



Benutzerhandbuch

AWS Deadline Cloud



Version latest

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Deadline Cloud: Benutzerhandbuch

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Was ist Deadline Cloud?	1
Funktionen von Deadline Cloud	1
Konzepte und Terminologie	2
Erste Schritte mit Deadline Cloud	4
Zugreifen auf Deadline Cloud	5
Zugehörige Services	5
So funktioniert Deadline Cloud	6
.....	7
Berechtigungen in Deadline Cloud	7
Softwareunterstützung mit Deadline Cloud	8
Erste Schritte	9
Einrichten Ihres AWS-Konto s	9
Richten Sie Ihren Monitor ein	10
Schritt 1: Richten Sie Ihren Monitor ein	10
Schritt 2: Definieren Sie die Farmdetails	14
Schritt 3: Definieren Sie die Warteschlangendetails	14
Schritt 4: Definieren Sie Flottendetails	16
Schritt 5: Konfigurieren Sie die Anforderungen für Mitarbeiter	17
Schritt 6: Definieren Sie Zugriffsebenen	17
Schritt 7: Überprüfen und erstellen	17
Richten Sie eine Entwickler-Workstation ein	18
Schritt 1: Farm erstellen	18
Schritt 2: Führen Sie den Worker Agent aus	23
Schritt 3: Jobs einreichen und ausführen	25
Schritt 4: Jobs mit Anhängen ausführen	33
Schritt 5: Fügen Sie eine vom Service verwaltete Flotte hinzu	42
Schritt 6: Bereinigen Sie die Farmressourcen	45
Richten Sie den Einreicher ein	48
Schritt 1: Installieren Sie den Deadline Cloud Submitter	49
Schritt 2: Deadline Cloud Monitor installieren und einrichten	57
Schritt 3: Starten Sie den Deadline Cloud Submitter	60
Benutze die Farm	64
Den Monitor verwenden	65
Teilen Sie die URL des Deadline Cloud-Monitors	66

Öffnen Sie den Deadline Cloud-Monitor	66
Warteschlangen- und Flottendetails anzeigen	68
Jobs, Schritte und Aufgaben anzeigen und verwalten	69
Jobdetails anzeigen	70
Einen Schritt anzeigen	71
Eine Aufgabe anzeigen	72
Anzeigen von -Protokollen	72
Laden Sie die fertige Ausgabe herunter	74
Farmen	76
Erstellen Sie eine Farm	76
Löschen Sie eine Farm	77
Bearbeiten Sie eine Farm	77
Warteschlangen	78
Erstellen einer Warteschlange	78
Erstellen Sie eine Warteschlangenumgebung	80
CondaStandard-Warteschlangenumgebung	81
Löschen einer Warteschlange	83
Bearbeiten einer Warteschlange	83
Ordnen Sie eine Warteschlange und eine Flotte zu	83
Verwaltung von Flotten	85
Vom Service verwaltete Flotten	85
VFX-Plattform	87
Vom Kunden verwaltete Flotten	88
Erstellen Sie ein CMF	88
Einrichtung des Worker-Hosts	94
Zugriff verwalten	99
Installieren Sie Software für Jobs	102
Konfigurieren von -Anmeldeinformationen	103
Erstellen eines AMI	104
Erstellen Sie eine Flotteninfrastruktur	107
Stellen Sie eine Connect zu einem Lizenzendpunkt her	118
Verwalten von Benutzern	123
Benutzer und Gruppen für den Monitor verwalten	123
Verwalten Sie Benutzer und Gruppen für Farmen, Warteschlangen und Flotten	125
Aufträge	128
Jobs einreichen	129

Weitere Optionen zum Einreichen von Jobs	131
Jobs planen	133
Prüfen Sie die Flottenkompatibilität	133
Skalierung der Flotte	135
Sitzungen	135
Abhängigkeiten der einzelnen Schritte	137
Auftragsstatus	139
Jobs ändern	142
Jobs werden verarbeitet	146
Fehlerbehebung bei Aufträgen	147
Warum ist die Erstellung meines Jobs fehlgeschlagen?	148
Warum ist mein Job nicht kompatibel?	148
Warum ist mein Job immer noch fertig?	148
Warum ist mein Job gescheitert?	149
Warum steht mein Schritt noch aus?	149
Speicher	150
Arbeitsanhänge	150
Verschlüsselung für S3-Buckets mit Stellenanhängen	151
Verwaltung von Job-Anhängen in S3-Buckets	152
Virtuelles Dateisystem	153
Gemeinsamer Speicher	155
Speicherprofile in Deadline Cloud	156
Verwaltung von Budgets und Nutzung	158
Annahmen zu den Kosten	158
Den Budgetmanager verwenden	159
Voraussetzung	160
Rufen Sie den Budgetmanager auf	160
Budget erstellen	161
Ein Budget anzeigen	162
Bearbeiten Sie ein Budget	162
Deaktivieren Sie ein Budget	163
Verwenden des Usage Explorers	163
Voraussetzung	164
Öffnen Sie den Usage Explorer	164
Verwenden Sie den Usage Explorer	163
Kostenmanagement	167

