenn die Funktion ausgeführt werden soll: für Viewer-Anfragen, Viewer-Antworten, Ursprungsanfragen oder Ursprungsantworten.

   Weitere Informationen finden Sie unter Wählen Sie das Ereignis aus, das die Funktion auslösen soll.
7. Fügen Sie in das Textfeld Function ARN//Name den ARN der Lambda-Funktion ein, die Sie ausführen möchten, wenn das gewählte Ereignis eintritt. Dies ist der Wert, den Sie aus der Lambda-Konsole kopiert haben.
8. Wählen Sie Text einschließen aus, wenn Sie in Ihrer Funktion auf den Anforderungstext zugreifen möchten.

   Wenn Sie nur den Anforderungstext ersetzen möchten, müssen Sie diese Option nicht auswählen.
9. Um dieselbe Funktion für mehrere Ereignistypen auszuführen, wiederholen Sie die Schritte 6 und 7.
10. Wählen Sie Änderungen speichern aus.
11. Um Trigger zu weiteren Cache-Verhaltensweisen für diese Verteilung hinzuzufügen, wiederholen Sie die Schritte 5 bis 10.

    Die Funktion beginnt mit der Verarbeitung von Anfragen für die angegebenen CloudFront Ereignisse, wenn die aktualisierte CloudFront Distribution bereitgestellt wird. Um zu ermitteln, ob eine Verteilung bereitgestellt ist, wählen Sie im Navigationsbereich die Option Distributions. Wenn eine Verteilung bereitgestellt wird, ändert sich der Wert der Statusspalte für die Verteilung von Bereitstellen auf Uhrzeit und Datum der Bereitstellung.

Testen und Debuggen von Lambda @Edge -Funktionen

Es ist wichtig, dass Sie Ihren Lambda @Edge -Funktionscode eigenständig testen, um sicherzustellen, dass er die beabsichtigte Aufgabe erfüllt, und Integrationstests durchzuführen, um sicherzustellen, dass die Funktion korrekt funktioniert. CloudFront

Während des Integrationstests oder nach der Bereitstellung Ihrer Funktion müssen Sie möglicherweise Fehler wie HTTP CloudFront 5xx-Fehler debuggen. Fehler können eine ungültige Antwort der Lambda-Funktion, Ausführungsfehler beim Auslösen der Funktion oder Fehler aufgrund einer Ablehnung der Ausführung durch den Lambda-Service sein. Die Abschnitte in diesem Thema enthalten Strategien, um festzustellen, welche Art von Fehler das Problem verursacht. Dazu finden Sie Schritte, die Sie unternehmen können, um das Problem zu beheben.

> **Note**
>
> Achten Sie bei der Überprüfung von CloudWatch Protokolldateien oder Metriken bei der Behebung von Fehlern darauf, dass diese an dem Ort angezeigt oder gespeichert werden, der dem Ort, an dem die Funktion ausgeführt wurde, AWS-Region am nächsten ist. Wenn Sie also eine Website oder Webanwendung mit Benutzern im Vereinigtes Königreich haben und Ihrer Distribution beispielsweise eine Lambda-Funktion zugeordnet ist, müssen Sie die Region ändern, um die CloudWatch Metriken oder Protokolldateien für London AWS-Region anzuzeigen. Weitere Informationen finden Sie unter the section called “ Ermitteln Sie die Lambda @Edge -Region”.

Themen

Testen Sie Ihre Lambda @Edge -Funktionen

Der Test Ihrer Lambda-Funktion besteht aus zwei Schritten: eigenständiger Test und Integrationstest.

Eigenständiger Test der Funktionalität

Bevor Sie Ihre Lambda-Funktion hinzufügen CloudFront, stellen Sie sicher, dass Sie die Funktionalität zuerst testen, indem Sie die Testfunktionen in der Lambda-Konsole oder andere Methoden verwenden. Weitere Informationen zum Testen in der Lambda-Konsole finden Sie unter Aufrufen einer Lambda-Funktion mithilfe der Konsole im Entwicklerhandbuch.AWS Lambda

Testen Sie den Betrieb Ihrer Funktion in CloudFront

Es ist wichtig, Integrationstests abzuschließen, bei denen Ihre Funktion einer Distribution zugeordnet ist und auf der Grundlage eines CloudFront Ereignisses ausgeführt wird. Stellen Sie sicher, dass die Funktion für das richtige Ereignis ausgelöst wird und eine für CloudFront gültige und korrekte Antwort zurückgibt. Stellen Sie beispielsweise sicher, dass die Ereignisstruktur korrekt ist, dass nur gültige Header enthalten sind usw.

Wenn Sie die Integrationstests mit Ihrer Funktion in der Lambda-Konsole wiederholen, folgen Sie den Schritten im Lambda @Edge -Tutorial, wenn Sie Ihren Code oder den CloudFront Trigger ändern, der Ihre Funktion aufruft. Stellen Sie beispielsweise sicher, dass Sie mit einer nummerierten Version Ihrer Funktion arbeiten, wie in diesem Schritt des Tutorials beschrieben: Schritt 4: Fügen Sie einen CloudFront Trigger hinzu, um die Funktion auszuführen.

Beachten Sie beim Vornehmen von Änderungen und deren Bereitstellung, dass es mehrere Minuten dauern kann, bis Ihre aktualisierte Funktion und CloudFront Trigger in allen Regionen repliziert sind. Dies dauert in der Regel wenige Minuten, kann jedoch auch bis zu 15 Minuten dauern.

Sie können überprüfen, ob die Replikation abgeschlossen ist, indem Sie zur CloudFront Konsole gehen und sich Ihre Distribution ansehen.

Um zu überprüfen, ob die Bereitstellung Ihrer Replikation abgeschlossen ist

1. Öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie den Distributionsnamen.
3. Überprüfen Sie, ob sich der Status der Verteilung von InProgress (Läuft) zurück auf Deployed (Bereitgestellt) geändert hat, d.h. Ihre Funktion wurde repliziert. Anschließend befolgen Sie die Schritte im nächsten Abschnitt, um zu überprüfen, ob die Funktion funktioniert.

Beachten Sie, dass das Testen in der Konsole nur die Logik Ihrer Funktion validiert und keine Servicekontingente (früher als Limits bezeichnet) anwendet, die spezifisch für Lambda@Edge sind.

Identifizieren Sie Lambda @Edge -Funktionsfehler in CloudFront

Nachdem Sie überprüft haben, ob Ihre Funktionslogik korrekt funktioniert, werden möglicherweise immer noch HTTP 5xx-Fehler angezeigt, wenn Ihre Funktion ausgeführt wird. CloudFront HTTP 5xx-Fehler können aus einer Vielzahl von Gründen zurückgegeben werden, darunter Lambda-Funktionsfehler oder andere Probleme in. CloudFront

- Wenn Sie Lambda @Edge -Funktionen verwenden, können Sie mithilfe von Diagrammen in der CloudFront Konsole herausfinden, was den Fehler verursacht, und ihn dann beheben. Sie können beispielsweise sehen, ob HTTP 5xx-Fehler durch CloudFront oder durch Lambda-Funktionen verursacht werden, und dann können Sie für bestimmte Funktionen zugehörige Protokolldateien einsehen, um das Problem zu untersuchen.
- Informationen zur allgemeinen Behebung von HTTP-Fehlern finden Sie in CloudFront den Schritten zur Fehlerbehebung im folgenden Thema:. Behebung von Statuscodes für die Fehlerantwort in CloudFront

Was verursacht Lambda @Edge -Funktionsfehler in CloudFront

Es gibt mehrere Gründe, aus denen eine Lambda-Funktion einen HTTP 5xx-Fehler verursachen kann. Die Schritte zur Fehlerbehebung hängen von der Art des Fehlers ab. Fehler können wie folgt kategorisiert werden:

Ein Lambda-Funktionsausführungsfehler.

> Ein Ausführungsfehler tritt auf, wenn Sie CloudFront keine Antwort von Lambda erhalten, weil die Funktion unbehandelte Ausnahmen enthält oder ein Fehler im Code vorliegt. Zum Beispiel, wenn der Code "callback(Error)" beinhaltet.

Eine ungültige Lambda-Funktionsantwort wird zurückgegeben an CloudFront

> CloudFront Erhält nach der Ausführung der Funktion eine Antwort von Lambda. Ein Fehler wird zurückgegeben, wenn die Objektstruktur der Antwort nicht der Lambda@Edge-Ereignisstruktur entspricht oder die Antwort ungültige Header oder andere ungültige Felder enthält.

Die Ausführung in CloudFront wird aufgrund von Lambda-Servicequotas (früher als Limits bezeichnet) gedrosselt

> Der Lambda-Service drosselt die Ausführung in jeder Region und gibt einen Fehler zurück, wenn Sie das Kontingent überschreiten. Weitere Informationen finden Sie unter Kontingente für Lambda@Edge.

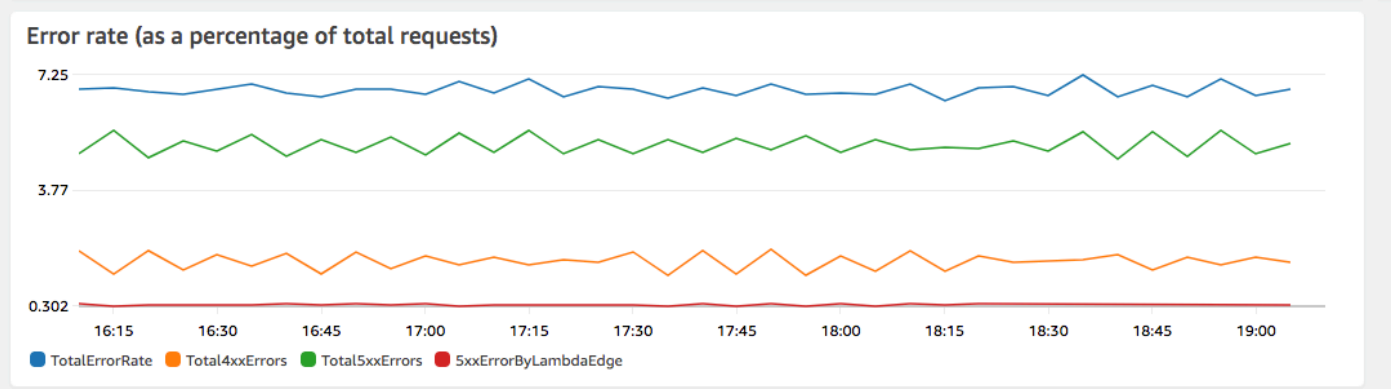
So bestimmten Sie den Typ des Fehlers

Um Ihnen bei der Entscheidung zu helfen, worauf Sie sich beim Debuggen und bei der Behebung von Fehlern konzentrieren sollten CloudFront, ist es hilfreich zu ermitteln, warum CloudFront ein HTTP-Fehler zurückgegeben wird. Zu Beginn können Sie die Grafiken verwenden, die im Abschnitt Überwachung der CloudFront Konsole auf dem AWS Management Console bereitgestellt werden. Weitere Informationen zum Anzeigen von Diagrammen im Bereich Überwachung der CloudFront Konsole finden Sie unterÜberwachen Sie CloudFront Metriken mit Amazon CloudWatch.

Die folgenden Diagramme sind besonders hilfreich, wenn Sie nachverfolgen möchten, ob Fehler von Ursprungs-Servern oder einer Lambda-Funktion zurückgegeben werden, und wenn Sie die Art des Problems eingrenzen möchten, wenn der Fehler aus einer Lambda-Funktion resultiert.

Fehlerratendiagramm

Eines der Diagramme, das Sie auf der Registerkarte Overview (Überblick) für Ihre Verteilungen anzeigen können, ist das Diagramm Error rates (Fehlerraten). Dieses Diagramm zeigt die Rate der Fehler als Prozentsatz aller Anforderungen an, die bei Ihren Verteilungen eingehen. Das Diagramm zeigt die Gesamtfehlerrate, die gesamten 4xx-Fehler, die gesamten 5xx-Fehler und die gesamten 5xx-Fehler an, die aus Lambda-Funktionen resultieren. Basierend auf Fehlertyp und Anzahl können Sie Schritte unternehmen, um diese Probleme zu untersuchen und zu beheben.



- Wenn Lambda-Fehler auftreten, können Sie eine genauere Untersuchung durchführen, indem Sie sich die spezifischen Arten von Fehlern ansehen, die von der Funktion zurückgegeben werden. Die Registerkarte Lambda@Edge errors (Lambda@Edge-Fehler) enthält Diagramme, die Funktionsfehler nach Typ kategorisieren, damit Sie das Problem für eine bestimmte Funktion herausfinden können.
- Wenn Sie CloudFront Fehler sehen, können Sie Fehler beheben und daran arbeiten, die ursprünglichen Fehler zu beheben oder Ihre CloudFront Konfiguration zu ändern. Weitere Informationen finden Sie unter Behebung von Statuscodes für die Fehlerantwort in CloudFront.

Diagramme für Ausführungsfehler und ungültige Funktionsanworten

Die Registerkarte Lambda@Edge errors (Lambda@Edge-Fehler) enthält Diagramme, die Lambda@Edge-Fehler für eine bestimmte Verteilung nach Typ kategorisieren. In einem Diagramm werden beispielsweise alle Ausführungsfehler von angezeigt AWS-Region.

Um die Behebung von Problemen zu vereinfachen, können Sie nach bestimmten Problemen suchen, indem Sie die Protokolldateien für bestimmte Funktionen nach Regionen öffnen und untersuchen.

Um Protokolldateien für eine bestimmte Funktion nach Region aufgeschlüsselt anzuzeigen

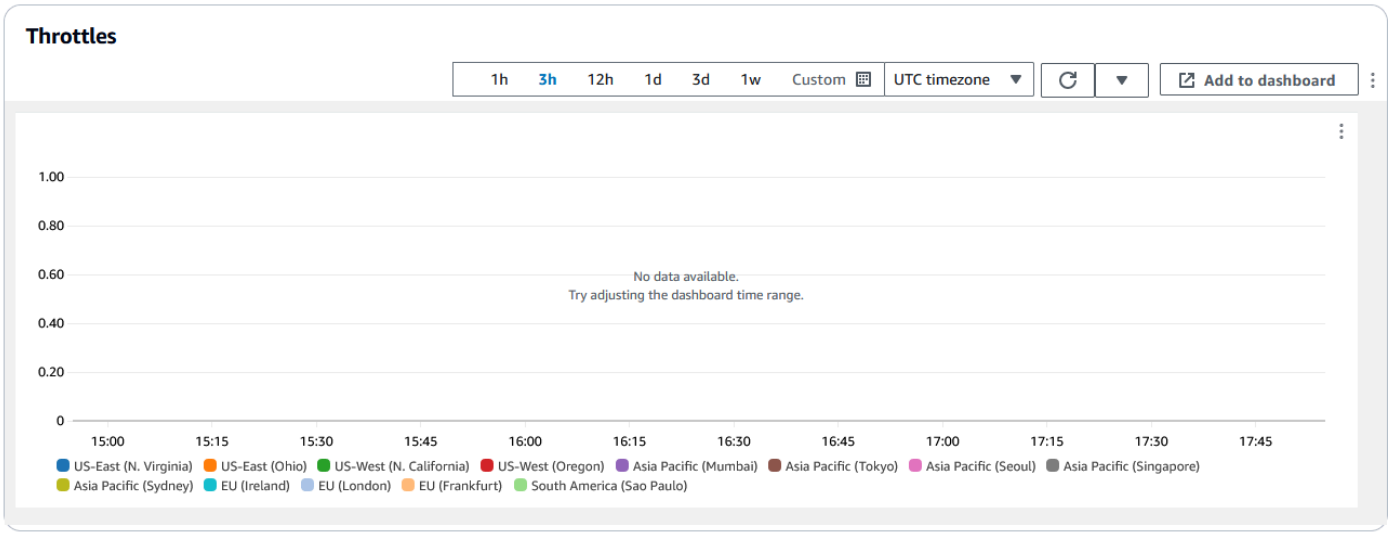
1. Wählen Sie auf der Registerkarte Lambda @Edge -Fehler unter Zugeordnete Lambda @Edge -Funktionen den Funktionsnamen und dann Metriken anzeigen aus.
2. Wählen Sie als Nächstes auf der Seite mit Ihrem Funktionsnamen in der oberen rechten Ecke die Option Funktionsprotokolle anzeigen und dann eine Region aus.

   Wenn Sie beispielsweise Probleme im Fehlerdiagramm für die Region USA West (Oregon) sehen, wählen Sie diese Region aus der Dropdownliste aus. Dadurch wird die CloudWatch Amazon-Konsole geöffnet.
3. Wählen Sie in der CloudWatch Konsole für diese Region unter Protokollstreams einen Protokollstream aus, um die Ereignisse für die Funktion anzuzeigen.

Lesen Sie darüber hinaus die folgenden Abschnitte in diesem Kapitel, um weitere Empfehlungen für die Behebung von Fehlern zu erhalten.

Drosselungsdiagramm

Die Registerkarte Lambda@Edge errors (Lambda@Edge-Fehler) enthält auch ein Diagramm zu Drosselungen. In einigen Situationen drosselt der Lambda-Service Ihre Funktionsaufrufe auf Regionsbasis, wenn Sie sich dem regionalen Kontingent (früher als Limit bezeichnet) für die Gleichzeitigkeit nähern. Wenn Sie einen limit exceeded (Grenzwert überschritten)-Fehler sehen, hat Ihre Funktion ein Kontingent erreicht, das der Lambda-Service für Ausführungen in einer Region nutzt. Weitere Informationen, u. a. auch zur Erhöhung des Kontingents, finden Sie unter Kontingente für Lambda@Edge.



Ein Beispiel zur Verwendung dieser Informationen beim Beheben von HTTP-Fehlern finden Sie unter Vier Schritte zum Debuggen der Bereitstelllung von Inhalten in AWS.

Fehlerbehebung bei ungültigen Lambda @Edge -Funktionsantworten (Validierungsfehler)

Wenn Sie feststellen, dass es sich bei Ihrem Problem um einen Lambda-Validierungsfehler handelt, bedeutet dies, dass Ihre Lambda-Funktion eine ungültige Antwort auf zurückgibt. CloudFront Folgen Sie den Anweisungen in diesem Abschnitt, um Maßnahmen zur Überprüfung Ihrer Funktion zu ergreifen und sicherzustellen, dass Ihre Antwort den Anforderungen entspricht. CloudFront

CloudFront validiert die Antwort einer Lambda-Funktion auf zwei Arten:

- Die Lambda-Antwort muss der gewünschten Objektstruktur entsprechen. Beispiele für eine fehlerhafte Objektstruktur sind: nicht interpretierbarer JSON-Code, fehlende Pflichtfelder und ein ungültiges Objekt in der Antwort. Weitere Informationen hierzu finden Sie unter Lambda@Edge-Ereignisstruktur.
- Die Antwort darf nur gültige Objektwerte beinhalten. Ein Fehler tritt auf, wenn die Antwort ein gültiges Objekt aber nicht unterstützte Werte enthält. Beispiele sind das Hinzufügen oder Aktualisieren von ungültigen Headern oder schreibgeschützten Headern (siehe Einschränkungen für Edge-Funktionen), das Überschreiten der maximalen Body-Größe (siehe Beschränkungen für die Größe der generierten Antwort im Thema Lambda@Edge Fehler) und ungültige Zeichen oder Werte (siehe Lambda@Edge-Ereignisstruktur).

Wenn Lambda eine ungültige Antwort auf zurückgibt CloudFront, werden Fehlermeldungen in Protokolldateien geschrieben, die in die Region CloudFront übertragen werden, CloudWatch in der die Lambda-Funktion ausgeführt wurde. Dies ist das Standardverhalten, an das die Protokolldateien gesendet werden, CloudWatch wenn eine ungültige Antwort eingeht. Wenn Sie jedoch eine Lambda-Funktion mit verknüpft haben, CloudFront bevor die Funktionalität veröffentlicht wurde, ist sie möglicherweise nicht für Ihre Funktion aktiviert. Weitere Informationen finden Sie unter Feststellen, ob Ihr Konto Protokolle an CloudWatch überträgt weiter unten im Thema.

CloudFront überträgt Protokolldateien in die Region, die der Region entspricht, in der Ihre Funktion ausgeführt wurde, in der Protokollgruppe, die Ihrer Distribution zugeordnet ist. Protokollgruppen haben das folgende Format:`/aws/cloudfront/LambdaEdge/`*`DistributionId`*, wo *`DistributionId`* ist die ID Ihrer Distribution. Informationen zur Bestimmung der Region, in der Sie die CloudWatch Protokolldateien finden, finden Sie unter Bestimmung der Lambda @Edge -Region weiter unten in diesem Thema.

Wenn der Fehler reproduzierbar ist, können Sie eine neue Anfrage erstellen, die zu dem Fehler führt, und dann die Anforderungs-ID in einer fehlgeschlagenen CloudFront Antwort (`X-Amz-Cf-Id`Header) suchen, um einen einzelnen Fehler in den Protokolldateien zu lokalisieren. Der Eintrag in der Protokolldatei enthält Informationen, die Ihnen bei der Identifizierung Ursache für den Fehler helfen können. Er zeigt außerdem die entsprechende Lambda-Anforderungs-ID an, mit der Sie die Ursache im Kontext einer einzelnen Anforderung analysieren können.

Wenn ein Fehler nur sporadisch auftritt, können Sie mithilfe von CloudFront Zugriffsprotokollen die Anforderungs-ID für eine fehlgeschlagene Anfrage ermitteln und anschließend die CloudWatch Protokolle nach den entsprechenden Fehlermeldungen durchsuchen. Weitere Informationen finden Sie im vorherigen Abschnitt Bestimmung der Fehlerart.

Behebung von Fehlern bei der Ausführung der Lambda @Edge -Funktion

Wenn es sich bei dem Problem um einen Lambda-Ausführungsfehler handelt, kann es hilfreich sein, Protokollierungsanweisungen für Lambda-Funktionen zu erstellen, Meldungen in CloudWatch Protokolldateien zu schreiben, die die Ausführung Ihrer Funktion überwachen CloudFront und feststellen, ob sie wie erwartet funktioniert. Anschließend können Sie in den CloudWatch Protokolldateien nach diesen Anweisungen suchen, um zu überprüfen, ob Ihre Funktion funktioniert.

> **Note**
>
> Auch wenn Sie Ihre Lambda@Edge-Funktion nicht geändert haben, kann diese durch Aktualisierungen der Lambda-Funktionsausführungsumgebung beeinträchtigt werden und

einen Ausführungsfehler ausgeben. Informationen zum Testen und Migrieren auf eine neuere Version finden Sie unter Kommende Updates für die AWS Lambda- und AWS Lambda @Edge -Ausführungsumgebung.

Ermitteln Sie die Lambda @Edge -Region

Um die Regionen zu sehen, in denen Ihre Lambda @Edge -Funktion Traffic empfängt, sehen Sie sich die Metriken für die Funktion auf der CloudFront Konsole auf der AWS Management Console an. Metriken werden für jede AWS Region angezeigt. Auf derselben Seite können Sie eine Region auswählen und Protokolldateien für diese anzeigen, um Probleme näher zu untersuchen. Sie müssen die CloudWatch Protokolldateien in der richtigen AWS Region überprüfen, um die Protokolldateien zu sehen, die bei der CloudFront Ausführung Ihrer Lambda-Funktion erstellt wurden.

Weitere Informationen zum Anzeigen von Diagrammen im Bereich Überwachung der CloudFront Konsole finden Sie unterÜberwachen Sie CloudFront Metriken mit Amazon CloudWatch.

Stellen Sie fest, ob Ihr Konto Logs per Push an CloudWatch

CloudFront Aktiviert standardmäßig die Protokollierung ungültiger Lambda-Funktionsantworten und überträgt die Protokolldateien CloudWatch mithilfe einer der Serviceverknüpfte Rollen für Lambda@Edge Wenn Sie Lambda @Edge -Funktionen haben, die Sie hinzugefügt haben, CloudFront bevor die Funktion für das Protokoll der ungültigen Lambda-Funktionsantwort veröffentlicht wurde, wird die Protokollierung aktiviert, wenn Sie Ihre Lambda @Edge -Konfiguration das nächste Mal aktualisieren, z. B. indem Sie einen Trigger hinzufügen. CloudFront

Gehen Sie wie folgt vor, um zu überprüfen, ob die Übertragung der Protokolldateien an für Ihr Konto aktiviert CloudWatch ist:

- Prüfen Sie, ob die Protokolle in erscheinen CloudWatch — Achten Sie darauf, dass Sie in der Region suchen, in der die Lambda @Edge -Funktion ausgeführt wurde. Weitere Informationen finden Sie unter Ermitteln Sie die Lambda @Edge -Region.
- Stellen Sie fest, ob die zugehörige serviceverknüpfte Rolle in Ihrem Konto in IAM vorhanden ist — Sie müssen die IAM-Rolle `AWSServiceRoleForCloudFrontLogger` in Ihrem Konto haben. Weitere Informationen über diese Rolle finden Sie unter Serviceverknüpfte Rollen für Lambda@Edge.

Lambda @Edge -Funktionen und Replikate löschen

Sie können eine Lambda@Edge-Funktion nur löschen, wenn die Replikate der Funktion von CloudFront gelöscht wurden. Replikate einer Lambda-Funktion werden in den folgenden Situationen automatisch gelöscht:

- Nachdem Sie die letzte Zuordnung für die Funktion aus allen Ihren CloudFront-Verteilungen entfernt haben. Wenn mehrere Verteilungen eine Funktion verwenden, werden die Replikate erst gelöscht, nachdem Sie die Funktionszuordnung aus der letzten Verteilung entfernt haben.
- Nachdem Sie die letzte Verteilung, der eine Funktion zugeordnet war, gelöscht haben.

Replicas werden in der Regel innerhalb von wenigen Stunden gelöscht. Lambda @Edge-Funktionsreplikate können nicht manuell gelöscht werden. Dadurch wird verhindert, dass ein Replikat gelöscht wird, das noch verwendet wird, was zu einem Fehler führen würde.

> **Warning**
>
> Erstellen Sie keine Anwendungen, die Lambda @Edge -Funktionsreplikate außerhalb von verwenden. CloudFront Diese Replikate werden gelöscht, wenn ihre Zuordnungen zu Verteilungen entfernt werden oder wenn die Verteilungen selbst gelöscht werden. Das Replikat, von dem eine externe Anwendung abhängt, könnte ohne Warnung entfernt werden, was zu einem Fehler führen würde.

Um eine Lambda @Edge -Funktionsassoziation aus einer CloudFront Distribution zu löschen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die ID der Distribution mit der Lambda @Edge -Funktionsassoziation, die Sie löschen möchten.
3. Wählen Sie die Registerkarte Behaviors aus.
4. Wählen Sie das Cache-Verhalten mit der Lambda @Edge -Funktionszuordnung aus, die Sie löschen möchten, und wählen Sie dann Bearbeiten aus.
5. Wählen Sie unter Funktionszuordnungen, Funktionstyp die Option Keine Zuordnung aus, um die Lambda @Edge -Funktionszuordnung zu löschen.
6. Wählen Sie Änderungen speichern aus.

Nachdem Sie eine Lambda @Edge -Funktionszuordnung aus einer CloudFront Distribution gelöscht haben, können Sie optional die Lambda-Funktion oder Funktionsversion aus löschen. AWS Lambda Warten Sie nach dem Löschen der Funktionszuordnung einige Stunden, damit die Lambda @Edge -Funktionsreplikate bereinigt werden können. Danach können Sie die Funktion mithilfe der Lambda-Konsole AWS CLI, der Lambda-API oder eines AWS SDK löschen.

Sie können auch eine bestimmte Version einer Lambda-Funktion löschen, wenn der Version keine CloudFront Distributionen zugeordnet sind. Warten Sie einige Stunden, nachdem Sie alle Verknüpfungen für eine Lambda-Funktionsversion entfernt haben. Dann können Sie die Funktionsversion löschen.

Lambda@Edge-Ereignisstruktur

In den folgenden Themen werden die Anforderungs- und Antwortereignisobjekte beschrieben, die CloudFront an eine Lambda @Edge -Funktion übergeben werden, wenn sie ausgelöst wird.

Themen

Dynamische Ursprungsauswahl

Sie können das Pfadmuster in einem Cache-Verhalten verwenden, um Anforderungen an einen Ursprung zu leiten, basierend auf dem Pfad und Namen des angeforderten Objekts, z. B. `images/*.jpg`. Mit Lambda@Edge können Sie Anforderungen auch auf der Grundlage anderer Merkmale, wie z. B. der Werte in Anforderungs-Headern, an einen Ursprung weiterleiten.

Es gibt eine Reihe von Möglichkeiten, wie diese dynamische Ursprungsauswahl genutzt werden kann. So können Sie z. B. Anforderungen über Ursprünge in verschiedenen geografischen Gebieten verteilen, um den globalen Lastausgleich zu unterstützen. Oder Sie können Anforderungen selektiv an verschiedene Ursprünge weiterleiten, die jeweils eine bestimmte Funktion erfüllen: Bot-Handling, SEO-Optimierung, Authentifizierung und so weiter. Code-Beispiele, die die Verwendung dieses Features demonstrieren, finden Sie unter Inhaltsbasierte dynamische Ursprungsauswahl –Beispiele.

Im ursprünglichen CloudFront Anforderungsereignis enthält das `origin` Objekt in der Ereignisstruktur Informationen über den Ursprung, an den die Anforderung weitergeleitet werden würde, basierend auf dem Pfadmuster. Sie können die Werte im `origin`-Objekt aktualisieren,

um eine Anforderung an eine andere Herkunft weiterzuleiten. Wenn Sie das `origin`-Objekt aktualisieren, müssen Sie den Ursprung in der Verteilung nicht definieren. Sie können ein Amazon S3-Ursprungsobjekt auch durch ein benutzerdefiniertes Ursprungsobjekt ersetzen und umgekehrt. Sie können jedoch nur einen einzigen Ursprung pro Anforderung angeben; entweder einen benutzerdefinierten Ursprung oder einen Amazon S3-Ursprung, aber nicht beide.

Anforderungsereignisse

Die folgenden Themen zeigen die Struktur des Objekts, das für Viewer- und CloudFront Origin-Request-Ereignisse an eine Lambda-Funktion übergeben wird. Diese Beispiele zeigen eine `GET`-Anfrage ohne Körper. Im Anschluss an die Beispiele finden Sie eine Liste aller möglichen Felder in Viewer- und Ursprungsanforderungsereignissen.

Themen

- Beispiel für Viewer-Anforderung
- Beispiel für Ursprungsanforderung
- Anforderungsereignisfelder

Beispiel für Viewer-Anforderung

Das folgende Beispiel zeigt ein Viewer-Anfrageereignisobjekt.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "viewer-request",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfnvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "host": [
              {
                "key": "Host",
                "value": "d111111abcdef8.cloudfront.net"
              }
```

```
            ],
            "user-agent": [
              {
                "key": "User-Agent",
                "value": "curl/7.66.0"
              }
            ],
            "accept": [
              {
                "key": "accept",
                "value": "*/*"
              }
            ]
          },
          "method": "GET",
          "querystring": "",
          "uri": "/"
        }
      }
    }
  ]
}
```

Beispiel für Ursprungsanforderung

Das folgende Beispiel zeigt ein Ursprungsanforderungsereignisobjekt.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "origin-request",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfnvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "x-forwarded-for": [
              {
                "key": "X-Forwarded-For",
                "value": "203.0.113.178"
```

```
            }
          ],
          "user-agent": [
            {
              "key": "User-Agent",
              "value": "Amazon CloudFront"
            }
          ],
          "via": [
            {
              "key": "Via",
              "value": "2.0 2afae0d44e2540f472c0635ab62c232b.cloudfront.net
(CloudFront)"
            }
          ],
          "host": [
            {
              "key": "Host",
              "value": "example.org"
            }
          ],
          "cache-control": [
            {
              "key": "Cache-Control",
              "value": "no-cache"
            }
          ]
        },
        "method": "GET",
        "origin": {
          "custom": {
            "customHeaders": {},
            "domainName": "example.org",
            "keepaliveTimeout": 5,
            "path": "",
            "port": 443,
            "protocol": "https",
            "readTimeout": 30,
            "responseCompletionTimeout": 30,
            "sslProtocols": [
              "TLSv1",
              "TLSv1.1",
              "TLSv1.2"
            ]
```

```
            }
          },
          "querystring": "",
          "uri": "/"
        }
      }
    }
  ]
}
```

Anforderungsereignisfelder

Anforderungsereignisobjektdaten sind in zwei Unterobjekten enthalten: `config` (`Records.cf.config`) und `request` (`Records.cf.request`). In den folgenden Listen werden die Felder jedes Unterobjekts beschrieben.

Felder im Config-Objekt

Die folgende Liste beschreibt die Felder im `config`-Objekt (`Records.cf.config`).

`distributionDomainName` (Schreibgeschützt)

Der Domänenname der Verteilung, die der Anforderung zugeordnet ist.

`distributionID` (Schreibgeschützt)

Die ID der Verteilung, die der Anforderung zugeordnet ist.

`eventType` (Schreibgeschützt)

Der Typ des Auslösers, der der Anforderung zugeordnet ist: `viewer-request` oder `origin-request`.

`requestId` (Schreibgeschützt)

Eine verschlüsselte Zeichenfolge, die eine viewer-to-CloudFront Anfrage eindeutig identifiziert. Der `requestId`-Wert erscheint auch als CloudFront in `x-edge-request-id`-Zugriffsprotokollen. Weitere Informationen erhalten Sie unter Standardprotokollierung (Zugriffsprotokolle) und Felder in der Protokolldatei.

Felder im Anforderungsobjekt

Die folgende Liste beschreibt die Felder im `request`-Objekt (`Records.cf.request`).

`clientIp` (Schreibgeschützt)

Die IP-Adresse des Viewers, der die Anforderung gestellt hat. Wenn der Viewer einen HTTP-Proxy oder einen Load Balancer verwendet hat, um die Anfrage zu senden, entspricht der Wert der IP-Adresse des Proxys bzw. des Load Balancers.

Header (lesen/schreiben)

Die Header der Anforderung. Beachten Sie Folgendes:

- Die Schlüssel im `headers`-Objekt sind kleingeschriebene Versionen von Standard-HTTP-Header-Namen. Über diese Kleinbuchstaben-Schlüssel haben Sie Zugriff auf die Headerwerte (ohne Berücksichtigung von Groß-/Kleinschreibung).
- Jedes Header-Objekt (z. B. `headers["accept"]` oder `headers["host"]`) ist ein Array mit Schlüssel-Wert-Paaren. Für einen bestimmten Header enthält das Array ein Schlüssel-Wert-Paar für jeden Wert in der Anforderung.
- `key`enthält den Namen des Headers, bei dem die Groß- und Kleinschreibung beachtet wird, so wie er in der HTTP-Anforderung enthalten ist `HostUser-Agent`, z. B. `X-Forwarded-ForCookie`,,,, usw.
- `value` enthält den Header-Wert, wie er in der HTTP-Anforderung angezeigt wird.
- Wenn Ihre Lambda-Funktion Anfrage-Header hinzufügt oder ändert und Sie das `key`-Header-Feld nicht einschließen, fügt Lambda@Edge automatisch den Header `key` mit dem von Ihnen angegebenen Header-Namen ein. Unabhängig davon, wie Sie den Header-Namen formatiert haben, wird der automatisch eingefügte Header-Schlüssel mit einem großen Anfangsbuchstaben für jeden Teil formatiert, wobei die einzelnen Teile durch Bindestriche (-) getrennt werden.

  Beispielsweise können Sie einen Header wie den folgenden ohne Header hinzufügen: `key`

  ```
  "user-agent": [
    {
      "value": "ExampleCustomUserAgent/1.X.0"
    }
  ]
  ```

  In diesem Beispiel fügt Lambda @Edge automatisch ein ei `"key": "User-Agent"`.

Weitere Informationen zu Einschränkungen zur Verwendung von Headern finden Sie unter Einschränkungen für Edge-Funktionen.

`method` (Schreibgeschützt)

Die HTTP-Methode der Anforderung.

`querystring` (Lesen/Schreiben)

Die Abfragezeichenfolge, falls vorhanden, in der Anforderung. Wenn die Anfrage keine Abfragezeichenfolge enthält, umfasst das Ereignisobjekt dennoch `querystring` mit einem leeren Wert. Weitere Informationen zu Abfragezeichenfolgen finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern.

`uri` (Lesen/Schreiben)

Der relative Pfad des angeforderten Objekts. Wenn Ihre Lambda-Funktion den `uri`-Wert ändert, beachten Sie Folgendes:

- Der neue `uri`-Wert muss mit einem Schrägstrich (/) beginnen.
- Wenn eine Funktion den `uri`-Wert ändert, ändert sich auch das Objekt, das die Betrachter anfordert.
- Wenn eine Funktion den `uri`-Wert ändert, ändert sich weder das Cache-Verhalten für die Anforderung noch der Ursprung, an den die Anforderung weitergeleitet wird.

`body` (Lesen/Schreiben)

Der Hauptteil der HTTP-Anforderung. Die `body`-Struktur kann folgende Felder enthalten:

`inputTruncated` (Schreibgeschützt)

Ein boolesches Flag, das angibt, ob der Textkörper von Lambda@Edge abgeschnitten wurde. Weitere Informationen finden Sie unter Einschränkungen für Anforderungstext mit der Option „Text einschließen“.

`action` (Lesen/Schreiben)

Die Aktion, die Sie für den Body durchführen möchten. Für `action` gibt es die folgenden Optionen:

- `read-only:` Dies ist die Standardeinstellung. Lambda@Edge ignoriert bei der Rückgaben der Antwort von der Lambda-Funktion alle Änderungen an `encoding` oder `data`. wenn `action` schreibgeschützt ist.
- `replace:` Geben Sie diese Option an, wenn Sie den an den Ursprung gesendeten Textkörper ersetzen möchten.

`encoding` (Lesen/Schreiben)

Die Kodierung für den Body. Wenn Lambda@Edge den Textkörper der Lambda-Funktion bereitstellt, konvertiert es den Textkörper zuerst zu base64-encoding. Wenn Sie `replace` für die `action` auswählen, um den Textkörper zu ersetzen, können Sie die Kodierung `base64` (Standard) oder `text` verwenden. Wenn Sie `encoding` als `base64` festlegen, der Body jedoch kein gültiges base64 ist, gibt CloudFront einen Fehler zurück.

`data` (Lesen/Schreiben)

Der Inhalt des Anforderungs-Bodys.

`origin` (lesen/schreiben) (nur Ursprungsereignisse)

Der Ursprung, an den die Anfrage gesendet werden soll. Die `origin` Struktur muss genau einen Ursprung enthalten, der ein benutzerdefinierter Ursprung oder ein Amazon S3 S3-Ursprung sein kann.

Je nachdem, welchen Herkunftstyp Sie angeben (benutzerdefiniert oder Amazon S3 S3-Herkunft), müssen Sie in Ihrer Anfrage die folgenden Felder angeben:

`customHeaders` (Lesen/Schreiben) (benutzerdefinierte und Amazon S3-Ursprünge)

(Optional) Sie können benutzerdefinierte Header in die Anfrage aufnehmen, indem Sie für jeden benutzerdefinierten Header einen Header-Namen und ein Wertepaar angeben. Sie können keine Header hinzufügen, die nicht erlaubt sind, und ein Header mit dem gleichen Namen kann in `Records.cf.request.headers` nicht vorhanden sein. Die Hinweise zu Anforderungs-Headern gelten auch für benutzerdefinierte Header. Weitere Informationen erhalten Sie unter Benutzerdefinierte Header, die nicht zu CloudFront ursprünglichen Anfragen hinzugefügt werden können und Einschränkungen für Edge-Funktionen.

`domainName` (Lesen/Schreiben) (benutzerdefinierte und Amazon S3-Ursprünge)

Der Domänenname des Ursprungs. Der Domänenname darf nicht leer sein.

- Für benutzerdefinierte Ursprünge – Geben Sie einen DNS-Domänennamen an, z. B. `www.example.com` Der Domänenname darf keinen Doppelpunkt (:) enthalten und keine IP-Adresse sein. Der Domänenname kann bis zu 253 Zeichen lang sein.
- Für Amazon S3-Ursprünge – Geben Sie den DNS-Domänennamen des Amazon S3-Buckets an, z. B. `amzn-s3-demo-bucket.s3.eu-west-1.amazonaws.com`. Der Name kann bis zu 128 Zeichen lang sein und muss in Kleinbuchstaben geschrieben werden.

`path` (Lesen/Schreiben) (benutzerdefinierte und Amazon S3-Ursprünge)

Der Verzeichnispfad auf dem Ursprung, aus dem die Anforderung den Inhalt abrufen soll. Der Pfad sollte mit einem Schrägstrich (/) beginnen, aber nicht damit enden (z. B. nicht mit `example-path/`). Nur für benutzerdefinierte Ursprünge sollte der Pfad URL-codiert sein und eine Maximallänge von 255 Zeichen haben.

`keepaliveTimeout` (lesen/schreiben) (nur benutzerdefinierte Ursprünge)

Wie lange (in Sekunden) CloudFront soll versucht werden, die Verbindung zum Ursprung aufrechtzuerhalten, nachdem das letzte Paket der Antwort empfangen wurde. Der Wert muss eine Zahl zwischen 1—120 (einschließlich) sein.

`port` (lesen/schreiben) (nur benutzerdefinierte Ursprünge)

Der Port, zu dem eine Verbindung zu Ihrem benutzerdefinierten Ursprung hergestellt werden CloudFront soll. Der Port muss 80, 443 oder 1024 bis einschließlich 65535 sein.

`protocol` (lesen/schreiben) (nur benutzerdefinierte Ursprünge)

Das Verbindungsprotokoll, das verwendet CloudFront werden soll, wenn Sie eine Verbindung zu Ihrem Origin herstellen. Dabei kann es sich um den Wert `http` oder `https` handeln.

`readTimeout` (Lesen/Schreiben) (benutzerdefinierte und Amazon S3-Ursprünge)

Wie lange (in Sekunden) CloudFront sollten Sie auf eine Antwort warten, nachdem Sie eine Anfrage an Ihren Absender gesendet haben. Dies gibt auch an, wie lange CloudFront warten soll, nachdem das Paket einer Antwort empfangen wurde, bevor das nächste Paket empfangen wird. Der Wert muss eine Zahl zwischen 1—120 (einschließlich) sein.

Wenn Sie ein höheres Kontingent benötigen, finden Sie weitere Informationen unter Antworttimeout pro Absender.

`responseCompletionTimeout` (Lesen/Schreiben) (benutzerdefinierte und Amazon S3-Ursprünge)

Die Zeit (in Sekunden), in der eine Anfrage vom CloudFront Absender geöffnet bleiben und auf eine Antwort warten kann. Wenn bis zu diesem Zeitpunkt noch keine vollständige Antwort vom Ursprung eingegangen ist, wird die Verbindung CloudFront beendet.

Der Wert für `responseCompletionTimeout` muss gleich oder größer als der Wert für sein`readTimeout`. Wenn Sie diesen Wert auf 0 setzen, werden alle zuvor von Ihnen

festgelegten Werte entfernt und der Standardwert wird wiederhergestellt. Sie können dies auch erreichen, indem Sie das `responseCompletionTimeout` Feld aus der Ereignisanforderung löschen.

`sslProtocols` (lesen/schreiben) (nur benutzerdefinierte Ursprünge)

Das SSL/TLS Mindestprotokoll, das CloudFront Sie beim Aufbau einer HTTPS-Verbindung mit Ihrem Ursprung verwenden können. Werte können einer der folgenden sein: `TLSv1.2`, `TLSv1.1`, `TLSv1` oder `SSLv3`.

`authMethod` (Lesen/Schreiben) (nur Amazon S3-Ursprünge)

Wenn Sie eine Ursprungszugriffsidentität (OAI) verwenden, setzen Sie dieses Feld auf `origin-access-identity`. Wenn Sie keine OAI verwenden, setzen Sie es auf `none`. Wenn Sie `authMethod` auf „`origin-access-identity`“ festlegen, gibt es mehrere Anforderungen:

- Sie müssen die `region` angeben (siehe das folgende Feld).
- Sie müssen dieselbe OAI verwenden, wenn Sie die Anforderung von einem Amazon S3-Ursprung zu einem anderen ändern.
- Sie können keine OAI verwenden, wenn Sie die Anforderung von einem benutzerdefinierten Ursprung zu einem Amazon-S3-Ursprung ändern.

> **Note**
>
> Dieses Feld unterstützt keine Ursprungszugriffssteuerung (OAC).

`region` (Lesen/Schreiben) (nur Amazon S3-Ursprünge)

Die AWS Region Ihres Amazon S3 S3-Buckets. Dies ist nur erforderlich, wenn Sie `authMethod` auf „`origin-access-identity`“ festlegen.

Antwortereignisse

Die folgenden Themen zeigen die Struktur des Objekts, das für Viewer- und Origin-Response-Ereignisse an eine Lambda-Funktion CloudFront übergeben wird. Im Anschluss an die Beispiele finden Sie eine Liste aller möglichen Felder in Viewer- und Ursprungantwortereignissen.

Themen

- Beispiel für Ursprungsantwort
- Beispiel für Viewer-Antwort
- Antwortereignisfelder

Beispiel für Ursprungsantwort

Das folgende Beispiel zeigt ein Ursprungsantwortereignisobjekt.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "origin-response",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfnvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "x-forwarded-for": [
              {
                "key": "X-Forwarded-For",
                "value": "203.0.113.178"
              }
            ],
            "user-agent": [
              {
                "key": "User-Agent",
                "value": "Amazon CloudFront"
              }
            ],
            "via": [
              {
                "key": "Via",
                "value": "2.0 8f22423015641505b8c857a37450d6c0.cloudfront.net
 (CloudFront)"
              }
            ],
            "host": [
              {
```

```
            "key": "Host",
            "value": "example.org"
          }
        ],
        "cache-control": [
          {
            "key": "Cache-Control",
            "value": "no-cache"
          }
        ]
      },
      "method": "GET",
      "origin": {
        "custom": {
          "customHeaders": {},
          "domainName": "example.org",
          "keepaliveTimeout": 5,
          "path": "",
          "port": 443,
          "protocol": "https",
          "readTimeout": 30,
          "responseCompletionTimeout": 30,
          "sslProtocols": [
            "TLSv1",
            "TLSv1.1",
            "TLSv1.2"
          ]
        }
      },
      "querystring": "",
      "uri": "/"
    },
    "response": {
      "headers": {
        "access-control-allow-credentials": [
          {
            "key": "Access-Control-Allow-Credentials",
            "value": "true"
          }
        ],
        "access-control-allow-origin": [
          {
            "key": "Access-Control-Allow-Origin",
            "value": "*"
```

```
            }
          ],
          "date": [
            {
              "key": "Date",
              "value": "Mon, 13 Jan 2020 20:12:38 GMT"
            }
          ],
          "referrer-policy": [
            {
              "key": "Referrer-Policy",
              "value": "no-referrer-when-downgrade"
            }
          ],
          "server": [
            {
              "key": "Server",
              "value": "ExampleCustomOriginServer"
            }
          ],
          "x-content-type-options": [
            {
              "key": "X-Content-Type-Options",
              "value": "nosniff"
            }
          ],
          "x-frame-options": [
            {
              "key": "X-Frame-Options",
              "value": "DENY"
            }
          ],
          "x-xss-protection": [
            {
              "key": "X-XSS-Protection",
              "value": "1; mode=block"
            }
          ],
          "content-type": [
            {
              "key": "Content-Type",
              "value": "text/html; charset=utf-8"
            }
          ],
```

```
            "content-length": [
              {
                "key": "Content-Length",
                "value": "9593"
              }
            ]
          },
          "status": "200",
          "statusDescription": "OK"
        }
      }
    }
  ]
}
```

Beispiel für Viewer-Antwort

Das folgende Beispiel zeigt ein Viewer-Antwortereignisobjekt.

```
{
  "Records": [
    {
      "cf": {
        "config": {
          "distributionDomainName": "d111111abcdef8.cloudfront.net",
          "distributionId": "EDFDVBD6EXAMPLE",
          "eventType": "viewer-response",
          "requestId": "4TyzHTaYWb1GX1qTfsHhEqV6HUDd_BzoBZnwfnvQc_1oF26ClkoUSEQ=="
        },
        "request": {
          "clientIp": "203.0.113.178",
          "headers": {
            "host": [
              {
                "key": "Host",
                "value": "d111111abcdef8.cloudfront.net"
              }
            ],
            "user-agent": [
              {
                "key": "User-Agent",
                "value": "curl/7.66.0"
              }
            ],
```

```
            "accept": [
              {
                "key": "accept",
                "value": "*/*"
              }
            ]
          },
          "method": "GET",
          "querystring": "",
          "uri": "/"
        },
        "response": {
          "headers": {
            "access-control-allow-credentials": [
              {
                "key": "Access-Control-Allow-Credentials",
                "value": "true"
              }
            ],
            "access-control-allow-origin": [
              {
                "key": "Access-Control-Allow-Origin",
                "value": "*"
              }
            ],
            "date": [
              {
                "key": "Date",
                "value": "Mon, 13 Jan 2020 20:14:56 GMT"
              }
            ],
            "referrer-policy": [
              {
                "key": "Referrer-Policy",
                "value": "no-referrer-when-downgrade"
              }
            ],
            "server": [
              {
                "key": "Server",
                "value": "ExampleCustomOriginServer"
              }
            ],
            "x-content-type-options": [
```

```
            {
              "key": "X-Content-Type-Options",
              "value": "nosniff"
            }
          ],
          "x-frame-options": [
            {
              "key": "X-Frame-Options",
              "value": "DENY"
            }
          ],
          "x-xss-protection": [
            {
              "key": "X-XSS-Protection",
              "value": "1; mode=block"
            }
          ],
          "age": [
            {
              "key": "Age",
              "value": "2402"
            }
          ],
          "content-type": [
            {
              "key": "Content-Type",
              "value": "text/html; charset=utf-8"
            }
          ],
          "content-length": [
            {
              "key": "Content-Length",
              "value": "9593"
            }
          ]
        },
        "status": "200",
        "statusDescription": "OK"
      }
    }
  }
]
}
```

Antwortereignisfelder

Die Daten des Antwortereignisobjekts sind in drei Unterobjekten enthalten: `config` (`Records.cf.config`), `request` (`Records.cf.request`) und `response` (`Records.cf.response`). Weitere Hinweise zu den Feldern im Anforderungsobjekt finden Sie unter Felder im Anforderungsobjekt. In den folgenden Listen werden die Felder in den Unterobjekten `config` und `response` beschrieben.

Felder im Config-Objekt

Die folgende Liste beschreibt die Felder im `config`-Objekt (`Records.cf.config`).

`distributionDomainName` (Schreibgeschützt)

Der Domänenname der Verteilung, die der Antwort zugeordnet ist.

`distributionID` (Schreibgeschützt)

Die ID der Verteilung, die der Antwort zugeordnet ist.

`eventType` (Schreibgeschützt)

Der Typ des Auslösers, der der Antwort zugeordnet ist: `origin-response` oder `viewer-response`.

`requestId` (Schreibgeschützt)

Eine verschlüsselte Zeichenfolge, die die viewer-to-CloudFront Anfrage, der diese Antwort zugeordnet ist, eindeutig identifiziert. Der `requestId` Wert erscheint auch in den CloudFront Zugriffsprotokollen als`x-edge-request-id`. Weitere Informationen erhalten Sie unter Standardprotokollierung (Zugriffsprotokolle) und Felder in der Protokolldatei.

Felder im Antwortobjekt

Die folgende Liste beschreibt die Felder im `response`-Objekt (`Records.cf.response`). Hinweise zum Generieren einer HTTP-Antwort mithilfe einer Lambda@Edge-Funktion finden Sie unter Generieren Sie HTTP-Antworten in Anforderungsauslösern.

`headers` (Lesen/Schreiben)

Die Header der Antwort. Beachten Sie Folgendes:

- Die Schlüssel im `headers`-Objekt sind kleingeschriebene Versionen von Standard-HTTP-Header-Namen. Über diese Kleinbuchstaben-Schlüssel haben Sie Zugriff auf die Headerwerte (ohne Berücksichtigung von Groß-/Kleinschreibung).
- Jedes Header-Objekt (z. B. `headers["content-type"]` oder `headers["content-length"]`) ist ein Array mit Schlüssel-Wert-Paaren. Für einen bestimmten Header enthält das Array ein Schlüssel-Wert-Paar für jeden Wert in der generierten Antwort.
- `key`enthält den Namen des Headers, bei dem die Groß- und Kleinschreibung beachtet wird, so wie er in der HTTP-Antwort erscheint, z. B. `Content-TypeContent-Length`,`Cookie`,, und so weiter.
- `value` enthält den Header-Wert, wie er in der HTTP-Antwort angezeigt wird.
- Wenn Ihre Lambda-Funktion Antwort-Header hinzufügt oder ändert und Sie das `key`-Header-Feld nicht einschließen, fügt Lambda@Edge automatisch den Header `key` mit dem von Ihnen angegebenen Header-Namen ein. Unabhängig davon, wie Sie den Header-Namen formatiert haben, wird der automatisch eingefügte Header-Schlüssel mit einem großen Anfangsbuchstaben für jeden Teil formatiert, wobei die einzelnen Teile durch Bindestriche (-) getrennt werden.

  Beispielsweise können Sie einen Header wie den folgenden ohne Header hinzufügen: `key`

  ```
  "content-type": [
    {
      "value": "text/html;charset=UTF-8"
    }
  ]
  ```

  In diesem Beispiel fügt Lambda @Edge automatisch ein ei `"key": "Content-Type"`.

Weitere Informationen zu Einschränkungen zur Verwendung von Headern finden Sie unter Einschränkungen für Edge-Funktionen.

`status`

Den HTTP-Statuscode der Antwort.

`statusDescription`

Die HTTP-Statusbeschreibung der Antwort.

Arbeiten Sie mit Anfragen und Antworten

Informationen zur Verwendung von Lambda @Edge -Anfragen und -Antworten finden Sie in den folgenden Themen:

Themen

Verwenden Sie Lambda @Edge -Funktionen mit Origin-Failover

Sie können Lambda @Edge -Funktionen mit CloudFront Distributionen verwenden, die Sie mit Ursprungsgruppen eingerichtet haben, z. B. für Origin-Failover, das Sie konfigurieren, um eine hohe Verfügbarkeit sicherzustellen. Um eine Lambda-Funktion mit einer Ursprungsgruppe zu verwenden, geben Sie die Funktion in einem Ursprungsanfragen- oder Ursprungsantwort-Auslöser für eine Ursprungsgruppe an, wenn Sie das Zwischenspeicher-Verhalten erstellen.

Weitere Informationen finden Sie hier:

- Erstellen Sie Ursprungsgruppen: Erstellen Sie eine Ursprungsgruppe
- Funktionsweise von Origin Failover mit Lambda@Edge: Verwenden von Origin Failover mit Lambda@Edge-Funktionen

Generieren Sie HTTP-Antworten in Anforderungsauslösern

Wenn Sie CloudFront eine Anfrage erhalten, können Sie eine Lambda-Funktion verwenden, um eine HTTP-Antwort zu generieren, die direkt an den Betrachter CloudFront zurückkehrt, ohne die Antwort an den Ursprung weiterzuleiten. Die Generierung von HTTP-Antworten reduziert die Auslastung des Ursprungs und reduziert in der Regel auch die Latenzzeit für den Viewer.

Einige häufige Szenarien für die Generierung von HTTP-Antworten sind die folgenden:

- Rückgabe einer kleinen Website an den Viewer.
- Rückgabe eines HTTP 301- oder 302-Statuscodes, um den Benutzer auf eine andere Webseite umzuleiten.

- Rückgabe eines HTTP 401-Statuscodes an den Viewer, wenn sich der Benutzer nicht authentifiziert hat.

Eine Lambda@Edge-Funktion kann eine HTTP-Antwort generieren, wenn die folgenden CloudFront-Ereignisse auftreten:

Viewer-Anforderungsereignisse

Wenn eine Funktion durch ein Viewer-Anforderungsereignis ausgelöst wird, CloudFront gibt sie die Antwort an den Betrachter zurück und speichert sie nicht im Cache.

Ursprungsanforderungsereignisse

Wenn eine Funktion durch ein Origin-Anforderungsereignis ausgelöst wird, CloudFront wird im Edge-Cache nach einer Antwort gesucht, die zuvor von der Funktion generiert wurde.

- Wenn sich die Antwort im Cache befindet, wird die Funktion nicht ausgeführt und CloudFront gibt die zwischengespeicherte Antwort an den Viewer zurück.
- Wenn sich die Antwort nicht im Cache befindet, wird die Funktion ausgeführt, CloudFront gibt die Antwort an den Betrachter zurück und speichert sie ebenfalls im Cache.

Beispiel-Code zum Generieren von HTTP-Antworten finden Sie unter Beispielfunktionen für Lambda@Edge. Sie können auch die HTTP-Antworten in Antwortauslösern ersetzen. Weitere Informationen finden Sie unter Aktualisieren Sie die HTTP-Antworten in den ursprünglichen Antwortauslösern.

Programmiermodell

Dieser Abschnitt enthält Informationen über das Programmiermodell für die Nutzung von Lambda@Edge zum Generieren von HTTP-Antworten.

Themen

- Antwortobjekt
- Fehler
- Pflichtfelder

Antwortobjekt

Die Antwort, die Sie als `result`-Parameter der `callback`-Methode zurückgeben, sollte folgenden Aufbau haben (beachten Sie, dass nur das `status`-Feld erforderlich ist).

```
const response = {
    body: 'content',
    bodyEncoding: 'text' | 'base64',
    headers: {
        'header name in lowercase': [{
           key: 'header name in standard case',
           value: 'header value'
         }],
         ...
    },
    status: 'HTTP status code (string)',
    statusDescription: 'status description'
};
```

Das Antwortobjekt kann die folgenden Werte enthalten:

`body`

Der Text, falls vorhanden, den Sie in der generierten Antwort zurückgeben CloudFront möchten.

`bodyEncoding`

Die Kodierung für den Wert, den Sie in `body` festgelegt haben. Die einzigen gültigen Codierungen sind `text` und `base64`. Wenn Sie `body` in das `response` Objekt aufnehmen, aber weglassen`bodyEncoding`, wird der CloudFront Hauptteil als Text behandelt.

Wenn Sie `bodyEncoding` als `base64` festlegen, der Body jedoch kein gültiges base64 ist, gibt CloudFront einen Fehler zurück.

`headers`

Header, die Sie in der generierten Antwort zurückgeben möchten CloudFront . Beachten Sie Folgendes:

- Die Schlüssel im `headers`-Objekt sind kleingeschriebene Versionen von Standard-HTTP-Header-Namen. Über diese Kleinbuchstaben-Schlüssel haben Sie Zugriff auf die Headerwerte (ohne Berücksichtigung von Groß-/Kleinschreibung).

- Jeder Header (z. B. `headers["accept"]` oder `headers["host"]`) ist ein Array mit Schlüssel-Wert-Paaren. Für einen bestimmten Header enthält das Array ein Schlüssel-Wert-Paar für jeden Wert in der generierten Antwort.
- `key` (optional) ist der von Groß- und Kleinschreibung unabhängige Name des Headers, wie er in einer HTTP-Anforderung erscheint (z. B. `accept` oder `host`).
- Geben Sie `value` als Header-Wert an.
- Wenn Sie den Header-Schlüsselteil des Schlüssel-Wert-Paares nicht aufnehmen, fügt Lambda@Edge automatisch einen Header-Schlüssel mit dem von Ihnen angegebenen Header-Namen ein. Unabhängig davon, wie Sie den Header-Namen formatiert haben, wird der automatisch eingefügte Header-Schlüssel mit einem großen Anfangsbuchstaben für jeden Teil formatiert, wobei die einzelnen Teile durch Bindestriche (-) getrennt werden.

  Beispielsweise können Sie einen Header wie den folgenden ohne Header-Schlüssel hinzufügen: `'content-type': [{ value: 'text/html;charset=UTF-8' }]`

  In diesem Beispiel erstellt Lambda@Edge den folgenden Header-Schlüssel: `Content-Type`.

Weitere Informationen zu Einschränkungen zur Verwendung von Headern finden Sie unter Einschränkungen für Edge-Funktionen.

`status`

Den HTTP-Statuscode . Geben Sie den Statuscode als Zeichenfolge an. CloudFront verwendet den bereitgestellten Statuscode für Folgendes:

- Rückgabe in der Antwort
- Cache im CloudFront Edge-Cache, wenn die Antwort durch eine Funktion generiert wurde, die durch ein Origin-Request-Ereignis ausgelöst wurde
- Loggen Sie sich ein CloudFront Standardprotokollierung (Zugriffsprotokolle)

Wenn der `status`-Wert nicht zwischen 200 und 599 liegt, gibt CloudFront einen Fehler an den Betrachter zurück.

`statusDescription`

Die Beschreibung, die CloudFront Sie zusammen mit dem HTTP-Statuscode in der Antwort zurückgeben möchten. Sie müssen keine Standardbeschreibungen verwenden (z. B. `OK` für einen HTTP-Statuscode 200).

Fehler

Es folgen mögliche Fehler für generierte HTTP-Antworten.

Antwort enthält einen Body und legt 204 (No Content) als Status fest

Wenn eine Funktion durch eine Viewer-Anfrage ausgelöst wird, CloudFront gibt sie dem Viewer einen HTTP 502-Statuscode (Bad Gateway) zurück, wenn beide der folgenden Bedingungen zutreffen:

- Der Wert von `status` ist 204 (No Content)
- Die Antwort enthält einen Wert für `body`

Der Grund hierfür ist, dass Lambda@Edge eine optionale Einschränkung aus RFC 2616 umsetzt, die besagt, dass eine HTTP 204-Antwort keinen Nachrichtentext zu enthalten braucht.

Beschränkungen für die Größe der generierten Antwort

Die maximale Größe einer durch eine Lambda-Funktion generierten Antwort hängt von dem Ereignis ab, das die Funktion ausgelöst hat:

- Viewer-Anfrage-Ereignisse – 40 KB
- Ursprungsanfrageereignisse – 1 MB

Wenn die Antwort größer als die zulässige Größe ist, wird ein HTTP 502-Statuscode (Bad Gateway) an den Betrachter CloudFront zurückgegeben.

Pflichtfelder

Das Feld `status` ist ein Pflichtfeld.

Alle anderen Felder sind optional.

Aktualisieren Sie die HTTP-Antworten in den ursprünglichen Antwortauslösern

Wenn eine HTTP-Antwort vom Ursprungsserver CloudFront empfängt und dem Cache-Verhalten ein Origin-Response-Trigger zugeordnet ist, können Sie die HTTP-Antwort so ändern, dass die vom Ursprung zurückgegebenen Antworten außer Kraft gesetzt werden.

Einige häufige Szenarien für die Aktualisierung von HTTP-Antworten sind die folgenden:

- Ändern des Status, um einen HTTP 200-Statuscode festzulegen, und Erstellen statischer Body-Inhalte für die Rückgabe an den Viewer, wenn ein Ursprung einen Fehlerstatuscode (4xx oder 5xx)

zurückgibt. Einen Beispiel-Code finden Sie unter Beispiel: Verwenden Sie einen Origin-Response-Trigger, um den Fehlerstatuscode auf 200 zu aktualisieren.

- Ändern des Status, um einen HTTP 301- oder HTTP 302-Statuscode festzulegen, um den Benutzer auf eine andere Website umzuleiten, wenn ein Ursprung einen Fehlerstatuscode zurückgibt (4xx oder 5xx). Einen Beispiel-Code finden Sie unter Beispiel: Verwenden Sie einen Origin-Response-Trigger, um den Fehlerstatuscode auf 302 zu aktualisieren.

> **Note**
>
> Die Funktion muss einen Statuswert zwischen `200` und `599` (einschließlich) zurückgeben, andernfalls CloudFront gibt sie einen Fehler an den Viewer zurück.

Sie können auch die HTTP-Antworten in Viewer- und Ursprungsanfrageereignissen ersetzen. Weitere Informationen finden Sie unter Generieren Sie HTTP-Antworten in Anforderungsauslösern.

Wenn Sie mit der HTTP-Antwort arbeiten, stellt Lambda@Edge den Textkörper, der vom Ursprungsserver zurückgegeben wird, nicht für den Ursprungsantwortauslöser bereit. Sie können einen statischen Inhaltstext erzeugen, indem Sie ihn auf den gewünschten Wert setzen, oder den Text innerhalb der Funktion entfernen, indem Sie den Wert auf leer setzen. Wenn Sie das Textkörperfeld in Ihrer Funktion nicht aktualisieren, wird der ursprüngliche Textkörper, der vom Ursprungsserver zurückgegeben wird, an den Viewer zurückgegeben.

Greifen Sie auf den Anforderungstext zu, indem Sie die Option „Text einschließen“ wählen

Sie können jetzt auswählen, ob Lambda@Edge den Textkörper in einer Anforderung für eine nicht schreibgeschützte HTTP-Methode (POST, PUT, DELETE etc.) weitergeben soll, sodass Sie in Ihrer Lambda-Funktion darauf zugreifen können. Sie können den Lesezugriff wählen oder angeben, dass Sie den Textkörper ersetzen möchten.

Um diese Option zu aktivieren, wählen Sie Include Body (Body einbeziehen) aus, wenn Sie einen CloudFront -Auslöser für Ihre für ein Betrachteranforderungs- oder Ursprungsanforderungsereignis vorgesehene Funktion erstellen. Weitere Informationen finden Sie unter Trigger für eine Lambda @Edge -Funktion hinzufügen. Mehr zur Nutzung von Include Body (Textkörper einbeziehen) mit Ihrer Funktion finden Sie unter Lambda@Edge-Ereignisstruktur.

Sie können dieses Feature in den folgenden Szenarien nutzen:

- Verarbeitung von Webformularen (z. B. "Kontakt"-Formulare) ohne Rückgabe von Kundeneingabedaten an den Ursprungs-Server
- Erfassen von Web-Beacon-Daten, die von Viewer-Browsern gesendet werden, und Verarbeiten am Edge

Einen Beispiel-Code finden Sie unter Beispielfunktionen für Lambda@Edge.

> **Note**
>
> Wenn der Anforderungs-Body groß ist, wird er von Lambda@Edge abgeschnitten. Ausführliche Informationen zur maximalen Größe und Kürzung finden Sie unter Einschränkungen für Anforderungstext mit der Option „Text einschließen“.

Beispielfunktionen für Lambda@Edge

Sehen Sie sich die folgenden Beispiele für die Verwendung von Lambda-Funktionen mit Amazon CloudFront an.

> **Note**
>
> Wenn Sie Runtime Node.js 18 oder höher für Ihre Lambda @Edge -Funktion wählen, wird automatisch eine `index.mjs` Datei für Sie erstellt. Um die folgenden Codebeispiele zu verwenden, benennen Sie die `index.mjs` Datei `index.js` stattdessen in um.

Themen

Allgemeine Beispiele

Die folgenden Beispiele zeigen gängige Verwendungsmöglichkeiten von Lambda @Edge in CloudFront.

Themen

- Beispiel: Testen A/B
- Beispiel: Überschreiben Sie einen Antwort-Header

Beispiel: Testen A/B

Sie können das folgende Beispiel verwenden, um zwei verschiedene Versionen eines Images zu testen, ohne Weiterleitungen zu erstellen oder die URL zu ändern. In diesem Beispiel werden die Cookies in der Viewer-Anfrage gelesen und die Anforderungs-URL wird entsprechend geändert. Wenn der Betrachter kein Cookie mit einem der erwarteten Werte sendet, weist das Beispiel den Betrachter nach dem Zufallsprinzip einem der URLs folgenden zu.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    if (request.uri !== '/experiment-pixel.jpg') {
        // do not process if this is not an A-B test request
        callback(null, request);
        return;
    }

    const cookieExperimentA = 'X-Experiment-Name=A';
    const cookieExperimentB = 'X-Experiment-Name=B';
    const pathExperimentA = '/experiment-group/control-pixel.jpg';
    const pathExperimentB = '/experiment-group/treatment-pixel.jpg';

    /*
     * Lambda at the Edge headers are array objects.
     *
     * Client may send multiple Cookie headers, i.e.:
     * > GET /viewerRes/test HTTP/1.1
```

```
     * > User-Agent: curl/7.18.1 (x86_64-unknown-linux-gnu) libcurl/7.18.1
 OpenSSL/1.0.1u zlib/1.2.3
     * > Cookie: First=1; Second=2
     * > Cookie: ClientCode=abc
     * > Host: example.com
     *
     * You can access the first Cookie header at headers["cookie"][0].value
     * and the second at headers["cookie"][1].value.
     *
     * Header values are not parsed. In the example above,
     * headers["cookie"][0].value is equal to "First=1; Second=2"
     */
    let experimentUri;
    if (headers.cookie) {
        for (let i = 0; i < headers.cookie.length; i++) {
            if (headers.cookie[i].value.indexOf(cookieExperimentA) >= 0) {
                console.log('Experiment A cookie found');
                experimentUri = pathExperimentA;
                break;
            } else if (headers.cookie[i].value.indexOf(cookieExperimentB) >= 0) {
                console.log('Experiment B cookie found');
                experimentUri = pathExperimentB;
                break;
            }
        }
    }

    if (!experimentUri) {
        console.log('Experiment cookie has not been found. Throwing dice...');
        if (Math.random() < 0.75) {
            experimentUri = pathExperimentA;
        } else {
            experimentUri = pathExperimentB;
        }
    }

    request.uri = experimentUri;
    console.log(`Request uri set to "${request.uri}"`);
    callback(null, request);
};
```

Python

```
import json
import random

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    if request['uri'] != '/experiment-pixel.jpg':
        # Not an A/B Test
        return request

    cookieExperimentA, cookieExperimentB = 'X-Experiment-Name=A', 'X-Experiment-
Name=B'
    pathExperimentA, pathExperimentB = '/experiment-group/control-pixel.jpg', '/
experiment-group/treatment-pixel.jpg'

    '''
    Lambda at the Edge headers are array objects.

    Client may send multiple cookie headers. For example:
    > GET /viewerRes/test HTTP/1.1
    > User-Agent: curl/7.18.1 (x86_64-unknown-linux-gnu) libcurl/7.18.1
 OpenSSL/1.0.1u zlib/1.2.3
    > Cookie: First=1; Second=2
    > Cookie: ClientCode=abc
    > Host: example.com

    You can access the first Cookie header at headers["cookie"][0].value
    and the second at headers["cookie"][1].value.

    Header values are not parsed. In the example above,
    headers["cookie"][0].value is equal to "First=1; Second=2"
    '''

    experimentUri = ""

    for cookie in headers.get('cookie', []):
        if cookieExperimentA in cookie['value']:
            print("Experiment A cookie found")
            experimentUri = pathExperimentA
            break
        elif cookieExperimentB in cookie['value']:
```

```
            print("Experiment B cookie found")
            experimentUri = pathExperimentB
            break

    if not experimentUri:
        print("Experiment cookie has not been found. Throwing dice...")
        if random.random() < 0.75:
            experimentUri = pathExperimentA
        else:
            experimentUri = pathExperimentB

    request['uri'] = experimentUri
    print(f"Request uri set to {experimentUri}")
    return request
```

Beispiel: Überschreiben Sie einen Antwort-Header

Das folgende Beispiel zeigt, wie Sie den Wert eines Antwortheaders basierend auf dem Wert eines anderen Headers modifizieren:

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
   const response = event.Records[0].cf.response;
   const headers = response.headers;

   const headerNameSrc = 'X-Amz-Meta-Last-Modified';
   const headerNameDst = 'Last-Modified';

   if (headers[headerNameSrc.toLowerCase()]) {
      headers[headerNameDst.toLowerCase()] = [
         headers[headerNameSrc.toLowerCase()][0],
      ];
      console.log(`Response header "${headerNameDst}" was set to ` +
               `"${headers[headerNameDst.toLowerCase()][0].value}"`);
   }

   callback(null, response);
};
```

Python

```
import json

def lambda_handler(event, context):
    response = event["Records"][0]["cf"]["response"]
    headers = response["headers"]

    headerNameSrc = "X-Amz-Meta-Last-Modified"
    headerNameDst = "Last-Modified"

    if headers.get(headerNameSrc.lower(), None):
        headers[headerNameDst.lower()] = [headers[headerNameSrc.lower()][0]]
        print(f"Response header {headerNameDst.lower()} was set to
 {headers[headerNameSrc.lower()][0]}")

    return response
```

Antworten generieren — Beispiele

Die folgenden Beispiele zeigen, wie Sie Lambda @Edge verwenden können, um Antworten zu generieren.

Themen

- Beispiel: Statischen Inhalt bereitstellen (generierte Antwort)
- Beispiel: Generieren Sie eine HTTP-Weiterleitung (generierte Antwort)

Beispiel: Statischen Inhalt bereitstellen (generierte Antwort)

Das folgende Beispiel zeigt, wie Sie mit einer Lambda-Funktion Inhalte von statischen Websites bereitstellen können, wodurch sich die Verarbeitungslast auf dem Ursprungs-Server und die Latenz insgesamt verringert.

> **Note**
>
> Sie können HTTP-Antworten für Viewer-Anfrage- und Ursprungsanfrageereignisse generieren. Weitere Informationen finden Sie unter the section called “Generieren Sie HTTP-Antworten in Anforderungsauslösern”.

Sie können auch den Text der HTTP-Antwort in Ursprungsantwortereignissen ersetzen oder entfernen. Weitere Informationen finden Sie unter the section called “Aktualisieren Sie die HTTP-Antworten in den ursprünglichen Antwortauslösern”.

Node.js

```
'use strict';

const content = `
<\!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Simple Lambda@Edge Static Content Response</title>
  </head>
  <body>
    <p>Hello from Lambda@Edge!</p>
  </body>
</html>
`;

exports.handler = (event, context, callback) => {
    /*
     * Generate HTTP OK response using 200 status code with HTML body.
     */
    const response = {
        status: '200',
        statusDescription: 'OK',
        headers: {
            'cache-control': [{
                key: 'Cache-Control',
                value: 'max-age=100'
            }],
            'content-type': [{
                key: 'Content-Type',
                value: 'text/html'
            }]
        },
        body: content,
    };
    callback(null, response);
```

```
};
```

Python

```
import json

CONTENT = """
<\!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Simple Lambda@Edge Static Content Response</title>
</head>
<body>
    <p>Hello from Lambda@Edge!</p>
</body>
</html>
"""

def lambda_handler(event, context):
    # Generate HTTP OK response using 200 status code with HTML body.
    response = {
        'status': '200',
        'statusDescription': 'OK',
        'headers': {
            'cache-control': [
                {
                    'key': 'Cache-Control',
                    'value': 'max-age=100'
                }
            ],
            "content-type": [
                {
                    'key': 'Content-Type',
                    'value': 'text/html'
                }
            ]
        },
        'body': CONTENT
    }
    return response
```

Beispiel: Generieren Sie eine HTTP-Weiterleitung (generierte Antwort)

Das folgende Beispiel zeigt, wie Sie eine HTTP-Umleitung generieren.

> **Note**
>
> Sie können HTTP-Antworten für Viewer-Anfrage- und Ursprungsanfrageereignisse generieren. Weitere Informationen finden Sie unter Generieren Sie HTTP-Antworten in Anforderungsauslösern.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    /*
     * Generate HTTP redirect response with 302 status code and Location header.
     */
    const response = {
        status: '302',
        statusDescription: 'Found',
        headers: {
            location: [{
                key: 'Location',
                value: 'https://docs.aws.amazon.com/lambda/latest/dg/lambda-
edge.html',
            }],
        },
    };
    callback(null, response);
};
```

Python

```
def lambda_handler(event, context):

    # Generate HTTP redirect response with 302 status code and Location header.

    response = {
        'status': '302',
        'statusDescription': 'Found',
```

```
        'headers': {
            'location': [{
                'key': 'Location',
                'value': 'https://docs.aws.amazon.com/lambda/latest/dg/lambda-
edge.html'
            }]
        }
    }

    return response
```

Abfragezeichenfolgen — Beispiele

Die folgenden Beispiele zeigen, wie Sie Lambda @Edge mit Abfragezeichenfolgen verwenden können.

Themen

- Beispiel: Fügen Sie einen Header hinzu, der auf einem Abfragezeichenfolgenparameter basiert
- Beispiel: Normalisieren Sie die Parameter der Abfragezeichenfolge, um die Cache-Trefferquote zu verbessern
- Beispiel: Leiten Sie nicht authentifizierte Benutzer auf eine Anmeldeseite weiter

Beispiel: Fügen Sie einen Header hinzu, der auf einem Abfragezeichenfolgenparameter basiert

Im folgenden Beispiel wird gezeigt, wie Sie das Schlüssel-Wert-Paar eines Abfragezeichenfolgeparameter abrufen und auf Grundlage dieser Werte einen Header hinzufügen können.

Node.js

```
'use strict';

const querystring = require('querystring');
exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;

    /* When a request contains a query string key-value pair but the origin server
     * expects the value in a header, you can use this Lambda function to
     * convert the key-value pair to a header. Here's what the function does:
```

```
    * 1. Parses the query string and gets the key-value pair.
    * 2. Adds a header to the request using the key-value pair that the function
 got in step 1.
    */

   /* Parse request querystring to get javascript object */
   const params = querystring.parse(request.querystring);

   /* Move auth param from querystring to headers */
   const headerName = 'Auth-Header';
   request.headers[headerName.toLowerCase()] = [{ key: headerName, value:
 params.auth }];
   delete params.auth;

   /* Update request querystring */
   request.querystring = querystring.stringify(params);

   callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs, urlencode

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    '''
    When a request contains a query string key-value pair but the origin server
    expects the value in a header, you can use this Lambda function to
    convert the key-value pair to a header. Here's what the function does:
        1. Parses the query string and gets the key-value pair.
        2. Adds a header to the request using the key-value pair that the function
 got in step 1.
    '''

    # Parse request querystring to get dictionary/json
    params = {k : v[0] for k, v in parse_qs(request['querystring']).items()}

    # Move auth param from querystring to headers
    headerName = 'Auth-Header'
    request['headers'][headerName.lower()] = [{'key': headerName, 'value':
 params['auth']}]
```

```
    del params['auth']

    # Update request querystring
    request['querystring'] = urlencode(params)

    return request
```

Beispiel: Normalisieren Sie die Parameter der Abfragezeichenfolge, um die Cache-Trefferquote zu verbessern

Das folgende Beispiel zeigt, wie Sie Ihre Cache-Trefferquote verbessern können, indem Sie die folgenden Änderungen an Abfragezeichenfolgen vornehmen, CloudFront bevor Anfragen an Ihren Ursprung weitergeleitet werden:

- Alphabetisches Anordnen von Schlüssel-Wert-Paaren nach dem Parametername.
- Ändern der Schreibung von Schlüssel-Wert-Paaren in Kleinschreibung.

Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern.

Node.js

```
'use strict';

const querystring = require('querystring');

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    /* When you configure a distribution to forward query strings to the origin and
     * to cache based on an allowlist of query string parameters, we recommend
     * the following to improve the cache-hit ratio:
     * - Always list parameters in the same order.
     * - Use the same case for parameter names and values.
     *
     * This function normalizes query strings so that parameter names and values
     * are lowercase and parameter names are in alphabetical order.
     *
     * For more information, see:
     * https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/
QueryStringParameters.html
     */
```

```
    console.log('Query String: ', request.querystring);

    /* Parse request query string to get javascript object */
    const params = querystring.parse(request.querystring.toLowerCase());
    const sortedParams = {};

    /* Sort param keys */
    Object.keys(params).sort().forEach(key => {
        sortedParams[key] = params[key];
    });

    /* Update request querystring with normalized  */
    request.querystring = querystring.stringify(sortedParams);

    callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs, urlencode

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    '''
    When you configure a distribution to forward query strings to the origin and
    to cache based on an allowlist of query string parameters, we recommend
    the following to improve the cache-hit ratio:
    Always list parameters in the same order.
    - Use the same case for parameter names and values.

    This function normalizes query strings so that parameter names and values
    are lowercase and parameter names are in alphabetical order.

    For more information, see:
    https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/
QueryStringParameters.html
    '''
    print("Query string: ", request["querystring"])

    # Parse request query string to get js object
    params = {k : v[0] for k, v in parse_qs(request['querystring'].lower()).items()}
```

```
    # Sort param keys
    sortedParams = sorted(params.items(), key=lambda x: x[0])

    # Update request querystring with normalized
    request['querystring'] = urlencode(sortedParams)

    return request
```

Beispiel: Leiten Sie nicht authentifizierte Benutzer auf eine Anmeldeseite weiter

Im folgenden Beispiel wird gezeigt, wie Benutzer zu einer Anmeldeseite umgeleitet werden, wenn sie ihre Anmeldeinformationen nicht eingegeben haben.

Node.js

```
'use strict';

function parseCookies(headers) {
    const parsedCookie = {};
    if (headers.cookie) {
        headers.cookie[0].value.split(';').forEach((cookie) => {
            if (cookie) {
                const parts = cookie.split('=');
                parsedCookie[parts[0].trim()] = parts[1].trim();
            }
        });
    }
    return parsedCookie;
}

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    /* Check for session-id in request cookie in viewer-request event,
     * if session-id is absent, redirect the user to sign in page with original
     * request sent as redirect_url in query params.
     */

    /* Check for session-id in cookie, if present then proceed with request */
    const parsedCookies = parseCookies(headers);
    if (parsedCookies && parsedCookies['session-id']) {
```

```
        callback(null, request);
        return;
    }

    /* URI encode the original request to be sent as redirect_url in query params */
    const encodedRedirectUrl = encodeURIComponent(`https://
${headers.host[0].value}${request.uri}?${request.querystring}`);
    const response = {
        status: '302',
        statusDescription: 'Found',
        headers: {
            location: [{
                key: 'Location',
                value: `https://www.example.com/signin?redirect_url=
${encodedRedirectUrl}`,
            }],
        },
    };
    callback(null, response);
};
```

Python

```
import urllib

def parseCookies(headers):
    parsedCookie = {}
    if headers.get('cookie'):
        for cookie in headers['cookie'][0]['value'].split(';'):
            if cookie:
                parts = cookie.split('=')
                parsedCookie[parts[0].strip()] = parts[1].strip()
    return parsedCookie

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    '''
    Check for session-id in request cookie in viewer-request event,
    if session-id is absent, redirect the user to sign in page with original
    request sent as redirect_url in query params.
    '''
```

```
    # Check for session-id in cookie, if present, then proceed with request
    parsedCookies = parseCookies(headers)

    if parsedCookies and parsedCookies['session-id']:
        return request

    # URI encode the original request to be sent as redirect_url in query params
    redirectUrl = "https://%s%s?%s" % (headers['host'][0]['value'], request['uri'],
request['querystring'])
    encodedRedirectUrl = urllib.parse.quote_plus(redirectUrl.encode('utf-8'))

    response = {
        'status': '302',
        'statusDescription': 'Found',
        'headers': {
            'location': [{
                'key': 'Location',
                'value': 'https://www.example.com/signin?redirect_url=%s' %
encodedRedirectUrl
            }]
        }
    }
    return response
```

Personalisieren von Inhalten nach Land oder Gerätetyp-Header – Beispiele

Die folgenden Beispiele zeigen, wie Sie Lambda @Edge verwenden können, um das Verhalten an den Standort oder den Typ des vom Betrachter verwendeten Geräts anzupassen.

Themen

- Beispiel: Leiten Sie Zuschaueranfragen an eine landesspezifische URL weiter
- Beispiel: Stellen Sie je nach Gerät verschiedene Versionen eines Objekts bereit

Beispiel: Leiten Sie Zuschaueranfragen an eine landesspezifische URL weiter

Im folgenden Beispiel wird gezeigt, wie eine HTTP-Umleitungsantwort mit einer länderspezifischen URL erzeugt und die Antwort an den Viewer zurückgegeben wird. Dies ist nützlich, wenn Sie länderspezifische Antworten bereitstellen möchten. Beispiel:

- Wenn Sie über länderspezifischen Subdomänen verfügen, z. B. us.example.com und tw.example.com, können Sie eine Umleitungsantwort erzeugen, wenn ein Viewer example.com anfordert.
- Wenn Sie Videos streamen, aber keine Rechte für das Streamen von Inhalten in einem bestimmten Land besitzen, können Sie Benutzer in diesem Land auf eine Seite umleiten, auf der erklärt wird, warum sie das Video nicht ansehen können.

Beachten Sie Folgendes:

- Sie müssen Ihre Verteilung so konfigurieren, dass die Zwischenspeicherung auf Grundlage des `CloudFront-Viewer-Country`-Headers erfolgt. Weitere Informationen finden Sie unter Basierend auf den ausgewählten Anforderungsheadern.
- CloudFront fügt den `CloudFront-Viewer-Country` Header nach dem Zuschaueranforderungsereignis hinzu. Wenn Sie dieses Beispiel verwenden möchten, müssen Sie einen Auslöser für das ursprüngliche Anfrageereignis erstellen.

Node.js

```
'use strict';

/* This is an origin request function */
exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    /*
     * Based on the value of the CloudFront-Viewer-Country header, generate an
     * HTTP status code 302 (Redirect) response, and return a country-specific
     * URL in the Location header.
     * NOTE: 1. You must configure your distribution to cache based on the
     *          CloudFront-Viewer-Country header. For more information, see
     *          https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-
headers
     *       2. CloudFront adds the CloudFront-Viewer-Country header after the
 viewer
     *          request event. To use this example, you must create a trigger for
 the
     *          origin request event.
     */
```

```
    let url = 'https://example.com/';
    if (headers['cloudfront-viewer-country']) {
        const countryCode = headers['cloudfront-viewer-country'][0].value;
        if (countryCode === 'TW') {
            url = 'https://tw.example.com/';
        } else if (countryCode === 'US') {
            url = 'https://us.example.com/';
        }
    }

    const response = {
        status: '302',
        statusDescription: 'Found',
        headers: {
            location: [{
                key: 'Location',
                value: url,
            }],
        },
    };
    callback(null, response);
};
```

Python

```
# This is an origin request function

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    '''
    Based on the value of the CloudFront-Viewer-Country header, generate an
    HTTP status code 302 (Redirect) response, and return a country-specific
    URL in the Location header.
    NOTE: 1. You must configure your distribution to cache based on the
            CloudFront-Viewer-Country header. For more information, see
            https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
          2. CloudFront adds the CloudFront-Viewer-Country header after the viewer
            request event. To use this example, you must create a trigger for the
            origin request event.
    '''
```

```
    url = 'https://example.com/'
    viewerCountry = headers.get('cloudfront-viewer-country')
    if viewerCountry:
        countryCode = viewerCountry[0]['value']
        if countryCode == 'TW':
            url = 'https://tw.example.com/'
        elif countryCode == 'US':
            url = 'https://us.example.com/'

    response = {
        'status': '302',
        'statusDescription': 'Found',
        'headers': {
            'location': [{
                'key': 'Location',
                'value': url
            }]
        }
    }

    return response
```

Beispiel: Stellen Sie je nach Gerät verschiedene Versionen eines Objekts bereit

Das folgende Beispiel zeigt, wie Sie für verschiedene Versionen eines Objekts auf Grundlage des Gerätetyps, den der Benutzer verwendet (z. B. ein mobiles Gerät oder ein Tablet), bereitstellen. Beachten Sie Folgendes:

- Sie müssen Ihre Verteilung so konfigurieren, dass die Zwischenspeicherung auf Grundlage der `CloudFront-Is-*-Viewer`-Header erfolgt. Weitere Informationen finden Sie unter Basierend auf den ausgewählten Anforderungsheadern.
- CloudFront fügt die `CloudFront-Is-*-Viewer` Header nach dem Viewer-Anforderungsereignis hinzu. Wenn Sie dieses Beispiel verwenden möchten, müssen Sie einen Auslöser für das ursprüngliche Anfrageereignis erstellen.

Node.js

```
'use strict';
```

```
/* This is an origin request function */
exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const headers = request.headers;

    /*
     * Serve different versions of an object based on the device type.
     * NOTE: 1. You must configure your distribution to cache based on the
     *          CloudFront-Is-*-Viewer headers. For more information, see
     *          the following documentation:
     *          https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-
headers
     *          https://docs.aws.amazon.com/console/cloudfront/cache-on-device-type
     *       2. CloudFront adds the CloudFront-Is-*-Viewer headers after the viewer
     *          request event. To use this example, you must create a trigger for
 the
     *          origin request event.
     */

    const desktopPath = '/desktop';
    const mobilePath = '/mobile';
    const tabletPath = '/tablet';
    const smarttvPath = '/smarttv';

    if (headers['cloudfront-is-desktop-viewer']
        && headers['cloudfront-is-desktop-viewer'][0].value === 'true') {
        request.uri = desktopPath + request.uri;
    } else if (headers['cloudfront-is-mobile-viewer']
               && headers['cloudfront-is-mobile-viewer'][0].value === 'true') {
        request.uri = mobilePath + request.uri;
    } else if (headers['cloudfront-is-tablet-viewer']
               && headers['cloudfront-is-tablet-viewer'][0].value === 'true') {
        request.uri = tabletPath + request.uri;
    } else if (headers['cloudfront-is-smarttv-viewer']
               && headers['cloudfront-is-smarttv-viewer'][0].value === 'true') {
        request.uri = smarttvPath + request.uri;
    }
    console.log(`Request uri set to "${request.uri}"`);

    callback(null, request);
};
```

Python

```
# This is an origin request function
def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    headers = request['headers']

    '''
    Serve different versions of an object based on the device type.
    NOTE: 1. You must configure your distribution to cache based on the
            CloudFront-Is-*-Viewer headers. For more information, see
            the following documentation:
            https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
            https://docs.aws.amazon.com/console/cloudfront/cache-on-device-type
          2. CloudFront adds the CloudFront-Is-*-Viewer headers after the viewer
            request event. To use this example, you must create a trigger for the
            origin request event.
    '''

    desktopPath = '/desktop';
    mobilePath = '/mobile';
    tabletPath = '/tablet';
    smarttvPath = '/smarttv';

    if 'cloudfront-is-desktop-viewer' in headers and headers['cloudfront-is-desktop-
viewer'][0]['value'] == 'true':
        request['uri'] = desktopPath + request['uri']
    elif 'cloudfront-is-mobile-viewer' in headers and headers['cloudfront-is-mobile-
viewer'][0]['value'] == 'true':
        request['uri'] = mobilePath + request['uri']
    elif 'cloudfront-is-tablet-viewer' in headers and headers['cloudfront-is-tablet-
viewer'][0]['value'] == 'true':
        request['uri'] = tabletPath + request['uri']
    elif 'cloudfront-is-smarttv-viewer' in headers and headers['cloudfront-is-
smarttv-viewer'][0]['value'] == 'true':
        request['uri'] = smarttvPath + request['uri']

    print("Request uri set to %s" % request['uri'])

    return request
```

Inhaltsbasierte dynamische Ursprungsauswahl –Beispiele

Die folgenden Beispiele zeigen, wie Sie Lambda @Edge verwenden können, um basierend auf den Informationen in der Anfrage zu verschiedenen Ursprüngen weiterzuleiten.

Themen

Beispiel: Verwenden Sie einen Origin-Anforderungs-Trigger, um von einem benutzerdefinierten Ursprung zu einem Amazon S3 S3-Ursprung zu wechseln

Diese Funktion demonstriert, wie mit Hilfe eines Ursprungsanforderungsauslösers von einem benutzerdefinierten Ursprung zu einem Amazon S3-Ursprung gewechselt werden kann, von dem der Inhalt auf Basis der Anforderungseigenschaften abgerufen wird.

Node.js

```
'use strict';

 const querystring = require('querystring');

 exports.handler = (event, context, callback) => {
     const request = event.Records[0].cf.request;

     /**
      * Reads query string to check if S3 origin should be used, and
      * if true, sets S3 origin properties.
      */

     const params = querystring.parse(request.querystring);
```

```
    if (params['useS3Origin']) {
        if (params['useS3Origin'] === 'true') {
            const s3DomainName = 'amzn-s3-demo-bucket.s3.amazonaws.com';

            /* Set S3 origin fields */
            request.origin = {
                s3: {
                    domainName: s3DomainName,
                    region: '',
                    authMethod: 'origin-access-identity',
                    path: '',
                    customHeaders: {}
                }
            };
            request.headers['host'] = [{ key: 'host', value: s3DomainName}];
        }
    }

    callback(null, request);
};
```

Python

```
from urllib.parse import parse_qs

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    '''
    Reads query string to check if S3 origin should be used, and
    if true, sets S3 origin properties
    '''
    params = {k: v[0] for k, v in parse_qs(request['querystring']).items()}
    if params.get('useS3Origin') == 'true':
        s3DomainName = 'amzn-s3-demo-bucket.s3.amazonaws.com'

        # Set S3 origin fields
        request['origin'] = {
            's3': {
                'domainName': s3DomainName,
                'region': '',
                'authMethod': 'origin-access-identity',
                'path': '',
```

```
                'customHeaders': {}
            }
        }
        request['headers']['host'] = [{'key': 'host', 'value': s3DomainName}]
    return request
```

Beispiel: Verwenden Sie einen Origin-Request-Trigger, um die Amazon S3 S3-Ursprungsregion zu ändern

Diese Funktion demonstriert, wie mit Hilfe eines Ursprungsanforderungsauslösers von einem Amazon S3-Ursprung gewechselt werden kann, von dem der Inhalt auf Basis der Anforderungseigenschaften abgerufen wird.

In diesem Beispiel verwenden wir den Wert des `CloudFront-Viewer-Country`-Headers zum Aktualisieren des S3-Bucket-Domänennamens auf einen Bucket in einer Region, die sich näher am Viewer befindet. Dies kann auf verschiedene Weise nützlich sein:

- Es werden Latenzen reduziert, wenn die angegebene Region näher am Land des Viewers liegt.
- Es sorgt für Datenhoheit, indem es sicherstellt, dass die Daten aus einem Ursprung stammen, der in dem Land liegt, aus dem die Anfrage stammt.

In diesem Beispiel gehen Sie wie folgt vor:

- Konfigurieren Sie Ihre Verteilung so, dass die Zwischenspeicherung auf Grundlage des `CloudFront-Viewer-Country`-Headers erfolgt. Weitere Informationen finden Sie unter Basierend auf den ausgewählten Anforderungsheadern.
- Erstellen Sie einen Trigger für diese Funktion im Origin-Request-Event. CloudFrontfügt den `CloudFront-Viewer-Country` Header nach dem Viewer-Anforderungsereignis hinzu. Um dieses Beispiel zu verwenden, müssen Sie also sicherstellen, dass die Funktion für eine ursprüngliche Anfrage ausgeführt wird.

> **Note**
>
> Der folgende Beispielcode verwendet dieselbe Origin-Zugriffsidentität (OAI) für alle S3-Buckets, die Sie für Ihren Ursprung verwenden. Weitere Informationen finden Sie unter Origin-Zugriffsidentität.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;

    /**
     * This blueprint demonstrates how an origin-request trigger can be used to
     * change the origin from which the content is fetched, based on request
 properties.
     * In this example, we use the value of the CloudFront-Viewer-Country header
     * to update the S3 bucket domain name to a bucket in a Region that is closer to
     * the viewer.
     *
     * This can be useful in several ways:
     *      1) Reduces latencies when the Region specified is nearer to the viewer's
     *         country.
     *      2) Provides data sovereignty by making sure that data is served from an
     *         origin that's in the same country that the request came from.
     *
     * NOTE: 1. You must configure your distribution to cache based on the
     *          CloudFront-Viewer-Country header. For more information, see
     *          https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-
headers
     *       2. CloudFront adds the CloudFront-Viewer-Country header after the
 viewer
     *          request event. To use this example, you must create a trigger for
 the
     *          origin request event.
     */

    const countryToRegion = {
        'DE': 'eu-central-1',
        'IE': 'eu-west-1',
        'GB': 'eu-west-2',
        'FR': 'eu-west-3',
        'JP': 'ap-northeast-1',
        'IN': 'ap-south-1'
    };

    if (request.headers['cloudfront-viewer-country']) {
        const countryCode = request.headers['cloudfront-viewer-country'][0].value;
        const region = countryToRegion[countryCode];
```

```
        /**
         * If the viewer's country is not in the list you specify, the request
         * goes to the default S3 bucket you've configured.
         */
        if (region) {
            /**
             * If you've set up OAI, the bucket policy in the destination bucket
             * should allow the OAI GetObject operation, as configured by default
             * for an S3 origin with OAI. Another requirement with OAI is to provide
             * the Region so it can be used for the SIGV4 signature. Otherwise, the
             * Region is not required.
             */
            request.origin.s3.region = region;
            const domainName = `amzn-s3-demo-bucket-in-${region}.s3.
${region}.amazonaws.com`;
            request.origin.s3.domainName = domainName;
            request.headers['host'] = [{ key: 'host', value: domainName }];
        }
    }

    callback(null, request);
};
```

Python

```
def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    '''
    This blueprint demonstrates how an origin-request trigger can be used to
    change the origin from which the content is fetched, based on request
 properties.
    In this example, we use the value of the CloudFront-Viewer-Country header
    to update the S3 bucket domain name to a bucket in a Region that is closer to
    the viewer.

    This can be useful in several ways:
        1) Reduces latencies when the Region specified is nearer to the viewer's
            country.
        2) Provides data sovereignty by making sure that data is served from an
            origin that's in the same country that the request came from.
```

```
    NOTE: 1. You must configure your distribution to cache based on the
            CloudFront-Viewer-Country header. For more information, see
            https://docs.aws.amazon.com/console/cloudfront/cache-on-selected-headers
          2. CloudFront adds the CloudFront-Viewer-Country header after the viewer
            request event. To use this example, you must create a trigger for the
            origin request event.
    '''

    countryToRegion = {
        'DE': 'eu-central-1',
        'IE': 'eu-west-1',
        'GB': 'eu-west-2',
        'FR': 'eu-west-3',
        'JP': 'ap-northeast-1',
        'IN': 'ap-south-1'
    }

    viewerCountry = request['headers'].get('cloudfront-viewer-country')
    if viewerCountry:
        countryCode = viewerCountry[0]['value']
        region = countryToRegion.get(countryCode)

        # If the viewer's country in not in the list you specify, the request
        # goes to the default S3 bucket you've configured
        if region:
            '''
            If you've set up OAI, the bucket policy in the destination bucket
            should allow the OAI GetObject operation, as configured by default
            for an S3 origin with OAI. Another requirement with OAI is to provide
            the Region so it can be used for the SIGV4 signature. Otherwise, the
            Region is not required.
            '''
            request['origin']['s3']['region'] = region
            domainName = 'amzn-s3-demo-bucket-in-{0}.s3.
{0}.amazonaws.com'.format(region)
            request['origin']['s3']['domainName'] = domainName
            request['headers']['host'] = [{'key': 'host', 'value': domainName}]

    return request
```

Beispiel: Verwenden Sie einen Origin-Anforderungs-Trigger, um von einem Amazon S3 S3-Ursprung zu einem benutzerdefinierten Ursprung zu wechseln

Diese Funktion demonstriert, wie mit Hilfe eines Ursprungsanforderungsauslösers von einem benutzerdefinierten Ursprung gewechselt werden kann, von dem der Inhalt auf Basis der Anforderungseigenschaften abgerufen wird.

Node.js

```
'use strict';

const querystring = require('querystring');

 exports.handler = (event, context, callback) => {
     const request = event.Records[0].cf.request;

     /**
      * Reads query string to check if custom origin should be used, and
      * if true, sets custom origin properties.
      */

     const params = querystring.parse(request.querystring);

     if (params['useCustomOrigin']) {
         if (params['useCustomOrigin'] === 'true') {

             /* Set custom origin fields*/
             request.origin = {
                 custom: {
                     domainName: 'www.example.com',
                     port: 443,
                     protocol: 'https',
                     path: '',
                     sslProtocols: ['TLSv1', 'TLSv1.1'],
                     readTimeout: 5,
                     keepaliveTimeout: 5,
                     customHeaders: {}
                 }
             };
             request.headers['host'] = [{ key: 'host', value: 'www.example.com'}];
         }
     }
    callback(null, request);
```

```
};
```

Python

```
from urllib.parse import parse_qs

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    # Reads query string to check if custom origin should be used, and
    # if true, sets custom origin properties

    params = {k: v[0] for k, v in parse_qs(request['querystring']).items()}

    if params.get('useCustomOrigin') == 'true':
            # Set custom origin fields
            request['origin'] = {
                'custom': {
                    'domainName': 'www.example.com',
                    'port': 443,
                    'protocol': 'https',
                    'path': '',
                    'sslProtocols': ['TLSv1', 'TLSv1.1'],
                    'readTimeout': 5,
                    'keepaliveTimeout': 5,
                    'customHeaders': {}
                }
            }
            request['headers']['host'] = [{'key': 'host', 'value':
 'www.example.com'}]

    return request
```

Beispiel: Verwenden Sie einen Origin-Anforderungsauslöser, um den Verkehr schrittweise von einem Amazon S3 S3-Bucket in einen anderen zu übertragen

Diese Funktion zeigt, wie Sie den Verkehr schrittweise und kontrolliert von einem Amazon S3 S3-Bucket in einen anderen übertragen können.

Node.js

```
'use strict';
```

```
    function getRandomInt(min, max) {
        /* Random number is inclusive of min and max*/
        return Math.floor(Math.random() * (max - min + 1)) + min;
 }

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;
    const BLUE_TRAFFIC_PERCENTAGE = 80;

    /**
      * This Lambda function demonstrates how to gradually transfer traffic from
      * one S3 bucket to another in a controlled way.
      * We define a variable BLUE_TRAFFIC_PERCENTAGE which can take values from
      * 1 to 100. If the generated randomNumber less than or equal to
 BLUE_TRAFFIC_PERCENTAGE, traffic
      * is re-directed to blue-bucket. If not, the default bucket that we've
 configured
      * is used.
      */

    const randomNumber = getRandomInt(1, 100);

if (randomNumber <= BLUE_TRAFFIC_PERCENTAGE) {
         const domainName = 'blue-bucket.s3.amazonaws.com';
         request.origin.s3.domainName = domainName;
         request.headers['host'] = [{ key: 'host', value: domainName}];
     }
    callback(null, request);
};
```

Python

```
import math
import random

def getRandomInt(min, max):
    # Random number is inclusive of min and max
    return math.floor(random.random() * (max - min + 1)) + min

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    BLUE_TRAFFIC_PERCENTAGE = 80
```

```
    '''
    This Lambda function demonstrates how to gradually transfer traffic from
    one S3 bucket to another in a controlled way.
    We define a variable BLUE_TRAFFIC_PERCENTAGE which can take values from
    1 to 100. If the generated randomNumber less than or equal to
BLUE_TRAFFIC_PERCENTAGE, traffic
    is re-directed to blue-bucket. If not, the default bucket that we've configured
    is used.
    '''

    randomNumber = getRandomInt(1, 100)

    if randomNumber <= BLUE_TRAFFIC_PERCENTAGE:
        domainName = 'blue-bucket.s3.amazonaws.com'
        request['origin']['s3']['domainName'] = domainName
        request['headers']['host'] = [{'key': 'host', 'value': domainName}]

    return request
```

Beispiel: Verwenden Sie einen Origin-Anforderungs-Trigger, um den Ursprungs-Domainnamen auf der Grundlage des Country-Headers zu ändern

Diese Funktion demonstriert, wie Sie den Ursprungsdomänennamen, basierend auf dem `CloudFront-Viewer-Country`-Header, ändern können, sodass der Inhalt von einem Ursprungsland aus bereitgestellt wird, das näher am Land des Viewers liegt.

Die Implementierung dieser Funktionalität für Ihre Verteilung bietet u. a. ggf. folgende Vorteile:

- Es werden Latenzen reduziert, wenn die angegebene Region näher am Land des Viewers liegt.
- Es wird für Datenhoheit gesorgt, indem sichergestellt wird, dass die Daten aus einem Ursprung stammen, der in dem Land liegt, aus dem die Anfrage stammt

Beachten Sie, dass Sie Ihre Verteilung für die Zwischenspeicherung auf Basis des `CloudFront-Viewer-Country`-Headers konfigurieren müssen, um diese Funktionalität zu aktivieren. Weitere Informationen finden Sie unter the section called “Basierend auf den ausgewählten Anforderungsheadern”.

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
     const request = event.Records[0].cf.request;

  if (request.headers['cloudfront-viewer-country']) {
        const countryCode = request.headers['cloudfront-viewer-country'][0].value;
        if (countryCode === 'GB' || countryCode === 'DE' || countryCode === 'IE' )
 {
            const domainName = 'eu.example.com';
            request.origin.custom.domainName = domainName;
            request.headers['host'] = [{key: 'host', value: domainName}];
        }
     }

    callback(null, request);
};
```

Python

```
def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    viewerCountry = request['headers'].get('cloudfront-viewer-country')
    if viewerCountry:
        countryCode = viewerCountry[0]['value']
        if countryCode == 'GB' or countryCode == 'DE' or countryCode == 'IE':
            domainName = 'eu.example.com'
            request['origin']['custom']['domainName'] = domainName
            request['headers']['host'] = [{'key': 'host', 'value': domainName}]
    return request
```

Fehlerstatus aktualisieren — Beispiele

Die folgenden Beispiele enthalten Anleitungen dazu, wie Sie Lambda @Edge verwenden können, um den Fehlerstatus zu ändern, der an Benutzer zurückgegeben wird.

Themen

Beispiel: Verwenden Sie einen Origin-Response-Trigger, um den Fehlerstatuscode auf 200 zu aktualisieren

Diese Funktion demonstriert, wie Sie den Antwortstatus auf 200 aktualisieren und statischen Body-Content für die Rückgabe an den Viewer generieren können:

- Die Funktion wird in einer Ursprungsantwort ausgelöst.
- Der Antwortstatus vom Ursprungs-Server ist ein Fehlerstatuscode (4xx oder 5xx).

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const response = event.Records[0].cf.response;

    /**
     * This function updates the response status to 200 and generates static
     * body content to return to the viewer in the following scenario:
     * 1. The function is triggered in an origin response
     * 2. The response status from the origin server is an error status code (4xx or
 5xx)
     */

    if (response.status >= 400 && response.status <= 599) {
        response.status = 200;
        response.statusDescription = 'OK';
        response.body = 'Body generation example';
    }

    callback(null, response);
};
```

Python

```
def lambda_handler(event, context):
    response = event['Records'][0]['cf']['response']

    '''
    This function updates the response status to 200 and generates static
    body content to return to the viewer in the following scenario:
    1. The function is triggered in an origin response
    2. The response status from the origin server is an error status code (4xx or
 5xx)
    '''

    if int(response['status']) >= 400 and int(response['status']) <= 599:
        response['status'] = 200
        response['statusDescription'] = 'OK'
        response['body'] = 'Body generation example'
    return response
```

Beispiel: Verwenden Sie einen Origin-Response-Trigger, um den Fehlerstatuscode auf 302 zu aktualisieren

Diese Funktion demonstriert, wie Sie den HTTP-Statuscode auf 302 aktualisieren können, um ihn auf einen anderen Pfad (Cache-Verhalten) umzuleiten, der einen anderen Ursprung hat. Beachten Sie Folgendes:

- Die Funktion wird in einer Ursprungsantwort ausgelöst.
- Der Antwortstatus vom Ursprungs-Server ist ein Fehlerstatuscode (4xx oder 5xx).

Node.js

```
'use strict';

exports.handler = (event, context, callback) => {
    const response = event.Records[0].cf.response;
    const request = event.Records[0].cf.request;

    /**
     * This function updates the HTTP status code in the response to 302, to
 redirect to another
```

```
    * path (cache behavior) that has a different origin configured. Note the
 following:
    * 1. The function is triggered in an origin response
    * 2. The response status from the origin server is an error status code (4xx or
 5xx)
    */

   if (response.status >= 400 && response.status <= 599) {
       const redirect_path = `/plan-b/path?${request.querystring}`;

       response.status = 302;
       response.statusDescription = 'Found';

       /* Drop the body, as it is not required for redirects */
       response.body = '';
       response.headers['location'] = [{ key: 'Location', value: redirect_path }];
   }

   callback(null, response);
};
```

Python

```
def lambda_handler(event, context):
    response = event['Records'][0]['cf']['response']
    request = event['Records'][0]['cf']['request']

    '''
    This function updates the HTTP status code in the response to 302, to redirect
 to another
    path (cache behavior) that has a different origin configured. Note the
 following:
    1. The function is triggered in an origin response
    2. The response status from the origin server is an error status code (4xx or
 5xx)
    '''

    if int(response['status']) >= 400 and int(response['status']) <= 599:
        redirect_path = '/plan-b/path?%s' % request['querystring']

        response['status'] = 302
        response['statusDescription'] = 'Found'
```

```
        # Drop the body as it is not required for redirects
        response['body'] = ''
        response['headers']['location'] = [{'key': 'Location', 'value':
redirect_path}]

    return response
```

Greifen Sie auf den Anfragetext zu — Beispiele

Die folgenden Beispiele zeigen, wie Sie Lambda @Edge verwenden können, um mit POST-Anfragen zu arbeiten.

> **Note**
>
> Um diese Beispiele zu verwenden, müssen Sie die Option Textkörper einschließen in der Lambda-Funktionszuordnung der Verteilung aktivieren. Sie ist standardmäßig nicht aktiviert.
>
> - Um diese Einstellung in der CloudFront Konsole zu aktivieren, aktivieren Sie das Kontrollkästchen Body in the Lambda Function Association einbeziehen.
> - Um diese Einstellung in der CloudFront API oder mit zu aktivieren AWS CloudFormation, setzen Sie das `IncludeBody` Feld auf `true` in`LambdaFunctionAssociation`.

Themen

Beispiel: Verwenden Sie einen Anforderungsauslöser, um ein HTML-Formular zu lesen

Diese Funktion zeigt, wie Sie den Body einer POST-Anforderung verarbeiten können, die durch ein HTML-Formular (Webformular) erzeugt wird (z. B. ein "Kontaktformular"). Beispielsweise könnten Sie ein HTML-Formular wie das Folgende nutzen:

```
<html>
  <form action="https://example.com" method="post">
    Param 1: <input type="text" name="name1"><br>
    Param 2: <input type="text" name="name2"><br>
    input type="submit" value="Submit">
  </form>
```

```
</html>
```

Für die folgende Beispielfunktion muss die Funktion in einer CloudFront -Betrachteranforderung oder -Ursprungsanforderung ausgelöst werden.

Node.js

```
'use strict';

const querystring = require('querystring');

/**
 * This function demonstrates how you can read the body of a POST request
 * generated by an HTML form (web form). The function is triggered in a
 * CloudFront viewer request or origin request event type.
 */

exports.handler = (event, context, callback) => {
    const request = event.Records[0].cf.request;

    if (request.method === 'POST') {
        /* HTTP body is always passed as base64-encoded string. Decode it. */
        const body = Buffer.from(request.body.data, 'base64').toString();

        /* HTML forms send the data in query string format. Parse it. */
        const params = querystring.parse(body);

        /* For demonstration purposes, we only log the form fields here.
         * You can put your custom logic here. For example, you can store the
         * fields in a database, such as Amazon DynamoDB, and generate a response
         * right from your Lambda@Edge function.
         */
        for (let param in params) {
            console.log(`For "${param}" user submitted "${params[param]}".\n`);
        }
    }
    return callback(null, request);
};
```

Python

```
import base64
from urllib.parse import parse_qs
```

```
'''
Say there is a POST request body generated by an HTML such as:

<html>
<form action="https://example.com" method="post">
    Param 1: <input type="text" name="name1"><br>
    Param 2: <input type="text" name="name2"><br>
    input type="submit" value="Submit">
</form>
</html>

'''

'''
This function demonstrates how you can read the body of a POST request
generated by an HTML form (web form). The function is triggered in a
CloudFront viewer request or origin request event type.
'''

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']

    if request['method'] == 'POST':
        # HTTP body is always passed as base64-encoded string. Decode it
        body = base64.b64decode(request['body']['data'])

        # HTML forms send the data in query string format. Parse it
        params = {k: v[0] for k, v in parse_qs(body).items()}

        '''
        For demonstration purposes, we only log the form fields here.
        You can put your custom logic here. For example, you can store the
        fields in a database, such as Amazon DynamoDB, and generate a response
        right from your Lambda@Edge function.
        '''
        for key, value in params.items():
            print("For %s use submitted %s" % (key, value))

    return request
```

Beispiel: Verwenden Sie einen Anforderungstrigger, um ein HTML-Formular zu ändern

Diese Funktion zeigt, wie Sie den Body einer POST-Anforderung bearbeiten können, die durch ein HTML-Formular (Webformular) erzeugt wird. Die Funktion wird in einer CloudFront Viewer-Anfrage oder einer Origin-Anfrage ausgelöst.

Node.js

```
'use strict';

const querystring = require('querystring');

exports.handler = (event, context, callback) => {
    var request = event.Records[0].cf.request;
    if (request.method === 'POST') {
        /* Request body is being replaced. To do this, update the following
        /* three fields:
         *    1) body.action to 'replace'
         *    2) body.encoding to the encoding of the new data.
         *
         *       Set to one of the following values:
         *
         *           text - denotes that the generated body is in text format.
         *               Lambda@Edge will propagate this as is.
         *           base64 - denotes that the generated body is base64 encoded.
         *               Lambda@Edge will base64 decode the data before sending
         *               it to the origin.
         *    3) body.data to the new body.
         */
        request.body.action = 'replace';
        request.body.encoding = 'text';
        request.body.data = getUpdatedBody(request);
    }
    callback(null, request);
};

function getUpdatedBody(request) {
    /* HTTP body is always passed as base64-encoded string. Decode it. */
    const body = Buffer.from(request.body.data, 'base64').toString();

    /* HTML forms send data in query string format. Parse it. */
    const params = querystring.parse(body);
```

```
    /* For demonstration purposes, we're adding one more param.
     *
     * You can put your custom logic here. For example, you can truncate long
     * bodies from malicious requests.
     */
    params['new-param-name'] = 'new-param-value';
    return querystring.stringify(params);
}
```

Python

```
import base64
from urllib.parse import parse_qs, urlencode

def lambda_handler(event, context):
    request = event['Records'][0]['cf']['request']
    if request['method'] == 'POST':
        '''
        Request body is being replaced. To do this, update the following
        three fields:
            1) body.action to 'replace'
            2) body.encoding to the encoding of the new data.

            Set to one of the following values:

                text - denotes that the generated body is in text format.
                    Lambda@Edge will propagate this as is.
                base64 - denotes that the generated body is base64 encoded.
                    Lambda@Edge will base64 decode the data before sending
                    it to the origin.
            3) body.data to the new body.
        '''
        request['body']['action'] = 'replace'
        request['body']['encoding'] = 'text'
        request['body']['data'] = getUpdatedBody(request)
    return request

def getUpdatedBody(request):
    # HTTP body is always passed as base64-encoded string. Decode it
    body = base64.b64decode(request['body']['data'])

    # HTML forms send data in query string format. Parse it
    params = {k: v[0] for k, v in parse_qs(body).items()}
```

```
    # For demonstration purposes, we're adding one more param

    # You can put your custom logic here. For example, you can truncate long
    # bodies from malicious requests
    params['new-param-name'] = 'new-param-value'
    return urlencode(params)
```

Einschränkungen für Edge-Funktionen

In den folgenden Themen werden die Einschränkungen beschrieben, die für CloudFront Functions und Lambda @Edge gelten. Einige Einschränkungen gelten für alle Edge-Funktionen, während andere nur für CloudFront Functions oder Lambda @Edge gelten.

Jedes Thema enthält detaillierte Informationen zu den Einschränkungen und Einschränkungen, die Sie bei der Entwicklung und Bereitstellung von Edge-Funktionen berücksichtigen sollten. CloudFront

Wenn Sie sich mit diesen Einschränkungen vertraut machen, können Sie sicherstellen, dass Ihre Edge-Funktionen erwartungsgemäß funktionieren und den unterstützten Funktionen entsprechen.

Themen

Weitere Informationen zu Kontingenten (auch zuvor Limits genannt) finden Sie unter Kontingente für CloudFront Funktionen und Kontingente für Lambda@Edge.

Einschränkungen für alle Edge-Funktionen

Die folgenden Einschränkungen gelten für alle Edge-Funktionen, sowohl für CloudFront Functions als auch für Lambda @Edge.

Themen

- HTTP-Header
- Abfragezeichenfolgen
- URI
- Kodierung von URI, Abfragezeichenfolge und Headern
- Microsoft Smooth Streaming
- Tagging

AWS-Konto -Eigentümerschaft

Um eine Edge-Funktion einer CloudFront Distribution zuzuordnen, müssen die Funktion und die Distribution derselben AWS-Konto gehören.

CloudFront Funktionen mit Lambda @Edge kombinieren

Für jedes Cache-Verhalten gelten die folgenden Einschränkungen:

- Jeder Ereignistyp (Viewer-Anforderung, Ursprungsanforderung, Ursprungsantwort und Viewer-Antwort) kann nur eine Edge-Funktionszuordnung aufweisen.
- Sie können CloudFront Functions und Lambda @Edge nicht in Viewer-Ereignissen (Viewer-Anfrage und Viewer-Antwort) kombinieren.

Alle anderen Kombinationen von Edge-Funktionen sind erlaubt. In der folgenden Tabelle werden die erlaubten Kombinationen erläutert.

		CloudFront Funktionen	
		Viewer-Anforderung	Viewer-Antwort
Lambda@Edge	Viewer-Anforderung	Nicht zulässig	Nicht zulässig
	Ursprungsanfrage	Zulässig	Zulässig
	Ursprungsantwort	Zulässig	Zulässig
	Viewer-Antwort	Nicht zulässig	Nicht zulässig

HTTP-Statuscodes

CloudFront ruft keine Edge-Funktionen für Zuschauer-Antwortereignisse auf, wenn der Ursprung den HTTP-Statuscode 400 oder höher zurückgibt.

Lambda@Edge-Funktionen für Ursprungsantwortereignisse werden für alle Ursprungsantworten aufgerufen, auch wenn der Ursprung einen HTTP-Statuscode 400 oder höher zurückgibt. Weitere Informationen finden Sie unter Aktualisieren Sie die HTTP-Antworten in den ursprünglichen Antwortauslösern.

HTTP-Header

Bestimmte HTTP-Header sind nicht zulässig, was bedeutet, dass sie nicht für Edge-Funktionen zugänglich sind und die Funktionen sie nicht hinzufügen können. Andere Header sind schreibgeschützt, was bedeutet, dass Funktionen sie lesen, aber nicht hinzufügen, ändern oder löschen können.

Themen

- Unzulässige Header
- Schreibgeschützte Header

Unzulässige Header

Die folgenden HTTP-Header sind nicht für Edge-Funktionen verfügbar und die Funktionen können sie nicht hinzufügen. Wenn Ihre Funktion einen dieser Header hinzufügt, schlägt sie bei der CloudFront Überprüfung fehl und CloudFront gibt den HTTP-Statuscode 502 (Bad Gateway) an den Viewer zurück.

- `Connection`
- `Expect`
- `Keep-Alive`
- `Proxy-Authenticate`
- `Proxy-Authorization`
- `Proxy-Connection`
- `Trailer`
- `Upgrade`
- `X-Accel-Buffering`

- `X-Accel-Charset`
- `X-Accel-Limit-Rate`
- `X-Accel-Redirect`
- `X-Amz-Cf-*`
- `X-Amzn-Auth`
- `X-Amzn-Cf-Billing`
- `X-Amzn-Cf-Id`
- `X-Amzn-Cf-Xff`
- `X-Amzn-Errortype`
- `X-Amzn-Fle-Profile`
- `X-Amzn-Header-Count`
- `X-Amzn-Header-Order`
- `X-Amzn-Lambda-Integration-Tag`
- `X-Amzn-RequestId`
- `X-Cache`
- `X-Edge-*`
- `X-Forwarded-Proto`
- `X-Real-IP`

Schreibgeschützte Header

Die folgenden Header sind schreibgeschützt. Ihre Funktion kann sie lesen und als Eingabe für die Funktionslogik verwenden, doch sie kann die Werte nicht ändern. Wenn Ihre Funktion einen schreibgeschützten Header hinzufügt oder bearbeitet, schlägt die CloudFront Überprüfung der Anforderung fehl und CloudFront gibt den HTTP-Statuscode 502 (Bad Gateway) an den Viewer zurück.

Schreibgeschützte Header bei Viewer-Anforderungsereignissen

Die folgenden Header sind bei Viewer-Anforderungsereignissen schreibgeschützt.

- `Content-Length`
- `Host`
- `Transfer-Encoding`

- `Via`

Schreibgeschützte Header in Ursprungsanforderungsereignissen (nur Lambda@Edge)

Die folgenden Header sind in Ursprungsanforderungsereignissen schreibgeschützt, die nur in Lambda@Edge vorhanden sind.

- `Accept-Encoding`
- `Content-Length`
- `If-Modified-Since`
- `If-None-Match`
- `If-Range`
- `If-Unmodified-Since`
- `Transfer-Encoding`
- `Via`

Schreibgeschützte Header in Ursprungsantwortereignissen (nur Lambda@Edge)

Die folgenden Header sind in Ursprungsantwortereignissen schreibgeschützt, die nur in Lambda@Edge vorhanden sind.

- `Transfer-Encoding`
- `Via`

Schreibgeschützte Header bei Viewer-Antwortereignissen

Die folgenden Header sind in Viewer-Antwortereignissen sowohl für CloudFront Functions als auch für Lambda @Edge schreibgeschützt.

- `Warning`
- `Via`

Die folgenden Header sind bei Viewer-Antwortereignissen für Lambda@Edge schreibgeschützt.

- `Content-Length`
- `Content-Encoding`

- `Transfer-Encoding`

Abfragezeichenfolgen

Die folgenden Einschränkungen gelten für Funktionen, die eine Abfragezeichenfolge in einem Anforderungs-URI lesen, aktualisieren oder erstellen.

- (Nur Lambda@Edge) Um auf die Abfragezeichenfolge in einer Ursprungsanforderung oder einer Ursprungsantwortfunktion zuzugreifen, muss die Cache-Richtlinie oder die Ursprungsanforderungsrichtlinie auf Alle auf Abfragezeichenfolgen gesetzt werden.
- Eine Funktion kann eine Abfragezeichenfolge für Viewer-Anforderungs- und Ursprungsanforderungsereignisse erstellen oder aktualisieren (Ursprungsanforderungsereignisse existieren nur in Lambda@Edge).
- Eine Funktion kann für Ursprungsantwort- und Viewer-Antwortereignisse eine Abfragezeichenfolge lesen, aber nicht erstellen oder aktualisieren (Ursprungsantwortereignisse existieren nur in Lambda@Edge).
- Wenn eine Funktion eine Abfragezeichenfolge erstellt oder aktualisiert, gelten folgende Einschränkungen:
  - Die Abfragezeichenfolge darf keine Leerzeichen, Steuerzeichen oder die Fragment-ID (#) enthalten.
  - Die Gesamtgröße der URI und der Abfragezeichenfolge darf nicht mehr als 8.192 Zeichen umfassen.
  - Wir empfehlen die Verwendung der Prozentkodierung für die URI und die Abfragezeichenfolge. Weitere Informationen finden Sie unter Kodierung von URI, Abfragezeichenfolge und Headern.

URI

Wenn eine Funktion Änderungen an dem URI für eine Anforderung durchführt, ändert dies weder das Cache-Verhalten für die Anforderung noch den Ursprung, an den Anforderung weitergeleitet wird.

Die Gesamtgröße der URI und der Abfragezeichenfolge darf nicht mehr als 8.192 Zeichen umfassen.

Kodierung von URI, Abfragezeichenfolge und Headern

Die Werte für den URI, die Abfragezeichenfolge und die Header, die an Edge-Funktionen übergeben werden, sind UTF-8-kodiert. Ihre Funktion sollte die UTF-8-Kodierung für die von ihr

zurückgegebenen URI-, Abfragezeichenfolgen- und Header-Werte verwenden. Die Prozentkodierung ist kompatibel mit der UTF-8-Kodierung kompatibel.

In der folgenden Liste wird erklärt, wie CloudFront mit der Kodierung für den URI, die Abfragezeichenfolge und die Header umgegangen wird:

- Wenn die Werte in der Anfrage UTF-8-kodiert sind, werden die Werte an Ihre CloudFront Funktion weitergeleitet, ohne sie zu ändern.
- Wenn die Werte in der Anfrage ISO-8859-1-codiert sind, werden die Werte in die UTF-8-Kodierung CloudFront konvertiert, bevor sie an Ihre Funktion weitergeleitet werden.
- Wenn Werte in der Anfrage mit einer anderen Zeichenkodierung codiert sind, wird CloudFront davon ausgegangen, dass sie ISO-8859-1-kodiert sind, und versucht, sie von ISO-8859-1 nach UTF-8 zu konvertieren.

> **Important**
>
> Die konvertierten Zeichen sind möglicherweise eine falsche Interpretation der Werte in der ursprünglichen Anfrage. Dies kann dazu führen, dass Ihre Funktion oder Ihr Ursprung ein unbeabsichtigtes Ergebnis produzieren.

Die Werte für den URI, die Abfragezeichenfolge und die Header, die an Ihren Ursprung weitergeleitet werden, hängen davon ab, ob eine Funktion die CloudFront Werte ändert:

- Wenn eine Funktion den URI, die Abfragezeichenfolge oder den Header nicht ändert, CloudFront leitet sie die Werte, die sie in der Anfrage erhalten hat, an Ihren Ursprung weiter.
- Wenn eine Funktion den URI, die Abfragezeichenfolge oder den Header ändert, CloudFront leitet sie die UTF-8-kodierten Werte weiter.

Microsoft Smooth Streaming

Sie können Edge-Funktionen nicht mit einer CloudFront Distribution verwenden, die Sie zum Streamen von Mediendateien verwenden, die Sie in das Microsoft Smooth Streaming-Format transkodiert haben.

Tagging

Sie können Edge-Funktionen keine Tags hinzufügen. Weitere Informationen zum Taggen in finden Sie CloudFront unterKennzeichnen Sie eine Distribution.

Einschränkungen von Funktionen CloudFront

Die folgenden Einschränkungen gelten nur für CloudFront Funktionen.

Inhalt

Hinweise zu Kontingenten (früher als Beschränkungen bezeichnet) finden Sie unterKontingente für CloudFront Funktionen.

Logs (Protokolle)

Funktionsprotokolle in CloudFront Functions werden auf 10 KB gekürzt.

Anforderungstext

CloudFront Funktionen können nicht auf den Hauptteil der HTTP-Anfrage zugreifen.

Verwenden temporärer Anmeldeinformationen mit der CloudFront KeyValueStore API

Sie können AWS Security Token Service (AWS STS) verwenden, um temporäre Sicherheitsanmeldeinformationen (auch als Sitzungstoken bezeichnet) zu generieren. Mit Sitzungstoken können Sie vorübergehend eine AWS Identity and Access Management (IAM-) Rolle übernehmen, sodass Sie darauf zugreifen AWS-Services können.

Um die CloudFront KeyValueStore API aufzurufen, verwenden Sie einen regionalen Endpunkt, AWS STS um ein Sitzungstoken der Version 2 zurückzugeben. Wenn Sie den globalen Endpunkt for AWS STS (`sts.amazonaws.com`) verwenden, AWS STS wird ein Sitzungstoken der Version 1 generiert, das von Signature Version 4A (Sigv4A) nicht unterstützt wird. Infolgedessen erhalten Sie einen Authentifizierungsfehler.

Um die CloudFront KeyValueStore API aufzurufen, können Sie die folgenden Optionen verwenden:

AWS CLI und AWS SDKs

Sie können das AWS CLI oder ein AWS SDK so konfigurieren, dass regionale AWS STS Endpunkte verwendet werden. Weitere Informationen finden Sie unter AWS STS Regionalisierte Endpunkte im AWS SDK- und Tools-Referenzhandbuch.

Weitere Informationen zu verfügbaren AWS STS Endpunkten finden Sie im IAM-Benutzerhandbuch unter Regionen und Endpunkte.

SAML

Sie können SAML für die Verwendung regionaler Endpunkte konfigurieren. AWS STS Weitere Informationen finden Sie im Blogbeitrag So verwenden Sie regionale SAML-Endpunkte für Failover.

`SetSecurityTokenServicePreferences`-API

Anstatt einen regionalen AWS STS Endpunkt zu verwenden, können Sie den globalen Endpunkt so konfigurieren, AWS STS dass Sitzungstoken der Version 2 zurückgegeben werden. Verwenden Sie dazu den SetSecurityTokenServicePreferencesAPI-Vorgang zur Konfiguration Ihres AWS-Konto.

Example Beispiel: IAM-CLI-Befehl

```
aws iam set-security-token-service-preferences --global-endpoint-token-version
 v2Token
```

> **Tip**
>
> Wir empfehlen, anstelle dieser Option die AWS STS regionalen Endpunkte zu verwenden. Regionale Endpunkte bieten höhere Verfügbarkeit und Failover-Szenarien.

Benutzerdefinierter Identitätsanbieter

Wenn Sie einen benutzerdefinierten Identitätsanbieter verwenden, der den Verbund übernimmt und die Rolle übernimmt, verwenden Sie eine der vorherigen Optionen für das übergeordnete Identitätsanbietersystem, das für die Generierung des Sitzungstokens verantwortlich ist.

Laufzeit

Die Runtime-Umgebung von CloudFront Functions unterstützt keine dynamische Codeauswertung und schränkt den Zugriff auf das Netzwerk, das Dateisystem, Umgebungsvariablen und Timer ein. Weitere Informationen finden Sie unter Eingeschränkte Funktionen.

> **Note**
>
> Um diese CloudFront Funktion verwenden zu können CloudFront KeyValueStore, muss sie JavaScript Runtime 2.0 verwenden.

Computing-Auslastung

CloudFront Funktionen haben eine Obergrenze für die Ausführungszeit, gemessen an der Rechenauslastung. Die Computing-Auslastung ist eine Zahl zwischen 0 und 100, mit der die Zeit angegeben wird, die die Ausführung der Funktion als Prozentsatz der maximal zulässigen Zeit in Anspruch genommen hat. Zum Beispiel bedeutet eine Rechenauslastung von 35, dass die Funktion in 35 % der maximal zulässigen Zeit abgeschlossen wurde.

Wenn Sie eine Funktion testen können Sie den Wert der Computing-Auslastung in der Ausgabe des Testereignisses sehen. Bei Produktionsfunktionen können Sie die Metrik zur Rechenauslastung auf der Seite Überwachung in der CloudFront Konsole oder in anzeigen CloudWatch.

Einschränkungen für Lambda@Edge

Die folgenden Einschränkungen gelten nur für Lambda@Edge.

Inhalt

- Einschränkungen für Anforderungstext mit der Option „Text einschließen“
- Antwort-Timeout und Keep-Alive-Timeout (nur benutzerdefinierte Ursprünge)

Hinweise zu Kontingenten finden Sie unter Kontingente für Lambda@Edge.

DNS-Auflösung

CloudFront führt eine DNS-Auflösung für den ursprünglichen Domainnamen durch, bevor es die Lambda @Edge -Funktion Ihrer ursprünglichen Anfrage ausführt. Wenn beim DNS-Dienst für Ihre Domain Probleme auftreten und der Domainname nicht aufgelöst werden CloudFront kann, um die IP-Adresse zu erhalten, wird Ihre Lambda @Edge -Funktion nicht aufgerufen. CloudFrontgibt einen HTTP 502-Statuscode (Bad Gateway) an den Client zurück. Weitere Informationen finden Sie unter DNS-Fehler (`NonS3OriginDnsError`).

Wenn Ihre Funktionslogik den ursprünglichen Domainnamen ändert, CloudFront wird nach Abschluss der Ausführung der Funktion eine weitere DNS-Auflösung für den aktualisierten Domainnamen durchgeführt.

Weitere Informationen zur Verwaltung von DNS-Failover finden Sie unter Konfiguration des DNS-Failovers im Amazon Route 53-Entwicklerhandbuch.

HTTP-Statuscodes

Lambda @Edge -Funktionen für Zuschauer-Antwortereignisse können den HTTP-Statuscode der Antwort nicht ändern, unabhängig davon, ob die Antwort vom Ursprung oder vom CloudFront Cache stammt.

Version der Lambda-Funktion

Sie müssen eine nummerierte Version der Lambda-Funktion verwenden, nicht jedoch `$LATEST` oder Aliase.

Lambda-Region

Die Lambda-Funktion muss sich in der Region USA Ost (Nord-Virginia) befinden.

Lambda-Rollenberechtigungen

Um Auslöser hinzufügen zu können, muss die Ihrer Lambda-Funktion zugewiesene IAM-Ausführungsrolle von den Haupt-Services `lambda.amazonaws.com` und

`edgelambda.amazonaws.com` eingenommen werden können. Weitere Informationen finden Sie unter Richten Sie IAM-Berechtigungen und -Rollen für Lambda @Edge ein.

Lambda-Funktionen

Die folgenden Lambda-Funktionen werden von Lambda@Edge nicht unterstützt:

- Andere Lambda-Laufzeitmanagement-Konfigurationen als Auto (Standard)
- Konfiguration Ihrer Lambda-Funktion für den Zugriff auf Ressourcen in Ihrer VPC
- Warteschlangen für tote Buchstaben der Lambda-Funktion
- Lambda-Umgebungsvariablen (mit Ausnahme von reservierten Umgebungsvariablen, die automatisch unterstützt werden)
- Lambda-Funktionen mit der Verwaltung von AWS Lambda Abhängigkeiten mit Ebenen
- Verwenden von AWS X-Ray
- Parallelität per Lambda

> **Note**
>
> Lambda @Edge -Funktionen nutzen dieselben regionalen Parallelitätsfunktionen wie alle Lambda-Funktionen. Weitere Informationen finden Sie unter Kontingente für Lambda@Edge.

- Erstellen Sie eine Lambda-Funktion mit einem Container-Image
- Lambda-Funktionen, die die arm64-Architektur verwenden
- Lambda-Funktionen mit mehr als 512 MB flüchtigem Speicher
- Verwenden Sie einen vom Kunden verwalteten Schlüssel zur Verschlüsselung Ihrer ZIP-Bereitstellungspakete

Unterstützte Laufzeiten

Lambda @Edge unterstützt die neuesten Versionen von Node.js und Python-Laufzeiten. Eine Liste der unterstützten Versionen und ihrer future Verfallsdaten finden Sie unter Unterstützte Laufzeiten im AWS Lambda Entwicklerhandbuch.

Tip

- Es hat sich bewährt, die neuesten Versionen der bereitgestellten Laufzeiten für Leistungsverbesserungen und neue Funktionen zu verwenden.
- Mit veralteten Versionen von Node.js können Sie keine Funktionen erstellen oder aktualisieren. Sie können bestehende Funktionen mit diesen Versionen nur Distributionen zuordnen. CloudFront Funktionen mit diesen Versionen, die Distributionen zugeordnet sind, werden weiterhin ausgeführt. Wir empfehlen Ihnen jedoch, Ihre Funktion auf neuere Versionen von Node.js umzustellen. Weitere Informationen finden Sie unter Runtime Deprecation Policy im AWS Lambda Developer Guide und im Release-Zeitplan für Node.js unter. GitHub

CloudFrontKopfzeilen

Lambda @Edge -Funktionen können jeden der unter aufgeführten CloudFront Header lesen, bearbeiten, entfernen oder hinzufügen. CloudFront Anforderungsheader hinzufügen

Hinweise

- Wenn Sie diese Header hinzufügen CloudFront möchten, müssen Sie sie so konfigurieren CloudFront , dass sie mithilfe einer Cache-Richtlinie oder einer Origin-Request-Richtlinie hinzugefügt werden.
- CloudFront fügt die Header nach dem Viewer-Anforderungsereignis hinzu, was bedeutet, dass die Header für Lambda @Edge -Funktionen in einer Viewer-Anfrage nicht verfügbar sind. Die Header sind nur für Lambda @Edge -Funktionen in einer Ursprungsanfrage und einer Ursprungsantwort verfügbar.
- Wenn die Viewer-Anfrage Header mit diesen Namen enthält und Sie das Hinzufügen dieser Header mithilfe einer Cache-Richtlinie oder einer Origin-Anforderungsrichtlinie konfiguriert CloudFront haben, werden die Header-Werte, die in der Viewer-Anfrage enthalten waren, CloudFront überschrieben. Funktionen, die dem Betrachter zugewandt sind, sehen den Header-Wert aus der Viewer-Anfrage, wohingegen Funktionen, die an den Ursprung gerichtet sind, den Header-Wert sehen, der hinzugefügt wurde. CloudFront

- Wenn eine Viewer-Anforderungsfunktion den `CloudFront-Viewer-Country` Header hinzufügt, schlägt sie bei der Überprüfung fehl und CloudFront gibt den HTTP-Statuscode 502 (Bad Gateway) an den Viewer zurück.

Einschränkungen für Anforderungstext mit der Option „Text einschließen“

Wenn Sie die Option Text einbeziehen wählen, um den Anforderungstext für Ihre Lambda @Edge -Funktion verfügbar zu machen, gelten die folgenden Informationen und Größenbeschränkungen für die Teile des Hauptteils, die offengelegt oder ersetzt werden.

- CloudFront immer base64 kodiert den Hauptteil der Anfrage, bevor er Lambda @Edge zur Verfügung gestellt wird.
- Wenn der Anforderungstext groß ist, CloudFront kürzt er ihn, bevor er Lambda @Edge zur Verfügung gestellt wird, wie folgt:
  - Bei Viewer-Anforderungsereignissen wird der Fließtext bei 40 KB abgeschnitten.
  - Bei Ursprungsanforderungsereignissen wird der Fließtext bei 1 MB abgeschnitten.
- Wenn Sie schreibgeschützt auf den Anfragetext zugreifen, wird der vollständige ursprüngliche CloudFront Anfragetext an den Ursprung gesendet.
- Wenn Ihre Lambda @Edge -Funktion den Hauptteil der Anfrage ersetzt, gelten die folgenden Größenbeschränkungen für den Hauptteil, den die Funktion zurückgibt:
  - Wenn die Lambda@Edge-Funktion den Textkörper als Klartext zurückgibt:
    - Für Viewer-Anforderungsereignisse liegt die Obergrenze für den Hauptteil bei 40 KB.
    - Für Ereignisse mit Ursprungsanfragen liegt die Obergrenze für den Hauptteil bei 1 MB.
  - Wenn die Lambda@Edge Funktion den Textkörper als base64-codierten Text zurückgibt:
    - Bei Viewer-Anforderungsereignissen liegt die Obergrenze für den Hauptteil bei 53,2 KB.
    - Für Ereignisse mit Ursprungsanfragen liegt die Obergrenze für den Hauptteil bei 1,33 MB.

> **Note**
>
> Wenn Ihre Lambda @Edge -Funktion einen Text zurückgibt, der diese Grenzwerte überschreitet, schlägt Ihre Anfrage mit dem HTTP 502-Statuscode (Fehler bei der Lambda-Validierung) fehl. Wir empfehlen Ihnen, Ihre Lambda @Edge -Funktion so zu aktualisieren, dass der Körper diese Grenzwerte nicht überschreitet.

Antwort-Timeout und Keep-Alive-Timeout (nur benutzerdefinierte Ursprünge)

Wenn Sie Lambda @Edge -Funktionen verwenden, um das Antwort-Timeout oder das Keep-Alive-Timeout für Ihre Distributionsursprünge festzulegen, stellen Sie sicher, dass Sie einen Wert angeben, den Ihr Ursprung unterstützen kann. Weitere Informationen finden Sie unter Quoten für Antwort- und Keep-Alive-Timeouts.

Berichte, Metriken und Protokolle

CloudFront bietet verschiedene Optionen für die Berichterstattung, Überwachung und Protokollierung Ihrer CloudFront Ressourcen. Sie können die folgenden Aufgaben ausführen:

- Berichte zur Nutzung und Aktivität Ihrer CloudFront Distributionen anzeigen und herunterladen, darunter Abrechnungsberichte, Cache-Statistiken, beliebte Inhalte und Top-Referrer.
- Überwachen und verfolgen Sie CloudFront, einschließlich Ihrer Edge-Computing-Funktionen, direkt in der CloudFront Konsole oder mithilfe von Amazon CloudWatch. CloudFront sendet Metriken an CloudWatch für Verteilungen und Edge-Funktionen, sowohl an Lambda @Edge als auch an Functions. CloudFront
- Zeigen Sie Protokolle für die Viewer-Anfragen an, die Ihre CloudFront Distributionen mit Standardprotokollen oder Echtzeitprotokollen erhalten. Zusätzlich zu Viewer-Anforderungsprotokollen können Sie CloudWatch Logs verwenden, um Protokolle für Ihre Edge-Funktionen, sowohl Lambda @Edge als auch CloudFront Functions, abzurufen. Sie können es auch verwenden AWS CloudTrail , um Protokolle der CloudFront API-Aktivitäten in Ihrem AWS-Konto abzurufen.
- Verfolgen Sie Konfigurationsänderungen an Ihren CloudFront Ressourcen mithilfe von AWS Config.

Weitere Informationen zu diesen Optionen finden Sie in den folgenden Themen.

Themen

AWS Abrechnungs- und Nutzungsberichte für CloudFront

AWS bietet zwei Nutzungsberichte für CloudFront:

- Der AWS Abrechnungsbericht bietet eine allgemeine Übersicht über alle Aktivitäten AWS-Services , die Sie verwenden, einschließlich CloudFront:

- Der AWS Nutzungsbericht ist eine Zusammenfassung der Aktivitäten für einen bestimmten Service, aggregiert nach Stunde, Tag oder Monat. Er enthält auch Nutzungstabellen, die eine grafische Darstellung Ihrer CloudFront Nutzung bieten.

> **Note**
>
> Wie bei anderen AWS-Services Anbietern wird Ihnen nur das CloudFront berechnet, was Sie tatsächlich nutzen. Weitere Informationen finden Sie unter CloudFront Preise.

Themen

Sehen Sie sich den AWS Abrechnungsbericht an für CloudFront

Auf der Seite Rechnungen in der AWS Fakturierung und Kostenmanagement Konsole finden Sie eine Zusammenfassung Ihrer AWS Nutzung und Gebühren, sortiert nach Diensten.

Um den AWS Abrechnungsbericht einzusehen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Fakturierung und Kostenmanagement Konsole unter https://console.aws.amazon.com/costmanagement/.
2. Wählen Sie im Navigationsbereich Rechnungen aus.
3. Wählen Sie einen Abrechnungszeitraum (zum Beispiel August 2023).
4. Wählen Sie auf der Registerkarte Gebühren nach Service die Option Global oder den AWS-Region Namen aus CloudFront, und erweitern Sie dann die Option.
5. Um einen detaillierten Abrechnungsbericht im CSV-Format herunterzuladen, wählen Sie Alle als CSV herunterladen aus.

Weitere Informationen zu Ihrer AWS Rechnung finden Sie im AWS Billing Benutzerhandbuch unter Ihre Rechnung anzeigen.

Der Abrechnungsbericht enthält die folgenden Werte, die gelten für CloudFront:

- ProductCode – `AmazonCloudFront`
- UsageType— Einer der folgenden Werte:
  - Ein Code zur Identifizierung des Typs der Datenübertragung
  - `Invalidations`
  - `Executions-CloudFrontFunctions`
  - `KeyValueStore-APIOperations`
  - `KeyValueStore-EdgeReads`
  - `RealTimeLog-KinesisDataStream`
  - `SSL-Cert-Custom`
- ItemDescription— Eine Beschreibung des Abrechnungstarifs für die UsageType.
- UsageStart Datum und UsageEndDate— Der Tag, für den sich die Nutzung bezieht, in koordinierter Weltzeit (UTC).
- UsageQuantity— Einer der folgenden Werte:
  - Die Anzahl der Anforderungen in dem angegebenen Zeitraum
  - Die übertragene Datenmenge in Gigabyte
  - Die Anzahl der invalidierten Objekte
  - Die Summe der anteiligen Monate, in denen Sie über SSL-Zertifikate für CloudFront aktivierte Distributionen verfügten. Wenn Sie beispielsweise ein Zertifikat für eine aktivierte Verteilung für einen kompletten Monat und ein anderes Zertifikat für eine aktivierte Verteilung für die Hälfte des Monats aktiviert ist, beträgt dieser Wert 1,5.

Sehen Sie sich den Nutzungsbericht für an AWS CloudFront

AWS bietet einen CloudFront Nutzungsbericht, der detaillierter als der Abrechnungsbericht, aber weniger detailliert als die CloudFront Zugriffsprotokolle ist. Der Nutzungsbericht bietet gesammelte Nutzungsdaten nach Stunde, Tag und Monat und listet Operationen nach Region und Verwendungstyp auf, z. B. die Übertragung von Daten aus der Australien-Region.

Um den AWS Nutzungsbericht einzusehen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Fakturierung und Kostenmanagement Konsole unter https://console.aws.amazon.com/costmanagement/.
2. Wählen Sie im Navigationsbereich die Option Cost & Reports aus.

3. Wählen Sie im Abschnitt AWS Nutzungsbericht die Option Nutzungsbericht erstellen aus.
4. Wählen Sie auf der Seite Nutzungsbericht herunterladen unter Services Amazon aus CloudFront
5. Wählen Sie den Nutzungstyp aus.
6. Wählen Sie die Operation.
7. Wählen Sie den Zeitraum für den Bericht. Wenn Sie Benutzerdefinierter Datumsbereich wählen, müssen Sie den Datumsbereich für den Bericht manuell angeben.
8. Wählen Sie unter Berichtsgranularität die Option Stündlich, Täglich oder Monatlich aus.
9. Wählen Sie Herunterladen und dann XML-Bericht oder CSV-Bericht aus.

Weitere Informationen zum AWS Nutzungsbericht finden Sie unter AWS Nutzungsbericht im AWS Data Exports Benutzerhandbuch.

Der CloudFront Nutzungsbericht enthält die folgenden Werte:

- Service – `AmazonCloudFront`
- Operation — HTTP-Methode. Werte sind u. a. angegeben für `DELETE`, `GET`, `HEAD`, `OPTIONS`, `PATCH`, `POST` und `PUT`.
- UsageType— Einer der folgenden Werte:
  - Ein Code zur Identifizierung des Typs der Datenübertragung
  - `Invalidations`
  - `Executions-CloudFrontFunctions`
  - `KeyValueStore-APIOperations`
  - `KeyValueStore-EdgeReads`
  - `RealTimeLog-KinesisDataStream`
  - `SSL-Cert-Custom`
- Ressource — Entweder die ID der CloudFront Distribution, die der Nutzung zugeordnet ist, oder die Zertifikat-ID eines SSL-Zertifikats, das Sie einer CloudFront Distribution zugeordnet haben.
- StartTime/EndTime— Der Tag, für den sich die Nutzung bezieht, in koordinierter Weltzeit (UTC).
- UsageValue— 1) Die Anzahl der Anfragen während des angegebenen Zeitraums oder 2) die übertragene Datenmenge in Byte.

Wenn Sie Amazon S3 als Quelle für verwenden CloudFront, sollten Sie erwägen, den Nutzungsbericht auch für Amazon S3 zu erstellen. Wenn Sie Amazon S3 jedoch für andere Zwecke

als als Quelle für Ihre CloudFront Distributionen verwenden, ist möglicherweise nicht klar, welcher Teil für Ihre CloudFront Nutzung gilt.

> **Tip**
>
> Wenn Sie detaillierte Informationen zu jeder Anfrage für CloudFront Ihre Objekte erhalten möchten, aktivieren Sie die CloudFront Zugriffsprotokolle für Ihre Distribution. Weitere Informationen finden Sie unter the section called “Standardprotokollierung (Zugriffsprotokolle)”.

Weitere Informationen zu den CloudFront Gebühren und Nutzungsarten in Ihren Berichten finden Sie unterthe section called “Interpretieren Sie Ihre AWS Rechnungs- und Nutzungsberichte für CloudFront”.

Interpretieren Sie Ihre AWS Rechnungs- und Nutzungsberichte für CloudFront

Sobald Sie den Abrechnungsbericht und den Nutzungsbericht haben, können Sie dieses Thema nutzen, um zu verstehen, wie Sie die einzelnen CloudFront Gebühren, die auf Ihrer Rechnung erscheinen, und die entsprechende Nutzungsart für jede Gebühr interpretieren. Dieses Thema enthält die Codes und AWS-Region Abkürzungen, die in beiden Berichten vorkommen können.

Die meisten Codes in den beiden Spalten haben eine Abkürzung aus zwei Buchstaben, die den Ausführungsort der Aktivität angibt. In der folgenden Tabelle wird ein Code *`region`* in Ihrer AWS Rechnung und im Nutzungsbericht durch eine der folgenden aus zwei Buchstaben bestehenden Abkürzungen ersetzt:

- AP: Hongkong, Philippinen, Südkorea, Taiwan, und Singapur (Asien-Pazifik)
- AU: Australien
- CA: Kanada
- EU: Europa und Israel
- IN: Indien
- JP: Japan
- ME: Naher Osten
- SA: Südamerika

- US: Vereinigte Staaten
- ZA: Südafrika

Weitere Informationen zur Preisgestaltung von AWS-Region finden Sie unter CloudFront Amazon-Preise.

Hinweise

- In dieser Tabelle sind die Gebühren für die Übertragung Ihrer Objekte von einem Amazon S3 S3-Bucket zu CloudFront Edge-Standorten nicht enthalten. Diese Gebühren, sofern vorhanden, finden sich im Bereich AWS -Datenübertragung Ihrer AWS -Rechnung.
- In der ersten Spalte werden die Gebühren aufgeführt, die in Ihrem AWS Rechnungsbericht aufgeführt sind, und es wird erklärt, was die einzelnen Gebühren bedeuten.
- In der zweiten Spalte werden Elemente aufgeführt, die im AWS Nutzungsbericht aufgeführt sind, und es wird der Zusammenhang zwischen Rechnungsgebühren und Nutzungsberichtselementen angezeigt.

CloudFront Gebühren in Ihrer AWS Rechnung	Werte in der UsageType Spalte im AWS Nutzungsbericht
region- DataTransfer -Out-Bytes Gesamtzahl der Byte, die von CloudFron t Edge-Standorten als Antwort *region* auf Benutzer `GET` und `HEAD` Anfragen bereitgestellt wurden.	*region*-Out-Bytes-HTTP-statisch: Über HTTP übertragene Bytes für Objekte mit TTL ≥ 3 600 Sekunden *region*-Out-Bytes-HTTPS-statisch: Über HTTPS übertragene Bytes für Objekte mit TTL ≥ 3 600 Sekunden *region*-Out-Bytes-HTTP-dynamisch: Über HTTP übertragene Bytes für Objekte mit TTL < 3 600 Sekunden *region*-Out-Bytes-HTTPS-dynamisch:

CloudFront Gebühren in Ihrer AWS Rechnung	Werte in der UsageType Spalte im AWS Nutzungsbericht
	Über HTTPS übertragene Bytes für Objekte mit TTL < 3 600 Sekunden *region*-Out-Bytes-HTTP-Proxy: Bytes, die als Antwort CloudFront auf,,, und Anfragen über HTTP an `DELETE` Zuschauer zurückgegeben wurden. `OPTIONS` `PATCH` `POST` `PUT` *region*-Out-Bytes-HTTPS-Proxy: Bytes, die als Antwort CloudFront auf,,, und Anfragen über HTTPS an `DELETE` Zuschauer zurückgegeben wurden. `OPTIONS` `PATCH` `POST` `PUT` Dies schließt Bytes ein, die über gRPC CloudFront an die Zuschauer zurückgegeben werden.

CloudFront Gebühren in Ihrer AWS Rechnung	Werte in der UsageType Spalte im AWS Nutzungsbericht
region- -Aus DataTransfer - OBytes Gesamtzahl der Byte, die als Reaktion auf,,, und PUT Anfragen von CloudFront Edge-Standorten an DELETE Ihren Ursprung oder Ihre Edge-Funktion übertragen wurden. OPTIONS PATCH POST Die Gebühren beinhalten die Datenübertragung für WebSocket Daten vom Client zum Server.	*region*-Out- OBytes -HTTP-Proxy Gesamtzahl der Byte, die als Antwort auf,,, und Anfragen über HTTP von CloudFront Edge-Standorten zu DELETE Ihrer Ursprungs- oder Edge-Funktion übertragen wurden. OPTIONS PATCH POST PUT *region*-Out- OBytes -HTTPS-Proxy Gesamtzahl der Byte, die als Antwort auf,,, und Anfragen über HTTPS von CloudFront Edge-Standorten zu DELETE Ihrer Ursprungs- oder Edge-Funktion übertragen wurden. OPTIONS PATCH POST PUT Dazu gehören Bytes, die über gRPC von CloudFront Edge-Standorten zu Ihrem Ursprung oder Ihren CloudFront Funktionen übertragen werden.
region-Anfragen — Stufe 1 Anzahl der HTTP-GET- und -HEAD-Anforderungen	*region*-Anforderungen-HTTP-statisch Anzahl der HTTP-GET- und -HEAD-Anforderungen für Objekte mit TTL ≥ 3 600 Sekunden *region*-Anforderungen-HTTP-dynamisch Anzahl der HTTP-GET- und -HEAD-Anforderungen für Objekte mit TTL < 3 600 Sekunden

CloudFront Gebühren in Ihrer AWS Rechnung	Werte in der UsageType Spalte im AWS Nutzungsbericht
region-Anfragen der Stufe 2-HTTPS Anzahl der HTTPS-GET- und -HEAD-Anforderungen	*region*-Request-HTTPS-statisch Anzahl der HTTPS-GET- und -HEAD-Anforderungen für Objekte mit TTL ≥ 3 600 Sekunden *region*-Anforderungen-HTTPS-dynamisch Anzahl der HTTPS-GET- und -HEAD-Anforderungen für Objekte mit TTL < 3 600 Sekunden
region-Anforderungen-HTTP-Proxy Anzahl der HTTP-DELETE,,, und PUT Anfragen OPTIONSPATCHPOST, die an Ihre Origin- oder CloudFront Edge-Funktion weitergeleitet werden. Beinhaltet auch die Anzahl der WebSocket HTTP-Anfragen (GETAnfragen mit dem `Upgrade: websocket` Header), die an Ihre CloudFront Origin- oder Edge-Funktion weitergeleitet werden.	*region*-Requests-HTTP-Proxy Entspricht dem entsprechenden Artikel auf Ihrer Rechnung. CloudFront

CloudFront Gebühren in Ihrer AWS Rechnung	Werte in der UsageType Spalte im AWS Nutzungsbericht
region-Anforderungen-HTTPS-Proxy Anzahl der HTTPS-`DELETE`,,, und `PUT` Anfragen `OPTIONS` `PATCH` `POST`, die an Ihre Origin- oder CloudFront Edge-Funktion weitergeleitet werden. Dies umfasst auch folgende Anforderungstypen: • Die Anzahl der WebSocketHTTPS-Anfragen (`GET`Anfragen mit dem `Upgrade: websocket` Header), die an Ihre Origin- oder Edge-Funktion CloudFront weitergeleitet werden. • Die Anzahl der HTTPS-gRPC-Anfragen.	*region*-Requests-HTTPS-Proxy Entspricht dem entsprechenden Artikel auf Ihrer Rechnung. CloudFront
region-Anforderungen-HTTPS-Proxy-Datei Anzahl der HTTPS-,, und `POST` Anfragen `DELETE` `OPTIONS` `PATCH`, die mit einer Verschlüsselung auf Feldebene verarbeitet wurden, die an Ihre Origin- oder Edge-Funktion weitergeleitet wird. CloudFront	*region*-Requests-HTTPS-Proxy-Datei Entspricht dem entsprechenden Artikel auf Ihrer Rechnung. CloudFront
region-Bytes- OriginShield Gesamtzahl der Bytes, die vom Ursprung auf einen beliebigen regionalen Edge-Cache übertragen wurden, einschließlich des regionalen Edge-Caches, der als Origin Shield aktiviert ist.	*region*-Bytes- OriginShield Entspricht dem entsprechenden Artikel auf Ihrer CloudFront Rechnung.

CloudFront Gebühren in Ihrer AWS Rechnung	Werte in der UsageType Spalte im AWS Nutzungsbericht
region-OBytes-OriginShield Gesamtzahl der Bytes, die zum Ursprung von einem beliebigen regionalen Edge-Cache übertragen wurden, einschließlich des regionalen Edge-Caches, der als Origin Shield aktiviert ist.	*region*-OBytes-OriginShield Entspricht dem entsprechenden Artikel auf Ihrer CloudFront Rechnung.
region-Anfragen- OriginShield Anzahl der Anfragen, die zu Origin Shield als inkrementelle Ebene gehen. Bei dynamischen (nicht zwischenspeicherbaren) Anforderungen, die an den Ursprung weitergeleitet werden, ist Origin Shield immer eine inkrementelle Ebene. Bei zwischenspeicherbaren Anfragen ist Origin Shield manchmal eine inkrementelle Ebene. Weitere Informationen finden Sie unter the section called “Schätzung der Origin Shield-Kosten”.	*region*-Anfragen- OriginShield Entspricht dem entsprechenden Artikel auf Ihrer CloudFront Rechnung.
Invalidations Die Gebühr für die Ungültigerklärung von Objekten (das Entfernen der Objekte aus CloudFront Randbereichen). Weitere Informationen finden Sie unter Zahlen Sie für die Ungültigerklärung der Datei.	Invalidations Entspricht dem entsprechenden Artikel auf Ihrer CloudFront Rechnung.

CloudFront Gebühren in Ihrer AWS Rechnung	Werte in der UsageType Spalte im AWS Nutzungsbericht
SSL-Cert-Custom Die Gebühr für die Verwendung eines SSL-Zertifikats mit einem CloudFront alternativen Domainnamen wie example.com anstelle des CloudFront Standard-SSL-Zertifikats und des Domainnamens, CloudFront der Ihrer Distribut ion zugewiesen wurde.	SSL-Cert-Custom Entspricht dem entsprechenden Artikel auf Ihrer CloudFront Rechnung.
RealTimeLog-KinesisDataStream Die Gebühr für die Anzahl der Zeilen, die für Echtzeitprotokolle generiert wurden.	RealTimeLog-KinesisDataStream Entspricht dem entsprechenden Artikel auf Ihrer CloudFront Rechnung.
Hinrichtungen- CloudFrontFunctions Die Gebühr für die Anzahl der CloudFront Funktionsaufrufen.	Hinrichtungen- CloudFrontFunctions Entspricht dem entsprechenden Artikel auf Ihrer CloudFront Rechnung.
region-Lambda-Edge-Anfrage Die Gebühr für die Anzahl der Lambda @Edge -Funktionsaufrufen.	***region***-Lambda-Edge-Anfrage Entspricht dem entsprechenden Artikel auf Ihrer Rechnung. CloudFront
region-Lambda-Edge-GB-Sekunde Die Gebühr für den Zeitraum vom Aufruf Ihrer Lambda @Edge -Funktion bis zu ihrer Rückkehr oder Beendigung.	***region***-Lambda-Edge-GB-Sekunde Entspricht dem entsprechenden Artikel auf Ihrer Rechnung. CloudFront
KeyValueStore-EdgeReads Die Gebühr für die Anzahl der Leseaufrufe der CloudFront KeyValueS toreMethoden`get()`,`exists()`, und`meta()`. Weitere Informationen finden Sie unter Hilfsmethoden für Schlüsselwertspeicher.	KeyValueStore-EdgeReads Entspricht dem entsprechenden Posten auf Ihrer CloudFront Rechnung.

CloudFront Gebühren in Ihrer AWS Rechnung	Werte in der UsageType Spalte im AWS Nutzungsbericht
KeyValueStore-APIOperations Die Gebühr für die Anzahl der CloudFront KeyValueStoreAPI-Aufrufe.	KeyValueStore-APIOperations Entspricht dem entsprechenden Artikel auf Ihrer CloudFront Rechnung.

CloudFront Berichte in der Konsole anzeigen

Jeder Bericht bietet detaillierte Informationen und Visualisierungen, sodass Sie die Bereitstellung von Inhalten optimieren, Leistungsengpässe identifizieren und datengestützte Entscheidungen treffen können. Ganz gleich, ob Sie die Cache-Effizienz überwachen, Verkehrsmuster analysieren oder Ihre Zuschauer besser verstehen möchten — mit diesen Berichten können Sie Ihre Distributionen effektiv überwachen und analysieren. CloudFront

In der Konsole können Sie sich die folgenden Berichte über Ihre CloudFront Aktivitäten ansehen:

Themen

- CloudFront Cache-Statistikberichte anzeigen
- Berichte über CloudFront beliebte Objekte anzeigen
- Berichte zu den CloudFront wichtigsten Referrern anzeigen
- CloudFront Nutzungsberichte anzeigen
- Zuschauerberichte anzeigen CloudFront

Die meisten dieser Berichte basieren auf den Daten in den CloudFront Zugriffsprotokollen, die detaillierte Informationen zu jeder eingehenden Benutzeranfrage CloudFront enthalten. Sie müssen Zugriffsprotokolle nicht aktivieren, um die Berichte anzeigen zu können. Weitere Informationen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).

CloudFront Cache-Statistikberichte anzeigen

Der CloudFront Amazon-Cache-Statistikbericht enthält die folgenden Informationen:

- Anfragen insgesamt — Die Gesamtzahl der Anfragen für alle HTTP-Statuscodes (z. B. 200 oder 404) und alle Methoden (zum Beispiel GET, HEAD oder POST)

- Prozentsatz der Zuschaueranfragen nach Ergebnistyp — Treffer, Fehlschläge und Fehler als Prozentsatz der Gesamtzahl der Zuschaueranfragen für die ausgewählte CloudFront Distribution
- An Zuschauer übertragene Byte — Gesamtzahl der Byte und der Anzahl fehlgeschlagener Byte
- HTTP-Statuscodes — Viewer-Anfragen nach HTTP-Statuscode
- Prozentsatz der GET-Anfragen, deren Download nicht abgeschlossen wurde — Viewer-GET-Anfragen, die das Herunterladen des angeforderten Objekts nicht abgeschlossen haben, als Prozentsatz der gesamten Anfragen

Die Daten für diese Statistiken stammen aus derselben Quelle wie die CloudFront Zugriffsprotokolle. Sie müssen die Zugriffsprotokollierung jedoch nicht aktivieren, um Cache-Statistiken anzuzeigen.

Sie können Diagramme für einen bestimmten Datumsbereich in den vergangenen 60 Tagen mit einzelnen Datenpunkten pro Stunde oder pro Tag anzeigen. In der Regel können Sie aktuelle Daten bis zu Anfragen anzeigen, die CloudFront erst vor einer Stunde empfangenen hat; gelegentlich werden Daten aber mit bis zu 24 Stunden Verspätung übermittelt.

Themen

- CloudFront Cache-Statistikberichte in der Konsole anzeigen
- Laden Sie Daten im CSV-Format herunter
- Wie hängen Cache-Statistikdiagramme mit Daten in den CloudFront Standardprotokollen (Zugriffsprotokollen) zusammen

CloudFront Cache-Statistikberichte in der Konsole anzeigen

Sie können den CloudFront Cache-Statistikbericht in der Konsole anzeigen.

Um den CloudFront Cache-Statistikbericht anzuzeigen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich die Option Cache-Statistiken aus.
3. Wählen Sie im Bereich CloudFront Cache Statistics Reports unter Start Date (Startdatum) und End Date (Enddatum) den Datumsbereich aus, für den Sie die Diagramme zu Cache-Statistiken anzeigen möchten. Die verfügbaren Datumsbereiche sind von dem Wert abhängig, den Sie unter Granularity (Granularität) auswählen:

- Daily (Täglich) – Um Diagramme mit einem einzigen Datenpunkt pro Tag anzuzeigen, wählen Sie einen Datumsbereich in den letzten 60 Tagen aus.
- Hourly (Stündlich) – Um Diagramme mit einem einzigen Datenpunkt pro Stunde anzeigen, wählen Sie einen Datumsbereich von bis zu 14 Tagen in den letzten 60 Tagen aus.

Datum und Uhrzeit entsprechen der Zeitzone UTC (Coordinated Universal Time).

4. Wählen Sie unter Granularity (Granularität) aus, ob Sie Diagramme mit einem Datenpunkt pro Tag oder einem Datenpunkt pro Stunde anzeigen möchten. Wenn Sie einen Datumsbereich größer als 14 Tage angeben, ist die Option "Ein Datenpunkt pro Stunde" nicht verfügbar.
5. Wählen Sie unter Viewer Location (Standort des Betrachters) den Kontinent aus, von dem die Anfragen gesendet werden, oder wählen Sie die Option All Locations (Alle Standorte). Cache-Statistikdiagramme enthalten Daten für Anfragen, die vom angegebenen Speicherort CloudFront empfangen wurden.
6. Wählen Sie in der Liste Distribution die Verteilungen aus, für die Sie in den Nutzungsdiagrammen Daten anzeigen möchten:

   - Eine individuelle Verteilung — In den Diagrammen werden Daten für die ausgewählte CloudFront Verteilung angezeigt. In der Verteilerliste werden die Vertriebs-ID und gegebenenfalls alternative Domainnamen (CNAMEs) für die Verteilung angezeigt. Wenn eine Distribution keine alternativen Domainnamen hat, enthält die Liste Ursprungsdomänennamen für den Vertrieb.
   - Alle Verteilungen — Die Diagramme zeigen summierte Daten für alle Verteilungen, die mit der aktuellen Verteilung verknüpft sind, mit Ausnahme der Verteilungen AWS-Konto, die Sie gelöscht haben.
7. Wählen Sie Aktualisieren.

> **Tip**
>
> - Um Daten für einen täglichen oder stündlichen Datenpunkt in einem Diagramm anzuzeigen, bewegen Sie den Mauszeiger über den Datenpunkt.
> - Bei Diagrammen, die übertragene Daten zeigen, können Sie die vertikale Skala auf Gigabyte, Megabyte oder Kilobyte ändern.

Laden Sie Daten im CSV-Format herunter

Sie können den Cache Statistics-Bericht im CSV-Format herunterladen. In diesem Abschnitt wird erklärt, wie der heruntergeladen wird. Außerdem werden die Werte in dem Bericht beschrieben.

So laden Sie den Cache Statistics-Bericht im CSV-Format herunter

1. Wählen Sie beim Anzeigen des Cache-Statistikberichts CSV aus.
2. Wählen Sie im Dialogfeld zum Öffnen der Datei, ob Sie die Datei öffnen oder speichern möchten.

Informationen zu dem Bericht

Die ersten Zeilen des Berichts enthalten die folgenden Informationen:

Version

Die Version des Formats für diese CSV-Datei

Bericht

Der Name des Berichts.

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde

StartDateUTC

Der Beginn des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

EndDateUTC

Das Ende des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

GeneratedTimeUTC

Das Datum und die Uhrzeit, zu der Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

Granularity

Ob die einzelnen Zeilen in dem Bericht für eine Stunde oder einen Tag stehen

ViewerLocation

Der Kontinent, von dem die Betrachteranfragen gesendet wurden, oder ALL, wenn Sie einen Bericht für alle Standorte herunterladen möchten

Daten im Cache Statistics-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder ALL, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden).
Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

ViewerLocation

Der Kontinent, von dem die Betrachteranfragen gesendet wurden, oder ALL, wenn Sie einen Bericht für alle Standorte herunterladen möchten

TimeBucket

Die Stunde bzw. der Tag, für den die Daten gelten, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

RequestCount

Die Gesamtzahl der Anfragen für alle HTTP-Statuscodes (z. B. 200 oder 404) und alle Methoden (z. B. GET, HEAD oder POST)

HitCount

Die Anzahl der Viewer-Anfragen, für die das Objekt von einem CloudFront Edge-Cache aus bedient wird.

MissCount

Die Anzahl der Viewer-Anfragen, für die sich das Objekt derzeit nicht in einem Edge-Cache befindet. Das Objekt CloudFront muss also von Ihrem Ursprung abgerufen werden.

ErrorCount

Die Anzahl der Betrachteranfragen, die zu einem Fehler führten, sodass CloudFront das Objekt nicht gesendet hat

IncompleteDownloadCount

Die Anzahl der Betrachteranfragen, für die der Betrachter den Download eines Objekts begonnen aber nicht abgeschlossen hat

HTTP2xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein 2xx-Wert (erfolgreich abgeschlossen) zurückgegeben wurde

HTTP3xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein 3xx-Wert (zusätzliche Aktion erforderlich) zurückgegeben wurde

HTTP4xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein 4xx-Wert (Client-Fehler) zurückgegeben wurde

HTTP5xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein 5xx-Wert (Server-Fehler) zurückgegeben wurde

TotalBytes

Die Gesamtzahl der Byte, die den Zuschauern als Antwort CloudFront auf alle Anfragen für alle HTTP-Methoden zur Verfügung gestellt wurden.

BytesFromMisses

Die Anzahl der an Viewer übertragenen Bytes für Objekte, die zum Zeitpunkt der Anfrage nicht im Edge-Cache vorhanden waren. Dieser Wert ist eine gute Näherung für die Anzahl der Byte, die von Ihrem Ursprung zu den CloudFront Edge-Caches übertragen wurden. Hier sind allerdings Anfragen für Objekte, die sich bereits im Edge-Cache befinden aber abgelaufen sind, nicht enthalten.

Wie hängen Cache-Statistikdiagramme mit Daten in den CloudFront Standardprotokollen (Zugriffsprotokollen) zusammen

Die folgende Tabelle zeigt, wie die Cache-Statistikdiagramme in der CloudFront Konsole den Werten in den CloudFront Zugriffsprotokollen entsprechen. Weitere Hinweise zu CloudFront Zugriffsprotokollen finden Sie unterStandardprotokollierung (Zugriffsprotokolle).

Anfragen insgesamt

Dieses Diagramm zeigt die Gesamtzahl der Anfragen für alle HTTP-Statuscodes (z. B. 200 oder 404) und alle Methoden (z. B. `GET`, `HEAD` oder `POST`) an. Die gesamten Anfragen in dieser Tabelle entsprechen der Gesamtzahl der Anfragen im Zugriffsprotokoll für den gleichen Zeitraum.

Prozentsatz der Betrachteranfragen nach Ergebnistyp

Dieses Diagramm zeigt Treffer, Fehlschläge und Fehler als Prozentsatz der Gesamtzahl der Zuschaueranfragen für die ausgewählte CloudFront Distribution:

- Treffer — Eine Viewer-Anfrage, für die das Objekt von einem CloudFront Edge-Cache aus bedient wird. In den Zugriffsprotokollen sind das die Anfragen, für die unter `x-edge-response-result-type` als Wert `Hit` aufgeführt ist.
- Miss — Eine Viewer-Anfrage, für die sich das Objekt derzeit nicht in einem Edge-Cache befindet, also das Objekt von Ihrem Ursprung abrufen CloudFront muss. In den Zugriffsprotokollen sind das die Anfragen, für die unter `x-edge-response-result-type` als Wert `Miss` aufgeführt ist.
- Fehler — Eine Viewer-Anfrage, die zu einem Fehler geführt hat, also das Objekt CloudFront nicht bedient hat. In den Zugriffsprotokollen sind das die Anfragen, für die unter `x-edge-response-result-type` als Wert `Error`, `LimitExceeded` oder `CapacityExceeded` aufgeführt ist.

Das Diagramm enthält keine Aktualisierungstreffer. Dies sind Anforderungen für Objekte, die sich zwar im Edge-Cache befinden, aber abgelaufen sind. In den Zugriffsprotokollen sind das die Anfragen, für die unter `x-edge-response-result-type` als Wert `RefreshHit` aufgeführt ist.

An Betrachter weitergeleitete Bytes

Dieses Diagramm zeigt zwei Werte:

- Byte insgesamt — Die Gesamtzahl der Byte, die Zuschauern als Antwort CloudFront auf alle Anfragen für alle HTTP-Methoden bereitgestellt wurden. In CloudFront Zugriffsprotokollen ist die

Gesamtzahl der Byte die Summe der Werte in der `sc-bytes` Spalte für alle Anfragen im selben Zeitraum.

- Bytes from Misses (Bytes aus Fehlschlägen) – die Anzahl der Bytes, die Betrachtern für Objekte bereitgestellt wurden, die sich zum Zeitpunkt der Anforderung nicht im Edge-Cache befunden haben. In CloudFront Zugriffsprotokollen ist Byte aus Fehlschlägen die Summe der Werte in der `sc-bytes` Spalte für Anfragen, für die der Wert von `x-edge-result-type` ist`Miss`. Dieser Wert ist eine gute Näherung für die Anzahl der Byte, die von Ihrem Ursprung an CloudFront Edge-Caches übertragen wurden. Hier sind allerdings Anfragen für Objekte, die sich bereits im Edge-Cache befinden aber abgelaufen sind, nicht enthalten.

HTTP-Statuscodes

In diesem Diagramm werden Betrachteranfragen nach HTTP-Statuscodes angezeigt. In CloudFront Zugriffsprotokollen werden Statuscodes in der `sc-status` folgenden Spalte angezeigt:

- 2xx – Die Anforderung war erfolgreich.
- 3xx – Es sind zusätzliche Aktionen erforderlich. Beispiel: 301 (dauerhaft verschoben) bedeutet, dass das angeforderte Objekt an einen anderen Speicherort verschoben wurde.
- 4xx – Für den Client ist anscheinend ein Fehler aufgetreten. Beispiel: 404 (nicht gefunden) bedeutet, dass der Client ein Objekt angefordert hat, das nicht gefunden wurde.
- 5xx – Der Ursprungs-Server hat nicht auf die Anforderung geantwortet. Beispiel: 503 (Service nicht verfügbar) bedeutet, dass der Ursprungsserver zurzeit nicht verfügbar ist.

Prozentsatz der GET-Anfragen, für die der Download nicht abgeschlossen wurde

In diesem Diagramm werden `GET`-Anfragen, bei denen der Download des angeforderten Objekts nicht abgeschlossen wurde, als Prozentsatz an den Anfragen insgesamt angezeigt. In der Regel kann der Download eines Objekts nicht abgeschlossen werden, weil der Betrachter den Download abgebrochen hat, z. B. indem auf einen anderen Link geklickt oder der Browsers geschlossen wurde. In CloudFront Zugriffsprotokollen haben diese Anfragen den Wert `200` in der `sc-status` Spalte und den Wert `Error` in der `x-edge-result-type` Spalte.

Berichte über CloudFront beliebte Objekte anzeigen

Sehen Sie sich den Amazon-Bericht über CloudFront beliebte Objekte an, um die 50 beliebtesten Objekte für eine Verteilung in einem bestimmten Zeitraum in den letzten 60 Tagen zu sehen. Sie können auch Statistiken zu diesen Objekten einsehen, darunter die folgenden:

- Anzahl der Anfragen für das Objekt
- Anzahl der Treffer und Fehlschläge
- Trefferrate
- Anzahl der bei Fehlschlägen bereitgestellten Byte
- Gesamtzahl der bedienten Byte
- Anzahl unvollständiger Downloads
- Anzahl der Anfragen nach HTTP-Statuscode (2xx, 3xx, 4xx und 5xx)

Die Daten für diese Statistiken stammen aus derselben Quelle wie die Zugriffsprotokolle. CloudFront Sie müssen die Zugriffsprotokollierung jedoch nicht aktivieren, um beliebte Objekte anzeigen zu können.

Themen

Berichte über CloudFront beliebte Objekte in der Konsole anzeigen

Sie können den Bericht über CloudFront beliebte Objekte in der Konsole einsehen.

Um beliebte Objekte für eine CloudFront Distribution anzuzeigen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich die Option Beliebte Objekte aus.
3. Wählen Sie im Bereich CF; Popular Objects ReportCloudFront (Popular Objects-Bericht) unter Start Date (Startdatum) und End Date (Enddatum) den Datumsbereich aus, für den Sie eine Liste der beliebten Objekte anzeigen möchten. Sie können alle Datumsbereiche in den vergangenen 60 Tagen auswählen.

   Datum und Uhrzeit entsprechen der Zeitzone UTC (Coordinated Universal Time).

4. Wählen Sie in der Liste Distribution die Verteilung aus, für die Sie eine Liste der beliebten Objekte anzeigen möchten.
5. Wählen Sie Aktualisieren.

Wie CloudFront berechnet die Statistik beliebter Objekte

Um eine genaue Zählung der 50 wichtigsten Objekte in Ihrer Verteilung zu erhalten, CloudFront zählen Sie die Anfragen für alle Ihre Objekte ab Mitternacht in 10-Minuten-Intervallen und führen für die nächsten 24 Stunden eine fortlaufende Summe der 150 wichtigsten Objekte. (speichert CloudFront außerdem die täglichen Gesamtwerte für die 150 wichtigsten Objekte für 60 Tage.)

Am Ende der Liste steigen Objekte ständig auf oder fallen von der Liste ab, sodass es sich bei den Gesamtwerten für diese Objekte um Näherungswerte handelt. Die 50 Objekte, die in der Liste mit 150 Objekten ganz oben stehen, können in der Liste steigen und fallen, aber sie werden selten ganz aus der Liste gestrichen, sodass die Gesamtwerte für diese Objekte zuverlässiger sind.

Wenn ein Objekt aus der Liste der 150 wichtigsten Objekte gestrichen wird und dann im Laufe eines Tages wieder auf der Liste auftaucht, wird eine geschätzte Anzahl von Anfragen für den Zeitraum CloudFront hinzugefügt, in dem das Objekt in der Liste fehlte. Die Schätzung basiert auf der Anzahl der Anfragen für das Objekt, das in diesem Zeitraum am Ende der Liste geführt wurde.

Wenn das Objekt später am Tag in die Top 50 der Objekte aufsteigt, führen die Schätzungen der Anzahl der Anfragen, die CloudFront empfangen wurden, während das Objekt unter den ersten 150 Objekten war, in der Regel dazu, dass die Anzahl der Anfragen im Bericht über beliebte Objekte die Anzahl der Anfragen übersteigt, die in den Zugriffsprotokollen für dieses Objekt erscheinen.

Laden Sie Daten im CSV-Format herunter

Sie können den Popular Objects-Bericht im CSV-Format herunterladen. In diesem Abschnitt wird erklärt, wie der heruntergeladen wird. Außerdem werden die Werte in dem Bericht beschrieben.

So laden Sie den Popular Objects-Bericht im CSV-Format herunter

1. Wählen Sie CSV aus, wenn Sie sich den Bericht über beliebte Objekte ansehen.
2. Wählen Sie im Dialogfeld zum Öffnen der Datei, ob Sie die Datei öffnen oder speichern möchten.

Informationen zu dem Bericht

Die ersten Zeilen des Berichts enthalten die folgenden Informationen:

Version

Die Version des Formats für diese CSV-Datei

Bericht

Der Name des Berichts.

DistributionID

Die ID der Verteilung, für die Sie den Bericht ausgeführt haben.

StartDateUTC

Der Beginn des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

EndDateUTC

Das Ende des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

GeneratedTimeUTC

Das Datum und die Uhrzeit, zu der Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

Daten im Popular Objects-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die Sie den Bericht ausgeführt haben.

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden). Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

Objekt

Die letzten 500 Zeichen der URL für das Objekt

RequestCount

Die Gesamtzahl der Anfragen für dieses Objekt

HitCount

Die Anzahl der Viewer-Anfragen, für die das Objekt von einem CloudFront Edge-Cache aus bedient wird.

MissCount

Die Anzahl der Viewer-Anfragen, für die sich das Objekt derzeit nicht in einem Edge-Cache befindet. Das Objekt CloudFront muss also von Ihrem Ursprung abgerufen werden.

HitCountPct

Der Wert von `HitCount` als Prozentsatz des Werts unter `RequestCount`

BytesFromMisses

Das Volumen der an die Viewer übertragenen Bytes für dieses Objekt, wenn es zum Zeitpunkt der Anfrage nicht im Edge-Cache vorhanden war

TotalBytes

Die Gesamtzahl der Byte, die den Zuschauern CloudFront für dieses Objekt als Antwort auf alle Anfragen für alle HTTP-Methoden zur Verfügung gestellt wurden.

IncompleteDownloadCount

Die Anzahl der Betrachteranfragen für dieses Objekt, für die der Betrachter den Download eines Objekts begonnen aber nicht abgeschlossen hat

HTTP2xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein 2xx-Wert (erfolgreich abgeschlossen) zurückgegeben wurde

HTTP3xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein 3xx-Wert (zusätzliche Aktion erforderlich) zurückgegeben wurde

HTTP4xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein 4xx-Wert (Client-Fehler) zurückgegeben wurde

HTTP5xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein 5xx-Wert (Server-Fehler) zurückgegeben wurde

Wie die Daten im Bericht über beliebte Objekte mit den Daten in den CloudFront Standardprotokollen (Zugriffsprotokollen) zusammenhängen

Die folgende Liste zeigt, wie die Werte im Bericht über beliebte Objekte in der CloudFront Konsole den Werten in den CloudFront Zugriffsprotokollen entsprechen. Weitere Informationen zu CloudFront-Zugriffsprotokollen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).

URL

Die letzten 500 Zeichen der URL, die Betrachter verwenden, um auf das Objekt zuzugreifen

Anforderungen

Die Gesamtzahl der Anfragen für das Objekt Dieser Wert entspricht im Allgemeinen weitgehend der Anzahl der `GET` Anfragen für das Objekt in den CloudFront Zugriffsprotokollen.

Treffer

Die Anzahl der Betrachteranfragen, für die das Objekt aus einem CloudFront -Edge-Cache gesendet wurde. In den Zugriffsprotokollen sind das die Anfragen, für die unter `x-edge-response-result-type` als Wert `Hit` aufgeführt ist.

Fehlgriffe

Die Anzahl der Viewer-Anfragen, für die sich das Objekt nicht in einem Edge-Cache befand, also das Objekt von Ihrem Ursprung CloudFront abgerufen wurde. In den Zugriffsprotokollen sind das die Anfragen, für die unter `x-edge-response-result-type` als Wert `Miss` aufgeführt ist.

Trefferrate

Der Wert in der Spalte Hits (Treffer) als Prozentsatz des Werts in der Spalte Requests (Anfragen).

Bytes für Fehlgriffe

Die Anzahl der an Viewer übertragenen Bytes für Objekte, die zum Zeitpunkt der Anfrage nicht im Edge-Cache vorhanden waren. In CloudFront Zugriffsprotokollen entspricht der Wert Byte aus Fehlschlägen der Summe der Werte in der `sc-bytes` Spalte für Anfragen, für die der `x-edge-result-type` Wert `Miss` von

Bytes insgesamt

Die Gesamtzahl der Byte, die Zuschauern CloudFront als Antwort auf alle Anfragen nach dem Objekt für alle HTTP-Methoden zur Verfügung gestellt wurden. In CloudFront Zugriffsprotokollen ist die Gesamtzahl der Byte die Summe der Werte in der `sc-bytes` Spalte für alle Anfragen im selben Zeitraum.

Unvollständige Downloads

Die Anzahl der Betrachteranfragen, für die der Download des angeforderten Objekts nicht abgeschlossen wurde. In der Regel kann der Download eines Objekts nicht abgeschlossen werden, weil der Betrachter den Download abgebrochen hat, z. B. indem auf einen anderen Link geklickt oder der Browsers geschlossen wurde. In CloudFront Zugriffsprotokollen haben diese Anfragen den Wert `200` in der `sc-status` Spalte und den Wert `Error` in der `x-edge-result-type` Spalte.

2xx

Die Anzahl der Anfragen mit dem HTTP-Statuscode `2xx`, `Successful`. In CloudFront Zugriffsprotokollen werden Statuscodes in der `sc-status` Spalte angezeigt.

3xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein `3xx`, `Redirection`. `3xx` Statuscodes zeigen an, dass zusätzliche Aktionen erforderlich sind. Beispiel: 301 (dauerhaft verschoben) bedeutet, dass das angeforderte Objekt an einen anderen Speicherort verschoben wurde.

4xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein `4xx`, `Client Error`. `4xx` Statuscodes zeigen an, dass zusätzliche Aktionen erforderlich sind. Beispiel: 404 (nicht gefunden) bedeutet, dass der Client ein Objekt angefordert hat, das nicht gefunden wurde.

5xx

Die Anzahl der Betrachteranfragen, für die als HTTP-Statuscode ein `5xx`, `Server Error`. `5xx` Statuscodes zeigen an, dass der Ursprungs-Server nicht auf die Anfrage geantwortet hat. Beispiel: 503 (Service nicht verfügbar) bedeutet, dass der Ursprungsserver zurzeit nicht verfügbar ist.

Berichte zu den CloudFront wichtigsten Referrern anzeigen

Der CloudFront Bericht über die häufigsten Verweise enthält für jeden Zeitraum der letzten 60 Tage Folgendes:

- Die 25 häufigsten Verweise (Domänen von Websites, von denen die meisten HTTP- und HTTPS-Anfragen für Objekte stammen, die CloudFront für Ihre Distribution verteilt werden)
- Anzahl der Anfragen von einem Referrer

- Anzahl der Anfragen von einem Referrer als Prozentsatz der Gesamtzahl der Anfragen im angegebenen Zeitraum

Die Daten für den Bericht über die häufigsten Verweise stammen aus derselben Quelle wie CloudFront die Zugriffsprotokolle. Sie müssen die Zugriffsprotokollierung jedoch nicht aktivieren, um die wichtigsten Verweise anzeigen zu können.

Top-Referrer können Suchmaschinen, andere Websites, die direkt auf Ihre Objekte verweisen, oder Ihre eigene Website sein. Wenn beispielsweise `https://example.com/index.html` Links zu 10 Grafiken enthalten, `example.com` ist dies der Referrer für alle 10 Grafiken.

> **Note**
>
> Wenn ein Benutzer eine URL direkt in die Adressleiste des Browsers eingibt, wird kein Referrer für das angeforderte Objekt gesendet.

Themen

Berichte über CloudFront die häufigsten Verweise in der Konsole anzeigen

Sie können den Bericht über die häufigsten CloudFront Referrer in der Konsole einsehen.

Um die häufigsten Referrer für eine Distribution anzuzeigen CloudFront

1. Melden Sie sich bei an AWS Management Console und öffnen Sie die CloudFront Konsole unter. https://console.aws.amazon.com/cloudfront/v4/home
2. Wählen Sie im Navigationsbereich die Option Top Referrers aus.
3. Wählen Sie im Bereich CF; Top Referrers ReportCloudFront (Top Referrers-Bericht) unter Start Date (Startdatum) und End Date (Enddatum) den Datumsbereich aus, für den Sie eine Liste der häufigsten Referrer anzeigen möchten.

Datum und Uhrzeit entsprechen der Zeitzone UTC (Coordinated Universal Time).

4. Wählen Sie in der Liste Distribution die Verteilung aus, für die Sie eine Liste der häufigsten Referrer anzeigen möchten.
5. Wählen Sie Aktualisieren.

Wie CloudFront berechnet die Statistik der häufigsten Verweise

Um eine genaue Anzahl der 25 häufigsten Verweise zu erhalten, CloudFront zählt die Anfragen für all Ihre Objekte in 10-Minuten-Intervallen und erstellt eine fortlaufende Summe der 75 häufigsten Verweise. Am Ende der Liste steigen Referrer ständig auf oder fallen von der Liste ab, sodass es sich bei den Gesamtwerten für diese Referrer um Näherungswerte handelt.

Die 25 Verweiser an der Spitze der Liste von 75 Referrern können in der Liste aufsteigen und fallen, aber sie werden selten ganz von der Liste gestrichen, sodass die Gesamtwerte für diese Verweiser in der Regel zuverlässiger sind.

Laden Sie Daten im CSV-Format herunter

Sie können den Top Referrers-Bericht im CSV-Format herunterladen. In diesem Abschnitt wird erklärt, wie der heruntergeladen wird. Außerdem werden die Werte in dem Bericht beschrieben.

So laden Sie den Top Referrers-Bericht im CSV-Format herunter

1. Wählen Sie CSV aus, wenn Sie sich den Bericht mit den häufigsten Referrern ansehen.
2. Wählen Sie im Dialogfeld zum Öffnen der Datei, ob Sie die Datei öffnen oder speichern möchten.

Informationen zu dem Bericht

Die ersten Zeilen des Berichts enthalten die folgenden Informationen:

Version

Die Version des Formats für diese CSV-Datei

Bericht

Der Name des Berichts.

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder ALL, wenn der Bericht für alle Verteilungen ausgeführt wurde

StartDateUTC

Der Beginn des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

EndDateUTC

Das Ende des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

GeneratedTimeUTC

Das Datum und die Uhrzeit, zu der Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

Daten im Top Referrers-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder ALL, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden).
Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

Referrer

Der Domain-Name des Referrers

RequestCount

Die Gesamtzahl der Anfragen von dem Domain-Namen in der Spalte `Referrer`

RequestsPct

Die Anzahl der von einem Referrer gesendeten Anfragen als Prozentsatz an der Gesamtzahl der Anfragen während des angegebenen Zeitraums

Wie die Daten im Bericht mit den häufigsten Referrern mit den Daten in den CloudFront Standardprotokollen (Zugriffsprotokollen) zusammenhängen

Die folgende Liste zeigt, wie die Werte im Bericht „Top Referrers“ in der CloudFront Konsole den Werten in den Zugriffsprotokollen entsprechen. CloudFront Weitere Informationen zu CloudFront-Zugriffsprotokollen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).

Referrer

Der Domain-Name des Referrers In den Zugriffsprotokollen werden die Referrer in der Spalte `cs(Referer)` aufgeführt.

Anzahl der Anfragen

Die Gesamtzahl der Anfragen von dem Domain-Namen in der Spalte Referrer. Dieser Wert entspricht in der Regel weitgehend der Anzahl der `GET` Anfragen des Referrers in CloudFront den Zugriffsprotokollen.

Anfrage %

Die Anzahl der von einem Referrer gesendeten Anfragen als Prozentsatz an der Gesamtzahl der Anfragen während des angegebenen Zeitraums Wenn mehr als 25 Referrer vorhanden sind, können Sie Request % (Anfrage %) nicht basierend auf den Daten in dieser Tabelle berechnen, da in der Spalte Request Count (Anzahl der Anfragen) nicht alle Anfragen für den angegebenen Zeitraum enthalten sind.

CloudFront Nutzungsberichte anzeigen

Die CloudFront Nutzungsberichte enthalten die folgenden Informationen:

- Anzahl der Anfragen — Zeigt die Gesamtzahl der Anfragen an, die in jedem Zeitintervall für die angegebene CloudFront Verteilung von Edge-Standorten in der ausgewählten Region CloudFront beantwortet wurden.
- Nach Protokoll übertragene Daten und nach Ziel übertragene Daten — Beide zeigen die Gesamtmenge der Daten, die von CloudFront Edge-Standorten in der ausgewählten Region in jedem Zeitintervall für die angegebene CloudFront Verteilung übertragen wurden. Sie trennen die Daten auf unterschiedliche Weise, wie folgt:
  - Nach Protokoll – Trennt die Daten nach Protokoll: HTTP oder HTTPS.

- Nach Ziel — Die Daten werden nach Zielorten aufgeteilt: an Ihre Zuschauer oder an Ihren Absender.

Der CloudFront Nutzungsbericht basiert auf dem AWS Nutzungsbericht für CloudFront. Für diesen Bericht ist keine zusätzliche Konfiguration erforderlich. Weitere Informationen finden Sie unter Sehen Sie sich den Nutzungsbericht für an AWS CloudFront.

Sie können Berichte für einen bestimmten Zeitraum der letzten 60 Tage mit Datenpunkten jede Stunde oder jeden Tag anzeigen. In der Regel können Sie Daten zu Anfragen einsehen, die erst vor vier Stunden CloudFront eingegangen sind. Gelegentlich können sich die Daten jedoch um bis zu 24 Stunden verzögern.

Weitere Informationen finden Sie unter Wie hängen die Nutzungsdiagramme mit den Daten im CloudFront Nutzungsbericht zusammen.

Themen

CloudFront Nutzungsberichte in der Konsole anzeigen

Sie können den CloudFront Nutzungsbericht in der Konsole einsehen.

Um CloudFront Nutzungsberichte anzuzeigen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Klicken Sie im Navigationsbereich auf Nutzungsberichte.
3. Wählen Sie im Bereich CloudFront Cache Usage Reports unter Start Date (Startdatum) und End Date (Enddatum) den Datumsbereich aus, für den Sie die Nutzungsdiagramme anzeigen möchten. Die verfügbaren Datumsbereiche sind von dem Wert abhängig, den Sie unter Granularity (Granularität) auswählen:

   - Daily (Täglich) – Um Diagramme mit einem einzigen Datenpunkt pro Tag anzuzeigen, wählen Sie einen Datumsbereich in den letzten 60 Tagen aus.

- Hourly (Stündlich) – Um Diagramme mit einem einzigen Datenpunkt pro Stunde anzeigen, wählen Sie einen Datumsbereich von bis zu 14 Tagen in den letzten 60 Tagen aus.

Datum und Uhrzeit entsprechen der Zeitzone UTC (Coordinated Universal Time).

4. Wählen Sie unter Granularity (Granularität) aus, ob Sie Diagramme mit einem Datenpunkt pro Tag oder einem Datenpunkt pro Stunde anzeigen möchten. Wenn Sie einen Datumsbereich größer als 14 Tage angeben, ist die Option "Ein Datenpunkt pro Stunde" nicht verfügbar.
5. Wählen Sie unter Abrechnungsregion die CloudFront Abrechnungsregion mit den Daten aus, die Sie anzeigen möchten, oder wählen Sie Alle Regionen. Nutzungsdiagramme enthalten Daten für Anfragen, die CloudFront an Edge-Standorten in der angegebenen Region verarbeitet. Die Region, in der Anfragen CloudFront bearbeitet werden, entspricht möglicherweise nicht dem Standort Ihrer Zuschauer.

   Wählen Sie nur Regionen aus, die in der Preisklasse für Ihren Vertrieb enthalten sind. Andernfalls enthalten die Nutzungstabellen wahrscheinlich keine Daten. Wenn Sie beispielsweise Preisklasse 200 für Ihren Vertrieb ausgewählt haben, sind die Abrechnungsregionen Südamerika und Australien nicht enthalten, sodass Ihre Anfragen aus diesen Regionen in der CloudFront Regel nicht bearbeitet werden. Weitere Informationen zu Preisklassen finden Sie unter CloudFrontPreisgestaltung.
6. Wählen Sie in der Liste Distribution die Verteilungen aus, für die Sie in den Nutzungsdiagrammen Daten anzeigen möchten:

   - Eine individuelle Verteilung — In den Diagrammen werden Daten für die ausgewählte CloudFront Verteilung angezeigt. In der Verteilerliste werden die Vertriebs-ID und gegebenenfalls alternative Domainnamen (CNAMEs) für die Verteilung angezeigt. Wenn für eine Verteilung keine alternativen Domain-Namen vorhanden sind, enthält die Liste die Domain-Namen der Ursprungsserver für die Verteilung.
   - Alle Verteilungen (ausgenommen gelöschte Verteilungen) — In den Diagrammen werden summierte Daten für alle Verteilungen angezeigt, die mit dem aktuellen AWS Konto verknüpft sind, mit Ausnahme von Verteilungen, die Sie gelöscht haben.
   - Alle gelöschten Verteilungen — Die Diagramme zeigen summierte Daten für alle Verteilungen an, die dem AWS Girokonto zugeordnet sind und in den letzten 60 Tagen gelöscht wurden.
7. Wählen Sie „Grafiken aktualisieren“.

> **Tip**
>
> - Um Daten für einen täglichen oder stündlichen Datenpunkt in einem Diagramm anzuzeigen, bewegen Sie den Mauszeiger über den Datenpunkt.
> - Beachten Sie bei Diagrammen, die übertragene Daten zeigen, dass Sie die vertikale Skala auf Gigabyte, Megabyte oder Kilobyte ändern können.

Laden Sie Daten im CSV-Format herunter

Sie können den Nutzungsbericht im CSV-Format herunterladen. In diesem Abschnitt wird erklärt, wie der heruntergeladen wird. Außerdem werden die Werte in dem Bericht beschrieben.

So laden Sie den Usage-Bericht im CSV-Format herunter

1. Wählen Sie beim Anzeigen des Nutzungsberichts CSV aus.
2. Wählen Sie im Dialogfeld zum Öffnen der Datei, ob Sie die Datei öffnen oder speichern möchten.

Informationen zu dem Bericht

Die ersten Zeilen des Berichts enthalten die folgenden Informationen:

Version

Die Version des Formats für diese CSV-Datei

Bericht

Der Name des Berichts.

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde; oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde; oder `ALL_DELETED`, wenn der Bericht für alle gelöschten Verteilungen ausgeführt wurde

StartDateUTC

Der Beginn des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

EndDateUTC

Das Ende des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

GeneratedTimeUTC

Das Datum und die Uhrzeit, zu der Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

Granularity

Ob die einzelnen Zeilen in dem Bericht für eine Stunde oder einen Tag stehen

BillingRegion

Der Kontinent, von dem die Betrachteranfragen gesendet wurden, oder `ALL`, wenn Sie einen Bericht für alle Fakturierungsregionen herunterladen möchten

Daten im Usage-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde; oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde; oder `ALL_DELETED`, wenn der Bericht für alle gelöschten Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden). Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

BillingRegion

Die CloudFront Abrechnungsregion, für die Sie den Bericht ausgeführt haben, oder`ALL`.

TimeBucket

Die Stunde bzw. der Tag, für den die Daten gelten, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

HTTP

Die Anzahl der HTTP-Anfragen, CloudFront auf die in jedem Zeitintervall für die angegebene CloudFront Verteilung von Edge-Standorten in der ausgewählten Region aus geantwortet wurde. Gültige Werte sind:

- Die Anzahl der GET HEAD AND-Anfragen, die CloudFront dazu führen, dass Daten an Ihre Zuschauer übertragen werden
- Die Anzahl derDELETE,, OPTIONSPATCH, und PUT AnfragenPOST, die dazu führen, dass Daten CloudFront an Ihren Ursprung übertragen werden

HTTPS

Die Anzahl der HTTPS-Anfragen, die in jedem Zeitintervall für die angegebene CloudFront Verteilung von Edge-Standorten in der ausgewählten Region CloudFront beantwortet wurden. Gültige Werte sind:

- Die Anzahl der GET HEAD AND-Anfragen, die CloudFront dazu führen, dass Daten an Ihre Zuschauer übertragen werden
- Die Anzahl derDELETE,, OPTIONSPATCH, und PUT AnfragenPOST, die dazu führen, dass Daten CloudFront an Ihren Ursprung übertragen werden

HTTPBytes

Die Gesamtmenge der Daten, die während des Zeitraums für die angegebene CloudFront Verteilung von CloudFront Edge-Standorten in der ausgewählten Abrechnungsregion über HTTP übertragen wurden. Gültige Werte sind:

- Daten, die als Antwort CloudFront auf GET und HEAD Anfragen von Ihren Zuschauern an Ihre Zuschauer übertragen werden
- Daten, die von Ihren Zuschauern an CloudFront AnfragenDELETE,OPTIONS, PATCHPOST, und PUT Anfragen übertragen werden
- Daten CloudFront , die von Ihren Zuschauern als Antwort aufDELETE,, OPTIONS PATCHPOST, und PUT Anfragen an Ihre Zuschauer übertragen werden

HTTPSBytes

Die Gesamtmenge der Daten, die während des Zeitraums für die angegebene CloudFront Verteilung von CloudFront Edge-Standorten in der ausgewählten Abrechnungsregion über HTTPS übertragen wurden. Gültige Werte sind:

- Daten, die als Antwort CloudFront auf GET und HEAD Anfragen von Ihren Zuschauern an Ihre Zuschauer übertragen wurden

- Daten, die von Ihren Zuschauern an CloudFront Anfragen`DELETE`,`OPTIONS`, `PATCH``POST`, und `PUT` Anfragen übertragen werden
- Daten CloudFront , die von Ihren Zuschauern als Antwort auf`DELETE`,, `OPTIONS` `PATCH``POST`, und `PUT` Anfragen an Ihre Zuschauer übertragen werden

BytesIn

Die Gesamtmenge der Daten, die für`DELETE`,, `OPTIONS` `PATCH``POST`, und `PUT` Anfragen in der ausgewählten Region in jedem Zeitintervall für die angegebene CloudFront Verteilung von zu Ihrem Ursprung übertragen wurden. CloudFront

BytesOut

Die Gesamtmenge der Daten, die in jedem Zeitintervall für die angegebene CloudFront Verteilung über HTTP und HTTPS von CloudFront Ihren Zuschauern in der ausgewählten Region übertragen wurden. Gültige Werte sind:

- Daten, die als Antwort CloudFront auf `GET` und `HEAD` Anfragen von Ihren Zuschauern an Ihre Zuschauer übertragen wurden
- Daten CloudFront , die als Antwort auf`DELETE`,, `OPTIONS``PATCH`, `POST` und `PUT` Anfragen von an Ihre Zuschauer übertragen werden

Wie hängen die Nutzungsdiagramme mit den Daten im CloudFront Nutzungsbericht zusammen

Die folgende Liste zeigt, wie die Nutzungsdiagramme in der CloudFront Konsole den Werten in der Spalte Verwendungstyp im CloudFront Nutzungsbericht entsprechen.

Themen

Anzahl der Anfragen

Dieses Diagramm zeigt die Gesamtzahl der Anfragen, die in jedem Zeitintervall für die angegebene CloudFront Verteilung von Edge-Standorten in der ausgewählten Region CloudFront beantwortet wurden, getrennt nach Protokoll (HTTP oder HTTPS) und Typ (statisch, dynamisch oder Proxy).

Anzahl der HTTP-Anfragen

- *region*-requests-HTTP-static: Anzahl der HTTP `GET` - und `HEAD` Anfragen, die für Objekte mit einer TTL von ≥ 3600 Sekunden bedient wurden
- *region*-Requests-HTTP-dynamic: Anzahl der HTTP- und Anfragen, die für Objekte mit einer TTL von < 3600 Sekunden bedient wurden `GET` `HEAD`
- *region*-Requests-HTTP-Proxy: Anzahl der HTTP-,,, und Anfragen, die an Ihren Ursprung weitergeleitet werden `DELETE` `OPTIONS` `PATCH` `POST` `PUT` CloudFront

Anzahl der HTTPS-Anfragen

- *region*-requests-HTTPS-static: Anzahl der HTTPS- und Anfragen, die für Objekte mit einer TTL von ≥ 3600 Sekunden bedient wurden `GET` `HEAD`
- *region*-requests-HTTPS-dynamic: Anzahl der HTTPS- und Anfragen, die für Objekte mit einer TTL von < 3600 Sekunden bedient wurden `GET` `HEAD`
- *region*-requests-HTTPS-Proxy: Anzahl der HTTPS-,,, und Anfragen, die an Ihren Ursprung weitergeleitet werden `DELETE` `OPTIONS` `PATCH` `POST` `PUT` CloudFront

Weitergeleitete Daten nach Protokoll

Dieses Diagramm zeigt die Gesamtmenge der Daten, die von CloudFront Edge-Standorten in der ausgewählten Region in jedem Zeitintervall für die angegebene CloudFront Verteilung übertragen wurden, getrennt nach Protokoll (HTTP oder HTTPS), Typ (statisch, dynamisch oder Proxy) und Ziel (Zuschauer oder Herkunft).

Über HTTP weitergeleitete Daten

- *region*-Out-Bytes-HTTP-Static: Über HTTP bereitgestellte Bytes für Objekte mit TTL ≥ 3600 Sekunden
- *region*-Out-bytes-HTTP-dynamic: Über HTTP bereitgestellte Bytes für Objekte mit einer TTL von < 3600 Sekunden
- *region*-Out-bytes-HTTP-Proxy: Bytes, die als Antwort auf,,, und Anfragen über HTTP an Zuschauer zurückgegeben wurden CloudFront `DELETE` `OPTIONS` `PATCH` `POST` `PUT`
- *region*-Out- OBytes -HTTP-Proxy: Gesamtzahl der Byte, die als Antwort auf,,, und Anfragen per HTTP von CloudFront Edge-Standorten zu Ihrem Ursprung übertragen wurden `DELETE` `OPTIONS` `PATCH` `POST` `PUT`

Über HTTPS weitergeleitete Daten

- *region*-Out-bytes-HTTPS-static: Über HTTPS bereitgestellte Bytes für Objekte mit TTL ≥ 3600 Sekunden
- *region*-Out-Bytes-HTTPS-Dynamic: Über HTTPS bereitgestellte Bytes für Objekte mit einer TTL von < 3600 Sekunden
- *region*-Out-bytes-HTTPS-Proxy: Bytes, die als Antwort auf,,, und Anfragen über HTTPS an Zuschauer zurückgegeben wurden CloudFront `DELETE OPTIONS PATCH POST PUT`
- *region*-Out- OBytes -HTTPS-Proxy: Gesamtzahl der Byte, die als Antwort auf,,, und Anfragen über HTTPS von CloudFront Edge-Standorten zu Ihrem Ursprung übertragen wurden `DELETE OPTIONS PATCH POST PUT`

Weitergeleitete Daten nach Zieladresse

Dieses Diagramm zeigt die Gesamtmenge der Daten, die von CloudFront Edge-Standorten in der ausgewählten Region in jedem Zeitintervall für die angegebene CloudFront Verteilung übertragen wurden, getrennt nach Ziel (Zuschauer oder Herkunft), Protokoll (HTTP oder HTTPS) und Typ (statisch, dynamisch oder Proxy).

Daten, die von CloudFront zu Ihren Zuschauern übertragen wurden

- *region*-Out-Bytes-HTTP-Static: Über HTTP bereitgestellte Bytes für Objekte mit TTL ≥ 3600 Sekunden
- *region*-Out-bytes-HTTPS-static: Über HTTPS bereitgestellte Bytes für Objekte mit TTL ≥ 3600 Sekunden
- *region*-Out-bytes-HTTP-dynamic: Über HTTP bereitgestellte Bytes für Objekte mit einer TTL von < 3600 Sekunden
- *region*-Out-Bytes-HTTPS-Dynamic: Über HTTPS bereitgestellte Bytes für Objekte mit einer TTL von < 3600 Sekunden
- *region*-Out-bytes-HTTP-Proxy: Bytes, die als Antwort auf,,, und Anfragen über HTTP an Zuschauer zurückgegeben wurden CloudFront `DELETE OPTIONS PATCH POST PUT`
- *region*-Out-bytes-HTTPS-Proxy: Bytes, die als Antwort auf,,, und Anfragen über HTTPS an Zuschauer zurückgegeben wurden CloudFront `DELETE OPTIONS PATCH POST PUT`

Daten, die CloudFront von Ihrem Ursprung übertragen wurden

- *region*-Out- OBytes -HTTP-Proxy: Gesamtzahl der Byte, die als Antwort auf,,, und Anfragen per HTTP von CloudFront Edge-Standorten zu `DELETE` Ihrem Ursprung übertragen wurden `OPTIONS` `PATCH` `POST` `PUT`
- *region*-Out- OBytes -HTTPS-Proxy: Gesamtzahl der Byte, die als Antwort auf,,, und Anfragen über HTTPS von CloudFront Edge-Standorten zu Ihrem Ursprung übertragen wurden `DELETE` `OPTIONS` `PATCH` `POST` `PUT`

Zuschauerberichte anzeigen CloudFront

Die CloudFront Zuschauerberichte enthalten die folgenden Informationen für einen beliebigen Zeitraum der letzten 60 Tage:

- Geräte — Die Arten von Geräten, die am häufigsten für den Zugriff auf Ihre Inhalte verwendet werden (z. B. Desktop oder Handy)
- Browser — Die zehn Browser, die am häufigsten für den Zugriff auf Ihre Inhalte verwendet werden (wie Chrome oder Firefox)
- Betriebssysteme — Die 10 Betriebssysteme, die am häufigsten für den Zugriff auf Ihre Inhalte verwendet werden (wie Linux, macOS oder Windows)
- Standorte — Die 50 Standorte (Länder oder US-Bundesstaaten/Territorien) der Zuschauer, die am häufigsten auf deine Inhalte zugreifen
  - Du kannst dir auch Standorte mit stündlichen Datenpunkten für einen beliebigen Zeitraum von bis zu 14 Tagen in den letzten 60 Tagen anzeigen lassen

> **Note**
>
> Sie müssen die Zugriffsprotokollierung nicht aktivieren, um Zuschauerdiagramme und Berichte zu sehen.

Themen

- Wie die Daten im Standortbericht mit den Daten in den CloudFront Standardprotokollen (Zugriffsprotokollen) zusammenhängen

Sehen Sie sich die Diagramme und Berichte der Zuschauer in der Konsole an

Sie können die Diagramme und Berichte der CloudFront Zuschauer in der Konsole anzeigen.

Um die Diagramme und Berichte der CloudFront Zuschauer anzusehen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Viewers aus.
3. Wählen Sie im Bereich CloudFront Viewers unter Start Date (Startdatum) und End Date (Enddatum) den Datumsbereich aus, für den Sie die Betrachterdiagramme und -berichte anzeigen möchten.

   Für das Diagramm zu den Standorten sind die verfügbaren Datumsbereiche von dem Wert abhängig, den Sie unter Granularity (Granularität) auswählen:

   - Daily (Täglich) – Um Diagramme mit einem einzigen Datenpunkt pro Tag anzuzeigen, wählen Sie einen Datumsbereich in den letzten 60 Tagen aus.
   - Hourly (Stündlich) – Um Diagramme mit einem einzigen Datenpunkt pro Stunde anzeigen, wählen Sie einen Datumsbereich von bis zu 14 Tagen in den letzten 60 Tagen aus.

   Datum und Uhrzeit entsprechen der Zeitzone UTC (Coordinated Universal Time).
4. (Nur Diagramme zu Browsern und Betriebssystemen) Geben Sie unter Grouping (Gruppierung) an, ob Sie Browser und Betriebssysteme nach Namen (Chrome, Firefox) oder nach Namen und Version (Chrom 40.0, Firefox 35.0) gruppieren möchten.
5. (Nur Diagramme zu Standorten) Wählen Sie unter Granularity (Granularität) aus, ob Sie Diagramme mit einem Datenpunkt pro Tag oder einem Datenpunkt pro Stunde anzeigen möchten. Wenn Sie einen Datumsbereich größer als 14 Tage angeben, ist die Option "Ein Datenpunkt pro Stunde" nicht verfügbar.
6. (Nur Diagramme zu Standorten) Geben Sie unter Details an, ob die häufigsten Standorte nach Ländern oder nach US-Bundesstaaten angezeigt werden sollen.
7. Wählen Sie in der Liste Distribution die Verteilung aus, für die Sie in den Nutzungsdiagrammen Daten anzeigen möchten:

- Eine individuelle Verteilung — In den Diagrammen werden Daten für die ausgewählte CloudFront Verteilung angezeigt. Unter Distribution werden die Verteilungs-ID und ein alternativer Domain-Namen (CNAME) für die Verteilung angezeigt, sofern vorhanden. Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.
- Alle Verteilungen (außer gelöschten) – Die Diagramme zeigen zusammengefasste Daten für alle Verteilungen an, die dem aktuellen AWS -Konto zugeordnet sind. Ausgenommen sind Verteilungen, die Sie gelöscht haben.

8. Wählen Sie Aktualisieren.

Um Daten für einen täglichen oder stündlichen Datenpunkt in einem Diagramm anzuzeigen, bewegen Sie den Mauszeiger über den Datenpunkt.

Laden Sie Daten im CSV-Format herunter

Sie können alle Viewer-Berichte im CSV-Format herunterladen. In diesem Abschnitt wird erklärt, wie die Berichte heruntergeladen werden; außerdem werden die Werte in dem Bericht beschrieben.

So laden Sie den Viewer-Bericht im CSV-Format herunter

1. Wählen Sie beim Anzeigen des Viewer-Berichts CSV aus.
2. Wählen Sie die Daten aus, die Sie herunterladen möchten, z. B. zu Devices (Geräten) oder Devices Trends (Gerätetendenzen).
3. Wählen Sie im Dialogfeld zum Öffnen der Datei, ob Sie die Datei öffnen oder speichern möchten.

In den Berichten der Zuschauer enthaltene Daten

Die ersten Zeilen jedes Berichts enthalten die folgenden Informationen:

Version

Die Version des Formats für diese CSV-Datei

Bericht

Der Name des Berichts.

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder ALL, wenn der Bericht für alle Verteilungen ausgeführt wurde

StartDateUTC

Der Beginn des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

EndDateUTC

Das Ende des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

GeneratedTimeUTC

Das Datum und die Uhrzeit, zu der Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

Grouping (nur Berichte zu Browsern und Betriebssystemen)

Ob die Daten nach dem Namen oder nach dem Namen und der Version des Browsers oder des Betriebssystems gruppiert werden

Granularity

Ob die einzelnen Zeilen in dem Bericht für eine Stunde oder einen Tag stehen

Details (nur Berichte zu Standorten)

Ob Anfragen nach Land oder US-Staat aufgeführt werden

In den folgenden Themen werden die Informationen in den verschiedenen Viewer-Berichten beschrieben.

Themen

- Operating Systems Trends-Bericht
- Locations-Bericht
- Location Trends-Bericht

Devices-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden). Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

Anforderungen

Die Anzahl der Anfragen, die von jedem Gerätetyp CloudFront empfangen wurden.

RequestsPct

Die Anzahl der Anfragen, die von jedem Gerätetyp CloudFront empfangen wurden, als Prozentsatz der Gesamtzahl der Anfragen, die von allen Geräten CloudFront empfangen wurden.

Benutzerdefiniert

Anfragen, für die der Wert des HTTP-`User-Agent`-Headers nicht einem der Standard-Gerätetypen wie `Desktop` oder `Mobile` zugeordnet werden konnte

Device Trends-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden). Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

TimeBucket

Die Uhrzeit oder das Datum für die Nutzung in koordinierter Weltzeit (Coordinated Universal Time, UTC).

Desktop

Die Anzahl der Anfragen, die während des Zeitraums von Desktop-Computern CloudFront eingegangen sind.

Mobil

Die Anzahl der Anfragen, die während des Zeitraums von Mobilgeräten CloudFront eingegangen sind. Mobile Geräte können sowohl Tablets als auch Smartphones sein. Wenn nicht festgestellt CloudFront werden kann, ob eine Anfrage von einem Mobilgerät oder einem Tablet stammt, wird sie in der `Mobile` Spalte gezählt.

Smart-TV

Die Anzahl der Anfragen, die TVs während des Zeitraums von smart CloudFront eingegangen sind.

Tablet

Die Anzahl der Anfragen, die während des Zeitraums von Tablets CloudFront eingegangen sind. Wenn nicht festgestellt CloudFront werden kann, ob eine Anfrage von einem Mobilgerät oder einem Tablet stammt, wird sie in der `Mobile` Spalte gezählt.

Unbekannt

Anfragen, für die der Wert des HTTP-`User-Agent`-Headers nicht einem der Standard-Gerätetypen wie `Desktop` oder `Mobile` zugeordnet werden konnte

Leer

Die Anzahl der CloudFront eingegangenen Anfragen, die während des Zeitraums keinen Wert im `User-Agent` HTTP-Header enthielten.

Browsers-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden).
Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

Gruppe

Der Browser oder der Browser und die Version, von dem Anfragen CloudFront empfangen wurden, abhängig vom Wert von`Grouping`. Neben den Browsernamen sind die folgenden Werte möglich:

- Bot/Crawler (Bot/Crawler) – Vor allem Anforderungen von Suchmaschinen, die Ihre Inhalte indizieren.
- Empty (Leer) – Anforderungen, bei denen der Wert des HTTP-Headers `User-Agent` leer war.
- Andere — Browser, die zwar CloudFront identifiziert wurden, aber nicht zu den beliebtesten gehören. Wenn `Bot/Crawler`, `Empty` und/oder `Unknown` nicht unter den ersten neun Werten sind, werden sie ebenfalls unter `Other` aufgeführt.
- Unknown (Unbekannt) – Anforderungen bei denen der Wert des HTTP-Headers `User-Agent` keinem Standard-Browser zugeordnet werden konnte. Die meisten Anfragen in dieser Kategorie stammen von benutzerdefinierten Anwendungen oder Skripts.

Anforderungen

Die Anzahl der Anfragen, die von jedem Browsertyp CloudFront empfangen wurden.

RequestsPct

Die Anzahl der Anfragen, die von jedem Browsertyp CloudFront empfangen wurden, als Prozentsatz der Gesamtzahl der Anfragen, die während des Zeitraums CloudFront eingegangen sind.

Browser Trends-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden).
Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

TimeBucket

Die Uhrzeit oder das Datum für die Nutzung in koordinierter Weltzeit (Coordinated Universal Time, UTC).

(Browser)

Die restlichen Spalten in dem Bericht enthalten die Browser oder die Browser und deren Versionen, entsprechend dem Wert unter `Grouping`. Neben den Browsernamen sind die folgenden Werte möglich:

- Bot/Crawler (Bot/Crawler) – Vor allem Anforderungen von Suchmaschinen, die Ihre Inhalte indizieren.
- Empty (Leer) – Anforderungen, bei denen der Wert des HTTP-Headers `User-Agent` leer war.
- Andere — Browser, die zwar CloudFront identifiziert wurden, aber nicht zu den beliebtesten gehören. Wenn `Bot/Crawler`, `Empty` und/oder `Unknown` nicht unter den ersten neun Werten sind, werden sie ebenfalls unter `Other` aufgeführt.
- Unknown (Unbekannt) – Anforderungen bei denen der Wert des HTTP-Headers `User-Agent` keinem Standard-Browser zugeordnet werden konnte. Die meisten Anfragen in dieser Kategorie stammen von benutzerdefinierten Anwendungen oder Skripts.

Operating Systems-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden). Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

Gruppe

Das Betriebssystem oder das Betriebssystem und die Version, von dem CloudFront Anfragen erhalten hat, entsprechend dem Wert unter `Grouping`. Neben den Namen der Betriebssysteme sind die folgenden Werte möglich:

- Bot/Crawler (Bot/Crawler) – Vor allem Anforderungen von Suchmaschinen, die Ihre Inhalte indizieren.
- Empty (Leer) – Anforderungen, bei denen der Wert des HTTP-Headers `User-Agent` leer war.
- Andere — Betriebssysteme, die zwar CloudFront identifiziert wurden, aber nicht zu den beliebtesten gehören. Wenn `Bot/Crawler`, `Empty` und/oder `Unknown` nicht unter den ersten neun Werten sind, werden sie ebenfalls unter `Other` aufgeführt.
- Unknown (Unbekannt) – Anforderungen bei denen der Wert des HTTP-Headers `User-Agent` keinem Standard-Browser zugeordnet werden konnte. Die meisten Anfragen in dieser Kategorie stammen von benutzerdefinierten Anwendungen oder Skripts.

Anforderungen

Die Anzahl der Anfragen, die von jedem Betriebssystemtyp CloudFront empfangen wurden.

RequestsPct

Die Anzahl der Anfragen, die von jedem Betriebssystemtyp CloudFront empfangen wurden, als Prozentsatz der Gesamtzahl der Anfragen, die während des Zeitraums CloudFront eingegangen sind.

Operating Systems Trends-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden).
Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

TimeBucket

Die Uhrzeit oder das Datum für die Nutzung in koordinierter Weltzeit (Coordinated Universal Time, UTC).

(Betriebssysteme)

Die restlichen Spalten in dem Bericht enthalten die Betriebssysteme oder die Betriebssysteme und deren Versionen, entsprechend dem Wert unter `Grouping`. Neben den Namen der Betriebssysteme sind die folgenden Werte möglich:

- Bot/Crawler (Bot/Crawler) – Vor allem Anforderungen von Suchmaschinen, die Ihre Inhalte indizieren.
- Empty (Leer) – Anforderungen, bei denen der Wert des HTTP-Headers `User-Agent` leer war.
- Andere — Betriebssysteme, die zwar CloudFront identifiziert wurden, aber nicht zu den beliebtesten gehören. Wenn `Bot/Crawler`, `Empty` und/oder `Unknown` nicht unter den ersten neun Werten sind, werden sie ebenfalls unter `Other` aufgeführt.
- Unknown (Unbekannt) – Anforderungen, bei denen das Betriebssystem nicht im HTTP-Header `User-Agent` angegeben ist.

Locations-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder `ALL`, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden). Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

LocationCode

Die Abkürzung für den Standort, von dem Anfragen CloudFront empfangen wurden. Weitere Informationen zu möglichen Werten finden Sie in der Beschreibung der Standorte unter Wie die Daten im Standortbericht mit den Daten in den CloudFront Standardprotokollen (Zugriffsprotokollen) zusammenhängen.

LocationName

Der Name des Standorts, von dem Anfragen CloudFront empfangen wurden.

Anforderungen

Die Anzahl der Anfragen, die von jedem Standort CloudFront eingegangen sind.

RequestsPct

Die Anzahl der Anfragen, die von jedem Standort CloudFront empfangen wurden, als Prozentsatz der Gesamtzahl der Anfragen, die während des Zeitraums von allen Standorten CloudFront eingegangen sind.

TotalBytes

Die Anzahl der Byte, die Zuschauern in diesem Land oder Bundesstaat für die angegebene Verteilung und den angegebenen Zeitraum CloudFront zugestellt wurden.

Location Trends-Bericht

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die der Bericht ausgeführt wurde, oder ALL, wenn der Bericht für alle Verteilungen ausgeführt wurde

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden). Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

TimeBucket

Die Uhrzeit oder das Datum für die Nutzung in koordinierter Weltzeit (Coordinated Universal Time, UTC).

(Standorte)

Die restlichen Spalten im Bericht enthalten die Standorte, die von denen CloudFront Anfragen erhalten hat. Weitere Informationen zu möglichen Werten finden Sie in der Beschreibung der Standorte unter Wie die Daten im Standortbericht mit den Daten in den CloudFront Standardprotokollen (Zugriffsprotokollen) zusammenhängen.

Wie die Daten im Standortbericht mit den Daten in den CloudFront Standardprotokollen (Zugriffsprotokollen) zusammenhängen

Die folgende Liste zeigt, wie die Daten im Standortbericht in der CloudFront Konsole den Werten in den CloudFront Zugriffsprotokollen entsprechen. Weitere Informationen zu CloudFront Zugriffsprotokollen finden Sie unterStandardprotokollierung (Zugriffsprotokolle).

Ort

Das Land oder der US-Bundesstaat, in dem sich der Betrachter befindet. Die Spalte `c-ip` in den Zugriffsprotokollen enthält die IP-Adresse des Geräts, auf dem der Betrachter ausgeführt wird. Wir verwenden Geolocation-Daten zur Bestimmung des geografischen Standorts auf Basis der IP-Adresse des Geräts.

Wenn Sie den Bericht Standorte nach Ländern aufgeschlüsselt anzeigen, basiert die Länderliste auf ISO 3166-2, Codes für die Darstellung von Namen von Ländern und ihren Untergliederungen — Teil 2: Ländercode. Die Liste der Länder enthält die folgenden zusätzlichen Werte:

- Anonymous Proxy (Anonymer Proxy) – Die Anforderung stammt von einem anonymen Proxy.
- Satellite Provider (Satellitenanbieter) – Die Anforderung stammt von einem Satellitenanbieter, der Internetverbindungen für mehrere Länder bereitstellt. Zuschauer befinden sich möglicherweise in Ländern mit einem hohen Betrugsrisiko.

- Europe (Unknown) (Europa (Unbekannt) – Die Anforderung stammt von einer IP-Adresse in einem Block, der von mehreren europäischen Ländern verwendet wird. Das Land, aus dem die Anfrage stammt, kann nicht ermittelt werden. CloudFront verwendet Europa (Unbekannt) als Standard.
- Asia/Pacific (Unknown) (Asien/Pazifik (Unbekannt)) – Die Anforderung stammt von einer IP-Adresse in einem Block, der von mehreren Ländern in der Region Asien/Pazifik verwendet wird. Das Land, aus dem die Anfrage stammt, kann nicht bestimmt werden. CloudFront verwendet Asien/Pazifik (Unbekannt) als Standard.

Wenn Sie den Locations-Bericht nach US-Bundesstaaten anzeigen, beachten Sie, dass der Bericht US-Hoheitsgebiete und Regionen für die US-Streitkräfte enthalten kann.

> **Note**
>
> Wenn der Standort eines Benutzers nicht bestimmt werden CloudFront kann, wird der Standort in den Zuschauerberichten als Unbekannt angezeigt.

Anzahl der Anfragen

Die Gesamtzahl der Anfragen aus dem Land oder US-Bundesstaat, in dem sich der Betrachter befindet, für die angegebene Verteilung und im angegebenen Zeitraum. Dieser Wert entspricht in der Regel weitgehend der Anzahl der `GET` Anfragen von IP-Adressen in diesem Land oder Bundesstaat in CloudFront den Zugriffsprotokollen.

Anfrage %

Einer der folgenden Werte, entsprechend dem Wert unter Details:

- Countries (Länder) – Die Anforderungen aus diesem Land als Prozentsatz der Gesamtzahl der Anforderungen.
- US-Bundesstaaten – Die Anfragen aus diesem US-Bundesstaat als Prozentsatz an der Gesamtzahl der Anfragen aus den USA.

Wenn Anfragen aus mehr als 50 Ländern aufgetreten sind, können Sie Request % (Anfrage %) nicht basierend auf den Daten in dieser Tabelle berechnen, da in der Spalte Request Count (Anzahl der Anfragen)nicht alle Anfragen für den angegebenen Zeitraum enthalten sind.

Bytes

Die Anzahl der Byte, die Zuschauern in diesem Land oder Bundesstaat für die angegebene Verteilung und den angegebenen Zeitraum zur Verfügung CloudFront gestellt wurden. Um die Anzeige der Daten in dieser Spalte auf KB, MB oder GB zu ändern, wählen Sie den Link in der Spaltenüberschrift.

Überwachen Sie CloudFront Metriken mit Amazon CloudWatch

Amazon CloudFront ist in Amazon integriert CloudWatch und veröffentlicht automatisch Betriebsmetriken für Distributionen und Edge-Funktionen (sowohl Lambda @Edge als auch CloudFront Functions). Sie können diese Metriken verwenden, um Probleme zu beheben, nachzuverfolgen und zu debuggen. Viele dieser Metriken werden in einer Reihe von Diagrammen in der CloudFront Konsole angezeigt und sind auch über die CloudFront API oder CLI zugänglich. Alle diese Metriken sind in der CloudWatch Konsole oder über die CloudWatch API oder CLI verfügbar. CloudFront Metriken werden nicht auf CloudWatch Kontingente (früher als Limits bezeichnet) angerechnet und es fallen keine zusätzlichen Kosten an.

Zusätzlich zu den Standardmetriken für CloudFront Verteilungen können Sie gegen zusätzliche Kosten zusätzliche Metriken aktivieren. Die zusätzlichen Metriken gelten für CloudFront Verteilungen und müssen für jede Verteilung separat aktiviert werden. Weitere Informationen zu den Kosten finden Sie unter the section called “Schätzen Sie die Kosten für die zusätzlichen CloudFront Metriken ab”.

Sie können auch Alarme auf der Grundlage dieser Metriken in der CloudFront Konsole oder in der CloudWatch Konsole, API oder CLI einrichten. Sie können beispielsweise einen Alarm basierend auf der Metrik `5xxErrorRate` festlegen, die den Prozentsatz aller Viewer-Anforderungen darstellt, für die sich der HTTP-Statuscode der Antwort im Bereich `500` bis einschließlich `599` befindet. Wenn die Fehlerquote einen bestimmten Wert für eine bestimmte Zeit erreicht (z. B. 5 % der Anforderungen für 5 kontinuierliche Minuten), wird der Alarm ausgelöst. Sie geben den Wert des Alarms und seine Zeiteinheit an, wenn Sie den Alarm erstellen.

Hinweise

- Wenn Sie in der CloudFront Konsole einen CloudWatch Alarm erstellen, wird in der Region USA Ost (Nord-Virginia) ein Alarm für Sie erstellt (`us-east-1`). Wenn Sie von der CloudWatch Konsole aus einen Alarm erstellen, müssen Sie dieselbe Region verwenden.

Da CloudFront es sich um einen globalen Service handelt, werden die Messwerte für den Service nach USA Ost (Nord-Virginia) gesendet.

- Bei der Erstellung von Alarmen gelten die CloudWatch Standardpreise.

Themen

Funktionsmetriken anzeigen CloudFront und erweitern

In der Konsole können Sie Betriebsmetriken zu Ihren CloudFront Distributionen und Edge-Funktionen einsehen. CloudFront

So können Sie Funktionsmetriken anzeigen CloudFront und bearbeiten in CloudFront

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Monitoring (Überwachung) aus.
3. Um Diagramme zur Aktivität für eine bestimmte CloudFront Verteilungs- oder Edge-Funktion anzuzeigen, wählen Sie eine aus und wählen Sie dann Verteilungsmetriken anzeigen oder Metriken anzeigen aus.
4. Sie können die Diagramme anpassen, indem Sie die folgenden Schritte ausführen:

    a. Zum Ändern des Zeitraums für die im Diagramm angezeigten Informationen wählen Sie 1h (1 Stunde), 3 h (3 Stunden) oder einen anderen Zeitraum aus. Alternativ können Sie auch einen benutzerdefinierten Bereich angeben.

    b. Um zu ändern, wie oft die Informationen im Diagramm CloudFront aktualisiert werden, klicken Sie auf den Abwärtspfeil neben dem Aktualisierungssymbol und wählen Sie dann eine Aktualisierungsrate aus. Die Standardaktualisierungsrate beträgt 1 Minute, Sie können jedoch auch andere Optionen wählen.

5. Um CloudFront Diagramme in der CloudWatch Konsole anzuzeigen, wählen Sie Zum Dashboard hinzufügen. Sie müssen die Region USA Ost (Nord-Virginia) verwenden, um die Grafiken in der CloudWatch Konsole anzuzeigen.

Themen

CloudFront Standardverteilungsmetriken

Die folgenden Standardmetriken sind für alle CloudFront Verteilungen ohne zusätzliche Kosten enthalten:

Anforderungen

Die Gesamtzahl der von allen HTTP-Methoden und sowohl für HTTP- als auch für HTTPS-Anfragen empfangenen Zuschaueranfragen. CloudFront

Heruntergeladene Bytes

Die Gesamtzahl der Byte, die von Zuschauern für `GET` und `HEAD` Anfragen heruntergeladen wurden.

Hochgeladene Bytes

Die Gesamtzahl der Byte, die Zuschauer hochgeladen CloudFront`OPTIONS`, genutzt `POST` und `PUT` angefordert haben.

4xx-Fehlerrate

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `4xx` lautet.

5xx-Fehlerrate

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `5xx` lautet.

Gesamte Fehlerrate

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `4xx` oder `5xx` lautet.

Diese Metriken werden in Diagrammen für jede CloudFront Verteilung auf der Monitoring-Seite der CloudFront Konsole angezeigt. In jedem Diagramm wird die Gesamtzahl in Zeiteinheiten von 1 Minute angezeigt. Zusätzlich zum Anzeigen der Diagramme können Sie Metrikberichte auch als CSV-Dateien herunterladen.

Schalten Sie zusätzliche CloudFront Verteilungsmetriken ein

Zusätzlich zu den Standardmetriken können Sie für eine Zusatzgebühr weitere Metriken aktivieren. Weitere Informationen zu den Kosten finden Sie unter the section called “Schätzen Sie die Kosten für die zusätzlichen CloudFront Metriken ab”.

Diese zusätzlichen Metriken müssen für jede Verteilung separat aktiviert werden:

Cache-Trefferrate

Der Prozentsatz aller zwischenspeicherbaren Anfragen, für die der Inhalt aus dem Cache CloudFront bereitgestellt wurde. HTTP `POST`- und `PUT`-Anforderungen und Fehler werden nicht als cachebare Anforderungen betrachtet.

Ursprungslatenz

Die Gesamtzeit zwischen dem CloudFront Empfang einer Anfrage und dem Beginn der Antwort an das Netzwerk (nicht an den Betrachter) für Anfragen, die vom Ursprung und nicht vom CloudFront Cache aus bedient werden. Dies wird auch als Latenz des ersten Byte oder bezeichnet time-to-first-byte.

Fehlerquote nach Statuscode

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort ein bestimmter Code im Bereich `4xx` oder `5xx` ist. Diese Metrik ist für alle folgenden Fehlercodes verfügbar: `401`, `403`, `404`, `502`, `503` und `504`.

Sie können zusätzliche Metriken in der CloudFront Konsole, mit AWS CloudFormation, mit der AWS Command Line Interface (AWS CLI) oder mit der CloudFront API aktivieren.

Console

Um zusätzliche Metriken zu aktivieren

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.

2. Wählen Sie im Navigationsbereich Monitoring (Überwachung) aus.
3. Wählen Sie die Verteilung aus, für die zusätzliche Metriken aktiviert werden sollen, und wählen Sie View distribution metrics (Verteilungsmetriken anzeigen) aus.
4. Wählen Sie Manage additional metrics (Zusätzliche Metriken verwalten) aus.
5. Aktivieren Sie im Fenster Manage additional metrics (Zusätzliche Metriken verwalten) die Option Enabled (Aktiviert). Nachdem Sie die zusätzlichen Metriken aktiviert haben, können Sie das Fenster Manage additional metrics (Zusätzliche Metriken verwalten) schließen.

   Nachdem Sie die zusätzlichen Metriken aktiviert haben, werden sie in Diagrammen angezeigt. In jedem Diagramm wird die Gesamtzahl in Zeiteinheiten von 1 Minute angezeigt. Zusätzlich zum Anzeigen der Diagramme können Sie Metrikberichte auch als CSV-Dateien herunterladen.

CloudFormation

Verwenden Sie den `AWS::CloudFront::MonitoringSubscription` Ressourcentyp CloudFormation, um zusätzliche Messwerte für zu aktivieren. Das folgende Beispiel zeigt die AWS CloudFormation Vorlagensyntax im YAML-Format zur Aktivierung zusätzlicher Metriken.