Bewährte Methoden für das Kostenmanagement	168
Sicherheit	171
Datenschutz	172
Verschlüsselung im Ruhezustand	173
Verschlüsselung während der Übertragung	173
Schlüsselverwaltung	173
Datenschutz für den Datenverkehr zwischen Netzwerken	184
Abmelden	184
Identitäts- und Zugriffsverwaltung	185
Zielgruppe	186
Authentifizierung mit Identitäten	186
Verwalten des Zugriffs mit Richtlinien	191
So funktioniert Deadline Cloud mit IAM	193
Beispiele für identitätsbasierte Richtlinien	201
AWS verwaltete Richtlinien	205
Fehlerbehebung	209
Compliance-Validierung	211
Ausfallsicherheit	213
Sicherheit der Infrastruktur	213
Konfigurations- und Schwachstellenanalyse	214
Serviceübergreifende Confused-Deputy-Prävention	214
AWS PrivateLink	216
Überlegungen	216
Deadline Cloud Endpunkte	217
Endpunkte erstellen	218
Bewährte Methoden für die Gewährleistung der Sicherheit	218
Datenschutz	219
IAM-Berechtigungen	220
Führen Sie Jobs als Benutzer und Gruppen aus	220
Netzwerk	220
Daten zum Job	221
Struktur der Farm	221
Warteschlangen für Arbeitsanhänge	222
Benutzerdefinierte Software-Buckets	224
Worker-Hosts	225
Workstations	226

Überwachen	228
Protokollierung mit CloudTrail	229
Deadline Cloud-Informationen finden Sie unter CloudTrail	230
Grundlegendes zu Deadline Cloud-Protokolldateieinträgen	234
Überwachung mit CloudWatch	235
Auf EventBridge Ereignisse reagieren	236
Änderung der Empfehlung zur Flottengröße	237
Kontingente	239
AWS CloudFormation Ressourcen	240
Deadline Cloud und AWS CloudFormation Vorlagen	240
Erfahren Sie mehr über AWS CloudFormation	240
Dokumentverlauf	242
AWS Glossar	243
.....	ccxliv

Was ist AWS Deadline Cloud?

Mit Deadline Cloud können AWS-Service Sie Rendering-Projekte und -Jobs auf Amazon Elastic Compute Cloud (Amazon EC2) -Instances direkt von Pipelines und Workstations aus zur Erstellung digitaler Inhalte erstellen und verwalten.

Deadline Cloud bietet Konsolenschnittstellen, lokale Anwendungen, Befehlszeilentools und eine API. Mit Deadline Cloud können Sie Farmen, Flotten, Jobs, Benutzergruppen und Speicher erstellen, verwalten und überwachen. Sie können auch Hardwareanforderungen angeben, Umgebungen für bestimmte Workloads erstellen und die Tools zur Inhaltserstellung, die für Ihre Produktion erforderlich sind, in Ihre Deadline Cloud-Pipeline integrieren.

Deadline Cloud bietet eine einheitliche Oberfläche, über die Sie all Ihre Rendering-Projekte an einem Ort verwalten können. Sie können Benutzer verwalten, ihnen Projekte zuweisen und Berechtigungen für Jobrollen erteilen.

Themen

- [Funktionen von Deadline Cloud](#)
- [Konzepte und Terminologie für Deadline Cloud](#)
- [Erste Schritte mit Deadline Cloud](#)
- [Zugreifen auf Deadline Cloud](#)
- [Zugehörige Services](#)
- [So funktioniert Deadline Cloud](#)

Funktionen von Deadline Cloud

Hier sind einige der wichtigsten Möglichkeiten, wie Deadline Cloud Ihnen bei der Ausführung und Verwaltung von Visual Computing-Workloads helfen kann:

- Erstellen Sie schnell Ihre Farmen, Warteschlangen und Flotten. Überwachen Sie ihren Status und gewinnen Sie Einblicke in den Betrieb Ihrer Farm und Ihre Jobs.
- Verwalten Sie Deadline Cloud-Benutzer und -Gruppen zentral und weisen Sie Berechtigungen zu.
- Verwalten Sie die Anmeldesicherheit für Projektbenutzer und externe Identitätsanbieter mit AWS IAM Identity Center.