```
Type: AWS::CloudFront::MonitoringSubscription
Properties:
  DistributionId: EDFDVBD6EXAMPLE
  MonitoringSubscription:
    RealtimeMetricsSubscriptionConfig:
      RealtimeMetricsSubscriptionStatus: Enabled
```

CLI

Verwenden Sie einen der folgenden Befehle, um zusätzliche Metriken mit dem AWS Command Line Interface (AWS CLI) zu verwalten:

Um zusätzliche Metriken für eine Verteilung zu aktivieren

- Verwenden Sie den Befehl create-monitoring-subscription wie im folgenden Beispiel. *EDFDVBD6EXAMPLE*Ersetzen Sie es durch die ID der Verteilung, für die Sie zusätzliche Metriken aktivieren.

```
aws cloudfront create-monitoring-subscription --
distribution-id EDFDVBD6EXAMPLE --monitoring-subscription
 RealtimeMetricsSubscriptionConfig={RealtimeMetricsSubscriptionStatus=Enabled}
```

Um zu sehen, ob zusätzliche Metriken für eine Verteilung aktiviert sind

- Verwenden Sie den Befehl get-monitoring-subscription wie im folgenden Beispiel. *EDFDVBD6EXAMPLE*Ersetzen Sie es durch die ID der Verteilung, die Sie überprüfen.

```
aws cloudfront get-monitoring-subscription --distribution-id EDFDVBD6EXAMPLE
```

Um zusätzliche Metriken für eine Verteilung zu deaktivieren

- Verwenden Sie den Befehl delete-monitoring-subscription wie im folgenden Beispiel. *EDFDVBD6EXAMPLE*Ersetzen Sie es durch die ID der Verteilung, für die Sie zusätzliche Metriken deaktivieren möchten.

```
aws cloudfront delete-monitoring-subscription --distribution-id EDFDVBD6EXAMPLE
```

API

Verwenden Sie eine der folgenden CloudFront API-Operationen, um zusätzliche Metriken mit der API zu verwalten.

- Um zusätzliche Metriken für eine Distribution zu aktivieren, verwenden Sie CreateMonitoringSubscription.
- Um zu sehen, ob zusätzliche Metriken für eine Verteilung aktiviert sind, verwenden Sie GetMonitoringSubscription.
- Um zusätzliche Metriken für eine Verteilung zu deaktivieren, verwenden Sie DeleteMonitoringSubscription.

Weitere Informationen zu diesen API-Vorgängen finden Sie in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Schätzen Sie die Kosten für die zusätzlichen CloudFront Metriken ab

Wenn Sie zusätzliche Metriken für eine Verteilung aktivieren, werden bis zu 8 Messwerte CloudWatch in die Region USA Ost (Nord-Virginia) CloudFront gesendet. CloudWatch berechnet für jede Metrik einen niedrigen, festen Tarif. Diese Rate wird nur einmal pro Monat pro Metrik berechnet (bis zu 8 Metriken pro Distribution). Da es sich um einen festen Tarif handelt, bleiben Ihre Kosten gleich, unabhängig von der Anzahl der Anfragen oder Antworten, die die CloudFront Distribution empfängt oder sendet. Die Rate pro Metrik finden Sie auf der CloudWatch Amazon-Preisseite und im CloudWatchPreisrechner. Zusätzliche API-Gebühren fallen an, wenn Sie die Metriken mit der CloudWatch API abrufen.

Standardmetriken für Lambda @Edge -Funktionen

Sie können CloudWatch Metriken verwenden, um Probleme mit Ihren Lambda @Edge -Funktionen in Echtzeit zu überwachen. Für diese Metriken fallen keine zusätzlichen Gebühren an.

Wenn Sie eine Lambda @Edge -Funktion an ein Cache-Verhalten in einer CloudFront Distribution anhängen, beginnt Lambda, automatisch Metriken an zu CloudWatch senden. Metriken sind für alle Lambda-Regionen verfügbar, aber um Metriken in der CloudWatch Konsole anzuzeigen oder die Metrikdaten von der CloudWatch API abzurufen, müssen Sie die Region USA Ost (Nord-Virginia) verwenden (`us-east-1`). Der Name der Metrikgruppe ist wie folgt formatiert:`AWS/CloudFront/`*`distribution-ID`*, wobei die ID der CloudFront Distribution *`distribution-ID`* ist, der die Lambda @Edge -Funktion zugeordnet ist. Weitere Informationen zu CloudWatch Metriken finden Sie im CloudWatch Amazon-Benutzerhandbuch.

Die folgenden Standardmetriken werden in Diagrammen für jede Lambda @Edge -Funktion auf der Monitoring-Seite der CloudFront Konsole angezeigt:

- `5xx`-Fehlerrate für Lambda@Edge
- Lambda-Ausführungsfehler
- Ungültige Lambda-Antworten
- Lambda-Drosselungen

Die Diagramme umfassen die Anzahl der Aufrufe, Fehler, Drosselungen und so weiter. In jedem Diagramm wird die Gesamtzahl in Zeiteinheiten von 1 Minute, gruppiert nach AWS -Region, angezeigt.

Wenn Sie einen Anstieg der Fehler feststellen, die Sie untersuchen möchten, können Sie eine Funktion auswählen und dann die Protokolldateien nach AWS Regionen anzeigen, bis Sie festgestellt haben, welche Funktion die Probleme verursacht und in welcher AWS Region. Weitere Informationen zur Behebung von Lambda@Edge-Fehlern finden Sie unter:

- the section called “So bestimmten Sie den Typ des Fehlers”
- Vier Schritte zum Debuggen Ihrer Inhaltsbereitstellung am AWS

Metriken für CloudFront Standardfunktionen

CloudFront Functions sendet Betriebsmetriken an Amazon, CloudWatch damit Sie Ihre Funktionen überwachen können. Das Anzeigen dieser Metriken kann Ihnen helfen, Probleme zu beheben, zu verfolgen und zu debuggen. CloudFront Functions veröffentlicht die folgenden Kennzahlen an CloudWatch:

- Invocations (`FunctionInvocations`) – Die Häufigkeit, mit der die Funktion in einem bestimmten Zeitraum gestartet (aufgerufen) wurde.
- Validierungsfehler (`FunctionValidationErrors`) – Die Anzahl der Validierungsfehler, die von der Funktion in einem bestimmten Zeitraum erzeugt werden. Validierungsfehler treten auf, wenn die Funktion erfolgreich ausgeführt wird, aber ungültige Daten (ein ungültiges Ereignisobjekt) zurückgibt.
- Ausführungsfehler (`FunctionExecutionErrors`) – Die Anzahl der Ausführungsfehler, die in einem bestimmten Zeitraum aufgetreten sind. Ausführungsfehler treten auf, wenn die Funktion nicht erfolgreich abgeschlossen werden kann.
- Auslastung berechnen (`FunctionComputeUtilization`) – Die Zeit, die die Ausführung der Funktion in Anspruch nahm, als Prozentsatz der maximal zulässigen Zeit. Zum Beispiel bedeutet ein Wert von 35, dass die Funktion in 35 % der maximal zulässigen Zeit abgeschlossen wurde. Diese Metrik ist eine Zahl zwischen 0 und 100.

  Wenn dieser Wert 100 erreicht oder nahe 100 liegt, hat die Funktion die zulässige Ausführungszeit verwendet oder ist kurz davor, sie zu nutzen, und nachfolgende Anfragen werden möglicherweise gedrosselt. Wenn Ihre Funktion bei einer Auslastung von 80% oder mehr ausgeführt wird, empfehlen wir Ihnen, Ihre Funktion zu überprüfen, um die Ausführungszeit zu verkürzen und die Auslastung zu verbessern. Beispielsweise möchten Sie möglicherweise nur Fehler protokollieren, komplexe Regex-Ausdrücke vereinfachen oder unnötiges Parsen komplexer JSON-Objekte vermeiden.

- Drosselungen (`FunctionThrottles`) – Die Häufigkeit, mit der die Funktion in einem bestimmten Zeitraum gedrosselt wurde. Funktionen können aus folgenden Gründen gedrosselt werden:
  - Die Funktion überschreitet kontinuierlich die maximal zulässige Ausführungszeit.
  - Die Funktion führt zu Kompilierungsfehlern.
  - Es gibt eine ungewöhnlich hohe Anzahl von Anforderungen pro Sekunde.

CloudFront KeyValueStore sendet außerdem die folgenden Betriebskennzahlen an Amazon CloudWatch:

- Anfragen lesen (`KvsReadRequests`) — Gibt an, wie oft die Funktion innerhalb eines bestimmten Zeitraums erfolgreich aus dem Schlüsselwertspeicher gelesen hat.
- **`KvsReadErrors`**Lesefehler () — Gibt an, wie oft die Funktion innerhalb eines bestimmten Zeitraums nicht aus dem Schlüsselwertspeicher lesen konnte.

Alle diese Metriken werden CloudWatch in der Region USA Ost (Nord-Virginia) (`us-east-1`) im CloudFront Namespace veröffentlicht. Sie können diese Metriken auch in der CloudWatch Konsole anzeigen. In der CloudWatch Konsole können Sie die Metriken pro Funktion oder pro Funktion pro Verteilung anzeigen.

Sie können damit auch Alarme einrichten CloudWatch , die auf diesen Metriken basieren.
Sie können beispielsweise einen Alarm basierend auf der Ausführungszeitmetrik (`FunctionComputeUtilization`) festlegen, die den Prozentsatz der verfügbaren Zeit darstellt, die Ihre Funktion zur Ausführung benötigte. Wenn die Ausführungszeit für einen bestimmten Zeitraum einen bestimmten Wert erreicht. Wenn Sie beispielsweise mehr als 70% der verfügbaren Zeit für 15 ununterbrochene Minuten wählen, wird der Alarm ausgelöst. Sie geben den Wert des Alarms und seine Zeiteinheit an, wenn Sie den Alarm erstellen.

> **Note**
>
> CloudFront Functions sendet CloudWatch nur Metriken an Funktionen in der `LIVE` Phase, die als Reaktion auf Produktionsanfragen und -antworten ausgeführt werden. Wenn Sie eine Funktion testen, sendet CloudFront keine Metriken an CloudWatch. Die Testausgabe enthält Informationen über Fehler, Rechenauslastung und Funktionsprotokolle (`console.log()`Anweisungen), aber diese Informationen werden nicht an gesendet CloudWatch.

Informationen darüber, wie Sie diese Metriken mit der CloudWatch API abrufen können, finden Sie unterthe section called “CloudFront Metriken”.

Erstellen von -Alarmen für -Metriken

In der CloudFront Konsole können Sie Alarme so einrichten, dass Sie von Amazon Simple Notification Service (Amazon SNS) auf der Grundlage bestimmter CloudFront Messwerte benachrichtigt werden.

Um Alarme für Messwerte zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Klicken Sie im Navigationsbereich auf Alarms (Alarme).
3. Wählen Sie Create Alarm (Alarm erstellen) aus.
4. Geben Sie für Details Folgendes an:
   a. Alarmname — Ein Name für den Alarm.
   b. Verteilung — Die CloudFront Distribution, für die Sie den Alarm erstellen.
5. Geben Sie unter Bedingung Folgendes an:
   a. Metrik — Die Metrik, für die Sie den Alarm erstellen.
   b. „IF“ <condition>— Der Schwellenwert, ab dem ein Alarm ausgelöst und eine Benachrichtigung an das Amazon SNS SNS-Thema gesendet werden CloudWatch soll. Um beispielsweise eine Benachrichtigung zu erhalten, wenn die `5xx`-Fehlerquote 1 % überschreitet, geben Sie Folgendes an:

      5xx Fehlerrate > 1
   c. „FÜR“ aufeinanderfolgende Perioden — Der Zeitraum, in dem die Bedingung erfüllt sein muss, bevor ein Alarm ausgelöst wird. Achten Sie bei der Auswahl eines Werts auf ein angemessenes Gleichgewicht zwischen einem Wert, der keinen Alarm bei vorübergehenden Problemen auslöst, sondern bei anhaltenden oder echten Problemen einen Alarm auslöst.
   d. (Optional) Benachrichtigen — Das Amazon SNS SNS-Thema, an das eine Benachrichtigung gesendet werden soll, wenn diese Metrik einen Alarm auslöst.
6. Wählen Sie Alarm erstellen aus.

Hinweise

- Verwenden Sie bei der Eingabe der Werte für die Bedingung ganze Zahlen ohne Satzzeichen. Wenn Sie beispielsweise Tausend angeben möchten, geben Sie ei **`1000`**.
- Für die drei Fehlerquotenkategorien `4xx`, `5xx` und „Total (Insgesamt)“ der Wert, den Sie als Prozentsatz angeben.
- Bei Anforderungen, heruntergeladenen Bytes und hochgeladenen Bytes handelt es sich bei dem von Ihnen angegebenen Wert um Einheiten. Zum Beispiel 1073742000 Bytes.

Weitere Informationen zum Erstellen von Amazon SNS SNS-Themen finden Sie unter Erstellen eines Amazon SNS SNS-Themas im Amazon Simple Notification Service Developer Guide.

Laden Sie Metrikdaten im CSV-Format herunter

Sie können die CloudWatch Metrikdaten für eine CloudFront Verteilung im CSV-Format herunterladen.

Um Metrikdaten im CSV-Format herunterzuladen

1. Melden Sie sich bei an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie im Navigationsbereich Monitoring (Überwachung) aus.
3. Wählen Sie die Verteilung und dann Verteilungsmetriken anzeigen aus.
4. Wählen Sie CSV herunterladen und wählen Sie dann den Zeitraum aus (z. B. Für den letzten Tag (Zeitraum von 1 Stunde)).
5. Nachdem Ihre Datei heruntergeladen wurde, öffnen Sie sie, um die folgenden Informationen anzuzeigen.

Themen

- Informationen zu dem Bericht
- Daten in dem Bericht zu Metriken

Informationen zu dem Bericht

Die ersten Zeilen des Berichts enthalten die folgenden Informationen:

Version

Die CloudFront Berichtsversion.

Bericht

Der Name des Berichts.

DistributionID

Die ID der Verteilung, für die Sie den Bericht ausgeführt haben.

StartDateUTC

Der Beginn des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

EndDateUTC

Das Ende des Datumsbereichs, für den Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

GeneratedTimeUTC

Das Datum und die Uhrzeit, zu der Sie den Bericht ausgeführt haben, in koordinierter Weltzeit (Coordinated Universal Time, UTC).

Granularity

Der Zeitraum für jede Zeile im Bericht, z. B, `ONE_MINUTE`.

Daten in dem Bericht zu Metriken

Der Bericht enthält die folgenden Werte:

DistributionID

Die ID der Verteilung, für die Sie den Bericht ausgeführt haben.

FriendlyName

Ein alternativer Domänennamen (CNAMEs) für die Verteilung (sofern vorhanden). Wenn eine Verteilung keine alternativen Domänennamen hat, enthält die Liste einen Ursprungsdomänennamen für die Verteilung.

TimeBucket

Die Uhrzeit oder das Datum für die Nutzung in koordinierter Weltzeit (Coordinated Universal Time, UTC).

Anforderungen

Die Gesamtzahl der Anforderungen für alle HTTP-Statuscodes (z. B. `200` oder `404`) und alle Methoden (z. B. `GET`, `HEAD`, `POST` usw.) während des Zeitraums.

BytesDownloaded

Die Anzahl der Bytes, die während des Zeitraums für die angegebene Verteilung von Betrachtern heruntergeladen worden sind.

BytesUploaded

Die Anzahl der Bytes, die während des Zeitraums für die angegebene Verteilung von Viewern hochgeladen wurden.

TotalErrorRatePct

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `4xx`- oder `5xx`-Fehler war, in Prozent.

4 xxErrorRate Pkt

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `4xx`-Fehler war, in Prozent.

5 Pkt xxErrorRate

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `5xx`-Fehler war, in Prozent.

Wenn Sie zusätzliche Metriken für Ihre Verteilung aktiviert haben, enthält der Bericht auch die folgenden zusätzlichen Werte:

401 ErrorRatePct

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `401`-Fehler war, in Prozent.

403 ErrorRatePct

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `403`-Fehler war, in Prozent.

404 ErrorRatePct

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `404`-Fehler war, in Prozent.

502 ErrorRatePct

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `502`-Fehler war, in Prozent.

503 ErrorRatePct

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `503`-Fehler war, in Prozent.

504 ErrorRatePct

Anforderungen, für die während des Zeitraums der angegebenen Verteilung der HTTP-Statuscode ein `504`-Fehler war, in Prozent.

OriginLatency

Die Gesamtzeit in Millisekunden vom CloudFront Empfang einer Anfrage bis zum Beginn der Antwort an das Netzwerk (nicht an den Betrachter) für Anfragen, die vom Ursprung und nicht vom Cache aus bedient wurden. CloudFront Dies wird auch als Latenz des ersten Byte oder bezeichnet. time-to-first-byte

CacheHitRate

Der Prozentsatz aller zwischenspeicherbaren Anfragen, für die der Inhalt aus dem Cache CloudFront bereitgestellt wurde. HTTP `POST`- und `PUT`-Anforderungen und Fehler werden nicht als cachebare Anforderungen betrachtet.

Arten von Metriken für CloudFront

Sie können die CloudWatch API oder AWS Command Line Interface (AWS CLI) verwenden, um die CloudFront Metriken in Programmen oder Anwendungen abzurufen, die Sie erstellen. Sie können die Rohdaten verwenden, um eigene benutzerdefinierte Dashboards, eigene alarmierende Tools usw. zu erstellen.

Weitere Informationen finden Sie get-metric-datain der AWS CLI Befehlsreferenz oder unter GetMetricDataAPI-Operation in der Amazon CloudWatch API-Referenz.

Themen

> **Note**
>
> Um die CloudFront Metriken von der CloudWatch API abzurufen, müssen Sie die Region USA Ost (Nord-Virginia) (`us-east-1`) verwenden. Sie müssen auch bestimmte Werte und Typen für jede Metrik kennen.

Werte für alle CloudFront-Metriken

Die folgenden Werte gelten für alle CloudFront Metriken:

`Namespace`

Der Wert für `Namespace` ist immer `AWS/CloudFront`.

Dimensionen

Jede CloudFront Metrik hat die folgenden Dimensionen:

`DistributionId`

Die ID der CloudFront Distribution, für die Sie Kennzahlen abrufen möchten.

`FunctionName`

Der Name der Funktion (in CloudFront Funktionen), für die Sie Metriken abrufen möchten.

Diese Dimension gilt nur für Funktionen.

`Region`

Der Wert für `Region` ist immer`Global`, weil CloudFront es sich um einen globalen Dienst handelt.

Werte für CloudFront Vertriebsmetriken

Verwenden Sie Informationen aus der folgenden Liste, um Details zu bestimmten CloudFront Verteilungsmetriken aus der CloudWatch API abzurufen. Einige dieser Metriken sind nur verfügbar, wenn Sie zusätzliche Metriken für die Verteilung aktiviert haben.

> **Note**
>
> Für jede Metrik ist nur eine Statistik, `Average` oder `Sum`, anwendbar. Die folgende Liste gibt an, welche Statistik auf diese Metrik anwendbar ist.

4xx-Fehlerrate

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `4xx` lautet.

- Metrikname: `4xxErrorRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

401-Fehlerquote

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `401` lautet. Um diese Metrik abzurufen, müssen Sie zunächst zusätzliche Metriken aktivieren.

- Metrikname: `401ErrorRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

403-Fehlerquote

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `403` lautet. Um diese Metrik abzurufen, müssen Sie zunächst zusätzliche Metriken aktivieren.

- Metrikname: `403ErrorRate`

- Gültige Statistik: `Average`
- Einheit: `Percent`

404-Fehlerquote

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `404` lautet. Um diese Metrik abzurufen, müssen Sie zunächst zusätzliche Metriken aktivieren.

- Metrikname: `404ErrorRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

5xx-Fehlerrate

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `5xx` lautet.

- Metrikname: `5xxErrorRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

502-Fehlerquote

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `502` lautet. Um diese Metrik abzurufen, müssen Sie zunächst zusätzliche Metriken aktivieren.

- Metrikname: `502ErrorRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

503-Fehlerquote

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `503` lautet. Um diese Metrik abzurufen, müssen Sie zunächst zusätzliche Metriken aktivieren.

- Metrikname: `503ErrorRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

504-Fehlerquote

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `504` lautet. Um diese Metrik abzurufen, müssen Sie zunächst zusätzliche Metriken aktivieren.

- Metrikname: `504ErrorRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

Heruntergeladene Bytes

Die Gesamtzahl der von Viewern für `GET`-, `HEAD`- und `OPTIONS`-Anforderungen heruntergeladenen Bytes.

- Metrikname: `BytesDownloaded`
- Gültige Statistik: `Sum`
- Einheit: `None`

Hochgeladene Bytes

Die Gesamtzahl der Bytes, die Viewer mit CloudFront unter Verwendung von `POST`- und `PUT`-Anforderungen zu Ihrem Ursprung hochgeladen haben.

- Metrikname: `BytesUploaded`
- Gültige Statistik: `Sum`
- Einheit: `None`

Cache-Trefferrate

Der Prozentsatz aller zwischenspeicherbaren Anfragen, für die der Inhalt aus dem Cache CloudFront bereitgestellt wurde. HTTP `POST`- und `PUT`-Anforderungen und Fehler werden nicht als cachebare Anforderungen betrachtet. Um diese Metrik abzurufen, müssen Sie zunächst zusätzliche Metriken aktivieren.

- Metrikname: `CacheHitRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

Ursprungslatenz

Die Gesamtzeit in Millisekunden vom CloudFront Empfang einer Anfrage bis zum Beginn der Antwort an das Netzwerk (nicht an den Betrachter) für Anfragen, die vom Ursprung und nicht vom Cache aus bedient werden. CloudFront Dies wird auch als Latenz des ersten Byte oder bezeichnet. time-to-first-byte Um diese Metrik abzurufen, müssen Sie zunächst zusätzliche Metriken aktivieren.

- Metrikname: `OriginLatency`

- Gültige Statistik: `Percentile`
- Einheit: `Milliseconds`

> **Note**
>
> Um eine `Percentile` Statistik von der CloudWatch API abzurufen, verwenden Sie den `ExtendedStatistics` Parameter not`Statistics`. Weitere Informationen finden Sie GetMetricStatisticsin der Amazon CloudWatch API-Referenz oder in der Referenzdokumentation für AWS SDKs.

Anforderungen

Die Gesamtzahl der von allen HTTP-Methoden und sowohl für HTTP- als auch für HTTPS-Anfragen empfangenen Zuschaueranfragen. CloudFront

- Metrikname: `Requests`
- Gültige Statistik: `Sum`
- Einheit: `None`

Gesamte Fehlerrate

Der Prozentsatz aller Viewer-Anforderungen, für die der HTTP-Statuscode der Antwort `4xx` oder `5xx` lautet.

- Metrikname: `TotalErrorRate`
- Gültige Statistik: `Average`
- Einheit: `Percent`

Werte für CloudFront Funktionsmetriken

Verwenden Sie Informationen aus der folgenden Liste, um Details zu bestimmten CloudFront Funktionsmetriken aus der CloudWatch API abzurufen.

> **Note**
>
> Für jede Metrik ist nur eine Statistik, `Average` oder `Sum`, anwendbar. Die folgende Liste gibt an, welche Statistik auf diese Metrik anwendbar ist.

Aufrufe

Die Häufigkeit, mit der die Funktion in einem bestimmten Zeitraum gestartet (aufgerufen) wurde.

- Metrikname: `FunctionInvocations`
- Gültige Statistik: `Sum`
- Einheit: `None`

Validierungsfehler

Die Anzahl der Validierungsfehler, die von der Funktion in einem bestimmten Zeitraum erzeugt werden. Validierungsfehler treten auf, wenn die Funktion erfolgreich ausgeführt wird, aber ungültige Daten (ein ungültiges Ereignisobjekt) zurückgibt.

- Metrikname: `FunctionValidationErrors`
- Gültige Statistik: `Sum`
- Einheit: `None`

Ausführungsfehler

Die Anzahl der Ausführungsfehler, die in einem bestimmten Zeitraum aufgetreten sind. Ausführungsfehler treten auf, wenn die Funktion nicht erfolgreich abgeschlossen werden kann.

- Metrikname: `FunctionExecutionErrors`
- Gültige Statistik: `Sum`
- Einheit: `None`

Computing-Auslastung

Die Zeit (0–100), die die Ausführung der Funktion in Anspruch genommen hat, als Prozentsatz der maximal zulässigen Zeit. Zum Beispiel bedeutet ein Wert von 35, dass die Funktion in 35 % der maximal zulässigen Zeit abgeschlossen wurde.

- Metrikname: `FunctionComputeUtilization`
- Gültige Statistik: `Average`
- Einheit: `Percent`

Drosselungen

Die Häufigkeit, mit der die Funktion in einem bestimmten Zeitraum gedrosselt wurde.

- Metrikname: `FunctionThrottles`
- Gültige Statistik: `Sum`
- Einheit: `None`

CloudFront und Edge-Funktionsprotokollierung

Amazon CloudFront bietet verschiedene Arten der Protokollierung an. Sie können die Zuschaueranfragen protokollieren, die zu Ihren CloudFront Distributionen kommen, oder Sie können die CloudFront Serviceaktivität (API-Aktivität) in Ihrem AWS Konto protokollieren. Sie können auch Protokolle von Ihren CloudFront Functions- und Lambda @Edge -Funktionen abrufen.

Protokollieren von Anfragen

CloudFront bietet die folgenden Möglichkeiten, die Anfragen zu protokollieren, die an Ihre Distributionen gesendet werden.

Standardprotokolle (Zugriffsprotokolle)

CloudFront Standardprotokolle enthalten detaillierte Aufzeichnungen über jede Anfrage, die an eine Distribution gestellt wird. Sie können die Protokolle für Szenarien wie Sicherheits- und Zugriffsprüfungen verwenden.

CloudFront Standardprotokolle werden an das von Ihnen angegebene Lieferziel übermittelt.

Weitere Informationen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).

Echtzeit-Protokolle

CloudFront Echtzeitprotokolle liefern Informationen über Anfragen an eine Distribution in Echtzeit (Protokolldatensätze werden innerhalb von Sekunden nach Eingang der Anfragen zugestellt). Sie können die Abtastrate für Ihre Echtzeit-Protokolle wählen, d. h. den Prozentsatz der Anfragen, für die Sie Echtzeit-Protokolleinträge erhalten möchten. Sie können auch die spezifischen Felder auswählen, die Sie in den Protokolldatensätzen erhalten möchten.

CloudFront Echtzeitprotokolle werden an den Datenstream Ihrer Wahl in Amazon Kinesis Data Streams übermittelt. CloudFront Gebühren für Echtzeitprotokolle, zusätzlich zu den Gebühren, die Ihnen für die Nutzung von Kinesis Data Streams entstehen.

Weitere Informationen finden Sie unter Verwenden Sie Echtzeitprotokolle.

Protokollieren von Edge-Funktionen

Sie können Amazon CloudWatch Logs verwenden, um Protokolle für Ihre Edge-Funktionen, sowohl Lambda @Edge als auch CloudFront Functions, abzurufen. Sie können über die CloudWatch

Konsole oder die Logs-API auf die CloudWatch Protokolle zugreifen. Weitere Informationen finden Sie unter the section called “Protokolle für Edge-Funktionen”.

Protokollieren von Service-Aktivität

Sie können AWS CloudTrail sie verwenden, um die CloudFront Serviceaktivität (API-Aktivität) in Ihrem AWS Konto zu protokollieren. CloudTrail bietet eine Aufzeichnung der API-Aktionen, die von einem Benutzer, einer Rolle oder einem AWS Dienst in ausgeführt wurden CloudFront. Anhand der von gesammelten Informationen können Sie die API-Anfrage ermitteln CloudTrail, an die die Anfrage gestellt wurde CloudFront, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details.

Weitere Informationen finden Sie unter Protokollieren Amazon CloudFront Amazon-API-Aufrufen mit AWS CloudTrail.

Weitere Informationen zur Protokollierung finden Sie in den folgenden Themen:

Themen

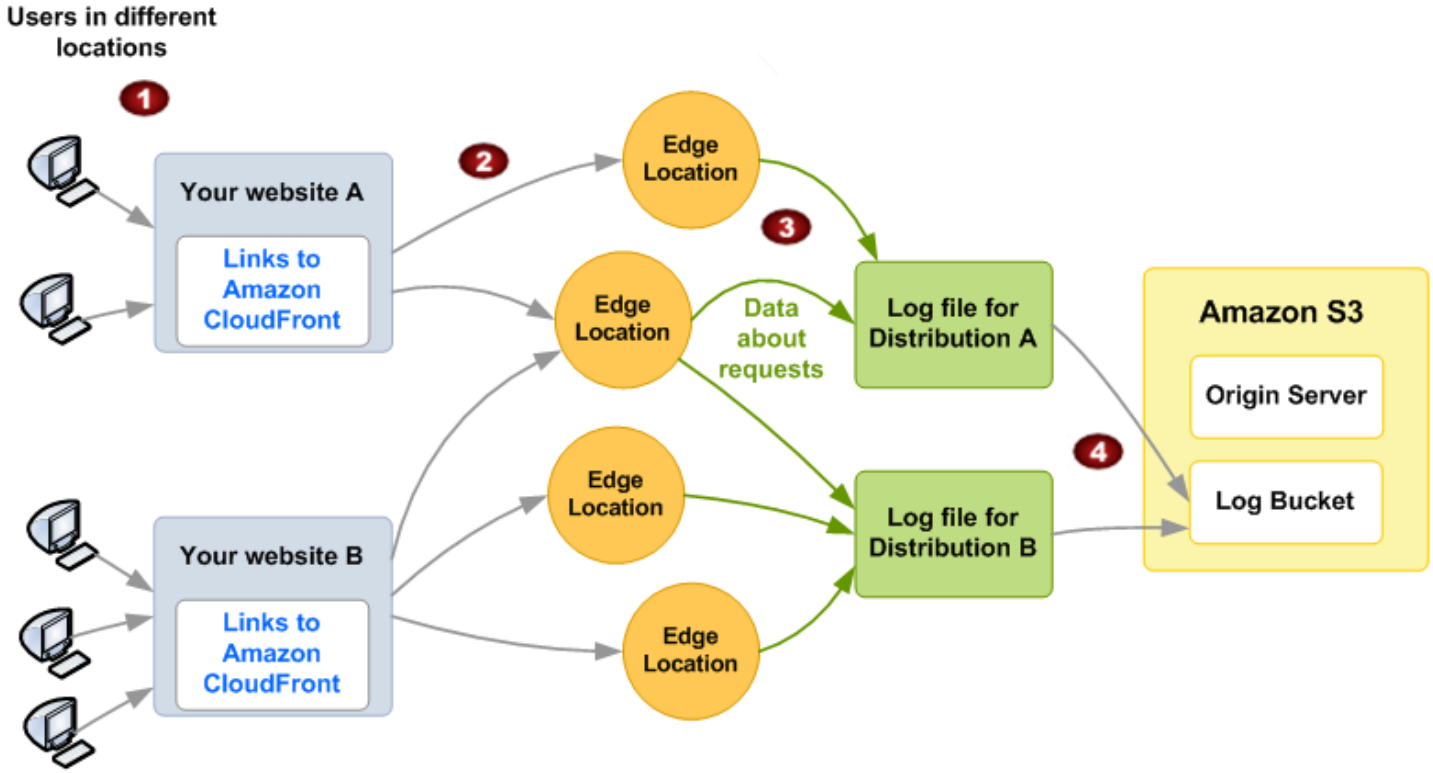
- Standardprotokollierung (Zugriffsprotokolle)
- Verwenden Sie Echtzeitprotokolle
- Protokolle für Edge-Funktionen
- Protokollieren Amazon CloudFront Amazon-API-Aufrufen mit AWS CloudTrail

Standardprotokollierung (Zugriffsprotokolle)

Sie können so konfigurieren CloudFront , dass Protokolldateien erstellt werden, die detaillierte Informationen zu jeder CloudFront empfangenen Benutzer- (Zuschauer-) Anfrage enthalten. Diese werden als Standardprotokolle, oder auch als Zugriffsprotokolle bezeichnet.

Jedes Protokoll enthält Informationen wie die Uhrzeit, zu der die Anfrage empfangen wurde, die Verarbeitungszeit, die Anforderungspfade und die Serverantworten. Sie können diese Zugriffsprotokolle verwenden, um Antwortzeiten zu analysieren und Probleme zu beheben.

Das folgende Diagramm zeigt, wie Informationen über Anfragen für Ihre Objekte CloudFront protokolliert werden. In diesem Beispiel sind die Distributionen so konfiguriert, dass sie Zugriffsprotokolle an einen Amazon S3 S3-Bucket senden.



1. In diesem Beispiel haben Sie zwei Websites, A und B, und zwei entsprechende CloudFront Distributionen. Benutzer fordern Ihre Objekte an URLs , die mit Ihren Distributionen verknüpft sind.
2. CloudFront leitet jede Anfrage an den entsprechenden Edge-Standort weiter.
3. CloudFront schreibt Daten zu jeder Anfrage in eine Protokolldatei, die für diese Verteilung spezifisch ist. In diesem Beispiel werden Informationen über Anfragen, die sich auf Verteilung A beziehen, in eine Protokolldatei für Verteilung A aufgenommen. Informationen zu Anfragen, die sich auf Verteilung B beziehen, werden in eine Protokolldatei für Verteilung B aufgenommen.
4. CloudFront speichert regelmäßig die Protokolldatei für eine Distribution im Amazon S3 S3-Bucket, den Sie bei der Aktivierung der Protokollierung angegeben haben. CloudFront beginnt dann, Informationen über nachfolgende Anfragen in einer neuen Protokolldatei für die Verteilung zu speichern.

   Wenn Zuschauer während einer bestimmten Stunde nicht auf Ihre Inhalte zugreifen, erhalten Sie für diese Stunde keine Protokolldateien.

> **Note**
>
> Wir empfehlen dir, die Protokolle zu verwenden, um die Art der Anfragen nach deinen Inhalten nachzuvollziehen, und nicht, um alle Anfragen vollständig zu erfassen. CloudFront

stellt Zugriffsprotokolle nach bestem Wissen und Gewissen bereit. Der Protokolleintrag für eine bestimmte Anfrage wird möglicherweise viel später übermittelt, als die Anfrage tatsächlich verarbeitet wurde; in seltenen Fällen kann es auch sein, dass ein Protokolleintrag gar nicht übermittelt wird. Wenn ein Protokolleintrag nicht in den Zugriffsprotokollen enthalten ist, stimmt die Anzahl der Einträge in den Zugriffsprotokollen nicht mit deren Anzahl in den Abrechnungs- und Nutzungsberichten für AWS überein.

CloudFront unterstützt zwei Versionen der Standardprotokollierung. Die Standardprotokollierung (Legacy) unterstützt nur das Senden Ihrer Zugriffsprotokolle an Amazon S3. Die Standardprotokollierung (v2) unterstützt zusätzliche Lieferziele. Sie können beide oder eine der beiden Protokollierungsoptionen für Ihre Distribution konfigurieren. Weitere Informationen finden Sie unter den folgenden Themen:

Themen

- Konfigurieren Sie die Standardprotokollierung (v2)
- Standardprotokollierung konfigurieren (Legacy)
- Referenz zur Standardprotokollierung

Tip

CloudFront bietet auch Echtzeitprotokolle, die Ihnen Informationen über Anfragen an eine Distribution in Echtzeit liefern (Protokolle werden innerhalb von Sekunden nach Eingang der Anfragen zugestellt). Sie können Echtzeitprotokolle verwenden, um basierend auf der Leistung der Bereitstellung von Inhalten Überwachungsaktionen und Analysen auszuführen und Maßnahmen zu ergreifen. Weitere Informationen finden Sie unter Verwenden Sie Echtzeitprotokolle.

Konfigurieren Sie die Standardprotokollierung (v2)

Sie können die Standardprotokollierung aktivieren, wenn Sie eine Distribution erstellen oder aktualisieren. Die Standardprotokollierung (v2) umfasst die folgenden Funktionen:

- Senden Sie Zugriffsprotokolle an Amazon CloudWatch Logs, Amazon Data Firehose und Amazon Simple Storage Service (Amazon S3).

- Wählen Sie die gewünschten Protokollfelder aus. Sie können auch eine Teilmenge von Echtzeit-Protokollfeldern auswählen.
- Wählen Sie zusätzliche Ausgabe-Protokolldateiformate aus.

Wenn Sie Amazon S3 verwenden, stehen Ihnen die folgenden optionalen Funktionen zur Verfügung:

- Senden Sie Protokolle, um sich anzumelden AWS-Regionen.
- Organisieren Sie Ihre Logs mit Partitionierung.
- Aktivieren Sie Hive-kompatible Dateinamen.

Weitere Informationen finden Sie unter Protokolle an Amazon S3 senden.

Gehen Sie wie folgt vor, um mit der Standardprotokollierung zu beginnen:

1. Richten Sie Ihre erforderlichen Berechtigungen für die angegebenen Personen ein AWS-Service , die Ihre Protokolle erhalten sollen.
2. Konfigurieren Sie die Standardprotokollierung CloudFront über die Konsole oder die CloudWatch API.
3. Sehen Sie sich Ihre Zugriffsprotokolle an.

> **Note**
>
> - Wenn Sie die Standardprotokollierung (v2) aktivieren, hat dies keine Auswirkungen auf die Standardprotokollierung (Legacy) oder ändert sie nicht. Sie können zusätzlich zur Standardprotokollierung (v2) weiterhin die Standardprotokollierung (Legacy) für Ihre Distribution verwenden. Weitere Informationen finden Sie unter Standardprotokollierung konfigurieren (Legacy).
> - Wenn Sie die Standardprotokollierung (Legacy) bereits aktiviert haben und die Standardprotokollierung (v2) für Amazon S3 aktivieren möchten, empfehlen wir, dass Sie einen anderen Amazon S3 S3-Bucket angeben oder einen separaten Pfad in demselben Bucket verwenden (z. B. ein Protokollpräfix oder eine Partitionierung verwenden). Auf diese Weise behalten Sie den Überblick darüber, welche Protokolldateien mit welcher Distribution verknüpft sind, und verhindert, dass sich Protokolldateien gegenseitig überschreiben.

Berechtigungen

CloudFront verwendet CloudWatch ausgelieferte Protokolle, um Zugriffsprotokolle zu übermitteln. Dazu benötigen Sie Berechtigungen für die angegebenen Daten, AWS-Service sodass Sie die Übermittlung von Protokollen aktivieren können.

Um die erforderlichen Berechtigungen für jedes Protokollierungsziel zu sehen, wählen Sie eines der folgenden Themen im Amazon CloudWatch Logs-Benutzerhandbuch aus.

- CloudWatch Protokolle
- Firehose
- Amazon S3

Nachdem Sie die Berechtigungen für Ihr Protokollierungsziel eingerichtet haben, können Sie die Standardprotokollierung für Ihre Distribution aktivieren.

> **Note**
>
> CloudFront unterstützt das Senden von Zugriffsprotokollen an verschiedene Konten AWS-Konten (Cross-Accounts). Um die kontoübergreifende Zustellung zu ermöglichen, müssen beide Konten (Ihr Konto und das Empfängerkonto) über die erforderlichen Berechtigungen verfügen. Weitere Informationen finden Sie im Aktivieren Sie die Standardprotokollierung für die kontoübergreifende Übermittlung Abschnitt oder im Beispiel für die kontoübergreifende Lieferung im Amazon CloudWatch Logs-Benutzerhandbuch.

Aktivieren Sie die Standardprotokollierung

Um die Standardprotokollierung zu aktivieren, können Sie die CloudFront Konsole oder die CloudWatch API verwenden.

Inhalt

Aktivieren Sie die Standardprotokollierung (CloudFrontKonsole)

Um die Standardprotokollierung für eine CloudFront Distribution zu aktivieren (Konsole)

1. Verwenden Sie die CloudFront Konsole, um eine bestehende Distribution zu aktualisieren.
2. Wählen Sie die Registerkarte Logging (Protokollierung) aus.
3. Wählen Sie Hinzufügen und anschließend den Dienst aus, der Ihre Logs erhalten soll:
   - CloudWatch Logs
   - Firehose
   - Amazon S3
4. Wählen Sie für das Ziel die Ressource für Ihren Service aus. Wenn Sie Ihre Ressource noch nicht erstellt haben, können Sie Erstellen wählen oder sich die folgende Dokumentation ansehen.
   - Geben Sie für CloudWatch Logs den Namen der Protokollgruppe ein.
   - Geben Sie für Firehose den Firehose-Lieferstream ein.
   - Geben Sie für Amazon S3 den Bucket-Namen ein.

   > **Tip**
   >
   > Um ein Präfix anzugeben, geben Sie das Präfix nach dem Bucket-Namen ein, z. `amzn-s3-demo-bucket.s3.amazonaws.com/MyLogPrefix` B. Wenn Sie kein Präfix angeben, CloudFront wird automatisch eines für Sie hinzugefügt. Weitere Informationen finden Sie unter Protokolle an Amazon S3 senden.

5. Für zusätzliche Einstellungen — optional — können Sie die folgenden Optionen angeben:
   a. Wählen Sie unter Feldauswahl die Namen der Protokollfelder aus, die Sie an Ihr Ziel senden möchten. Sie können Zugriffsprotokollfelder und eine Teilmenge von Echtzeit-Protokollfeldern auswählen.
   b. (Nur Amazon S3) Geben Sie für die Partitionierung den Pfad zur Partitionierung Ihrer Protokolldateidaten an.
   c. (Nur Amazon S3) Für ein Hive-kompatibles Dateiformat können Sie das Kontrollkästchen aktivieren, um Hive-kompatible S3-Pfade zu verwenden. Dies vereinfacht das Laden neuer Daten in Ihre Hive-kompatiblen Tools.
   d. Geben Sie unter Ausgabeformat Ihr bevorzugtes Format an.

> **Note**
>
> Wenn Sie Parquet wählen, fallen für diese Option CloudWatch Gebühren für die Konvertierung Ihrer Zugriffsprotokolle nach Apache Parquet an. Weitere Informationen zur Preisgestaltung finden Sie im Abschnitt Vending Logs. CloudWatch

e. Geben Sie unter Feldtrennzeichen an, wie Protokollfelder getrennt werden sollen.

6. Führen Sie die Schritte aus, um Ihre Distribution zu aktualisieren oder zu erstellen.
7. Um ein weiteres Ziel hinzuzufügen, wiederholen Sie die Schritte 3—6.
8. Vergewissern Sie sich auf der Seite Protokolle, dass der Status der Standardprotokolle neben der Verteilung aktiviert lautet.
9. (Optional) Um die Cookie-Protokollierung zu aktivieren, wählen Sie „Verwalten“, „Einstellungen“ und aktivieren Sie die Cookie-Protokollierung. Wählen Sie dann „Änderungen speichern“.

> **Tip**
>
> Die Cookie-Protokollierung ist eine globale Einstellung, die für die gesamte Standardprotokollierung für Ihre Distribution gilt. Sie können diese Einstellung nicht für separate Lieferziele überschreiben.

Weitere Informationen zu den standardmäßigen Versand- und Protokollfeldern für die Protokollierung finden Sie unterReferenz zur Standardprotokollierung.

Aktivieren Sie die Standardprotokollierung (CloudWatchAPI)

Sie können die CloudWatch API auch verwenden, um die Standardprotokollierung für Ihre Distributionen zu aktivieren.

> **Hinweise**
>
> - Wenn Sie die CloudWatch API aufrufen, um die Standardprotokollierung zu aktivieren, müssen Sie die Region USA Ost (Nord-Virginia) (`us-east-1`) angeben, auch wenn Sie die regionsübergreifende Zustellung an ein anderes Ziel aktivieren möchten. Wenn Sie beispielsweise Ihre Zugriffsprotokolle an einen S3-Bucket in der Region Europa (Irland)

() senden möchten, verwenden Sie die CloudWatch API in der `us-east-1` Region. `eu-west-1`

- Es gibt eine zusätzliche Option, um Cookies in die Standardprotokollierung aufzunehmen. In der CloudFront API ist dies der `IncludeCookies` Parameter. Wenn Sie die Zugriffsprotokollierung mithilfe der CloudWatch API konfigurieren und angeben, dass Sie Cookies einbeziehen möchten, müssen Sie die CloudFront Konsole oder CloudFront API verwenden, um Ihre Distribution so zu aktualisieren, dass sie Cookies enthält. Andernfalls CloudFront können keine Cookies an Ihr Protokollziel gesendet werden. Weitere Informationen finden Sie unter Protokollierung von Cookies.

Um die Standardprotokollierung für eine Distribution (CloudWatch API) zu aktivieren

1. Nachdem Sie eine Distribution erstellt haben, rufen Sie den Amazon-Ressourcennamen (ARN) ab.

   Sie finden den ARN auf der Verteilungsseite in der CloudFront Konsole oder Sie können den GetDistributionAPI-Vorgang verwenden. Ein Distribution-ARN folgt dem Format: `arn:aws:cloudfront::123456789012:distribution/d111111abcdef8`
2. Verwenden Sie als Nächstes den CloudWatch PutDeliverySourceAPI-Vorgang, um eine Lieferquelle für die Verteilung zu erstellen.

   a. Geben Sie einen Namen für die Lieferquelle ein.

   b. Übergeben `resourceArn` Sie die Verteilung.

   c. Geben Sie für `logType` `ACCESS_LOGS` den Typ der Protokolle an, die gesammelt werden.

   d. Example Beispiel für einen AWS CLI put-delivery-source Befehl

      Im Folgenden finden Sie ein Beispiel für die Konfiguration einer Zustellungsquelle für eine Distribution.

      ```
      aws logs put-delivery-source --name S3-delivery --resource-arn
       arn:aws:cloudfront::123456789012:distribution/d111111abcdef8 --log-type
       ACCESS_LOGS
      ```

      Ausgabe

      ```
      {
       "deliverySource": {
      ```

```
 "name": "S3-delivery",
 "arn": "arn:aws:logs:us-east-1:123456789012:delivery-source:S3-delivery",
 "resourceArns": [
 "arn:aws:cloudfront::123456789012:distribution/d111111abcdef8"
 ],
 "service": "cloudfront",
 "logType": "ACCESS_LOGS"
 }
}
```

3. Verwenden Sie den PutDeliveryDestinationAPI-Vorgang, um zu konfigurieren, wo Ihre Protokolle gespeichert werden sollen.

   a. Geben Sie für `destinationResourceArn` den ARN des Ziels an. Dies kann eine CloudWatch Logs-Protokollgruppe, ein Firehose-Lieferstream oder ein Amazon S3 S3-Bucket sein.

   b. Geben Sie für `outputFormat` das Ausgabeformat für Ihre Logs an.

   c. Example Beispiel für einen AWS CLI put-delivery-destination Befehl

      Im Folgenden finden Sie ein Beispiel für die Konfiguration eines Lieferziels für einen Amazon S3 S3-Bucket.

      ```
      aws logs put-delivery-destination --name S3-destination --delivery-destination-
      configuration destinationResourceArn=arn:aws:s3:::amzn-s3-demo-bucket
      ```

      Ausgabe

      ```
      {
          "name": "S3-destination",
          "arn": "arn:aws:logs:us-east-1:123456789012:delivery-destination:S3-
      destination",
          "deliveryDestinationType": "S3",
          "deliveryDestinationConfiguration": {
              "destinationResourceArn": "arn:aws:s3:::amzn-s3-demo-bucket"
          }
      }
      ```

> **Note**
>
> Wenn Sie Protokolle kontoübergreifend bereitstellen, müssen Sie den PutDeliveryDestinationPolicyAPI-Vorgang verwenden, um dem Zielkonto eine AWS Identity and Access Management (IAM-) Richtlinie zuzuweisen. Die IAM-Richtlinie ermöglicht die Übertragung von einem Konto zu einem anderen Konto.

4. Verwenden Sie den CreateDeliveryAPI-Vorgang, um die Lieferquelle mit dem Ziel zu verknüpfen, das Sie in den vorherigen Schritten erstellt haben. Dieser API-Vorgang verknüpft die Lieferquelle mit dem Endziel.

   a. Geben Sie für `deliverySourceName` den Quellnamen an.

   b. Geben Sie für `deliveryDestinationArn` den ARN für das Lieferziel an.

   c. Geben Sie für `fieldDelimiter` die einzelnen Protokollfelder die Zeichenfolge an.

   d. Geben Sie für `recordFields` die gewünschten Protokollfelder an.

   e. Wenn Sie S3 verwenden, geben Sie an, ob `enableHiveCompatiblePath` und verwendet werden sollsuffixPath.

   Example Beispiel für einen AWS CLI Create-Delivery-Befehl

   Im Folgenden finden Sie ein Beispiel für die Erstellung einer Lieferung.

```
aws logs create-delivery --delivery-source-name cf-delivery --delivery-destination-
arn arn:aws:logs:us-east-1:123456789012:delivery-destination:S3-destination
```

   Ausgabe

```
{
    "id": "abcNegnBoTR123",
    "arn": "arn:aws:logs:us-east-1:123456789012:delivery:abcNegnBoTR123",
    "deliverySourceName": "cf-delivery",
    "deliveryDestinationArn": "arn:aws:logs:us-east-1:123456789012:delivery-
destination:S3-destination",
    "deliveryDestinationType": "S3",
    "recordFields": [
        "date",
        "time",
```

```
        "x-edge-location",
        "sc-bytes",
        "c-ip",
        "cs-method",
        "cs(Host)",
        "cs-uri-stem",
        "sc-status",
        "cs(Referer)",
        "cs(User-Agent)",
        "cs-uri-query",
        "cs(Cookie)",
        "x-edge-result-type",
        "x-edge-request-id",
        "x-host-header",
        "cs-protocol",
        "cs-bytes",
        "time-taken",
        "x-forwarded-for",
        "ssl-protocol",
        "ssl-cipher",
        "x-edge-response-result-type",
        "cs-protocol-version",
        "fle-status",
        "fle-encrypted-fields",
        "c-port",
        "time-to-first-byte",
        "x-edge-detailed-result-type",
        "sc-content-type",
        "sc-content-len",
        "sc-range-start",
        "sc-range-end",
        "c-country",
        "cache-behavior-path-pattern"
    ],
     "fieldDelimiter": ""
}
```

5. Vergewissern Sie sich in der CloudFront Konsole auf der Seite Protokolle, dass der Standardprotokollstatus neben der Verteilung auf Aktiviert steht.

   Weitere Informationen zur standardmäßigen Übermittlung der Protokollierung und zu den Protokollfeldern finden Sie unterReferenz zur Standardprotokollierung.

> **Note**
>
> Um die Standardprotokollierung (v2) für die CloudFront Verwendung zu aktivieren AWS CloudFormation, können Sie die folgenden CloudWatch Protokolleigenschaften verwenden:
>
> - Lieferung
> - DeliveryDestination
> - DeliverySource
>
> Das `ResourceArn` ist die CloudFront Distribution und `LogType` muss `ACCESS_LOGS` dem unterstützten Protokolltyp entsprechen.

Aktivieren Sie die Standardprotokollierung für die kontoübergreifende Übermittlung

Wenn Sie die Standardprotokollierung für Ihr Konto aktivieren AWS-Konto und Ihre Zugriffsprotokolle an ein anderes Konto senden möchten, stellen Sie sicher, dass Sie das Quellkonto und das Zielkonto korrekt konfigurieren. Das Quellkonto mit der CloudFront Verteilung sendet seine Zugriffsprotokolle an das Zielkonto.

In diesem Beispielverfahren sendet das Quellkonto *111111111111* () seine Zugriffsprotokolle an einen Amazon S3 S3-Bucket im Zielkonto (*222222222222*). Um Zugriffsprotokolle an einen Amazon S3 S3-Bucket im Zielkonto zu senden, verwenden Sie den AWS CLI.

Konfigurieren Sie das Zielkonto

Führen Sie für das Zielkonto das folgende Verfahren aus.

Um das Zielkonto zu konfigurieren

1. Um das Ziel für die Protokollzustellung zu erstellen, können Sie den folgenden AWS CLI Befehl eingeben. In diesem Beispiel wird die *MyLogPrefix* Zeichenfolge verwendet, um ein Präfix für Ihre Zugriffsprotokolle zu erstellen.

   ```
   aws logs put-delivery-destination --name cloudfront-delivery-destination --
   delivery-destination-configuration "destinationResourceArn=arn:aws:s3:::amzn-s3-
   demo-bucket-cloudfront-logs/MyLogPrefix"
   ```

   Ausgabe

```
{
    "deliveryDestination": {
        "name": "cloudfront-delivery-destination",
        "arn": "arn:aws:logs:us-east-1:222222222222:delivery-
destination:cloudfront-delivery-destination",
        "deliveryDestinationType": "S3",
        "deliveryDestinationConfiguration": {"destinationResourceArn":
 "arn:aws:s3:::amzn-s3-demo-bucket-cloudfront-logs/MyLogPrefix"}
    }
}
```

> **Note**
>
> Wenn Sie einen S3-Bucket ohne Präfix angeben, CloudFront wird automatisch das `AWSLogs/<account-ID>/CloudFront` als Präfix angehängt, das `suffixPath` im S3-Lieferziel erscheint. Weitere Informationen finden Sie unter S3 DeliveryConfiguration.

2. Fügen Sie die Ressourcenrichtlinie für das Ziel der Protokollzustellung hinzu, damit das Quellkonto eine Protokollzustellung erstellen kann.

   Ersetzen Sie in der folgenden Richtlinie *111111111111* durch die Quellkonto-ID und geben Sie den Lieferziel-ARN aus der Ausgabe in Schritt 1 an.

   JSON

   ```
   {
       "Version": "2012-10-17",
       "Statement": [
           {
               "Sid": "AllowCreateDelivery",
               "Effect": "Allow",
               "Principal": {"AWS": "111111111111"},
               "Action": ["logs:CreateDelivery"],
               "Resource": "arn:aws:logs:us-east-1:222222222222:delivery-
   destination:cloudfront-delivery-destination"
           }
       ]
   }
   ```

3. Speichern Sie die Datei, z. `deliverypolicy.json` B.

4. Geben Sie den folgenden AWS CLI Befehl ein, um die vorherige Richtlinie an das Lieferziel anzuhängen.

```
aws logs put-delivery-destination-policy --delivery-destination-name cloudfront-delivery-destination --delivery-destination-policy file://deliverypolicy.json
```

5. Fügen Sie die folgende Erklärung zur Amazon S3-Ziel-Bucket-Richtlinie hinzu und ersetzen Sie dabei den Ressourcen-ARN und die Quellkonto-ID. Diese Richtlinie ermöglicht es dem `delivery.logs.amazonaws.com` Service Principal, die `s3:PutObject` Aktion durchzuführen.

```
{
    "Sid": "AWSLogsDeliveryWrite",
    "Effect": "Allow",
    "Principal": {"Service": "delivery.logs.amazonaws.com"},
    "Action": "s3:PutObject",
    "Resource": "arn:aws:s3:::amzn-s3-demo-bucket-cloudfront-logs/*",
    "Condition": {
        "StringEquals": {
            "s3:x-amz-acl": "bucket-owner-full-control",
            "aws:SourceAccount": "111111111111"
        },
        "ArnLike": {"aws:SourceArn": "arn:aws:logs:us-east-1:111111111111:delivery-source:*"}
    }
}
```

6. Wenn Sie dies AWS KMS für Ihren Bucket verwenden, fügen Sie der KMS-Schlüsselrichtlinie die folgende Anweisung hinzu, um dem `delivery.logs.amazonaws.com` Dienstprinzipal Berechtigungen zu erteilen.

```
{
    "Sid": "Allow Logs Delivery to use the key",
    "Effect": "Allow",
    "Principal": {"Service": "delivery.logs.amazonaws.com"},
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
```

```
    "Resource": "*",
    "Condition": {
        "StringEquals": {"aws:SourceAccount": "111111111111"},
        "ArnLike": {"aws:SourceArn": "arn:aws:logs:us-east-1:111111111111:delivery-
source:*"}
    }
}
```

Konfigurieren Sie das Quellkonto

Gehen Sie nach der Konfiguration des Zielkontos wie folgt vor, um die Zustellungsquelle zu erstellen und die Protokollierung für die Verteilung im Quellkonto zu aktivieren.

Um das Quellkonto zu konfigurieren

1. Erstellen Sie eine Übermittlungsquelle für die CloudFront Standardprotokollierung, sodass Sie Protokolldateien an CloudWatch Logs senden können.

   Sie können den folgenden AWS CLI Befehl eingeben und dabei den Namen und den ARN Ihrer Distribution ersetzen.

   ```
   aws logs put-delivery-source --name s3-cf-delivery --resource-arn
    arn:aws:cloudfront::111111111111:distribution/E1TR1RHV123ABC --log-type
    ACCESS_LOGS
   ```

   Ausgabe

   ```
   {
       "deliverySource": {
           "name": "s3-cf-delivery",
           "arn": "arn:aws:logs:us-east-1:111111111111:delivery-source:s3-cf-
   delivery",
           "resourceArns":
    ["arn:aws:cloudfront::111111111111:distribution/E1TR1RHV123ABC"],
           "service": "cloudfront",
           "logType": "ACCESS_LOGS"
       }
   }
   ```

2. Erstellen Sie eine Lieferung, um die Protokollzustellungsquelle des Quellkontos und das Protokollübermittlungsziel des Zielkontos zuzuordnen.

Geben Sie im folgenden AWS CLI Befehl den Lieferziel-ARN aus der Ausgabe in Schritt 1: Zielkonto konfigurieren an.

```
aws logs create-delivery --delivery-source-name s3-cf-delivery --
delivery-destination-arn arn:aws:logs:us-east-1:222222222222:delivery-
destination:cloudfront-delivery-destination
```

Ausgabe

```
{
    "delivery": {
        "id": "OPmOpLahVzhx1234",
        "arn": "arn:aws:logs:us-east-1:111111111111:delivery:OPmOpLahVzhx1234",
        "deliverySourceName": "s3-cf-delivery",
        "deliveryDestinationArn": "arn:aws:logs:us-east-1:222222222222:delivery-
destination:cloudfront-delivery-destination",
        "deliveryDestinationType": "S3",
        "recordFields": [
            "date",
            "time",
            "x-edge-location",
            "sc-bytes",
            "c-ip",
            "cs-method",
            "cs(Host)",
            "cs-uri-stem",
            "sc-status",
            "cs(Referer)",
            "cs(User-Agent)",
            "cs-uri-query",
            "cs(Cookie)",
            "x-edge-result-type",
            "x-edge-request-id",
            "x-host-header",
            "cs-protocol",
            "cs-bytes",
            "time-taken",
            "x-forwarded-for",
            "ssl-protocol",
            "ssl-cipher",
            "x-edge-response-result-type",
            "cs-protocol-version",
```

```
            "fle-status",
            "fle-encrypted-fields",
            "c-port",
            "time-to-first-byte",
            "x-edge-detailed-result-type",
            "sc-content-type",
            "sc-content-len",
            "sc-range-start",
            "sc-range-end",
            "c-country",
            "cache-behavior-path-pattern"
        ],
        "fieldDelimiter": "\t"
    }
}
```

3. Stellen Sie sicher, dass Ihre kontoübergreifende Lieferung erfolgreich ist.

    a. Melden Sie sich *`source`* über das Konto bei der CloudFront Konsole an und wählen Sie Ihre Distribution aus. Auf der Registerkarte Protokollierung finden Sie unter Typ einen Eintrag, der für die kontoübergreifende S3-Protokollzustellung erstellt wurde.

    b. Melden Sie sich *`destination`* über das Konto bei der Amazon S3-Konsole an und wählen Sie Ihren Amazon S3 S3-Bucket aus. Sie werden das Präfix *`MyLogPrefix`* im Bucket-Namen und in allen Zugriffsprotokollen sehen, die in diesen Ordner gesendet wurden.

Format der Ausgabedatei

Abhängig vom ausgewählten Lieferziel können Sie eines der folgenden Formate für Protokolldateien angeben:

- JSON
- Einfach
- w3c
- Raw
- Parkett (nur Amazon S3)

> **Note**
>
> Sie können das Ausgabeformat nur festlegen, wenn Sie das Lieferziel zum ersten Mal erstellen. Diese kann später nicht mehr aktualisiert werden. Um das Ausgabeformat zu ändern, löschen Sie die Lieferung und erstellen Sie ein neues.

Weitere Informationen finden Sie PutDeliveryDestinationin der Amazon CloudWatch Logs API-Referenz.

Bearbeiten Sie die Standard-Logging-Einstellungen

Sie können die Protokollierung mithilfe der CloudFront Konsole oder der CloudWatch API aktivieren oder deaktivieren und andere Protokolleinstellungen aktualisieren. Ihre Änderungen der Protokollierungseinstellungen werden innerhalb von 12 Stunden wirksam.

Weitere Informationen finden Sie unter den folgenden Themen:

- Informationen zum Aktualisieren einer Distribution mithilfe der CloudFront Konsole finden Sie unterEine Verteilung aktualisieren.
- Informationen zum Aktualisieren einer Distribution mithilfe der CloudFront API finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.
- Weitere Informationen zu CloudWatch Logs-API-Vorgängen finden Sie in der Amazon CloudWatch Logs API-Referenz.

Greifen Sie auf Protokollfelder zu

Sie können dieselben Protokollfelder auswählen, die die Standardprotokollierung (Legacy) unterstützt. Weitere Informationen finden Sie unter Felder für Protokolldateien.

Darüber hinaus können Sie die folgenden Echtzeit-Protokollfelder auswählen.

1. **`timestamp(ms)`**— Zeitstempel in Millisekunden.
2. **`origin-fbl`**— Die Anzahl der Sekunden der Latenz im ersten Byte zwischen und Ihrem Ursprung. CloudFront
3. **`origin-lbl`**— Die Anzahl der Sekunden der Latenz im letzten Byte zwischen CloudFront und Ihrem Ursprung.
4. **`asn`**— Die autonome Systemnummer (ASN) des Betrachters.

5. **`c-country`**— Eine Landesvorwahl, die den geografischen Standort des Betrachters angibt, der durch die IP-Adresse des Betrachters bestimmt wird. Die Liste der Ländercodes finden Sie unter ISO 3166-1 alpha-2.
6. **`cache-behavior-path-pattern`**— Das Pfadmuster, das das Cache-Verhalten identifiziert, das der Zuschaueranfrage entsprach.

Sendet Logs an CloudWatch Logs

Um Logs an CloudWatch Logs zu senden, erstellen oder verwenden Sie eine bestehende CloudWatch Logs-Protokollgruppe. Weitere Informationen zur Konfiguration einer CloudWatch Logs-Log-Gruppe finden Sie unter Arbeiten mit Log-Gruppen und Log-Streams.

Nachdem Sie Ihre Protokollgruppe erstellt haben, müssen Sie über die erforderlichen Berechtigungen verfügen, um die Standardprotokollierung zuzulassen. Weitere Informationen zu den erforderlichen Berechtigungen finden Sie unter An Logs sent to CloudWatch Logs im Amazon CloudWatch Logs-Benutzerhandbuch.

> **Hinweise**
>
> - Wenn Sie den Namen der CloudWatch Logs-Protokollgruppe angeben, verwenden Sie nur das Regex-Muster. `[\w-]` Weitere Informationen zur PutDeliveryDestinationAPI-Operation finden Sie in der Amazon CloudWatch Logs API-Referenz.
> - Stellen Sie sicher, dass Ihre Ressourcenrichtlinie für Protokollgruppen die Größenbeschränkung nicht überschreitet. Weitere Informationen finden Sie im Abschnitt Überlegungen zur Größenbeschränkung für die Ressourcenrichtlinie für Protokollgruppen im Thema CloudWatch Protokolle.

Beispiel für ein Zugriffsprotokoll, das an CloudWatch Logs gesendet wurde

```
{
"date": "2024-11-14",
"time": "21:34:06",
"x-edge-location": "SOF50-P2",
"asn": "16509",
"timestamp(ms)": "1731620046814",
"origin-fbl": "0.251",
"origin-lbl": "0.251",
```

```
"x-host-header": "d111111abcdef8.cloudfront.net",
"cs(Cookie)": "examplecookie=value"
}
```

Logs an Firehose senden

Um Protokolle an Firehose zu senden, erstellen oder verwenden Sie einen vorhandenen Firehose-Lieferstream.

Informationen zur Erstellung Ihres Lieferstreams finden Sie unter Erstellen eines Amazon Data Firehose-Lieferdatenstroms.

Nachdem Sie Ihren Lieferstream erstellt haben, müssen Sie über die erforderlichen Berechtigungen verfügen, um die Standardprotokollierung zu ermöglichen. Weitere Informationen finden Sie unter An Firehose gesendete Logs im Amazon CloudWatch Logs-Benutzerhandbuch.

> **Note**
>
> Wenn Sie den Namen des Firehose-Streams angeben, verwenden Sie nur das Regex-Muster. `[\w-]` Weitere Informationen zur PutDeliveryDestinationAPI-Operation finden Sie in der Amazon CloudWatch Logs API-Referenz.

Beispiel für ein Zugriffsprotokoll, das an Firehose gesendet wurde

```
{"date":"2024-11-15","time":"19:45:51","x-edge-location":"SOF50-
P2","asn":"16509","timestamp(ms)":"1731699951183","origin-fbl":"0.254","origin-
lbl":"0.254","x-host-
header":"d111111abcdef8.cloudfront.net","cs(Cookie)":"examplecookie=value"}
{"date":"2024-11-15","time":"19:45:52","x-edge-location":"SOF50-
P2","asn":"16509","timestamp(ms)":"1731699952950","origin-fbl":"0.125","origin-
lbl":"0.125","x-host-
header":"d111111abcdef8.cloudfront.net","cs(Cookie)":"examplecookie=value"}
```

Protokolle an Amazon S3 senden

Um Ihre Zugriffsprotokolle an Amazon S3 zu senden, erstellen oder verwenden Sie einen vorhandenen S3-Bucket. Wenn Sie die Anmeldung aktivieren CloudFront, geben Sie den Bucket-Namen an. Informationen zum Erstellen eines Buckets finden Sie unter Bucket erstellen im Amazon Simple Storage Service-Benutzerhandbuch.

Nachdem Sie Ihren Bucket erstellt haben, müssen Sie über die erforderlichen Berechtigungen verfügen, um die Standardprotokollierung zu ermöglichen. Weitere Informationen finden Sie unter An Amazon S3 gesendete Logs im Amazon CloudWatch Logs-Benutzerhandbuch.

- Nachdem Sie die Protokollierung aktiviert haben, AWS werden automatisch die erforderlichen Bucket-Richtlinien für Sie hinzugefügt.
- Sie können auch S3-Buckets im AWS-Regionen Opt-In verwenden.

> **Note**
>
> Wenn Sie die Standardprotokollierung (Legacy) bereits aktiviert haben und die Standardprotokollierung (v2) für Amazon S3 aktivieren möchten, empfehlen wir, dass Sie einen anderen Amazon S3 S3-Bucket angeben oder einen separaten Pfad in demselben Bucket verwenden (z. B. ein Protokollpräfix oder eine Partitionierung verwenden). Auf diese Weise behalten Sie den Überblick darüber, welche Protokolldateien mit welcher Distribution verknüpft sind, und verhindert, dass sich Protokolldateien gegenseitig überschreiben.

Themen

Geben Sie einen S3-Bucket an

Beachten Sie Folgendes, wenn Sie einen S3-Bucket als Lieferziel angeben.

Der S3-Bucket-Name kann nur das Regex-Muster verwenden. `[\w-]` Weitere Informationen zur PutDeliveryDestinationAPI-Operation finden Sie in der Amazon CloudWatch Logs API-Referenz.

Wenn Sie ein Präfix für Ihren S3-Bucket angegeben haben, werden Ihre Logs unter diesem Pfad angezeigt. Wenn Sie kein Präfix angeben, CloudFront wird das `AWSLogs/{account-id}/CloudFront` Präfix automatisch für Sie angehängt.

Weitere Informationen finden Sie unter Beispielpfade für den Zugriff auf Logs.

Partitionierung

Sie können die Partitionierung verwenden, um Ihre Zugriffsprotokolle zu organisieren, wenn sie CloudFront an Ihren S3-Bucket gesendet werden. Auf diese Weise können Sie Ihre Zugriffsprotokolle auf der Grundlage des gewünschten Pfads organisieren und lokalisieren.

Sie können die folgenden Variablen verwenden, um einen Ordnerpfad zu erstellen.

- `{DistributionId}` oder `{distributionid}`
- `{yyyy}`
- `{MM}`
- `{dd}`
- `{HH}`
- `{accountid}`

Sie können eine beliebige Anzahl von Variablen verwenden und Ordnernamen in Ihrem Pfad angeben. CloudFrontverwendet dann diesen Pfad, um eine Ordnerstruktur für Sie im S3-Bucket zu erstellen.

Beispiele

- *my_distribution_log_data*/{DistributionId}/*logs*
- /*cloudfront*/{DistributionId}/*my_distribution_log_data*/{yyyy}/{MM}/{dd}/{HH}/*logs*

> **Note**
>
> Sie können eine der beiden Variablen als Distributions-ID im Suffixpfad verwenden. Wenn Sie jedoch Zugriffsprotokolle an senden, müssen Sie die `{distributionid}` Variable verwenden AWS Glue, da AWS Glue erwartet wird, dass Partitionsnamen in Kleinbuchstaben geschrieben werden. Aktualisieren Sie Ihre bestehende Protokollkonfiguration, um CloudFront sie durch zu `{DistributionId}` ersetzen`{distributionid}`.

Hive-kompatibles Dateinamenformat

Sie können diese Option verwenden, damit S3-Objekte, die zugestellte Zugriffsprotokolle enthalten, eine Präfixstruktur verwenden, die die Integration mit Apache Hive ermöglicht. Weitere Informationen finden Sie unter CreateDelivery-API-Operation.

Example Beispiel

```
/cloudfront/DistributionId={DistributionId}/my_distribution_log_data/year={yyyy}/
month={MM}/day={dd}/hour={HH}/logs
```

Weitere Informationen zur Partitionierung und zu den Hive-kompatiblen Optionen finden Sie unter dem DeliveryConfigurationS3-Element in der Amazon CloudWatch Logs API-Referenz.

Beispielpfade für den Zugriff auf Logs

Wenn Sie einen S3-Bucket als Ziel angeben, können Sie die folgenden Optionen verwenden, um den Pfad zu Ihren Zugriffsprotokollen zu erstellen:

- Ein Amazon S3 S3-Bucket mit oder ohne Präfix
- Partitionierung, indem Sie eine CloudFront bereitgestellte Variable verwenden oder eine eigene eingeben
- Aktivierung der HIVE-kompatiblen Option

Die folgenden Tabellen zeigen, wie Ihre Zugriffsprotokolle in Ihrem Bucket angezeigt werden, abhängig von den ausgewählten Optionen.

Amazon S3 S3-Bucket mit einem Präfix

Name des Amazon S3 S3-Buckets	Partition, die Sie im Suffixpfad angeben	Der Suffixpfad wurde aktualisiert	Hive-kompatibel aktiviert?	Zugriffsprotokolle werden gesendet an
`amzn-s3-demo-bucket/MyLogPrefix`	Keine	Keine	Nein	`amzn-s3-demo-bucket/MyLogPrefix/`

Name des Amazon S3 S3-Buckets	Partition, die Sie im Suffixpfad angeben	Der Suffixpfad wurde aktualisiert	Hive-kompatibel aktiviert?	Zugriffsprotokolle werden gesendet an
`amzn-s3-demo-bucket/MyLogPrefix`	`myFolderA/`	`myFolderA/`	Nein	`amzn-s3-demo-bucket/MyLogPrefix/myFolderA/`
`amzn-s3-demo-bucket/MyLogPrefix`	`myFolderA/{yyyy}`	`myFolderA/{yyyy}`	Ja	`amzn-s3-demo-bucket/MyLogPrefix/myFolderA/year=2025`

Amazon S3 S3-Bucket ohne Präfix

Name des Amazon S3 S3-Buckets	Partition, die Sie im Suffixpfad angeben	Der Suffixpfad wurde aktualisiert	Hive-kompatibel aktiviert?	Zugriffsprotokolle werden gesendet an
`amzn-s3-demo-bucket`	Keine	`AWSLogs/{account-id}/CloudFront/`	Nein	`amzn-s3-demo-bucket/AWSLogs/` *`<your-account-ID>`* `/CloudFront/`
`amzn-s3-demo-bucket`	`myFolderA/`	`AWSLogs/{account-id}/CloudFront/myFolderA/`	Nein	`amzn-s3-demo-bucket/AWSLogs/` *`<your-account-ID>`* `/`

Name des Amazon S3 S3-Buckets	Partition, die Sie im Suffixpfad angeben	Der Suffixpfad wurde aktualisiert	Hive-kompatibel aktiviert?	Zugriffsprotokolle werden gesendet an
				`CloudFront/myFolderA/`
`amzn-s3-demo-bucket`	`myFolderA/`	`AWSLogs/{account-id}/CloudFront/myFolderA/`	Ja	`amzn-s3-demo-bucket/AWSLogs/aws-account-id=`*`<your-account-ID>`* `/CloudFront/myFolderA/`
`amzn-s3-demo-bucket`	`myFolderA/{yyyy}`	`AWSLogs/{account-id}/CloudFront/myFolderA/{yyyy}`	Ja	`amzn-s3-demo-bucket/AWSLogs/aws-account-id=`*`<your-account-ID>`* `/CloudFront/myFolderA/year=2025`

AWS-Konto ID als Partition

Name des Amazon S3 S3-Buckets	Partition, die Sie im Suffixpfad angeben	Der Suffixpfad wurde aktualisiert	Hive-kompatibel aktiviert?	Zugriffsprotokolle werden gesendet an
`amzn-s3-demo-bucket`	Keine	`AWSLogs/{account-i`	Ja	`amzn-s3-demo-bucket/AWSLogs`

Name des Amazon S3 S3-Buckets	Partition, die Sie im Suffixpfad angeben	Der Suffixpfad wurde aktualisiert	Hive-kompatibel aktiviert?	Zugriffsprotokolle werden gesendet an
		d}/CloudFront/		/aws-account-id=<your-account-ID> /CloudFront/
amzn-s3-demo-bucket	myFolderA/{accountid}	AWSLogs/{account-id}/CloudFront/myFolderA/{accountid}	Ja	amzn-s3-demo-bucket/AWSLogs/aws-account-id=<your-account-ID>/CloudFront/myFolderA/accountid= <your-account-ID>

Hinweise

- Die {account-id} Variable ist reserviert für CloudFront. CloudFrontfügt diese Variable automatisch zu Ihrem Suffixpfad hinzu, wenn Sie einen Amazon S3 S3-Bucket ohne Präfix angeben. Wenn Ihre Protokolle Hive-kompatibel sind, wird diese Variable als angezeigt. aws-account-id
- Sie können die {accountid} Variable verwenden, CloudFront um Ihre Konto-ID zum Suffixpfad hinzuzufügen. Wenn Ihre Protokolle HIVE-kompatibel sind, wird diese Variable als angezeigt. accountid
- Weitere Informationen zum Suffixpfad finden Sie unter S3. DeliveryConfiguration

Beispiel für ein Zugriffsprotokoll, das an Amazon S3 gesendet wurde

```
#Fields: date time x-edge-location asn timestamp(ms) x-host-header cs(Cookie)
2024-11-14    22:30:25    SOF50-P2    16509    1731623425421
d111111abcdef8.cloudfront.net    examplecookie=value2
```

Deaktivieren Sie die Standardprotokollierung

Sie können die Standardprotokollierung für Ihre Distribution deaktivieren, wenn Sie sie nicht mehr benötigen.

Um die Standardprotokollierung zu deaktivieren

1. Melden Sie sich in der CloudFront -Konsole an.
2. Wählen Sie Vertrieb und dann Ihre Vertriebs-ID aus.
3. Wählen Sie Protokollierung und wählen Sie dann unter Standardprotokollziele das Ziel aus.
4. Wählen Sie „Verwalten“ und anschließend „Löschen“.
5. Wiederholen Sie den vorherigen Schritt, wenn Sie mehr als eine Standardprotokollierung haben.

> **Note**
>
> Wenn Sie die Standardprotokollierung aus der CloudFront Konsole löschen, werden durch diese Aktion nur die Lieferung und das Lieferziel gelöscht. Die Lieferquelle wird dadurch nicht aus Ihrer AWS-Konto gelöscht. Um eine Lieferquelle zu löschen, geben Sie den Namen der Lieferquelle im `aws logs delete-delivery-source --name DeliverySourceName` Befehl an. Weitere Informationen finden Sie DeleteDeliverySourcein der Amazon CloudWatch Logs API-Referenz.

Fehlerbehebung

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu beheben, wenn Sie mit der CloudFront Standardprotokollierung (v2) arbeiten.

Die Lieferquelle ist bereits vorhanden

Wenn Sie die Standardprotokollierung für eine Verteilung aktivieren, erstellen Sie eine Zustellungsquelle. Anschließend verwenden Sie diese Versandquelle, um Lieferungen an den

gewünschten Zieltyp zu erstellen: CloudWatch Logs, Firehose, Amazon S3. Derzeit können Sie nur eine Lieferquelle pro Verteilung angeben. Wenn Sie versuchen, eine weitere Zustellungsquelle für dieselbe Verteilung zu erstellen, wird die folgende Fehlermeldung angezeigt.

```
This ResourceId has already been used in another Delivery Source in this account
```

Um eine weitere Zustellungsquelle zu erstellen, löschen Sie zuerst die vorhandene. Weitere Informationen finden Sie DeleteDeliverySourcein der Amazon CloudWatch Logs API-Referenz.

Ich habe den Suffixpfad geändert und der Amazon S3 S3-Bucket kann meine Protokolle nicht empfangen

Wenn Sie die Standardprotokollierung (v2) aktiviert haben und einen Bucket-ARN ohne Präfix angeben, CloudFront wird der folgende Standard an den Suffixpfad angehängt:. `AWSLogs/{account-id}/CloudFront` Wenn Sie die CloudFront Konsole oder den UpdateDeliveryConfigurationAPI-Vorgang verwenden, um einen anderen Suffixpfad anzugeben, müssen Sie die Amazon S3 S3-Bucket-Richtlinie aktualisieren, sodass sie denselben Pfad verwendet.

Example Beispiel: Aktualisierung des Suffixpfads

1. Ihr Standard-Suffixpfad ist `AWSLogs/{account-id}/CloudFront` und Sie ersetzen ihn durch. `myFolderA`
2. Da sich Ihr neuer Suffixpfad von dem in der Amazon S3 S3-Bucket-Richtlinie angegebenen Pfad unterscheidet, werden Ihre Zugriffsprotokolle nicht übermittelt.
3. Sie können einen der folgenden Schritte ausführen:
   - Aktualisieren Sie die Amazon S3 S3-Bucket-Berechtigung von `amzn-s3-demo-bucket/AWSLogs/<your-account-ID>/CloudFront/*` bis`amzn-s3-demo-bucket/myFolderA/*`.
   - Aktualisieren Sie Ihre Logging-Konfiguration, um wieder das Standardsuffix zu verwenden: `AWSLogs/{account-id}/CloudFront`

Weitere Informationen finden Sie unter Berechtigungen.

Löschen Sie die Protokolldateien

CloudFront löscht Protokolldateien nicht automatisch von Ihrem Ziel. Informationen zum Löschen von Protokolldateien finden Sie in den folgenden Themen:

Amazon S3

- Löschen von Objekten im Amazon Simple Storage Service Console-Benutzerhandbuch

CloudWatch Protokolle

- Arbeiten mit Protokollgruppen und Protokollstreams im Amazon CloudWatch Logs-Benutzerhandbuch
- DeleteLogGroupin der Amazon CloudWatch Logs API-Referenz

Firehose

- DeleteDeliveryStreamin der Amazon Data Firehose API-Referenz

Preisgestaltung

CloudFront berechnet keine Gebühren für die Aktivierung von Standardprotokollen. Je nachdem, welches Ziel Sie für die Protokollzustellung auswählen, können jedoch Gebühren für die Lieferung, Aufnahme, Speicherung oder den Zugriff anfallen. Weitere Informationen finden Sie unter Amazon CloudWatch Logs Pricing. Wählen Sie unter Bezahltes Kontingent den Tab Logs aus und sehen Sie sich dann unter Vended Logs die Informationen für jedes Lieferziel an.

Weitere Informationen zu den jeweiligen AWS-Service Preisen finden Sie in den folgenden Themen:

- Amazon CloudWatch Logs — Preise
- Amazon Data Firehose — Preisgestaltung
- Amazon S3 – Preise

> **Note**
>
> Für die Protokollzustellung an Amazon S3 fallen keine zusätzlichen Gebühren an, allerdings fallen Amazon S3 S3-Gebühren für die Speicherung und den Zugriff auf die Protokolldateien an. Wenn Sie die Parquet-Option aktivieren, um Ihre Zugriffsprotokolle in Apache Parquet zu konvertieren, fallen für diese Option Gebühren an. CloudWatch Weitere Informationen zur Preisgestaltung finden Sie im Abschnitt Vending Logs. CloudWatch

Standardprotokollierung konfigurieren (Legacy)

Hinweise

- Dieses Thema bezieht sich auf die vorherige Version der Standardprotokollierung. Informationen zur neuesten Version finden Sie unter Konfigurieren Sie die Standardprotokollierung (v2).
- Wenn Sie die Standardprotokollierung (Legacy) bereits aktiviert haben und die Standardprotokollierung (v2) für Amazon S3 aktivieren möchten, empfehlen wir, dass Sie einen anderen Amazon S3 S3-Bucket angeben oder einen separaten Pfad in demselben Bucket verwenden (z. B. ein Protokollpräfix oder eine Partitionierung verwenden). Auf diese Weise behalten Sie den Überblick darüber, welche Protokolldateien mit welcher Distribution verknüpft sind, und verhindert, dass sich Protokolldateien gegenseitig überschreiben.

Gehen Sie wie folgt vor, um mit der Standardprotokollierung (Legacy) zu beginnen:

1. Wählen Sie einen Amazon S3 S3-Bucket aus, der Ihre Protokolle empfängt, und fügen Sie die erforderlichen Berechtigungen hinzu.
2. Konfigurieren Sie die Standardprotokollierung (Legacy) CloudFront über die Konsole oder die CloudFront API. Sie können nur einen Amazon S3 S3-Bucket auswählen, um Ihre Protokolle zu empfangen.
3. Sehen Sie sich Ihre Zugriffsprotokolle an.

Wählen Sie einen Amazon S3 S3-Bucket für Standardprotokolle

Wenn Sie die Protokollierung für eine Distribution aktivieren, geben Sie den Amazon S3 S3-Bucket CloudFront an, in dem Sie Protokolldateien speichern möchten. Wenn Sie Amazon S3 als Ihren Ursprung verwenden, empfehlen wir Ihnen, einen separaten Bucket für Ihre Protokolldateien zu verwenden.

Geben Sie den Amazon S3 S3-Bucket CloudFront an, in dem Sie Zugriffs-Logs speichern möchten, `amzn-s3-demo-bucket.s3.amazonaws.com` z. B.

Sie können Protokolldateien für mehrere Verteilungen in demselben Bucket speichern. Wenn Sie die Protokollierung aktivieren, können Sie ein optionales Präfix für den Dateinamen festlegen; so behalten Sie den Überblick darüber, welche Protokolldateien welchen Verteilungen zugeordnet sind.

Über die Auswahl eines S3-Buckets

- In Ihrem Bucket muss die Zugriffskontrollliste (ACL) aktiviert sein. Wenn Sie in der CloudFront Konsole einen Bucket ohne aktivierte ACL auswählen, wird eine Fehlermeldung angezeigt. Siehe Berechtigungen.
- Wählen Sie keinen Amazon-S3-Bucket, wenn S3-Objekteigentümerschaft auf Bucket-Eigentümer erzwungen festgelegt ist. Diese Einstellung ist ACLs für den Bucket und die darin enthaltenen Objekte deaktiviert, wodurch CloudFront verhindert wird, dass Protokolldateien an den Bucket gesendet werden.
- Wählen Sie im Folgenden keinen Amazon S3 S3-Bucket aus AWS-Regionen. CloudFront liefert keine Standardprotokolle an Buckets in diesen Regionen:
  - Afrika (Kapstadt)
  - Asien-Pazifik (Hongkong)
  - Asien-Pazifik (Hyderabad)
  - Asien-Pazifik (Jakarta)
  - Asien-Pazifik (Melbourne)
  - Kanada West (Calgary)
  - Europa (Milan)
  - Europa (Spain)
  - Europa (Zürich)
  - Israel (Tel Aviv)
  - Naher Osten (Bahrain)
  - Naher Osten (VAE)

Berechtigungen

Important

Ab April 2023 müssen Sie S3 ACLs für neue S3-Buckets aktivieren, die für CloudFront Standardprotokolle verwendet werden. Sie können die Option aktivieren ACLs , wenn Sie einen Bucket erstellen, oder die Aktivierung ACLs für einen vorhandenen Bucket. Weitere Informationen zu den Änderungen finden Sie unter Häufig gestellte Fragen zu Standardeinstellungen für neue S3-Buckets im Benutzerhandbuch zu Amazon Simple

Storage Service und unter Achtung: Sicherheitsänderungen in Amazon S3 im April 2023 im AWS News-Blog.

Sie AWS-Konto müssen über die folgenden Berechtigungen für den Bucket verfügen, den Sie für Protokolldateien angeben:

- Die ACL für den Bucket muss Ihnen gewährenFULL_CONTROL. Wenn Sie der Bucket-Eigentümer sind, verfügt Ihr Konto standardmäßig über diese Berechtigung. Sind Sie das nicht, muss der Bucket-Eigentümer die ACL für den Bucket aktualisieren.
- `s3:GetBucketAcl`
- `s3:PutBucketAcl`

ACL für den Bucket

Wenn Sie eine Distribution erstellen oder aktualisieren und die Protokollierung aktivieren, CloudFront verwendet diese Berechtigungen, um die ACL für den Bucket zu aktualisieren, um dem `awslogsdelivery` Konto die entsprechenden Berechtigungen zu `FULL_CONTROL` erteilen. Das `awslogsdelivery`-Konto schreibt Protokolldateien in den Bucket. Wenn Ihr Konto nicht über die erforderlichen Berechtigungen zum Aktualisieren der ACL verfügt, schlägt das Erstellen oder Aktualisieren der Verteilung fehl.

Wenn Sie programmgesteuert eine Anfrage senden, um einen Bucket zu erstellen, aber ein Bucket mit dem angegebenen Namen bereits vorhanden ist, setzt S3 in einigen Fällen die Berechtigungen für den Bucket auf den Standardwert zurück. Wenn Sie das Speichern von Zugriffsprotokollen in einem S3-Bucket konfiguriert CloudFront haben und Sie keine Protokolle mehr in diesem Bucket abrufen, überprüfen Sie die Berechtigungen für den Bucket, um sicherzustellen, dass dieser CloudFront über die erforderlichen Berechtigungen verfügt.

Wiederherstellung der ACL für den Bucket

Wenn Sie Berechtigungen für das `awslogsdelivery`-Konto aufheben, kann CloudFront keine Protokolle in dem S3-Bucket speichern. Um wieder mit dem Speichern von Protokollen für Ihre Distribution beginnen CloudFront zu können, stellen Sie die ACL-Berechtigung wieder her, indem Sie einen der folgenden Schritte ausführen:

- Deaktivieren Sie die Protokollierung für Ihre Distribution und aktivieren Sie sie dann erneut. CloudFront Weitere Informationen finden Sie unter Standardprotokollierung.

- Fügen Sie die ACL-Berechtigung für `awslogsdelivery` manuell hinzu, indem Sie in der Amazon S3-Konsole zu dem S3-Bucket gehen und die Berechtigung hinzufügen. Um die ACL für `awslogsdelivery` hinzuzufügen, müssen Sie die kanonische ID für das Konto angeben, nämlich:

  ```
  c4c1ede66af53448b93c283ce9448c4ba468c9432aa01d700d3878632f77d2d0
  ```

  Weitere Informationen zum Hinzufügen ACLs zu S3-Buckets finden Sie unter Konfiguration ACLs im Amazon Simple Storage Service-Benutzerhandbuch.

ACL für jede Protokolldatei

Neben der ACL für den Bucket gibt es auch ACLs für jede einzelne Protokolldatei. Der Bucket-Eigentümer verfügt für jede Protokolldatei über die Berechtigung `FULL_CONTROL`, der Eigentümer der Verteilung (wenn dieser vom Bucket-Eigentümer abweicht) hat keine Berechtigung und das `awslogsdelivery`-Konto verfügt über Lese- und Schreibberechtigungen.

Deaktivieren der Protokollierung

Wenn Sie die Protokollierung deaktivieren, CloudFront werden weder ACLs für den Bucket noch für die Protokolldateien gelöscht. Sie können die bei ACLs Bedarf löschen.

Erforderliche Schlüsselrichtlinie für SSE-KMS Buckets

Wenn der S3-Bucket für Ihre Standardprotokolle serverseitige Verschlüsselung mit AWS KMS keys (SSE-KMS) unter Verwendung eines vom Kunden verwalteten Schlüssels verwendet, müssen Sie der Schlüsselrichtlinie für Ihren vom Kunden verwalteten Schlüssel die folgende Erklärung hinzufügen. Dies ermöglicht das Schreiben CloudFront von Protokolldateien in den Bucket. Sie können SSE-KMS nicht mit dem verwenden Von AWS verwalteter Schlüssel , da CloudFront dann keine Protokolldateien in den Bucket geschrieben werden können.

```
{
    "Sid": "Allow CloudFront to use the key to deliver logs",
    "Effect": "Allow",
    "Principal": {
        "Service": "delivery.logs.amazonaws.com"
    },
    "Action": "kms:GenerateDataKey*",
    "Resource": "*"
}
```

Wenn der S3-Bucket für Ihre Standardprotokolle SSE-KMS mit einem S3-Bucket-Schlüssel verwendet, müssen Sie der Richtlinienanweisung auch die `kms:Decrypt` entsprechende Berechtigung hinzufügen. In diesem Fall sieht die vollständige Richtlinienanweisung wie folgt aus.

```
{
    "Sid": "Allow CloudFront to use the key to deliver logs",
    "Effect": "Allow",
    "Principal": {
        "Service": "delivery.logs.amazonaws.com"
    },
    "Action": [
        "kms:GenerateDataKey*",
        "kms:Decrypt"
    ],
    "Resource": "*"
}
```

> **Note**
>
> Wenn Sie SSE-KMS für Ihren S3-Bucket aktivieren, geben Sie den vollständigen ARN für den vom Kunden verwalteten Schlüssel an. Weitere Informationen finden Sie unter Serverseitige Verschlüsselung mit AWS KMS keys (SSE-KMS) angeben im Amazon Simple Storage Service-Benutzerhandbuch.

Aktivieren Sie die Standardprotokollierung (veraltet)

Verwenden Sie die CloudFront Konsole oder die CloudFront API, um Standardprotokolle zu aktivieren.

Inhalt

- Aktivieren Sie die Standardprotokollierung (Legacy) (CloudFrontKonsole)
- Aktivieren Sie die Standardprotokollierung (Legacy) (CloudFrontAPI)

Aktivieren Sie die Standardprotokollierung (Legacy) (CloudFrontKonsole)

Um Standardprotokolle für eine CloudFront Distribution zu aktivieren (Konsole)

1. Verwenden Sie die CloudFront Konsole, um eine neue Distribution zu erstellen oder eine bestehende zu aktualisieren.
2. Wählen Sie im Bereich Standardprotokollierung für Protokollzustellung die Option On aus.
3. (Optional) Wählen Sie für die Cookie-Protokollierung die Option Ein, wenn Sie Cookies in Ihre Protokolle aufnehmen möchten. Weitere Informationen finden Sie unter Protokollierung von Cookies.

   > **Tip**
   >
   > Die Cookie-Protokollierung ist eine globale Einstellung, die für alle Standardprotokolle Ihrer Distribution gilt. Sie können diese Einstellung nicht für separate Lieferziele überschreiben.

4. Geben Sie für den Abschnitt Liefern an Amazon S3 (Legacy) an.
5. Geben Sie Ihren Amazon S3 S3-Bucket an. Wenn Sie noch keinen haben, können Sie Create wählen oder sich die Dokumentation ansehen, um einen Bucket zu erstellen.
6. (Optional) Geben Sie unter Protokollpräfix die Zeichenfolge CloudFront an, die Sie den Namen der Zugriffs-Logdateien für diese Distribution voranstellen möchten, `exampleprefix/` z. B. Der abschließende Schrägstrich (/) ist optional, jedoch empfohlen, um das Durchsuchen Ihrer Protokolldateien zu vereinfachen. Weitere Informationen finden Sie unter Protokollpräfix.
7. Führen Sie die Schritte aus, um Ihre Distribution zu aktualisieren oder zu erstellen.
8. Vergewissern Sie sich auf der Seite Protokolle, dass der Status der Standardprotokolle neben der Verteilung aktiviert ist.

   Weitere Informationen zur standardmäßigen Übermittlung der Protokollierung und zu den Protokollfeldern finden Sie unterReferenz zur Standardprotokollierung.

Aktivieren Sie die Standardprotokollierung (Legacy) (CloudFrontAPI)

Sie können die CloudFront API auch verwenden, um Standardprotokolle für Ihre Distributionen zu aktivieren.

Um Standardprotokolle für eine Distribution (CloudFront API) zu aktivieren

- Verwenden Sie die UpdateDistributionAPI-Operation CreateDistributionoder und konfigurieren Sie das LoggingConfigObjekt.

Bearbeiten Sie die Standardprotokollierungseinstellungen

Sie können die Protokollierung aktivieren oder deaktivieren, den Amazon S3 S3-Bucket ändern, in dem Ihre Protokolle gespeichert sind, und das Präfix für Protokolldateien ändern, indem Sie die CloudFront Konsole oder die CloudFront API verwenden. Ihre Änderungen der Protokollierungseinstellungen werden innerhalb von 12 Stunden wirksam.

Weitere Informationen finden Sie unter den folgenden Themen:

- Informationen zum Aktualisieren einer Distribution mithilfe der CloudFront Konsole finden Sie unterEine Verteilung aktualisieren.
- Informationen zum Aktualisieren einer Distribution mithilfe der CloudFront API finden Sie UpdateDistributionin der Amazon CloudFront API-Referenz.

Protokolle an Amazon S3 senden

Wenn Sie Ihre Protokolle an Amazon S3 senden, werden Ihre Protokolle im folgenden Format angezeigt.

Dateinamenformat

Der Name jeder Protokolldatei, die in Ihrem Amazon S3 S3-Bucket CloudFront gespeichert wird, verwendet das folgende Dateinamenformat:

<optional prefix>/*<distribution ID>*.*YYYY*-*MM*-*DD*-*HH*.*unique-ID*.gz

Datum und Uhrzeit entsprechen der Zeitzone UTC (Coordinated Universal Time).

Wenn Sie beispielsweise `example-prefix` als Präfix verwenden und Ihre Verteilungs-ID `EMLARXS9EXAMPLE` lautet, sehen Ihre Dateinamen folgendermaßen aus:

```
example-prefix/EMLARXS9EXAMPLE.2019-11-14-20.RT4KCN4SGK9.gz
```

Wenn Sie die Protokollierung für eine Verteilung aktivieren, können Sie ein optionales Präfix für den Dateinamen festlegen; so behalten Sie den Überblick darüber, welche Protokolldateien welchen

Verteilungen zugeordnet sind. Wenn Sie einen Wert für das Präfix der Protokolldatei angeben und Ihr Präfix nicht mit einem Schrägstrich (/) CloudFront endet, wird automatisch einer angehängt. Wenn Ihr Präfix mit einem Schrägstrich endet, wird CloudFront kein weiterer hinzugefügt.

Das `.gz` Ende des Dateinamens weist darauf hin, dass die Protokolldatei mit gzip komprimiert CloudFront wurde.

Dateiformat des Standardprotokolls

Jeder Eintrag in einer Protokolldatei enthält detaillierte Informationen zu den einzelnen Viewer-Anfragen. Die Protokolldateien weisen folgende Merkmale auf:

- Sie verwenden das erweiterte W3C-Format für Protokolldateien.
- Sie enthalten tabulatorgetrennte Werte.
- Sie enthalten Datensätze in nicht unbedingt chronologischer Reihenfolge.
- Sie enthalten zwei Headerzeilen: eine mit der Version des Dateiformats und eine andere mit den W3C-Feldern für jeden einzelnen Datensatz.
- Sie enthalten URL-codierte Äquivalente für Leerzeichen und bestimmte andere Zeichen in Feldwerten.

  URL-kodierte Äquivalente werden für die folgenden Zeichen verwendet:

  - ASCII-Zeichencodes 0 bis 32, inklusive
  - ASCII-Zeichencodes 127 und höher
  - Alle Zeichen in der folgenden Tabelle

  Der URL-Codierungsstandard ist in RFC 1738 definiert.

URL-codierter Wert	Zeichen
%3C	<
%3E	>
%22	"
%23	#
%25	%

URL-codierter Wert	Zeichen
%7B	{
%7D	}
%7C	|
%5C	\
%5E	^
%7E	~
%5B	[
%5D	]
%60	`
%27	'
%20	Leerzeichen

Logdateien löschen

CloudFront löscht Protokolldateien nicht automatisch aus Ihrem Amazon S3 S3-Bucket. Informationen zum Löschen von Protokolldateien aus einem Amazon S3 S3-Bucket finden Sie unter Objekte löschen im Amazon Simple Storage Service Console-Benutzerhandbuch.

Preisgestaltung

Die Standardprotokollierung ist eine optionale Funktion von CloudFront. CloudFront berechnet keine Gebühren für die Aktivierung von Standardprotokollen. Es fallen jedoch die üblichen Amazon S3-Gebühren für das Speichern und Zugreifen auf die Dateien auf Amazon S3 an. Sie können sie jederzeit löschen.

Weitere Informationen zu Preisen finden Sie unter Amazon S3-Preise.

Weitere Informationen zur CloudFront Preisgestaltung finden Sie unter CloudFront Preisgestaltung.

Referenz zur Standardprotokollierung

Die folgenden Abschnitte gelten sowohl für die Standardprotokollierung (v2) als auch für die Standardprotokollierung (Legacy).

Themen

Zeitpunkt der Übermittlung der Protokolldatei

CloudFront liefert Protokolle für eine Verteilung bis zu mehrmals pro Stunde. Im Allgemeinen enthält eine Protokolldatei Informationen über die Anfragen, die während eines bestimmten Zeitraums CloudFront eingegangen sind. CloudFront In der Regel wird die Protokolldatei für diesen Zeitraum innerhalb einer Stunde nach den Ereignissen, die im Protokoll aufgeführt sind, an Ihr Ziel gesendet. Beachten Sie jedoch, dass einige oder auch alle Protokolldateieinträge für einen bestimmten Zeitraum manchmal mit einer Verzögerung von bis zu 24 Stunden übermittelt werden können. Wenn Protokolleinträge verzögert werden, werden sie in einer Protokolldatei CloudFront gespeichert, deren Dateiname das Datum und die Uhrzeit des Zeitraums enthält, in dem die Anfragen aufgetreten sind, nicht das Datum und die Uhrzeit der Zustellung der Datei.

Bei der Erstellung einer Protokolldatei werden Informationen für Ihre Verteilung von allen Edge-Standorten CloudFront konsolidiert, die während des Zeitraums, den die Protokolldatei abdeckt, Anfragen für Ihre Objekte erhalten haben.

CloudFront kann mehr als eine Datei für einen bestimmten Zeitraum speichern, je nachdem, wie viele Anfragen für CloudFront die mit einer Verteilung verknüpften Objekte eingehen.

CloudFront beginnt etwa vier Stunden, nachdem Sie die Protokollierung aktiviert haben, mit der zuverlässigen Bereitstellung von Zugriffsprotokollen. Möglicherweisen erhalten Sie ein paar Zugriffsprotokolle auch schon vorher.

> **Note**
>
> Wenn während eines bestimmten Zeitraums Ihre Objekte nicht von Benutzern angefordert werden, erhalten Sie keine Protokolldateien für diesen Zeitraum.

Protokollierung von Anforderungen, wenn Anforderungs-URL oder Header die maximale Größe überschreiten

Wenn die Gesamtgröße aller Anfrage-Header, einschließlich Cookies, 20 KB überschreitet oder wenn die URL die Größe von 8.192 Byte überschreitet, kann CloudFront die Anforderung nicht vollständig analysieren und die Anforderung nicht protokollieren. Da die Anfrage nicht protokolliert wird, wird in den Protokolldateien der HTTP-Fehler-Statuscode nicht zurückgegeben.

Wenn der Anfragetext die maximale Größe überschreitet, wird die Anfrage einschließlich des HTTP-Fehlerstatuscodes protokolliert.

Felder in der Protokolldatei

Die Protokolldatei für eine Verteilung enthält 33 Felder. Die folgende Liste enthält jeden Feldnamen in der Reihenfolge sowie eine Beschreibung der Informationen in diesem Feld.

1. **`date`**

   Das Datum, an dem das Ereignis aufgetreten ist, im Format `YYYY-MM-DD`. Beispiel, `2019-06-30`. Datum und Uhrzeit entsprechen der Zeitzone UTC (Coordinated Universal Time). Bei WebSocket Verbindungen ist dies das Datum, an dem die Verbindung geschlossen wurde.
2. **`time`**

   Die Uhrzeit, zu der der CloudFront Server die Anfrage nicht mehr beantwortet hat (in UTC), `01:42:39` z. B. Bei WebSocket Verbindungen ist dies der Zeitpunkt, zu dem die Verbindung geschlossen wird.
3. **`x-edge-location`**

   Der Edge-Standort, an dem die Anfrage verarbeitet wurde. Jede Kantenposition wird durch einen aus drei Buchstaben bestehenden Code und eine willkürlich zugewiesene Zahl (z. B. DFW3) identifiziert. Der Code aus drei Buchstaben entspricht in der Regel dem Code der International Air Transport Association (IATA) für einen Flughafen in der Nähe des Edge-Standorts. (Diese Abkürzungen ändern sich möglicherweise in der Zukunft.)

4. **`sc-bytes`**

   Die Gesamtzahl der Bytes, die der Server als Antwort auf die Anforderung an den Viewer übermittelt hat, einschließlich Headern. Für WebSocket und gRPC-Verbindungen ist dies die Gesamtzahl der Byte, die vom Server über die Verbindung an den Client gesendet werden.

5. **`c-ip`**

   Die IP-Adresse des Betrachters, die der Anfrage gestellt hat, z. B. `192.0.2.183` oder `2001:0db8:85a3::8a2e:0370:7334`. Wenn der Viewer einen HTTP-Proxy oder eine Load Balancer verwendet hat, um die Anforderung zu senden, entspricht der Wert dieses Feldes der IP-Adresse des Proxys bzw. des Load Balancers. Siehe auch das Feld `x-forwarded-for`.

6. **`cs-method`**

   Die vom Viewer empfangene HTTP-Anforderungsmethode.

7. **`cs(Host)`**

   Der Domainname der CloudFront Distribution (z. B. d111111abcdef8.cloudfront.net).

8. **`cs-uri-stem`**

   Der Teil der Anforderungs-URL, der den Pfad und das Objekt identifiziert (z. B, `/images/cat.jpg`). Fragezeichen (?) In URLs - und Query-Strings sind nicht im Protokoll enthalten.

9. **`sc-status`**

   Enthält einen der folgenden Werte:

   - Den HTTP-Statuscode der Antwort des Servers (z. B. `200`).
   - `000`, was anzeigt, dass der Viewer die Verbindung geschlossen hat, bevor der Server auf die Anforderung antworten konnte. Wenn der Viewer die Verbindung schließt, nachdem der Server mit dem Senden der Antwort begonnen hat, enthält dieses Feld den HTTP-Statuscode der Antwort, mit deren Senden der Server begonnen hatte.

10. **`cs(Referer)`**

    Der Wert für den `Referer`-Header in der Anfrage. Der Name der Domäne, von der die Anforderung ausgegangen ist. Häufig vorkommende Referrer sind Suchmaschinen, andere Websites, die direkt auf Ihre Objekte verlinken, und Ihre eigene Website.

11. **`cs(User-Agent)`**

Der Wert für den `User-Agent`-Header in der Anforderung. Der `User-Agent`-Header bezeichnet die Quelle für die Anforderung, z. B. den Gerätetyp und den Browser, der die Anforderung abgesendet hat, oder die Suchmaschine, wenn die Anforderung von einer Suchmaschine stammt.

12 **`cs-uri-query`**

Der Teil der Anforderungs-URL mit der Abfragezeichenfolge, wenn vorhanden.

Wenn eine URL keine Abfragezeichenfolge enthält, ist der Wert dieses Felds ein Bindestrich (-). Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern.

13 **`cs(Cookie)`**

Der `Cookie`-Header in der Anforderung einschließlich der Name-Wert-Paaren und zugehörigen Attributen.

Wenn Sie die Cookie-Protokollierung aktivieren, werden die Cookies bei allen Anfragen CloudFront protokolliert, unabhängig davon, welche Cookies Sie an den Ursprung weiterleiten. Wenn eine Anforderung keinen Cookie-Header enthält, ist der Wert dieses Felds ein Bindestrich (-). Weitere Informationen zu Cookies finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.

14 **`x-edge-result-type`**

Art der Klassifizierung der Antwort durch den Server, nachdem das letzte Byte den Server verlassen hat. Manchmal wird der Ergebnistyp zwischen dem Zeitpunkt, an dem Server zum Senden der Antwort bereit ist, und dem Zeitpunkt, an dem er das Senden der Antwort abgeschlossen hat, geändert. Siehe auch das Feld `x-edge-response-result-type`.

Angenommen, im HTTP-Streaming findet der Server ein Segment des Streams im Cache. In diesem Szenario würde der Wert dieses Feldes normalerweise sei `Hit`. Wenn der Viewer jedoch die Verbindung schließt, bevor der Server das ganze Segment übermittelt hat, ist der endgültige Ergebnistyp (und damit der Wert dieses Felds) `Error`.

WebSocket und gRPC-Verbindungen haben `Miss` für dieses Feld den Wert von, da der Inhalt nicht zwischengespeichert werden kann und direkt an den Ursprung weitergeleitet wird.

Mögliche Werte sind:

- `Hit` – Der Server hat das Objekt aus dem Cache für den Betrachter bereitgestellt.

- `RefreshHit` – Der Server hat das Objekt im Edge-Zwischenspeicher gefunden, es war jedoch abgelaufen. Daher nahm der Server Kontakt mit dem Ursprung auf, um zu überprüfen, ob der Zwischenspeicher die neueste Version des Objekts enthalten hatte.
- `Miss` – Die Anforderung konnte nicht durch ein Objekt im Zwischenspeicher bedient werden. Daher hat der Server die Anforderung an den Ursprung weitergeleitet und das Ergebnis an den Betrachter ausgegeben.
- `LimitExceeded`— Die Anfrage wurde abgelehnt, weil ein CloudFront Kontingent (früher als Limit bezeichnet) überschritten wurde.
- `CapacityExceeded` – Der Server hat den HTTP-Statuscode 503 zurückgegeben, da er zum Zeitpunkt der Anforderung nicht über genügend Kapazitäten für die Bereitstellung des Objekts verfügte.
- `Error` – In der Regel bedeutet dies, dass die Anforderung zu einem Client-Fehler (d. h. der Wert des Felds `sc-status` liegt im `4xx`-Bereich) oder zu einem Serverfehler geführt hat (d. h. der Wert des Felds `sc-status` liegt im `5xx`-Bereich). Wenn der Wert des Felds `sc-status` `200` oder wenn der Wert dieses Felds `Error` ist und der Wert des Felds `x-edge-response-result-type` nicht `Error` ist, war die HTTP-Anforderung erfolgreich. Die Client-Verbindung wurde jedoch getrennt, bevor alle Bytes empfangen wurden.
- `Redirect` – Der Server hat den Betrachter entsprechend den Verteilungseinstellungen von HTTP zu HTTPS umgeleitet.
- `LambdaExecutionError`— Die der Distribution zugeordnete Lambda @Edge -Funktion wurde aufgrund einer fehlerhaften Zuordnung, eines Funktionstimeouts, eines AWS Abhängigkeitsproblems oder eines anderen allgemeinen Verfügbarkeitsproblems nicht abgeschlossen.

15 **`x-edge-request-id`**

Eine undurchsichtige Zeichenfolge, die eine Anfrage eindeutig identifiziert. CloudFront sendet diese Zeichenfolge auch im `x-amz-cf-id` Antwort-Header.

16 **`x-host-header`**

Der Wert, den der Viewer in den `Host`-Header der Anforderung eingefügt hat.
Wenn Sie den CloudFront Domainnamen in Ihrem Objekt verwenden URLs (z. B. d111111abcdef8.cloudfront.net), enthält dieses Feld diesen Domainnamen. Wenn Sie alternative Domainnamen (CNAMEs) in Ihrem Objekt verwenden URLs (z. B. www.example.com), enthält dieses Feld den alternativen Domainnamen.

Wenn Sie alternative Domain-Namen verwenden, finden Sie unter `cs(Host)` in Feld 7 den Namen der Domain, die Ihrer Verteilung zugewiesen ist.

17. **`cs-protocol`**

    Das Protokoll der Viewer-Anforderung (`http`, `https`, `grpcs`, `ws` oder `wss`).

18. **`cs-bytes`**

    Die Gesamtzahl der Bytes an Daten, die in der Anforderung des Viewers einschließlich Headern enthalten sind. Für WebSocket und gRPC-Verbindungen ist dies die Gesamtzahl der Byte, die vom Client an den Server über die Verbindung gesendet wurden.

19. **`time-taken`**

    Die Anzahl der Sekunden (auf die Tausendstelsekunde genau, z. B. 0,082) ab dem Zeitpunkt, an dem der Server die Anforderung des Viewers empfängt, bis zu dem Zeitpunkt, an dem der Server das letzte Byte der Antwort in die Ausgabewarteschlange schreibt, wie auf dem Server gemessen. Aus der Perspektive der Viewers vergeht bis zum Empfang der gesamten Antwort aufgrund der Netzwerklatenz und des TCP-Puffervorgangs insgesamt mehr Zeit, als mit diesem Wert angegeben wird.

20. **`x-forwarded-for`**

    Wenn der Viewer einen HTTP-Proxy oder Load Balancer verwendet hat, um die Anforderung zu senden, entspricht der Wert des Felds `c-ip` der IP-Adresse des Proxys oder Load Balancers. In diesem Fall stellt dieses Feld die IP-Adresse des Viewers dar, von dem die Anfrage stammt. Dieses Feld kann mehrere durch Kommas getrennte IP-Adressen enthalten. Jede IP-Adresse kann eine IPv4 Adresse (zum Beispiel`192.0.2.183`) oder eine IPv6 Adresse (zum Beispiel) sein. `2001:0db8:85a3::8a2e:0370:7334`

    Wenn der Viewer keinen HTTP-Proxy oder Load Balancer verwendet hat, ist der Wert dieses Feldes ein Bindestrich (-).

21. **`ssl-protocol`**

    Wenn für die Anfrage HTTPS verwendet wurde, enthält dieses Feld das SSL/TLS Protokoll, das der Betrachter und der Server für die Übertragung der Anfrage und Antwort ausgehandelt haben. Eine Liste der möglichen Werte finden Sie unter den unterstützten SSL/TLS ProtokollenUnterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront.

Wenn als `cs-protocol` in Feld 17 `http` angegeben ist, ist der Wert für dieses Feld ein Bindestrich (-).

22 **`ssl-cipher`**

Wenn für die Anfrage HTTPS verwendet wurde, enthält dieses Feld die SSL/TLS Chiffre, die der Betrachter und der Server für die Verschlüsselung der Anfrage und Antwort ausgehandelt haben. Eine Liste der möglichen Werte finden Sie unter den unterstützten SSL/TLS Verschlüsselungen. Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront

Wenn als `cs-protocol` in Feld 17 `http` angegeben ist, ist der Wert für dieses Feld ein Bindestrich (-).

23 **`x-edge-response-result-type`**

Die Art, wie der Server die Antwort direkt vor der Rücksendung der Antwort an den Viewer klassifiziert hat. Siehe auch das Feld `x-edge-result-type`. Mögliche Werte sind:

- `Hit` – Der Server hat das Objekt aus dem Cache für den Betrachter bereitgestellt.
- `RefreshHit` – Der Server hat das Objekt im Edge-Zwischenspeicher gefunden, es war jedoch abgelaufen. Daher nahm der Server Kontakt mit dem Ursprung auf, um zu überprüfen, ob der Zwischenspeicher die neueste Version des Objekts enthalten hatte.
- `Miss` – Die Anforderung konnte nicht durch ein Objekt im Zwischenspeicher bedient werden. Daher hat der Server die Anforderung an den Ursprungs-Server weitergeleitet und das Ergebnis an den Betrachter ausgegeben.
- `LimitExceeded`— Die Anfrage wurde abgelehnt, weil ein CloudFront Kontingent (früher als Limit bezeichnet) überschritten wurde.
- `CapacityExceeded` – Der Server hat den Fehler 503 zurückgegeben, da er zum Zeitpunkt der Anforderung nicht über genügend Kapazitäten verfügte, um das Objekt bereitzustellen.
- `Error` – In der Regel bedeutet dies, dass die Anforderung zu einem Client-Fehler (d. h. der Wert des Felds `sc-status` liegt im `4xx`-Bereich) oder zu einem Serverfehler geführt hat (d. h. der Wert des Felds `sc-status` liegt im `5xx`-Bereich).

  Wenn der Wert des Felds `x-edge-result-type` `Error` ist und der Wert dieses Felds nicht `Error` ist, wurde die Verbindung vor dem Abschluss des Downloads durch den Client getrennt.
- `Redirect` – Der Server hat den Betrachter entsprechend den Verteilungseinstellungen von HTTP zu HTTPS umgeleitet.

- `LambdaExecutionError`— Die der Distribution zugeordnete Lambda @Edge -Funktion wurde aufgrund einer fehlerhaften Zuordnung, eines Funktionstimeouts, eines AWS Abhängigkeitsproblems oder eines anderen allgemeinen Verfügbarkeitsproblems nicht abgeschlossen.

24. **`cs-protocol-version`**

    Die HTTP-Version, die der Viewer in der Anfrage angegeben hat: Mögliche Werte sind `HTTP/0.9`, `HTTP/1.0`, `HTTP/1.1`, `HTTP/2.0` und `HTTP/3.0`.

25. **`fle-status`**

    Wenn die Verschlüsselung auf Feldebene für eine Verteilung konfiguriert ist, enthält dieses Feld einen Code, der angibt, ob der Anforderungstext erfolgreich verarbeitet wurde. Wenn der Server den Anforderungstext erfolgreich verarbeitet, Werte in den angegebenen Feldern verschlüsselt und die Anforderung an den Ursprung weiterleitet, ist der Wert dieses Felds `Processed`. Der Wert von `x-edge-result-type` kann in diesem Fall immer noch auf einen clientseitigen oder serverseitigen Fehler hinweisen.

    Mögliche Werte für dieses Feld sind:

    - `ForwardedByContentType` – Der Server hat die Anforderung ohne Parsing oder Verschlüsselung an den Ursprung weitergeleitet, da kein Content-Typ konfiguriert wurde.
    - `ForwardedByQueryArgs` – Der Server hat die Anforderung ohne Parsing oder Verschlüsselung an den Ursprung weitergeleitet, da die Anforderung ein Abfrageargument enthält, das nicht in der Konfiguration für die Verschlüsselung auf Feldebene enthalten war.
    - `ForwardedDueToNoProfile` – Der Server hat die Anforderung ohne Parsing oder Verschlüsselung an den Ursprung weitergeleitet, da in der Konfiguration für die Verschlüsselung auf Feldebene kein Profil angegeben wurde.
    - `MalformedContentTypeClientError` – Der Server hat die Anforderung zurückgewiesen und einen HTTP-400-Statuscode an den Betrachter zurückgegeben, da der Wert des `Content-Type`-Headers ein ungültiges Format hatte.
    - `MalformedInputClientError` – Der Server hat die Anforderung zurückgewiesen und einen HTTP 400-Statuscode an den Betrachter zurückgegeben, da der Anforderungstext ein ungültiges Format hatte.
    - `MalformedQueryArgsClientError` – Der Server hat die Anforderung zurückgewiesen und einen HTTP-400-Statuscode an den Betrachter zurückgegeben, weil ein Abfrageargument leer war oder ein ungültiges Format hatte.

- `RejectedByContentType` – Der Server hat die Anforderung zurückgewiesen und einen HTTP-400-Statuscode an den Betrachter zurückgegeben, da in der Konfiguration für die Verschlüsselung auf Feldebene kein Content-Typ angegeben wurde.
- `RejectedByQueryArgs` – Der Server hat die Anforderung zurückgewiesen und einen HTTP-400-Statuscode an den Betrachter zurückgegeben, da in der Konfiguration für die Verschlüsselung auf Feldebene kein Abfrageargument angegeben wurde.
- `ServerError` – Der Ursprungs-Server hat einen Fehler zurückgegeben.

Wenn die Anforderung das Kontingent für die Verschlüsselung auf Feldebene überschreitet (früher als Limit bezeichnet), enthält dieses Feld einen der folgenden Fehlercodes und der Server gibt dem Viewer den HTTP-Statuscode 400 zurück. Eine Liste der aktuellen Kontingente für die Verschlüsselung auf Feldebene finden Sie unter Kontingente für Verschlüsselung auf Feldebene.

- `FieldLengthLimitClientError` – Ein Feld, das für die Verschlüsselung konfiguriert ist, hat die maximal zulässige Länge überschritten.
- `FieldNumberLimitClientError` – Eine Anforderung, die die Verteilung verschlüsseln soll, enthält mehr Felder als zulässig.
- `RequestLengthLimitClientError` – Die Länge des Anfragetexts hat die maximal zulässige Länge, wenn die Verschlüsselung auf Feldebene konfiguriert ist, überschritten.

Wenn die Verschlüsselung auf Feldebene nicht für die Verteilung konfiguriert ist, ist der Wert dieses Feldes ein Bindestrich (-).

26. **`fle-encrypted-fields`**

Die Anzahl der Felder für die Verschlüsselung auf Feldebene, die der Server verschlüsselt und an den Ursprung weitergeleitet hat. CloudFront Server streamen die verarbeitete Anfrage an den Ursprung, während sie Daten verschlüsseln, sodass dieses Feld auch dann einen Wert haben kann, wenn der Wert von ein Fehler `fle-status` ist.

Wenn die Verschlüsselung auf Feldebene nicht für die Verteilung konfiguriert ist, ist der Wert dieses Feldes ein Bindestrich (-).

27. **`c-port`**

Die Portnummer der Anforderung des Viewers.

28. **`time-to-first-byte`**

Die Anzahl der Sekunden zwischen dem Empfangen der Anforderung und dem Schreiben des ersten Bytes der Antwort, gemessen auf dem Server.

29**x-edge-detailed-result-type**

Dieses Feld enthält den gleichen Wert wie der Wert für das `x-edge-result-type`-Feld, außer in den folgenden Fällen:

- Wenn das Objekt dem Viewer aus der Origin-Shield-Ebene bereitgestellt wurde, enthält dieses Feld `OriginShieldHit`.
- Wenn sich das Objekt nicht im CloudFront Cache befand und die Antwort von einer Lambda @Edge -Funktion für die ursprüngliche Anfrage generiert wurde, enthält `MissGeneratedResponse` dieses Feld.
- Wenn der Wert des `x-edge-result-type`-Felds `Error` lautet, enthält dieses Feld einen der folgenden Werte mit weiteren Informationen zum Fehler:
  - `AbortedOrigin` – Der Server hat ein Problem mit dem Ursprung festgestellt.
  - `ClientCommError` – Die Antwort auf den Betrachter wurde aufgrund eines Kommunikationsproblems zwischen dem Server und dem Betrachter unterbrochen.
  - `ClientGeoBlocked` – Die Verteilung ist so konfiguriert, dass Anforderungen vom geografischen Standort des Viewers abgelehnt werden.
  - `ClientHungUpRequest` — Der Betrachter wurde beim Senden der Anfrage vorzeitig gestoppt.
  - `Error` – Es ist ein Fehler aufgetreten, dessen Fehlertyp zu keiner der anderen Kategorien passt. Dieser Fehlertyp kann auftreten, wenn der Server eine Fehlerantwort aus dem Cache ausgibt.
  - `InvalidRequest` – Der Server hat eine ungültige Anforderung vom Betrachter erhalten.
  - `InvalidRequestBlocked` — Der Zugriff auf die angeforderte Ressource ist blockiert.
  - `InvalidRequestCertificate`— Die Verteilung stimmt nicht mit dem SSL/TLS Zertifikat überein, für das die HTTPS-Verbindung hergestellt wurde.
  - `InvalidRequestHeader` – Die Anforderung enthielt einen ungültigen Header.
  - `InvalidRequestMethod` — Die Verteilung ist nicht für eine Verarbeitung der verwendeten HTTP-Anforderungsmethode konfiguriert. Dies kann vorkommen, wenn die Verteilung nur Anforderungen unterstützt, die zwischengespeichert werden können.
  - `OriginCommError` – Bei der Anforderung ist eine Zeitüberschreitung aufgetreten, während eine Verbindung mit dem Ursprung hergestellt wurde oder Daten aus dem Ursprung gelesen wurden.
  - `OriginConnectError` – Der Server konnte keine Verbindung zum Ursprung herstellen.

- `OriginContentRangeLengthError` – Der `Content-Length`-Header in der Antwort des Ursprungs stimmt nicht mit der Länge im `Content-Range`-Header überein.
- `OriginDnsError` – Der Server konnte den Domänennamen des Ursprungs nicht auflösen.
- `OriginError` — Der Ursprung gab eine falsche Antwort zurück.
- `OriginHeaderTooBigError` – Ein vom Ursprung zurückgegebener Header ist für eine Verarbeitung durch den Edge-Server zu groß.
- `OriginInvalidResponseError` — Der Ursprung gab eine ungültige Antwort zurück.
- `OriginReadError` – Der Server konnte nicht vom Ursprung lesen.
- `OriginWriteError` – Der Server konnte nicht in den Ursprung schreiben.
- `OriginZeroSizeObjectError` — Ein Nullgrößenobjekt, das vom Ursprung gesendet wurde, führte zu einem Fehler.
- `SlowReaderOriginError` — Der Betrachter hat die Nachricht, die den Ursprungsfehler verursacht hat, nur langsam gelesen.

30 **`sc-content-type`**

Der Wert des HTTP `Content-Type`-Headers der Antwort.

31 **`sc-content-len`**

Der Wert des HTTP `Content-Length`-Headers der Antwort.

32 **`sc-range-start`**

Wenn die Antwort den HTTP `Content-Range`-Header enthält, enthält dieses Feld den Bereichsstartwert.

33 **`sc-range-end`**

Wenn die Antwort den HTTP `Content-Range`-Header enthält, enthält dieses Feld den Bereichsendwert.

34 **`distribution-tenant-id`**

Die ID des Distributionsmandanten.

Im Folgenden finden Sie ein Beispiel für eine Protokolldatei für eine Distribution.

```
#Version: 1.0
```

```
#Fields: date time x-edge-location sc-bytes c-ip cs-method cs(Host) cs-uri-stem sc-
status cs(Referer) cs(User-Agent) cs-uri-query cs(Cookie) x-edge-result-type x-edge-
request-id x-host-header cs-protocol cs-bytes time-taken x-forwarded-for ssl-protocol
 ssl-cipher x-edge-response-result-type cs-protocol-version fle-status fle-encrypted-
fields c-port time-to-first-byte x-edge-detailed-result-type sc-content-type sc-
content-len sc-range-start sc-range-end
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /
index.html 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;
%20x64)%20AppleWebKit/537.36%20(KHTML,%20like
%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
 SOX4xwn4XV6Q4rgb7XiVGOHms_BGlTAC4KyHmureZmBNrjGdRLiNIQ== d111111abcdef8.cloudfront.net
 https 23 0.001 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.001 Hit
 text/html 78 - -
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /
index.html 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;
%20x64)%20AppleWebKit/537.36%20(KHTML,%20like
%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
 k6WGMNkEzR5BEM_SaF47gjtX9zBDO2m349OY2an0QPEaUum1ZOLrow== d111111abcdef8.cloudfront.net
 https 23 0.000 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.000 Hit
 text/html 78 - -
2019-12-04 21:02:31 LAX1 392 192.0.2.100 GET d111111abcdef8.cloudfront.net /
index.html 200 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;
%20x64)%20AppleWebKit/537.36%20(KHTML,%20like
%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Hit
 f37nTMVvnKvV2ZSvEsivup_c2kZ7VXzYdjC-GUQZ5qNs-89BlWazbw== d111111abcdef8.cloudfront.net
 https 23 0.001 - TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256 Hit HTTP/2.0 - - 11040 0.001 Hit
 text/html 78 - -
2019-12-13 22:36:27 SEA19-C1 900 192.0.2.200 GET d111111abcdef8.cloudfront.net /
favicon.ico 502 http://www.example.com/ Mozilla/5.0%20(Windows
%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Error
 1pkpNfBQ39sYMnjjUQjmH2w1wdJnbHYTbag21o_3OfcQgPzdL2RSSQ== www.example.com http 675
 0.102 - - - Error HTTP/1.1 - - 25260 0.102 OriginDnsError text/html 507 - -
2019-12-13 22:36:26 SEA19-C1 900 192.0.2.200 GET d111111abcdef8.cloudfront.net / 502
 - Mozilla/5.0%20(Windows%20NT%2010.0;%20Win64;%20x64)%20AppleWebKit/537.36%20(KHTML,
%20like%20Gecko)%20Chrome/78.0.3904.108%20Safari/537.36 - - Error
 3AqrZGCnF_g0-5KOvfA7c9XLcf4YGvMFSeFdIetR1N_2y8jSis8Zxg== www.example.com http 735
 0.107 - - - Error HTTP/1.1 - - 3802 0.107 OriginDnsError text/html 507 - -
2019-12-13 22:37:02 SEA19-C2 900 192.0.2.200 GET d111111abcdef8.cloudfront.net / 502
 - curl/7.55.1 - - Error kBkDzGnceVtWHqSCqBUqtA_cEs2T3tFUBbnBNkB9El_uVRhHgcZfcw==
 www.example.com http 387 0.103 - - - Error HTTP/1.1 - - 12644 0.103 OriginDnsError
 text/html 507 - -
```

Analysieren Sie Protokolle

Da Sie mehrere Zugriffsprotokolle pro Stunde erhalten können, empfehlen wir, alle für einen bestimmten Zeitraum erhaltenen Protokolldateien in einer Datei zusammenzufassen. Sie können die Daten für diesen Zeitraum dann genauer und vollständig analysieren.

Eine Möglichkeit, Ihre Zugriffsprotokolle zu analysieren, ist die Verwendung von Amazon Athena;. Athena ist ein interaktiver Abfragedienst, mit dem Sie Daten für AWS Dienste analysieren können, darunter CloudFront. Weitere Informationen finden Sie unter Abfragen von CloudFront Amazon-Protokollen im Amazon Athena-Benutzerhandbuch.

Darüber hinaus werden in den folgenden AWS Blogbeiträgen einige Möglichkeiten zur Analyse von Zugriffsprotokollen erörtert.

- CloudFront Amazon-Anforderungsprotokollierung (für Inhalte, die über HTTP geliefert werden)
- Verbesserte CloudFront Protokolle, jetzt mit Abfragezeichenfolgen

Verwenden Sie Echtzeitprotokolle

Mit CloudFront Echtzeitprotokollen können Sie Informationen über Anfragen an eine Distribution in Echtzeit abrufen (Protokolle werden innerhalb von Sekunden nach Eingang der Anfragen zugestellt). Sie können Echtzeitprotokolle verwenden, um basierend auf der Leistung der Bereitstellung von Inhalten Überwachungsaktionen und Analysen auszuführen und Maßnahmen zu ergreifen.

CloudFront Echtzeitprotokolle sind konfigurierbar. Sie können wählen:

- Die Abtastrate für Ihre Echtzeit-Protokolle – d. h. der Prozentsatz der Anforderungen, für die Sie Echtzeit-Protokolldatensätze erhalten möchten.
- Die spezifischen Felder, die Sie in den Protokolldatensätzen empfangen möchten.
- Das spezifische Cache-Verhalten (Pfadmuster), für das Sie Echtzeit-Protokolle erhalten möchten.

CloudFront Echtzeitprotokolle werden an den Datenstream Ihrer Wahl in Amazon Kinesis Data Streams übermittelt. Sie können Ihren eigenen Kinesis Data Stream Consumer erstellen oder Amazon Data Firehose verwenden, um die Protokolldaten an Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon OpenSearch Service (Service) oder einen OpenSearch Protokollverarbeitungsservice eines Drittanbieters zu senden.

CloudFront Gebühren für Echtzeitprotokolle, zusätzlich zu den Gebühren, die Ihnen für die Nutzung von Kinesis Data Streams entstehen. Weitere Informationen zu den Preisen finden Sie unter CloudFront Amazon-Preise und Amazon Kinesis Data Streams-Preise.

> **Important**
>
> Wir empfehlen Ihnen, die Protokolle zu verwenden, um die Art der Anfragen nach Ihren Inhalten zu verstehen, und nicht, um alle Anfragen vollständig zu erfassen. CloudFront liefert Protokolle in Echtzeit nach bestem Wissen und Gewissen. Der Protokolleintrag für eine bestimmte Anfrage wird möglicherweise viel später übermittelt, als die Anfrage tatsächlich verarbeitet wurde; in seltenen Fällen kann es auch sein, dass ein Protokolleintrag gar nicht übermittelt wird. Wenn ein Protokolleintrag in den Echtzeitprotokollen weggelassen wird, entspricht die Anzahl der Einträge in den Echtzeitprotokollen nicht der Nutzung, die in den AWS Abrechnungs- und Nutzungsberichten angegeben ist.

Themen

Erstellen und verwenden Sie Echtzeit-Protokollkonfigurationen

Um Informationen über Anfragen an eine Distribution in Echtzeit zu erhalten, können Sie Protokollkonfigurationen in Echtzeit verwenden. Protokolle werden innerhalb von Sekunden nach Eingang der Anfragen zugestellt. Sie können eine Echtzeit-Protokollkonfiguration in der CloudFront Konsole, mit der AWS Command Line Interface (AWS CLI) oder mit der CloudFront API erstellen.

Um eine Echtzeit-Protokollkonfiguration zu verwenden, fügen Sie sie einem oder mehreren Cache-Verhalten in einer CloudFront Distribution hinzu.

Console

Um eine Echtzeit-Protokollkonfiguration zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Seite Logs in der CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home?#/logs.

2. Wählen Sie den Tab Echtzeitkonfigurationen.
3. Wählen Sie Create configuration (Konfiguration erstellen).
4. Geben Sie unter Name einen Namen für die Konfiguration ein.
5. Geben Sie unter Samplingrate den Prozentsatz der Anfragen ein, für die Sie Protokolldatensätze erhalten möchten.
6. Wählen Sie unter Felder die Felder aus, die Sie in den Echtzeitprotokollen erhalten möchten.
   - Um alle CMCD-Felder in Ihre Logs aufzunehmen, wählen Sie CMCD all keys aus.
7. Wählen Sie für Endpoint einen oder mehrere Kinesis-Datenstreams aus, um Echtzeitprotokolle zu erhalten.

   > **Note**
   >
   > CloudFront Echtzeitprotokolle werden an den Datenstrom übermittelt, den Sie in Kinesis Data Streams angeben. Um Ihre Echtzeit-Logs zu lesen und zu analysieren, können Sie Ihren eigenen Kinesis Data Stream Consumer erstellen. Sie können Firehose auch verwenden, um die Protokolldaten an Amazon S3, Amazon Redshift, Amazon OpenSearch Service oder einen Protokollverarbeitungsdienst eines Drittanbieters zu senden.

8. Wählen Sie für die IAM-Rolle die Option Neue Servicerolle erstellen oder wählen Sie eine bestehende Rolle aus. Sie müssen über die Berechtigung zum Erstellen von IAM-Rollen verfügen.
9. (Optional) Wählen Sie für CloudFrontVerteilung ein Verteilungs- und Cache-Verhalten aus, das an die Echtzeit-Protokollkonfiguration angehängt werden soll.
10. Wählen Sie Create configuration (Konfiguration erstellen).

Bei erfolgreicher Ausführung zeigt die Konsole die Details der soeben erstellten Echtzeit-Protokollkonfiguration an.

Weitere Informationen finden Sie unter Machen Sie sich mit Echtzeit-Protokollkonfigurationen vertraut.

AWS CLI

Verwenden Sie den aws cloudfront create-realtime-log-config Befehl AWS CLI, um eine Echtzeit-Protokollkonfiguration mit dem zu erstellen. Sie können die Eingabeparameter des Befehls in

einer Eingabedatei bereitstellen, anstatt jeden einzelnen Parameter als Befehlszeileneingabe anzugeben.

So erstellen Sie eine Echtzeit-Protokollkonfiguration (CLI mit Eingabedatei):

1. Verwenden Sie den folgenden Befehl, um eine Datei mit dem Namen `rtl-config.yaml` zu erstellen, die alle Eingabeparameter für den create-realtime-log-config-Befehl enthält.

```
aws cloudfront create-realtime-log-config --generate-cli-skeleton yaml-input >
 rtl-config.yaml
```

2. Öffnen Sie die Datei mit dem Namen `rtl-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, um die gewünschten Einstellungen für die Echtzeit-Protokollkonfiguration anzugeben, und speichern Sie die Datei. Beachten Sie Folgendes:

   - Für `StreamType` ist der einzige gültige Wert `Kinesis`.

   Weitere Informationen zu den Echtzeit-Konfigurationseinstellungen im Langformat finden Sie unter Machen Sie sich mit Echtzeit-Protokollkonfigurationen vertraut.

3. Verwenden Sie den folgenden Befehl, um die Echtzeit-Protokollkonfiguration mithilfe von Eingabeparametern aus der `rtl-config.yaml`-Datei zu erstellen.

```
aws cloudfront create-realtime-log-config --cli-input-yaml file://rtl-
config.yaml
```

Bei erfolgreicher Ausführung zeigt die Ausgabe des Befehls die Details der soeben erstellten Echtzeit-Protokollkonfiguration an.

So fügen Sie eine Echtzeit-Protokollkonfiguration an eine vorhandene Verteilung an (CLI mit Eingabedatei):

1. Verwenden Sie den folgenden Befehl, um die Verteilungskonfiguration für die CloudFront Distribution zu speichern, die Sie aktualisieren möchten. Ersetzen Sie *`distribution_ID`* durch die ID der Verteilung.

```
aws cloudfront get-distribution-config --id distribution_ID --output yaml >
 dist-config.yaml
```

2. Öffnen Sie die Datei mit dem Namen `dist-config.yaml`, die Sie gerade erstellt haben. Bearbeiten Sie die Datei, indem Sie die folgenden Änderungen an jeder Cache-Verhaltensweise vornehmen, die Sie aktualisieren, um eine Echtzeit-Protokollkonfiguration zu verwenden.

   - Fügen Sie in der Cache-Verhaltensweise ein Feld mit dem Namen hinz `RealtimeLogConfigArn`. Verwenden Sie für den Wert des Feldes den ARN der Echtzeit-Protokollkonfiguration, die Sie dieser Cache-Verhaltensweise anfügen möchten.
   - Benennen Sie das Feld `ETag` in `IfMatch` um, ändern Sie jedoch nicht den Wert des Feldes.

   Speichern Sie die Datei, wenn Sie fertig sind.

3. Verwenden Sie den folgenden Befehl, um die Verteilung so zu aktualisieren, dass die Echtzeit-Protokollkonfiguration verwendet wird. Ersetzen Sie *`distribution_ID`* durch die ID der Verteilung.