- Verwalten Sie den Zugriff auf Projektressourcen sicher mit AWS Identity and Access Management (IAM-) Richtlinien und Rollen.
- Verwenden Sie Tags, um Projektressourcen zu organisieren und schnell zu finden.
- Verwalten Sie die Nutzung der Projektressourcen und die geschätzten Kosten für Ihr Projekt.
- Stellen Sie eine breite Palette von Rechenverwaltungsoptionen bereit, um das Rendern in der Cloud oder persönlich zu unterstützen.

Konzepte und Terminologie für Deadline Cloud

Um Ihnen den Einstieg in AWS Deadline Cloud zu erleichtern, werden in diesem Thema einige der wichtigsten Konzepte und Begrifflichkeiten erläutert.

Budgetmanager

Der Budgetmanager ist Teil des Deadline Cloud-Monitors. Verwenden Sie den Budgetmanager, um Budgets zu erstellen und zu verwalten. Sie können ihn auch verwenden, um Aktivitäten einzuschränken, um das Budget einzuhalten.

Deadline Cloud-Kundenbibliothek

Die Client Library umfasst eine Befehlszeilenschnittstelle und eine Bibliothek zur Verwaltung von Deadline Cloud. Zu den Funktionen gehören das Senden von Jobpaketen auf der Grundlage der Open Job Description-Spezifikation an Deadline Cloud, das Herunterladen von Ausgaben für Jobanhänge und die Überwachung Ihrer Farm über die Befehlszeilenschnittstelle.

Anwendung zur Erstellung digitaler Inhalte (DCC)

Anwendungen zur Erstellung digitaler Inhalte (DCCs) sind Produkte von Drittanbietern, mit denen Sie digitale Inhalte erstellen. Beispiele für DCCs sind Maya, und. Nuke Houdini Deadline Cloud bietet integrierte Plugins für Jobeinreicher für bestimmte DCCs.

Farm

Eine Farm ist ein Ort, an dem sich Ihre Projektressourcen befinden. Sie besteht aus Warteschlangen und Flotten.

Flotte

Eine Flotte ist eine Gruppe von Worker-Knoten, die das Rendern durchführen. Worker-Knoten verarbeiten Jobs. Eine Flotte kann mehreren Warteschlangen zugeordnet werden, und eine Warteschlange kann mehreren Flotten zugeordnet werden.

Aufgabe

Ein Job ist eine Rendering-Anfrage. Benutzer reichen Jobs ein. Jobs enthalten spezifische Jobeigenschaften, die als Schritte und Aufgaben beschrieben werden.

Arbeitsanhänge

Ein Jobanhang ist eine Deadline Cloud-Funktion, mit der Sie Eingaben und Ausgaben für Jobs verwalten können. Auftragsdateien werden während des Rendervorgangs als Auftragsanhänge hochgeladen. Bei diesen Dateien kann es sich um Texturen, 3D-Modelle, Lichtenanlagen und ähnliche Objekte handeln.

Auftragseigenschaften

Auftragseigenschaften sind Einstellungen, die Sie beim Absenden eines Renderjobs definieren. Einige Beispiele umfassen den Bildbereich, den Ausgabepfad, Auftragsanhänge, renderfähige Kamera und mehr. Die Eigenschaften variieren je nach dem DCC, von dem das Rendering eingereicht wurde.

Auftragsvorlage

Eine Jobvorlage definiert die Laufzeitumgebung und alle Prozesse, die als Teil eines Deadline Cloud-Jobs ausgeführt werden.

Warteschlange

In einer Warteschlange befinden sich eingereichte Jobs und deren Rendern ist geplant. Eine Warteschlange muss einer Flotte zugeordnet werden, um ein erfolgreiches Rendern zu ermöglichen. Eine Warteschlange kann mehreren Flotten zugeordnet werden.