```
aws cloudfront update-distribution --id distribution_ID --cli-input-yaml file://
dist-config.yaml
```

Wenn dies erfolgreich ist, zeigt die Ausgabe des Befehls die Details der gerade aktualisierten Verteilung an.

API

Verwenden Sie den API-Vorgang, um eine Echtzeit-Protokollkonfiguration mit der CloudFront CreateRealtimeLogConfigAPI zu erstellen. Weitere Informationen zu den Parametern, die Sie in diesem API-Aufruf angeben, finden Machen Sie sich mit Echtzeit-Protokollkonfigurationen vertraut Sie in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Nachdem Sie eine Echtzeit-Protokollkonfiguration erstellt haben, können Sie sie mithilfe einer der folgenden API-Operationen an ein Cache-Verhalten anhängen:

- Um es an ein Cache-Verhalten in einer vorhandenen Distribution anzuhängen, verwenden Sie UpdateDistribution.

- Um es an ein Cache-Verhalten in einer neuen Distribution anzuhängen, verwenden Sie CreateDistribution.

Geben Sie für diese beiden API-Operationen den ARN der Echtzeit-Protokollkonfiguration vor `RealtimeLogConfigArn` Ort innerhalb eines Cache-Verhaltens an. Weitere Informationen zu den anderen Feldern, die Sie in diesen API-Aufrufen angeben, finden Referenz für alle Verteilungseinstellungen Sie in der API-Referenzdokumentation für Ihr AWS SDK oder einen anderen API-Client.

Machen Sie sich mit Echtzeit-Protokollkonfigurationen vertraut

Um CloudFront Echtzeitprotokolle zu verwenden, erstellen Sie zunächst eine Echtzeit-Protokollkonfiguration. Die Echtzeit-Protokollkonfiguration enthält Informationen darüber, welche Protokollfelder Sie empfangen möchten, die Abtastrate für Protokolldatensätze und den Kinesis-Datenstrom, in den Sie die Protokolle bereitstellen möchten.

Insbesondere enthält eine Echtzeit-Protokollkonfiguration die folgenden Einstellungen:

Inhalt

- Name
- Abtastrate
- Felder
- Endpunkt (Kinesis Data Streams)
- IAM-Rolle

Name

Ein Name zur Identifizierung der Echtzeit-Protokollkonfiguration.

Abtastrate

Die Abtastrate ist eine ganze Zahl zwischen 1 und 100 (einschließlich), die den Prozentsatz der Viewer-Anfragen bestimmt, die an Kinesis Data Streams als Echtzeit-Protokolldatensätze gesendet werden. Um jede Viewer-Anforderung in Ihre Echtzeit-Protokolle aufzunehmen, geben Sie 100 für die Abtastrate an. Sie können eine niedrigere Abtastrate wählen, um die Kosten zu senken, während Sie dennoch eine repräsentative Stichprobe von Anforderungsdaten in Ihren Echtzeit-Protokollen erhalten.

Felder

Eine Liste der Felder, die in jedem Echtzeit-Protokolldatensatz enthalten sind. Jeder Protokolldatensatz kann bis zu 40 Felder enthalten und Sie können auswählen, ob alle verfügbaren Felder empfangen werden, oder nur die Felder, die Sie für die Überwachung und Analyse der Leistung benötigen.

Die folgende Liste enthält jeden Feldnamen und eine Beschreibung der Informationen in diesem Feld. Die Felder werden in der Reihenfolge aufgelistet, in der sie in den Protokolldatensätzen angezeigt werden, die an Kinesis Data Streams übermittelt werden.

Die Felder 46-63 sind Common Media Client Data (CMCD), an die Media Player-Clients CDNs bei jeder Anfrage senden können. Sie können diese Daten verwenden, um jede Anfrage zu verstehen, z. B. den Medientyp (Audio, Video), die Wiedergabegeschwindigkeit und die Streaming-Länge. Diese Felder erscheinen nur dann in Ihren Echtzeitprotokollen, wenn sie an gesendet werden CloudFront.

1. **`timestamp`**

   Die Angabe zu Datum und Uhrzeit, an der der Edge-Server die Reaktion auf die Anforderung abgeschlossen hat.

2. **`c-ip`**

   Die IP-Adresse des Betrachters, die der Anfrage gestellt hat, z. B. `192.0.2.183` oder `2001:0db8:85a3::8a2e:0370:7334`. Wenn der Viewer einen HTTP-Proxy oder eine Load Balancer verwendet hat, um die Anforderung zu senden, entspricht der Wert dieses Feldes der IP-Adresse des Proxys bzw. des Load Balancers. Siehe auch das Feld `x-forwarded-for`.

3. **`s-ip`**

   Die IP-Adresse des CloudFront Servers, der die Anfrage bearbeitet hat, z. B. `192.0.2.183` oder`2001:0db8:85a3::8a2e:0370:7334`.

4. **`time-to-first-byte`**

   Die Anzahl der Sekunden zwischen dem Empfangen der Anforderung und dem Schreiben des ersten Bytes der Antwort, gemessen auf dem Server.

5. **`sc-status`**

   Den HTTP-Statuscode der Antwort des Servers (z. B. `200`).

6. **`sc-bytes`**

Die Gesamtzahl der Bytes, die der Server als Antwort auf die Anforderung an den Viewer übermittelt hat, einschließlich Headern. Für WebSocket und gRPC-Verbindungen ist dies die Gesamtzahl der Byte, die vom Server über die Verbindung an den Client gesendet werden.

7. **`cs-method`**

   Die vom Viewer empfangene HTTP-Anforderungsmethode.

8. **`cs-protocol`**

   Das Protokoll der Viewer-Anforderung (`http`, `https`, `grpcs`, `ws` oder `wss`).

9. **`cs-host`**

   Der Wert, den der Viewer in den `Host`-Header der Anforderung eingefügt hat.
   Wenn Sie den CloudFront Domainnamen in Ihrem Objekt verwenden URLs (z. B. d111111abcdef8.cloudfront.net), enthält dieses Feld diesen Domainnamen. Wenn Sie alternative Domainnamen (CNAMEs) in Ihrem Objekt verwenden URLs (z. B. www.example.com), enthält dieses Feld den alternativen Domainnamen.

10. **`cs-uri-stem`**

    Die gesamte Anforderungs-URL, einschließlich der Abfragezeichenfolge (falls vorhanden), jedoch ohne den Domänennamen. Beispiel, `/images/cat.jpg?mobile=true`.

    > **Note**
    >
    > In Standardprotokollen enthält der `cs-uri-stem`-Wert nicht die Abfragezeichenfolge.

11. **`cs-bytes`**

    Die Gesamtzahl der Bytes an Daten, die in der Anforderung des Viewers einschließlich Headern enthalten sind. Für WebSocket und gRPC-Verbindungen ist dies die Gesamtzahl der Byte, die vom Client an den Server über die Verbindung gesendet wurden.

12. **`x-edge-location`**

    Der Edge-Standort, an dem die Anfrage verarbeitet wurde. Jeder Edge-Standort wird durch einen aus drei Buchstaben bestehenden Code und eine willkürlich zugewiesene Zahl identifiziert (z. B.). DFW3 Der Code aus drei Buchstaben entspricht in der Regel dem Code der International Air Transport Association (IATA) für einen Flughafen in der Nähe des Edge-Standorts. (Diese Abkürzungen ändern sich möglicherweise in der Zukunft.)

13. **`x-edge-request-id`**

    Eine undurchsichtige Zeichenfolge, die eine Anfrage eindeutig identifiziert. CloudFront sendet diese Zeichenfolge auch im `x-amz-cf-id` Antwort-Header.

14. **`x-host-header`**

    Der Domainname der CloudFront Distribution (z. B. d111111abcdef8.cloudfront.net).

15. **`time-taken`**

    Die Anzahl der Sekunden (auf die Tausendstelsekunde genau, z. B. 0,082) ab dem Zeitpunkt, an dem der Server die Anforderung des Viewers empfängt, bis zu dem Zeitpunkt, an dem der Server das letzte Byte der Antwort in die Ausgabewarteschlange schreibt, wie auf dem Server gemessen. Aus der Perspektive der Viewers vergeht bis zum Empfang der gesamten Antwort aufgrund der Netzwerklatenz und des TCP-Puffervorgangs insgesamt mehr Zeit, als mit diesem Wert angegeben wird.

16. **`cs-protocol-version`**

    Die HTTP-Version, die der Viewer in der Anfrage angegeben hat: Mögliche Werte sind `HTTP/0.9`, `HTTP/1.0`, `HTTP/1.1`, `HTTP/2.0` und `HTTP/3.0`.

17. **`c-ip-version`**

    Die IP-Version der Anfrage (oder). IPv4 IPv6

18. **`cs-user-agent`**

    Der Wert für den `User-Agent`-Header in der Anfrage. Der `User-Agent`-Header bezeichnet die Quelle für die Anforderung, z. B. den Gerätetyp und den Browser, der die Anforderung abgesendet hat, oder die Suchmaschine, wenn die Anforderung von einer Suchmaschine stammt.

19. **`cs-referer`**

    Der Wert für den `Referer`-Header in der Anfrage. Der Name der Domäne, von der die Anforderung ausgegangen ist. Häufig vorkommende Referrer sind Suchmaschinen, andere Websites, die direkt auf Ihre Objekte verlinken, und Ihre eigene Website.

20. **`cs-cookie`**

    Der `Cookie`-Header in der Anforderung einschließlich der Name-Wert-Paaren und zugehörigen Attributen.

> **Note**
>
> Dieses Feld wird auf 800 Bytes abgeschnitten.

21 **`cs-uri-query`**

Der Teil der Anforderungs-URL mit der Abfragezeichenfolge, wenn vorhanden.

22 **`x-edge-response-result-type`**

Die Art, wie der Server die Antwort direkt vor der Rücksendung der Antwort an den Viewer klassifiziert hat. Siehe auch das Feld `x-edge-result-type`. Mögliche Werte sind:

- `Hit` – Der Server hat das Objekt aus dem Cache für den Betrachter bereitgestellt.
- `RefreshHit` – Der Server hat das Objekt im Edge-Zwischenspeicher gefunden, es war jedoch abgelaufen. Daher nahm der Server Kontakt mit dem Ursprung auf, um zu überprüfen, ob der Zwischenspeicher die neueste Version des Objekts enthalten hatte.
- `Miss` – Die Anforderung konnte nicht durch ein Objekt im Zwischenspeicher bedient werden. Daher hat der Server die Anforderung an den Ursprungs-Server weitergeleitet und das Ergebnis an den Betrachter ausgegeben.
- `LimitExceeded`— Die Anfrage wurde abgelehnt, weil ein CloudFront Kontingent (früher als Limit bezeichnet) überschritten wurde.
- `CapacityExceeded` – Der Server hat den Fehler 503 zurückgegeben, da er zum Zeitpunkt der Anforderung nicht über genügend Kapazitäten verfügte, um das Objekt bereitzustellen.
- `Error` – In der Regel bedeutet dies, dass die Anforderung zu einem Client-Fehler (d. h. der Wert des Felds `sc-status` liegt im `4xx`-Bereich) oder zu einem Serverfehler geführt hat (d. h. der Wert des Felds `sc-status` liegt im `5xx`-Bereich).

  Wenn der Wert des Felds `x-edge-result-type` `Error` ist und der Wert dieses Felds nicht `Error` ist, wurde die Verbindung vor dem Abschluss des Downloads durch den Client getrennt.
- `Redirect` – Der Server hat den Betrachter entsprechend den Verteilungseinstellungen von HTTP zu HTTPS umgeleitet.
- `LambdaExecutionError`— Die der Distribution zugeordnete Lambda @Edge -Funktion wurde aufgrund einer fehlerhaften Zuordnung, eines Funktionstimeouts, eines AWS Abhängigkeitsproblems oder eines anderen allgemeinen Verfügbarkeitsproblems nicht abgeschlossen.

23 **`x-forwarded-for`**

Wenn der Viewer einen HTTP-Proxy oder Load Balancer verwendet hat, um die Anforderung zu senden, entspricht der Wert des Felds `c-ip` der IP-Adresse des Proxys oder Load Balancers. In diesem Fall stellt dieses Feld die IP-Adresse des Viewers dar, von dem die Anfrage stammt. Dieses Feld kann mehrere durch Kommas getrennte IP-Adressen enthalten. Jede IP-Adresse kann eine IPv4 Adresse (zum Beispiel`192.0.2.183`) oder eine IPv6 Adresse (zum Beispiel) sein. `2001:0db8:85a3::8a2e:0370:7334`

24 **`ssl-protocol`**

Wenn für die Anfrage HTTPS verwendet wurde, enthält dieses Feld das SSL/TLS Protokoll, das der Betrachter und der Server für die Übertragung der Anfrage und Antwort ausgehandelt haben. Eine Liste der möglichen Werte finden Sie unter den unterstützten SSL/TLS ProtokollenUnterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront.

25 **`ssl-cipher`**

Wenn für die Anfrage HTTPS verwendet wurde, enthält dieses Feld die SSL/TLS Chiffre, die der Betrachter und der Server für die Verschlüsselung der Anfrage und Antwort ausgehandelt haben. Eine Liste der möglichen Werte finden Sie unter den unterstützten SSL/TLS Verschlüsselungen. Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront

26 **`x-edge-result-type`**

Art der Klassifizierung der Antwort durch den Server, nachdem das letzte Byte den Server verlassen hat. Manchmal wird der Ergebnistyp zwischen dem Zeitpunkt, an dem Server zum Senden der Antwort bereit ist, und dem Zeitpunkt, an dem er das Senden der Antwort abgeschlossen hat, geändert. Siehe auch das Feld `x-edge-response-result-type`.

Angenommen, im HTTP-Streaming findet der Server ein Segment des Streams im Cache. In diesem Szenario würde der Wert dieses Feldes normalerweise sei `Hit`. Wenn der Viewer jedoch die Verbindung schließt, bevor der Server das ganze Segment übermittelt hat, ist der endgültige Ergebnistyp (und damit der Wert dieses Felds) `Error`.

WebSocket und gRPC-Verbindungen haben `Miss` für dieses Feld den Wert von, da der Inhalt nicht zwischengespeichert werden kann und direkt an den Ursprung weitergeleitet wird.

Mögliche Werte sind:

- `Hit` – Der Server hat das Objekt aus dem Cache für den Betrachter bereitgestellt.

- `RefreshHit` – Der Server hat das Objekt im Edge-Zwischenspeicher gefunden, es war jedoch abgelaufen. Daher nahm der Server Kontakt mit dem Ursprung auf, um zu überprüfen, ob der Zwischenspeicher die neueste Version des Objekts enthalten hatte.
- `Miss` – Die Anforderung konnte nicht durch ein Objekt im Zwischenspeicher bedient werden. Daher hat der Server die Anforderung an den Ursprung weitergeleitet und das Ergebnis an den Betrachter ausgegeben.
- `LimitExceeded`— Die Anfrage wurde abgelehnt, weil ein CloudFront Kontingent (früher als Limit bezeichnet) überschritten wurde.
- `CapacityExceeded` – Der Server hat den HTTP-Statuscode 503 zurückgegeben, da er zum Zeitpunkt der Anforderung nicht über genügend Kapazitäten für die Bereitstellung des Objekts verfügte.
- `Error` – In der Regel bedeutet dies, dass die Anforderung zu einem Client-Fehler (d. h. der Wert des Felds `sc-status` liegt im `4xx`-Bereich) oder zu einem Serverfehler geführt hat (d. h. der Wert des Felds `sc-status` liegt im `5xx`-Bereich). Wenn der Wert des Felds `sc-status` `200` oder wenn der Wert dieses Felds `Error` ist und der Wert des Felds `x-edge-response-result-type` nicht `Error` ist, war die HTTP-Anforderung erfolgreich. Die Client-Verbindung wurde jedoch getrennt, bevor alle Bytes empfangen wurden.
- `Redirect` – Der Server hat den Betrachter entsprechend den Verteilungseinstellungen von HTTP zu HTTPS umgeleitet.
- `LambdaExecutionError`— Die der Distribution zugeordnete Lambda @Edge -Funktion wurde aufgrund einer fehlerhaften Zuordnung, eines Funktionstimeouts, eines AWS Abhängigkeitsproblems oder eines anderen allgemeinen Verfügbarkeitsproblems nicht abgeschlossen.

27. **`fle-encrypted-fields`**

    Die Anzahl der Felder für die Verschlüsselung auf Feldebene, die der Server verschlüsselt und an den Ursprung weitergeleitet hat. CloudFront Server streamen die verarbeitete Anfrage an den Ursprung, während sie Daten verschlüsseln, sodass dieses Feld auch dann einen Wert haben kann, wenn der Wert von ein Fehler `fle-status` ist.

28. **`fle-status`**

    Wenn die Verschlüsselung auf Feldebene für eine Verteilung konfiguriert ist, enthält dieses Feld einen Code, der angibt, ob der Anforderungstext erfolgreich verarbeitet wurde. Wenn der Server den Anforderungstext erfolgreich verarbeitet, Werte in den angegebenen Feldern verschlüsselt und die Anforderung an den Ursprung weiterleitet, ist der Wert dieses Felds `Processed`. Der

Wert von `x-edge-result-type` kann in diesem Fall immer noch auf einen clientseitigen oder serverseitigen Fehler hinweisen.

Mögliche Werte für dieses Feld sind:

- `ForwardedByContentType` – Der Server hat die Anforderung ohne Parsing oder Verschlüsselung an den Ursprung weitergeleitet, da kein Content-Typ konfiguriert wurde.
- `ForwardedByQueryArgs` – Der Server hat die Anforderung ohne Parsing oder Verschlüsselung an den Ursprung weitergeleitet, da die Anforderung ein Abfrageargument enthält, das nicht in der Konfiguration für die Verschlüsselung auf Feldebene enthalten war.
- `ForwardedDueToNoProfile` – Der Server hat die Anforderung ohne Parsing oder Verschlüsselung an den Ursprung weitergeleitet, da in der Konfiguration für die Verschlüsselung auf Feldebene kein Profil angegeben wurde.
- `MalformedContentTypeClientError` – Der Server hat die Anforderung zurückgewiesen und einen HTTP-400-Statuscode an den Betrachter zurückgegeben, da der Wert des `Content-Type`-Headers ein ungültiges Format hatte.
- `MalformedInputClientError` – Der Server hat die Anforderung zurückgewiesen und einen HTTP 400-Statuscode an den Betrachter zurückgegeben, da der Anforderungstext ein ungültiges Format hatte.
- `MalformedQueryArgsClientError` – Der Server hat die Anforderung zurückgewiesen und einen HTTP-400-Statuscode an den Betrachter zurückgegeben, weil ein Abfrageargument leer war oder ein ungültiges Format hatte.
- `RejectedByContentType` – Der Server hat die Anforderung zurückgewiesen und einen HTTP-400-Statuscode an den Betrachter zurückgegeben, da in der Konfiguration für die Verschlüsselung auf Feldebene kein Content-Typ angegeben wurde.
- `RejectedByQueryArgs` – Der Server hat die Anforderung zurückgewiesen und einen HTTP-400-Statuscode an den Betrachter zurückgegeben, da in der Konfiguration für die Verschlüsselung auf Feldebene kein Abfrageargument angegeben wurde.
- `ServerError` – Der Ursprungs-Server hat einen Fehler zurückgegeben.

Wenn die Anforderung das Kontingent für die Verschlüsselung auf Feldebene überschreitet (früher als Limit bezeichnet), enthält dieses Feld einen der folgenden Fehlercodes und der Server gibt dem Viewer den HTTP-Statuscode 400 zurück. Eine Liste der aktuellen Kontingente für die Verschlüsselung auf Feldebene finden Sie unter Kontingente für Verschlüsselung auf Feldebene.

- `FieldLengthLimitClientError` – Ein Feld, das für die Verschlüsselung konfiguriert ist, hat die maximal zulässige Länge überschritten.

- `FieldNumberLimitClientError` – Eine Anforderung, die die Verteilung verschlüsseln soll, enthält mehr Felder als zulässig.
- `RequestLengthLimitClientError` – Die Länge des Anfragetexts hat die maximal zulässige Länge, wenn die Verschlüsselung auf Feldebene konfiguriert ist, überschritten.

29 **`sc-content-type`**

Der Wert des HTTP `Content-Type`-Headers der Antwort.

30 **`sc-content-len`**

Der Wert des HTTP `Content-Length`-Headers der Antwort.

31 **`sc-range-start`**

Wenn die Antwort den HTTP `Content-Range`-Header enthält, enthält dieses Feld den Bereichsstartwert.

32 **`sc-range-end`**

Wenn die Antwort den HTTP `Content-Range`-Header enthält, enthält dieses Feld den Bereichsendwert.

33 **`c-port`**

Die Portnummer der Anforderung des Viewers.

34 **`x-edge-detailed-result-type`**

Dieses Feld enthält den gleichen Wert wie der Wert für das `x-edge-result-type`-Feld, außer in den folgenden Fällen:

- Wenn das Objekt dem Viewer aus der Origin-Shield-Ebene bereitgestellt wurde, enthält dieses Feld `OriginShieldHit`.
- Wenn sich das Objekt nicht im CloudFront Cache befand und die Antwort von einer Lambda @Edge -Funktion für die ursprüngliche Anfrage generiert wurde, enthält `MissGeneratedResponse` dieses Feld.
- Wenn der Wert des `x-edge-result-type`-Felds `Error` lautet, enthält dieses Feld einen der folgenden Werte mit weiteren Informationen zum Fehler:
  - `AbortedOrigin` – Der Server hat ein Problem mit dem Ursprung festgestellt.
  - `ClientCommError` – Die Antwort auf den Betrachter wurde aufgrund eines Kommunikationsproblems zwischen dem Server und dem Betrachter unterbrochen.

- `ClientGeoBlocked` – Die Verteilung ist so konfiguriert, dass Anforderungen vom geografischen Standort des Viewers abgelehnt werden.
- `ClientHungUpRequest` — Der Betrachter wurde beim Senden der Anfrage vorzeitig gestoppt.
- `Error` – Es ist ein Fehler aufgetreten, dessen Fehlertyp zu keiner der anderen Kategorien passt. Dieser Fehlertyp kann auftreten, wenn der Server eine Fehlerantwort aus dem Cache ausgibt.
- `InvalidRequest` – Der Server hat eine ungültige Anforderung vom Betrachter erhalten.
- `InvalidRequestBlocked` — Der Zugriff auf die angeforderte Ressource ist blockiert.
- `InvalidRequestCertificate`— Die Verteilung stimmt nicht mit dem SSL/TLS Zertifikat überein, für das die HTTPS-Verbindung hergestellt wurde.
- `InvalidRequestHeader` – Die Anforderung enthielt einen ungültigen Header.
- `InvalidRequestMethod` — Die Verteilung ist nicht für eine Verarbeitung der verwendeten HTTP-Anforderungsmethode konfiguriert. Dies kann vorkommen, wenn die Verteilung nur Anforderungen unterstützt, die zwischengespeichert werden können.
- `OriginCommError` – Bei der Anforderung ist eine Zeitüberschreitung aufgetreten, während eine Verbindung mit dem Ursprung hergestellt wurde oder Daten aus dem Ursprung gelesen wurden.
- `OriginConnectError` – Der Server konnte keine Verbindung zum Ursprung herstellen.
- `OriginContentRangeLengthError` – Der `Content-Length`-Header in der Antwort des Ursprungs stimmt nicht mit der Länge im `Content-Range`-Header überein.
- `OriginDnsError` – Der Server konnte den Domänennamen des Ursprungs nicht auflösen.
- `OriginError` — Der Ursprung gab eine falsche Antwort zurück.
- `OriginHeaderTooBigError` – Ein vom Ursprung zurückgegebener Header ist für eine Verarbeitung durch den Edge-Server zu groß.
- `OriginInvalidResponseError` — Der Ursprung gab eine ungültige Antwort zurück.
- `OriginReadError` – Der Server konnte nicht vom Ursprung lesen.
- `OriginWriteError` – Der Server konnte nicht in den Ursprung schreiben.
- `OriginZeroSizeObjectError` — Ein Nullgrößenobjekt, das vom Ursprung gesendet wurde, führte zu einem Fehler.
- `SlowReaderOriginError` — Der Betrachter hat die Nachricht, die den Ursprungsfehler verursacht hat, nur langsam gelesen.

35. **`c-country`**

    Ein Ländercode, der den geografischen Standort des Viewers darstellt, der von der IP-Adresse des Viewers festgelegt wird. Die Liste der Ländercodes finden Sie unter ISO 3166-1 alpha-2.

36. **`cs-accept-encoding`**

    Der Wert für den `Accept-Encoding`-Header in der Viewer-Anforderung.

37. **`cs-accept`**

    Der Wert für den `Accept`-Header in der Viewer-Anforderung.

38. **`cache-behavior-path-pattern`**

    Das Pfadmuster, das das Cache-Verhalten identifiziert, das der Viewer-Anforderung entspricht.

39. **`cs-headers`**

    Die HTTP-Header (Namen und Werte) in der Viewer-Anforderung.

    > **Note**
    >
    > Dieses Feld wird auf 800 Bytes abgeschnitten.

40. **`cs-header-names`**

    Die Namen der HTTP-Header (keine Werte) in der Viewer-Anforderung.

    > **Note**
    >
    > Dieses Feld wird auf 800 Bytes abgeschnitten.

41. **`cs-headers-count`**

    Die Anzahl der HTTP-Header in der Viewer-Anforderung.

42. **`primary-distribution-id`**

    Wenn Continuous Deployment aktiviert ist, identifiziert diese ID, welche Distribution in der aktuellen Distribution die primäre ist.

43. **`primary-distribution-dns-name`**

Wenn Continuous Deployment aktiviert ist, zeigt dieser Wert den primären Domainnamen an, der sich auf die aktuelle CloudFront Distribution bezieht (z. B. d111111abcdef8.cloudfront.net).

44. **`origin-fbl`**

    Die Anzahl der Sekunden der First-Byte-Latenz zwischen und Ihrem Ursprung. CloudFront

45. **`origin-lbl`**

    Die Anzahl der Sekunden der Latenz im letzten Byte zwischen CloudFront und Ihrem Ursprung.

46. **`asn`**

    Die autonome Systemnummer (ASN) des Viewers.

47.

    > **CMCD-Felder in Echtzeitprotokollen**
    >
    > Weitere Informationen zu diesen Feldern finden Sie im Dokument CTA Specification Web Application Video Ecosystem — Common Media Client Data CTA-5004.

48. **`cmcd-encoded-bitrate`**

    Die kodierte Bitrate des angeforderten Audio- oder Videoobjekts.

49. **`cmcd-buffer-length`**

    Die Pufferlänge des angeforderten Medienobjekts.

50. **`cmcd-buffer-starvation`**

    Ob der Puffer irgendwann zwischen der vorherigen Anforderung und der Objektanforderung ausgelaugt wurde. Dies kann dazu führen, dass sich der Player in einem Pufferstatus befindet, was die Video- oder Audiowiedergabe zum Erliegen bringen kann.

51. **`cmcd-content-id`**

    Eine eindeutige Zeichenfolge, die den aktuellen Inhalt identifiziert.

52. **`cmcd-object-duration`**

    Die Wiedergabedauer des angeforderten Objekts (in Millisekunden).

53. **`cmcd-deadline`**

Der Stichtag ab dem Zeitpunkt der Anforderung, zu dem das erste Beispiel dieses Objekts verfügbar sein muss, damit ein Unterlauf des Puffers oder andere Wiedergabeprobleme vermieden werden.

54. **`cmcd-measured-throughput`**

    Der vom Client gemessene Durchsatz zwischen Client und Server.

55. **`cmcd-next-object-request`**

    Der relative Pfad des nächsten angeforderten Objekts.

56. **`cmcd-next-range-request`**

    Wenn es sich bei der nächsten Anfrage um eine teilweise Objektanforderung handelt, gibt diese Zeichenfolge den Bytebereich an, der angefordert werden soll.

57. **`cmcd-object-type`**

    Der Medientyp des aktuell angeforderten Objekts.

58. **`cmcd-playback-rate`**

    1 bei Echtzeit, 2 bei doppelter Geschwindigkeit, 0 bei Nichtwiedergabe.

59. **`cmcd-requested-maximum-throughput`**

    Der angeforderte maximale Durchsatz, den der Kunde für die Bereitstellung von Ressourcen für ausreichend erachtet.

60. **`cmcd-streaming-format`**

    Das Streaming-Format, das die aktuelle Anfrage definiert.

61. **`cmcd-session-id`**

    Eine GUID, die die aktuelle Wiedergabesitzung identifiziert.

62. **`cmcd-stream-type`**

    Token, das die Verfügbarkeit von Segmenten identifiziert. `v`= alle Segmente sind verfügbar. `l`= Segmente werden im Laufe der Zeit verfügbar.

63. **`cmcd-startup`**

    Der Schlüssel wird ohne Wert angegeben, wenn das Objekt beim Start, bei der Suche oder bei der Wiederherstellung nach einem Ereignis, bei dem der Puffer leer ist, dringend benötigt wird.

64. **`cmcd-top-bitrate`**

    Die Wiedergabeversion mit der höchsten Bitrate, die der Client abspielen kann.

65. **`cmcd-version`**

    Die Version dieser Spezifikation, die für die Interpretation der definierten Schlüsselnamen und Werte verwendet wird. Wenn dieser Schlüssel weggelassen wird, müssen der Client und der Server die Werte so interpretieren, als ob sie in Version 1 definiert sind.

66. **`r-host`**

    Dieses Feld wird für ursprüngliche Anfragen gesendet und gibt die Domäne des Ursprungsservers an, der für die Bereitstellung des Objekts verwendet wird. Im Fehlerfall können Sie dieses Feld verwenden, um den letzten Quellserver zu finden, der versucht wurde, zum Beispiel: *`cd8jhdejh6a`*`.mediapackagev2.us-east-1.amazonaws.com`.

67. **`sr-reason`**

    Dieses Feld gibt einen Grund an, warum der Ursprung ausgewählt wurde. Es ist leer, wenn eine Anfrage an den primären Ursprung erfolgreich ist.

    Wenn ein Ursprungs-Failover auftritt, enthält das Feld den HTTP-Fehlercode, der zum Failover geführt hat, z. B. oder. `Failover:403` `Failover:502` Wenn im Fall eines ursprünglichen Failovers auch die wiederholte Anfrage fehlschlägt und Sie keine benutzerdefinierten Fehlerseiten konfiguriert haben, wird die Antwort des `r-status` zweiten Ursprungs angezeigt. Wenn Sie jedoch benutzerdefinierte Fehlerseiten zusammen mit dem ursprünglichen Failover konfiguriert haben, enthält dies die Antwort des zweiten Ursprungs, falls die Anfrage fehlgeschlagen ist und stattdessen eine benutzerdefinierte Fehlerseite zurückgegeben wurde.

    Wenn kein Ursprungs-Failover, sondern eine Auswahl des Ursprungs (Media Quality-Aware Resilience, MQAR) erfolgt, wird dies protokolliert als. `MediaQuality` Weitere Informationen finden Sie unter Resilienz im Hinblick auf Medienqualität.

68. **`x-edge-mqcs`**

    Dieses Feld gibt den Media Quality Confidence Score (MQCS) (Bereich: 0 — 100) für Mediensegmente an, die in den CMSD-Antwort-Headern von v2 CloudFront abgerufen wurden. MediaPackage Dieses Feld ist für Anfragen verfügbar, die einem Cache-Verhalten entsprechen, das über eine MQAR-fähige Ursprungsgruppe verfügt. CloudFront protokolliert dieses Feld für Mediensegmente, die zusätzlich zu den ursprünglichen Anfragen auch aus dem Cache bedient werden. Weitere Informationen finden Sie unter Resilienz im Hinblick auf Medienqualität.

69**distribution-tenant-id**

Die ID des Distributionsmandanten.

Endpunkt (Kinesis Data Streams)

Der Endpunkt enthält Informationen zu den Kinesis Data Streams, an die Sie Echtzeitprotokolle senden möchten. Sie stellen den Amazon-Ressourcennamen (ARN) des Streams oder der Funktion bereit.

Weitere Informationen zum Erstellen von Kinesis Data Streams finden Sie in den folgenden Themen im Amazon Kinesis Data Streams Developer Guide.

- Streams erstellen und verwalten
- Führen Sie grundlegende Kinesis Data Streams Streams-Operationen mit dem AWS CLI
- Einen Stream erstellen (verwendet den AWS SDK für Java)

Wenn Sie einen Datensteam erstellen, müssen Sie die Anzahl der Shards angeben. Verwenden Sie die folgenden Informationen, um die Anzahl der benötigten Shards zu schätzen.

So schätzen Sie die Anzahl der Shards für Ihren Kinesis-Datenstream:

1. Berechnen (oder schätzen) Sie die Anzahl der Anforderungen pro Sekunde, die Ihre CloudFront-Verteilung erhält.

   Sie können die CloudFrontNutzungsberichte (in der CloudFront Konsole) und die CloudFront Metriken (in den CloudWatch Konsolen CloudFront und Amazon) verwenden, um Ihre Anfragen pro Sekunde zu berechnen.
2. Bestimmen Sie die typische Größe eines einzelnen Echtzeit-Protokolldatensatzes.

   Im Allgemeinen ist ein einzelner Protokolldatensatz etwa 500 Byte groß. Ein typischer Datensatz, der alle verfügbaren Felder enthält, ist in der Regel etwa 1 KB groß.

   Wenn Sie Ihre Protokoll-Datensatzgröße nicht genau kennen, können Sie Echtzeitprotokolle mit einer niedrigen Abtastrate (z. B. 1 %) aktivieren und dann die durchschnittliche Datensatzgröße anhand von Überwachungsdaten in Kinesis Data Streams berechnen (Gesamtzahl der eingehenden Bytes dividiert durch die Gesamtzahl der Datensätze).

3. Wählen Sie auf der Preisseite für Amazon Kinesis Data Streams unter AWS -PreisrechnerJetzt Ihren individuellen Kostenvoranschlag erstellen aus.

   - Geben Sie im Rechner die Anzahl der Anfragen (Datensätze) pro Sekunde ein.
   - Geben Sie die durchschnittliche Datensatzgröße eines einzelnen Protokolldatensatzes ein.
   - Wählen Sie Berechnungen anzeigen.

   Der Preisrechner zeigt Ihnen die Anzahl der benötigten Shards und die geschätzten Kosten.

IAM-Rolle

Die AWS Identity and Access Management (IAM-) Rolle, die die CloudFront Erlaubnis erteilt, Echtzeitprotokolle an Ihren Kinesis-Datenstrom zu übermitteln.

Wenn Sie mit der CloudFront Konsole eine Echtzeit-Protokollkonfiguration erstellen, können Sie Neue Servicerolle erstellen wählen, damit die Konsole die IAM-Rolle für Sie erstellt.

Wenn Sie eine Echtzeit-Protokollkonfiguration mit AWS CloudFormation oder der CloudFront API (AWS CLI oder dem SDK) erstellen, müssen Sie die IAM-Rolle selbst erstellen und den Rollen-ARN angeben. Verwenden Sie die folgenden Richtlinien, um die IAM-Rolle selbst zu erstellen:

IAM-Rollen-Vertrauensrichtlinie

Um die folgende Vertrauensrichtlinie für IAM-Rollen zu verwenden, *111122223333* ersetzen Sie sie durch Ihre AWS-Konto Nummer. Das `Condition` Element in dieser Richtlinie trägt dazu bei, das Problem des verwirrten Stellvertreters zu vermeiden, da CloudFront Sie diese Rolle nur im Namen einer Abteilung in Ihrem AWS-Konto Unternehmen übernehmen können.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudfront.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
```

```
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                }
            }
        }
    ]
}
```

IAM-Rollen-Berechtigungsrichtlinie für einen unverschlüsselten Daten-Stream

Um die folgende Richtlinie zu verwenden, *arn:aws:kinesis:us-east-2:123456789012:stream/StreamName* ersetzen Sie sie durch den ARN Ihres Kinesis-Datenstroms.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStreamSummary",
                "kinesis:DescribeStream",
                "kinesis:PutRecord",
                "kinesis:PutRecords"
            ],
            "Resource": [
                "arn:aws:kinesis:us-east-2:123456789012:stream/StreamName"
            ]
        }
    ]
}
```

IAM-Rollen-Berechtigungsrichtlinie für einen verschlüsselten Daten-Stream

Um die folgende Richtlinie zu verwenden, *arn:aws:kinesis:us-east-2:123456789012:stream/StreamName* ersetzen Sie sie durch den ARN Ihres Kinesis-

Datenstreams und *arn:aws:kms:us-east-2:123456789012:key/e58a3d0b-fe4f-4047-a495-ae03cc73d486* durch den ARN Ihres AWS KMS key.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStreamSummary",
                "kinesis:DescribeStream",
                "kinesis:PutRecord",
                "kinesis:PutRecords"
            ],
            "Resource": [
                "arn:aws:kinesis:us-east-2:123456789012:stream/StreamName"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"
            ],
            "Resource": [
                "arn:aws:kms:us-east-2:123456789012:key/e58a3d0b-fe4f-4047-a495-
ae03cc73d486"
            ]
        }
    ]
}
```

Einen Kinesis Data Streams Streams-Consumer erstellen

Um Ihre Echtzeit-Protokolle zu lesen und zu analysieren, erstellen oder verwenden Sie einen Kinesis Data Streams--Verbraucher. Wenn Sie einen Consumer für CloudFront Echtzeit-Logs erstellen, ist es wichtig zu wissen, dass die Felder in jedem Echtzeit-Protokolldatensatz immer in der gleichen

Reihenfolge übermittelt werden, wie im Felder Abschnitt aufgeführt. Stellen Sie sicher, dass Sie Ihren Verbraucher für diese feste Reihenfolge erstellen.

Betrachten Sie beispielsweise eine Echtzeit-Protokollkonfiguration, die nur die folgenden drei Felder enthält: `time-to-first-byte`, `sc-status` und `c-country`. In diesem Szenario ist das letzte Feld, `c-country`, immer Feld Nummer 3 in jedem Protokolldatensatz. Wenn Sie der Echtzeit-Protokollkonfiguration jedoch später Felder hinzufügen, kann sich die Platzierung jedes Feldes in einem Datensatz ändern.

Wenn Sie beispielsweise die Felder `sc-bytes` und `time-taken` zur Echtzeit-Protokollkonfiguration hinzufügen, werden diese Felder entsprechend der im Felder-Abschnitt angezeigten Reihenfolge in jeden Protokolldatensatz eingefügt. Die resultierende Reihenfolge aller fünf Felder lautet dann `time-to-first-byte`, `sc-status`, `sc-bytes`, `time-taken` und `c-country`. Das Feld `c-country`hatte ursprünglich die Feldnummer 3, hat jetzt jedoch Feldnummer 5. Stellen Sie sicher, dass Ihre Verbraucheranwendung Felder verarbeiten kann, die die Position in einem Protokolldatensatz ändern, falls Sie Felder zu Ihrer Echtzeit-Protokollkonfiguration hinzufügen.

Beheben Sie Probleme mit Echtzeitprotokollen

Nachdem Sie eine Echtzeit-Protokollkonfiguration erstellt haben, stellen Sie möglicherweise fest, dass keine Datensätze (oder nicht alle Datensätze) an Kinesis Data Streams übermittelt werden. In diesem Fall sollten Sie zunächst überprüfen, ob Ihre CloudFront -Verteilung Viewer-Anforderungen empfängt. Wenn dies der Fall ist, können Sie die folgende Einstellung überprüfen, um die Fehlerbehebung fortzusetzen.

IAM-Rollenberechtigungen

CloudFront Verwendet die IAM-Rolle in der Echtzeit-Protokollkonfiguration, um Protokolldatensätze in Echtzeit an Ihren Kinesis-Datenstrom zu übermitteln. Stellen Sie sicher, dass die Rollen-Vertrauensrichtlinie und die Rollen-Berechtigungsrichtlinie mit den in angezeigten Richtlinien übereinstimme IAM-Rolle.

Kinesis Data Streams-Drosselung

Wenn Echtzeit-Protokolldatensätze schneller in Ihren Kinesis-Datenstream CloudFront geschrieben werden, als der Stream verarbeiten kann, drosselt Kinesis Data Streams möglicherweise die Anfragen von. CloudFront In diesem Fall können Sie die Anzahl der Shards in Ihrem Kinesis-Datenstream erhöhen. Jeder Shard unterstützt Schreibvorgänge von bis zu 1.000 Datensätzen pro Sekunde bis zu einem maximalen Datenschreibvolumen von 1 MB pro Sekunde.

Protokolle für Edge-Funktionen

Sie können Amazon CloudWatch Logs verwenden, um Protokolle für Ihre Edge-Funktionen, sowohl Lambda @Edge als auch CloudFront Functions, abzurufen. Sie können über die CloudWatch Konsole oder die Logs-API auf die CloudWatch Protokolle zugreifen.

> **Important**
>
> Wir empfehlen Ihnen, die Protokolle zu verwenden, um die Art der Anfragen nach Ihren Inhalten zu verstehen, und nicht, um alle Anfragen vollständig zu erfassen. CloudFront liefert Edge-Funktionsprotokolle nach bestem Wissen und Gewissen. Der Protokolleintrag für eine bestimmte Anfrage wird möglicherweise viel später übermittelt, als die Anfrage tatsächlich verarbeitet wurde; in seltenen Fällen kann es auch sein, dass ein Protokolleintrag gar nicht übermittelt wird. Wenn ein Protokolleintrag nicht in den Protokollen für Edge-Funktionen enthalten ist, stimmt die Anzahl der Einträge in den Protokollen für Edge-Funktionen nicht mit deren Anzahl in den Abrechnungs- und Nutzungsberichten für AWS überein.

Themen

Lambda@Edge-Protokolle

Lambda @Edge sendet automatisch CloudWatch Funktionsprotokolle an Logs und erstellt Protokollstreams dort, AWS-Regionen wo die Funktionen aufgerufen werden. Wenn Sie eine Funktion in erstellen oder ändern AWS Lambda, können Sie entweder den Standardnamen der CloudWatch Protokollgruppe verwenden oder ihn anpassen.

- Der Standardname für die Protokollgruppe *<FunctionName>* ist der Name, den Sie bei der Erstellung der Funktion angegeben haben. `/aws/lambda/`*<FunctionName>* Beim Senden von CloudWatch Protokollen an fügt Lambda @Edge dem Funktionsnamen automatisch das `us-east-1` Präfix hinzu, sodass der Protokollgruppenname lautet`/aws/lambda/us-east-1.`*<FunctionName>*. Dieses Präfix entspricht dem AWS-Region Ort, an dem die Funktion erstellt wurde. Dieses Präfix bleibt Teil des Protokollgruppennamens, auch in anderen Regionen, in denen die Funktion aufgerufen wird.

- Wenn Sie einen benutzerdefinierten Protokollgruppennamen angeben, wie z. B.`/MyLogGroup`, fügt Lambda @Edge das Regionspräfix nicht hinzu. Der Name der Protokollgruppe bleibt in allen anderen Regionen, in denen die Funktion aufgerufen wird, derselbe.

> **Note**
>
> Wenn Sie eine benutzerdefinierte Protokollgruppe erstellen und denselben Namen wie den Standard angeben`/aws/lambda/<FunctionName>`, fügt Lambda @Edge dem Funktionsnamen das `us-east-1` Präfix hinzu.

Neben der Anpassung des Protokollgruppennamens unterstützen Lambda @Edge -Funktionen auch JSON- und Klartext-Protokollformate sowie Filterung auf Protokollebene. Weitere Informationen finden Sie unter Konfiguration erweiterter Protokollierungssteuerungen für die Lambda-Funktion im AWS Lambda Entwicklerhandbuch.

> **Note**
>
> Lambda@Edge drosselt Protokolle basierend auf dem angeforderten Volumen und der Größe der Protokolle.

Sie müssen die CloudWatch Protokolldateien in der richtigen Region überprüfen, um Ihre Lambda @Edge -Funktionsprotokolldateien zu sehen. Um die Regionen zu sehen, in denen Ihre Lambda @Edge -Funktion ausgeführt wird, sehen Sie sich in der CloudFront Konsole Diagramme mit den Metriken für die Funktion an. Metriken werden für jede -Region angezeigt. Auf derselben Seite können Sie eine Region auswählen und Protokolldateien für diese anzeigen, um Probleme zu untersuchen.

Weitere Informationen zur Verwendung von CloudWatch Logs mit Lambda @Edge -Funktionen finden Sie in den folgenden Themen:

- Weitere Informationen zum Anzeigen von Diagrammen im Bereich Überwachung der CloudFront Konsole finden Sie unterthe section called “Überwachen Sie CloudFront Metriken mit Amazon CloudWatch”.

- Informationen zu den Berechtigungen, die zum Senden von Daten an CloudWatch Logs erforderlich sind, finden Sie unterthe section called “Richten Sie IAM-Berechtigungen und -Rollen ein”.
- Weitere Informationen zum Hinzufügen der Protokollierung zu einer Lambda@Edge-Funktion finden Sie unter AWS Lambda -Funktionsprotokollierung in Node.js oder AWS Lambda -Funktionsprotokollierung in Python im AWS Lambda -Entwicklerhandbuch.
- Informationen zu CloudWatch Logs-Kontingenten (früher als Limits bezeichnet) finden Sie unter CloudWatch Logs-Kontingente im Amazon CloudWatch Logs-Benutzerhandbuch.

CloudFront Funktionen, Protokolle

Wenn der Code einer CloudFront Funktion `console.log()` Anweisungen enthält, sendet CloudFront Functions diese Protokollzeilen automatisch an CloudWatch Logs. Wenn es keine `console.log()` Anweisungen gibt, wird nichts an CloudWatch Logs gesendet.

CloudFront Functions erstellt immer Log-Streams in der Region USA Ost (Nord-Virginia) (`us-east-1`), unabhängig davon, an welchem Edge-Standort die Funktion ausgeführt wurde. Der Name der Protokollgruppe hat das Format `/aws/cloudfront/function/<FunctionName>`. Dabei ist *`<FunctionName>`* der Name, den Sie der Funktion bei der Erstellung gegeben haben. Der Name des Protokollstreams ist im Format `YYYY/M/D/`*`UUID`*.

Im Folgenden wird ein Beispiel für eine Protokollnachricht gezeigt, die an CloudWatch Logs gesendet wurde. Jede Zeile beginnt mit einer ID, die eine CloudFront Anfrage eindeutig identifiziert. Die Nachricht beginnt mit einer `START` Zeile, die die CloudFront Verteilungs-ID enthält, und endet mit einer `END` Zeile. Zwischen den Zeilen `END` und `START` befinden sich die durch `console.log()`-Anweisungen in der Funktion erzeugten Protokollzeilen.

```
U7b4hR_RaxMADupvKAvr8_m9gsGXvioUggLV5Oyq-vmAtH8HADpjhw== START DistributionID:
 E3E5D42GADAXZZ
U7b4hR_RaxMADupvKAvr8_m9gsGXvioUggLV5Oyq-vmAtH8HADpjhw== Example function log output
U7b4hR_RaxMADupvKAvr8_m9gsGXvioUggLV5Oyq-vmAtH8HADpjhw== END
```

> **Note**
>
> CloudFront Functions sendet CloudWatch nur Protokolle für Funktionen in der `LIVE` Phase, die als Reaktion auf Produktionsanfragen und -antworten ausgeführt werden. Wenn Sie eine Funktion testen, sendet CloudFront keine Protokolle an CloudWatch. Die Testausgabe enthält Informationen über Fehler, Rechenauslastung und Funktionsprotokolle

(`console.log()`Anweisungen), aber diese Informationen werden nicht an gesendet CloudWatch.

CloudFront Functions verwendet eine dienstbezogene AWS Identity and Access Management (IAM-) Rolle, um Protokolle an Logs in Ihrem CloudWatch Konto zu senden. Eine serviceverknüpfte Rolle ist eine IAM-Rolle, die direkt mit einer verknüpft ist. AWS-Service Mit Diensten verknüpfte Rollen sind vom Dienst vordefiniert und enthalten alle Berechtigungen, die der Dienst benötigt, um andere AWS-Services Rollen für Sie aufzurufen. CloudFront Functions verwendet die AWSServiceRoleForCloudFrontLoggerdienstverknüpfte Rolle. Weitere Informationen zu dieser Rolle finden Sie unter the section called “Serviceverknüpfte Rollen für Lambda@Edge” (Lambda@Edge verwendet dieselbe serviceverknüpfte Rolle).

Wenn eine Funktion aufgrund eines Überprüfungs- oder Ausführungsfehlers fehlschlägt, werden die Informationen in Standardprotokollen und Echtzeitprotokollen protokolliert. Spezifische Informationen zu dem Fehler finden Sie in den `x-edge-detailed-result-type` Feldern `x-edge-result-typex-edge-response-result-type`, und.

Protokollieren Amazon CloudFront Amazon-API-Aufrufen mit AWS CloudTrail

CloudFront ist integriert in AWS CloudTrail, ein Service, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem ausgeführt wurden AWS-Service. CloudTrail erfasst alle API-Aufrufe CloudFront als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der CloudFront Konsole und Codeaufrufen für die CloudFront API-Operationen. Anhand der von gesammelten Informationen können Sie die Anfrage CloudTrail, an die die Anfrage gestellt wurde CloudFront, die IP-Adresse, von der aus die Anfrage gestellt wurde, den Zeitpunkt der Anfrage und weitere Details ermitteln.

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit Anmeldeinformationen des Root-Benutzers oder des Benutzers gestellt wurde.
- Die Anforderung wurde im Namen eines IAM-Identity-Center-Benutzers erstellt.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anforderung aus einem anderen AWS-Service gesendet wurde.

CloudTrail ist in Ihrem aktiv AWS-Konto , wenn Sie das Konto erstellen, und Sie haben automatisch Zugriff auf den CloudTrail Eventverlauf. Der CloudTrail Ereignisverlauf bietet eine einsehbare, durchsuchbare, herunterladbare und unveränderliche Aufzeichnung der aufgezeichneten Verwaltungsereignisse der letzten 90 Tage in einem. AWS-Region Weitere Informationen finden Sie im AWS CloudTrail Benutzerhandbuch unter Arbeiten mit dem CloudTrail Ereignisverlauf. Für die Anzeige des Ereignisverlaufs CloudTrail fallen keine Gebühren an.

Für eine fortlaufende Aufzeichnung der Ereignisse in AWS-Konto den letzten 90 Tagen erstellen Sie einen Trail- oder CloudTrailLake-Event-Datenspeicher.

CloudTrail Pfade

Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Alle mit dem erstellten Pfade AWS Management Console sind regionsübergreifend. Sie können mithilfe von AWS CLI einen Einzel-Region- oder einen Multi-Region-Trail erstellen. Es wird empfohlen, einen Trail mit mehreren Regionen zu erstellen, da Sie alle Aktivitäten AWS-Regionen in Ihrem Konto erfassen. Wenn Sie einen Einzel-Region-Trail erstellen, können Sie nur die Ereignisse anzeigen, die im AWS-Region des Trails protokolliert wurden. Weitere Informationen zu Trails finden Sie unter Erstellen eines Trails für Ihr AWS-Konto und Erstellen eines Trails für eine Organisation im AWS CloudTrail -Benutzerhandbuch.

Sie können eine Kopie Ihrer laufenden Verwaltungsereignisse kostenlos an Ihren Amazon S3 S3-Bucket senden, CloudTrail indem Sie einen Trail erstellen. Es fallen jedoch Amazon S3 S3-Speichergebühren an. Weitere Informationen zur CloudTrail Preisgestaltung finden Sie unter AWS CloudTrail Preise. Informationen zu Amazon-S3-Preisen finden Sie unter Amazon S3 – Preise.

CloudTrail Datenspeicher für Ereignisse in Lake

CloudTrail Mit Lake können Sie SQL-basierte Abfragen für Ihre Ereignisse ausführen. CloudTrail Lake konvertiert bestehende Ereignisse im zeilenbasierten JSON-Format in das Apache ORC-Format. ORC ist ein spaltenförmiges Speicherformat, das für den schnellen Abruf von Daten optimiert ist. Die Ereignisse werden in Ereignisdatenspeichern zusammengefasst, bei denen es sich um unveränderliche Sammlungen von Ereignissen handelt, die auf Kriterien basieren, die Sie mit Hilfe von erweiterten Ereignisselektoren auswählen. Die Selektoren, die Sie auf einen Ereignisdatenspeicher anwenden, steuern, welche Ereignisse bestehen bleiben und für Sie zur Abfrage verfügbar sind. Weitere Informationen zu CloudTrail Lake finden Sie unter Arbeiten mit AWS CloudTrail Lake im AWS CloudTrail Benutzerhandbuch.

CloudTrail Für das Speichern und Abfragen von Ereignisdaten in Lake fallen Kosten an. Beim Erstellen eines Ereignisdatenspeichers wählen Sie die Preisoption aus, die für den

Ereignisdatenspeicher genutzt werden soll. Die Preisoption bestimmt die Kosten für die Erfassung und Speicherung von Ereignissen sowie die standardmäßige und maximale Aufbewahrungsdauer für den Ereignisdatenspeicher. Weitere Informationen zur CloudTrail Preisgestaltung finden Sie unter AWS CloudTrail Preise.

> **Note**
>
> CloudFront ist ein globaler Service. CloudTrail zeichnet Ereignisse CloudFront in der Region USA Ost (Nord-Virginia) auf. Weitere Informationen finden Sie im AWS CloudTrail Benutzerhandbuch unter Globale Serviceereignisse.
> Wenn Sie temporäre Sicherheitsanmeldeinformationen verwenden AWS Security Token Service, werden Anrufe an regionale Endpunkte, z. B.`us-west-2`, in der CloudTrail entsprechenden Region angemeldet.
> Weitere Informationen zu CloudFront Endpunkten finden Sie unter CloudFront Endpunkte und Kontingente in der. Allgemeine AWS-Referenz

CloudFront Datenereignisse in CloudTrail

Datenereignisse liefern Informationen über die Ressourcenoperationen, die auf oder in einer Ressource ausgeführt werden (z. B. Lesen oder Schreiben in eine CloudFront Distribution). Sie werden auch als Vorgänge auf Datenebene bezeichnet. Datenereignisse sind oft Aktivitäten mit hohem Volume. Protokolliert standardmäßig CloudTrail keine Datenereignisse. Der CloudTrail Ereignisverlauf zeichnet keine Datenereignisse auf.

Für Datenereignisse werden zusätzliche Gebühren fällig. Weitere Informationen zur CloudTrail Preisgestaltung finden Sie unter AWS CloudTrail Preisgestaltung.

Sie können Datenereignisse für die CloudFront Ressourcentypen mithilfe der CloudTrail Konsole oder CloudTrail API-Operationen protokollieren. AWS CLI Weitere Informationen zum Protokollieren von Datenereignissen finden Sie unter Protokollieren von Datenereignissen mit dem AWS Management Console und Protokollieren von Datenereignissen mit dem AWS Command Line Interface im AWS CloudTrail -Benutzerhandbuch.

In der folgenden Tabelle sind die CloudFront Ressourcentypen aufgeführt, für die Sie Datenereignisse protokollieren können. In der Spalte Datenereignistyp (Konsole) wird der Wert angezeigt, den Sie in der Liste Datenereignistyp auf der CloudTrail Konsole auswählen können. In der Wertspalte resources.type wird der `resources.type` Wert angezeigt, den Sie bei der

Konfiguration erweiterter Event-Selektoren mithilfe von oder angeben würden. AWS CLI CloudTrail APIs In der CloudTrail Spalte APIs Protokollierte Daten werden die API-Aufrufe angezeigt, die CloudTrail für den Ressourcentyp protokolliert wurden.

Typ des Datenereignisses (Konsole)	resources.type-Wert	Daten, die APIs protokolliert wurden CloudTrail
CloudFront KeyValueStore	`AWS::CloudFront::KeyValueStore`	• DeleteKeys • DescribeKeyValueStore • GetKey • ListKeys • PutKeys • UpdateKeys

Sie können erweiterte Event-Selektoren so konfigurieren, dass sie nach den Feldern `eventName`, `readOnly` und `resources.`ARN filtern, sodass nur die Ereignisse protokolliert werden, die für Sie wichtig sind. Weitere Informationen zu diesen Feldern finden Sie unter AdvancedFieldSelector in der API-Referenz zu AWS CloudTrail

CloudFront Managementereignisse in CloudTrail

Verwaltungsereignisse bieten Informationen über Verwaltungsvorgänge, die an Ressourcen in Ihrem ausgeführt werden AWS-Konto. Sie werden auch als Vorgänge auf Steuerebene bezeichnet. CloudTrail Protokolliert standardmäßig Verwaltungsereignisse.

Amazon CloudFront protokolliert alle Operationen auf der CloudFront Kontrollebene als Managementereignisse. Eine Liste der Vorgänge auf der CloudFront Amazon-Kontrollebene, bei denen CloudFront sich angemeldet wird CloudTrail, finden Sie in der Amazon CloudFront API-Referenz.

CloudFront Beispiele für Ereignisse

Ein Ereignis stellt eine einzelne Anfrage aus einer beliebigen Quelle dar und enthält Informationen über den angeforderten API-Vorgang, Datum und Uhrzeit des Vorgangs, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass Ereignisse nicht in einer bestimmten Reihenfolge angezeigt werden.

Inhalt

Beispiel: UpdateDistribution

Das folgende Beispiel zeigt ein CloudTrail Ereignis, das den UpdateDistributionVorgang demonstriert.

Für CloudFront API-Aufrufe `eventSource` ist der`cloudfront.amazonaws.com`.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:role-session-name",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/role-session-name",
        "accountId": "111122223333",
        "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2024-02-02T19:23:50Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2024-02-02T19:26:01Z",
    "eventSource": "cloudfront.amazonaws.com",
    "eventName": "UpdateDistribution",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "52.94.133.137",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/121.0.0.0 Safari/537.36",
    "requestParameters": {
        "distributionConfig": {
```

```
            "defaultRootObject": "",
            "aliases": {
                "quantity": 3,
                "items": [
                    "alejandro_rosalez.awsps.myinstance.com",
                    "cross-testing.alejandro_rosalez.awsps.myinstance.com",
                    "*.alejandro_rosalez.awsps.myinstance.com"
                ]
            },
            "cacheBehaviors": {
                "quantity": 0,
                "items": []
            },
            "httpVersion": "http2and3",
            "originGroups": {
                "quantity": 0,
                "items": []
            },
            "viewerCertificate": {
                "minimumProtocolVersion": "TLSv1.2_2021",
                "cloudFrontDefaultCertificate": false,
                "aCMCertificateArn": "arn:aws:acm:us-east-1:111122223333:certificate/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
                "sSLSupportMethod": "sni-only"
            },
            "webACLId": "arn:aws:wafv2:us-east-1:111122223333:global/webacl/testing-
acl/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "customErrorResponses": {
                "quantity": 0,
                "items": []
            },
            "logging": {
                "includeCookies": false,
                "prefix": "",
                "enabled": false,
                "bucket": ""
            },
            "priceClass": "PriceClass_All",
            "restrictions": {
                "geoRestriction": {
                    "restrictionType": "none",
                    "quantity": 0,
                    "items": []
                }
```

```
            },
            "isIPV6Enabled": true,
            "callerReference": "1578329170895",
            "continuousDeploymentPolicyId": "",
            "enabled": true,
            "defaultCacheBehavior": {
                "targetOriginId": "d111111abcdef8",
                "minTTL": 0,
                "compress": false,
                "maxTTL": 31536000,
                "functionAssociations": {
                    "quantity": 0,
                    "items": []
                },
                "trustedKeyGroups": {
                    "quantity": 0,
                    "items": [],
                    "enabled": false
                },
                "smoothStreaming": false,
                "fieldLevelEncryptionId": "",
                "defaultTTL": 86400,
                "lambdaFunctionAssociations": {
                    "quantity": 0,
                    "items": []
                },
                "viewerProtocolPolicy": "redirect-to-https",
                "forwardedValues": {
                    "cookies": {"forward": "none"},
                    "queryStringCacheKeys": {
                        "quantity": 0,
                        "items": []
                    },
                    "queryString": false,
                    "headers": {
                        "quantity": 1,
                        "items": ["*"]
                    }
                },
                "trustedSigners": {
                    "items": [],
                    "enabled": false,
                    "quantity": 0
                },
```

```
                "allowedMethods": {
                    "quantity": 2,
                    "items": [
                        "HEAD",
                        "GET"
                    ],
                    "cachedMethods": {
                        "quantity": 2,
                        "items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                }
            },
            "staging": false,
            "origins": {
                "quantity": 1,
                "items": [
                    {
                        "originPath": "",
                        "connectionTimeout": 10,
                        "customOriginConfig": {
                            "originReadTimeout": 30,
                            "hTTPSPort": 443,
                            "originProtocolPolicy": "https-only",
                            "originKeepaliveTimeout": 5,
                            "hTTPPort": 80,
                            "originSslProtocols": {
                                "quantity": 3,
                                "items": [
                                    "TLSv1",
                                    "TLSv1.1",
                                    "TLSv1.2"
                                ]
                            }
                        },
                        "id": "d111111abcdef8",
                        "domainName": "d111111abcdef8.cloudfront.net",
                        "connectionAttempts": 3,
                        "customHeaders": {
                            "quantity": 0,
                            "items": []
                        },
```

```
                        "originShield": {"enabled": false},
                        "originAccessControlId": ""
                    }
                ]
            },
            "comment": "HIDDEN_DUE_TO_SECURITY_REASONS"
        },
        "id": "EDFDVBD6EXAMPLE",
        "ifMatch": "E1RTLUR9YES76O"
    },
    "responseElements": {
        "distribution": {
            "activeTrustedSigners": {
                "quantity": 0,
                "enabled": false
            },
            "id": "EDFDVBD6EXAMPLE",
            "domainName": "d111111abcdef8.cloudfront.net",
            "distributionConfig": {
                "defaultRootObject": "",
                "aliases": {
                    "quantity": 3,
                    "items": [
                        "alejandro_rosalez.awsps.myinstance.com",
                        "cross-testing.alejandro_rosalez.awsps.myinstance.com",
                        "*.alejandro_rosalez.awsps.myinstance.com"
                    ]
                },
                "cacheBehaviors": {"quantity": 0},
                "httpVersion": "http2and3",
                "originGroups": {"quantity": 0},
                "viewerCertificate": {
                    "minimumProtocolVersion": "TLSv1.2_2021",
                    "cloudFrontDefaultCertificate": false,
                    "aCMCertificateArn": "arn:aws:acm:us-
east-1:111122223333:certificate/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
                    "sSLSupportMethod": "sni-only",
                    "certificateSource": "acm",
                    "certificate": "arn:aws:acm:us-east-1:111122223333:certificate/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
                },
                "webACLId": "arn:aws:wafv2:us-east-1:111122223333:global/webacl/
testing-acl/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
                "customErrorResponses": {"quantity": 0},
```

```
                "logging": {
                    "includeCookies": false,
                    "prefix": "",
                    "enabled": false,
                    "bucket": ""
                },
                "priceClass": "PriceClass_All",
                "restrictions": {
                    "geoRestriction": {
                        "restrictionType": "none",
                        "quantity": 0
                    }
                },
                "isIPV6Enabled": true,
                "callerReference": "1578329170895",
                "continuousDeploymentPolicyId": "",
                "enabled": true,
                "defaultCacheBehavior": {
                    "targetOriginId": "d111111abcdef8",
                    "minTTL": 0,
                    "compress": false,
                    "maxTTL": 31536000,
                    "functionAssociations": {"quantity": 0},
                    "trustedKeyGroups": {
                        "quantity": 0,
                        "enabled": false
                    },
                    "smoothStreaming": false,
                    "fieldLevelEncryptionId": "",
                    "defaultTTL": 86400,
                    "lambdaFunctionAssociations": {"quantity": 0},
                    "viewerProtocolPolicy": "redirect-to-https",
                    "forwardedValues": {
                        "cookies": {"forward": "none"},
                        "queryStringCacheKeys": {"quantity": 0},
                        "queryString": false,
                        "headers": {
                            "quantity": 1,
                            "items": ["*"]
                        }
                    },
                    "trustedSigners": {
                        "enabled": false,
                        "quantity": 0
```

```
                    },
                    "allowedMethods": {
                        "quantity": 2,
                        "items": [
                            "HEAD",
                            "GET"
                        ],
                        "cachedMethods": {
                            "quantity": 2,
                            "items": [
                                "HEAD",
                                "GET"
                            ]
                        }
                    }
                },
                "staging": false,
                "origins": {
                    "quantity": 1,
                    "items": [
                        {
                            "originPath": "",
                            "connectionTimeout": 10,
                            "customOriginConfig": {
                                "originReadTimeout": 30,
                                "hTTPSPort": 443,
                                "originProtocolPolicy": "https-only",
                                "originKeepaliveTimeout": 5,
                                "hTTPPort": 80,
                                "originSslProtocols": {
                                    "quantity": 3,
                                    "items": [
                                        "TLSv1",
                                        "TLSv1.1",
                                        "TLSv1.2"
                                    ]
                                }
                            },
                            "id": "d111111abcdef8",
                            "domainName": "d111111abcdef8.cloudfront.net",
                            "connectionAttempts": 3,
                            "customHeaders": {"quantity": 0},
                            "originShield": {"enabled": false},
                            "originAccessControlId": ""
```

```
                        }
                    ]
                },
                "comment": "HIDDEN_DUE_TO_SECURITY_REASONS"
            },
            "aliasICPRecordals": [
                {
                    "cNAME": "alejandro_rosalez.awsps.myinstance.com",
                    "iCPRecordalStatus": "APPROVED"
                },
                {
                    "cNAME": "cross-testing.alejandro_rosalez.awsps.myinstance.com",
                    "iCPRecordalStatus": "APPROVED"
                },
                {
                    "cNAME": "*.alejandro_rosalez.awsps.myinstance.com",
                    "iCPRecordalStatus": "APPROVED"
                }
            ],
            "aRN": "arn:aws:cloudfront::111122223333:distribution/EDFDVBD6EXAMPLE",
            "status": "InProgress",
            "lastModifiedTime": "Feb 2, 2024 7:26:01 PM",
            "activeTrustedKeyGroups": {
                "enabled": false,
                "quantity": 0
            },
            "inProgressInvalidationBatches": 0
        },
        "eTag": "E1YHBLAB2BJY1G"
    },
    "requestID": "4e6b66f9-d548-11e3-a8a9-73e33example",
    "eventID": "5ab02562-0fc5-43d0-b7b6-90293example",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "apiVersion": "2020_05_31",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "cloudfront.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
```

```
}
```

Beispiel: UpdateKeys

Das folgende Beispiel zeigt ein CloudTrail Ereignis, das den UpdateKeysVorgang demonstriert.

Bei CloudFront KeyValueStore API-Aufrufen `eventSource` ist `edgekeyvaluestore.amazonaws.com` stattcloudfront.amazonaws.com.

```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:role-session-name",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/role-session-name",
        "accountId": "111122223333",
        "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "attributes": {
                "creationDate": "2023-11-01T23:41:14Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-11-01T23:41:28Z",
    "eventSource": "edgekeyvaluestore.amazonaws.com",
    "eventName": "UpdateKeys",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "3.235.183.252",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/121.0.0.0 Safari/537.36,
    "requestParameters": {
        "kvsARN": "arn:aws:cloudfront::111122223333:key-value-store/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "ifMatch": "KV3O6B1CX531EBP",
        "deletes": [
```

```
            {"key": "key1"}
        ]
    },
    "responseElements": {
        "itemCount": 0,
        "totalSizeInBytes": 0,
        "eTag": "KVDC9VEVZ71ZGO"
    },
    "requestID": "5ccf104c-acce-4ea1-b7fc-73e33example",
    "eventID": "a0b1b5c7-906c-439d-9925-90293example",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::CloudFront::KeyValueStore",
            "ARN": "arn:aws:cloudfront::111122223333:key-value-store/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "111122223333",
    "eventCategory": "Data",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "111122223333.cloudfront-kvs.global.api.aws"
    }
}
```


Informationen zu CloudTrail Datensatzinhalten finden Sie im AWS CloudTrail Benutzerhandbuch unter CloudTrail Datensatzinhalte.

Verfolgen Sie Konfigurationsänderungen mit AWS Config

Um Konfigurationen Ihrer AWS Ressourcen aufzuzeichnen und auszuwerten, können Sie diese AWS Config Funktion verwenden, die Ihnen einen detaillierten Überblick über die Konfiguration Ihrer Distributionen bietet. Dazu gehört auch, wie die Ressourcen zueinander in Beziehung stehen und wie sie in der Vergangenheit konfiguriert wurden, sodass Sie Änderungen im Laufe der Zeit überprüfen können.

Sie können es auch verwenden AWS Config , um Konfigurationsänderungen an Ihren CloudFront Verteilungseinstellungen aufzuzeichnen. Sie können Änderungen an Vertriebsstaaten, Preisklassen, Ursprüngen, Einstellungen für geografische Einschränkungen und Lambda @Edge -Konfigurationen erfassen.

> **Note**
>
> AWS Config zeichnet keine Schlüssel-Wert-Tags für CloudFront Streaming-Distributionen auf.

Inhalt

Richten Sie ein mit AWS Config CloudFront

Bei der Einrichtung können Sie wählen AWS Config, ob Sie alle unterstützten AWS Ressourcen oder nur einige bestimmte Ressourcen aufzeichnen möchten, z. B. CloudFront nur Änderungen für. Eine Liste der unterstützten CloudFront Ressourcen finden Sie im CloudFront Abschnitt Amazon des Themas Unterstützte Ressourcentypen im AWS Config Developer Guide.

> **Hinweise**
>
> - Um die Konfigurationsänderungen an Ihrer CloudFront Distribution nachzuverfolgen, müssen Sie sich bei der CloudFront Konsole im Osten der USA (Nord-Virginia) anmelden AWS-Region.
> - Es kann zu einer Verzögerung bei der Aufzeichnung von Ressourcen mit kommen AWS Config. AWS Config zeichnet Ressourcen erst auf, nachdem sie erkannt wurden.

Console

Zur Einrichtung mit AWS Config CloudFront

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Config Konsole.
2. Wählen Sie Get Started Now.
3. Geben Sie auf der Seite Einstellungen unter Aufzuzeichnende Ressourcentypen die AWS Ressourcentypen an, die Sie aufzeichnen AWS Config möchten. Wenn Sie nur CloudFront Änderungen aufzeichnen möchten, wählen Sie Bestimmte Typen und dann unter die Verteilung oder Streaming-Verteilung aus CloudFront, für die Sie Änderungen verfolgen möchten.

   Wählen Sie nach der Ersteinrichtung zum Hinzufügen oder Ändern der nachzuverfolgenden Verteilungen Einstellungen auf der linken Seite aus.
4. Geben Sie zusätzliche erforderliche Optionen an für AWS Config: eine Benachrichtigung einrichten, einen Speicherort für die Konfigurationsinformationen angeben und Regeln für die Bewertung von Ressourcentypen hinzufügen.

Weitere Informationen finden Sie unter Einrichtung AWS Config mit der Konsole im AWS Config Entwicklerhandbuch.

AWS CLI

Informationen zur Einrichtung AWS Config CloudFront mit der AWS CLI finden Sie unter Einrichtung AWS Config mit der AWS CLI im AWS Config Entwicklerhandbuch.

AWS Config API

Informationen zur Einrichtung AWS Config mit der CloudFront Verwendung der AWS Config API finden Sie unter StartConfigurationRecorderAPI-Vorgang in der AWS Config API-Referenz.

CloudFront Konfigurationshistorie anzeigen

Nachdem Sie mit der Aufzeichnung von Konfigurationsänderungen an Ihren Distributionen AWS Config begonnen haben, können Sie den Konfigurationsverlauf jeder Distribution abrufen, für CloudFront die Sie konfiguriert haben.

Sie können den Konfigurationsverlauf auf folgende Weise anzeigen.

Console

Für jede aufgezeichnete Ressource können Sie eine Zeitleistenseite mit einem Verlauf der Konfigurationsdetails anzeigen. Wählen Sie zum Anzeigen dieser Seite das graue Symbol in der Spalte Config Timeline (Konfig.-Timeline) der Seite Dedicated Hosts.

Weitere Informationen finden Sie im AWS Config Entwicklerhandbuch unter Konfigurationsdetails in der AWS Config Konsole anzeigen.

AWS CLI

Um eine Liste all Ihrer Distributionen abzurufen, führen Sie den list-discovered-resourcesBefehl aus, wie im folgenden Beispiel gezeigt.

```
aws configservice list-discovered-resources --resource-type
 AWS::CloudFront::Distribution
```

Führen Sie den get-resource-config-historyBefehl aus, um die Konfigurationsdetails einer Distribution für ein bestimmtes Zeitintervall abzurufen.

Weitere Informationen finden Sie unter View Configuration Details Using the CLI (Anzeigen von Konfigurationsdetails mithilfe der Befehlszeilenschnittstelle (CLI)) im AWS Config -Developer-Handbuch.

AWS Config API

Verwenden Sie den ListDiscoveredResourcesAPI-Vorgang, um eine Liste all Ihrer Distributionen abzurufen.

Verwenden Sie den GetResourceConfigHistoryAPI-Vorgang, um die Konfigurationsdetails einer Distribution für ein bestimmtes Zeitintervall abzurufen. Weitere Informationen finden Sie in der AWS Config -API-Referenz.

Evaluieren Sie CloudFront Konfigurationen mit AWS Config Regeln

Mithilfe von AWS Config Regeln können Sie Konfigurationen anhand der gewünschten Konfigurationen auswerten. Mithilfe von AWS Config Regeln können Sie beispielsweise beurteilen, ob Ihre CloudFront Ressourcen den gängigen bewährten Sicherheitsmethoden entsprechen. Sie können verwaltete Regeln wie Viewer-Richtlinie HTTPS, SNI aktiviert, OAC aktiviert, Origin Failover aktiviert, AWS WAF WebACL oder AWS Shield Advanced Ressourcenrichtlinien auswählen, die ausgelöst werden sollen, wenn sich die Konfiguration ändert.

Mit verwalteten Regeln können in regelmäßigen Abständen Evaluierungen mit einer von Ihnen festgelegten Häufigkeit durchgeführt werden. AWS Firewall Manager stützt sich auf AWS Config automatische Warnmeldungen und Abhilfemaßnahmen. Weitere Informationen finden Sie unter Ressourcen anhand von AWS Config Regeln evaluieren und Liste AWS Config verwalteter Regeln im AWS Config Entwicklerhandbuch.

Sicherheit bei Amazon CloudFront

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von einer Rechenzentrums- und Netzwerkarchitektur, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame Verantwortung von Ihnen AWS und Ihnen. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, die AWS Dienste in der AWS Cloud ausführt. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Auditoren von Drittanbietern testen und überprüfen die Effektivität unserer Sicherheitsmaßnahmen im Rahmen der AWS -Compliance-Programme regelmäßig. Weitere Informationen zu den Compliance-Programmen, die für Amazon gelten CloudFront, finden Sie unter AWS Services in Scope by Compliance Program.
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Service, den Sie nutzen. In Ihre Verantwortung fallen außerdem weitere Faktoren, wie z. B. die Vertraulichkeit der Daten, die Anforderungen Ihrer Organisation sowie geltende Gesetze und Vorschriften.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Nutzung anwenden können CloudFront. In den folgenden Themen erfahren Sie, wie Sie die Konfiguration vornehmen CloudFront , um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer CloudFront Ressourcen unterstützen.

Themen

Datenschutz bei Amazon CloudFront

Das AWS Modell der gilt für den Datenschutz bei Amazon CloudFront. Wie in diesem Modell beschrieben, AWS ist verantwortlich für den Schutz der globalen Infrastruktur, auf der alle Systeme laufen AWS Cloud. Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter Häufig gestellte Fragen zum Datenschutz. Informationen zum Datenschutz in Europa finden Sie im Blog-Beitrag AWS -Modell der geteilten Verantwortung und in der DSGVO im AWS -Sicherheitsblog.

Aus Datenschutzgründen empfehlen wir, dass Sie AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Verwenden Sie SSL/TLS, um mit Ressourcen zu kommunizieren. AWS Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein. AWS CloudTrail Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter Arbeiten mit CloudTrail Pfaden im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter Federal Information Processing Standard (FIPS) 140-3.

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit der Konsole, der CloudFront API oder auf andere AWS-Services Weise arbeiten oder diese verwenden. AWS CLI AWS SDKs Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet

werden. Wenn Sie eine URL für einen externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

Amazon CloudFront bietet verschiedene Optionen, mit denen Sie die bereitgestellten Inhalte schützen können:

- Konfigurieren von HTTPS-Verbindungen.
- Konfigurieren Sie die Verschlüsselung auf Feldebene, um während der Übertragung zusätzliche Sicherheit für bestimmte Daten zu bieten.
- Beschränken des Zugriffs auf Inhalte, sodass nur bestimmte Personen oder Personen eines bestimmten Bereichs diese sehen können.

Die folgenden Themen erläutern die Optionen im Detail.

Themen

Verschlüsselung während der Übertragung

Um Ihre Daten während der Übertragung zu verschlüsseln, konfigurieren Sie Amazon CloudFront so, dass Zuschauer HTTPS verwenden müssen, um Ihre Dateien anzufordern, sodass Verbindungen bei der CloudFront Kommunikation mit Zuschauern verschlüsselt werden. Sie können auch so konfigurieren CloudFront , dass HTTPS verwendet wird, um Dateien von Ihrem Ursprung abzurufen, sodass Verbindungen bei der CloudFront Kommunikation mit Ihrem Ursprung verschlüsselt werden.

Weitere Informationen finden Sie unter Verwenden Sie HTTPS mit CloudFront.

Die Verschlüsselung auf Feldebene fügt zusammen mit HTTPS eine zusätzliche Sicherheitsschicht hinzu, mit der Sie bestimmte Daten während der gesamten Systemverarbeitung schützen können, sodass nur bestimmte Anwendungen sie sehen können. Durch die Konfiguration der Verschlüsselung auf Feldebene können Sie vertrauliche Informationen CloudFront, die von Benutzern übermittelt wurden, sicher auf Ihre Webserver hochladen. Die sensiblen Informationen, die Ihnen von Clients zur Verfügung gestellt werden, werden an dem Edge-Standort verschlüsselt, der näher am Benutzer

ist. Die Verschlüsselung bleibt für den gesamten Anwendungs-Stack erhalten. Auf diese Weise wird sichergestellt, dass nur Anwendungen, die die Daten benötigen (und über die Anmeldeinformationen zur Entschlüsselung verfügen) dazu in der Lage sind.

Weitere Informationen finden Sie unter Vertrauliche Daten durch Verschlüsselung auf Feldebene schützen.

Die CloudFront API-Endpunkte `cloudfront.amazonaws.com` und akzeptieren nur `cloudfront-fips.amazonaws.com` HTTPS-Verkehr. Das heißt, wenn Sie Informationen über die CloudFront API senden und empfangen, werden Ihre Daten — einschließlich Verteilungskonfigurationen, Cache-Richtlinien und Richtlinien für Ursprungsanfragen, Schlüsselgruppen und öffentliche Schlüssel sowie Funktionscode in CloudFront Funktionen — bei der Übertragung immer verschlüsselt. Darüber hinaus werden alle Anfragen, die an die CloudFront API-Endpunkte gesendet werden, mit Anmeldeinformationen signiert und angemeldet. AWS AWS CloudTrail

Der Funktionscode und die Konfiguration in CloudFront Functions werden bei der Übertragung immer verschlüsselt, wenn sie an den Edge-Standort (Points of PresencePOPs) und zwischen anderen Speicherorten, die von CloudFront verwendet werden, kopiert werden.

Verschlüsselung im Ruhezustand

Der Funktionscode und die Konfiguration in CloudFront Functions werden immer in einem verschlüsselten Format am Edge-Standort und an anderen Speicherorten gespeichert POPs, die von verwendet werden CloudFront.

Einschränken des Zugriffs auf Inhalte

Viele Unternehmen, die Inhalte über das Internet bereitstellen, möchten den Zugriff auf Dokumente, Geschäftsdaten, Medien-Streams oder Inhalte, die nur für ausgewählte Benutzer gedacht sind, beschränken. Um diese Inhalte mithilfe von Amazon sicher bereitzustellen CloudFront, können Sie einen oder mehrere der folgenden Schritte ausführen:

Verwenden Sie signierte Cookies URLs oder Cookies

> Sie können den Zugriff auf Inhalte einschränken, die für bestimmte Nutzer bestimmt sind — beispielsweise Nutzer, die eine Gebühr bezahlt haben —, indem Sie diese privaten Inhalte mithilfe CloudFront signierter oder signierter Cookies bereitstellen. URLs Weitere Informationen finden Sie unter Stellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit.

Beschränken des Zugriffs auf Inhalte in Amazon S3-Buckets

Wenn Sie den Zugriff auf Ihre Inhalte einschränken, indem Sie beispielsweise CloudFront signierte URLs oder signierte Cookies verwenden, möchten Sie auch nicht, dass andere Personen Dateien über die direkte URL der Datei aufrufen. Stattdessen möchten Sie, dass der Zugriff auf die Dateien nur über die CloudFront -URL möglich ist, sodass Ihre Schutzmechanismen funktionieren.

Wenn Sie einen Amazon S3 S3-Bucket als Ursprung für eine CloudFront Distribution verwenden, können Sie eine Origin Access Control (OAC) einrichten, die es ermöglicht, den Zugriff auf den S3-Bucket einzuschränken. Weitere Informationen finden Sie unter the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”.

Beschränken des Zugriffs auf Inhalte von einem Application Load Balancer

Wenn Sie einen Application Load Balancer in Elastic Load Balancing als Ursprung verwenden CloudFront , können Sie die Konfiguration so konfigurierenCloudFront , dass Benutzer nicht direkt auf den Application Load Balancer zugreifen können. Auf diese Weise können Benutzer nur über den Application Load Balancer auf den Application Load Balancer zugreifen CloudFront, sodass Sie alle Vorteile der Verwendung nutzen CloudFront können. Weitere Informationen finden Sie unter Beschränken Sie den Zugriff auf Application Load Balancers.

Verwenden Sie das Internet AWS WAF ACLs

Sie können einen Firewall-Dienst für Webanwendungen verwenden AWS WAF, um eine Web-Zugriffskontrollliste (Web ACL) zu erstellen, um den Zugriff auf Ihre Inhalte einzuschränken. Basierend auf den von Ihnen angegebenen Bedingungen, wie z. B. den IP-Adressen, von denen Anfragen stammen, oder den Werten von Abfragezeichenfolgen, CloudFront reagiert der Dienst auf Anfragen entweder mit dem angeforderten Inhalt oder mit einem HTTP-403-Statuscode (Forbidden). Weitere Informationen finden Sie unter AWS WAF Schutzmaßnahmen verwenden.

Verwenden von geografische Einschränkungen

Mithilfe einer geografischen Einschränkung oder Geoblockierung können Sie verhindern, dass Benutzer aus bestimmten geografischen Regionen auf Inhalte zugreifen, die Sie über eine CloudFront-Verteilung bereitstellen. Beim Konfigurieren geografischer Einschränkungen können Sie aus verschiedenen Optionen wählen. Weitere Informationen finden Sie unter Beschränken Sie die geografische Verteilung Ihrer Inhalte.

Identity and Access Management für Amazon CloudFront

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf Ressourcen sicher zu AWS kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Ressourcen zu verwenden. CloudFront IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Themen

Zielgruppe

Die Art und Weise, wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, in der Sie tätig sind. CloudFront

Dienstbenutzer — Wenn Sie den CloudFront Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die erforderlichen Anmeldeinformationen und Berechtigungen zur Verfügung. Wenn Sie für Ihre Arbeit mehr CloudFront Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Funktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern müssen. Unter Fehlerbehebung bei CloudFront Amazon-Identität und Zugriff finden Sie nützliche Informationen für den Fall, dass Sie keinen Zugriff auf eine Feature in CloudFront haben.

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für die CloudFront Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf CloudFront. Es ist Ihre Aufgabe, zu bestimmen, auf welche CloudFront Funktionen und Ressourcen Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anforderungen an Ihren IAM-Administrator senden, um die Berechtigungen der Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die

Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM nutzen kann CloudFront, finden Sie unterSo CloudFront arbeitet Amazon mit IAM.

IAM-Administrator: Wenn Sie als IAM-Administrator fungieren, sollten Sie Einzelheiten dazu kennen, wie Sie Richtlinien zur Verwaltung des Zugriffs auf CloudFront verfassen können. Beispiele für CloudFront identitätsbasierte Richtlinien, die Sie in IAM verwenden können, finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon CloudFront

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen als IAM-Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAM Identity Center) -Benutzer, die Single Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als Verbundidentität anmelden, hat der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie über den Verbund darauf zugreifen AWS , übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter So melden Sie sich bei Ihrem an AWS-Konto im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, um Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch zu signieren. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode für die Selbstsignierung von Anforderungen finden Sie unter AWS Signature Version 4 für API-Anforderungen im IAM-Benutzerhandbuch.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen bereitstellen. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter Multi-Faktor-Authentifizierung im AWS IAM Identity Center - Benutzerhandbuch und AWS Multi-Faktor-Authentifizierung (MFA) in IAM im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Sie können darauf zugreifen, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen. Verwenden Sie diese nur, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter Aufgaben, die Root-Benutzer-Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Verbundidentität

Als bewährte Methode sollten menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen den Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensbenutzerverzeichnis, einem Web-Identitätsanbieter AWS Directory Service, dem Identity Center-Verzeichnis oder einem beliebigen Benutzer, der mithilfe AWS-Services von Anmeldeinformationen zugreift, die über eine Identitätsquelle bereitgestellt wurden. Wenn föderierte Identitäten darauf zugreifen AWS-Konten, übernehmen sie Rollen, und die Rollen stellen temporäre Anmeldeinformationen bereit.

Für die zentrale Zugriffsverwaltung empfehlen wir Ihnen, AWS IAM Identity Center zu verwenden. Sie können Benutzer und Gruppen in IAM Identity Center erstellen, oder Sie können eine Verbindung zu einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und diese synchronisieren, um sie in all Ihren AWS-Konten Anwendungen zu verwenden. Informationen zu IAM Identity Center finden Sie unter Was ist IAM Identity Center? im AWS IAM Identity Center - Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAM-Benutzer ist eine Identität innerhalb Ihres Unternehmens AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wenn möglich, empfehlen wir, temporäre Anmeldeinformationen zu verwenden, anstatt IAM-Benutzer zu erstellen, die langfristige Anmeldeinformationen wie Passwörter und Zugriffsschlüssel haben. Bei speziellen Anwendungsfällen, die langfristige Anmeldeinformationen mit IAM-Benutzern erfordern, empfehlen wir jedoch, die Zugriffsschlüssel zu rotieren. Weitere Informationen finden Sie unter Regelmäßiges

Rotieren von Zugriffsschlüsseln für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Eine IAM-Gruppe ist eine Identität, die eine Sammlung von IAM-Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise eine Gruppe benennen IAMAdminsund dieser Gruppe Berechtigungen zur Verwaltung von IAM-Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter Anwendungsfälle für IAM-Benutzer im IAM-Benutzerhandbuch.

IAM-Rollen

Eine IAM-Rolle ist eine Identität innerhalb von Ihnen AWS-Konto , die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, jedoch nicht mit einer bestimmten Person verknüpft. Um vorübergehend eine IAM-Rolle in der zu übernehmen AWS Management Console, können Sie von einer Benutzer- zu einer IAM-Rolle (Konsole) wechseln. Sie können eine Rolle übernehmen, indem Sie eine AWS CLI oder AWS API-Operation aufrufen oder eine benutzerdefinierte URL verwenden. Weitere Informationen zu Methoden für die Verwendung von Rollen finden Sie unter Methoden für die Übernahme einer Rolle im IAM-Benutzerhandbuch.

IAM-Rollen mit temporären Anmeldeinformationen sind in folgenden Situationen hilfreich:

- Verbundbenutzerzugriff – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie unter Erstellen von Rollen für externe Identitätsanbieter (Verbund) im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Wenn Sie steuern möchten, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in IAM. Informationen zu Berechtigungssätzen finden Sie unter Berechtigungssätze im AWS IAM Identity Center -Benutzerhandbuch.
- Temporäre IAM-Benutzerberechtigungen – Ein IAM-Benutzer oder eine -Rolle kann eine IAM-Rolle übernehmen, um vorübergehend andere Berechtigungen für eine bestimmte Aufgabe zu erhalten.

- Kontoübergreifender Zugriff – Sie können eine IAM-Rolle verwenden, um einem vertrauenswürdigen Prinzipal in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zu den Unterschieden zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.
- Serviceübergreifender Zugriff — Einige AWS-Services verwenden Funktionen in anderen AWS-Services. Wenn Sie beispielsweise in einem Service einen Anruf tätigen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
  - Forward Access Sessions (FAS) — Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen an AWS-Service nachgelagerte Dienste zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.
  - Servicerolle – Eine Servicerolle ist eine IAM-Rolle, die ein Service übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.
  - Dienstbezogene Rolle — Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.
- Auf Amazon ausgeführte Anwendungen EC2 — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein

Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Verwenden einer IAM-Rolle, um Berechtigungen für Anwendungen zu gewähren, die auf EC2 Amazon-Instances ausgeführt werden.

Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Die Berechtigungen in den Richtlinien legen fest, ob eine Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu Struktur und Inhalten von JSON-Richtliniendokumenten finden Sie unter Übersicht über JSON-Richtlinien im IAM-Benutzerhandbuch.

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

IAM-Richtlinien definieren Berechtigungen für eine Aktion unabhängig von der Methode, die Sie zur Ausführung der Aktion verwenden. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen von der AWS Management Console AWS CLI, der oder der AWS API abrufen.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter

Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können AWS-Konto. Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie und einer Inline-Richtlinie wählen, finden Sie unter Auswählen zwischen verwalteten und eingebundenen Richtlinien im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Zugriffskontrolllisten () ACLs

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3 und Amazon VPC sind Beispiele für Dienste, die Unterstützung ACLs bieten. AWS WAF Weitere Informationen finden Sie unter Übersicht über ACLs die Zugriffskontrollliste (ACL) im Amazon Simple Storage Service Developer Guide.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- Berechtigungsgrenzen – Eine Berechtigungsgrenze ist ein erweitertes Feature, mit der Sie die maximalen Berechtigungen festlegen können, die eine identitätsbasierte Richtlinie einer IAM-Entität (IAM-Benutzer oder -Rolle) erteilen kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter Berechtigungsgrenzen für IAM-Entitäten im IAM-Benutzerhandbuch.
- Dienststeuerungsrichtlinien (SCPs) — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Objekte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Root-Benutzer des AWS-Kontos Entitäten. Weitere Informationen zu Organizations und SCPs finden Sie unter Richtlinien zur Servicesteuerung im AWS Organizations Benutzerhandbuch.
- Ressourcenkontrollrichtlinien (RCPs) — RCPs sind JSON-Richtlinien, mit denen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten festlegen können, ohne die IAM-Richtlinien aktualisieren zu müssen, die jeder Ressource zugeordnet sind, deren Eigentümer Sie sind. Das RCP schränkt die Berechtigungen für Ressourcen in Mitgliedskonten ein und kann sich auf die effektiven Berechtigungen für Identitäten auswirken, einschließlich der Root-Benutzer des AWS-Kontos, unabhängig davon, ob sie zu Ihrer Organisation gehören. Weitere Informationen zu Organizations RCPs, einschließlich einer Liste AWS-Services dieser Support-Leistungen RCPs, finden Sie unter Resource Control Policies (RCPs) im AWS Organizations Benutzerhandbuch.
- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und

der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAM-Benutzerhandbuch unter Bewertungslogik für Richtlinien.

So CloudFront arbeitet Amazon mit IAM

Bevor Sie IAM verwenden, um den Zugriff auf zu verwalten CloudFront, sollten Sie sich darüber informieren, mit welchen IAM-Funktionen Sie arbeiten können. CloudFront

IAM-Funktionen, die Sie mit Amazon verwenden können CloudFront

IAM-Feature	CloudFront Unterstützung
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Nein
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Richtlinienbedingungsschlüssel (services pezifisch)	Ja
ACLs	Nein
ABAC (Tags in Richtlinien)	Teilweise
Temporäre Anmeldeinformationen	Ja
Forward Access Sessions (FAS)	Nein

IAM-Feature	CloudFront Unterstützung
Servicerollen	Nein
Serviceverknüpfte Rollen	Ja

Einen allgemeinen Überblick darüber, wie CloudFront und andere AWS Dienste mit den meisten IAM-Funktionen funktionieren, finden Sie im IAM-Benutzerhandbuch unter AWS Dienste, die mit IAM funktionieren.

Identitätsbasierte Richtlinien für CloudFront

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der IAM-Referenz für JSON-Richtlinienelemente im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien für CloudFront

Beispiele für CloudFront identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon CloudFront

Ressourcenbasierte Richtlinien finden Sie in CloudFront

Unterstützt ressourcenbasierte Richtlinien: Nein

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und

Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource unterscheiden AWS-Konten, muss ein IAM-Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource erteilen. Sie erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen. Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich. Weitere Informationen finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Richtlinienaktionen für CloudFront

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API-Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keinen passenden API-Vorgang gibt. Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Eine Liste der CloudFront Aktionen finden Sie unter Von Amazon definierte Aktionen CloudFront in der Service Authorization Reference.

Bei Richtlinienaktionen wird vor der Aktion das folgende Präfix CloudFront verwendet:

```
cloudfront
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [
      "cloudfront:action1",
      "cloudfront:action2"
         ]
```

Beispiele für CloudFront identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon CloudFront

Politische Ressourcen für CloudFront

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Anweisungen müssen entweder ein – `Resource`oder ein `NotResource`-Element enthalten. Als bewährte Methode geben Sie eine Ressource mit dem zugehörigen Amazon-Ressourcennamen (ARN) an. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"
```

Eine Liste der CloudFront Ressourcentypen und ihrer ARNs Eigenschaften finden Sie unter Von Amazon definierte Ressourcen CloudFront in der Service Authorization Reference. Informationen darüber, mit welchen Aktionen Sie den ARN jeder Ressource angeben können, finden Sie unter Von Amazon definierte Aktionen CloudFront.

Beispiele für CloudFront identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon CloudFront

Bedingungsschlüssel für Richtlinien für CloudFront

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition` block) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. Sie können bedingte Ausdrücke erstellen, die Bedingungsoperatoren verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen AND-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, AWS wertet die Bedingung mithilfe einer logischen OR Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Beispielsweise können Sie einem IAM-Benutzer die Berechtigung für den Zugriff auf eine Ressource nur dann gewähren, wenn sie mit dessen IAM-Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie unter IAM-Richtlinienelemente: Variablen und Tags im IAM-Benutzerhandbuch.

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontextschlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

Eine Liste der CloudFront Bedingungsschlüssel finden Sie unter Bedingungsschlüssel für Amazon CloudFront in der Service Authorization Reference. Informationen zu den Aktionen und Ressourcen, mit denen Sie einen Bedingungsschlüssel verwenden können, finden Sie unter Von Amazon definierte Aktionen CloudFront.

Beispiele für CloudFront identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon CloudFront

ACLs in CloudFront

Unterstützt ACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

ABAC mit CloudFront

Unterstützt ABAC (Tags in Richtlinien): Teilweise

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen basierend auf Attributen definiert werden. In AWS werden diese Attribute als Tags bezeichnet. Sie können Tags an IAM-Entitäten (Benutzer oder Rollen) und an viele AWS Ressourcen anhängen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt von ABAC. Anschließend entwerfen Sie ABAC-Richtlinien, um Operationen zuzulassen, wenn das Tag des Prinzipals mit dem Tag der Ressource übereinstimmt, auf die sie zugreifen möchten.

ABAC ist in Umgebungen hilfreich, die schnell wachsen, und unterstützt Sie in Situationen, in denen die Richtlinienverwaltung mühsam wird.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter Definieren von Berechtigungen mit ABAC-Autorisierung im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe Attributbasierte Zugriffskontrolle (ABAC) verwenden im IAM-Benutzerhandbuch.

CloudFront unterstützt ABAC nur für Distributionen.

Verwenden temporärer Anmeldeinformationen mit CloudFront

Unterstützt temporäre Anmeldeinformationen: Ja

Einige funktionieren AWS-Services nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich Informationen, die mit temporären

Anmeldeinformationen AWS-Services funktionieren AWS-Services , finden Sie im IAM-Benutzerhandbuch unter Diese Option funktioniert mit IAM.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Passwort anmelden. Wenn Sie beispielsweise AWS über den Single Sign-On-Link (SSO) Ihres Unternehmens darauf zugreifen, werden bei diesem Vorgang automatisch temporäre Anmeldeinformationen erstellt. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Wechseln von Rollen finden Sie unter Wechseln von einer Benutzerrolle zu einer IAM-Rolle (Konsole) im IAM-Benutzerhandbuch.

Mithilfe der AWS API AWS CLI oder können Sie temporäre Anmeldeinformationen manuell erstellen. Sie können diese temporären Anmeldeinformationen dann für den Zugriff verwenden AWS. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Sicherheitsanmeldeinformationen in IAM.

Zugriffssitzungen weiterleiten für CloudFront

Unterstützt Forward Access Sessions (FAS): Nein

Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anforderung, Anfragen an nachgelagerte Dienste AWS-Service zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für CloudFront

Unterstützt Servicerollen: Nein

Eine Servicerolle ist eine IAM-Rolle, die ein Service annimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.

> **Warning**
>
> Durch das Ändern der Berechtigungen für eine Servicerolle kann die CloudFront Funktionalität beeinträchtigt werden. Bearbeiten Sie Servicerollen nur, CloudFront wenn Sie dazu eine Anleitung erhalten.

Dienstbezogene Rollen für CloudFront

Unterstützt dienstbezogene Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

CloudFront verwendet dienstbezogene Rollen, um Aktionen für Sie auszuführen. Weitere Informationen zum Erstellen oder Verwalten von CloudFront dienstbezogenen Rollen finden Sie unter. Verwenden von serviceverknüpften Rollen für CloudFront Weitere Informationen zum Erstellen oder Verwalten von dienstverknüpften Lambda @Edge -Rollen finden Sie unter. Serviceverknüpfte Rollen für Lambda@Edge

Details zum Erstellen oder Verwalten von serviceverknüpften Rollen finden Sie unter AWS -Services, die mit IAM funktionieren. Suchen Sie in der Tabelle nach einem Service mit einem `Yes` in der Spalte Service-linked role (Serviceverknüpfte Rolle). Wählen Sie den Link Yes (Ja) aus, um die Dokumentation für die serviceverknüpfte Rolle für diesen Service anzuzeigen.

Beispiele für identitätsbasierte Richtlinien für Amazon CloudFront

Benutzer und Rollen haben standardmäßig nicht die Berechtigung, CloudFront-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mithilfe der API AWS Management Console, AWS Command Line Interface (AWS CLI) oder AWS ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter Erstellen von IAM-Richtlinien (Konsole) im IAM-Benutzerhandbuch.

Einzelheiten zu Aktionen und Ressourcentypen, die von definiert wurden CloudFront, einschließlich des Formats von ARNs für jeden der Ressourcentypen, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon CloudFront in der Service Authorization Reference.

Themen

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand CloudFront Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Beginnen Sie mit AWS verwalteten Richtlinien und wechseln Sie zu Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um Ihren Benutzern und Workloads zunächst Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter AWS -verwaltete Richtlinien oder AWS -verwaltete Richtlinien für Auftrags-Funktionen im IAM-Benutzerhandbuch.
- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum

Anwenden von Berechtigungen finden Sie unter Richtlinien und Berechtigungen in IAM im IAM-Benutzerhandbuch.

- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. AWS CloudFormation B. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.
- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter Richtlinienvalidierung mit IAM Access Analyzer im IAM-Benutzerhandbuch.
- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter Sicherer API-Zugriff mit MFA im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter Bewährte Methoden für die Sicherheit in IAM im IAM-Benutzerhandbuch.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der API oder. AWS CLI AWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Berechtigungen für den programmgesteuerten Zugriff CloudFront

Das folgende Beispiel zeigt eine Berechtigungsrichtlinie. Der Abschnitt `Sid` (die Anweisungs-ID) ist optional.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowAllCloudFrontPermissions",
         "Effect": "Allow",
```

```
            "Action": ["cloudfront:*"],
            "Resource": "*"
        }
    ]
}
```

Die Richtlinie gewährt Berechtigungen zur Ausführung aller CloudFront Operationen, was für den programmgesteuerten Zugriff CloudFront ausreicht. Wenn Sie die Konsole für den Zugriff verwenden CloudFront, finden Sie weitere Informationen unter. Für die Verwendung der CloudFront Konsole sind Berechtigungen erforderlich

Eine Liste der Aktionen und den ARN, den Sie angeben, um die Erlaubnis zur Verwendung der einzelnen Aktionen zu erteilen oder zu verweigern, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon CloudFront in der Service Authorization Reference.

Für die Verwendung der CloudFront Konsole sind Berechtigungen erforderlich

Um vollen Zugriff auf die CloudFront Konsole zu gewähren, gewähren Sie die Berechtigungen in der folgenden Berechtigungsrichtlinie:

JSON

```
{
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "acm:ListCertificates",
            "cloudfront:*",
            "cloudwatch:DescribeAlarms",
            "cloudwatch:PutMetricAlarm",
            "cloudwatch:GetMetricStatistics",
            "elasticloadbalancing:DescribeLoadBalancers",
            "iam:ListServerCertificates",
            "sns:ListSubscriptionsByTopic",
            "sns:ListTopics",
            "waf:GetWebACL",
            "waf:ListWebACLs"
         ],
         "Resource":"*"
```

```
      },
      {
         "Effect":"Allow",
         "Action":[
            "s3:ListAllMyBuckets",
            "s3:PutBucketPolicy"
         ],
         "Resource":"arn:aws:s3:::*"
      }
   ]
}
```

Gründe, warum die Berechtigungen erforderlich sind

`acm:ListCertificates`

Wenn Sie Distributionen mithilfe der CloudFront Konsole erstellen und aktualisieren und so konfigurieren CloudFront möchten, dass HTTPS zwischen dem Betrachter und CloudFront oder zwischen CloudFront und dem Ursprung erforderlich ist, können Sie eine Liste der ACM-Zertifikate anzeigen.

Diese Berechtigung ist nicht erforderlich, wenn Sie die CloudFront Konsole nicht verwenden.

`cloudfront:*`

Ermöglicht es Ihnen, alle CloudFront Aktionen auszuführen.

`cloudwatch:DescribeAlarms` und **`cloudwatch:PutMetricAlarm`**

Ermöglicht das Erstellen und Anzeigen von CloudWatch Alarmen in der CloudFront Konsole. Weitere Informationen finden Sie auch unter `sns:ListSubscriptionsByTopic` und `sns:ListTopics`.

Diese Berechtigungen sind nicht erforderlich, wenn Sie die CloudFront-Konsole nicht verwenden.

`cloudwatch:GetMetricStatistics`

Lassen Sie uns CloudWatch Metriken in der CloudFront Konsole CloudFront rendern.

Diese Berechtigung ist nicht erforderlich, wenn Sie die CloudFront Konsole nicht verwenden.

`elasticloadbalancing:DescribeLoadBalancers`

Zeigt beim Erstellen und Aktualisieren von Verteilungen eine Liste der Elastic-Load-Balancing-Load-Balancer in der Liste der verfügbaren Ursprünge an.

Diese Berechtigung ist nicht erforderlich, wenn Sie die CloudFront Konsole nicht verwenden.

`iam:ListServerCertificates`

Wenn Sie Distributionen mithilfe der CloudFront Konsole erstellen und aktualisieren und so konfigurieren CloudFront möchten, dass HTTPS zwischen dem Viewer und CloudFront oder zwischen dem CloudFront Ursprung erforderlich ist, können Sie eine Liste der Zertifikate im IAM-Zertifikatsspeicher anzeigen.

Diese Berechtigung ist nicht erforderlich, wenn Sie die CloudFront Konsole nicht verwenden.

`s3:ListAllMyBuckets`

ermöglicht beim Erstellen und Aktualisieren von Verteilungen die Durchführung der folgenden Operationen:

- Anzeigen einer Liste von S3-Buckets in der Liste der verfügbaren Ursprünge
- Anzeigen einer Liste von S3-Buckets, in denen Sie die Zugriffsprotokolle speichern können

Diese Berechtigung ist nicht erforderlich, wenn Sie die CloudFront Konsole nicht verwenden.

`S3:PutBucketPolicy`

Wenn Sie Verteilungen erstellen oder aktualisieren, die den Zugriff auf S3-Buckets beschränken, kann der Benutzer die Bucket-Richtlinie so aktualisieren, mit Zugriff auf die CloudFront-Ursprungszugriffsidentität gewährt wird. Weitere Informationen finden Sie unter the section called “Verwenden Sie eine ursprüngliche Zugriffsidentität (veraltet, nicht empfohlen)”.

Diese Berechtigung ist nicht erforderlich, wenn Sie die CloudFront Konsole nicht verwenden.

`sns:ListSubscriptionsByTopic` und **`sns:ListTopics`**

Wenn Sie CloudWatch Alarme in der CloudFront Konsole erstellen, können Sie ein SNS-Thema für Benachrichtigungen auswählen.

Diese Berechtigungen sind nicht erforderlich, wenn Sie die CloudFront-Konsole nicht verwenden.

`waf:GetWebACL` und **`waf:ListWebACLs`**

Ermöglicht das Anzeigen einer Liste von AWS WAF Websites ACLs in der CloudFront Konsole.

Diese Berechtigungen sind nicht erforderlich, wenn Sie die CloudFront-Konsole nicht verwenden.

Aktionen für die Konsole, für die nur Zugriffsrechte erforderlich sind CloudFront

Auf der Seite CloudFront Security Savings Bundle können Sie die folgenden CloudFront Aktionen ausführen. Die folgenden API-Aktionen sind nicht dafür vorgesehen, von Ihrem Code aufgerufen zu werden, und sind auch nicht im AWS CLI und enthalten AWS SDKs.

Aktion	Beschreibung
`CreateSavingsPlan`	Erteilt die Erlaubnis, einen neuen Sparplan zu erstellen.
`GetSavingsPlan`	Erteilt die Erlaubnis, einen Sparplan zu erstellen.
`ListRateCards`	Erteilt die Erlaubnis, CloudFront Preiskarten für das Konto aufzulisten.
`ListSavingsPlans`	Erteilt die Erlaubnis, Sparpläne im Konto aufzulisten.
`ListUsages`	Erteilt die Erlaubnis, CloudFront Nutzungsdaten aufzulisten.
`UpdateSavingsPlan`	Erteilt die Erlaubnis, einen Sparplan zu aktualisieren.

Hinweise

- Weitere Informationen zu CloudFront Sparplänen finden Sie im Bereich CloudFront Sicherheitssparpaket bei Amazon CloudFront FAQs.
- Wenn Sie einen Sparplan für erstellen CloudFront und ihn später löschen möchten, wenden Sie sich an AWS -Support.

Beispiele für vom Kunden verwaltete Richtlinien

Sie können Ihre eigenen benutzerdefinierten IAM-Richtlinien erstellen, um Berechtigungen für CloudFront API-Aktionen zuzulassen. Die benutzerdefinierten Richtlinien können Sie dann den IAM-

Benutzern oder -Gruppen zuweisen, welche die angegebenen Berechtigungen benötigen. Diese Richtlinien funktionieren, wenn Sie die CloudFront API AWS SDKs, die oder die AWS CLI verwenden. Die folgenden Beispiele zeigen Berechtigungen für einige häufige Anwendungsfälle. Die Richtlinie, die einem Benutzer vollen Zugriff gewährt CloudFront, finden Sie unterFür die Verwendung der CloudFront Konsole sind Berechtigungen erforderlich.

Beispiele

- Beispiel 1: Lesezugriff auf alle Verteilungen gewähren
- Beispiel 2: Erstellen, Aktualisieren und Löschen von Verteilungen erlauben
- Beispiel 3: Erstellen und Auflisten von Invalidierungen erlauben
- Beispiel 4: Erlauben Sie das Erstellen einer Distribution

Beispiel 1: Lesezugriff auf alle Verteilungen gewähren

Die folgende Berechtigungsrichtlinie gewährt dem Benutzer Berechtigungen zum Anzeigen aller Distributionen in der CloudFront Konsole:

JSON

```
{
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "acm:ListCertificates",
            "cloudfront:GetDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:ListDistributions",
            "cloudfront:ListCloudFrontOriginAccessIdentities",
            "elasticloadbalancing:DescribeLoadBalancers",
            "iam:ListServerCertificates",
            "sns:ListSubscriptionsByTopic",
            "sns:ListTopics",
            "waf:GetWebACL",
            "waf:ListWebACLs"
         ],
         "Resource":"*"
      },
```

```
      {
         "Effect":"Allow",
         "Action":[
            "s3:ListAllMyBuckets"
         ],
         "Resource":"arn:aws:s3:::*"
      }
   ]
}
```

Beispiel 2: Erstellen, Aktualisieren und Löschen von Verteilungen erlauben

Die folgende Berechtigungsrichtlinie ermöglicht es Benutzern, Distributionen mithilfe der Konsole zu erstellen, zu aktualisieren und zu löschen: CloudFront

JSON

```
{
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "acm:ListCertificates",
            "cloudfront:CreateDistribution",
            "cloudfront:DeleteDistribution",
            "cloudfront:GetDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:ListDistributions",
            "cloudfront:UpdateDistribution",
            "cloudfront:ListCloudFrontOriginAccessIdentities",
            "elasticloadbalancing:DescribeLoadBalancers",
            "iam:ListServerCertificates",
            "sns:ListSubscriptionsByTopic",
            "sns:ListTopics",
            "waf:GetWebACL",
            "waf:ListWebACLs"
         ],
         "Resource":"*"
      },
      {
         "Effect":"Allow",
```

```
         "Action":[
            "s3:ListAllMyBuckets",
            "s3:PutBucketPolicy"
         ],
         "Resource":"arn:aws:s3:::*"
      }
   ]
}
```

Die `cloudfront:ListCloudFrontOriginAccessIdentities`-Berechtigung erlaubt es Benutzern, automatisch einer vorhandenen Ursprungszugriffsidentität die Berechtigung für den Zugriff auf Objekte in einem Amazon S3-Bucket zu erteilen. Wenn Sie außerdem möchten, dass Benutzer Ursprungszugriffsidentitäten erstellen können, müssen Sie auch die `cloudfront:CreateCloudFrontOriginAccessIdentity`-Berechtigung erteilen.

Beispiel 3: Erstellen und Auflisten von Invalidierungen erlauben

Die folgende Berechtigungsrichtlinie erlaubt Benutzern, Invalidierungen zu erstellen und aufzulisten. Sie beinhaltet Lesezugriff auf CloudFront Distributionen, da Sie Invalidierungen erstellen und anzeigen, indem Sie zuerst die Einstellungen für eine Distribution anzeigen:

JSON

```
{
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "acm:ListCertificates",
            "cloudfront:GetDistribution",
            "cloudfront:GetStreamingDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:ListDistributions",
            "cloudfront:ListCloudFrontOriginAccessIdentities",
            "cloudfront:CreateInvalidation",
            "cloudfront:GetInvalidation",
            "cloudfront:ListInvalidations",
            "elasticloadbalancing:DescribeLoadBalancers",
            "iam:ListServerCertificates",
            "sns:ListSubscriptionsByTopic",
```

```
            "sns:ListTopics",
            "waf:GetWebACL",
            "waf:ListWebACLs"
         ],
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":[
            "s3:ListAllMyBuckets"
         ],
         "Resource":"arn:aws:s3:::*"
      }
   ]
}
```

Beispiel 4: Erlauben Sie das Erstellen einer Distribution

Die folgende Berechtigungsrichtlinie gewährt dem Benutzer die Berechtigung, Distributionen in der CloudFront Konsole zu erstellen und aufzulisten. Geben Sie für die `CreateDistribution` Aktion das Platzhalterzeichen (*) für den `Resource` anstelle eines Platzhalters für die Distribution ARN (`arn:aws:cloudfront::123456789012:distribution/*`) an. Weitere Informationen zu diesem `Resource` Element finden Sie unter IAM-JSON-Richtlinienelemente: Ressource im IAM-Benutzerhandbuch.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "cloudfront:CreateDistribution",
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "cloudfront:ListDistributions",
            "Resource": "*"
```

```
        }
    ]
}
```

AWS verwaltete Richtlinien für Amazon CloudFront

Um Benutzern, Gruppen und Rollen Berechtigungen hinzuzufügen, ist es einfacher, AWS verwaltete Richtlinien zu verwenden, als Richtlinien selbst zu schreiben. Es erfordert Zeit und Fachwissen, um von Kunden verwaltete IAM-Richtlinien zu erstellen, die Ihren Benutzern nur die benötigten Berechtigungen bieten. Um schnell loszulegen, können Sie unsere AWS verwalteten Richtlinien verwenden. Diese Richtlinien decken allgemeine Anwendungsfälle ab und sind in Ihrem AWS-Konto verfügbar. Weitere Informationen zu AWS verwalteten Richtlinien finden Sie im IAM-Benutzerhandbuch unter AWS Verwaltete Richtlinien.

AWS Dienste verwalten und aktualisieren AWS verwaltete Richtlinien. Sie können die Berechtigungen in AWS verwalteten Richtlinien nicht ändern. Services fügen einer von AWS verwalteten Richtlinien gelegentlich zusätzliche Berechtigungen hinzu, um neue Features zu unterstützen. Diese Art von Update betrifft alle Identitäten (Benutzer, Gruppen und Rollen), an welche die Richtlinie angehängt ist. Services aktualisieren eine von AWS verwaltete Richtlinie am ehesten, wenn eine neue Funktion gestartet wird oder neue Berechtigungen verfügbar werden. Dienste entfernen keine Berechtigungen aus einer AWS verwalteten Richtlinie, sodass durch Richtlinienaktualisierungen Ihre bestehenden Berechtigungen nicht beeinträchtigt werden.

AWS Unterstützt außerdem verwaltete Richtlinien für Jobfunktionen, die sich über mehrere Dienste erstrecken. Die ReadOnlyAccess AWS verwaltete Richtlinie bietet beispielsweise schreibgeschützten Zugriff auf alle AWS Dienste und Ressourcen. Wenn ein Dienst eine neue Funktion startet, werden nur Leseberechtigungen für neue Operationen und Ressourcen AWS hinzugefügt. Eine Liste und Beschreibungen der Richtlinien für Auftragsfunktionen finden Sie in Verwaltete AWS -Richtlinien für Auftragsfunktionen im IAM-Leitfaden.

Themen

- AWS verwaltete Richtlinie: CloudFrontReadOnlyAccess
- AWS verwaltete Richtlinie: CloudFrontFullAccess
- AWS verwaltete Richtlinie: AWSCloud FrontLogger

- AWS verwaltete Richtlinie: AWSLambda Replicator
- AWS verwaltete Richtlinie: Front AWSCloud VPCOrigin ServiceRolePolicy
- CloudFront Aktualisierungen AWS verwalteter Richtlinien

AWS verwaltete Richtlinie: CloudFrontReadOnlyAccess

Sie können die CloudFrontReadOnlyAccess-Richtlinie an Ihre IAM-Identitäten anfügen. Diese Richtlinie ermöglicht nur Leseberechtigungen für Ressourcen. CloudFront Sie ermöglicht auch schreibgeschützte Berechtigungen für andere AWS Dienstressourcen, die sich auf die Konsole beziehen CloudFront und in der Konsole sichtbar sind. CloudFront

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `cloudfront:Describe*`— Ermöglicht Prinzipalen das Abrufen von Informationen über Metadaten zu Ressourcen. CloudFront
- `cloudfront:Get*`— Ermöglicht Prinzipalen, detaillierte Informationen und Konfigurationen für CloudFront Ressourcen abzurufen.
- `cloudfront:List*`— Ermöglicht Prinzipalen das Abrufen von Ressourcenlisten. CloudFront
- `cloudfront-keyvaluestore:Describe*`— Ermöglicht Prinzipalen das Abrufen von Informationen über den Schlüsselwertspeicher.
- `cloudfront-keyvaluestore:Get*`- Ermöglicht es den Prinzipalen, detaillierte Informationen und Konfigurationen für den Schlüsselwertspeicher abzurufen.
- `cloudfront-keyvaluestore:List*`- Ermöglicht Prinzipalen das Abrufen von Listen der Schlüsselwertspeicher.
- `acm:DescribeCertificate`— Ermöglicht Prinzipalen das Abrufen von Details zu einem ACM-Zertifikat.
- `acm:ListCertificates` – Ermöglicht es Prinzipalen, eine Liste von ACM-Zertifikaten abzurufen.
- `iam:ListServerCertificates` – Ermöglicht es Prinzipalen, eine Liste der in IAM gespeicherten Serverzertifikate abzurufen.
- `route53:List*` – Ermöglicht es Prinzipalen, Listen von Route 53-Ressourcen abzurufen.

- `waf:ListWebACLs`— Ermöglicht Prinzipalen das Abrufen einer Liste von Webeingängen. ACLs AWS WAF
- `waf:GetWebACL`— Ermöglicht es Prinzipalen, detaillierte Informationen über das Web ACLs in abzurufen. AWS WAF
- `wafv2:ListWebACLs`— Ermöglicht es Prinzipalen, eine Liste der eingegebenen Websites ACLs abzurufen. AWS WAF
- `wafv2:GetWebACL`— Ermöglicht es Prinzipalen, detaillierte Informationen über das Web ACLs in abzurufen. AWS WAF

Informationen zu den Berechtigungen für diese Richtlinie finden Sie unter CloudFrontReadOnlyAccess in der Referenz zu von AWS verwalteten Richtlinien.

AWS verwaltete Richtlinie: CloudFrontFullAccess

Sie können die CloudFrontFullAccess-Richtlinie an Ihre IAM-Identitäten anfügen. Diese Richtlinie ermöglicht Administratorberechtigungen für CloudFront Ressourcen. Sie ermöglicht auch schreibgeschützte Berechtigungen für andere AWS Dienstressourcen, die sich auf die Konsole beziehen CloudFront und in der CloudFront Konsole sichtbar sind.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `s3:ListAllMyBuckets` – Ermöglicht es Prinzipalen, eine Liste aller Amazon S3 Buckets abzurufen.
- `acm:DescribeCertificate`— Ermöglicht Prinzipalen, Details zu einem ACM-Zertifikat abzurufen.
- `acm:ListCertificates` – Ermöglicht es Prinzipalen, eine Liste von ACM-Zertifikaten abzurufen.
- `acm:RequestCertificate`— Ermöglicht Prinzipalen, verwaltete Zertifikate von ACM anzufordern.
- `cloudfront:*`— Ermöglicht Prinzipalen, alle Aktionen für alle Ressourcen auszuführen. CloudFront
- `cloudfront-keyvaluestore:*`- Ermöglicht Prinzipalen, alle Aktionen im Schlüsselwertspeicher auszuführen.
- `iam:ListServerCertificates` – Ermöglicht es Prinzipalen, eine Liste der in IAM gespeicherten Serverzertifikate abzurufen.

- `waf:ListWebACLs`— Ermöglicht es den Prinzipalen, eine Liste der eingegebenen Websites ACLs abzurufen. AWS WAF
- `waf:GetWebACL`— Ermöglicht es Prinzipalen, detaillierte Informationen über das Web ACLs in abzurufen. AWS WAF
- `wafv2:ListWebACLs`— Ermöglicht es Prinzipalen, eine Liste der eingegebenen Websites ACLs abzurufen. AWS WAF
- `wafv2:GetWebACL`— Ermöglicht es Prinzipalen, detaillierte Informationen über das Web ACLs in abzurufen. AWS WAF
- `kinesis:ListStreams` – Ermöglicht Prinzipalen das Abrufen einer Liste von Amazon Kinesis Streams.
- `ec2:DescribeInstances`- Ermöglicht Principals, detaillierte Informationen zu Instances in Amazon EC2 zu erhalten.
- `elasticloadbalancing:DescribeLoadBalancers`- Ermöglicht Prinzipalen, detaillierte Informationen über Load Balancer in Elastic Load Balancing abzurufen.
- `ec2:DescribeInternetGateways`- Ermöglicht Principals, detaillierte Informationen über Internet-Gateways bei Amazon zu erhalten. EC2
- `kinesis:DescribeStream` – Ermöglicht Prinzipalen das Abrufen detaillierter Informationen zu einem Kinesis Stream.
- `iam:ListRoles` – Erlaubt Prinzipalen das Abrufen einer Liste von Rollen in IAM.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie unter CloudFrontFullAccess in der Referenz zu von AWS verwalteten Richtlinien.

> **Important**
>
> Wenn Sie Zugriffsprotokolle erstellen und speichern CloudFront möchten, müssen Sie zusätzliche Berechtigungen erteilen. Weitere Informationen finden Sie unter Berechtigungen.

AWS verwaltete Richtlinie: AWSCloud FrontLogger

Sie können die AWSCloudFrontLoggerRichtlinie nicht an Ihre IAM-Identitäten anhängen. Diese Richtlinie ist mit einer dienstbezogenen Rolle verknüpft, mit der Sie Aktionen CloudFront in Ihrem Namen ausführen können. Weitere Informationen finden Sie unter the section called “Serviceverknüpfte Rollen für Lambda@Edge”.

Diese Richtlinie ermöglicht es CloudFront , Protokolldateien an Amazon zu übertragen CloudWatch. Details zu Berechtigungen dieser Richtlinie finden Sie unter the section called “Dienstbezogene Rollenberechtigungen für Logger CloudFront”.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie unter AWSCloudFrontLogger in der Referenz zu von AWS verwalteten Richtlinien.

AWS verwaltete Richtlinie: AWSLambda Replicator

Sie können die AWSLambdaReplicator-Richtlinie nicht an Ihre IAM-Identitäten anhängen. Diese Richtlinie ist mit einer dienstbezogenen Rolle verknüpft, die es ermöglicht, Aktionen CloudFront in Ihrem Namen durchzuführen. Weitere Informationen finden Sie unter the section called “Serviceverknüpfte Rollen für Lambda@Edge”.

Diese Richtlinie ermöglicht das Erstellen CloudFront , Löschen und Deaktivieren von Funktionen, AWS Lambda auf die Lambda @Edge -Funktionen repliziert werden sollen. AWS-Regionen Details zu Berechtigungen dieser Richtlinie finden Sie unter the section called “Serviceverknüpfte Rollenberechtigungen für Lambda Replicator”.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie unter AWSLambdaReplicator in der Referenz für AWS verwaltete Richtlinien.

AWS verwaltete Richtlinie: Front AWSCloud VPCOrigin ServiceRolePolicy

Sie können die AWSCloudVPCOriginServiceRolePolicyFront-Richtlinie nicht an Ihre IAM-Entitäten anhängen. Diese Richtlinie ist mit einer dienstbezogenen Rolle verknüpft, mit der Sie Aktionen CloudFront in Ihrem Namen ausführen können. Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für CloudFront.

Diese Richtlinie ermöglicht CloudFront die Verwaltung EC2 elastischer Netzwerkschnittstellen und Sicherheitsgruppen in Ihrem Namen. Details zu Berechtigungen dieser Richtlinie finden Sie unter the section called “Dienstbezogene Rollenberechtigungen für CloudFront VPC Origins”.

Die Berechtigungen für diese Richtlinie finden Sie unter AWSCloudFront VPCOrigin ServiceRolePolicy in der Referenz für AWS verwaltete Richtlinien.

CloudFront Aktualisierungen AWS verwalteter Richtlinien

Hier finden Sie Informationen zu Aktualisierungen AWS verwalteter Richtlinien CloudFront seit Beginn der Nachverfolgung dieser Änderungen durch diesen Dienst. Abonnieren Sie den RSS-Feed auf der

Seite CloudFront Dokumentenverlauf, um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten.

Änderung	Beschreibung	Datum
CloudFrontReadOnlyAccess – Aktualisierung auf eine bestehende Richtlinie	CloudFront neue Berechtigung für ACM hinzugefügt. Die neue Berechtigung ermöglicht es Prinzipalen, Details zu einem ACM-Zerti fikat abzurufen.	28. April 2025
CloudFrontFullAccess – Aktualisierung auf eine bestehende Richtlinie	CloudFront neue Berechtig ungen für ACM hinzugefügt. Die neuen Berechtigungen ermöglichen es Prinzipal en, Details zu einem ACM-Zertifikat abzurufen und ein verwaltetes Zertifikat von ACM anzufordern.	28. April 2025
CloudFrontFullAccess – Aktualisierung auf eine bestehende Richtlinie	CloudFront neue Berechtig ungen für Amazon EC2 und Elastic Load Balancing hinzugefügt. Die neuen Berechtigungen ermöglichen es CloudFront , detaillierte Informationen über Load Balancer in Elastic Load Balancing und Instances und Internet-Gateways in Amazon abzurufen. EC2	20. November 2024

Änderung	Beschreibung	Datum
AWSCloudVorderseite VPCOrigin ServiceRolePolicy — Neue Richtlinie	CloudFront hat eine neue Richtlinie hinzugefügt. Diese Richtlinie ermöglicht CloudFront die Verwaltung EC2 elastischer Netzwerks chnittstellen und Sicherhei tsgruppen in Ihrem Namen.	20. November 2024
CloudFrontReadOnlyAccess und CloudFrontFullAccess – Update von zwei vorhandenen Richtlinien.	CloudFront neue Berechtig ungen für Key-Value-Stores hinzugefügt. Die neuen Berechtigungen ermöglichen es Benutzern , Informationen über Key-Value-Stores abzurufen und entsprechende Maßnahmen zu ergreifen.	19. Dezember 2023
CloudFrontReadOnlyAccess – Aktualisierung auf eine bestehende Richtlinie	CloudFront eine neue Berechtigung zur Beschreib ung von CloudFront Funktione n hinzugefügt. Diese Berechtigung ermöglicht es dem Benutzer, der Gruppe oder der Rolle, Informati onen und Metadaten über eine Funktion zu lesen, nicht jedoch den Code der Funktion.	8. September 2021
CloudFront hat begonnen, Änderungen zu verfolgen	CloudFront hat begonnen, Änderungen für die AWS verwalteten Richtlinien zu verfolgen.	8. September 2021

Verwenden von serviceverknüpften Rollen für CloudFront

Amazon CloudFront verwendet AWS Identity and Access Management (IAM) serviceverknüpfte Rollen. Eine serviceverknüpfte Rolle ist eine einzigartige Art von IAM-Rolle, mit der direkt verknüpft ist. CloudFront Mit Diensten verknüpfte Rollen sind vordefiniert CloudFront und enthalten alle Berechtigungen, die der Dienst benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen.

Eine dienstbezogene Rolle CloudFront erleichtert die Einrichtung, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. CloudFront definiert die Berechtigungen ihrer dienstbezogenen Rollen und CloudFront kann, sofern nicht anders definiert, nur ihre Rollen übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Diese Berechtigungsrichtlinie kann keinen anderen IAM-Entitäten zugewiesen werden.

Sie können eine serviceverknüpfte Rolle erst löschen, nachdem ihre verwandten Ressourcen gelöscht wurden. Dadurch werden Ihre CloudFront Ressourcen geschützt, da Sie nicht versehentlich die Zugriffsberechtigung für die Ressourcen entziehen können.

Informationen zu anderen Diensten, die dienstverknüpfte Rollen unterstützen, finden Sie unter AWS Dienste, die mit IAM funktionieren. Suchen Sie in der Spalte Dienstverknüpfte Rollen nach den Diensten, für die Ja steht. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Dienstbezogene Rollenberechtigungen für CloudFront VPC Origins

CloudFront VPC Origins verwendet die serviceverknüpfte Rolle mit dem Namen AWSServiceRoleForCloudFrontVPCOrigin— Ermöglicht CloudFront die Verwaltung EC2 elastischer Netzwerkschnittstellen und Sicherheitsgruppen in Ihrem Namen.

Die serviceverknüpfte Rolle AWSServiceRoleForCloudFrontVPCOrigin vertraut darauf, dass die folgenden Services die Rolle annehmen:

- `vpcorigin.cloudfront.amazonaws.com`

Die Rollenberechtigungsrichtlinie mit dem Namen AWSCloud Front VPCOrigin ServiceRolePolicy ermöglicht es CloudFront VPC Origins, die folgenden Aktionen für die angegebenen Ressourcen durchzuführen:

- Aktion: `ec2:CreateNetworkInterface` für `arn:aws:ec2:*:*:network-interface/*`

- Aktion: `ec2:CreateNetworkInterface` an und `arn:aws:ec2:*:*:subnet/*` `arn:aws:ec2:*:*:security-group/*`
- Aktion: `ec2:CreateSecurityGroup` für `arn:aws:ec2:*:*:security-group/*`
- Aktion: `ec2:CreateSecurityGroup` für `arn:aws:ec2:*:*:vpc/*`
- Aktion:`ec2:ModifyNetworkInterfaceAttribute`, `ec2:DeleteNetworkInterfaceec2:DeleteSecurityGroup`,`ec2:AssignIpv6Addresses`, und `ec2:UnassignIpv6Addresses` weiter `all AWS resources that the actions support`
- Aktion:`ec2:DescribeNetworkInterfaces`, `ec2:DescribeSecurityGroupsec2:DescribeInstances`,`ec2:DescribeInternetGateways`,`ec2` und `ec2:DescribeAddresses` weiter `all AWS resources that the actions support`
- Aktion: `ec2:CreateTags` an `arn:aws:ec2:*:*:security-group/*` und `arn:aws:ec2:*:*:network-interface/*`
- Aktion: `elasticloadbalancing:DescribeLoadBalancerselasticloadbalancing:DescribeListener` und `elasticloadbalancing:DescribeTargetGroups` weiter `all AWS resources that the actions support`

Sie müssen Berechtigungen konfigurieren, damit eine Benutzer, Gruppen oder Rollen eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen können. Weitere Informationen finden Sie unter serviceverknüpfte Rollenberechtigung im IAM-Benutzerhandbuch.

Eine serviceverknüpfte Rolle für CloudFront VPC Origins erstellen

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Wenn Sie einen VPC-Ursprung in der AWS Management Console, der oder der AWS API erstellen AWS CLI, erstellt CloudFront VPC Origins die serviceverknüpfte Rolle für Sie.

Wenn Sie diese serviceverknüpfte Rolle löschen und sie dann erneut erstellen müssen, können Sie dasselbe Verfahren anwenden, um die Rolle in Ihrem Konto neu anzulegen. Wenn Sie einen VPC-Ursprung erstellen, erstellt CloudFront VPC Origins die serviceverknüpfte Rolle erneut für Sie.

Eine serviceverknüpfte Rolle für CloudFront VPC Origins bearbeiten

CloudFront In VPC Origins können Sie die AWSServiceRoleForCloudFrontVPCOrigin serviceverknüpfte Rolle nicht bearbeiten. Da möglicherweise verschiedene Entitäten auf die Rolle

verweisen, kann der Rollenname nach dem Erstellen einer serviceverknüpften Rolle nicht mehr geändert werden. Sie können jedoch die Beschreibung der Rolle mit IAM bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Löschen Sie eine serviceverknüpfte Rolle für CloudFront VPC Origins

Wenn Sie ein Feature oder einen Dienst, die bzw. der eine serviceverknüpften Rolle erfordert, nicht mehr benötigen, sollten Sie diese Rolle löschen. Auf diese Weise haben Sie keine ungenutzte juristische Stelle, die nicht aktiv überwacht oder verwaltet wird. Sie müssen jedoch die Ressourcen für Ihre serviceverknüpften Rolle zunächst bereinigen, bevor Sie sie manuell löschen können.

> **Note**
>
> Wenn der CloudFront Dienst die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu löschen, schlägt das Löschen möglicherweise fehl. Wenn dies passiert, warten Sie einige Minuten und versuchen Sie es erneut.

Um CloudFront VPC Origins-Ressourcen zu löschen, die von der AWSServiceRoleForCloudFrontVPCOrigin

- Löschen Sie die VPC-Ursprungsressourcen in Ihrem Konto.
  - Es kann einige Zeit dauern CloudFront , bis das Löschen der Ressourcen aus Ihrem Konto abgeschlossen ist. Wenn Sie die serviceverknüpfte Rolle nicht sofort löschen können, warten Sie und versuchen Sie es erneut.

So löschen Sie die serviceverknüpfte Rolle mit IAM

Verwenden Sie die IAM-Konsole, die oder die AWS API AWS CLI, um die AWSServiceRoleForCloudFrontVPCOrigin dienstverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter Löschen einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Unterstützte Regionen für serviceverknüpfte Rollen mit CloudFront VPC Origins

CloudFront VPC Origins unterstützt nicht die Verwendung von serviceverknüpften Rollen in allen Regionen, in denen der Service verfügbar ist. Sie können die Rolle AWSServiceRoleForCloudFrontVPCOrigin in den folgenden Regionen verwenden.

Name der Region	Regions-ID	Support in CloudFront
USA Ost (Nord-Virginia)	us-east-1	Ja
USA Ost (Ohio)	us-east-2	Ja
USA West (Nordkalifornien)	us-west-1 (außer AZ usw1-az2)	Ja
USA West (Oregon)	us-west-2	Ja
Afrika (Kapstadt)	af-south-1	Ja
Asien-Pazifik (Hongkong)	ap-east-1	Ja
Asien-Pazifik (Jakarta)	ap-southeast-3	Ja
Asien-Pazifik (Melbourne)	ap-southeast-4	Ja
Asien-Pazifik (Mumbai)	ap-south-1	Ja
Asien-Pazifik (Hyderabad)	ap-south-2	Ja
Asien-Pazifik (Osaka)	ap-northeast-3	Ja
Asien-Pazifik (Seoul)	ap-northeast-2	Ja
Asien-Pazifik (Singapore)	ap-southeast-1	Ja
Asien-Pazifik (Sydney)	ap-southeast-2	Ja
Asien-Pazifik (Tokyo)	ap-northeast-1 (außer AZ apne1-az3)	Ja
Kanada (Zentral)	ca-central-1 (außer AZ cac1-az3)	Ja
Kanada West (Calgary)	ca-west-1	Ja
Europa (Frankfurt)	eu-central-1	Ja

Name der Region	Regions-ID	Support in CloudFront
Europa (Ireland)	eu-west-1	Ja
Europa (London)	eu-west-2	Ja
Europa (Mailand)	eu-south-1	Ja
Europa (Paris)	eu-west-3	Ja
Europa (Spanien)	eu-south-2	Ja
Europa (Stockholm)	eu-north-1	Ja
Europa (Zürich)	eu-central-2	Ja
Israel (Tel Aviv)	il-central-1	Ja
Naher Osten (Bahrain)	me-south-1	Ja
Naher Osten (VAE)	me-central-1	Ja
Südamerika (São Paulo)	sa-east-1	Ja

Fehlerbehebung bei CloudFront Amazon-Identität und Zugriff

Mithilfe der folgenden Informationen können Sie häufig auftretende Probleme diagnostizieren und beheben, die bei der Arbeit mit CloudFront und IAM auftreten können.

Themen

- Ich bin nicht berechtigt, eine Aktion durchzuführen in CloudFront
- Ich bin nicht berechtigt, iam auszuführen: PassRole
- Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine CloudFront Ressourcen ermöglichen

Ich bin nicht berechtigt, eine Aktion durchzuführen in CloudFront

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht zur Durchführung einer Aktion berechtigt sind, müssen Ihre Richtlinien aktualisiert werden, damit Sie die Aktion durchführen können.

Der folgende Beispielfehler tritt auf, wenn der IAM-Benutzer `mateojackson` versucht, über die Konsole Details zu einer fiktiven *`my-example-widget`*-Ressource anzuzeigen, jedoch nicht über `cloudfront:`*`GetWidget`*-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 cloudfront:GetWidget on resource: my-example-widget
```

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `cloudfront:`*`GetWidget`*-Aktion auf die *`my-example-widget`*-Ressource zugreifen kann.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht zum Durchführen der `iam:PassRole`-Aktion autorisiert sind, müssen Ihre Richtlinien aktualisiert werden, um eine Rolle an CloudFront übergeben zu können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in CloudFront auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine CloudFront Ressourcen ermöglichen

Sie können eine Rolle erstellen, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob diese Funktionen CloudFront unterstützt werden, finden Sie unter. So CloudFront arbeitet Amazon mit IAM
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs auf einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter Gewähren von Zugriff für extern authentifizierte Benutzer (Identitätsverbund) im IAM-Benutzerhandbuch.
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Protokollierung und Überwachung in Amazon CloudFront

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Verfügbarkeit CloudFront und Leistung Ihrer AWS Lösungen. Sie sollten Überwachungsdaten aus allen Teilen Ihrer AWS Lösung sammeln, damit Sie einen etwaigen Ausfall an mehreren Stellen leichter debuggen können. AWS bietet verschiedene Tools zur Überwachung Ihrer CloudFront Ressourcen und Aktivitäten sowie zur Reaktion auf potenzielle Vorfälle:

CloudWatch Amazon-Alarme

Mithilfe von CloudWatch Alarmen beobachten Sie eine einzelne Metrik über einen von Ihnen festgelegten Zeitraum. Wenn die Metrik einen bestimmten Schwellenwert überschreitet, wird

eine Benachrichtigung an ein Amazon SNS SNS-Thema oder eine AWS Auto Scaling Richtlinie gesendet. CloudWatch Alarme lösen keine Aktionen aus, wenn sich eine Metrik in einem bestimmten Status befindet. Der Status muss sich stattdessen geändert haben und für eine festgelegte Anzahl an Zeiträumen aufrechterhalten worden sein.

Weitere Informationen finden Sie unter Überwachen Sie CloudFront Metriken mit Amazon CloudWatch.

AWS CloudTrail Logs

CloudTrail bietet eine Aufzeichnung der API-Aktionen, die von einem Benutzer, einer Rolle oder einem AWS Dienst in ausgeführt wurden CloudFront. Anhand der von gesammelten Informationen können Sie die API-Anfrage ermitteln CloudTrail, an die die Anfrage gestellt wurde CloudFront, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details.

Weitere Informationen finden Sie unter Protokollieren Amazon CloudFront Amazon-API-Aufrufen mit AWS CloudTrail.

CloudFront Standardprotokolle und Echtzeitprotokolle

CloudFront Protokolle enthalten detaillierte Aufzeichnungen über Anfragen, die an eine Distribution gestellt werden. Diese Protokolle sind für viele Anwendungen nützlich. Beispielsweise können Protokollinformationen bei Sicherheits- und Zugriffsprüfungen nützlich sein.

Weitere Informationen erhalten Sie unter Standardprotokollierung (Zugriffsprotokolle) und Erstellen und verwenden Sie Echtzeit-Protokollkonfigurationen.

Protokolle für Edge-Funktionen

Von Edge-Funktionen generierte Protokolle, sowohl CloudFront Functions als auch Lambda @Edge, werden direkt an Amazon CloudWatch Logs gesendet und nirgends von CloudFront gespeichert. CloudFront Functions verwendet eine dienstbezogene AWS Identity and Access Management (IAM) -Rolle, um von Kunden generierte Protokolle direkt an Logs in Ihrem Konto zu CloudWatch senden.

Weitere Informationen finden Sie unter Protokolle für Edge-Funktionen.

CloudFront Konsolenberichte

Die CloudFront Konsole enthält eine Vielzahl von Berichten, darunter den Cache-Statistikbericht, den Bericht über beliebte Objekte und den Bericht mit den häufigsten Verweisen. Die meisten CloudFront Konsolenberichte basieren auf den Daten in den CloudFront Zugriffsprotokollen,

die detaillierte Informationen über jede eingehende Benutzeranfrage enthalten CloudFront . Sie müssen Zugriffsprotokolle nicht aktivieren, um die Berichte anzeigen zu können.

Weitere Informationen finden Sie unter CloudFront Berichte in der Konsole anzeigen.

Konformitätsvalidierung für Amazon CloudFront

Externe Prüfer bewerten die Sicherheit und Konformität von Amazon im CloudFront Rahmen mehrerer AWS Compliance-Programme. Zu diesen Programmen gehören SOC, PCI, HIPAA und andere.

Eine Liste der AWS Services im Rahmen bestimmter Compliance-Programme finden Sie unter AWS Services im Umfang nach Compliance-Programmen. Allgemeine Informationen finden Sie unter AWS -Compliance-Programme.

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte herunterladen in AWS Artifact.

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung CloudFront hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. AWS stellt die folgenden Ressourcen zur Verfügung, die Sie bei der Einhaltung der Vorschriften unterstützen:

- Schnellstartanleitungen zu Sicherheit und Compliance — In diesen Bereitstellungsleitfäden werden architektonische Überlegungen erörtert und Schritte für die Implementierung von Umgebungen beschrieben, auf denen auf Sicherheit und Compliance ausgerichtete Basisumgebungen eingerichtet werden. AWS
- Architektur für HIPAA-Sicherheit und -Compliance am AWS — In diesem Whitepaper wird beschrieben, wie Unternehmen HIPAA-konforme Anwendungen entwickeln können. AWS

  Das AWS HIPAA-Compliance-Programm umfasst CloudFront (mit Ausnahme der Bereitstellung von Inhalten über Embedded) einen HIPAA-fähigen Service. CloudFront POPs Wenn Sie einen Business Associate Addendum (BAA) abgeschlossen haben AWS, können Sie ihn (mit Ausnahme der Inhaltsbereitstellung über CloudFront Embedded POPs) verwenden, um Inhalte bereitzustellen, die geschützte Gesundheitsinformationen CloudFront (PHI) enthalten. Weitere Informationen finden Sie unter HIPAA-Compliance.
- AWS Ressourcen zur Einhaltung von Vorschriften — Diese Sammlung von Arbeitsmappen und Leitfäden kann auf Ihre Branche und Ihren Standort zutreffen.

- AWS Config— Dieser AWS Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.
- AWS Security Hub— Dieser AWS Service nutzt Sicherheitskontrollen, um Ressourcenkonfigurationen und Sicherheitsstandards zu bewerten und Sie bei der Einhaltung verschiedener Compliance-Rahmenbedingungen zu unterstützen. Weitere Informationen zur Verwendung von Security Hub zur Bewertung von CloudFront Ressourcen finden Sie unter Amazon CloudFront Controls im AWS Security Hub Benutzerhandbuch.

CloudFront Bewährte Verfahren zur Einhaltung

In diesem Abschnitt finden Sie bewährte Verfahren und Empfehlungen zur Einhaltung der Vorschriften, wenn Sie Amazon für CloudFront die Bereitstellung Ihrer Inhalte verwenden.

Wenn Sie PCI-konforme oder HIPAA-konforme Workloads ausführen, die auf dem Modell der AWS gemeinsamen Verantwortung basieren, empfehlen wir Ihnen, Ihre CloudFront Nutzungsdaten der letzten 365 Tage für future Prüfzwecke zu protokollieren. Sie können Nutzungsdaten wie folgt protokollieren:

- Aktivieren CloudFront Sie die Zugriffsprotokolle. Weitere Informationen finden Sie unter Standardprotokollierung (Zugriffsprotokolle).
- Erfassen Sie Anfragen, die an die CloudFront API gesendet werden. Weitere Informationen finden Sie unter Protokollieren Amazon CloudFront Amazon-API-Aufrufen mit AWS CloudTrail.

Darüber hinaus finden Sie im Folgenden Einzelheiten zur Konformität mit den PCI-DSS- und SOC-Standards. CloudFront

Payment Card Industry Data Security Standard (PCI DSS)

CloudFront (mit Ausnahme der Inhaltsbereitstellung über CloudFront Embedded POPs) unterstützt die Verarbeitung, Speicherung und Übertragung von Kreditkartendaten durch einen Händler oder Dienstanbieter und wurde als konform mit dem Payment Card Industry (PCI) Data Security Standard (DSS) validiert. Weitere Informationen zu PCI DSS, einschließlich der Möglichkeit, eine Kopie des AWS PCI Compliance Package anzufordern, finden Sie unter PCI DSS Level 1.

Aus Sicherheitsgründen empfehlen wir, Kreditkarteninformationen nicht in CloudFront Edge-Caches zwischenzuspeichern. Sie können Ihren Absender beispielsweise so konfigurieren, dass Antworten, die Kreditkarteninformationen enthalten, wie z. B. die letzten vier Ziffern einer Kreditkartennummer

und die Kontaktinformationen des Karteninhabers, eine `Cache-Control:no-cache="` *`field-name`* `"` Kopfzeile enthalten.

System and Organization Controls (SOC)

CloudFront (mit Ausnahme der Bereitstellung von Inhalten über CloudFront Embedded POPs) entspricht den Maßnahmen zur System- und Organisationskontrolle (SOC), einschließlich SOC 1, SOC 2 und SOC 3. SOC-Berichte sind unabhängige Prüfungsberichte von Drittanbietern, die belegen, wie wichtige Compliance-Kontrollen und -Ziele AWS erreicht werden. Diese Audits stellen sicher, dass geeignete Sicherheitsmaßnahmen und Verfahren zum Schutz vor Beeinträchtigungen von Sicherheit, Vertraulichkeit und Verfügbarkeit von Kunden- und Unternehmensdaten vorhanden sind. Die Ergebnisse dieser Prüfungen durch Dritte sind auf der AWS SOC-Compliance-Website verfügbar. Dort finden Sie in den veröffentlichten Berichten weitere Informationen zu den Kontrollen, die den AWS Betrieb und die Einhaltung der Vorschriften unterstützen.

Resilienz bei Amazon CloudFront

Die AWS globale Infrastruktur basiert auf AWS Regionen und Availability Zones. AWS Regionen stellen mehrere physisch getrennte und isolierte Availability Zones bereit, die mit Netzwerken mit geringer Latenz, hohem Durchsatz und hochredundanten Vernetzungen verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Availability Zones ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser hoch verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu AWS Regionen und Availability Zones finden Sie unter AWS Globale Infrastruktur.

CloudFront Ursprungs-Failover

Zusätzlich zur Unterstützung der AWS globalen Infrastruktur CloudFront bietet Amazon eine Origin-Failover-Funktion, um Ihre Anforderungen an die Datenstabilität zu erfüllen. CloudFront ist ein globaler Service, der Ihre Inhalte über ein weltweites Netzwerk von Rechenzentren bereitstellt, die als Edge-Standorte oder Points of Presence () POPs bezeichnet werden. Wenn Ihre Inhalte nicht bereits an einem Edge-Standort zwischengespeichert sind, ruft CloudFront sie aus einem Ursprungsserver ab, den Sie als Quelle für die definitive Version des Inhalts identifiziert haben.

Sie können die Ausfallsichheit und Verfügbarkeit für bestimmte Szenarien erhöhen, indem Sie CloudFront mit einem Origin Failover einrichten. Zu Beginn erstellen Sie eine Ursprungsgruppe,

in der Sie einen primären Ursprung und einen CloudFront zweiten Ursprung angeben. CloudFront wechselt automatisch zum zweiten Ursprung, wenn der primäre Ursprung bestimmte HTTP-Statuscode-Fehlerantworten zurückgibt. Weitere Informationen finden Sie unter Optimieren Sie die Hochverfügbarkeit mit CloudFront Origin Failover.

Infrastruktursicherheit bei Amazon CloudFront

Als verwalteter Service CloudFront ist Amazon durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter AWS Cloud-Sicherheit. Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe für den Zugriff CloudFront über das Netzwerk. Kunden müssen Folgendes unterstützen:

- Transport Layer Security (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Verschlüsselungs-Suiten mit Perfect Forward Secrecy (PFS) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Außerdem müssen Anforderungen mit einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel signiert sein, der einem IAM-Prinzipal zugeordnet ist. Alternativ können Sie mit AWS Security Token Service (AWS STS) temporäre Sicherheitsanmeldeinformationen erstellen, um die Anforderungen zu signieren.

CloudFront Functions verwendet eine hochsichere Isolationsbarriere zwischen AWS Konten, um sicherzustellen, dass Kundenumgebungen vor Seitenkanalangriffen wie Spectre und Meltdown geschützt sind. Funktionen können nicht auf Daten anderer Kunden zugreifen oder diese ändern. Funktionen werden in einem dedizierten Single-Thread-Prozess auf einer dedizierten CPU ohne Hyperthreading ausgeführt An jedem beliebigen CloudFront Edge Location Point of Presence (POP) bedient CloudFront Functions jeweils nur einen Kunden, und alle kundenspezifischen Daten werden zwischen den Funktionsausführungen gelöscht.

Fehlerbehebung

In diesem Abschnitt können Sie häufig auftretende Probleme beheben, die bei der Einrichtung von Amazon für CloudFront den Vertrieb Ihrer Inhalte auftreten können.

Jedes Thema enthält ausführliche Anleitungen zur Identifizierung der Hauptursache häufiger Probleme sowie step-by-step Anleitungen zu deren Lösung.

Themen

Fehlerbehebung bei Problemen mit der Verteilung

Verwenden Sie die Informationen hier, um Ihnen bei der Diagnose und Behebung von Zertifikatsfehlern, Problemen mit verweigertem Zugriff oder anderen häufig auftretenden Problemen zu helfen, die bei der Einrichtung Ihrer Website oder Anwendung mit CloudFront Amazon-Distributionen auftreten können.

Themen

CloudFront gibt einen Fehler zurück Access Denied

Wenn Sie einen Amazon S3 S3-Bucket als Ursprung für Ihre CloudFront Distribution verwenden, wird in den folgenden Beispielen möglicherweise die Fehlermeldung Access Denied (403) angezeigt.

Inhalt

- Sie haben ein fehlendes Objekt aus dem Amazon S3 S3-Ursprung angegeben
- Ihrem Amazon S3 S3-Ursprung fehlen IAM-Berechtigungen
- Sie verwenden ungültige Anmeldeinformationen oder verfügen nicht über ausreichende Berechtigungen

Sie haben ein fehlendes Objekt aus dem Amazon S3 S3-Ursprung angegeben

Stellen Sie sicher, dass das angeforderte Objekt in Ihrem Bucket vorhanden ist. Bei Objektnamen wird zwischen Groß- und Kleinschreibung unterschieden. Die Eingabe eines ungültigen Objektnamens kann den Fehlercode „Zugriff verweigert“ zurückgeben.

Wenn Sie beispielsweise dem CloudFront Tutorial zum Erstellen einer Basisdistribution folgen, erstellen Sie einen Amazon S3 S3-Bucket als Ursprung und laden eine `index.html` Beispieldatei hoch.

Wenn Sie in Ihrem Webbrowser `https://d111111abcdef8.cloudfront.net/INDEX.HTML` statt von eingeben, wird möglicherweise eine ähnliche Meldung angezeigt`https://d111111abcdef8.cloudfront.net/index.html`, da bei der `index.html` Datei im URL-Pfad zwischen Groß- und Kleinschreibung unterschieden wird.

```
<Error>
<Code>AccessDenied</Code>
<Message>Access Denied</Message>
<RequestId>22Q367AHT7Y1ABCD</RequestId>
<HostId>
ABCDE/Vg+7PSNa/d/IfFQ8Fb92TGQ0KH0ZwG5iEKbc6+e06DdMS1ZW+ryB9GFRIVtS66rSSy6So=
</HostId>
</Error>
```

Ihrem Amazon S3 S3-Ursprung fehlen IAM-Berechtigungen

Stellen Sie sicher, dass Sie den richtigen Amazon S3 S3-Bucket als Ursprungsdomain und -namen ausgewählt haben. Der Ursprung (Amazon S3) muss über die richtigen Berechtigungen verfügen.

Wenn Sie nicht die richtigen Berechtigungen angeben, kann für Ihre Zuschauer die folgende Meldung „Zugriff verweigert“ angezeigt werden.

```
<Code>AccessDenied</Code>
<Message>User: arn:aws:sts::856369053181:assumed-role/OriginAccessControlRole/
EdgeCredentialsProxy+EdgeHostAuthenticationClient is not authorized to perform:
```

```
 kms:Decrypt on the resource associated with this ciphertext because the resource does
 not exist in this Region, no resource-based policies allow access, or a resource-based
 policy explicitly denies access</Message>
<RequestId>22Q367AHT7Y1ABCD/RequestId>
<HostId>
ABCDE/Vg+7PSNa/d/IfFQ8Fb92TGQ0KH0ZwG5iEKbc6+e06DdMS1ZW+ryB9GFRIVtS66rSSy6So=
</HostId>
</Error>
```

> **Note**
>
> In dieser Fehlermeldung handelt es sich bei der Konto-ID 856369053181 um ein verwaltetes Konto. AWS

Wenn Sie Inhalte von Amazon S3 verteilen und außerdem AWS Key Management Service (AWS KMS) service-side encryption (SSE-KMS) verwenden, müssen Sie zusätzliche IAM-Berechtigungen für den KMS-Schlüssel und den Amazon S3 S3-Bucket angeben. Ihre CloudFront Distribution benötigt diese Berechtigungen, um den KMS-Schlüssel zu verwenden, der für die Verschlüsselung des ursprünglichen Amazon S3 S3-Buckets verwendet wird.

Die Konfigurationen der Amazon S3 S3-Bucket-Richtlinie ermöglichen es der CloudFront Distribution, die verschlüsselten Objekte für die Inhaltsbereitstellung abzurufen.

Um Ihre Amazon S3 S3-Bucket- und KMS-Schlüsselberechtigungen zu überprüfen

1. Stellen Sie sicher, dass der KMS-Schlüssel, den Sie verwenden, derselbe Schlüssel ist, den Ihr Amazon S3 S3-Bucket für die Standardverschlüsselung verwendet. Weitere Informationen finden Sie unter Serverseitige Verschlüsselung mit AWS KMS (SSE-KMS) angeben im Amazon Simple Storage Service-Benutzerhandbuch.
2. Stellen Sie sicher, dass die Objekte im Bucket mit demselben KMS-Schlüssel verschlüsselt sind. Sie können ein beliebiges Objekt aus dem Amazon S3 S3-Bucket auswählen und die serverseitigen Verschlüsselungseinstellungen überprüfen, um den KMS-Schlüssel-ARN zu überprüfen.
3. Bearbeiten Sie die Amazon S3 S3-Bucket-Richtlinie, um die CloudFront Erlaubnis zu erteilen, den `GetObject` API-Vorgang vom Amazon S3 S3-Bucket aus aufzurufen. Ein Beispiel für eine Amazon S3 S3-Bucket-Richtlinie, die Origin Access Control verwendet, finden Sie unterErteilen Sie die CloudFront Erlaubnis, auf den S3-Bucket zuzugreifen.

4. Bearbeiten Sie die KMS-Schlüsselrichtlinie, um, und die CloudFront Erlaubnis zu erteilen `EncryptDecrypt`, die Aktionen auszuführen`GenerateDataKey*`. Geben Sie ein `Condition` Element an, sodass nur die angegebene Distribution die aufgelisteten Aktionen ausführen kann, damit nur die angegebene CloudFront Distribution die aufgelisteten Aktionen ausführen kann. Sie können die Richtlinie an Ihre bestehende AWS KMS Richtlinie anpassen. Ein Beispiel für eine KMS-Schlüsselrichtlinie finden Sie unterSSE-KMS.

Wenn Sie Origin Access Identity (OAI) anstelle von OAC verwenden, unterscheiden sich die Berechtigungen für den Amazon S3 S3-Bucket geringfügig, da Sie die Berechtigung für eine Identität statt für die erteilen. AWS-Service Weitere Informationen finden Sie unter Erteilen Sie einer Origin-Zugriffsidentität die Erlaubnis, Dateien im Amazon S3 S3-Bucket zu lesen.

Wenn Sie Ihre Dateien in Ihrer Distribution immer noch nicht anzeigen können, finden Sie weitere Informationen unter. Ich kann die Dateien in meiner Verteilung nicht anzeigen.

Sie verwenden ungültige Anmeldeinformationen oder verfügen nicht über ausreichende Berechtigungen

Eine Fehlermeldung „Zugriff verweigert“ kann angezeigt werden, wenn Sie falsche oder abgelaufene AWS SCT Anmeldeinformationen (Zugriffsschlüssel und geheimer Schlüssel) verwenden oder Ihrer IAM-Rolle oder Ihrem IAM-Benutzer die erforderliche Berechtigung zum Ausführen einer Aktion für eine CloudFront Ressource fehlt. Weitere Informationen zu Fehlermeldungen mit Zugriffsverweigerung finden Sie unter Problembehandlung bei Fehlermeldungen mit Zugriffsverweigerung im IAM-Benutzerhandbuch.

Informationen zur Verwendung von IAM mit finden Sie CloudFront unter. Identity and Access Management für Amazon CloudFront

CloudFront gibt einen InvalidViewerCertificate Fehler zurück, wenn ich versuche, einen alternativen Domainnamen hinzuzufügen

Wenn beim Versuch, Ihrer Distribution einen alternativen Domainnamen (CNAME) hinzuzufügen, ein `InvalidViewerCertificate` Fehler CloudFront zurückgegeben wird, lesen Sie sich die folgenden Informationen durch, um das Problem zu beheben. Dieser Fehler kann darauf hinweisen, dass Sie eines der folgenden Probleme lösen müssen, bevor Sie den alternativen Domänennamen erfolgreich hinzufügen können.

Die folgenden Fehler sind in der Reihenfolge aufgeführt, in der CloudFront geprüft wird, ob eine Autorisierung für das Hinzufügen eines alternativen Domainnamens vorliegt. Dies kann Ihnen bei der Behebung von Problemen helfen, da Sie anhand des CloudFront zurückgegebenen Fehlers feststellen können, welche Überprüfungsprüfungen erfolgreich abgeschlossen wurden.

Ihrer Verteilung ist kein Zertifikat angefügt.

Um einen alternativen Domänennamen (CNAME) hinzuzufügen, müssen Sie Ihrer Verteilung ein vertrauenswürdiges, gültiges Zertifikat hinzufügen. Überprüfen Sie die Anforderungen, erwerben Sie ein gültiges Zertifikat, das diese Anforderungen erfüllt, und wiederholen Sie den Vorgang. Weitere Informationen finden Sie unter Voraussetzungen für die Verwendung von alternativen Domänennamen.

Es gibt zu viele Zertifikate in der Zertifikatskette für das von Ihnen angefügte Zertifikat.

Die Zahl der Zertifikate in einer Zertifikatskette ist auf fünf beschränkt. Reduzieren Sie die Anzahl der Zertifikate in der Kette und wiederholen Sie den Vorgang.

Die Zertifikatskette enthält mindestens ein Zertifikat, das für das aktuelle Datum nicht gültig ist.

Die Zertifikatskette für ein von Ihnen hinzugefügtes Zertifikat enthält mindestens ein ungültiges Zertifikat, das entweder noch nicht gültig oder abgelaufen ist. Überprüfen Sie die Felder Not Valid Before (Nicht gültig vor) und Not Valid After (Nicht gültig nach) in den Zertifikaten in Ihrer Zertifikatskette, um sicherzustellen, dass alle Zertifikate basierend auf den von Ihnen aufgelisteten Daten gültig sind.

Das von Ihnen angefügte Zertifikat wurde nicht von einer vertrauenswürdigen Zertifizierungsstelle (Certificate Authority, CA) signiert.

Das Zertifikat, das Sie anhängen, CloudFront um einen alternativen Domainnamen zu verifizieren, kann kein selbstsigniertes Zertifikat sein. Es muss von einer vertrauenswürdigen CA signiert worden sein. Weitere Informationen finden Sie unter Voraussetzungen für die Verwendung von alternativen Domänennamen.

Das von Ihnen angefügte Zertifikat ist nicht korrekt formatiert.

Das Format des Domänennamens und der IP-Adresse im Zertifikat und das Format des Zertifikats selbst müssen den Standards für Zertifikate folgen.

Es ist ein CloudFront interner Fehler aufgetreten.

CloudFront wurde durch ein internes Problem blockiert und es konnten keine Validierungsprüfungen für Zertifikate durchgeführt werden. CloudFront Gibt in diesem Szenario

einen HTTP 500-Statuscode zurück und weist darauf hin, dass ein internes CloudFront Problem beim Anhängen des Zertifikats vorliegt. Warten Sie einige Minuten und wiederholen Sie dann den Vorgang, um dem Zertifikat den alternativen Domänennamen hinzuzufügen.

Das von Ihnen angefügte Zertifikat deckt den alternativen Domänennamen nicht ab, den Sie hinzufügen möchten.

Für jeden alternativen Domainnamen, den Sie hinzufügen, CloudFront müssen Sie ein gültiges SSL/TLS-Zertifikat von einer vertrauenswürdigen Zertifizierungsstelle (CA) anhängen, die den Domainnamen abdeckt, um Ihre Autorisierung zur Verwendung des Domainnamens zu überprüfen. Bitte aktualisieren Sie Ihr Zertifikat, sodass es einen Domänennamen enthält, der den CNAME-Wert abdeckt, den Sie hinzufügen möchten. Weitere Informationen und Beispiele zur Verwendung von Domänennamen mit Platzhalterzeichen finden Sie unter Voraussetzungen für die Verwendung von alternativen Domänennamen.

CloudFront gibt einen falsch konfigurierten DNS-Eintragsfehler zurück, wenn ich versuche, einen neuen CNAME hinzuzufügen

Wenn Sie einen vorhandenen DNS-Platzhaltereintrag haben, der auf eine CloudFront Distribution verweist, kann beim Versuch, einen neuen CNAME mit einem genaueren Namen hinzuzufügen, der folgende Fehler auftreten:

```
One or more aliases specified for the distribution includes an incorrectly configured
 DNS record that points to another CloudFront distribution. You must update the DNS
 record to correct the problem.
```

Dieser Fehler tritt auf, weil das DNS mit dem CNAME CloudFront abgefragt wird und der DNS-Platzhaltereintrag in eine andere Verteilung aufgelöst wird.

Um dieses Problem zu beheben, erstellen Sie zunächst eine weitere Verteilung und dann einen DNS-Eintrag, der auf die neue Verteilung verweist. Fügen Sie abschließend den spezifischeren CNAME hinzu. Weitere Informationen zum Hinzufügen finden Sie CNAMEs unterFügen Sie einen alternativen Domainnamen hinzu.

Ich kann die Dateien in meiner Verteilung nicht anzeigen.

Wenn Sie die Dateien in Ihrer CloudFront Distribution nicht anzeigen können, finden Sie in den folgenden Themen einige gängige Lösungen.

Haben Sie sich sowohl für Amazon S3 als CloudFront auch für Amazon S3 angemeldet?

Um Amazon CloudFront mit einem Amazon S3-Ursprung zu verwenden, müssen Sie sich CloudFront sowohl für Amazon S3 als auch für Amazon S3 separat registrieren. Weitere Informationen zur Registrierung für Amazon S3 CloudFront und Amazon S3 finden Sie unterRichten Sie Ihre ein AWS-Konto.

Sind Ihre Amazon-S3-Bucket- und Objektberechtigungen korrekt festgelegt?

Wenn Sie CloudFront mit einem Amazon S3 S3-Ursprung verwenden, werden die Originalversionen Ihrer Inhalte in einem S3-Bucket gespeichert. Um Ihren Zuschauern die Inhalte bereitzustellen, empfehlen wir Ihnen, CloudFront Origin Access Control (OAC) zu verwenden, um den Zugriff auf den Amazon S3 S3-Bucket zu sichern. Das bedeutet, dass Ihr S3-Bucket nur über CloudFront erreichbar ist. OAC steuert den Zuschauerzugriff und die sichere Bereitstellung über CloudFront. Weitere Informationen zu OAC finden Sie unter. the section called “Beschränken Sie den Zugriff auf einen Amazon S3 S3-Ursprung”

Weitere Informationen zur Verwaltung Ihres Bucket-Zugriffs finden Sie unter Blockieren des öffentlichen Zugriffs auf Ihren Amazon S3 S3-Speicher im Amazon S3 S3-Benutzerhandbuch.

Objekteigenschaften und Bucket-Eigenschaften existieren unabhängig voneinander. Sie müssen jedem Objekt in Amazon S3 explizit Berechtigungen zuweisen. Objekte erben ihre Eigenschaften nicht von Buckets; Objekteigenschaften müssen unabhängig vom jeweiligen Bucket festgelegt werden.

Ist Ihr alternativer Domänenname (CNAME) richtig konfiguriert?

Wenn Sie bereits über einen CNAME-Datensatz für den Namen Ihrer Domain verfügen, aktualisieren Sie den vorhandenen Datensatz oder ersetzen Sie ihn mit einem neuen, der auf den Domain-Namen Ihrer Verteilung verweist.

Stellen Sie darüber hinaus sicher, dass der CNAME-Datensatz auf den Domänennamen Ihrer Verteilung, und nicht auf das Amazon S3-Bucket zeigt. Sie können überprüfen, ob der CNAME-Datensatz in Ihrem DNS-System auf den Domainnamen Ihrer Verteilung zeigt. Verwenden Sie dafür ein DNS-Tool wie dig.

Das folgende Beispiel zeigt eine Dig-Anfrage für eine Domain mit dem Namen `images.example.com` sowie den relevanten Teil der Antwort. Unter `ANSWER SECTION` finden Sie eine Zeile, die den `CNAME`-Wert enthält. Der CNAME-Eintrag für Ihren Domainnamen ist korrekt

eingerichtet, wenn der Wert auf der rechten Seite von CNAME der Domainname Ihrer CloudFront Distribution ist. Wenn das Bucket des Amazon S3-Ursprungsservers oder ein anderer Domainname angegeben wird, ist der CNAME-Datensatz nicht korrekt eingerichtet.

```
[prompt]> dig images.example.com

; <<> DiG 9.3.3rc2 <<> images.example.com
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15917
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 2, ADDITIONAL: 0
;; QUESTION SECTION:
;images.example.com.   IN  A
;; ANSWER SECTION:
images.example.com. 10800 IN CNAME d111111abcdef8.cloudfront.net.
...
...
```

Weitere Informationen zu finden Sie CNAMEs unter. Verwenden Sie Benutzerdefiniert, URLs indem Sie alternative Domainnamen hinzufügen (CNAMEs)

Verweisen Sie auf die richtige URL für Ihre CloudFront Distribution?

Stellen Sie sicher, dass die URL, auf die Sie verweisen, den Domainnamen (oder CNAME) Ihrer CloudFront Distribution verwendet, nicht Ihren Amazon S3 S3-Bucket oder Ihren benutzerdefinierten Ursprung.

Benötigen Sie Hilfe bei der Fehlerbehebung in Zusammenhang mit einem benutzerdefinierten Ursprungsserver?

Wenn Sie bei der Fehlerbehebung AWS für einen benutzerdefinierten Ursprung helfen müssen, müssen wir wahrscheinlich die `X-Amz-Cf-Id` Header-Einträge Ihrer Anfragen überprüfen. Wenn Sie diese Einträge nicht bereits protokollieren, sollten Sie das für die Zukunft in Erwägung ziehen. Weitere Informationen finden Sie unter the section called “Verwenden Sie Amazon EC2 (oder einen anderen benutzerdefinierten Ursprung)”. Weitere Hilfe erhalten Sie im AWS -Supportcenter.

Fehlermeldung: Zertifikat: <certificate-id>wird verwendet von CloudFront

Problem: Sie versuchen, ein SSL/TLS-Zertifikat aus dem IAM-Zertifikatsspeicher zu löschen, und Sie erhalten die Meldung „Zertifikat: <certificate-id>wird verwendet von“. CloudFront

Lösung: Jede CloudFront Distribution muss entweder dem CloudFront Standardzertifikat oder einem benutzerdefinierten Zertifikat zugeordnet sein, um das SSL/TLS certificate. Before you can delete an SSL/TLS certificate, you must either rotate the certificate (replace the current custom SSL/TLS certificate with another custom SSL/TLS certificate) or revert from using a custom SSL/TLS Standardzertifikat verwenden zu können. CloudFront Um dies zu beheben, führen Sie die Schritte in einem der folgenden Verfahren aus:

- SSL/TLS Zertifikate rotieren
- Kehren Sie von einem benutzerdefinierten SSL/TLS-Zertifikat zum Standardzertifikat zurück CloudFront

Behebung von Statuscodes für die Fehlerantwort in CloudFront

Wenn CloudFront ein Objekt von Ihrem Absender angefordert wird und der Absender einen HTTP-Statuscode 4xx oder 5xx zurückgibt, liegt ein Problem mit der Kommunikation zwischen Ihnen CloudFront und Ihrem Absender vor.

Dieses Thema enthält auch Schritte zur Fehlerbehebung für diese Statuscodes bei der Verwendung von Lambda @Edge oder CloudFront Functions.

Die folgenden Themen enthalten detaillierte Erläuterungen zu den möglichen Ursachen für diese Fehlerreaktionen und bieten step-by-step Anleitungen zur Diagnose und Lösung der zugrunde liegenden Probleme.

Themen

- HTTP 400-Statuscode (Bad Request)
- HTTP 401-Statuscode (Nicht autorisiert)
- HTTP 403-Statuscode (Ungültige Methode)
- HTTP-Statuscode 403 (Erlaubnis verweigert)
- HTTP 404-Statuscode (nicht gefunden)
- HTTP 412-Statuscode (Vorbedingung fehlgeschlagen)
- HTTP 500-Statuscode (Interner Serverfehler)
- HTTP 502-Statuscode (Bad Gateway)
- HTTP 503-Statuscode (Service nicht verfügbar)
- HTTP 504-Statuscode (Gateway Timeout)

HTTP 400-Statuscode (Bad Request)

CloudFront gibt eine ungültige 400-Anforderung zurück, wenn der Client einige ungültige Daten in der Anfrage sendet, z. B. fehlende oder falsche Inhalte in der Nutzlast oder in den Parametern. Dies könnte auch einen generischen Client-Fehler darstellen.

Amazon S3 Origin gibt einen 400-Fehler zurück

Wenn Sie mit Ihrer CloudFront Distribution einen Amazon S3 S3-Ursprung verwenden, sendet Ihre Distribution möglicherweise Fehlerantworten mit dem HTTP-Statuscode 400 Bad Request und eine Meldung ähnlich der folgenden:

Der Autorisierungsheader ist falsch; die Region '*<AWS Region>*' ist falsch; " *<AWS Region>* wird erwartet

Zum Beispiel:

Der Autorisierungs-Header ist fehlerhaft; die Region "us-east-1" ist falsch; erwartet wird "us-west-2"

Dieses Problem kann im folgenden Szenario auftreten:

1. Der Ursprung Ihrer CloudFront Distribution ist ein Amazon S3 S3-Bucket.
2. Sie haben den S3-Bucket von einer AWS Region in eine andere verschoben. Das heißt, Sie haben den S3-Bucket gelöscht und später einen neuen Bucket mit demselben Bucket-Namen erstellt, aber in einer anderen AWS Region als der, in der sich der ursprüngliche S3-Bucket befand.

Um diesen Fehler zu beheben, aktualisieren Sie Ihre CloudFront Distribution so, dass sie den S3-Bucket in der aktuellen AWS Region des Buckets findet.

Um Ihre CloudFront Distribution zu aktualisieren

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudFront Konsole unterhttps://console.aws.amazon.com/cloudfront/v4/home.
2. Wählen Sie die Verteilung aus, die diesen Fehler verursacht.
3. Wählen Sie Ursprünge und Ursprungsgruppen aus.
4. Suchen Sie den Ursprung für den S3-Bucket, den Sie verschoben haben. Aktivieren Sie das Kontrollkästchen neben diesem Ursprung und wählen Sie dann Bearbeiten aus.
5. Wählen Sie Yes, Edit aus. Sie müssen keine Einstellungen ändern, bevor Sie Yes, Edit (Ja, Bearbeiten) auswählen.

Wenn Sie diese Schritte abgeschlossen haben, stellt Ihre CloudFront Distribution erneut bereit. Während der Bereitstellung der Distribution wird in der Spalte Letzte Änderung der Status Bereitgestellt angezeigt. Einige Zeit nach Abschluss der Bereitstellung sollten Sie keine `AuthorizationHeaderMalformed` Fehlerantworten mehr erhalten.

Der Ursprung des Application Load Balancer gibt einen 400-Fehler zurück

Wenn Sie mit Ihrer CloudFront Distribution einen Application Load Balancer Balancer-Ursprung verwenden, kann ein 400-Fehler unter anderem folgende Ursachen haben:

- Der Client hat eine falsch formatierte Anforderung gesendet, die die HTTP-Spezifikation nicht erfüllt.
- Der Anforderungsheader überschreitet 16 KB pro Anforderungszeile, 16 KB pro einzelnen Header oder 64 KB für den gesamten Anforderungsheader.
- Der Client hat die Verbindung beendet, bevor er den vollständigen Anfragetext gesendet hat.

HTTP 401-Statuscode (Nicht autorisiert)

Der Antwortstatuscode 401 Unauthorized weist darauf hin, dass die Client-Anfrage nicht abgeschlossen wurde, da sie keine gültigen Authentifizierungsdaten für die angeforderte Ressource enthält. Dieser Statuscode wird mit einem `WWW-Authenticate` HTTP-Antwort-Header gesendet, der Informationen darüber enthält, wie der Client die Ressource erneut anfordern kann, nachdem er den Benutzer zur Eingabe von Authentifizierungsdaten aufgefordert hat. Weitere Informationen finden Sie unter 401 Unauthorized.

Wenn Ihr Absender erwartet CloudFront, dass ein `Authorization` Header die Anfragen authentifiziert, CloudFront muss er den `Authorization` Header an den Ursprung weiterleiten, um den Fehler 401 Unauthorized zu vermeiden. Wenn Sie CloudFront eine Viewer-Anfrage an Ihren Ursprung weiterleiten, CloudFront werden standardmäßig einige Viewer-Header entfernt, einschließlich des Headers. `Authorization` Um sicherzustellen, dass Ihr Ursprung immer den `Authorization`-Header in Ursprungsanforderungen erhält, haben Sie folgende Möglichkeiten:

- Fügen Sie den `Authorization` Header mithilfe einer Cache-Richtlinie zum Cache-Schlüssel hinzu. Alle Header im Cache-Schlüssel werden automatisch in Ursprungsanforderungen eingeschlossen. Weitere Informationen finden Sie unter Steuern Sie den Cache-Schlüssel mit einer Richtlinie.
- Verwenden Sie eine Herkunftsanforderungsrichtlinie, die alle Betrachter-Header an den Ursprung weiterleitet. Sie können den `Authorization` Header in einer ursprünglichen

Anforderungsrichtlinie nicht einzeln weiterleiten. Wenn Sie jedoch alle Viewer-Header weiterleiten, CloudFront wird der `Authorization` Header in Viewer-Anfragen einbezogen. CloudFrontstellt die Richtlinie für verwaltete `AllViewer` Ursprungsanfragen für diesen Anwendungsfall bereit. Weitere Informationen finden Sie unter Richtlinien für verwaltete Origin-Anfragen verwenden.

Weitere Informationen finden Sie unter Wie kann ich so konfigurieren, CloudFront dass der Autorisierungsheader an den Ursprung weitergeleitet wird?

HTTP 403-Statuscode (Ungültige Methode)

CloudFront gibt einen 403-Fehler (Ungültige Methode) zurück, wenn Sie versuchen, eine HTTP-Methode zu verwenden, die Sie nicht in der CloudFront Distribution angegeben haben. Sie können eine der folgenden Optionen für Ihre Distribution angeben:

- CloudFront Nur Weiterleitungen `GET` und `HEAD` Anfragen.
- CloudFront nur Weiterleitungen `GET` und `HEAD` `OPTIONS` Anfragen.
- CloudFront leitet`GET`,,`HEAD`, `OPTIONS` `PUT` `PATCHPOST`, und `DELETE` Anfragen weiter. (Wenn Sie diese Option auswählen, müssen Sie möglicherweise den Zugriff auf Ihren Amazon S3 S3-Bucket oder Ihren benutzerdefinierten Ursprung einschränken, damit Benutzer keine Operationen ausführen können, die Sie nicht möchten. Möglicherweise möchten Sie nicht, dass Benutzer über Berechtigungen zum Löschen von Objekten von Ihrem Ursprung verfügen.

HTTP-Statuscode 403 (Erlaubnis verweigert)

Ein HTTP 403-Fehler bedeutet, dass der Client nicht berechtigt ist, auf die angeforderte Ressource zuzugreifen. Der Client versteht die Anfrage, kann den Zuschauerzugriff jedoch nicht autorisieren. Im Folgenden sind die häufigsten Ursachen aufgeführt, wenn dieser Statuscode CloudFront zurückgegeben wird:

Themen

- Alternativer CNAME ist falsch konfiguriert
- AWS WAF wird bei der CloudFront Verteilung oder am Ursprung konfiguriert
- Custom Origin gibt einen 403-Fehler zurück
- Amazon S3 Origin gibt einen 403-Fehler zurück
- Geografische Einschränkungen geben einen 403-Fehler zurück

- Die Konfiguration einer signierten URL oder eines signierten Cookies gibt einen 403-Fehler zurück
- Gestapelte Verteilungen verursachen einen 403-Fehler

Alternativer CNAME ist falsch konfiguriert

Stellen Sie sicher, dass Sie den richtigen CNAME für unsere Distribution angegeben haben. Um einen alternativen CNAME anstelle der CloudFront Standard-URL zu verwenden:

1. Erstellen Sie einen CNAME-Eintrag in Ihrem DNS, um den CNAME auf die Verteilungs-URL zu CloudFront verweisen.
2. Fügen Sie den CNAME zu Ihrer CloudFront Verteilungskonfiguration hinzu.

Wenn Sie den DNS-Eintrag erstellen, aber den CNAME nicht zu Ihrer CloudFront Verteilungskonfiguration hinzufügen, gibt die Anfrage einen 403-Fehler zurück. Weitere Informationen zur Konfiguration eines benutzerdefinierten CNAME-Codes finden Sie unter. Verwenden Sie Benutzerdefiniert, URLs indem Sie alternative Domainnamen hinzufügen (CNAMEs)

AWS WAF wird bei der CloudFront Verteilung oder am Ursprung konfiguriert

Wenn AWS WAF sich zwischen dem Client und befindet CloudFront, CloudFront kann nicht zwischen einem 403-Fehlercode, der von Ihrem Ursprung zurückgegeben wird, und einem 403-Fehlercode, der zurückgegeben wird, AWS WAF wenn eine Anfrage blockiert wird, unterschieden werden.

Um die Quelle des 403-Statuscodes zu finden, überprüfen Sie Ihre ACL-Regel (AWS WAF Web Access Control List) auf eine blockierte Anfrage. Weitere Informationen finden Sie unter den folgenden Themen:

- AWS WAF Web-Zugriffskontrolllisten (WebACLs)
- Testen und Optimieren Ihrer AWS WAF Schutzmaßnahmen

Custom Origin gibt einen 403-Fehler zurück

Wenn Sie einen benutzerdefinierten Ursprung verwenden, wird möglicherweise ein 403-Fehler angezeigt, wenn Sie am Ursprung eine benutzerdefinierte Firewall-Konfiguration haben. Um Fehler zu beheben, richten Sie die Anfrage direkt an den Ursprung. Wenn Sie den Fehler auch ohne replizieren können CloudFront, verursacht der Ursprung den 403-Fehler.

Wenn der benutzerdefinierte Ursprung den Fehler verursacht, überprüfen Sie die Ursprungsprotokolle, um herauszufinden, was den Fehler verursacht haben könnte. Weitere Informationen finden Sie in den folgenden Themen zur Problembehandlung:

- Wie behebe ich HTTP 403-Fehler von API Gateway?
- Wie behebe ich verbotene HTTP 403-Fehler im Application Load Balancer?

Amazon S3 Origin gibt einen 403-Fehler zurück

Aus den folgenden Gründen wird möglicherweise ein 403-Fehler angezeigt:

- CloudFront hat keinen Zugriff auf den Amazon S3 S3-Bucket. Dies kann passieren, wenn Origin Access Identity (OAI) oder Origin Access Control (OAC) für Ihre Distribution nicht aktiviert sind und der Bucket privat ist.
- Der angegebene Pfad in der angeforderten URL ist nicht korrekt.
- Das angeforderte Objekt existiert nicht.
- Der Host-Header wurde mit dem REST-API-Endpunkt weitergeleitet. Weitere Informationen finden Sie unter HTTP-Host-Header-Bucket-Spezifikation im Amazon Simple Storage Service-Benutzerhandbuch.
- Sie haben benutzerdefinierte Fehlerseiten konfiguriert. Weitere Informationen finden Sie unter Wie CloudFront werden Fehler verarbeitet, wenn Sie benutzerdefinierte Fehlerseiten konfiguriert haben.

Geografische Einschränkungen geben einen 403-Fehler zurück

Wenn Sie geografische Einschränkungen (auch als Geoblocking bezeichnet) aktiviert haben, um zu verhindern, dass Benutzer an bestimmten geografischen Standorten auf Inhalte zugreifen können, die Sie über eine CloudFront Distribution verteilen, erhalten blockierte Benutzer die Fehlermeldung 403.

Weitere Informationen finden Sie unter Beschränken Sie die geografische Verteilung Ihrer Inhalte.

Die Konfiguration einer signierten URL oder eines signierten Cookies gibt einen 403-Fehler zurück

Wenn Sie für die Verhaltenskonfiguration Ihrer Distribution die Option Zuschauerzugriff einschränken aktiviert haben, URLs führen Anfragen, die keine signierten oder signierten Cookies verwenden, zu einem 403-Fehler. Weitere Informationen finden Sie unter den folgenden Themen:

- Stellen Sie private Inhalte mit signierten URLs und signierten Cookies bereit
- Wie behebe ich Probleme im Zusammenhang mit einer signierten URL oder signierten Cookies CloudFront?

Gestapelte Verteilungen verursachen einen 403-Fehler

Wenn Sie zwei oder mehr Verteilungen innerhalb einer Kette von Anfragen an den ursprünglichen Endpunkt haben, wird ein 403-Fehler CloudFront zurückgegeben. Es wird nicht empfohlen, eine Distribution vor einer anderen zu platzieren.

HTTP 404-Statuscode (nicht gefunden)

CloudFront gibt einen 404-Fehler (Nicht gefunden) zurück, wenn der Client versucht, auf eine Ressource zuzugreifen, die nicht existiert. Wenn Sie diesen Fehler bei Ihrer CloudFront Distribution erhalten, kann dies unter anderem folgende Ursachen haben:

- Die Ressource ist nicht vorhanden.
- Die URL ist falsch.
- Benutzerdefinierter Ursprung gibt 404 zurück.
- Benutzerdefinierte Fehlerseiten, die einen 404-Fehler zurückgeben. (Jeder Fehlercode kann in 404 übersetzt werden.) Weitere Informationen finden Sie unter Wie CloudFront werden Fehler verarbeitet, wenn Sie benutzerdefinierte Fehlerseiten konfiguriert haben.
- Die benutzerdefinierte Fehlerseite wurde versehentlich gelöscht, was zu einem 404-Fehler führte, da die Anfrage nach der gelöschten benutzerdefinierten Fehlerseite sucht. Weitere Informationen finden Sie unter Wie CloudFront werden Fehler verarbeitet, wenn Sie keine benutzerdefinierten Fehlerseiten konfiguriert haben.
- Falscher Herkunftspfad. Wenn der Quellpfad aufgefüllt ist, wird sein Wert an den Pfad jeder Anfrage vom Browser angehängt, bevor die Anfrage an den Ursprung weitergeleitet wird. Weitere Informationen finden Sie unter Ursprungspfad.

HTTP 412-Statuscode (Vorbedingung fehlgeschlagen)

CloudFront gibt den Fehlercode 412 (Precondition Failed) zurück, wenn der Zugriff auf die Zielressource verweigert wurde. In einigen Fällen ist ein Server so konfiguriert, dass er Anfragen erst akzeptiert, wenn bestimmte Bedingungen erfüllt sind. Wenn eine der angegebenen Bedingungen

nicht erfüllt ist, erlaubt der Server dem Client nicht, auf die angegebene Ressource zuzugreifen. Stattdessen antwortet der Server mit einem 412-Fehlercode.

Zu den häufigsten Ursachen für einen 412-Fehler CloudFront gehören:

- Bedingte Anfragen für andere Methoden als `GET` oder `HEAD` wenn die durch die `If-None-Match` Header `If-Unmodified-Since` oder definierte Bedingung nicht erfüllt ist. In diesem Fall kann die Anfrage, normalerweise ein Upload oder eine Änderung einer Ressource, nicht gestellt werden.
- Eine Bedingung in einem oder mehreren Anforderungsfeldern des CloudFront UpdateDistributionAPI-Vorgangs wird als falsch bewertet.

HTTP 500-Statuscode (Interner Serverfehler)

Ein HTTP 500-Statuscode (Interner Serverfehler) weist darauf hin, dass auf dem Server ein unerwarteter Fehler aufgetreten ist, der ihn daran gehindert hat, die Anfrage zu bearbeiten. Im Folgenden sind einige der häufigsten Ursachen für 500-Fehler bei Amazon aufgeführt CloudFront.

Themen

- Der Origin-Server gibt den Fehler 500 zurück an CloudFront

Der Origin-Server gibt den Fehler 500 zurück an CloudFront

Ihr Ursprungsserver gibt möglicherweise einen Fehler 500 an zurück CloudFront. Weitere Informationen finden Sie in den folgenden Themen zur Problembehandlung:

- Wenn Amazon S3 einen 500-Fehler zurückgibt, finden Sie weitere Informationen unter Wie behebe ich einen HTTP 500- oder 503-Fehler von Amazon S3?
- Wenn API Gateway einen 500-Fehler zurückgibt, finden Sie weitere Informationen unter Wie behebe ich 5xx-Fehler für die API Gateway Gateway-REST-API? .
- Wenn Elastic Load Balancing einen 500-Fehler zurückgibt, finden Sie weitere Informationen unter HTTP 500: Interner Serverfehler im Benutzerhandbuch für Application Load Balancers.

Wenn die obige Liste den Fehler 500 nicht behebt, liegt das Problem möglicherweise daran, dass ein CloudFront Point of Presence einen internen Serverfehler zurückgibt. Sie können sich an uns wenden, Supportum Unterstützung zu erhalten.

HTTP 502-Statuscode (Bad Gateway)

CloudFront gibt einen HTTP 502-Statuscode (Bad Gateway) zurück, wenn das angeforderte Objekt CloudFront nicht bedient werden konnte, weil es keine Verbindung zum Ursprungsserver herstellen konnte.

Wenn Sie Lambda @Edge verwenden, ist das Problem möglicherweise ein Lambda-Validierungsfehler. Wenn Sie einen HTTP 502-Fehler mit dem `NonS3OriginDnsError` Fehlercode erhalten, liegt wahrscheinlich ein DNS-Konfigurationsproblem vor, das die Verbindung zum Ursprung CloudFront verhindert.

Themen

Fehler bei der SSL/TLS-Aushandlung zwischen CloudFront und einem benutzerdefinierten Ursprungsserver

Wenn Sie einen benutzerdefinierten Ursprung verwenden, der HTTPS zwischen CloudFront und Ihrem Ursprung erfordert, können nicht übereinstimmende Domainnamen zu Fehlern führen. Das SSL/TLS-Zertifikat für Ihren Ursprung muss einen Domainnamen enthalten, der entweder der Ursprungsdomäne entspricht, die Sie für die CloudFront Verteilung angegeben haben, oder dem `Host` Header der Ursprungsanfrage.

Wenn die Domainnamen nicht übereinstimmen, schlägt der SSL/TLS-Handshake fehl und CloudFront gibt den HTTP-Statuscode 502 (Bad Gateway) zurück und setzt den Header auf. `X-Cache Error from cloudfront`

Um festzustellen, ob die Domainnamen im Zertifikat mit der Ursprungsdomain in der Distribution oder im `Host` Header übereinstimmen, können Sie einen Online-SSL-Checker oder OpenSSL verwenden. Wenn die Domain-Namen nicht übereinstimmen, haben Sie zwei Optionen:

- Verwenden Sie ein neues SSL-/TLS-Zertifikat mit den entsprechenden Domain-Namen.

  Wenn Sie AWS Certificate Manager (ACM) verwenden, finden Sie Informationen zum Anfordern eines neuen Zertifikats im AWS Certificate Manager Benutzerhandbuch unter Anfordern eines öffentlichen Zertifikats.
- Ändern Sie die Verteilungskonfiguration, sodass nicht CloudFront mehr versucht wird, SSL für die Verbindung mit Ihrem Ursprung zu verwenden.

Online-SSL-Prüfung

Sie finden ein Test-Tool für SSL-Verbindungen, indem Sie im Internet nach den Begriffen "online ssl checker" suchen. In der Regel müssen Sie den Namen Ihrer Domain eingeben, und das Tool gibt eine Vielzahl von Informationen zu Ihrem SSL-/TLS-Zertifikat aus. Vergewissern Sie sich, dass das Zertifikat Ihren Domänennamen aus den Feldern Allgemeiner Name oder Alternative Antragstellernamen enthält.

OpenSSL

Um bei der Behebung von HTTP 502-Fehlern von zu helfen CloudFront, können Sie mit OpenSSL versuchen, eine SSL/TLS connection to your origin server. If OpenSSL is not able to make a connection, that can indicate a problem with your origin server's SSL/TLS Konfiguration vorzunehmen. Wenn OpenSSL eine Verbindung herstellen kann, gibt es Informationen über das Zertifikat des Ursprungsservers zurück, einschließlich des allgemeinen Namens (Feld `Subject CN`) des Zertifikats und des alternativen Antragstellernamens (Feld `Subject Alternative Name`).

Verwenden Sie den folgenden OpenSSL-Befehl, um die Verbindung zu Ihrem Ursprungsserver zu testen (*`origin domain`*ersetzen Sie ihn durch den Domainnamen Ihres Ursprungsservers, z. B. example.com):

```
openssl s_client -connect origin domain name:443
```

Wenn Folgendes zutrifft:

- Ihr Ursprungsserver unterstützt mehrere Domänennamen mit mehreren SSL/TLS-Zertifikaten.
- Ihre Verteilung ist so konfiguriert, dass der `Host`-Header an den Ursprung weitergeleitet wird.

Fügen Sie dann die `-servername` Option wie im folgenden Beispiel zum OpenSSL-Befehl hinzu (*CNAME*ersetzen Sie sie durch den CNAME, der in Ihrer Distribution konfiguriert ist):

```
openssl s_client -connect origin domain name:443 -servername CNAME
```

Ursprungsserver ist über die unterstützten Verschlüsselungsverfahren/Protokolle nicht erreichbar

CloudFront stellt mithilfe von Chiffren und Protokollen eine Verbindung zu Originalservern her. Eine Liste der Verschlüsselungen und Protokolle, die unterstützt werden, CloudFront finden Sie unter. the section called “Unterstützte Protokolle und Chiffren zwischen und dem Ursprung CloudFront ” Wenn Ihr Absender nicht mit einer dieser Chiffren oder Protokolle im SSL/TLS-Austausch antwortet, schlägt die Verbindung fehl. CloudFront Sie können überprüfen, ob Ihr Ursprungsserver diese Verschlüsselungsverfahren und Protokolle unterstützt, indem Sie ein Online-Tool wie SSL-Labs verwenden. Geben Sie den Domain-Namen Ihres Ursprungs-Servers in das Feld Hostname ein und wählen Sie anschließend OK aus. Überprüfen Sie die Felder Common names (Allgemeine Namen) und Alternative names (Alternative Namen) in dem Test; entspricht ihr Inhalt dem Domain-Namen Ihres Ursprungs-Servers? Nachdem der Test beendet ist, suchen Sie die Abschnitte Protocols und Cipher Suites in den Testergebnissen; welche Verschlüsselungsverfahren oder Protokolle werden von Ihrem Ursprungs-Server unterstützt? Vergleichen Sie die Produkte mit der Liste von the section called “Unterstützte Protokolle und Chiffren zwischen und dem Ursprung CloudFront ”.

SSL-/TLS-Zertifikat auf dem Ursprungsserver ist abgelaufen, ungültig oder selbstsigniert oder die Zertifikatkette weist die falsche Reihenfolge auf

Wenn der Ursprungsserver Folgendes zurückgibt, CloudFront bricht er die TCP-Verbindung ab, gibt den HTTP-Statuscode 502 (Bad Gateway) zurück und setzt den Header auf: `X-Cache Error from cloudfront`

- Abgelaufenes Zertifikat
- Ungültiges Zertifikat
- Selbstsigniertes Zertifikat
- Zertifikatkette in der falschen Reihenfolge

> **Note**
>
> Wenn die gesamte Zertifikatskette, einschließlich des Zwischenzertifikats, nicht vorhanden ist, wird CloudFront die TCP-Verbindung unterbrochen.

Informationen zum Installieren eines SSL-/TLS-Zertifikat auf Ihrem benutzerdefinierten Ursprungsserver finden Sie unter the section called “Erfordern Sie HTTPS für einen benutzerdefinierten Ursprung”.

Ursprungsserver ist über die eingestellten Ports in den Ursprungseinstellungen nicht erreichbar

Wenn Sie in Ihrer CloudFront Distribution einen Ursprung erstellen, können Sie die Ports festlegen, über die eine CloudFront Verbindung zum Ursprung für HTTP- und HTTPS-Verkehr hergestellt wird. Standardmäßig sind das die TCP-Ports 80 und 443. Sie haben die Möglichkeit, diese Ports zu ändern. Wenn Ihr Absender den Datenverkehr auf diesen Ports aus irgendeinem Grund ablehnt oder wenn Ihr Backend-Server nicht auf die Ports reagiert, kann keine Verbindung CloudFront hergestellt werden.

Sie können diese Probleme zu beheben, indem Sie alle Firewalls überprüfen, die in Ihrer Infrastruktur ausgeführt werden, und sich vergewissern, dass sie die unterstützten IP-Bereiche nicht blockieren. Weitere Informationen finden Sie unter AWS IP-Adressbereiche im Amazon VPC-Benutzerhandbuch. Überprüfen Sie außerdem, ob Ihr Webserver auf dem Ursprungsserver ausgeführt wird.

Fehler bei der Lambda-Validierung

Wenn Sie Lambda@Edge verwenden, kann der HTTP-Statuscode 502 anzeigen, dass Ihre Lambda-Funktionsantwort falsch gebildet wurde oder ungültigen Inhalt enthielt. Weitere Informationen zur Behebung von Lambda@Edge-Fehlern finden Sie unter Testen und Debuggen von Lambda @Edge - Funktionen.

CloudFront Fehler bei der Funktionsvalidierung

Wenn Sie CloudFront Funktionen verwenden, kann ein HTTP-502-Statuscode darauf hinweisen, dass die CloudFront Funktion versucht, einen schreibgeschützten Header hinzuzufügen, zu löschen oder zu ändern. Dieser Fehler tritt beim Testen nicht auf, sondern tritt auf, nachdem Sie die Funktion bereitgestellt und die Anforderung ausgeführt haben. Um diesen Fehler zu beheben, überprüfen

und aktualisieren Sie Ihre CloudFront Funktion. Weitere Informationen finden Sie unter Funktionen aktualisieren.

DNS-Fehler (`NonS3OriginDnsError`)

Ein HTTP 502-Fehler mit dem `NonS3OriginDnsError` Fehlercode weist auf ein DNS-Konfigurationsproblem hin, das CloudFront verhindert, dass eine Verbindung zum Ursprung hergestellt werden kann. Wenn Sie diesen Fehler von erhalten CloudFront, stellen Sie sicher, dass die DNS-Konfiguration des Ursprungs korrekt ist und funktioniert.

Wenn es eine Anfrage für ein Objekt CloudFront erhält, das abgelaufen ist oder sich nicht in seinem Cache befindet, sendet es eine Anfrage an den Ursprung, um das Objekt abzurufen. Um eine erfolgreiche Anfrage an den Ursprung zu stellen, CloudFront führt eine DNS-Auflösung in der Ursprungsdomäne durch. Wenn beim DNS-Dienst für Ihre Domain Probleme auftreten, CloudFront kann der Domainname nicht aufgelöst werden, um die IP-Adresse abzurufen, was zu einem HTTP 502-Fehler (`NonS3OriginDnsError`) führt. Sie können dieses Problem beheben, indem Sie sich an Ihren DNS-Anbieter wenden. Wenn Sie Amazon Route 53 verwenden, finden Sie weitere Informationen unter Warum kann ich nicht auf meine Website zugreifen, die Route-53-DNS-Services verwendet?

Sie können dieses Problem weiterhin beheben, indem Sie sicherstellen, dass die autoritativen Name-Server für die Stamm-Domäne bzw. den Zone Apex (z. B. `example.com`) Ihres Ursprungs-Servers richtig funktionieren. Sie können die folgenden Befehle verwenden, um die Nameserver für Ihren Ursprungsserver-Apex zu finden, zum Beispiel mit einem Tool wie dig oder nslookup:

```
dig OriginAPEXDomainName NS +short
```

```
nslookup -query=NS OriginAPEXDomainName
```

Wenn Sie die Namen Ihrer Name-Server erhalten haben, verwenden Sie die folgenden Befehle, um den Domain-Namen Ihres Ursprungsservers von ihnen abzufragen; so können Sie sicherstellen, dass alle eine Antwort zurückgeben:

```
dig OriginDomainName @NameServer
```

```
nslookup OriginDomainName NameServer
```

> **Important**
>
> Stellen Sie sicher, dass Sie diese DNS-Fehlerbehebung auf einem Computer durchführen, der mit dem öffentlichen Internet verbunden ist. CloudFront löst die Ursprungsdomäne mithilfe von öffentlichem DNS im Internet auf. Daher ist es wichtig, die Fehlerbehebung in einem ähnlichen Kontext durchzuführen.

Wenn der Ursprung eine Subdomäne ist, deren DNS-Berechtigung an einen anderen Nameserver als die Stammdomäne delegiert ist, stellen Sie sicher, dass die Datensätze für den Nameserver (NS) und Autoritätsursprung (SOA) für die Subdomäne korrekt konfiguriert sind. Sie können mit Befehlen, die den vorherigen Beispielen ähneln, nach diesen Datensätzen suchen.

Weitere Informationen zu DNS finden Sie unter DNS-Konzepte (Domain Name System) in der Dokumentation zu Amazon Route 53.

Origin 502 Application Load Balancer Balancer-Fehlers

Wenn Sie Application Load Balancer als Quelle verwenden und einen 502-Fehler erhalten, finden Sie weitere Informationen unter Wie behebe ich Application Load Balancer Balancer-HTTP 502-Fehler? .

API-Gateway-Origin-502-Fehler

Wenn Sie API Gateway verwenden und einen 502-Fehler erhalten, finden Sie weitere Informationen unter Wie behebe ich HTTP 502-Fehler von API Gateway REST APIs mit Lambda-Proxyintegration? .

HTTP 503-Statuscode (Service nicht verfügbar)

Ein HTTP-Statuscode 503 (Service nicht verfügbar) zeigt in der Regel an, dass es auf dem Ursprungsserver ein Leistungsproblem gibt. In seltenen Fällen weist dies darauf hin, dass eine Anfrage aufgrund von Ressourcenbeschränkungen an einem Edge-Standort CloudFront vorübergehend nicht bearbeitet werden kann.

Wenn Sie Lambda @Edge oder CloudFront Functions verwenden, ist das Problem möglicherweise ein Ausführungsfehler oder ein Fehler, dass das Lambda @Edge -Limit überschritten wurde.

Themen

- Ursprungsserver verfügt nicht über ausreichend Kapazitäten für die vorliegende Anfragerate

- CloudFront hat den Fehler aufgrund von Ressourcenbeschränkungen am Edge-Standort verursacht
- Lambda @Edge oder Fehler bei der Ausführung der CloudFront Funktion
- Lambda @Edge -Limit überschritten

Ursprungsserver verfügt nicht über ausreichend Kapazitäten für die vorliegende Anfragerate

Wenn ein Ursprungsserver nicht verfügbar ist oder eingehende Anfragen nicht bearbeiten kann, gibt er den HTTP-Statuscode 503 (Service Unavailable) zurück. CloudFront leitet den Fehler dann an den Benutzer zurück. Sie lösen dieses Problem, indem Sie die folgenden Schritte ausführen:

- Wenn Sie Amazon S3 als Ihren Ursprungsserver verwenden:
  - Sie können 3.500 PUT/COPY/POST/DELETE or 5,500 GET/HEAD Anfragen pro Sekunde pro partitioniertem Amazon S3 S3-Präfix senden. Wenn Amazon S3 eine 503 Slow-Down-Antwort zurückgibt, deutet dies in der Regel auf eine zu hohe Anforderungsrate für ein bestimmtes Amazon S3 S3-Präfix hin.

    Da die Anforderungsraten pro Präfix in einem S3-Bucket gelten, sollten Objekte auf mehrere Präfixe verteilt werden. Da die Anforderungsrate für die Präfixe allmählich zunimmt, skaliert Amazon S3 so, dass Anfragen für jedes der Präfixe separat bearbeitet werden. Infolgedessen ist die Gesamtanforderungsrate, die der Bucket verarbeitet, ein Vielfaches der Anzahl der Präfixe.
  - Weitere Informationen zu Amazon-S3-Berechtigungen finden Sie unter Optimieren der Amazon-S3-Leistung im Benutzerhandbuch zu Amazon Simple Storage Service.
- Wenn Sie Elastic Load Balancing als Ihren Ursprungsserver verwenden:
  - Stellen Sie sicher, dass Ihre Backend-Instances auf Zustandsprüfungen reagieren können.
  - Stellen Sie sicher, dass Ihr Load Balancer und Ihre Backend-Instances die Last bewältigen können.

  Weitere Informationen finden Sie unter:
  - Wie behebe ich 503-Fehler, die bei der Verwendung von Classic Load Balancer zurückgegeben wurden?
  - Wie behebe ich 503-Fehler (Service nicht verfügbar) von meinem Application Load Balancer aus?
- Wenn Sie einen benutzerdefinierten Ursprung verwenden:

  - Untersuchen Sie die Anwendungsprotokolle, um sicherzustellen, dass Ihr Origin über ausreichende Ressourcen wie Arbeitsspeicher, CPU und Festplattengröße verfügt.
  - Wenn Sie Amazon EC2 als Backend verwenden, stellen Sie sicher, dass der Instance-Typ über die entsprechenden Ressourcen verfügt, um die eingehenden Anfragen zu bearbeiten. Weitere Informationen finden Sie unter Instance-Typen im EC2 Amazon-Benutzerhandbuch.
- Wenn Sie API Gateway verwenden:
  - Dieser Fehler steht im Zusammenhang mit der Backend-Integration, wenn die API-Gateway-API keine Antwort empfangen kann. Der Backend-Server könnte sein:
    - Die Kapazität ist überlastet und neue Client-Anfragen können nicht verarbeitet werden.
    - Wird vorübergehend gewartet.
  - Um diesen Fehler zu beheben, schauen Sie sich die Protokolle Ihrer API Gateway Gateway-Anwendung an, um festzustellen, ob ein Problem mit der Backend-Kapazität, der Integration oder etwas anderem vorliegt.

CloudFront hat den Fehler aufgrund von Ressourcenbeschränkungen am Edge-Standort verursacht

Dieser Fehler tritt in dem seltenen Fall auf, dass CloudFront Anfragen nicht an den nächstbesten verfügbaren Edge-Standort weitergeleitet werden können und somit eine Anfrage nicht erfüllt werden kann. Dieser Fehler tritt häufig auf, wenn Sie in Ihrer CloudFront-Verteilung Belastungstests durchführen. Um dies zu vermeiden, befolgen Sie die the section called “Belastungstests CloudFront”-Richtlinien zum Vermeiden des Fehlers 503 (Kapazität überschritten).

Wenn dies in Ihrer Produktionsumgebung passiert, wenden Sie sich an Support.

Lambda @Edge oder Fehler bei der Ausführung der CloudFront Funktion

Wenn Sie Lambda @Edge oder CloudFront Functions verwenden, kann ein HTTP-Statuscode 503 darauf hinweisen, dass Ihre Funktion einen Ausführungsfehler zurückgegeben hat.

Weitere Informationen zur Identifizierung und Behebung von Lambda @Edge -Fehlern finden Sie unterTesten und Debuggen von Lambda @Edge -Funktionen.

Weitere Hinweise zum Testen von CloudFront Funktionen finden Sie unterFunktionen testen.

Lambda @Edge -Limit überschritten

Wenn Sie Lambda @Edge verwenden, kann ein HTTP-Statuscode 503 darauf hinweisen, dass Lambda einen Fehler zurückgegeben hat. Der Fehler kann durch eine der folgenden Ursachen bedingt sein.

- Die Anzahl der Funktionsausführungen hat eines der Quoten überschritten, die Lambda festlegt, um Ausführungen in einem zu drosseln AWS-Region (gleichzeitige Ausführungen oder Aufrufthäufigkeit).
- Die Funktion hat das Timeout-Kontingent für die Lambda-Funktion überschritten.

Weitere Informationen zu den Lambda @Edge -Kontingenten finden Sie unterKontingente für Lambda@Edge. Weitere Informationen zur Identifizierung und Behebung von Lambda @Edge -Fehlern finden Sie unterthe section called “Testen und debuggen”. Die Lambda-Servicekontingente finden Sie auch im AWS Lambda Developer Guide.

HTTP 504-Statuscode (Gateway Timeout)

Ein HTTP 504-Statuscode (Gateway-Timeout) gibt an, dass bei der CloudFront Weiterleitung einer Anfrage an den Ursprung (weil sich das angeforderte Objekt nicht im Edge-Cache befand) einer der folgenden Fälle eingetreten ist:

- Der Ursprung hat einen HTTP 504-Statuscode an CloudFront zurückgegeben.
- Der Ursprung reagierte nicht, bevor die Anforderung ablief.

CloudFront gibt einen HTTP 504-Statuscode zurück, wenn der Datenverkehr zum Ursprung durch eine Firewall oder Sicherheitsgruppe blockiert wird oder wenn der Ursprung im Internet nicht zugänglich ist. Überprüfen Sie diese Punkte zuerst. Wenn der Zugriff nicht das Problem ist, sehen Sie sich Anwendungsverzögerungen und Server-Timeouts an, um die Probleme zu ermitteln und zu beheben.

Themen

- Konfigurieren Sie die Firewall auf Ihrem Ursprungsserver so, dass CloudFront Datenverkehr zugelassen wird
- Konfigurieren Sie die Sicherheitsgruppen auf Ihrem Ursprungsserver, um Datenverkehr zuzulassen CloudFront
- Erlauben des Zugriffs auf Ihren benutzerdefinierten Ursprungsserver über das Internet

- Suchen und Beheben verzögerter Antworten von Anwendungen auf Ihrem Ursprungsserver

Konfigurieren Sie die Firewall auf Ihrem Ursprungsserver so, dass CloudFront Datenverkehr zugelassen wird

Wenn die Firewall auf Ihrem Ursprungsserver den CloudFront Datenverkehr blockiert, wird ein HTTP-504-Statuscode CloudFront zurückgegeben. Stellen Sie daher sicher, dass dies nicht das Problem ist, bevor Sie nach anderen Problemen suchen.

Die Methode, die Sie nutzen, um zu bestimmen, ob es sich um ein Problem mit Ihrer Firewall handelt, hängt davon ab, welches System Ihr Ursprungs-Server verwendet:

- Wenn Sie eine IPTable Firewall auf einem Linux-Server verwenden, können Sie nach Tools und Informationen suchen, die Ihnen bei der Arbeit helfen IPTables.
- Wenn Sie die Windows-Firewall auf einem Windows-Server verwenden, finden Sie weitere Informationen unter Hinzufügen oder Bearbeiten einer Firewallregel in der Microsoft-Dokumentation.

Wenn Sie die Firewall-Konfiguration auf Ihrem Ursprungsserver auswerten, sollten Sie auf der Grundlage des veröffentlichten IP-Adressbereichs nach Firewalls oder Sicherheitsregeln Ausschau halten, die den Datenverkehr von CloudFront Edge-Standorten blockieren. Weitere Informationen finden Sie unter Standorte und IP-Adressbereiche von CloudFront Edge-Servern.

Wenn der CloudFront IP-Adressbereich eine Verbindung zu Ihrem Ursprungsserver herstellen darf, stellen Sie sicher, dass Sie die Sicherheitsregeln Ihres Servers aktualisieren, um die Änderungen zu berücksichtigen. Sie können ein Amazon-SNS-Thema abonnieren und Benachrichtigungen erhalten, wenn die IP-Adressbereichsdatei aktualisiert wird. Nachdem Sie die Benachrichtigung erhalten haben, können Sie Code verwenden, um die Datei abzurufen, sie zu analysieren und ggf. Anpassungen Ihrer lokalen Umgebung vorzunehmen. Weitere Informationen finden Sie im AWS News-Blog unter AWS Öffentliche IP-Adressänderungen über Amazon SNS abonnieren.

Konfigurieren Sie die Sicherheitsgruppen auf Ihrem Ursprungsserver, um Datenverkehr zuzulassen CloudFront

Wenn Ihr Origin Elastic Load Balancing verwendet, überprüfen Sie die ELB-Sicherheitsgruppen und stellen Sie sicher, dass die Sicherheitsgruppen eingehenden CloudFront Datenverkehr zulassen.

Sie können es auch verwenden AWS Lambda , um Ihre Sicherheitsgruppen automatisch zu aktualisieren, um eingehenden Datenverkehr von zuzulassen. CloudFront

Erlauben des Zugriffs auf Ihren benutzerdefinierten Ursprungsserver über das Internet

Wenn Sie nicht auf Ihren benutzerdefinierten Ursprungsserver zugreifen CloudFront können, weil er nicht öffentlich im Internet verfügbar ist, wird ein HTTP 504-Fehler CloudFront zurückgegeben.

CloudFront Edge-Standorte stellen über das Internet eine Verbindung zu den Ursprungsservern her. Wenn sich Ihr benutzerdefinierter Ursprung in einem privaten Netzwerk befindet, CloudFront kann er nicht erreicht werden. Aus diesem Grund können Sie private Server, einschließlich interner Classic Load Balancer, nicht als Ursprungsserver mit CloudFront verwenden.

Um zu überprüfen, ob der Internetverkehr eine Verbindung zu Ihrem Ursprungsserver herstellen kann, führen Sie die folgenden Befehle aus (wo *OriginDomainName* ist der Domainname für Ihren Server):

Für HTTPS-Datenverkehr:

- nc -zv 443 *OriginDomainName*
- Telnet 443 *OriginDomainName*

Für HTTP-Datenverkehr:

- NC-ZV 80 *OriginDomainName*
- Telnet 80 *OriginDomainName*

Suchen und Beheben verzögerter Antworten von Anwendungen auf Ihrem Ursprungsserver

Server-Timeouts sind häufig das Ergebnis einer Anwendung, die recht lange braucht, um zu reagieren, oder eines Timeout-Werts, der zu niedrig eingestellt ist.

Eine schnelle Abhilfe zum Vermeiden des HTTP-Fehlers 504 besteht darin, einen höheren CloudFront-Timeout-Wert für Ihre Verteilung festzulegen. Wir empfehlen jedoch, dass Sie zunächst sicherstellen, dass Sie alle Leistungs- und Latenzprobleme mit der Anwendung und dem Ursprungs-Server beheben. Anschließend können Sie einen angemessenen Timeout-Wert festlegen, der zur Vermeidung des HTTP-Fehlers 504 beiträgt und eine gute Reaktionsfähigkeit für Benutzer bietet.

Im Folgenden finden Sie eine Übersicht über die Schritte, die Sie ausführen können, um Leistungsprobleme zu ermitteln und zu beheben:

1. Messen Sie die normale Latenz und die Latenz bei hoher Last (Reaktionsfähigkeit) Ihrer Webanwendung.
2. Fügen Sie weitere Ressourcen, z. B. CPU oder Speicher (bei Bedarf) hinzu. Führen Sie andere Schritte aus, um die Probleme zu beheben, z. B. Optimierung von Datenbankabfragen für Szenarien mit hoher Last.
3. Passen Sie bei Bedarf den Timeout-Wert für Ihre CloudFront Distribution an.

Im Folgenden finden Sie Details zu jedem einzelnen Schritt.

Messen der normalen Latenz und der Latenz bei hoher Last

Um festzustellen, ob auf einem oder mehreren Backend-Webanwendungsservern eine hohe Latenz vorliegt, führen Sie den folgenden Linux curl-Befehl auf jedem Server aus:

```
curl -w "DNS Lookup Time: %{time_namelookup} \nConnect time: %{time_connect}
 \nTLS Setup: %{time_appconnect} \nRedirect Time: %{time_redirect} \nTime to first
 byte: %{time_starttransfer} \nTotal time: %{time_total} \n" -o /dev/null https://
www.example.com/yourobject
```

> **Note**
>
> Wenn Sie Windows auf Ihren Servern ausführen, können Sie curl für Windows suchen und herunterladen, um einen entsprechenden Befehl auszuführen.

Beachten Sie beim Messen und Auswerten der Latenz einer Anwendung, die auf Ihrem Server ausgeführt wird, folgende Punkte:

- Latenzwerte sind für jede Anwendung relativ. Allerdings ist eine Zeit bis zum ersten Byte in Millisekunden statt Sekunden oder mehr sinnvoll.
- Wenn Sie die Anwendungslatenz unter normalen Lastbedingungen messen und das Ergebnis in Ordnung ist, sollten Sie daran denken, dass bei Viewern dennoch Timeouts bei hoher Last auftreten können. Wenn eine hohe Nachfrage besteht, können Server Antworten verzögert senden oder gar nicht reagieren. Überprüfen Sie zur Vermeidung von Latenzproblemen bei hoher

Last Ihre Serverressourcen wie CPU, Arbeitsspeicher sowie Lese- und Schreibvorgänge auf Speichermedien, um sicherzustellen, dass Ihre Server über die Kapazitäten zur Skalierung für hohe Auslastung verfügen.

Sie können den folgenden Linux-Befehl zum Überprüfen des Speichers ausführen, der von Apache-Prozessen verwendet wird:

```
watch -n 1 "echo -n 'Apache Processes: ' && ps -C apache2 --no-headers | wc -l && free -m"
```

- Eine hohe CPU-Auslastung auf dem Server kann die Leistung einer Anwendung erheblich reduzieren. Wenn Sie eine EC2 Amazon-Instance für Ihren Backend-Server verwenden, überprüfen Sie die CloudWatch Metriken für den Server, um die CPU-Auslastung zu überprüfen. Weitere Informationen finden Sie im CloudWatch Amazon-Benutzerhandbuch. Oder wenn Sie Ihren eigenen Server verwenden, finden Sie in der Server-Hilfedokumentation Anweisungen zur Überprüfung der CPU-Auslastung.
- Überprüfen Sie, ob andere potenzielle Probleme unter hoher Last vorliegen, wie beispielsweise Datenbankabfragen, die bei einem hohen Volume von Anfragen langsam ausgeführt werden.

Hinzufügen von Ressourcen und Optimieren von Servern und Datenbanken

Nachdem Sie die Reaktionsfähigkeit Ihrer Anwendungen und Server bewertet haben, stellen Sie sicher, dass Sie über ausreichend Ressourcen für normalen Datenverkehr und Situationen mit hoher Auslastung verfügen:

- Wenn Sie einen eigenen Server nutzen, stellen Sie basierend auf Ihrer Bewertung sicher, dass dieser über ausreichend CPU, Arbeitsspeicher und Festplattenspeicher für die Verarbeitung von Viewer-Anfragen verfügt.
- Wenn Sie eine EC2 Amazon-Instance als Backend-Server verwenden, stellen Sie sicher, dass der Instance-Typ über die entsprechenden Ressourcen verfügt, um eingehende Anfragen zu bearbeiten. Weitere Informationen finden Sie unter Instance-Typen im EC2 Amazon-Benutzerhandbuch.

Außerdem sollten Sie ggf. die folgenden Optimierungsschritte ausführen, um Timeouts zu vermeiden:

- Wenn der Wert für die Zeit bis zum ersten Byte, der vom curl-Befehl zurückgegeben wird, hoch erscheint, ergreifen Sie Maßnahmen zur Verbesserung der Leistung Ihrer Anwendung. Durch Verbessern der Anwendungsreaktionsfähigkeit lassen sich wiederum Timeout-Fehler reduzieren.

- Optimieren Sie Datenbankabfragen, um sicherzustellen, dass sie hohe Abfragezahlen verarbeiten können, ohne dass die Leistung beeinträchtigt wird.
- Richten Sie (persistente) Keepalive-Verbindungen auf Ihrem Backend-Server ein. Mit dieser Option lassen sich Latenzen vermeiden, die auftreten, wenn Verbindungen für nachfolgende Anfragen oder Benutzer erneut hergestellt werden müssen.
- Wenn Sie Elastic Load Balancing als Ausgangspunkt verwenden, sind die folgenden möglichen Ursachen für einen 504-Fehler:
  - Der Load Balancer kann vor Ablauf des Verbindungs-Timeouts (10 Sekunden) keine Verbindung zum Ziel herstellen.
  - Der Load Balancer stellt eine Verbindung zum Ziel her, aber das Ziel reagiert nicht, bevor das Leerlauf-Timeout abgelaufen ist.
  - Die Network Access Control List (ACL) für das Subnetz lässt keinen Datenverkehr von den Zielen zu den Load Balancer-Knoten an den kurzlebigen Ports (1024-65535) zu.
  - Das Ziel gibt einen Inhaltslängen-Header zurück, der größer ist als der Entitätskörper. Der Load Balancer hat beim Warten auf die fehlenden Bytes eine Zeitüberschreitung.
  - Das Ziel ist eine Lambda-Funktion und Lambda reagiert nicht, bevor das Verbindungstimeout abgelaufen ist.

  Weitere Informationen zur Reduzierung der Latenz finden Sie unter Wie behebe ich Probleme mit hoher Latenz auf meinem ELB Classic Load Balancer?
- Wenn Sie die Datei MediaTailor als Ursprung verwenden, sind die folgenden möglichen Ursachen für einen 504-Fehler:
  - Wenn Verwandte falsch behandelt URLs werden, MediaTailor kann dies URLs von den Spielern missbilligt werden.
  - Wenn dies der offensichtliche Ursprung für MediaPackage ist MediaTailor, können offensichtliche MediaPackage 404-Fehler MediaTailor dazu führen, dass ein 504-Fehler zurückgegeben wird.
  - Es dauert mehr als 2 Sekunden, bis die Anfrage an den MediaTailor Ursprungsserver abgeschlossen ist.
- Wenn Sie Amazon API Gateway als Quelle verwenden, ist Folgendes eine mögliche Ursache für einen 504-Fehler:
  - Eine Integrationsanfrage dauert länger als der maximale Integrationstimeout-Parameter Ihres API-Gateway-REST-API-Parameters. Weitere Informationen finden Sie unter Wie kann ich API-HTTP-504-Timeout-Fehler mit API Gateway beheben?

Passen Sie bei Bedarf den CloudFront Timeout-Wert an

Wenn Sie eine langsame Anwendungsleistung, Ursprungs-Serverkapazität und weitere Probleme festgestellt und behoben haben, die Viewer aber weiterhin den HTTP-Fehler 504 erhalten, sollten Sie in Betracht ziehen, die Zeit, die in Ihrer Verteilung als Ursprungs-Reaktions-Timeout angegeben ist, zu ändern. Weitere Informationen finden Sie unter the section called “Timeout bei der Antwort”.

Belastungstests CloudFront

Herkömmliche Lasttestmethoden funktionieren nicht gut, CloudFront da DNS CloudFront verwendet wird, um die Lasten auf geografisch verteilte Edge-Standorte und innerhalb jedes Edge-Standorts auszugleichen. Wenn ein Client Inhalte von anfordert CloudFront, erhält der Client eine DNS-Antwort, die eine Reihe von IP-Adressen enthält. Wenn Sie testen, indem Sie Anfragen nur an eine der IP-Adressen senden, die DNS zurückgibt, testen Sie nur eine kleine Teilmenge der Ressourcen an einem CloudFront Edge-Standort, was die tatsächlichen Verkehrsmuster nicht genau wiedergibt. Je nach Menge der angeforderten Daten können Tests auf diese Weise die Leistung dieser kleinen Teilmenge von Servern überlasten und beeinträchtigen. CloudFront

CloudFront ist so konzipiert, dass es für Zuschauer mit unterschiedlichen Client-IP-Adressen und unterschiedlichen DNS-Resolvern in mehreren geografischen Regionen skaliert werden kann. Um Lasttests durchzuführen, mit denen die CloudFront Leistung genau bewertet wird, empfehlen wir, dass Sie alle folgenden Schritte ausführen:

- Senden Sie Clientanfragen von mehreren geografischen Regionen aus ab.
- Konfigurieren Sie Ihren Test so, dass jeder Client eine unabhängige DNS-Anfrage stellt. Jeder Client erhält dann einen anderen Satz von IP-Adressen von DNS.
- Verteilen Sie für jeden Client, der Anfragen stellt, Ihre Client-Anfragen auf den Satz von IP-Adressen, die von DNS zurückgegeben werden. Dadurch wird sichergestellt, dass die Last auf mehrere Server an einem CloudFront Edge-Standort verteilt wird.

Hinweise

- Lasttests sind für Cache-Verhalten nicht zulässig, die über Lambda @Edge -Viewer-Anforderungs- oder Viewer-Antwort-Trigger verfügen.
- Auslastungstests sind auf Origins, für die Origin Shield aktiviert ist, nicht erlaubt.

Kontingente

Sie können eine Erhöhung des CloudFront Kontingents beantragen, indem Sie die folgenden Optionen verwenden:

- Sie können die Service Quotas Quotas-Konsole oder die verwenden AWS Command Line Interface. Weitere Informationen finden Sie unter den folgenden Themen:
  - Eine Erhöhung des Kontingents im Service Quotas Quota-Benutzerhandbuch beantragen
  - request-service-quota-increase in der AWS CLI Befehlsreferenz
- Wenn ein CloudFront Kontingent nicht unter Service Quotas verfügbar ist, verwenden Sie den, AWS Support Center Console um einen Fall zur Erhöhung des Servicekontingents zu erstellen.

CloudFront unterliegt den folgenden Kontingenten.

Themen

Allgemeine Kontingente

Entity	Standardkontingent
Datenübertragungsrate pro Verteilung	150 Gbit/s Höheres Kontingent anfordern
Anforderungen pro Sekunde und Verteilung	250 000 Höheres Kontingent anfordern
Tags, die einer Verteilung hinzugefügt werden können	50 Höheres Kontingent anfordern
Dateien, die Sie mit einer Verteilung bereitstellen können	Kein Kontingent
Maximale Länge einer Anfrage oder einer ursprünglichen Antwort, einschließlich Header und Abfragezeichenfolgen, jedoch ohne den Hauptinhalt	20,480 Bytes
Maximale Länge einer URL	8,192 Bytes
Maximale Anzahl von Konfigurationen für die Protokollzustellung in Echtzeit pro AWS-Konto	150

Allgemeine Kontingente für Verteilungen

Entity	Standardkontingent
Alternative Domainnamen (CNAMEs) pro Distribution Weitere Informationen finden Sie unter Verwenden Sie Benutzerdefiniert, URLs indem Sie alternative Domainnamen hinzufügen (CNAMEs).	100 Höheres Kontingent anfordern

Entity	Standardkontingent
Cache-Verhalten pro Verteilung	75 Höheres Kontingent anfordern
Verbindungsversuche pro Ursprung Weitere Informationen finden Sie unter Verbindungsversuche.	1-3
Verbindungs-Timeout pro Ursprung Weitere Informationen finden Sie unter Verbindungstimeout.	1-10 Sekunden
Reaktions-Timeout per Ursprung Dies wird auch als Origin Read Timeout oder Origin Request Timeout bezeichnet. Weitere Informationen finden Sie unter Timeout bei der Antwort.	1—120 Sekunden Höheres Kontingent anfordern
Keep-Alive-Timeout pro Ursprung Weitere Informationen finden Sie unter Keep-Alive-Timeout (nur benutzerdefinierte und VPC-Ursprünge).	1-120 Sekunden Höheres Kontingent anfordern
Verteilungen pro AWS-Konto Weitere Informationen finden Sie unter Eine Verteilung erstellen.	500 Höheres Kontingent anfordern
Zugriffskontrolle für Verteilungen pro Herkunft	100 Höheres Kontingent anfordern
Verteilungen innerhalb der Kette von Anfragen bis zum Ausgangsendpunkt Wir empfehlen nicht, eine Distribution vor einer anderen zu platzieren. Eine Überschreitung dieses Kontingents führt zu einem 403-Fehler.	2

Entity	Standardkontingent
Dateikomprimierung: Bereich von Dateigrößen, der CloudFront komprimiert wird Weitere Informationen finden Sie unter Komprimierte Dateien bereitstellen.	1 000 bis 10 000 000 Bytes
Maximale Dateigröße pro HTTP-GET-Antwort, die zwischengespeichert werden kann. Nur die Antworten für eine HTTP-GET-Anfrage werden zwischengespeichert. Antworten für POST oder PUT werden nicht zwischengespeichert.	50 GB
Origin-Zugriffskontrollen pro AWS-Konto	100 Höheres Kontingent anfordern
Zugriffsidentitäten von Herkunft pro AWS-Konto	100 Höheres Kontingent anfordern
Ursprünge pro Verteilung	100 Höheres Kontingent anfordern
Ursprungsgruppen pro Verteilung	10 Höheres Kontingent anfordern
Bereitstellen von Verteilungen pro AWS-Konto Weitere Informationen finden Sie unter the section called “Verwenden Sie Continuous Deployment, um Änderungen sicher zu testen”.	20 Höheres Kontingent anfordern
Distributionen, die demselben VPC-Ursprung zugeordnet sind	50

Entity	Standardkontingent
VPC-Ursprünge pro AWS-Konto	25 Höheres Kontingent anfordern
Maximale Anzahl von Distributionen, die einer einzelnen statischen Anycast-IP-Liste zugeordnet werden können.	100 Höheres Kontingent anfordern

Allgemeine Kontingente für Richtlinien

Entity	Standardkontingent
Benutzerdefinierte Cache-Richtlinien pro AWS-Konto (Gilt nicht für CloudFront verwaltete Cache-Richtlinien)	20 Höheres Kontingent anfordern
Verteilungen, die derselben Cache-Richtlinie zugeordnet sind	100
Abfragezeichenfolgen pro Cache-Richtlinie	10 Höheres Kontingent anfordern
Header pro Cache-Richtlinie	10 Höheres Kontingent anfordern
Cookies pro Cache-Richtlinie	10 Höheres Kontingent anfordern

Entity	Standardkontingent
Kombinierte Gesamtlänge aller Abfragezeichenfolgen, Header und Cookie-Namen in einer Cache-Richtlinie	1024
Benutzerdefinierte Richtlinien für ursprüngliche Anfragen pro AWS-Konto (Gilt nicht für Richtlinien für CloudFront verwaltete Anfragen mit Herkunft)	20 Höheres Kontingent anfordern
Verteilungen, die derselben Ursprungsanforderungsrichtlinie zugeordnet sind	100
Abfragezeichenfolgen pro Ursprungsanforderungsrichtlinie	10 Höheres Kontingent anfordern
Header pro Ursprungsanforderungsrichtlinie	10 Höheres Kontingent anfordern
Cookies pro Ursprungsanforderungsrichtlinie	10 Höheres Kontingent anfordern
Kombinierte Gesamtlänge aller Abfragezeichenfolgen, Header und Cookie-Namen in einer Ursprungsanfrage-Richtlinie	1024
Richtlinien für benutzerdefinierte Antwort-Header pro AWS-Konto (Gilt nicht für Richtlinien für CloudFront verwaltete Antwort-Header)	20 Höheres Kontingent anfordern
Verteilungen, die derselben Antwort-Header.Richtlinie zugeordnet sind	100 Höheres Kontingent anfordern

Entity	Standardkontingent
Benutzerdefinierte Header pro Antwort-Header-Richtlinie	10 Höheres Kontingent anfordern
Richtlinien für die kontinuierliche Bereitstellung pro AWS-Konto	20 Höheres Kontingent anfordern

Kontingente für CloudFront Funktionen

Entity	Standardkontingent
Funktionen pro AWS-Konto	100
Maximale Funktionsgröße Dieses Kontingent ist nicht anpassbar. Um zusätzliche Daten für Ihre CloudFront Funktionen zu speichern, erstellen Sie einen Schlüssel wertspeicher und fügen Sie Ihre Schlüssel-Wert-Paare hinzu. Weitere Informationen finden Sie unter Amazon CloudFront KeyValueStore.	10 KB
Maximaler Funktionsspeicher	2 MB
Verteilungen, die derselben Funktion zugeordnet sind	100

Zusätzlich zu diesen Kontingenten gibt es einige weitere Einschränkungen bei der Verwendung von CloudFront Functions. Weitere Informationen finden Sie unter Einschränkungen von Funktionen CloudFront .

Kontingente für Schlüsselwertspeicher

Entity	Standardkontingent
Maximale Größe eines Schlüssels in einem Schlüssel-Wert-Paar	512 Byte
Maximale Größe des Werts in einem Schlüssel-Wert-Paar	1 KB
Maximale Anzahl von Schlüssel-Wert-Paaren, die Sie in einer einzigen API-Anfrage aktualisieren können	50 Schlüssel oder 3 MB Nutzlast, je nachdem, was zuerst erreicht wird
Maximale Größe eines einzelnen Schlüsselwertspeichers	5 MB
Maximale Anzahl von Funktionen, denen ein einzelner Schlüssel wertspeicher zugeordnet werden kann	10
Maximale Anzahl von Schlüsselwertspeichern pro Funktion	1
Maximale Anzahl von Schlüsselwertspeichern pro Konto	50 Höheres Kontingent anfordern

Kontingente für Lambda@Edge

Allgemeine Kontingente

Entity	Standardkontingent
Derartige AWS-Konto Distributionen können Lambda @Edge -Funktion en haben	500 Höheres Kontingent anfordern
Lambda@Edge-Funktionen pro Verteilung	100

Entity	Standardkontingent
	Höheres Kontingent anfordern
Gleichzeitige Ausführungen **Note** Lambda verwaltet die Parallelitätsquoten für Lambda @Edge. Alle Lambda-Funktionen in der AWS-Region teilen sich dieses Kontingent. Weitere Informationen finden Sie unter Funktionsskalierung im AWS Lambda -Entwicklerhandbuch.	1.000 (jeweils AWS-Region) Höheres Kontingent anfordern
Verteilungen, die derselben Funktion zugeordnet sind	500
Maximale komprimierte Größe einer Lambda-Funktion und aller enthaltenen Bibliotheken	50 MB
Lambda @Edge -Anfragen pro Sekunde (jeweils unterstützt AWS-Region). Weitere Informationen finden Sie unter Parallelitätsquoten im AWS Lambda Entwicklerhandbuch.	10.000

Kontingente, die sich nach Ereignistyp unterscheiden

Entity	Viewer-Anfrage- und Viewer-Antwortereignisse	Ursprungsanfrage- und Ursprungsantwortereignisse
Funktionsspeichergröße	128 MB	Wie Lambda-Kontingente
Funktions-Timeout. Die Funktion kann Netzwerkaufrufe an Ressourcen wie Amazon	5 Sekunden	30 Sekunden

Entity	Viewer-Anfrage- und Viewer-Antworterei gnisse	Ursprungsanfrage- und Ursprungs antwortereignisse
S3 S3-Buckets, DynamoDB-Tabellen oder EC2 Amazon-Instances in tätigen. AWS-Regionen		
Größe einer Antwort, die von einer Lambda-Fu nktion erzeugt wird, einschließlich Header und Textkörper	40 KB	1 MB

Hinweise

- Eine Liste der zusätzlichen Lambda @Edge -Kontingente, die über Service Quotas erhöht werden können, finden Sie unter CloudFrontAmazon-Endpunkte und Kontingente in der. Allgemeine AWS-Referenz
- Zusätzlich zu diesen Kontingenten gibt es bei der Verwendung der Lambda@Edge-Funktionen noch einige andere Einschränkungen. Weitere Informationen finden Sie unter Einschränkungen für Lambda@Edge.

Kontingente für SSL-Zertifikate

Entity	Standardkontingent
SSL-Zertifikate gelten für die AWS-Konto Bearbeitung von HTTPS-Anf ragen mit dedizierten IP-Adressen (kein Kontingent bei der Bearbeitung von HTTPS-Anfragen über SNI) Weitere Informationen finden Sie unter Verwenden Sie HTTPS mit CloudFront.	2 Höheres Kontingent anfordern
SSL-Zertifikate, die einer CloudFront Distribution zugeordnet werden können	1

Wenn Ihr SSL-Zertifikat speziell für die HTTPS-Kommunikation zwischen Zuschauern vorgesehen ist und CloudFront Sie AWS Certificate Manager (ACM) oder den IAM-Zertifikatsspeicher für die Bereitstellung oder den Import Ihres Zertifikats verwendet haben, gelten zusätzliche Kontingente. Weitere Informationen finden Sie unter Kontingente für die Verwendung von SSL/TLS Zertifikaten mit CloudFront (HTTPS nur zwischen Zuschauern und CloudFront nur).

Es gibt auch Kontingente für die Anzahl der SSL-Zertifikate, die Sie in AWS Certificate Manager (ACM) importieren oder hochladen AWS Identity and Access Management (IAM) können. Weitere Informationen finden Sie unter Erhöhen Sie die Kontingente für SSL/TLS-Zertifikate.

Kontingente für Aufhebungen

Entity	Standardkontingent
Dateiinvalidierung: maximale Anzahl von Dateien, die in aktiven Invalidierungsanfragen erlaubt sind, mit Ausnahme von Platzhalterinvalidierungen Weitere Informationen finden Sie unter Machen Sie Dateien ungültig, um Inhalte zu entfernen.	3,000
Dateiinvalidierung: maximal zulässige Anzahl aktiver Platzhalterinvalidierungen	15
Dateiinvalidierung: maximale Anzahl von Dateien, die eine Platzhalterinvalidierung verarbeiten kann	Kein Kontingent

Kontingente für Schlüsselgruppen

Entity	Standardkontingent
Öffentliche Schlüssel in einer einzelnen Schlüsselgruppe	5 Höheres Kontingent anfordern
Schlüsselgruppen, die einem einzelnen Cache-Verhalten zugeordnet sind	4

Entity	Standardkontingent
	Höheres Kontingent anfordern
Schlüsselgruppen pro AWS-Konto	10 Höheres Kontingent anfordern
Verteilungen, die einer einzelnen Schlüsselgruppe zugeordnet sind	100 Höheres Kontingent anfordern

Kontingente für WebSocket Verbindungen

Entity	Standardkontingent
Ursprungs-Reaktions-Timeout (Leerlauf-Timeout)	10 Minuten Wenn CloudFront innerhalb der letzten 10 Minuten keine Byte erkannt wurden, die vom Ursprung an den Client gesendet wurden, wird davon ausgegangen, dass die Verbindung inaktiv ist, und sie wird geschlossen.

Kontingente für Verschlüsselung auf Feldebene

Entity	Standardkontingent
Maximale Länge eines zu verschlüsselnden Feldes Weitere Informationen finden Sie unter Vertrauliche Daten durch Verschlüsselung auf Feldebene schützen.	16 KB
Maximale Anzahl von Feldern in einem Anforderungstext bei konfiguri erter Verschlüsselung auf Feldebene	10
Maximale Länge eines Anforderungstexts, wenn die Verschlüsselung auf Feldebene konfiguriert ist	1 MB
Maximale Anzahl von Verschlüsselungskonfigurationen auf Feldebene, die einer Konfiguration zugeordnet werden können AWS-Konto	10
Maximale Anzahl von Verschlüsselungsprofilen auf Feldebene, die einem zugeordnet werden können AWS-Konto	10
Maximale Anzahl der öffentlichen Schlüssel, die einem AWS-Konto hinzugefügt werden können	10
Maximale Anzahl der zu verschlüsselnden Felder, die in einem Profil angegeben werden kann	10
Maximale Anzahl von CloudFront Distributionen, die einer Verschlüs selungskonfiguration auf Feldebene zugeordnet werden können	20
Maximale Anzahl der Abfrageargument-Profilzuordnungen, die in eine Konfiguration für die Verschlüsselung auf Feldebene aufgenommen werden können	5

Kontingente für Cookies (Legacy-Cache-Einstellungen)

Diese Kontingente gelten für die älteren CloudFront Cache-Einstellungen. Wir empfehlen, anstelle der alten Einstellungen eine Cache-Richtlinie oder eine Origin-Request-Richtlinie zu verwenden.

Entity	Standardkontingent
Cookies pro Cache-Verhalten Weitere Informationen finden Sie unter Auf Cookies basierender Inhalt zwischenspeichern.	10 Höheres Kontingent anfordern
Gesamtzahl der Byte in Cookie-Namen (gilt nicht, wenn Sie so konfiguri eren CloudFront , dass alle Cookies an den Ursprung weitergeleitet werden)	512 abzüglich der Anzahl der Cookies

Kontingente für Abfragezeichenfolgen (Legacy-Cache-Einstellungen)

Diese Kontingente gelten für CloudFront die älteren Cache-Einstellungen. Wir empfehlen, anstelle der alten Einstellungen eine Cache-Richtlinie oder eine Origin-Request-Richtlinie zu verwenden.

Entity	Standardkontingent
Maximale Anzahl von Zeichen in einer Abfragezeichenfolge	128 Zeichen
Maximale Anzahl der Zeichen für alle Abfragezeichenfolgen im selben Parameter	512 Zeichen
Abfragestrings pro Cache-Verhalten Weitere Informationen finden Sie unter Inhalt auf der Grundlage von Abfragezeichenfolgenparametern zwischenspeichern.	10 Höheres Kontingent anfordern

Kontingente für Header

Entity	Standardkontingent
Header pro Cache-Verhalten (Legacy-Cache-Einstellungen)	10

Entity	Standardkontingent
Weitere Informationen finden Sie unter the section called “Inhalt auf der Grundlage von Anforderungsheadern zwischenspeichern”.	Höheres Kontingent anfordern
Header pro Cache-Verhalten weiterleiten	25 Höheres Kontingent anfordern
Benutzerdefinierte Header: maximale Anzahl von benutzerdefinierten Headern, die Sie konfigurieren können, um sie CloudFront zu ursprüngl ichen Anfragen hinzuzufügen Weitere Informationen finden Sie unter the section called “Fügen Sie benutzerdefinierte Header zu ursprünglichen Anfragen hinzu”.	30 Höheres Kontingent anfordern
Benutzerdefinierte Header: maximale Anzahl benutzerdefinierter Header, die Sie einer Antwort-Header-Richtlinie hinzufügen können	10 Höheres Kontingent anfordern
Benutzerdefinierte Header: maximale Länge eines Header-Namens	256 Zeichen
Benutzerdefinierte Header: maximale Länge eines Header-Wertes	1,783 Zeichen
Benutzerdefinierte Header: maximale Länge aller Werte und Namen kombiniert	10,240 Zeichen
Maximale Länge des Header-Werts `Content-Security-Policy`	1,783 Zeichen Höheres Kontingent anfordern
Maximale Länge eines `CORS (Access-Control-Allow-Origi n)` Header-Werts	1,783 Zeichen

Kontingente für Distributionen mit mehreren Mandanten

Entity	Standardkontingent
Maximale Anzahl von Distributionsmandanten pro AWS-Konto	10.000 Höheres Kontingent anfordern
Maximale Anzahl von Multi-Tenant-Distributionen pro AWS-Konto	20 Höheres Kontingent anfordern
Maximale Anzahl von Verbindungsgruppen pro AWS-Konto	100 Höheres Kontingent anfordern
Maximale Anzahl von Aliasen pro Distributionsmandant	100 Höheres Kontingent anfordern
Maximale Anzahl von Parametern pro Verteilungsmandant	5 Höheres Kontingent anfordern
Maximale Anzahl von Parametern pro Verteilung mit mehreren Mandanten	5 Höheres Kontingent anfordern
Maximale Anzahl von Parametern in einem Feld in einer Mehrmandantenverteilung	2 Höheres Kontingent anfordern
Maximale Anzahl von Verbindungsgruppen pro statischer Anycast-IP-Liste	5

Entity	Standardkontingent
	Höheres Kontingent anfordern

Weitere Informationen zu Multi-Tenant-Distributionen finden Sie unter. Erfahren Sie, wie Distributionen mit mehreren Mandanten funktionieren

Ähnliche Informationen

Weitere Informationen finden Sie unter CloudFront Amazon-Endpunkte und Kontingente in der Allgemeine AWS-Referenz.

Codebeispiele für die CloudFront Verwendung AWS SDKs

Die folgenden Codebeispiele zeigen die Verwendung CloudFront mit einem AWS Software Development Kit (SDK).

Aktionen sind Codeauszüge aus größeren Programmen und müssen im Kontext ausgeführt werden. Während Aktionen Ihnen zeigen, wie Sie einzelne Service-Funktionen aufrufen, können Sie Aktionen im Kontext der zugehörigen Szenarios anzeigen.

Szenarien sind Code-Beispiele, die Ihnen zeigen, wie Sie bestimmte Aufgaben ausführen, indem Sie mehrere Funktionen innerhalb eines Services aufrufen oder mit anderen AWS-Services kombinieren.

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Codebeispiele

- Löschen Sie CloudFront Signaturressourcen mithilfe des AWS SDK
- Erstellen Sie signierte Cookies URLs und Cookies mithilfe eines AWS SDK

- CloudFront Funktionen, Beispiele für CloudFront
  - Fügen Sie einem Antwortereignis des CloudFront Functions-Viewers HTTP-Sicherheitsheader hinzu
  - Fügen Sie einem CloudFront Functions-Viewer-Antwortereignis einen CORS-Header hinzu
  - Fügen Sie einem Antwortereignis des CloudFront Functions-Viewers einen Cache-Control-Header hinzu
  - Fügen Sie einem CloudFront Functions-Viewer-Anforderungsereignis einen echten Client-IP-Header hinzu
  - Fügen Sie einem CloudFront Functions-Viewer-Anforderungsereignis einen Origin-Header hinzu
  - Fügen Sie index.html zu einer Anfrage URLs ohne Dateinamen in einem CloudFront Functions-Viewer-Anforderungsereignis hinzu
  - Normalisieren Sie die Parameter der Abfragezeichenfolge in einer CloudFront Functions Viewer-Anforderung
  - In einem CloudFront Functions-Viewer-Anforderungsereignis zu einer neuen URL umleiten
  - Schreiben Sie einen Anforderungs-URI auf der Grundlage der KeyValueStore Konfiguration für ein CloudFront Functions-Viewer-Anforderungsereignis neu
  - Leiten Sie Anfragen in einem CloudFront Functions-Viewer-Anforderungsereignis an einen Ursprung weiter, der näher am Betrachter liegt
  - Verwenden Sie Schlüssel-Wert-Paare in einer CloudFront Functions-Viewer-Anfrage
  - Validieren Sie ein einfaches Token in einer CloudFront Functions-Viewer-Anfrage

Grundlegende Beispiele für die CloudFront Verwendung AWS SDKs

Die folgenden Codebeispiele zeigen, wie Sie die Grundlagen von Amazon CloudFront mit verwenden können AWS SDKs.

Beispiele

- Aktionen zur CloudFront Verwendung AWS SDKs
  - Verwendung CreateDistribution mit einem AWS SDK oder CLI

- CreateFunctionMit einem AWS SDK verwenden
- Verwendung von CreateInvalidation mit einer CLI
- CreateKeyGroupMit einem AWS SDK verwenden
- Verwendung CreatePublicKey mit einem AWS SDK oder CLI
- Verwendung DeleteDistribution mit einem AWS SDK oder CLI
- Verwendung von GetCloudFrontOriginAccessIdentity mit einer CLI
- Verwendung von GetCloudFrontOriginAccessIdentityConfig mit einer CLI
- Verwendung von GetDistribution mit einer CLI
- Verwendung GetDistributionConfig mit einem AWS SDK oder CLI
- Verwendung von ListCloudFrontOriginAccessIdentities mit einer CLI
- Verwendung ListDistributions mit einem AWS SDK oder CLI
- Verwendung UpdateDistribution mit einem AWS SDK oder CLI

Aktionen zur CloudFront Verwendung AWS SDKs

Die folgenden Codebeispiele zeigen, wie Sie einzelne CloudFront Aktionen mit ausführen AWS SDKs. Jedes Beispiel enthält einen Link zu GitHub, wo Sie Anweisungen zum Einrichten und Ausführen des Codes finden.

Diese Auszüge rufen die CloudFront API auf und sind Codeauszüge aus größeren Programmen, die im Kontext ausgeführt werden müssen. Sie können Aktionen im Kontext unter Szenarien für die CloudFront Verwendung AWS SDKs anzeigen.

Die folgenden Beispiele enthalten nur die am häufigsten verwendeten Aktionen. Eine vollständige Liste finden Sie in der Amazon CloudFront API-Referenz.

Beispiele

- Verwendung CreateDistribution mit einem AWS SDK oder CLI
- CreateFunctionMit einem AWS SDK verwenden
- Verwendung von CreateInvalidation mit einer CLI
- CreateKeyGroupMit einem AWS SDK verwenden
- Verwendung CreatePublicKey mit einem AWS SDK oder CLI
- Verwendung DeleteDistribution mit einem AWS SDK oder CLI
- Verwendung von GetCloudFrontOriginAccessIdentity mit einer CLI

- Verwendung von GetCloudFrontOriginAccessIdentityConfig mit einer CLI
- Verwendung von GetDistribution mit einer CLI
- Verwendung GetDistributionConfig mit einem AWS SDK oder CLI
- Verwendung von ListCloudFrontOriginAccessIdentities mit einer CLI
- Verwendung ListDistributions mit einem AWS SDK oder CLI
- Verwendung UpdateDistribution mit einem AWS SDK oder CLI

Verwendung **`CreateDistribution`** mit einem AWS SDK oder CLI

Die folgenden Code-Beispiele zeigen, wie `CreateDistribution` verwendet wird.

Beispiele für Aktionen sind Codeauszüge aus größeren Programmen und müssen im Kontext ausgeführt werden. Im folgenden Codebeispiel können Sie diese Aktion im Kontext sehen:

- Erstellen Sie einen mandantenfähigen Vertriebs- und Distributionsmandanten

CLI

AWS CLI

Beispiel 1: Um eine CloudFront Distribution zu erstellen

Im folgenden `create-distribution` Beispiel wird mithilfe von Befehlszeilenargumenten eine Distribution für einen S3-Bucket mit dem Namen `amzn-s3-demo-bucket` und der Angabe `index.html` als Standard-Root-Objekt erstellt.

```
aws cloudfront create-distribution \
    --origin-domain-name amzn-s3-demo-bucket.s3.amazonaws.com \
    --default-root-object index.html
```

Ausgabe:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EMLARXS9EXAMPLE",
    "ETag": "E9LHASXEXAMPLE",
    "Distribution": {
        "Id": "EMLARXS9EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
```

```
        "Status": "InProgress",
        "LastModifiedTime": "2019-11-22T00:55:15.705Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                        "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        }
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-
example",
                "ForwardedValues": {
                    "QueryString": false,
                    "Cookies": {
                        "Forward": "none"
                    },
                    "Headers": {
                        "Quantity": 0
                    },
```

```
                    "QueryStringCacheKeys": {
                        "Quantity": 0
                    }
                },
                "TrustedSigners": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "MinTTL": 0,
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "SmoothStreaming": false,
                "DefaultTTL": 86400,
                "MaxTTL": 31536000,
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": ""
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
```

```
                "Prefix": ""
            },
            "PriceClass": "PriceClass_All",
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "IsIPV6Enabled": true
        }
    }
}
```

Beispiel 2: Um eine CloudFront Distribution mit einer JSON-Datei zu erstellen

Im folgenden `create-distribution` Beispiel wird mithilfe einer JSON-Datei eine Distribution für einen S3-Bucket mit dem Namen `amzn-s3-demo-bucket` und der Angabe `index.html` als Standard-Root-Objekt erstellt.

```
aws cloudfront create-distribution \
    --distribution-config file://dist-config.json
```

Inhalt von `dist-config.json`:

```
{
    "CallerReference": "cli-example",
    "Aliases": {
        "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
```

```
                "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                "OriginPath": "",
                "CustomHeaders": {
                    "Quantity": 0
                },
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                }
            }
        ]
    },
    "OriginGroups": {
        "Quantity": 0
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
        "ForwardedValues": {
            "QueryString": false,
            "Cookies": {
                "Forward": "none"
            },
            "Headers": {
                "Quantity": 0
            },
            "QueryStringCacheKeys": {
                "Quantity": 0
            }
        },
        "TrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "ViewerProtocolPolicy": "allow-all",
        "MinTTL": 0,
        "AllowedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ],
            "CachedMethods": {
                "Quantity": 2,
                "Items": [
```

```
                    "HEAD",
                    "GET"
                ]
            }
        },
        "SmoothStreaming": false,
        "DefaultTTL": 86400,
        "MaxTTL": 31536000,
        "Compress": false,
        "LambdaFunctionAssociations": {
            "Quantity": 0
        },
        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
```

```
}
```

Eine Beispielausgabe finden Sie in Beispiel 1.

Beispiel 3: So erstellen Sie eine CloudFront Multi-Tenant-Distribution mit einem Zertifikat

Im folgenden `create-distribution` Beispiel wird eine CloudFront Distribution mit Mehrmandantenunterstützung erstellt und ein TLS-Zertifikat angegeben.

```
aws cloudfront create-distribution \
    --distribution-config file://dist-config.json
```

Inhalt von `dist-config.json`:

```
{
    "CallerReference": "cli-example-with-cert",
    "Comment": "CLI example distribution",
    "DefaultRootObject": "index.html",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
                "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "DomainName": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "OriginPath": "/{{tenantName}}",
                "CustomHeaders": {
                    "Quantity": 0
                },
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                }
            }
        ]
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
        "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5ABC",
        "ViewerProtocolPolicy": "allow-all",
        "AllowedMethods": {
            "Quantity": 2,
            "Items": ["HEAD", "GET"],
            "CachedMethods": {
                "Quantity": 2,
```

```
                "Items": ["HEAD", "GET"]
            }
        }
    },
    "Enabled": true,
    "ViewerCertificate": {
        "ACMCertificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/191306a1-db01-49ca-90ef-fc414ee5dabc",
        "SSLSupportMethod": "sni-only"
    },
    "HttpVersion": "http2",
    "ConnectionMode": "tenant-only",
    "TenantConfig": {
        "ParameterDefinitions": [
            {
                "Name": "tenantName",
                "Definition": {
                    "StringSchema": {
                        "Comment": "tenantName parameter",
                        "DefaultValue": "root",
                        "Required": false
                    }
                }
            }
        ]
    }
}
```

Ausgabe:

```
{
    "Location": "https://cloudfront.amazonaws.com/2020-05-31/distribution/
E1HVIAU7UABC",
    "ETag": "E20LT7R1BABC",
    "Distribution": {
        "Id": "E1HVIAU7U12ABC",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/E1HVIAU7U12ABC",
        "Status": "InProgress",
        "LastModifiedTime": "2025-07-10T20:33:31.117000+00:00",
        "InProgressInvalidationBatches": 0,
        "DomainName": "example.com",
        "ActiveTrustedSigners": {
            "Enabled": false,
```

```
            "Quantity": 0
        },
        "ActiveTrustedKeyGroups": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example-with-cert",
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                        "DomainName": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                        "OriginPath": "/{{tenantName}}",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        },
                        "ConnectionAttempts": 3,
                        "ConnectionTimeout": 10,
                        "OriginShield": {
                            "Enabled": false
                        },
                        "OriginAccessControlId": ""
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                "TrustedKeyGroups": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "AllowedMethods": {
```

```
                    "Quantity": 2,
                    "Items": ["HEAD", "GET"],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": ["HEAD", "GET"]
                    }
                },
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": "",
                "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5ABC",
                "GrpcConfig": {
                    "Enabled": false
                }
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "CLI example distribution",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": false,
                "ACMCertificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/1954f095-11b6-4daf-9952-0c308a00abc",
                "SSLSupportMethod": "sni-only",
                "MinimumProtocolVersion": "TLSv1.2_2021",
                "Certificate": "arn:aws:acm:us-
east-1:123456789012:certificate/1954f095-11b6-4daf-9952-0c308a00abc",
                "CertificateSource": "acm"
            },
```

```
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "TenantConfig": {
                "ParameterDefinitions": [
                    {
                        "Name": "tenantName",
                        "Definition": {
                            "StringSchema": {
                                "Comment": "tenantName parameter",
                                "DefaultValue": "root",
                                "Required": false
                            }
                        }
                    }
                ]
            },
            "ConnectionMode": "tenant-only"
        }
    }
}
```

Weitere Informationen finden Sie unter Arbeiten mit Distributionen im Amazon CloudFront Developer Guide.

Beispiel 4: So erstellen Sie eine CloudFront Multi-Tenant-Distribution ohne Zertifikat

Im folgenden `create-distribution` Beispiel wird eine CloudFront Distribution mit Mehrmandantenunterstützung, aber ohne TLS-Zertifikat erstellt.

```
aws cloudfront create-distribution \
    --distribution-config file://dist-config.json
```

Inhalt von `dist-config.json`:

```
{
    "CallerReference": "cli-example",
    "Comment": "CLI example distribution",
```

```
    "DefaultRootObject": "index.html",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
                "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "DomainName": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "OriginPath": "/{{tenantName}}",
                "CustomHeaders": {
                    "Quantity": 0
                },
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                }
            }
        ]
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
        "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5ABC",
        "ViewerProtocolPolicy": "allow-all",
        "AllowedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ],
            "CachedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ]
            }
        }
    },
    "Enabled": true,
    "HttpVersion": "http2",
    "ConnectionMode": "tenant-only",
    "TenantConfig": {
        "ParameterDefinitions": [
            {
                "Name": "tenantName",
                "Definition": {
```

```
                    "StringSchema": {
                        "Comment": "tenantName parameter",
                        "DefaultValue": "root",
                        "Required": false
                    }
                }
            }
        ]
    }
}
```

Ausgabe:

```
{
    "Location": "https://cloudfront.amazonaws.com/2020-05-31/distribution/
E2GJ5J9QN12ABC",
    "ETag": "E37YLVVQIABC",
    "Distribution": {
        "Id": "E2GJ5J9QNABC",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/E2GJ5J9QN12ABC",
        "Status": "InProgress",
        "LastModifiedTime": "2025-07-10T20:35:20.565000+00:00",
        "InProgressInvalidationBatches": 0,
        "DomainName": "example.com",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "ActiveTrustedKeyGroups": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example-no-cert",
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                        "DomainName": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                        "OriginPath": "/{{tenantName}}",
```

```
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        },
                        "ConnectionAttempts": 3,
                        "ConnectionTimeout": 10,
                        "OriginShield": {
                            "Enabled": false
                        },
                        "OriginAccessControlId": ""
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                "TrustedKeyGroups": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
```

```
                "FunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": "",
                "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5ABC",
                "GrpcConfig": {
                    "Enabled": false
                }
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "CLI example distribution",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "SSLSupportMethod": "sni-only",
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "TenantConfig": {
                "ParameterDefinitions": [
                    {
                        "Name": "tenantName",
                        "Definition": {
                            "StringSchema": {
                                "Comment": "tenantName parameter",
```

```
                                "DefaultValue": "root",
                                "Required": false
                            }
                        }
                    }
                ]
            },
            "ConnectionMode": "tenant-only"
        }
    }
}
```

Weitere Informationen finden Sie unter Distributionen konfigurieren im Amazon CloudFront Developer Guide.

- Einzelheiten zur API finden Sie CreateDistributionin der AWS CLI Befehlsreferenz.

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

Im folgenden Beispiel wird ein Amazon Simple Storage Service (Amazon S3) -Bucket als Inhaltsquelle verwendet.

Nach dem Erstellen der Verteilung erstellt der Code eine CloudFrontWaiterOption, mit der Sie warten müssen, bis die Verteilung bereitgestellt wurde, bevor die Verteilung zurückgegeben wird.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.core.internal.waiters.ResponseOrException;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import
 software.amazon.awssdk.services.cloudfront.model.CreateDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.Distribution;
```

```
import software.amazon.awssdk.services.cloudfront.model.GetDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.ItemSelection;
import software.amazon.awssdk.services.cloudfront.model.Method;
import software.amazon.awssdk.services.cloudfront.model.ViewerProtocolPolicy;
import software.amazon.awssdk.services.cloudfront.waiters.CloudFrontWaiter;
import software.amazon.awssdk.services.s3.S3Client;

import java.time.Instant;

public class CreateDistribution {

       private static final Logger logger =
 LoggerFactory.getLogger(CreateDistribution.class);

       public static Distribution createDistribution(CloudFrontClient
 cloudFrontClient, S3Client s3Client,
                       final String bucketName, final String keyGroupId, final
 String originAccessControlId) {

               final String region = s3Client.headBucket(b ->
 b.bucket(bucketName)).sdkHttpResponse().headers()
                               .get("x-amz-bucket-region").get(0);
               final String originDomain = bucketName + ".s3." + region +
 ".amazonaws.com";
               String originId = originDomain; // Use the originDomain value for
 the originId.

               // The service API requires some deprecated methods, such as
               // DefaultCacheBehavior.Builder#minTTL and #forwardedValue.
               CreateDistributionResponse createDistResponse =
 cloudFrontClient.createDistribution(builder -> builder
                               .distributionConfig(b1 -> b1
                                               .origins(b2 -> b2
                                                               .quantity(1)
                                                               .items(b3 -> b3

 .domainName(originDomain)

 .id(originId)

 .s3OriginConfig(builder4 -> builder4

              .originAccessIdentity(
```

```
                                ""))
.originAccessControlId(
              originAccessControlId)))
                                                .defaultCacheBehavior(b2 -> b2
.viewerProtocolPolicy(ViewerProtocolPolicy.ALLOW_ALL)
.targetOriginId(originId)
                                                                    .minTTL(200L)
.forwardedValues(b5 -> b5
.cookies(cp -> cp
              .forward(ItemSelection.NONE))
.queryString(true))
.trustedKeyGroups(b3 -> b3
.quantity(1)
.items(keyGroupId)
.enabled(true))
.allowedMethods(b4 -> b4
.quantity(2)
.items(Method.HEAD, Method.GET)
.cachedMethods(b5 -> b5
              .quantity(2)
              .items(Method.HEAD,
                              Method.GET))))
                                                .cacheBehaviors(b -> b
                                                                .quantity(1)
```

```
                                                                .items(b2 -> b2
.pathPattern("/index.html")
.viewerProtocolPolicy(
              ViewerProtocolPolicy.ALLOW_ALL)
.targetOriginId(originId)
.trustedKeyGroups(b3 -> b3
              .quantity(1)
              .items(keyGroupId)
              .enabled(true))
.minTTL(200L)
.forwardedValues(b4 -> b4
              .cookies(cp -> cp
                            .forward(ItemSelection.NONE))
              .queryString(true))
.allowedMethods(b5 -> b5.quantity(2)
              .items(Method.HEAD,
                            Method.GET)
              .cachedMethods(b6 -> b6
                            .quantity(2)
                            .items(Method.HEAD,
                                          Method.GET)))))
                                           .enabled(true)
                                           .comment("Distribution built with
java")
```

```
 .callerReference(Instant.now().toString())));

                final Distribution distribution =
 createDistResponse.distribution();
                logger.info("Distribution created. DomainName: [{}]  Id: [{}]",
 distribution.domainName(),
                                distribution.id());
                logger.info("Waiting for distribution to be deployed ...");
                try (CloudFrontWaiter cfWaiter =
 CloudFrontWaiter.builder().client(cloudFrontClient).build()) {
                        ResponseOrException<GetDistributionResponse>
 responseOrException = cfWaiter
                                        .waitUntilDistributionDeployed(builder ->
 builder.id(distribution.id()))
                                        .matched();
                        responseOrException.response()
                                        .orElseThrow(() -> new
 RuntimeException("Distribution not created"));
                        logger.info("Distribution deployed. DomainName: [{}]  Id:
 [{}]", distribution.domainName(),
                                        distribution.id());
                }
                return distribution;
        }
}
```

- Einzelheiten zur API finden Sie CreateDistributionunter AWS SDK for Java 2.x API-Referenz.

PowerShell

Tools für PowerShell V4

Beispiel 1: Erstellt eine CloudFront Basisdistribution, die mit Protokollierung und Caching konfiguriert ist.

```
$origin = New-Object Amazon.CloudFront.Model.Origin
$origin.DomainName = "amzn-s3-demo-bucket.s3.amazonaws.com"
$origin.Id = "UniqueOrigin1"
$origin.S3OriginConfig = New-Object Amazon.CloudFront.Model.S3OriginConfig
```

```
$origin.S3OriginConfig.OriginAccessIdentity = ""
New-CFDistribution `
      -DistributionConfig_Enabled $true `
      -DistributionConfig_Comment "Test distribution" `
      -Origins_Item $origin `
      -Origins_Quantity 1 `
      -Logging_Enabled $true `
      -Logging_IncludeCookie $true `
      -Logging_Bucket amzn-s3-demo-logging-bucket.s3.amazonaws.com `
      -Logging_Prefix "help/" `
      -DistributionConfig_CallerReference Client1 `
      -DistributionConfig_DefaultRootObject index.html `
      -DefaultCacheBehavior_TargetOriginId $origin.Id `
      -ForwardedValues_QueryString $true `
      -Cookies_Forward all `
      -WhitelistedNames_Quantity 0 `
      -TrustedSigners_Enabled $false `
      -TrustedSigners_Quantity 0 `
      -DefaultCacheBehavior_ViewerProtocolPolicy allow-all `
      -DefaultCacheBehavior_MinTTL 1000 `
      -DistributionConfig_PriceClass "PriceClass_All" `
      -CacheBehaviors_Quantity 0 `
      -Aliases_Quantity 0
```

- Einzelheiten zur API finden Sie unter CreateDistribution AWS -Tools für PowerShellCmdlet-Referenz (V4).

Tools für V5 PowerShell

Beispiel 1: Erstellt eine CloudFront Basisdistribution, die mit Protokollierung und Caching konfiguriert ist.

```
$origin = New-Object Amazon.CloudFront.Model.Origin
$origin.DomainName = "amzn-s3-demo-bucket.s3.amazonaws.com"
$origin.Id = "UniqueOrigin1"
$origin.S3OriginConfig = New-Object Amazon.CloudFront.Model.S3OriginConfig
$origin.S3OriginConfig.OriginAccessIdentity = ""
New-CFDistribution `
      -DistributionConfig_Enabled $true `
      -DistributionConfig_Comment "Test distribution" `
      -Origins_Item $origin `
      -Origins_Quantity 1 `
      -Logging_Enabled $true `
      -Logging_IncludeCookie $true `
```

```
    -Logging_Bucket amzn-s3-demo-logging-bucket.s3.amazonaws.com `
    -Logging_Prefix "help/" `
    -DistributionConfig_CallerReference Client1 `
    -DistributionConfig_DefaultRootObject index.html `
    -DefaultCacheBehavior_TargetOriginId $origin.Id `
    -ForwardedValues_QueryString $true `
    -Cookies_Forward all `
    -WhitelistedNames_Quantity 0 `
    -TrustedSigners_Enabled $false `
    -TrustedSigners_Quantity 0 `
    -DefaultCacheBehavior_ViewerProtocolPolicy allow-all `
    -DefaultCacheBehavior_MinTTL 1000 `
    -DistributionConfig_PriceClass "PriceClass_All" `
    -CacheBehaviors_Quantity 0 `
    -Aliases_Quantity 0
```

- Einzelheiten zur API finden Sie unter CreateDistribution AWS -Tools für PowerShellCmdlet-Referenz (V5).

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

`CreateFunction`Mit einem AWS SDK verwenden

Das folgende Codebeispiel zeigt, wie es verwendet wird`CreateFunction`.

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

```
import software.amazon.awssdk.core.SdkBytes;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.CloudFrontException;
```

```
import software.amazon.awssdk.services.cloudfront.model.CreateFunctionRequest;
import software.amazon.awssdk.services.cloudfront.model.CreateFunctionResponse;
import software.amazon.awssdk.services.cloudfront.model.FunctionConfig;
import software.amazon.awssdk.services.cloudfront.model.FunctionRuntime;
import java.io.InputStream;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class CreateFunction {

    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <functionName> <filePath>

                Where:
                    functionName - The name of the function to create.\s
                    filePath - The path to a file that contains the application
 logic for the function.\s
                """;

        if (args.length != 2) {
            System.out.println(usage);
            System.exit(1);
        }

        String functionName = args[0];
        String filePath = args[1];
        CloudFrontClient cloudFrontClient = CloudFrontClient.builder()
                .region(Region.AWS_GLOBAL)
                .build();

        String funArn = createNewFunction(cloudFrontClient, functionName,
 filePath);
        System.out.println("The function ARN is " + funArn);
        cloudFrontClient.close();
```

```
    }

    public static String createNewFunction(CloudFrontClient cloudFrontClient,
 String functionName, String filePath) {
        try {
            InputStream fileIs =
 CreateFunction.class.getClassLoader().getResourceAsStream(filePath);
            SdkBytes functionCode = SdkBytes.fromInputStream(fileIs);

            FunctionConfig config = FunctionConfig.builder()
                    .comment("Created by using the CloudFront Java API")
                    .runtime(FunctionRuntime.CLOUDFRONT_JS_1_0)
                    .build();

            CreateFunctionRequest functionRequest =
 CreateFunctionRequest.builder()
                    .name(functionName)
                    .functionCode(functionCode)
                    .functionConfig(config)
                    .build();

            CreateFunctionResponse response =
 cloudFrontClient.createFunction(functionRequest);
            return response.functionSummary().functionMetadata().functionARN();

        } catch (CloudFrontException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
        return "";
    }
}
```

- Einzelheiten zur API finden Sie CreateFunctionin der AWS SDK for Java 2.x API-Referenz.

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung von **`CreateInvalidation`** mit einer CLI

Die folgenden Code-Beispiele zeigen, wie `CreateInvalidation` verwendet wird.

CLI

AWS CLI

Um eine Invalidierung für eine CloudFront Distribution zu erstellen

Das folgende `create-invalidation` Beispiel erstellt eine Invalidierung für die angegebenen Dateien in der angegebenen CloudFront Distribution:

```
aws cloudfront create-invalidation \
    --distribution-id EDFDVBD6EXAMPLE \
    --paths "/example-path/example-file.jpg" "/example-path/example-file2.png"
```

Ausgabe:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EDFDVBD6EXAMPLE/invalidation/I1JLWSDAP8FU89",
    "Invalidation": {
        "Id": "I1JLWSDAP8FU89",
        "Status": "InProgress",
        "CreateTime": "2019-12-05T18:24:51.407Z",
        "InvalidationBatch": {
            "Paths": {
                "Quantity": 2,
                "Items": [
                    "/example-path/example-file2.png",
                    "/example-path/example-file.jpg"
                ]
            },
            "CallerReference": "cli-1575570291-670203"
        }
    }
}
```

Im vorherigen Beispiel generierte die AWS CLI automatisch ein zufälliges Ergebnis`CallerReference`. Um Ihre eigenen Parameter anzugeben oder um zu vermeiden`CallerReference`, dass die Invalidierungsparameter als Befehlszeilenargumente

übergeben werden, können Sie eine JSON-Datei verwenden. Im folgenden Beispiel wird eine Invalidierung für zwei Dateien erstellt, indem die Invalidierungsparameter in einer JSON-Datei mit dem Namen angegeben werden: `inv-batch.json`

```
aws cloudfront create-invalidation \
    --distribution-id EDFDVBD6EXAMPLE \
    --invalidation-batch file://inv-batch.json
```

Inhalt von `inv-batch.json`:

```
{
    "Paths": {
        "Quantity": 2,
        "Items": [
            "/example-path/example-file.jpg",
            "/example-path/example-file2.png"
        ]
    },
    "CallerReference": "cli-example"
}
```

Ausgabe:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EDFDVBD6EXAMPLE/invalidation/I2J0I21PCUYOIK",
    "Invalidation": {
        "Id": "I2J0I21PCUYOIK",
        "Status": "InProgress",
        "CreateTime": "2019-12-05T18:40:49.413Z",
        "InvalidationBatch": {
            "Paths": {
                "Quantity": 2,
                "Items": [
                    "/example-path/example-file.jpg",
                    "/example-path/example-file2.png"
                ]
            },
            "CallerReference": "cli-example"
        }
    }
}
```

- Einzelheiten zur API finden Sie unter CreateInvalidation AWS CLIBefehlsreferenz.

PowerShell

Tools für PowerShell V4

Beispiel 1: In diesem Beispiel wird eine neue Invalidierung für eine Distribution mit der ID EXAMPLENSTXAXE erstellt. Das CallerReference ist eine eindeutige ID, die vom Benutzer ausgewählt wurde. In diesem Fall wird ein Zeitstempel verwendet, der den 15. Mai 2019 um 9:00 Uhr darstellt. Die Variable $Paths speichert drei Pfade zu Bild- und Mediendateien, die der Benutzer nicht im Cache der Distribution haben möchte. Der Parameterwert -Paths_Quantity ist die Gesamtzahl der im Parameter -Paths_Item angegebenen Pfade.

```
$Paths = "/images/*.gif", "/images/image1.jpg", "/videos/*.mp4"
New-CFInvalidation -DistributionId "EXAMPLENSTXAXE" -
InvalidationBatch_CallerReference 20190515090000 -Paths_Item $Paths -
Paths_Quantity 3
```

Ausgabe:

```
Invalidation                          Location

------------                          --------

Amazon.CloudFront.Model.Invalidation https://cloudfront.amazonaws.com/2018-11-05/
distribution/EXAMPLENSTXAXE/invalidation/EXAMPLE8NOK9H
```

- Einzelheiten zur API finden Sie unter CreateInvalidationCmdlet-Referenz (V4).AWS -Tools für PowerShell

Tools für V5 PowerShell

Beispiel 1: In diesem Beispiel wird eine neue Invalidierung für eine Distribution mit der ID EXAMPLENSTXAXE erstellt. Das CallerReference ist eine eindeutige ID, die vom Benutzer ausgewählt wurde. In diesem Fall wird ein Zeitstempel verwendet, der den 15. Mai 2019 um 9:00 Uhr darstellt. Die Variable $Paths speichert drei Pfade zu Bild- und Mediendateien, die der Benutzer nicht im Cache der Distribution haben möchte. Der Parameterwert -Paths_Quantity ist die Gesamtzahl der im Parameter -Paths_Item angegebenen Pfade.

```
$Paths = "/images/*.gif", "/images/image1.jpg", "/videos/*.mp4"
```

```
New-CFInvalidation -DistributionId "EXAMPLENSTXAXE" -
InvalidationBatch_CallerReference 20190515090000 -Paths_Item $Paths -
Paths_Quantity 3
```

Ausgabe:

```
Invalidation                          Location

------------                          --------

Amazon.CloudFront.Model.Invalidation https://cloudfront.amazonaws.com/2018-11-05/
distribution/EXAMPLENSTXAXE/invalidation/EXAMPLE8NOK9H
```

- Einzelheiten zur API finden Sie unter CreateInvalidationCmdlet-Referenz (V5).AWS -Tools für PowerShell

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

`CreateKeyGroup`Mit einem AWS SDK verwenden

Das folgende Codebeispiel zeigt, wie es verwendet wird`CreateKeyGroup`.

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

Für eine Schlüsselgruppe ist mindestens ein öffentlicher Schlüssel erforderlich, der zur Überprüfung signierter Cookies URLs oder Cookies verwendet wird.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
```

```
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;

import java.util.UUID;

public class CreateKeyGroup {
    private static final Logger logger =
 LoggerFactory.getLogger(CreateKeyGroup.class);

    public static String createKeyGroup(CloudFrontClient cloudFrontClient, String
 publicKeyId) {
        String keyGroupId = cloudFrontClient.createKeyGroup(b ->
 b.keyGroupConfig(c -> c
                .items(publicKeyId)
                .name("JavaKeyGroup" + UUID.randomUUID())))
                .keyGroup().id();
        logger.info("KeyGroup created with ID: [{}]", keyGroupId);
        return keyGroupId;
    }
}
```

- Einzelheiten zur API finden Sie CreateKeyGroupunter AWS SDK for Java 2.x API-Referenz.

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung **`CreatePublicKey`** mit einem AWS SDK oder CLI

Die folgenden Code-Beispiele zeigen, wie `CreatePublicKey` verwendet wird.

CLI

AWS CLI

Um einen CloudFront öffentlichen Schlüssel zu erstellen

Im folgenden Beispiel wird ein CloudFront öffentlicher Schlüssel erstellt, indem die Parameter in einer JSON-Datei mit dem Namen bereitgestellt `pub-key-config.json` werden. Bevor Sie diesen Befehl verwenden können, benötigen Sie einen PEM-codierten öffentlichen Schlüssel. Weitere Informationen finden Sie unter Create an RSA Key Pair im Amazon CloudFront Developer Guide.

```
aws cloudfront create-public-key \
    --public-key-config file://pub-key-config.json
```

Die Datei `pub-key-config.json` ist ein JSON-Dokument im aktuellen Ordner, das Folgendes enthält. Beachten Sie, dass der öffentliche Schlüssel im PEM-Format codiert ist.

```
{
    "CallerReference": "cli-example",
    "Name": "ExampleKey",
    "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1UOWcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAzO3ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95ylUQd9LfYTBzVZYG2tSesplcOkjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6ZqlV3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLKOX5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nrwIDAQAB\n-----
END PUBLIC KEY-----\n",
    "Comment": "example public key"
}
```

Ausgabe:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/public-key/
KDFB19YGCR002",
    "ETag": "E2QWRUHEXAMPLE",
    "PublicKey": {
        "Id": "KDFB19YGCR002",
        "CreatedTime": "2019-12-05T18:51:43.781Z",
        "PublicKeyConfig": {
            "CallerReference": "cli-example",
            "Name": "ExampleKey",
            "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1UOWcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAzO3ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95ylUQd9LfYTBzVZYG2tSesplcOkjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6ZqlV3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLKOX5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nrwIDAQAB\n-----
END PUBLIC KEY-----\n",
            "Comment": "example public key"
        }
```

```
    }
}
```

- Einzelheiten zur API finden Sie CreatePublicKeyin der AWS CLI Befehlsreferenz.

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

Das folgende Codebeispiel liest einen öffentlichen Schlüssel ein und lädt ihn auf Amazon CloudFront hoch.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.CreatePublicKeyResponse;
import software.amazon.awssdk.utils.IoUtils;

import java.io.IOException;
import java.io.InputStream;
import java.util.UUID;

public class CreatePublicKey {
    private static final Logger logger =
 LoggerFactory.getLogger(CreatePublicKey.class);

    public static String createPublicKey(CloudFrontClient cloudFrontClient,
 String publicKeyFileName) {
        try (InputStream is =
 CreatePublicKey.class.getClassLoader().getResourceAsStream(publicKeyFileName)) {
            String publicKeyString = IoUtils.toUtf8String(is);
            CreatePublicKeyResponse createPublicKeyResponse = cloudFrontClient
                    .createPublicKey(b -> b.publicKeyConfig(c -> c
                            .name("JavaCreatedPublicKey" + UUID.randomUUID())
                            .encodedKey(publicKeyString)
```

```
                                .callerReference(UUID.randomUUID().toString())));
            String createdPublicKeyId = createPublicKeyResponse.publicKey().id();
            logger.info("Public key created with id: [{}]", createdPublicKeyId);
            return createdPublicKeyId;

        } catch (IOException e) {
            throw new RuntimeException(e);
        }
    }
}
```

- Einzelheiten zur API finden Sie CreatePublicKeyin der AWS SDK for Java 2.x API-Referenz.

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung **`DeleteDistribution`** mit einem AWS SDK oder CLI

Die folgenden Code-Beispiele zeigen, wie `DeleteDistribution` verwendet wird.

CLI

AWS CLI

Um eine CloudFront Distribution zu löschen

Im folgenden Beispiel wird die CloudFront Distribution mit der ID `EDFDVBD6EXAMPLE` gelöscht. Bevor Sie eine Distribution löschen können, müssen Sie sie deaktivieren. Verwenden Sie den Befehl update-distribution, um eine Distribution zu deaktivieren. Weitere Informationen finden Sie in den Beispielen für die Update-Distribution.

Wenn eine Distribution deaktiviert ist, können Sie sie löschen. Um eine Distribution zu löschen, müssen Sie die `--if-match` Option zum Bereitstellen der Distribution verwendenETag. Um die abzurufenETag, verwenden Sie den get-distribution-config Befehl get-distribution or.

```
aws cloudfront delete-distribution \
    --id EDFDVBD6EXAMPLE \
    --if-match E2QWRUHEXAMPLE
```

Wenn dieser Befehl erfolgreich ist, hat er keine Ausgabe.

- Einzelheiten zur API finden Sie DeleteDistributionin der AWS CLI Befehlsreferenz.

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

Im folgenden Codebeispiel wird eine Distribution auf „Deaktiviert“ aktualisiert, es wird ein Kellner verwendet, der auf die Bereitstellung der Änderung wartet und dann die Verteilung löscht.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.core.internal.waiters.ResponseOrException;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import
 software.amazon.awssdk.services.cloudfront.model.DeleteDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.DistributionConfig;
import software.amazon.awssdk.services.cloudfront.model.GetDistributionResponse;
import software.amazon.awssdk.services.cloudfront.waiters.CloudFrontWaiter;

public class DeleteDistribution {
        private static final Logger logger =
 LoggerFactory.getLogger(DeleteDistribution.class);

        public static void deleteDistribution(final CloudFrontClient
 cloudFrontClient, final String distributionId) {
                // First, disable the distribution by updating it.
                GetDistributionResponse response =
 cloudFrontClient.getDistribution(b -> b
                                .id(distributionId));
                String etag = response.eTag();
                DistributionConfig distConfig =
 response.distribution().distributionConfig();
```

```
                cloudFrontClient.updateDistribution(builder -> builder
                                .id(distributionId)
                                .distributionConfig(builder1 -> builder1

.cacheBehaviors(distConfig.cacheBehaviors())

.defaultCacheBehavior(distConfig.defaultCacheBehavior())
                                                .enabled(false)
                                                .origins(distConfig.origins())
                                                .comment(distConfig.comment())

.callerReference(distConfig.callerReference())

.defaultCacheBehavior(distConfig.defaultCacheBehavior())

.priceClass(distConfig.priceClass())
                                                .aliases(distConfig.aliases())
                                                .logging(distConfig.logging())

.defaultRootObject(distConfig.defaultRootObject())

.customErrorResponses(distConfig.customErrorResponses())

.httpVersion(distConfig.httpVersion())

.isIPV6Enabled(distConfig.isIPV6Enabled())

.restrictions(distConfig.restrictions())

.viewerCertificate(distConfig.viewerCertificate())
                                                .webACLId(distConfig.webACLId())

.originGroups(distConfig.originGroups()))
                                .ifMatch(etag));

                logger.info("Distribution [{}] is DISABLED, waiting for
deployment before deleting ...",
                                distributionId);
                GetDistributionResponse distributionResponse;
                try (CloudFrontWaiter cfWaiter =
CloudFrontWaiter.builder().client(cloudFrontClient).build()) {
                        ResponseOrException<GetDistributionResponse>
responseOrException = cfWaiter
```

```
                                        .waitUntilDistributionDeployed(builder ->
 builder.id(distributionId)).matched();
                        distributionResponse = responseOrException.response()
                                        .orElseThrow(() -> new
 RuntimeException("Could not disable distribution"));
                }

                DeleteDistributionResponse deleteDistributionResponse =
 cloudFrontClient
                                .deleteDistribution(builder -> builder
                                                .id(distributionId)

 .ifMatch(distributionResponse.eTag()));
                if (deleteDistributionResponse.sdkHttpResponse().isSuccessful())
 {
                        logger.info("Distribution [{}] DELETED", distributionId);
                }
        }
}
```

- Einzelheiten zur API finden Sie unter DeleteDistributionAPI-Referenz.AWS SDK for Java 2.x

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung von **`GetCloudFrontOriginAccessIdentity`** mit einer CLI

Die folgenden Code-Beispiele zeigen, wie `GetCloudFrontOriginAccessIdentity` verwendet wird.

CLI

AWS CLI

Um eine Zugriffsidentität für den CloudFront Ursprung zu erhalten

Im folgenden Beispiel wird die ursprüngliche CloudFront Zugriffsidentität (OAI) mit der ID abgerufen`E74FTE3AEXAMPLE`, einschließlich ihrer `ETag` und der zugehörigen kanonischen S3-ID. Die OAI-ID wird in der Ausgabe der Befehle -access-identity und create-cloud-front-origin -access-identities zurückgegeben. list-cloud-front-origin

```
aws cloudfront get-cloud-front-origin-access-identity --id E74FTE3AEXAMPLE
```

Ausgabe:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "CloudFrontOriginAccessIdentity": {
        "Id": "E74FTE3AEXAMPLE",
        "S3CanonicalUserId":
 "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
        "CloudFrontOriginAccessIdentityConfig": {
            "CallerReference": "cli-example",
            "Comment": "Example OAI"
        }
    }
}
```

- Einzelheiten zur API finden Sie in der Befehlsreferenz. GetCloudFrontOriginAccessIdentityAWS CLI

PowerShell

Tools für PowerShell V4

Beispiel 1: In diesem Beispiel wird eine bestimmte CloudFront Amazon-Ursprungszugriffsidentität zurückgegeben, die durch den Parameter -Id angegeben wird. Der Parameter -Id ist zwar nicht erforderlich, aber wenn Sie ihn nicht angeben, werden keine Ergebnisse zurückgegeben.

```
Get-CFCloudFrontOriginAccessIdentity -Id E3XXXXXXXXXXRT
```

Ausgabe:

```
     CloudFrontOriginAccessIdentityConfig    Id
    S3CanonicalUserId
     ------------------------------------    --
    -----------------
     Amazon.CloudFront.Model.CloudFrontOr... E3XXXXXXXXXXRT
    4b6e...
```

- Einzelheiten zur API finden Sie unter GetCloudFrontOriginAccessIdentity AWS -Tools für PowerShellCmdlet-Referenz (V4).

Tools für V5 PowerShell

Beispiel 1: In diesem Beispiel wird eine bestimmte CloudFront Amazon-Ursprungszugriffsidentität zurückgegeben, die durch den Parameter -Id angegeben wird. Der Parameter -Id ist zwar nicht erforderlich, aber wenn Sie ihn nicht angeben, werden keine Ergebnisse zurückgegeben.

```
Get-CFCloudFrontOriginAccessIdentity -Id E3XXXXXXXXXXRT
```

Ausgabe:

```
 CloudFrontOriginAccessIdentityConfig    Id
S3CanonicalUserId
 ------------------------------------    --
----------------
 Amazon.CloudFront.Model.CloudFrontOr... E3XXXXXXXXXXRT
4b6e...
```

- Einzelheiten zur API finden Sie unter GetCloudFrontOriginAccessIdentity AWS -Tools für PowerShellCmdlet-Referenz (V5).

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung von **`GetCloudFrontOriginAccessIdentityConfig`** mit einer CLI

Die folgenden Code-Beispiele zeigen, wie `GetCloudFrontOriginAccessIdentityConfig` verwendet wird.

CLI

AWS CLI

Um eine Konfiguration der CloudFront Origin-Zugriffsidentität zu erhalten

Im folgenden Beispiel werden Metadaten zur ursprünglichen CloudFront Zugriffsidentität (OAI) mit der ID abgerufen`E74FTE3AEXAMPLE`, einschließlich ihrer`ETag`. Die OAI-ID wird

in der Ausgabe der Befehle -access-identity und create-cloud-front-origin -access-identities zurückgegeben. list-cloud-front-origin

```
aws cloudfront get-cloud-front-origin-access-identity-config --id E74FTE3AEXAMPLE
```

Ausgabe:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "CloudFrontOriginAccessIdentityConfig": {
        "CallerReference": "cli-example",
        "Comment": "Example OAI"
    }
}
```

- Einzelheiten zur API finden Sie in der Befehlsreferenz. GetCloudFrontOriginAccessIdentityConfigAWS CLI

PowerShell

Tools für PowerShell V4

Beispiel 1: In diesem Beispiel werden Konfigurationsinformationen zu einer einzelnen CloudFront Amazon-Ursprungszugriffsidentität zurückgegeben, die durch den Parameter -Id angegeben wird. Fehler treten auf, wenn kein -Id-Parameter angegeben ist.

```
Get-CFCloudFrontOriginAccessIdentityConfig -Id E3XXXXXXXXXXRT
```

Ausgabe:

```
     CallerReference                                                Comment
     ---------------                                                -------
     mycallerreference: 2/1/2011 1:16:32 PM                         Caller
reference: 2/1/2011 1:16:32 PM
```

- Einzelheiten zur API finden Sie unter GetCloudFrontOriginAccessIdentityConfig AWS -Tools für PowerShellCmdlet-Referenz (V4).

Tools für V5 PowerShell

Beispiel 1: In diesem Beispiel werden Konfigurationsinformationen zu einer einzelnen CloudFront Amazon-Ursprungszugriffsidentität zurückgegeben, die durch den Parameter -Id angegeben wird. Fehler treten auf, wenn kein -Id-Parameter angegeben ist.

```
Get-CFCloudFrontOriginAccessIdentityConfig -Id E3XXXXXXXXXXRT
```

Ausgabe:

```
     CallerReference                                                Comment
     ---------------                                                -------
     mycallerreference: 2/1/2011 1:16:32 PM                         Caller
reference: 2/1/2011 1:16:32 PM
```

- Einzelheiten zur API finden Sie unter GetCloudFrontOriginAccessIdentityConfig AWS -Tools für PowerShellCmdlet-Referenz (V5).

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung von **`GetDistribution`** mit einer CLI

Die folgenden Code-Beispiele zeigen, wie `GetDistribution` verwendet wird.

CLI

AWS CLI

Um eine CloudFront Distribution zu erhalten

Im folgenden `get-distribution` Beispiel wird die CloudFront Distribution mit der ID abgerufen`EDFDVBD6EXAMPLE`, einschließlich ihrer`ETag`. Die Distributions-ID wird in den Befehlen create-distribution und list-distributions zurückgegeben.

```
aws cloudfront get-distribution \
    --id EDFDVBD6EXAMPLE
```

Ausgabe:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "Distribution": {
        "Id": "EDFDVBD6EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
        "Status": "Deployed",
        "LastModifiedTime": "2019-12-04T23:35:41.433Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                        "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        }
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-
example",
                "ForwardedValues": {
                    "QueryString": false,
```

```
                    "Cookies": {
                        "Forward": "none"
                    },
                    "Headers": {
                        "Quantity": 0
                    },
                    "QueryStringCacheKeys": {
                        "Quantity": 0
                    }
                },
                "TrustedSigners": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "MinTTL": 0,
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "SmoothStreaming": false,
                "DefaultTTL": 86400,
                "MaxTTL": 31536000,
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": ""
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
```

```
            },
            "Comment": "",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "PriceClass": "PriceClass_All",
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "IsIPV6Enabled": true
        }
    }
}
```

- Einzelheiten zur API finden Sie in der Befehlsreferenz. GetDistributionAWS CLI

PowerShell

Tools für PowerShell V4

Beispiel 1: Ruft die Informationen für eine bestimmte Distribution ab.

```
Get-CFDistribution -Id EXAMPLE0000ID
```

- Einzelheiten zur API finden Sie unter GetDistribution AWS -Tools für PowerShellCmdlet-Referenz (V4).

Tools für V5 PowerShell

Beispiel 1: Ruft die Informationen für eine bestimmte Distribution ab.

```
Get-CFDistribution -Id EXAMPLE0000ID
```

- Einzelheiten zur API finden Sie unter GetDistribution AWS -Tools für PowerShellCmdlet-Referenz (V5).

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung **`GetDistributionConfig`** mit einem AWS SDK oder CLI

Die folgenden Code-Beispiele zeigen, wie `GetDistributionConfig` verwendet wird.

CLI

AWS CLI

Um eine CloudFront Distributionskonfiguration zu erhalten

Im folgenden Beispiel werden Metadaten über die CloudFront Distribution mit der ID abgerufen`EDFDVBD6EXAMPLE`, einschließlich ihrer`ETag`. Die Distributions-ID wird in den Befehlen create-distribution und list-distributions zurückgegeben.

```
aws cloudfront get-distribution-config \
    --id EDFDVBD6EXAMPLE
```

Ausgabe:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "DistributionConfig": {
        "CallerReference": "cli-example",
        "Aliases": {
            "Quantity": 0
        },
        "DefaultRootObject": "index.html",
        "Origins": {
            "Quantity": 1,
```

```
            "Items": [
                {
                    "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                    "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                    "OriginPath": "",
                    "CustomHeaders": {
                        "Quantity": 0
                    },
                    "S3OriginConfig": {
                        "OriginAccessIdentity": ""
                    }
                }
            ]
        },
        "OriginGroups": {
            "Quantity": 0
        },
        "DefaultCacheBehavior": {
            "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
            "ForwardedValues": {
                "QueryString": false,
                "Cookies": {
                    "Forward": "none"
                },
                "Headers": {
                    "Quantity": 0
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            },
            "TrustedSigners": {
                "Enabled": false,
                "Quantity": 0
            },
            "ViewerProtocolPolicy": "allow-all",
            "MinTTL": 0,
            "AllowedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ],
                "CachedMethods": {
```

```
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ]
            }
        },
        "SmoothStreaming": false,
        "DefaultTTL": 86400,
        "MaxTTL": 31536000,
        "Compress": false,
        "LambdaFunctionAssociations": {
            "Quantity": 0
        },
        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
```

```
        "IsIPV6Enabled": true
    }
}
```

- Einzelheiten zur API finden Sie in der Befehlsreferenz. GetDistributionConfigAWS CLI

PowerShell

Tools für PowerShell V4

Beispiel 1: Ruft die Konfiguration für eine bestimmte Distribution ab.

```
Get-CFDistributionConfig -Id EXAMPLE0000ID
```

- Einzelheiten zur API finden Sie unter GetDistributionConfig AWS -Tools für PowerShellCmdlet-Referenz (V4).

Tools für V5 PowerShell

Beispiel 1: Ruft die Konfiguration für eine bestimmte Distribution ab.

```
Get-CFDistributionConfig -Id EXAMPLE0000ID
```

- Einzelheiten zur API finden Sie unter GetDistributionConfig AWS -Tools für PowerShellCmdlet-Referenz (V5).

Python

SDK für Python (Boto3)

> **Note**
>
> Es gibt noch mehr dazu. GitHub Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

```
class CloudFrontWrapper:
    """Encapsulates Amazon CloudFront operations."""
```

```
    def __init__(self, cloudfront_client):
        """
        :param cloudfront_client: A Boto3 CloudFront client
        """
        self.cloudfront_client = cloudfront_client


    def update_distribution(self):
        distribution_id = input(
            "This script updates the comment for a CloudFront distribution.\n"
            "Enter a CloudFront distribution ID: "
        )

        distribution_config_response =
self.cloudfront_client.get_distribution_config(
            Id=distribution_id
        )
        distribution_config = distribution_config_response["DistributionConfig"]
        distribution_etag = distribution_config_response["ETag"]

        distribution_config["Comment"] = input(
            f"\nThe current comment for distribution {distribution_id} is "
            f"'{distribution_config['Comment']}'.\n"
            f"Enter a new comment: "
        )
        self.cloudfront_client.update_distribution(
            DistributionConfig=distribution_config,
            Id=distribution_id,
            IfMatch=distribution_etag,
        )
        print("Done!")
```

- Einzelheiten zur API finden Sie GetDistributionConfigin AWS SDK for Python (Boto3) API Reference.

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung von **`ListCloudFrontOriginAccessIdentities`** mit einer CLI

Die folgenden Code-Beispiele zeigen, wie `ListCloudFrontOriginAccessIdentities` verwendet wird.

CLI

AWS CLI

Um die ursprünglichen CloudFront Zugriffsidentitäten aufzulisten

Im folgenden Beispiel wird eine Liste der ursprünglichen CloudFront Zugriffsidentitäten (OAIs) in Ihrem AWS Konto abgerufen:

```
aws cloudfront list-cloud-front-origin-access-identities
```

Ausgabe:

```
{
    "CloudFrontOriginAccessIdentityList": {
        "Items": [
            {
                "Id": "E74FTE3AEXAMPLE",
                "S3CanonicalUserId":
 "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
                "Comment": "Example OAI"
            },
            {
                "Id": "EH1HDMBEXAMPLE",
                "S3CanonicalUserId":
 "1489f6f2e6faacaae7ff64c4c3e6956c24f78788abfc1718c3527c263bf7a17EXAMPLE",
                "Comment": "Test OAI"
            },
            {
                "Id": "E2X2C9TEXAMPLE",
                "S3CanonicalUserId":
 "cbfeebb915a64749f9be546a45b3fcfd3a31c779673c13c4dd460911ae402c2EXAMPLE",
                "Comment": "Example OAI #2"
            }
        ]
    }
}
```

- Einzelheiten zur API finden Sie ListCloudFrontOriginAccessIdentitiesin der AWS CLI Befehlsreferenz.

PowerShell

Tools für PowerShell V4

Beispiel 1: In diesem Beispiel wird eine Liste der CloudFront Amazon-Origin-Zugriffsidentitäten zurückgegeben. Da der MaxItem Parameter - den Wert 2 angibt, enthalten die Ergebnisse zwei Identitäten.

```
Get-CFCloudFrontOriginAccessIdentityList -MaxItem 2
```

Ausgabe:

```
IsTruncated : True
Items       : {E326XXXXXXXXXT, E1YWXXXXXXX9B}
Marker      :
MaxItems    : 2
NextMarker  : E1YXXXXXXXXX9B
Quantity    : 2
```

- Einzelheiten zur API finden Sie unter ListCloudFrontOriginAccessIdentities AWS -Tools für PowerShellCmdlet-Referenz (V4).

Tools für V5 PowerShell

Beispiel 1: In diesem Beispiel wird eine Liste der CloudFront Amazon-Origin-Zugriffsidentitäten zurückgegeben. Da der MaxItem Parameter - den Wert 2 angibt, enthalten die Ergebnisse zwei Identitäten.

```
Get-CFCloudFrontOriginAccessIdentityList -MaxItem 2
```

Ausgabe:

```
IsTruncated : True
Items       : {E326XXXXXXXXXT, E1YWXXXXXXX9B}
Marker      :
MaxItems    : 2
NextMarker  : E1YXXXXXXXXX9B
Quantity    : 2
```

- Einzelheiten zur API finden Sie unter ListCloudFrontOriginAccessIdentities AWS -Tools für PowerShellCmdlet-Referenz (V5).

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung **`ListDistributions`** mit einem AWS SDK oder CLI

Die folgenden Code-Beispiele zeigen, wie `ListDistributions` verwendet wird.

CLI

AWS CLI

Um CloudFront Distributionen aufzulisten

Im folgenden Beispiel wird eine Liste der CloudFront Verteilungen in Ihrem AWS Konto abgerufen.

```
aws cloudfront list-distributions
```

Ausgabe:

```
{
    "DistributionList": {
        "Items": [
            {
                "Id": "E23YS8OEXAMPLE",
                "ARN": "arn:aws:cloudfront::123456789012:distribution/
E23YS8OEXAMPLE",
                "Status": "Deployed",
                "LastModifiedTime": "2024-08-05T18:23:40.375000+00:00",
                "DomainName": "abcdefgh12ijk.cloudfront.net",
                "Aliases": {
                    "Quantity": 0
                },
                "Origins": {
                    "Quantity": 1,
                    "Items": [
                        {
```

```
                            "Id": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                            "DomainName": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                            "OriginPath": "",
                            "CustomHeaders": {
                                "Quantity": 0
                            },
                            "S3OriginConfig": {
                                "OriginAccessIdentity": ""
                            },
                            "ConnectionAttempts": 3,
                            "ConnectionTimeout": 10,
                            "OriginShield": {
                                "Enabled": false
                            },
                            "OriginAccessControlId": "EIAP8PEXAMPLE"
                        }
                    ]
                },
                "OriginGroups": {
                    "Quantity": 0
                },
                "DefaultCacheBehavior": {
                    "TargetOriginId": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                    "TrustedSigners": {
                        "Enabled": false,
                        "Quantity": 0
                    },
                    "TrustedKeyGroups": {
                        "Enabled": false,
                        "Quantity": 0
                    },
                    "ViewerProtocolPolicy": "allow-all",
                    "AllowedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ],
                        "CachedMethods": {
                            "Quantity": 2,
                            "Items": [
```

```
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "SmoothStreaming": false,
                "Compress": true,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": "",
                "CachePolicyId": "658327ea-f89d-4fab-a63d-7e886EXAMPLE"
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "",
            "PriceClass": "PriceClass_All",
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "SSLSupportMethod": "vip",
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "HTTP2",
            "IsIPV6Enabled": true,
            "Staging": false
        }
    ]
}
```

```
}
```

- Einzelheiten zur API finden Sie ListDistributionsin der AWS CLI Befehlsreferenz.

PowerShell

Tools für PowerShell V4

Beispiel 1: Gibt Verteilungen zurück.

```
Get-CFDistributionList
```

- Einzelheiten zur API finden Sie unter ListDistributions AWS -Tools für PowerShellCmdlet-Referenz (V4).

Tools für V5 PowerShell

Beispiel 1: Gibt Verteilungen zurück.

```
Get-CFDistributionList
```

- Einzelheiten zur API finden Sie unter ListDistributions AWS -Tools für PowerShellCmdlet-Referenz (V5).

Python

SDK für Python (Boto3)

> **Note**
>
> Es gibt noch mehr dazu. GitHub Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

```
class CloudFrontWrapper:
    """Encapsulates Amazon CloudFront operations."""

    def __init__(self, cloudfront_client):
        """
```

```
        :param cloudfront_client: A Boto3 CloudFront client
        """
        self.cloudfront_client = cloudfront_client


    def list_distributions(self):
        print("CloudFront distributions:\n")
        distributions = self.cloudfront_client.list_distributions()
        if distributions["DistributionList"]["Quantity"] > 0:
            for distribution in distributions["DistributionList"]["Items"]:
                print(f"Domain: {distribution['DomainName']}")
                print(f"Distribution Id: {distribution['Id']}")
                print(
                    f"Certificate Source: "
                    f"{distribution['ViewerCertificate']['CertificateSource']}"
                )
                if distribution["ViewerCertificate"]["CertificateSource"] ==
 "acm":
                    print(
                        f"Certificate: {distribution['ViewerCertificate']
['Certificate']}"
                    )
                print("")
        else:
            print("No CloudFront distributions detected.")
```

- Einzelheiten zur API finden Sie ListDistributionsin AWS SDK for Python (Boto3) API Reference.

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwendung **`UpdateDistribution`** mit einem AWS SDK oder CLI

Die folgenden Code-Beispiele zeigen, wie `UpdateDistribution` verwendet wird.

CLI

AWS CLI

Beispiel 1: Um das Standard-Root-Objekt einer CloudFront Distribution zu aktualisieren

Im folgenden Beispiel wird das Standard-Stammobjekt `index.html` für die CloudFront Distribution mit der ID aktualisiert`EDFDVBD6EXAMPLE`.

```
aws cloudfront update-distribution \
    --id EDFDVBD6EXAMPLE \
    --default-root-object index.html
```

Ausgabe:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "Distribution": {
        "Id": "EDFDVBD6EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-12-06T18:55:39.870Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "6b10378d-49be-4c4b-a642-419ccaf8f3b5",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "example-website",
                        "DomainName": "www.example.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
```

```
                    },
                    "CustomOriginConfig": {
                        "HTTPPort": 80,
                        "HTTPSPort": 443,
                        "OriginProtocolPolicy": "match-viewer",
                        "OriginSslProtocols": {
                            "Quantity": 2,
                            "Items": [
                                "SSLv3",
                                "TLSv1"
                            ]
                        },
                        "OriginReadTimeout": 30,
                        "OriginKeepaliveTimeout": 5
                    }
                }
            ]
        },
        "OriginGroups": {
            "Quantity": 0
        },
        "DefaultCacheBehavior": {
            "TargetOriginId": "example-website",
            "ForwardedValues": {
                "QueryString": false,
                "Cookies": {
                    "Forward": "none"
                },
                "Headers": {
                    "Quantity": 1,
                    "Items": [
                        "*"
                    ]
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            },
            "TrustedSigners": {
                "Enabled": false,
                "Quantity": 0
            },
            "ViewerProtocolPolicy": "allow-all",
            "MinTTL": 0,
```

```
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "SmoothStreaming": false,
                "DefaultTTL": 86400,
                "MaxTTL": 31536000,
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": ""
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "PriceClass": "PriceClass_All",
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
```

```
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http1.1",
            "IsIPV6Enabled": true
        }
    }
}
```

Beispiel 2: Um eine CloudFront Distribution zu aktualisieren

Im folgenden Beispiel wird die CloudFront Distribution mit der ID deaktiviert, `EMLARXS9EXAMPLE` indem die Verteilungskonfiguration in einer JSON-Datei mit dem Namen `dist-config-disable.json` bereitgestellt wird. Um eine Distribution zu aktualisieren, müssen Sie die `--if-match` Option zur Bereitstellung der Distribution verwenden. `ETag` Um die abzurufen`ETag`, verwenden Sie den get-distribution-config Befehl get-distribution or. Beachten Sie, dass das `Enabled` Feld `false` in der JSON-Datei auf gesetzt ist.

Nachdem Sie das folgende Beispiel verwendet haben, um eine Distribution zu deaktivieren, können Sie sie mit dem Befehl delete-distribution löschen.

```
aws cloudfront update-distribution \
    --id EMLARXS9EXAMPLE \
    --if-match E2QWRUHEXAMPLE \
    --distribution-config file://dist-config-disable.json
```

Inhalt von `dist-config-disable.json`:

```
{
    "CallerReference": "cli-1574382155-496510",
    "Aliases": {
        "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
                "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-1574382155-273939",
```

```
                "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                "OriginPath": "",
                "CustomHeaders": {
                    "Quantity": 0
                },
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                }
            }
        ]
    },
    "OriginGroups": {
        "Quantity": 0
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "amzn-s3-demo-
bucket.s3.amazonaws.com-1574382155-273939",
        "ForwardedValues": {
            "QueryString": false,
            "Cookies": {
                "Forward": "none"
            },
            "Headers": {
                "Quantity": 0
            },
            "QueryStringCacheKeys": {
                "Quantity": 0
            }
        },
        "TrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "ViewerProtocolPolicy": "allow-all",
        "MinTTL": 0,
        "AllowedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ],
            "CachedMethods": {
                "Quantity": 2,
                "Items": [
```

```
                    "HEAD",
                    "GET"
                ]
            }
        },
        "SmoothStreaming": false,
        "DefaultTTL": 86400,
        "MaxTTL": 31536000,
        "Compress": false,
        "LambdaFunctionAssociations": {
            "Quantity": 0
        },
        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": false,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
```

```
}
```

Ausgabe:

```
{
    "ETag": "E9LHASXEXAMPLE",
    "Distribution": {
        "Id": "EMLARXS9EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-12-06T18:32:35.553Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-1574382155-496510",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-
bucket.s3.amazonaws.com-1574382155-273939",
                        "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        }
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
```

```
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-
bucket.s3.amazonaws.com-1574382155-273939",
                "ForwardedValues": {
                    "QueryString": false,
                    "Cookies": {
                        "Forward": "none"
                    },
                    "Headers": {
                        "Quantity": 0
                    },
                    "QueryStringCacheKeys": {
                        "Quantity": 0
                    }
                },
                "TrustedSigners": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "MinTTL": 0,
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "SmoothStreaming": false,
                "DefaultTTL": 86400,
                "MaxTTL": 31536000,
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": ""
            },
```

```
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "PriceClass": "PriceClass_All",
            "Enabled": false,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "IsIPV6Enabled": true
        }
    }
}
```

- Einzelheiten zur API finden Sie unter Befehlsreferenz UpdateDistribution.AWS CLI

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.GetDistributionRequest;
import software.amazon.awssdk.services.cloudfront.model.GetDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.Distribution;
import software.amazon.awssdk.services.cloudfront.model.DistributionConfig;
import
 software.amazon.awssdk.services.cloudfront.model.UpdateDistributionRequest;
import software.amazon.awssdk.services.cloudfront.model.CloudFrontException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class ModifyDistribution {
    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <id>\s

                Where:
                    id - the id value of the distribution.\s
                """;

        if (args.length != 1) {
            System.out.println(usage);
```

```
            System.exit(1);
        }

        String id = args[0];
        CloudFrontClient cloudFrontClient = CloudFrontClient.builder()
                .region(Region.AWS_GLOBAL)
                .build();

        modDistribution(cloudFrontClient, id);
        cloudFrontClient.close();
    }

    public static void modDistribution(CloudFrontClient cloudFrontClient, String
idVal) {
        try {
            // Get the Distribution to modify.
            GetDistributionRequest disRequest = GetDistributionRequest.builder()
                    .id(idVal)
                    .build();

            GetDistributionResponse response =
cloudFrontClient.getDistribution(disRequest);
            Distribution disObject = response.distribution();
            DistributionConfig config = disObject.distributionConfig();

            // Create a new DistributionConfig object and add new values to
comment and
            // aliases
            DistributionConfig config1 = DistributionConfig.builder()
                    .aliases(config.aliases()) // You can pass in new values here
                    .comment("New Comment")
                    .cacheBehaviors(config.cacheBehaviors())
                    .priceClass(config.priceClass())
                    .defaultCacheBehavior(config.defaultCacheBehavior())
                    .enabled(config.enabled())
                    .callerReference(config.callerReference())
                    .logging(config.logging())
                    .originGroups(config.originGroups())
                    .origins(config.origins())
                    .restrictions(config.restrictions())
                    .defaultRootObject(config.defaultRootObject())
                    .webACLId(config.webACLId())
                    .httpVersion(config.httpVersion())
                    .viewerCertificate(config.viewerCertificate())
```

```
                    .customErrorResponses(config.customErrorResponses())
                    .build();

            UpdateDistributionRequest updateDistributionRequest =
 UpdateDistributionRequest.builder()
                    .distributionConfig(config1)
                    .id(disObject.id())
                    .ifMatch(response.eTag())
                    .build();

            cloudFrontClient.updateDistribution(updateDistributionRequest);

        } catch (CloudFrontException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
}
```

- Einzelheiten zur API finden Sie UpdateDistributionin der AWS SDK for Java 2.x API-Referenz.

Python

SDK für Python (Boto3)

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

```
class CloudFrontWrapper:
    """Encapsulates Amazon CloudFront operations."""

    def __init__(self, cloudfront_client):
        """
        :param cloudfront_client: A Boto3 CloudFront client
        """
        self.cloudfront_client = cloudfront_client
```

```
    def update_distribution(self):
        distribution_id = input(
            "This script updates the comment for a CloudFront distribution.\n"
            "Enter a CloudFront distribution ID: "
        )

        distribution_config_response =
self.cloudfront_client.get_distribution_config(
            Id=distribution_id
        )
        distribution_config = distribution_config_response["DistributionConfig"]
        distribution_etag = distribution_config_response["ETag"]

        distribution_config["Comment"] = input(
            f"\nThe current comment for distribution {distribution_id} is "
            f"'{distribution_config['Comment']}'.\n"
            f"Enter a new comment: "
        )
        self.cloudfront_client.update_distribution(
            DistributionConfig=distribution_config,
            Id=distribution_id,
            IfMatch=distribution_etag,
        )
        print("Done!")
```

- Einzelheiten zur API finden Sie UpdateDistributionin AWS SDK for Python (Boto3) API Reference.

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Szenarien für die CloudFront Verwendung AWS SDKs

Die folgenden Codebeispiele zeigen Ihnen, wie Sie gängige Szenarien in CloudFront with implementieren AWS SDKs. Diese Szenarien zeigen Ihnen, wie Sie bestimmte Aufgaben erledigen

können, indem Sie mehrere Funktionen innerhalb CloudFront oder in Kombination mit anderen aufrufen AWS-Services. Jedes Szenario enthält einen Link zum vollständigen Quell-Code, wo Sie Anweisungen zum Einrichten und Ausführen des Codes finden.

Szenarien zielen auf eine mittlere Erfahrungsebene ab, um Ihnen zu helfen, Service-Aktionen im Kontext zu verstehen.

Beispiele

- AWS SDK für SaaS-Manager-Ressourcen erstellen
- Löschen Sie CloudFront Signaturressourcen mithilfe des AWS SDK
- Erstellen Sie signierte Cookies URLs und Cookies mithilfe eines AWS SDK

AWS SDK für SaaS-Manager-Ressourcen erstellen

Das folgende Codebeispiel zeigt, wie Sie eine Mehrmandantenverteilung und einen Distributionsmandanten mit verschiedenen Konfigurationen erstellen.

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

Das folgende Beispiel zeigt, wie eine Mehrmandantenverteilung mit Parametern und Platzhalterzertifikat erstellt wird.

```
import software.amazon.awssdk.core.internal.waiters.ResponseOrException;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.ConnectionMode;
import
 software.amazon.awssdk.services.cloudfront.model.CreateDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.Distribution;
import software.amazon.awssdk.services.cloudfront.model.GetDistributionResponse;
import software.amazon.awssdk.services.cloudfront.model.HttpVersion;
```

```
import software.amazon.awssdk.services.cloudfront.model.Method;
import software.amazon.awssdk.services.cloudfront.model.SSLSupportMethod;
import software.amazon.awssdk.services.cloudfront.model.ViewerProtocolPolicy;
import software.amazon.awssdk.services.cloudfront.waiters.CloudFrontWaiter;
import software.amazon.awssdk.services.s3.S3Client;

import java.time.Instant;

public class CreateMultiTenantDistribution {
    public static Distribution
 CreateMultiTenantDistributionWithCert(CloudFrontClient cloudFrontClient,
                                                                    S3Client
 s3Client,
                                                                    final String
 bucketName,
                                                                    final String
 certificateArn) {
        // fetch the origin info if necessary
        final String region = s3Client.headBucket(b ->
 b.bucket(bucketName)).sdkHttpResponse().headers()
                .get("x-amz-bucket-region").get(0);
        final String originDomain = bucketName + ".s3." + region +
 ".amazonaws.com";
        String originId = originDomain; // Use the originDomain value for the
 originId.

        CreateDistributionResponse createDistResponse =
 cloudFrontClient.createDistribution(builder -> builder
                .distributionConfig(b1 -> b1
                        .httpVersion(HttpVersion.HTTP2)
                        .enabled(true)
                        .comment("Template Distribution with cert built with
 java")
                        .connectionMode(ConnectionMode.TENANT_ONLY)
                        .callerReference(Instant.now().toString())
                        .viewerCertificate(certBuilder -> certBuilder
                                .acmCertificateArn(certificateArn)
                                .sslSupportMethod(SSLSupportMethod.SNI_ONLY))
                        .origins(b2 -> b2
                                .quantity(1)
                                .items(b3 -> b3
                                        .domainName(originDomain)
                                        .id(originId)
                                        .originPath("/{{tenantName}}")
```

```
                                        .s3OriginConfig(builder4 -> builder4
                                                .originAccessIdentity(
                                                        "")))))
                        .tenantConfig(b5 -> b5
                                .parameterDefinitions(b6 -> b6
                                        .name("tenantName")
                                        .definition(b7 -> b7
                                                .stringSchema(b8 -> b8
                                                        .comment("tenantName
 value")
                                                        .defaultValue("root")
                                                        .required(false)))))
                        .defaultCacheBehavior(b2 -> b2

 .viewerProtocolPolicy(ViewerProtocolPolicy.ALLOW_ALL)
                                .targetOriginId(originId)
                                .cachePolicyId("658327ea-f89d-4fab-
a63d-7e88639e58f6") // CachingOptimized Policy
                                .allowedMethods(b4 -> b4
                                        .quantity(2)
                                        .items(Method.HEAD, Method.GET)))
                ));

        final Distribution distribution = createDistResponse.distribution();
        try (CloudFrontWaiter cfWaiter =
 CloudFrontWaiter.builder().client(cloudFrontClient).build()) {
            ResponseOrException<GetDistributionResponse> responseOrException =
 cfWaiter
                    .waitUntilDistributionDeployed(builder ->
 builder.id(distribution.id()))
                    .matched();
            responseOrException.response()
                    .orElseThrow(() -> new RuntimeException("Distribution not
 created"));
        }
        return distribution;
    }

    public static Distribution
 CreateMultiTenantDistributionNoCert(CloudFrontClient cloudFrontClient,
                                                                S3Client s3Client,
                                                                final String
 bucketName) {
        // fetch the origin info if necessary
```

```
        final String region = s3Client.headBucket(b ->
 b.bucket(bucketName)).sdkHttpResponse().headers()
                .get("x-amz-bucket-region").get(0);
        final String originDomain = bucketName + ".s3." + region +
 ".amazonaws.com";
        String originId = originDomain; // Use the originDomain value for the
 originId.

        CreateDistributionResponse createDistResponse =
 cloudFrontClient.createDistribution(builder -> builder
                .distributionConfig(b1 -> b1
                        .httpVersion(HttpVersion.HTTP2)
                        .enabled(true)
                        .comment("Template Distribution with cert built with
 java")
                        .connectionMode(ConnectionMode.TENANT_ONLY)
                        .callerReference(Instant.now().toString())
                        .origins(b2 -> b2
                                .quantity(1)
                                .items(b3 -> b3
                                        .domainName(originDomain)
                                        .id(originId)
                                        .originPath("/{{tenantName}}")
                                        .s3OriginConfig(builder4 -> builder4
                                                .originAccessIdentity(
                                                        ""))))
                        .tenantConfig(b5 -> b5
                                .parameterDefinitions(b6 -> b6
                                        .name("tenantName")
                                        .definition(b7 -> b7
                                                .stringSchema(b8 -> b8
                                                        .comment("tenantName
 value")
                                                        .defaultValue("root")
                                                        .required(false)))))
                        .defaultCacheBehavior(b2 -> b2

 .viewerProtocolPolicy(ViewerProtocolPolicy.ALLOW_ALL)
                                .targetOriginId(originId)
                                .cachePolicyId("658327ea-f89d-4fab-
a63d-7e88639e58f6") // CachingOptimized Policy
                                .allowedMethods(b4 -> b4
                                        .quantity(2)
                                        .items(Method.HEAD, Method.GET)))
```

```
                ));

        final Distribution distribution = createDistResponse.distribution();
        try (CloudFrontWaiter cfWaiter =
 CloudFrontWaiter.builder().client(cloudFrontClient).build()) {
            ResponseOrException<GetDistributionResponse> responseOrException =
 cfWaiter
                    .waitUntilDistributionDeployed(builder ->
 builder.id(distribution.id()))
                    .matched();
            responseOrException.response()
                    .orElseThrow(() -> new RuntimeException("Distribution not
 created"));
        }
        return distribution;
    }
}
```

Das folgende Beispiel zeigt, wie ein mit dieser Vorlage verknüpfter Distributionsmandant erstellt wird, einschließlich der Verwendung des oben deklarierten Parameters. Beachten Sie, dass wir hier keine Zertifikatsinformationen hinzufügen müssen, da unsere Domain bereits von der übergeordneten Vorlage abgedeckt ist.

```
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import
 software.amazon.awssdk.services.cloudfront.model.CreateConnectionGroupResponse;
import
 software.amazon.awssdk.services.cloudfront.model.CreateDistributionTenantResponse;
import software.amazon.awssdk.services.cloudfront.model.DistributionTenant;
import
 software.amazon.awssdk.services.cloudfront.model.GetConnectionGroupResponse;
import software.amazon.awssdk.services.cloudfront.model.ValidationTokenHost;
import software.amazon.awssdk.services.route53.Route53Client;
import software.amazon.awssdk.services.route53.model.RRType;

import java.time.Instant;

public class CreateDistributionTenant {

    public static DistributionTenant
 createDistributionTenantNoCert(CloudFrontClient cloudFrontClient,
```

```
                                                                    Route53Client
route53Client,
                                                                    String
distributionId,
                                                                    String
domain,
                                                                    String
hostedZoneId) {
       CreateDistributionTenantResponse createResponse =
cloudFrontClient.createDistributionTenant(builder -> builder
               .distributionId(distributionId)
               .domains(b1 -> b1
                       .domain(domain))
               .parameters(b2 -> b2
                       .name("tenantName")
                       .value("myTenant"))
               .enabled(false)
               .name("no-cert-tenant")
       );

       final DistributionTenant distributionTenant =
createResponse.distributionTenant();

       // Then update the Route53 hosted zone to point your domain at the
distribution tenant
       // We fetch the RoutingEndpoint to point to via the default connection
group that was created for your tenant
       final GetConnectionGroupResponse fetchedConnectionGroup =
cloudFrontClient.getConnectionGroup(builder -> builder
               .identifier(distributionTenant.connectionGroupId()));

       route53Client.changeResourceRecordSets(builder -> builder
               .hostedZoneId(hostedZoneId)
               .changeBatch(b1 -> b1
                       .comment("ChangeBatch comment")
                       .changes(b2 -> b2
                               .resourceRecordSet(b3 -> b3
                                       .name(domain)
                                       .type("CNAME")
                                       .ttl(300L)
                                       .resourceRecords(b4 -> b4

.value(fetchedConnectionGroup.connectionGroup().routingEndpoint())))
                               .action("CREATE"))
```

```
                ));
        return distributionTenant;
    }

}
```

Wenn das Viewer-Zertifikat in der übergeordneten Vorlage weggelassen wurde, müssten Sie stattdessen Zertifikatsinformationen zu den damit verknüpften Mandanten hinzufügen. Das folgende Beispiel zeigt, wie Sie dies mithilfe eines ACM-Zertifikats tun können, das die erforderliche Domäne für den Mandanten abdeckt.

```
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import
 software.amazon.awssdk.services.cloudfront.model.CreateConnectionGroupResponse;
import
 software.amazon.awssdk.services.cloudfront.model.CreateDistributionTenantResponse;
import software.amazon.awssdk.services.cloudfront.model.DistributionTenant;
import
 software.amazon.awssdk.services.cloudfront.model.GetConnectionGroupResponse;
import software.amazon.awssdk.services.cloudfront.model.ValidationTokenHost;
import software.amazon.awssdk.services.route53.Route53Client;
import software.amazon.awssdk.services.route53.model.RRType;

import java.time.Instant;

public class CreateDistributionTenant {

    public static DistributionTenant
 createDistributionTenantWithCert(CloudFrontClient cloudFrontClient,

 Route53Client route53Client,
                                                                    String
 distributionId,
                                                                    String
 domain,
                                                                    String
 hostedZoneId,
                                                                    String
 certificateArn) {
        CreateDistributionTenantResponse createResponse =
 cloudFrontClient.createDistributionTenant(builder -> builder
```

```
                .distributionId(distributionId)
                .domains(b1 -> b1
                        .domain(domain))
                .enabled(false)
                .name("tenant-with-cert")
                .parameters(b2 -> b2
                        .name("tenantName")
                        .value("myTenant"))
                .customizations(b3 -> b3
                        .certificate(b4 -> b4
                                .arn(certificateArn))) // NOTE: Cert must be in
 Us-East-1 and cover the domain provided in this request

        );

        final DistributionTenant distributionTenant =
 createResponse.distributionTenant();

        // Then update the Route53 hosted zone to point your domain at the
 distribution tenant
        // We fetch the RoutingEndpoint to point to via the default connection
 group that was created for your tenant
        final GetConnectionGroupResponse fetchedConnectionGroup =
 cloudFrontClient.getConnectionGroup(builder -> builder
                .identifier(distributionTenant.connectionGroupId()));

        route53Client.changeResourceRecordSets(builder -> builder
                .hostedZoneId(hostedZoneId)
                .changeBatch(b1 -> b1
                        .comment("ChangeBatch comment")
                        .changes(b2 -> b2
                                .resourceRecordSet(b3 -> b3
                                        .name(domain)
                                        .type("CNAME")
                                        .ttl(300L)
                                        .resourceRecords(b4 -> b4

 .value(fetchedConnectionGroup.connectionGroup().routingEndpoint())))
                                .action("CREATE"))
                ));
        return distributionTenant;
    }

}
```

Das folgende Beispiel zeigt, wie dies mit einer von CloudFront -hosted verwalteten Zertifikatsanforderung bewerkstelligt wird. Dies ist ideal, wenn Sie noch keinen Traffic auf Ihre Domain haben. In diesem Fall erstellen wir eine, ConnectionGroup um eine zu generieren RoutingEndpoint. Dann verwenden wir das, RoutingEndpoint um DNS-Einträge zu erstellen, die den Domainbesitz verifizieren und auf die wir verweisen CloudFront. CloudFront stellt dann automatisch ein Token bereit, um den Domainbesitz zu validieren und ein verwaltetes Zertifikat zu erstellen.

```
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import
 software.amazon.awssdk.services.cloudfront.model.CreateConnectionGroupResponse;
import
 software.amazon.awssdk.services.cloudfront.model.CreateDistributionTenantResponse;
import software.amazon.awssdk.services.cloudfront.model.DistributionTenant;
import
 software.amazon.awssdk.services.cloudfront.model.GetConnectionGroupResponse;
import software.amazon.awssdk.services.cloudfront.model.ValidationTokenHost;
import software.amazon.awssdk.services.route53.Route53Client;
import software.amazon.awssdk.services.route53.model.RRType;

import java.time.Instant;

public class CreateDistributionTenant {

    public static DistributionTenant
 createDistributionTenantCfHosted(CloudFrontClient cloudFrontClient,

 Route53Client route53Client,
                                                                    String
 distributionId,
                                                                    String
 domain,
                                                                    String
 hostedZoneId) throws InterruptedException {
       CreateConnectionGroupResponse createConnectionGroupResponse =
 cloudFrontClient.createConnectionGroup(builder -> builder
               .ipv6Enabled(true)
               .name("cf-hosted-connection-group")
               .enabled(true));
```

```
        route53Client.changeResourceRecordSets(builder -> builder
                .hostedZoneId(hostedZoneId)
                .changeBatch(b1 -> b1
                        .comment("cf-hosted domain validation record")
                        .changes(b2 -> b2
                                .resourceRecordSet(b3 -> b3
                                        .name(domain)
                                        .type(RRType.CNAME)
                                        .ttl(300L)
                                        .resourceRecords(b4 -> b4

.value(createConnectionGroupResponse.connectionGroup().routingEndpoint())))
                                .action("CREATE"))
                ));

        // Give the R53 record time to propagate, if it isn't being returned by
servers yet, the following call will fail
        Thread.sleep(60000);

        CreateDistributionTenantResponse createResponse =
cloudFrontClient.createDistributionTenant(builder -> builder
                .distributionId(distributionId)
                .domains(b1 -> b1
                        .domain(domain))

.connectionGroupId(createConnectionGroupResponse.connectionGroup().id())
                .enabled(false)
                .name("cf-hosted-tenant")
                .parameters(b2 -> b2
                        .name("tenantName")
                        .value("myTenant"))
                .managedCertificateRequest(b3 -> b3
                        .validationTokenHost(ValidationTokenHost.CLOUDFRONT)
                )
        );

        return createResponse.distributionTenant();
    }

}
```

Das folgende Beispiel zeigt, wie Sie dies mit einer selbst gehosteten verwalteten Zertifikatsanforderung tun können. Dies ist ideal, wenn Sie Traffic auf Ihre Domain haben und Ausfallzeiten während einer Migration nicht tolerieren können. Am Ende dieses Beispiels wird der Tenant in einem Zustand erstellt, der auf die Domainvalidierung und die DNS-Einrichtung wartet. Folgen Sie den Schritten [hier] (https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/managed-cloudfront-certificates.html#complete-domain-ownership), um die Einrichtung abzuschließen, wenn Sie bereit sind, den Datenverkehr zu migrieren.

```
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import
 software.amazon.awssdk.services.cloudfront.model.CreateConnectionGroupResponse;
import
 software.amazon.awssdk.services.cloudfront.model.CreateDistributionTenantResponse;
import software.amazon.awssdk.services.cloudfront.model.DistributionTenant;
import
 software.amazon.awssdk.services.cloudfront.model.GetConnectionGroupResponse;
import software.amazon.awssdk.services.cloudfront.model.ValidationTokenHost;
import software.amazon.awssdk.services.route53.Route53Client;
import software.amazon.awssdk.services.route53.model.RRType;

import java.time.Instant;

public class CreateDistributionTenant {

    public static DistributionTenant
 createDistributionTenantSelfHosted(CloudFrontClient cloudFrontClient,
                                                                    String
 distributionId,
                                                                    String
 domain) {
        CreateDistributionTenantResponse createResponse =
 cloudFrontClient.createDistributionTenant(builder -> builder
                .distributionId(distributionId)
                .domains(b1 -> b1
                        .domain(domain))
                .parameters(b2 -> b2
                        .name("tenantName")
                        .value("myTenant"))
                .enabled(false)
                .name("self-hosted-tenant")
                .managedCertificateRequest(b3 -> b3
```

```
                        .validationTokenHost(ValidationTokenHost.SELF_HOSTED)
                        .primaryDomainName(domain)
                )
        );

        return createResponse.distributionTenant();
    }

}
```

- API-Details finden Sie in den folgenden Themen der AWS SDK for Java 2.x -API-Referenz.
  - CreateDistribution
  - CreateDistributionTenant

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unter. Verwendung CloudFront mit einem SDK AWS Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Löschen Sie CloudFront Signaturressourcen mithilfe des AWS SDK

Das folgende Codebeispiel zeigt, wie Ressourcen gelöscht werden, die für den Zugriff auf eingeschränkte Inhalte in einem Amazon Simple Storage Service (Amazon S3) -Bucket verwendet werden.

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.services.cloudfront.CloudFrontClient;
import software.amazon.awssdk.services.cloudfront.model.DeleteKeyGroupResponse;
```

```
import
 software.amazon.awssdk.services.cloudfront.model.DeleteOriginAccessControlResponse;
import software.amazon.awssdk.services.cloudfront.model.DeletePublicKeyResponse;
import software.amazon.awssdk.services.cloudfront.model.GetKeyGroupResponse;
import
 software.amazon.awssdk.services.cloudfront.model.GetOriginAccessControlResponse;
import software.amazon.awssdk.services.cloudfront.model.GetPublicKeyResponse;

public class DeleteSigningResources {
    private static final Logger logger =
 LoggerFactory.getLogger(DeleteSigningResources.class);

    public static void deleteOriginAccessControl(final CloudFrontClient
 cloudFrontClient,
            final String originAccessControlId) {
        GetOriginAccessControlResponse getResponse = cloudFrontClient
                .getOriginAccessControl(b -> b.id(originAccessControlId));
        DeleteOriginAccessControlResponse deleteResponse =
 cloudFrontClient.deleteOriginAccessControl(builder -> builder
                .id(originAccessControlId)
                .ifMatch(getResponse.eTag()));
        if (deleteResponse.sdkHttpResponse().isSuccessful()) {
            logger.info("Successfully deleted Origin Access Control [{}]",
 originAccessControlId);
        }
    }

    public static void deleteKeyGroup(final CloudFrontClient cloudFrontClient,
 final String keyGroupId) {

        GetKeyGroupResponse getResponse = cloudFrontClient.getKeyGroup(b ->
 b.id(keyGroupId));
        DeleteKeyGroupResponse deleteResponse =
 cloudFrontClient.deleteKeyGroup(builder -> builder
                .id(keyGroupId)
                .ifMatch(getResponse.eTag()));
        if (deleteResponse.sdkHttpResponse().isSuccessful()) {
            logger.info("Successfully deleted Key Group [{}]", keyGroupId);
        }
    }

    public static void deletePublicKey(final CloudFrontClient cloudFrontClient,
 final String publicKeyId) {
```

```
        GetPublicKeyResponse getResponse = cloudFrontClient.getPublicKey(b ->
 b.id(publicKeyId));

        DeletePublicKeyResponse deleteResponse =
 cloudFrontClient.deletePublicKey(builder -> builder
                .id(publicKeyId)
                .ifMatch(getResponse.eTag()));

        if (deleteResponse.sdkHttpResponse().isSuccessful()) {
            logger.info("Successfully deleted Public Key [{}]", publicKeyId);
        }
    }
}
```

- API-Details finden Sie in den folgenden Themen der AWS SDK for Java 2.x -API-Referenz.
  - DeleteKeyGroup
  - DeleteOriginAccessControl
  - DeletePublicKey

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Erstellen Sie signierte Cookies URLs und Cookies mithilfe eines AWS SDK

Das folgende Codebeispiel zeigt, wie signierte Cookies URLs und Cookies erstellt werden, die den Zugriff auf eingeschränkte Ressourcen ermöglichen.

Java

SDK für Java 2.x

> **Note**
>
> Es gibt noch mehr dazu GitHub. Hier finden Sie das vollständige Beispiel und erfahren, wie Sie das AWS -Code-Beispiel- einrichten und ausführen.

Benutze den CannedSignerRequestKurs, um Cookies mit einer vorgefertigten Richtlinie zu signieren URLs .

```
import software.amazon.awssdk.services.cloudfront.model.CannedSignerRequest;

import java.net.URL;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.time.Instant;
import java.time.temporal.ChronoUnit;

public class CreateCannedPolicyRequest {

    public static CannedSignerRequest createRequestForCannedPolicy(String
 distributionDomainName,
            String fileNameToUpload,
            String privateKeyFullPath, String publicKeyId) throws Exception {
        String protocol = "https";
        String resourcePath = "/" + fileNameToUpload;

        String cloudFrontUrl = new URL(protocol, distributionDomainName,
 resourcePath).toString();
        Instant expirationDate = Instant.now().plus(7, ChronoUnit.DAYS);
        Path path = Paths.get(privateKeyFullPath);

        return CannedSignerRequest.builder()
                .resourceUrl(cloudFrontUrl)
                .privateKey(path)
                .keyPairId(publicKeyId)
                .expirationDate(expirationDate)
                .build();
    }
}
```

Verwenden Sie die CustomSignerRequestKlasse, um Cookies mit einer benutzerdefinierten Richtlinie zu signieren URLs . Die Methoden `activeDate` und `ipRange` sind optionale Methoden.

```
import software.amazon.awssdk.services.cloudfront.model.CustomSignerRequest;

import java.net.URL;
```

```
import java.nio.file.Path;
import java.nio.file.Paths;
import java.time.Instant;
import java.time.temporal.ChronoUnit;

public class CreateCustomPolicyRequest {

    public static CustomSignerRequest createRequestForCustomPolicy(String
 distributionDomainName,
            String fileNameToUpload,
            String privateKeyFullPath, String publicKeyId) throws Exception {
        String protocol = "https";
        String resourcePath = "/" + fileNameToUpload;

        String cloudFrontUrl = new URL(protocol, distributionDomainName,
 resourcePath).toString();
        Instant expireDate = Instant.now().plus(7, ChronoUnit.DAYS);
        // URL will be accessible tomorrow using the signed URL.
        Instant activeDate = Instant.now().plus(1, ChronoUnit.DAYS);
        Path path = Paths.get(privateKeyFullPath);

        return CustomSignerRequest.builder()
                .resourceUrl(cloudFrontUrl)
                // .resourceUrlPattern("https://*.example.com/*")  // Optional.
                .privateKey(path)
                .keyPairId(publicKeyId)
                .expirationDate(expireDate)
                .activeDate(activeDate) // Optional.
                // .ipRange("192.168.0.1/24") // Optional.
                .build();
    }
}
```

Das folgende Beispiel zeigt die Verwendung der CloudFrontUtilitiesKlasse zur Erzeugung signierter Cookies und URLs. Sehen Sie sich dieses Codebeispiel unter an GitHub.

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import software.amazon.awssdk.services.cloudfront.CloudFrontUtilities;
import software.amazon.awssdk.services.cloudfront.cookie.CookiesForCannedPolicy;
import software.amazon.awssdk.services.cloudfront.cookie.CookiesForCustomPolicy;
import software.amazon.awssdk.services.cloudfront.model.CannedSignerRequest;
```

```
import software.amazon.awssdk.services.cloudfront.model.CustomSignerRequest;
import software.amazon.awssdk.services.cloudfront.url.SignedUrl;

public class SigningUtilities {
    private static final Logger logger =
 LoggerFactory.getLogger(SigningUtilities.class);
    private static final CloudFrontUtilities cloudFrontUtilities =
 CloudFrontUtilities.create();

    public static SignedUrl signUrlForCannedPolicy(CannedSignerRequest
 cannedSignerRequest) {
        SignedUrl signedUrl =
 cloudFrontUtilities.getSignedUrlWithCannedPolicy(cannedSignerRequest);
        logger.info("Signed URL: [{}]", signedUrl.url());
        return signedUrl;
    }

    public static SignedUrl signUrlForCustomPolicy(CustomSignerRequest
 customSignerRequest) {
        SignedUrl signedUrl =
 cloudFrontUtilities.getSignedUrlWithCustomPolicy(customSignerRequest);
        logger.info("Signed URL: [{}]", signedUrl.url());
        return signedUrl;
    }

    public static CookiesForCannedPolicy
 getCookiesForCannedPolicy(CannedSignerRequest cannedSignerRequest) {
        CookiesForCannedPolicy cookiesForCannedPolicy = cloudFrontUtilities
                .getCookiesForCannedPolicy(cannedSignerRequest);
        logger.info("Cookie EXPIRES header [{}]",
 cookiesForCannedPolicy.expiresHeaderValue());
        logger.info("Cookie KEYPAIR header [{}]",
 cookiesForCannedPolicy.keyPairIdHeaderValue());
        logger.info("Cookie SIGNATURE header [{}]",
 cookiesForCannedPolicy.signatureHeaderValue());
        return cookiesForCannedPolicy;
    }

    public static CookiesForCustomPolicy
 getCookiesForCustomPolicy(CustomSignerRequest customSignerRequest) {
        CookiesForCustomPolicy cookiesForCustomPolicy = cloudFrontUtilities
                .getCookiesForCustomPolicy(customSignerRequest);
        logger.info("Cookie POLICY header [{}]",
 cookiesForCustomPolicy.policyHeaderValue());
```

```
        logger.info("Cookie KEYPAIR header [{}]",
 cookiesForCustomPolicy.keyPairIdHeaderValue());
        logger.info("Cookie SIGNATURE header [{}]",
 cookiesForCustomPolicy.signatureHeaderValue());
        return cookiesForCustomPolicy;
    }
}
```

- Einzelheiten zur API finden Sie CloudFrontUtilitiesin der AWS SDK for Java 2.x API-Referenz.

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

CloudFront Funktionen, Beispiele für CloudFront

Die folgenden Codebeispiele zeigen, wie Sie CloudFront mit verwenden AWS SDKs.

Beispiele

- Fügen Sie einem Antwortereignis des CloudFront Functions-Viewers HTTP-Sicherheitsheader hinzu
- Fügen Sie einem CloudFront Functions-Viewer-Antwortereignis einen CORS-Header hinzu
- Fügen Sie einem Antwortereignis des CloudFront Functions-Viewers einen Cache-Control-Header hinzu
- Fügen Sie einem CloudFront Functions-Viewer-Anforderungsereignis einen echten Client-IP-Header hinzu
- Fügen Sie einem CloudFront Functions-Viewer-Anforderungsereignis einen Origin-Header hinzu
- Fügen Sie index.html zu einer Anfrage URLs ohne Dateinamen in einem CloudFront Functions-Viewer-Anforderungsereignis hinzu
- Normalisieren Sie die Parameter der Abfragezeichenfolge in einer CloudFront Functions Viewer-Anforderung
- In einem CloudFront Functions-Viewer-Anforderungsereignis zu einer neuen URL umleiten
- Schreiben Sie einen Anforderungs-URI auf der Grundlage der KeyValueStore Konfiguration für ein CloudFront Functions-Viewer-Anforderungsereignis neu

- Leiten Sie Anfragen in einem CloudFront Functions-Viewer-Anforderungsereignis an einen Ursprung weiter, der näher am Betrachter liegt
- Verwenden Sie Schlüssel-Wert-Paare in einer CloudFront Functions-Viewer-Anfrage
- Validieren Sie ein einfaches Token in einer CloudFront Functions-Viewer-Anfrage

Fügen Sie einem Antwortereignis des CloudFront Functions-Viewers HTTP-Sicherheitsheader hinzu

Das folgende Codebeispiel zeigt, wie einem Antwortereignis des CloudFront Functions-Viewers HTTP-Sicherheitsheader hinzugefügt werden.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
async function handler(event) {
    var response = event.response;
    var headers = response.headers;

    // Set HTTP security headers
    // Since JavaScript doesn't allow for hyphens in variable names, we use the
 dict["key"] notation
    headers['strict-transport-security'] = { value: 'max-age=63072000;
 includeSubdomains; preload'};
    headers['content-security-policy'] = { value: "default-src 'none'; img-src
 'self'; script-src 'self'; style-src 'self'; object-src 'none'; frame-ancestors
 'none'"};
    headers['x-content-type-options'] = { value: 'nosniff'};
    headers['x-frame-options'] = {value: 'DENY'};
    headers['x-xss-protection'] = {value: '1; mode=block'};
    headers['referrer-policy'] = {value: 'same-origin'};

    // Return the response to viewers
```

```
    return response;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Fügen Sie einem CloudFront Functions-Viewer-Antwortereignis einen CORS-Header hinzu

Das folgende Codebeispiel zeigt, wie ein CORS-Header zu einem Antwortereignis des CloudFront Functions-Viewers hinzugefügt wird.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
async function handler(event)  {
    var request = event.request;
    var response  = event.response;

    // If Access-Control-Allow-Origin CORS header is missing, add it.
    // Since JavaScript doesn't allow for hyphens in variable names, we use the
 dict["key"] notation.
    if (!response.headers['access-control-allow-origin'] &&
 request.headers['origin']) {
        response.headers['access-control-allow-origin'] = {value:
 request.headers['origin'].value};
        console.log("Access-Control-Allow-Origin was missing, adding it now.");
    }

    return response;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Fügen Sie einem Antwortereignis des CloudFront Functions-Viewers einen Cache-Control-Header hinzu

Das folgende Codebeispiel zeigt, wie ein Cache-Control-Header zu einem Antwortereignis im CloudFront Functions-Viewer hinzugefügt wird.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
async function handler(event) {
    var response = event.response;
    var headers = response.headers;

    if (response.statusCode >= 200 && response.statusCode < 400) {
        // Set the cache-control header
        headers['cache-control'] = {value: 'public, max-age=63072000'};
    }

    // Return response to viewers
    return response;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Fügen Sie einem CloudFront Functions-Viewer-Anforderungsereignis einen echten Client-IP-Header hinzu

Das folgende Codebeispiel zeigt, wie einem CloudFront Functions-Viewer-Anforderungsereignis ein echter Client-IP-Header hinzugefügt wird.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
async function handler(event) {
    var request = event.request;
    var clientIP = event.viewer.ip;

    //Add the true-client-ip header to the incoming request
    request.headers['true-client-ip'] = {value: clientIP};

    return request;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Fügen Sie einem CloudFront Functions-Viewer-Anforderungsereignis einen Origin-Header hinzu

Das folgende Codebeispiel zeigt, wie ein Origin-Header zu einem CloudFront Functions-Viewer-Anforderungsereignis hinzugefügt wird.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
async function handler(event) {
    var request = event.request;
    var headers = request.headers;
    var host = request.headers.host.value;

   // If origin header is missing, set it equal to the host header.
   if (!headers.origin)
       headers.origin = {value:`https://${host}`};

   return request;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Fügen Sie index.html zu einer Anfrage URLs ohne Dateinamen in einem CloudFront Functions-Viewer-Anforderungsereignis hinzu

Das folgende Codebeispiel zeigt, wie index.html zu einer Anfrage URLs ohne Dateinamen in einem CloudFront Functions-Viewer-Anforderungsereignis hinzugefügt wird.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
async function handler(event) {
    var request = event.request;
    var uri = request.uri;

    // Check whether the URI is missing a file name.
    if (uri.endsWith('/')) {
        request.uri += 'index.html';
    }
    // Check whether the URI is missing a file extension.
    else if (!uri.includes('.')) {
        request.uri += '/index.html';
    }

    return request;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Normalisieren Sie die Parameter der Abfragezeichenfolge in einer CloudFront Functions Viewer-Anforderung

Das folgende Codebeispiel zeigt, wie Abfragezeichenfolgenparameter in einer CloudFront Functions Viewer-Anfrage normalisiert werden.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
function handler(event) {
     var qs=[];
     for (var key in event.request.querystring) {
         if (event.request.querystring[key].multiValue) {
             event.request.querystring[key].multiValue.forEach((mv) =>
 {qs.push(key + "=" + mv.value)});
         } else {
             qs.push(key + "=" + event.request.querystring[key].value);
         }
     };

     event.request.querystring = qs.sort().join('&');


     return event.request;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

In einem CloudFront Functions-Viewer-Anforderungsereignis zu einer neuen URL umleiten

Das folgende Codebeispiel zeigt, wie in einem CloudFront Functions-Viewer-Anforderungsereignis zu einer neuen URL umgeleitet wird.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
async function handler(event) {
    var request = event.request;
    var headers = request.headers;
    var host = request.headers.host.value;
    var country = 'DE' // Choose a country code
    var newurl = `https://${host}/de/index.html`; // Change the redirect URL to
 your choice

    if (headers['cloudfront-viewer-country']) {
        var countryCode = headers['cloudfront-viewer-country'].value;
        if (countryCode === country) {
            var response = {
                statusCode: 302,
                statusDescription: 'Found',
                headers:
                    { "location": { "value": newurl } }
                }

            return response;
        }
    }
    return request;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Schreiben Sie einen Anforderungs-URI auf der Grundlage der KeyValueStore Konfiguration für ein CloudFront Functions-Viewer-Anforderungsereignis neu

Das folgende Codebeispiel zeigt, wie eine Anforderungs-URI auf der Grundlage der KeyValueStore Konfiguration für ein CloudFront Functions-Viewer-Anforderungsereignis neu geschrieben wird.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
import cf from 'cloudfront';

// (Optional) Replace KVS_ID with actual KVS ID
const kvsId = "KVS_ID";
// enable stickiness by setting a cookie from origin or using another edge
 function
const stickinessCookieName = "appversion";
// set to true to enable console logging
const loggingEnabled = false;

// function rewrites the request uri based on configuration in KVS
// example config in KVS in key:value format
// "latest": {"a_weightage": .8, "a_url": "v1", "b_url": "v2"}
// given above key and value in KVS the request uri will be rewritten
// for example http(s)://domain/latest/something/else will be rewritten as
 http(s)://domain/v1/something/else or http(s)://domain/v2/something/else
 depending on weightage
// if no configuration is found, then the request is returned as is
async function handler(event) {
    // NOTE: This example function is for a viewer request event trigger.
    // Choose viewer request for event trigger when you associate this function
 with a distribution.
    const request = event.request;
    const pathSegments = request.uri.split('/');
```

```
    const key = pathSegments[1];

    // if empty path segment or if there is valid stickiness cookie
    // then skip call to KVS and let the request continue.
    if (!key || hasValidSticknessCookie(request.cookies[stickinessCookieName],
 key)) {
        return event.request;
    }

    try {
        // get the prefix replacement from KVS
        const replacement = await getPathPrefixByWeightage(key);
        if (!replacement) {
            return event.request;
        }
        //Replace the first path with the replacement
        pathSegments[1] = replacement;
        log(`using prefix ${pathSegments[1]}`)
        const newUri = pathSegments.join('/');
        log(`${request.uri} -> ${newUri}`);
        request.uri = newUri;

        return request;
    } catch (err) {
        // No change to the path if the key is not found or any other error
        log(`request uri: ${request.uri}, error: ${err}`);
    }
    // no change to path - return request
    return event.request;
}

// function to get the prefix from KVS
async function getPathPrefixByWeightage(key) {
    const kvsHandle = cf.kvs(kvsId);
    // get the weightage config from KVS
    const kvsResponse = await kvsHandle.get(key);
    const weightageConfig = JSON.parse(kvsResponse);
    // no configuration - return null
    if (!weightageConfig || !isFinite(weightageConfig.a_weightage)) {
        return null;
    }
    // return the url based on weightage
    // return null if no url is configured
    if (Math.random() <= weightageConfig.a_weightage) {
```

```
        return weightageConfig.a_url ? weightageConfig.a_url: null;
    } else {
        return weightageConfig.b_url ? weightageConfig.b_url : null;
    }
}

// function to check if the stickiness cookie is valid
function hasValidStickinessCookie(stickinessCookie, pathSegment) {
    // if the value exists and it matches pathSegment
    return (stickinessCookie && stickinessCookie.value === pathSegment)
}

function log(message) {
    if (loggingEnabled) {
        console.log(message);
    }
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Leiten Sie Anfragen in einem CloudFront Functions-Viewer-Anforderungsereignis an einen Ursprung weiter, der näher am Betrachter liegt

Das folgende Codebeispiel zeigt, wie Anfragen in einem CloudFront Functions-Viewer-Anforderungsereignis an einen Ursprung weitergeleitet werden, der näher am Betrachter liegt.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
import cf from 'cloudfront';

function handler(event) {
    const request = event.request;
    const headers = request.headers;
    const country = headers['cloudfront-viewer-country'] &&
        headers['cloudfront-viewer-country'].value;

    //List of Regions with S3 buckets containing content
    const countryToRegion = {
        'DE': 'eu-central-1',
        'IE': 'eu-west-1',
        'GB': 'eu-west-2',
        'FR': 'eu-west-3',
        'JP': 'ap-northeast-1',
        'IN': 'ap-south-1'
    };

    const DEFAULT_REGION = 'us-east-1';

    const selectedRegion = (country && countryToRegion[country]) ||
 DEFAULT_REGION;

    const domainName =
        `cloudfront-functions-demo-bucket-in-${selectedRegion}.s3.
${selectedRegion}.amazonaws.com`;

    cf.updateRequestOrigin({
        "domainName": domainName,
        "originAccessControlConfig": {
            "enabled": true,
            "region": selectedRegion,
            "signingBehavior": "always",
            "signingProtocol": "sigv4",
            "originType": "s3"
        },
    });

    return request;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Verwenden Sie Schlüssel-Wert-Paare in einer CloudFront Functions-Viewer-Anfrage

Das folgende Codebeispiel zeigt, wie Schlüssel-Wert-Paare in einer CloudFront Functions-Viewer-Anfrage verwendet werden.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
import cf from 'cloudfront';

// This fails if there is no key value store associated with the function
const kvsHandle = cf.kvs();

// Remember to associate the KVS with your function before referencing KVS in
 your code.
// https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/kvs-with-
functions-associate.html
async function handler(event) {
    const request = event.request;
    // Use the first segment of the pathname as key
    // For example http(s)://domain/<key>/something/else
    const pathSegments = request.uri.split('/')
    const key = pathSegments[1]
    try {
        // Replace the first path of the pathname with the value of the key
        // For example http(s)://domain/<value>/something/else
        pathSegments[1] = await kvsHandle.get(key);
        const newUri = pathSegments.join('/');
        console.log(`${request.uri} -> ${newUri}`)
```

```
        request.uri = newUri;
    } catch (err) {
        // No change to the pathname if the key is not found
        console.log(`${request.uri} | ${err}`);
    }
    return request;
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Validieren Sie ein einfaches Token in einer CloudFront Functions-Viewer-Anfrage

Das folgende Codebeispiel zeigt, wie ein einfaches Token in einer CloudFront Functions-Viewer-Anfrage validiert wird.

JavaScript

JavaScript Runtime 2.0 für CloudFront Funktionen

> **Note**
>
> Es gibt noch mehr dazu GitHub. Das vollständige Beispiel und Informationen zur Einrichtung und Ausführung finden Sie im CloudFront Functions-Beispiel-Repository.

```
import crypto from 'crypto';
import cf from 'cloudfront';

//Response when JWT is not valid.
const response401 = {
    statusCode: 401,
    statusDescription: 'Unauthorized'
};

// Remember to associate the KVS with your function before calling the const
 kvsKey = 'jwt.secret'.
```

```
// https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/kvs-with-
functions-associate.html
const kvsKey = 'jwt.secret';
// set to true to enable console logging
const loggingEnabled = false;


function jwt_decode(token, key, noVerify, algorithm) {
    // check token
    if (!token) {
        throw new Error('No token supplied');
    }
    // check segments
    const segments = token.split('.');
    if (segments.length !== 3) {
        throw new Error('Not enough or too many segments');
    }

    // All segment should be base64
    const headerSeg = segments[0];
    const payloadSeg = segments[1];
    const signatureSeg = segments[2];

    // base64 decode and parse JSON
    const payload = JSON.parse(_base64urlDecode(payloadSeg));

    if (!noVerify) {
        const signingMethod = 'sha256';
        const signingType = 'hmac';

        // Verify signature. `sign` will return base64 string.
        const signingInput = [headerSeg, payloadSeg].join('.');

        if (!_verify(signingInput, key, signingMethod, signingType,
 signatureSeg)) {
            throw new Error('Signature verification failed');
        }

        // Support for nbf and exp claims.
        // According to the RFC, they should be in seconds.
        if (payload.nbf && Date.now() < payload.nbf*1000) {
            throw new Error('Token not yet active');
        }
```

```
        if (payload.exp && Date.now() > payload.exp*1000) {
            throw new Error('Token expired');
        }
    }

    return payload;
}

//Function to ensure a constant time comparison to prevent
//timing side channels.
function _constantTimeEquals(a, b) {
    if (a.length != b.length) {
        return false;
    }

    let xor = 0;
    for (let i = 0; i < a.length; i++) {
    xor |= (a.charCodeAt(i) ^ b.charCodeAt(i));
    }

    return 0 === xor;
}

function _verify(input, key, method, type, signature) {
    if(type === "hmac") {
        return _constantTimeEquals(signature, _sign(input, key, method));
    }
    else {
        throw new Error('Algorithm type not recognized');
    }
}

function _sign(input, key, method) {
    return crypto.createHmac(method, key).update(input).digest('base64url');
}

function _base64urlDecode(str) {
    return Buffer.from(str, 'base64url')
}

async function handler(event) {
    let request = event.request;

    //Secret key used to verify JWT token.
```

```
    //Update with your own key.
    const secret_key = await getSecret()

    if(!secret_key) {
        return response401;
    }

    // If no JWT token, then generate HTTP redirect 401 response.
    if(!request.querystring.jwt) {
        log("Error: No JWT in the querystring");
        return response401;
    }

    const jwtToken = request.querystring.jwt.value;

    try{
        jwt_decode(jwtToken, secret_key);
    }
    catch(e) {
        log(e);
        return response401;
    }

    //Remove the JWT from the query string if valid and return.
    delete request.querystring.jwt;
    log("Valid JWT token");
    return request;
}

// get secret from key value store
async function getSecret() {
    // initialize cloudfront kv store and get the key value
    try {
        const kvsHandle = cf.kvs();
        return await kvsHandle.get(kvsKey);
    } catch (err) {
        log(`Error reading value for key: ${kvsKey}, error: ${err}`);
        return null;
    }

}

function log(message) {
    if (loggingEnabled) {
```

```
        console.log(message);
    }
}
```

Eine vollständige Liste der AWS SDK-Entwicklerhandbücher und Codebeispiele finden Sie unterVerwendung CloudFront mit einem SDK AWS. Dieses Thema enthält auch Informationen zu den ersten Schritten und Details zu früheren SDK-Versionen.

Dokumentverlauf

In der folgenden Tabelle werden die wichtigen Änderungen beschrieben, die an CloudFront der Dokumentation vorgenommen wurden. Um Benachrichtigungen über Aktualisierungen zu erhalten, können Sie den RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Neue Origin-Timeout-Einstellungen hinzugefügt	Das Timeout für den Abschluss der Antwort wurde für alle Ursprünge hinzugefügt und das Antwort-Timeout (Origin-Read-Timeout) für S3-Ursprünge hinzugefügt.	30. Juli 2025
Vorkonfigurierte Standardverteilungseinstellungen hinzugefügt	Vorkonfigurierte Einstellungen für Standardverteilungen hinzugefügt.	17. Juni 2025
Neuer Konsolen-Workflow für die Einrichtung von Standardverteilungsdomänen hinzugefügt	Neuer Konsolen-Workflow für die Einrichtung einer Standardverteilungsdomäne hinzugefügt.	17. Juni 2025
Beispielparameter hinzugefügt	Es wurden Beispiele für die Verwendung von Parametern mit Domainnamen und Herkunftspfaden in Verteilungsmandanten hinzugefügt.	17. Juni 2025
Unterstützung für CloudFront Funktionen für CloudFront SaaS Manager hinzugefügt	Hilfsfunktionen und das `endpoint` Feld für das `context` Objekt hinzugefügt.	2. Mai 2025
Aktualisierungen der Standardprotokollierung (v2)	Die `{distributionid}` Partitionsvariable wurde hinzugefügt, um das Senden	1. Mai 2025

	von Zugriffsprotokollen zu unterstützen AWS Glue.	
Aktualisierungen der CloudFront verwalteten Richtlinien	Den `CloudFrontReadOnlyAccess` und den `CloudFrontFullAccess` verwalteten Richtlinien wurden ACM-Berechtigungen hinzugefügt.	28. April 2025
Unterstützung für Mehrmandantenvertrieb und Distributionsmieter hinzugefügt	Sie können eine Mehrmandantenverteilung erstellen, um allgemeine Verteilungseinstellungen auf der Grundlage Ihres Herkunftstyps festzulegen. Anschließend können Sie die Mehrmandantenverteilung wiederverwenden, um mehrere Verteilungsmandanten zu erstellen, die diese Einstellungen gemeinsam nutzen. Sie können dann bestimmte Verteilungsmandanten anpassen, wenn Sie zusätzliche Websites oder Anwendungen hinzufügen.	28. April 2025
Updates für Lambda @Edge -Funktionen	Lambda @Edge -Funktionen unterstützen jetzt erweiterte Protokollierungssteuerungen und das Anpassen des CloudWatch Protokollgruppennamens.	7. April 2025

Anycast statisch IPs	Sie können Anycast static verwenden, IPs um das Routing von Apex-Domains direkt an Ihre Distributionen zu ermöglichen. CloudFront	4. April 2025
Zusätzliche Hilfsmethoden für die Änderung des Ursprungs hinzugefügt	Die Hilfsmethoden `selectRequestOrigi nById()` und `createReq uestOriginGroup()` CloudFront Functions wurden hinzugefügt.	2. April 2025
Aktualisierungen der Standardprotokollierung (v2)	Die `{accountid}` Partition svariable und Beispiel-Suffixpfade für die Übermittl ung von Zugriffsprotokollen an Amazon S3 wurden hinzugefü gt.	14. Februar 2025
Zusätzliche Echtzeit-Protokoll felder für die Standardp rotokollierung hinzugefügt (v2)	Sie können die Protokoll felder `c-country` und die `cache-behavior-path-pattern` Echtzeit-Protokoll felder angeben, wenn Sie die Standardprotokollierung aktivieren (v2).	31. Januar 2025
Lambda@Edge unterstützt neuere Laufzeitversion	Lambda @Edge unterstützt jetzt Lambda-Funktionen mit der Node.js 22-Laufzeit.	22. November 2024

Unterstützung der Ausfallsicherheit bei Berücksichtigung der Medienqualität für CloudFront	Sie können die Funktion Media Quality-Aware Resiliency (MQAR) verwenden, sodass CloudFront automatisch der Ursprung in einer Ursprungsgruppe mit der höchsten Bewertung der Medienqualität ausgewählt wird.	21. November 2024
Hilfsmethode für die Änderung des Ursprungs	Neue Hilfsmethode CloudFront Functions für die Änderung des Ursprungs hinzugefügt.	21. November 2024
VPC-Ursprünge	Verwenden Sie CloudFront VPC-Ursprünge, um den Zugriff auf einen Application Load Balancer, Network Load Balancer oder EC2 Instance-Ursprung einzuschränken.	20. November 2024
Aktualisierungen der verwalteten Richtlinie	Von `CloudFrontFullAccess` verwaltete Richtlinie aktualisiert.	20. November 2024
Anycast statisch IPs	Sie können Anycast static IPs von anfordern, um es mit Ihren CloudFront Distributionen zu verwenden.	20. November 2024
Unterstützung für Standardprotokollierung hinzugefügt	CloudFront unterstützt die Standardprotokollierung (v2) und das Senden Ihrer Protokolle an Amazon CloudWatch Logs, Amazon Data Firehose und Amazon Simple Storage Service (Amazon S3).	20. November 2024

Unterstützung für gRPC hinzugefügt	CloudFront unterstützt jetzt gRPC-Anfragen für Ihre Distribution.	20. November 2024
Neue verwaltete Richtlinie für VPC-Ursprünge hinzugefügt	Neue verwaltete Richtlinie `AWSCloudFrontVPCOriginServiceRolePolicy` hinzugefügt.	20. November 2024
Lambda@Edge unterstützt neuere Laufzeitversion	Lambda @Edge unterstützt jetzt Lambda-Funktionen mit der Python 3.13-Laufzeit.	13. November 2024
Mit Regeln auswerten AWS Config	Evaluieren Sie Ihre CloudFront Konfigurationen mit AWS Config Regeln.	20. September 2024
Weitere Inhalte zur Fehlerbehebung wurden hinzugefügt	Es wurden weitere Inhalte zur Fehlerbehebung für die Statuscodes für HTTP-Fehlerantworten der Typen 4xx und 5xx hinzugefügt.	26. August 2024
Neue verwaltete Cache-Richtlinien hinzugefügt	Neue verwaltete Cache-Richtlinien hinzugefügt `UseOriginCacheControlHeaders` und`UseOriginCacheControlHeaders-QueryStrings` .	24. Mai 2024
Unterstützung für die Origin-Zugriffskontrolle hinzugefügt	Sie können jetzt eine Origin-Zugriffskontrolle (OAC) für AWS Elemental MediaPackage V2 und eine AWS Lambda Funktions-URL erstellen.	11. April 2024

Echtzeit-Protokollfelder für CMCD	Es wurden 18 CMCD-Felder (Common Media Client Data) für die Protokollierung in Echtzeit hinzugefügt.	9. April 2024
Erste Schritte mit einer CloudFront Standarddistribution	Aktualisiertes Tutorial für eine Standarddistribution, die einen Amazon S3 S3-Ursprung mit Origin Access Control (OAC) verwendet.	18. März 2024
Codebeispiele für die CloudFront Verwendung AWS SDKs	Es wurden Codebeispiele hinzugefügt, die zeigen, wie man es CloudFront mit einem AWS Software Development Kit (SDK) verwendet. Die Beispiele sind in Codeauszüge unterteilt, die Ihnen zeigen, wie Sie einzelne Servicefunktionen aufrufen können, und Beispiele, die Ihnen zeigen, wie Sie eine bestimmte Aufgabe ausführen können, indem Sie mehrere Funktionen innerhalb desselben Services aufrufen.	16. Februar 2024
AWS verwaltetes Richtlinienupdate	Die `CloudFrontReadOnlyAccess` - und `CloudFrontFullAccess` -IAM-Richtlinien unterstützen jetzt `KeyValueStore` -Vorgänge.	19. Dezember 2023
JavaScript Laufzeit 2.0	JavaScript Runtime 2.0-Funktionen für CloudFront Funktionen hinzugefügt.	21. November 2023

CloudFront KeyValueStore	Amazon unterstützt CloudFront jetzt CloudFront KeyValueStore. Bei dieser Funktion handelt es sich um einen sicheren, globalen Schlüssel wert-Datenspeicher mit niedriger Latenz, der den Lesezugriff von Functions aus ermöglicht. CloudFront Sie können erweiterte anpassbare Logik an CloudFront Edge-Standorten aktivieren.	21. November 2023
Lambda@Edge unterstützt neuere Laufzeitversion	Lambda@Edge unterstützt jetzt Lambda-Funktionen mit der Node.js-20-Laufzeit.	15. November 2023
Sicherheits-Dashboard	CloudFront erstellt ein Sicherheits-Dashboard, wenn Sie eine Verteilung erstellen. Aktivieren AWS WAF und verwalten Sie geografische Einschränkungen und zeigen Sie allgemeine Daten für Anfragen, Bots und Protokolle an.	8. November 2023
Sortieren von Abfrageze ichenfolgen in Funktionen	CloudFront unterstützt jetzt das Sortieren von Abfrageze ichenfolgen mithilfe von CloudFront Funktionen.	3. Oktober 2023
AWS WAF Sicherheitsempfehl ungen	Amazon zeigt CloudFront jetzt AWS WAF Sicherheitsempfehl ungen auf der CloudFront Konsole an.	26. September 2023

Unterstützung für die Bereitstellung veralteter (abgelaufener) Cache-Inhalte	CloudFront unterstützt die Direktiven `Stale-While-Revalidate` und `Stale-If-Error` Cache Control.	15. Mai 2023
Aktivieren Sie den AWS WAF Schutz mit einem Klick	Eine optimierte Methode zum Hinzufügen von AWS WAF Sicherheitsvorkehrungen zu Distributionen. CloudFront	10. Mai 2023
Aktiviert diese Option ACLs für neue S3-Buckets, die für Standardprotokolle verwendet werden	Hinweis und Links für die ACL-Standardeinstellung für neue S3-Buckets hinzugefügt.	11. April 2023
Erstellen eines Ursprungs mit Amazon S3 Object Lambda	Sie können ein Alias eines Zugriffspunkts von Amazon S3 Object Lambda als Ursprung für Ihre Verteilung verwenden.	31. März 2023
Passen Sie den HTTP-Status und den Text mithilfe von CloudFront Funktionen an	Sie können CloudFront Funktionen verwenden, um den Antwortstatuscode des Betrachters zu aktualisieren und den Antworttext zu ersetzen oder zu entfernen.	29. März 2023
Platzhalteroptionen für CORS-Header für Ports hinzugefügt	In CORS-Zugriffskontroll-Headern können Sie jetzt Platzhalterkonfigurationen für Ports verwenden.	20. März 2023
Neuer Link für das AWS Security Hub Benutzerhandbuch hinzugefügt	Die Sprache wurde aktualisiert und ein Link zu den neu organisierten CloudFront Amazon-Steuerelementen im AWS Security Hub Benutzerhandbuch hinzugefügt.	9. März 2023

CloudFront unterstützt jetzt Blocklisten („alle außer“) in den Richtlinien für ursprüngliche Anfragen	Verwenden Sie Blocklisten in Richtlinien für ursprüngliche Anfragen, um alle Abfrageze ichenfolgen, HTTP-Header oder Cookies, mit Ausnahme der angegebenen, in Anfragen aufzunehmen, die CloudFron t an den Ursprung gesendet werden.	22. Februar 2023
CloudFront fügt eine neue Richtlinie für verwaltete Origin-Anfragen hinzu, um alle Viewer-Header außer dem Host-Header weiterzuleiten	CloudFrontDie neue Richtlini e von Use für verwaltete Anfragen mit Ursprung bezieht alle Header der Viewer-An frage, mit Ausnahme des `Host` Headers, in Anfragen ein, die CloudFront an den Ursprung gesendet werden.	22. Februar 2023
Die Einschränkungen für Lambda@Edge wurden aktualisiert	Lambda @Edge unterstützt Lambda-Laufzeit-Verwaltungs konfigurationen, die auf Auto (Automatisch) festgelegt sind.	16. Februar 2023
Die IAM-Leitlinien für wurden aktualisiert CloudFront	Aktualisierung des Leitfadens zur Ausrichtung an bewährten IAM-Methoden. Weitere Informationen finden Sie unter Bewährte IAM-Methoden.	15. Februar 2023
Verbesserte Sicherheit mit Ursprungszugriffssteuerung	Sie können MediaStore Origins jetzt sichern, indem Sie nur den Zugriff auf die angegebenen CloudFront Distributionen gewähren.	9. Februar 2023

Neue Header zur Bestimmung der Header-Struktur des Viewers	Sie können jetzt die Header-Reihenfolge und die Anzahl der Header hinzufügen, um den Viewer leichter anhand der gesendeten Header identifizieren zu können.	13. Januar 2023
Lambda@Edge unterstützt neuere Laufzeitversion	Lambda@Edge unterstützt jetzt Lambda-Funktionen mit der Node.js-18-Laufzeit.	12. Januar 2023
Entfernen von Antwort-Headern mithilfe einer Antwort-Header-Richtlinie	Sie können jetzt eine Richtlinie für CloudFront Antwort-Header verwenden, um Header, die in der Antwort CloudFront eingegangen sind, vom Ursprung zu entfernen. Die angegebenen Header sind nicht in der Antwort enthalten, die CloudFront an die Zuschauer gesendet wird.	3. Januar 2023
Kontinuierliche Bereitstellung zum sicheren Testen von Konfigurationsänderungen	Sie können jetzt Änderungen an Ihrer CDN-Konfiguration bereitstellen, indem Sie Tests unter Verwendung eines Teils des Produktionsdatenverkehrs durchführen.	18. November 2022
Veröffentlichung des Headers `CloudFront-Viewer-JA3-Fingerprint`	Sie können jetzt anhand des JA3 Fingerabdrucks feststellen, ob die Anfrage von einem bekannten Client stammt.	16. November 2022

Platzhalteroptionen für CORS-Header hinzugefügt	In einigen CORS-Zugriffskontroll-Headern können Sie jetzt verschiedene Platzhalterkonfigurationen verwenden.	11. November 2022
Zusätzliche Metriken für CloudFront Verteilungen	Support für `MonitoringSubscription` in der CloudFront API und AWS CloudFormation.	3. Oktober 2022
Verbesserte Sicherheit mit Ursprungszugriffssteuerung	Sie können jetzt Amazon S3 S3-Ursprünge sichern, indem Sie nur den Zugriff auf die angegebenen CloudFront Distributionen gewähren.	24. August 2022
HTTP/3-Unterstützung für Distributionen CloudFront	Sie können jetzt HTTP/3 für Ihre Distribution wählen. CloudFront	15. August 2022
Fügen Sie Handshake-Details zum Header hinzu CloudFront-Viewer-TLS	Sie können nun Informationen über den verwendeten SSL/TLS Handshake anzeigen.	27. Juni 2022
Neue Metrik im Server-Timing-Header	Die neue Metrik `cdn-downstream-fbl` wurde zu den `Server-Timing` -Headern hinzugefügt.	13. Juni 2022
Neuer Header zum Abrufen von Informationen über die TLS-Version und Verschlüsselung	Sie können jetzt den `CloudFront-Viewer-TLS` Header verwenden, um Informationen über die Version von TLS (oder SSL) und die Chiffre abzurufen, die für die Verbindung zwischen dem Viewer und verwendet wurde. CloudFront	23. Mai 2022

Neue FunctionThrottles Metrik für Funktionen CloudFront	Mit Amazon können Sie jetzt überwachen CloudWatch, wie oft eine CloudFront Funktion in einem bestimmten Zeitraum gedrosselt wurde.	4. Mai 2022
CloudFront unterstützt die Lambda-Funktion URLs	Wenn Sie eine serverlos e Webanwendung mithilfe von Lambda-Funktionen mit Funktion erstellen URLs, können Sie jetzt eine Reihe von Vorteilen hinzufügen CloudFront .	6. April 2022
Server-Timing-Header in HTTP-Antworten	Sie können jetzt den `Server-Timing` Header in HTTP-Antworten aktivieren, die von gesendet werden CloudFron t , um Metriken anzuzeige n, die Ihnen helfen können, Einblicke in das Verhalten und die Leistung von zu gewinnen. CloudFront	30. März 2022
Verwenden Sie die Präfixlis te AWS-managed, um den eingehenden Datenverkehr zu begrenzen	Sie können jetzt den eingehenden HTTP- und HTTPS-Verkehr von nur den IP-Adressen, die zu den Ursprungsservern gehören, auf CloudFront Ihre Ursprünge beschränken.	7. Februar 2022

Neues Feature	CloudFront fügt Unterstützung für Antwort-Header-Richtlinien hinzu, mit denen Sie die HTTP-Header angeben können, die zu den HTTP-Antworten CloudFront hinzugefügt werden, die an Zuschauer (Webbrowser oder andere Clients) gesendet werden. Sie können die gewünschten Header (und die zugehörigen Werte) angeben, ohne Änderungen am Ursprung vorzunehmen oder Code zu schreiben. Weitere Informationen finden Sie unter Hinzufügen oder Entfernen von HTTP-Headern in Antworten. CloudFront	2. November 2021
Neuer `CloudFront-Viewer-Address` Anforderungsheader	CloudFront fügt Unterstützung für einen neuen Header hinzu`CloudFront-Viewer-Address`, der die IP-Adresse des Viewers enthält, an den die HTTP-Anfrage gesendet hat CloudFront. Weitere Informationen finden Sie unter Hinzufügen von CloudFront Anforderungsheadern.	25. Oktober 2021

Lambda @Edge unterstützt neue Runtime-Version	Lambda@Edge unterstützt jetzt Lambda-Funktionen mit der Python-3.9-Laufzeit. Weitere Informationen finden Sie unter Unterstützte Laufzeiten.	22. September 2021
AWS verwaltetes Richtlinienupdate	CloudFront hat die CloudFrontReadOnlyAccessRichtlinie aktualisiert. Weitere Informationen finden Sie unter CloudFront Aktualisierungen der AWS verwalteten Richtlinien.	8. September 2021
Neues Feature	CloudFront unterstützt jetzt ECDSA-Zertifikate für HTTPS-Verbindungen mit Zuschauerzugriff. Weitere Informationen finden Sie unter Unterstützte Protokolle und Chiffren zwischen Zuschauern und CloudFront Anforderungen für die Verwendung von Zertifikaten mit. SSL/TLS CloudFront	14. Juli 2021
Neues Feature	CloudFront unterstützt jetzt mehr Möglichkeiten, einen alternativen Domainnamen ohne Kontaktaufnahme von einer Distribution auf eine andere zu übertragen. Support Weitere Informationen finden Sie unter Verschieben eines alternativen Domainnamens in eine andere Distribution.	7. Juli 2021

Neue Sicherheitsrichtlinie	CloudFront unterstützt jetzt eine neue Sicherheitsrichtlinie, TLSv1.2_2021, mit einer kleineren Anzahl unterstützter Chiffren. Weitere Informationen finden Sie unter Unterstützte Protokolle und Verschlüsselungen zwischen Zuschauern und. CloudFront	23. Juni 2021
Neues Feature	Amazon unterstützt CloudFront jetzt CloudFront Functions, eine native Funktion CloudFront, mit der Sie einfache Funktionen JavaScript für umfangreiche, latenzempfindliche CDN-Anpassungen schreiben können. Weitere Informationen finden Sie unter Customizing at the Edge with Functions. CloudFront	3. Mai 2021
Lambda @Edge unterstützt neuere Runtime-Versionen	Lambda@Edge unterstützt jetzt Lambda-Funktionen mit der Node.js-14-Laufzeit. Weitere Informationen finden Sie unter Unterstützte Laufzeiten.	29. April 2021

Entfernen Sie die Dokumentation für RTMP-Distributionen	Amazon CloudFront hat RTMP-Distributionen (Real-Time Messaging Protocol) am 31. Dezember 2020 eingestellt. Die Dokumentation für RTMP-Distributionen wurde jetzt aus dem Amazon CloudFront Developer Guide entfernt.	10. Februar 2021
Neue Preisoption	Amazon CloudFront stellt das CloudFront Sicherheitssparpaket vor, mit dem Sie auf einfache Weise bis zu 30% der CloudFront Gebühren auf Ihrer AWS Rechnung sparen können. Weitere Informationen finden Sie im Sparpaket FAQs.	5. Februar 2021
Neues Tutorial	Der Amazon CloudFront Developer Guide enthält jetzt ein Tutorial zur Verwendung von Amazon, CloudFront um den Zugriff auf einen Application Load Balancer in Elastic Load Balancing einzuschränken. Weitere Informationen finden Sie unter Beschränken des Zugriffs auf Application Load Balancers.	18. Dezember 2020

Neue Option für die Verwaltung öffentlicher Schlüssel	CloudFront unterstützt jetzt die Verwaltung öffentlicher Schlüssel für signierte URLs und signierte Cookies über die CloudFront Konsole und die API, ohne dass der AWS-Konto Root-Benutzer Zugriff darauf haben muss. Weitere Informationen finden Sie unter Angabe der Unterzeichner, die signierte URLs und signierte Cookies erstellen können.	22. Oktober 2020
Neue Funktion — Origin Shield	CloudFront unterstützt jetzt CloudFront Origin Shield, eine zusätzliche Ebene in der CloudFront Caching-Infrastruktur, die dazu beiträgt, die Auslastung deines Origins zu minimieren, seine Verfügbarkeit zu verbessern und seine Betriebskosten zu senken. Weitere Informationen finden Sie unter Amazon CloudFront Origin Shield verwenden.	20. Oktober 2020

Neues Komprimierungsformat	CloudFront unterstützt jetzt die Brotli-Kompressionsformation, wenn Sie die Komprimierung von Objekten an CloudFron t Kantenpositionen konfiguri eren CloudFront . Sie können auch so konfigurieren CloudFront , dass Brotli-Ob jekte mithilfe eines normalisi erten Headers zwischeng espeichert werden. `Accept-Encoding` Weitere Informati onen finden Sie unter Bereitste llen komprimierter Dateien und Unterstützung für Komprimie rung.	14. September 2020
Neues TLS-Protokoll	CloudFront unterstützt jetzt das TLS 1.3-Protokoll für HTTPS-Verbindungen zwischen Zuschauern und CloudFront Distributionen. TLS 1.3 ist standardmäßig in allen CloudFront Sicherhei tsrichtlinien aktiviert. Weitere Informationen finden Sie unter Unterstützte Protokolle und Verschlüsselungen zwischen Zuschauern und. CloudFront	3. September 2020

Neue Echtzeitprotokolle	CloudFront unterstützt jetzt konfigurierbare Echtzeitp rotokolle. Mit Echtzeitp rotokollen können Sie Informationen zu Anforderu ngen an eine Verteilung in Echtzeit abrufen. Sie können Echtzeitprotokolle verwenden, um basierend auf der Leistung der Bereitstellung von Inhalten Überwachungsaktionen und Analysen auszuführen und Maßnahmen zu ergreifen. Weitere Informationen finden Sie unter Echtzeitprotokolle.	31. August 2020
API-Unterstützung für zusätzlic he Metriken	CloudFront unterstützt jetzt die Aktivierung von acht zusätzlic hen Echtzeit-Metriken mit der CloudFront API. Weitere Informationen finden Sie unter Zusätzliche Metriken aktiviere n.	28. August 2020
Neue CloudFront HTTP-Head er	CloudFront Es wurden zusätzliche HTTP-Header hinzugefügt, um Informationen über den Viewer wie Gerätetyp , geografische Position und mehr zu ermitteln. Weitere Informationen finden Sie unter Hinzufügen von CloudFront Anforderungsheadern.	23. Juli 2020

Neues Feature	CloudFront unterstützt jetzt Cache-Richtlinien und Ursprungsanforderungsrichtlinien, mit denen Sie den Cache-Schlüssel und die ursprünglichen Anfragen für Ihre Distributionen detaillierter steuern können. CloudFront Weitere Informationen finden Sie unter Steuern des Cache-Schlüssels und Steuern von Ursprungsanfragen.	22. Juli 2020
Neue Sicherheitsrichtlinie	CloudFront unterstützt jetzt eine neue Sicherheitsrichtlinie, TLSv1.2_2019, mit einer kleineren Anzahl unterstützter Chiffren. Weitere Informationen finden Sie unter Unterstützte Protokolle und Verschlüsselungen zwischen Zuschauern und. CloudFront	8. Juli 2020
Neue Einstellungen zur Steuerung von Timeouts und Versuchen bei der Übertragung	CloudFront Es wurden neue Einstellungen hinzugefügt, mit denen Timeouts und Versuche am Ursprung gesteuert werden können. Weitere Informationen finden Sie unter Steuern von Timeouts und Versuchen bei der Herkunft.	5. Juni 2020

Neue Dokumentation für die ersten Schritte CloudFront beim Erstellen einer sicheren statischen Website	Erstellen Sie zunächst eine sichere statische Website mit CloudFront Amazon S3 CloudFront, Lambda @Edge und mehr, die alle mit AWS CloudFormation bereitgestellt werden. Weitere Informationen finden Sie unter Erste Schritte mit einer sicheren statischen Website.	2. Juni 2020
Lambda @Edge unterstützt neuere Runtime-Versionen	Lambda @Edge unterstützt jetzt Lambda-Funktionen mit den Laufzeiten Node.js 12 und Python 3.8. Weitere Informationen finden Sie unter Unterstützte Laufzeiten.	27. Februar 2020
Neue Echtzeit-Metriken in CloudWatch	Amazon CloudFrontnow bietet acht zusätzliche Echtzeitm etriken in Amazon CloudWatc h. Weitere Informationen finden Sie unter Zusätzliche CloudFront Vertriebsmetriken aktivieren.	19. Dezember 2019
Neue Felder in Zugriffsp rotokollen	CloudFront fügt den Zugriffsp rotokollen sieben neue Felder hinzu. Weitere Informationen finden Sie unter Standardf elder für Protokolldateien.	12. Dezember 2019

AWS WordPress Plugin	Sie können das AWS WordPress Plugin verwenden, um Besuchern Ihrer WordPress Website ein beschleunigtes Seherlebnis zu bieten, indem Sie CloudFront. (Update: Seit dem 30. September 2022 ist das AWS WordPress For-Plugin veraltet.)	30. Oktober 2019
IAM-Berechtigungsrichtlinien auf Tagbasis und auf Ressourcenebene	CloudFront unterstützt jetzt zwei zusätzliche Methoden zur Angabe von IAM-Berechtigungsrichtlinien: Tag-basierte und Richtlinienberechtigungen auf Ressourcenebene. Weitere Informationen finden Sie unter Verwalten des Zugriffs auf Ressourcen.	8. August 2019
Support für die Programmiersprache Python	Sie können Funktionen in Lambda@Edge jetzt zusätzlich zu Node.js mit der Programmiersprache Python entwickeln. Beispielfunktionen, die eine Vielzahl von Szenarien abdecken, finden Sie unter Lambda@Edge-Beispielfunktionen.	1. August 2019

Die Überwachungsgrafiken wurden aktualisiert	Inhaltsaktualisierungen zur Beschreibung neuer Möglichkeiten, wie Sie Lambda-Funktionen, die mit Ihren CloudFront Distributionen verknüpft sind, direkt von der CloudFront Konsole aus überwachen können, um Fehler einfacher nachzuverfolgen und zu debuggen. Weitere Informationen finden Sie unter -Überwachung CloudFront.	20. Juni 2019
Konsolidierter Sicherheitsinhalt	In einem neuen Kapitel zur Sicherheit werden Informationen zu den CloudFront Funktionen und der Implementierung von Datenschutz, IAM, Protokollierung, Compliance und mehr zusammengefasst. Weitere Informationen finden Sie unter Sicherheit.	24. Mai 2019
Eine Domainvalidierung ist jetzt erforderlich	CloudFront erfordert jetzt, dass Sie ein SSL-Zertifikat verwenden, um zu überprüfen, ob Sie berechtigt sind, einen alternativen Domainnamen mit einer Distribution zu verwenden. Weitere Informationen finden Sie unter Verwenden alternativer Domainnamen in Verbindung mit HTTPS.	9. April 2019

Der PDF-Dateiname wurde aktualisiert	Der neue Dateiname für den Amazon CloudFront Developer Guide lautet: AmazonClo udFront _DevGuide. Der vorherige Name lautete: cf-dg	7. Januar 2019
Neue Features	CloudFront unterstützt jetzt WebSocket, ein TCP-basie rtes Protokoll, das nützlich ist, wenn Sie langlebige Verbindungen zwischen Clients und Servern benötigen . Sie können jetzt auch Origin-Failover für Szenarien einrichten CloudFront , die eine hohe Verfügbar keit erfordern. Weitere Informationen finden Sie unter Verwendung WebSocket mit CloudFront Distributionen und Optimieren der Hochverfü gbarkeit mit CloudFront Origin Failover.	20. November 2018

Neues Feature	CloudFront unterstützt jetzt eine detaillierte Fehlerprotokollierung für HTTP-Anfragen, die Lambda-Funktionen ausführen. Sie können die Protokolle speichern CloudWatch und sie zur Behebung von HTTP 5xx-Fehlern verwenden, wenn Ihre Funktion eine ungültige Antwort zurückgibt. Weitere Informationen finden Sie unter CloudWatch Metriken und CloudWatch Protokolle für Lambda-Funktionen.	8. Oktober 2018
Neues Feature	Sie können jetzt auswählen, ob Lambda@Edge den Body in einer Anforderung für eine nicht schreibgeschützte HTTP-Methode (POST, PUT, DELETE etc.) weitergeben soll, sodass Sie in Ihrer Lambda-Funktion darauf zugreifen können. Sie können den Lesezugriff wählen oder angeben, dass Sie den Textkörper ersetzen möchten. Weitere Informationen finden Sie unter Zugriff auf den Anforderungstext über die Option „Include Body“ (Text einschließen).	14. August 2018

Neues Feature	CloudFront unterstützt jetzt zusätzlich zu oder anstelle von gzip die Bereitstellung von Inhalten, die mithilfe von Brotli oder anderen Komprimierungsalgorithmen komprimiert wurden. Weitere Informationen finden Sie unter Bereitstellung komprimierter Dateien.	25. Juli 2018
Reorganisation	Der Amazon CloudFront Developer Guide wurde neu organisiert, um die Suche nach verwandten Inhalten zu vereinfachen und die Scanbarkeit und Navigation zu verbessern.	28. Juni 2018
Neue Funktion	Mit Lambda@Edge können Sie jetzt die Bereitstellung von Inhalten, die in einem Amazon-S3-Bucket gespeichert sind, weiter anpassen. Sie haben nun Zugriff auf zusätzliche Header, einschließlich benutzerdefinierter Header innerhalb von Ereignissen auf der Ursprungsseite. Weitere Informationen zur Personalisierung von Inhalten basierend auf Standort und Gerätetyp des Betrachters finden Sie in diesen Beispielen.	20. März 2018

Neue Funktion	Sie können jetzt Amazon verwenden, um HTTPS-Verbindungen CloudFront zu Ursprüngen mithilfe des Elliptic Curve Digital Signature Algorithm (ECDSA) auszuhandeln. ECDSA verwendet kleinere Schlüssel, die schneller, aber genauso sicher wie der ältere RSA-Algorithmus sind. Weitere Informationen finden Sie unter Unterstützte SSL/TLS Protokolle und Chiffren für die Kommunikation zwischen und Ihrem Ursprung und Über RSA CloudFront - und ECDSA-Chiffren.	15. März 2018
Neue Funktion	Lambda @Edge ermöglicht es Ihnen, Fehlerantworten von Ihrem Ursprung aus anzupassen, indem Sie Lambda-Funktionen als Reaktion auf HTTP-Fehler ausführen können, die Amazon CloudFrontreceives von Ihrem Ursprung aus hat. Weitere Informationen entnehmen Sie den Beispielen für Weiterleitungen an einen anderen Standort und die Generierung einer Antwort mit dem Statuscode 200 (OK).	21. Dezember 2017

Neue Funktion	Eine neue CloudFront Funktion, die Verschlüsselung auf Feldebene, hilft Ihnen dabei, die Sicherheit sensibler Daten wie Kreditkartennummern oder personenbezogener Daten (PII) wie Sozialversicherungsnummern weiter zu verbessern. Weitere Informationen finden Sie unter Verschlüsselung auf Feldebene zum Schutz vertraulicher Daten verwenden.	14. Dezember 2017
Der Dokumentenverlauf wurde archiviert	Der Verlauf älterer Dokumente wurde archiviert.	1. Dezember 2017

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.