Zuordnung zwischen Warteschlange und Flotte

Wenn eine Warteschlange einer Flotte zugeordnet ist, liegt eine Zuordnung zwischen Warteschlange und Flotte vor. Verwenden Sie eine Zuordnung, um Mitarbeitern aus einer Flotte Aufträge in dieser Warteschlange zuzuordnen. Sie können Zuordnungen starten und beenden, um die Arbeitsplanung zu steuern.

Schritt

Ein Schritt ist ein bestimmter Prozess, der im Job ausgeführt werden soll.

Frist für den Cloud-Einreicher

Ein Deadline Cloud-Einreicher ist ein DCC-Plugin (Digital Content Creation). Künstler verwenden es, um Jobs über eine DCC-Schnittstelle eines Drittanbieters einzureichen, mit der sie vertraut sind.

Tags

Ein Tag ist eine Bezeichnung, die Sie einer AWS Ressource zuweisen können. Jedes Tag besteht aus einem Schlüssel und einem optionalen Wert, den Sie definieren.

Mit Tags können Sie Ihre AWS Ressourcen auf unterschiedliche Weise kategorisieren. Sie könnten beispielsweise eine Reihe von Tags für die Amazon EC2 EC2-Instances Ihres Kontos definieren, mit deren Hilfe Sie den Besitzer und die Stack-Ebene jeder Instance verfolgen können.

Sie können Ihre AWS Ressourcen auch nach Zweck, Eigentümer oder Umgebung kategorisieren. Dieser Ansatz ist nützlich, wenn Sie über viele Ressourcen desselben Typs verfügen. Anhand der Tags, die Sie ihr zugewiesen haben, können Sie eine bestimmte Ressource schnell identifizieren.

Aufgabe

Eine Aufgabe ist eine einzelne Komponente eines Renderschritts.

Nutzungsbasierte Lizenzierung (UBL)

Die nutzungsbasierte Lizenzierung (UBL) ist ein On-Demand-Lizenzmodell, das für ausgewählte Produkte von Drittanbietern verfügbar ist. Bei diesem Modell handelt es sich um eine nutzungsabhängige Bezahlung, bei der Ihnen die Anzahl der Stunden und Minuten in Rechnung gestellt wird, die Sie nutzen.

Nutzungsexplorer

Der Usage Explorer ist eine Funktion von Deadline Cloud Monitor. Er bietet eine ungefähre Schätzung Ihrer Kosten und Nutzung.

Worker

Mitarbeiter gehören zu Flotten und führen die von Deadline Cloud zugewiesenen Aufgaben aus, um Schritte und Aufträge zu erledigen. Mitarbeiter speichern die Protokolle von Aufgabenvorgängen in Amazon CloudWatch Logs. Mitarbeiter können auch die Funktion für Jobanhänge verwenden, um Eingaben und Ausgaben mit einem Amazon Simple Storage Service (Amazon S3) -Bucket zu synchronisieren.

Erste Schritte mit Deadline Cloud

Verwenden Sie Deadline Cloud, um schnell eine Renderfarm mit Standardeinstellungen und Ressourcen wie der Amazon EC2 EC2-Instanzkonfiguration und Amazon Simple Storage Service (Amazon S3) -Buckets zu erstellen.

Sie können die Einstellungen und Ressourcen auch definieren, wenn Sie eine Renderfarm erstellen. Diese Methode nimmt mehr Zeit in Anspruch als die Verwendung der Standardeinstellungen und Ressourcen, bietet Ihnen jedoch mehr Kontrolle.

Nachdem Sie sich mit den [Konzepten und der Terminologie](#) von Deadline Cloud vertraut gemacht haben, finden Sie unter [Erste Schritte](#) step-by-step Anweisungen zum Erstellen Ihrer Farm, zum Hinzufügen von Benutzern und Links zu hilfreichen Informationen.

Zugreifen auf Deadline Cloud

Sie können auf eine der folgenden Arten auf Deadline Cloud zugreifen:

- Deadline Cloud-Konsole — Greifen Sie in einem Browser auf die Konsole zu, um eine Farm und ihre Ressourcen zu erstellen und den Benutzerzugriff zu verwalten. Weitere Informationen finden Sie unter [Erste Schritte](#).
- Deadline Cloud Monitor — Verwalten Sie Ihre Renderjobs, einschließlich der Aktualisierung von Prioritäten und Jobstatus. Überwachen Sie Ihre Farm und sehen Sie sich Protokolle und den Auftragsstatus an. Für Benutzer mit Inhaberberechtigungen bietet der Deadline Cloud-Monitor auch Zugriff darauf, die Nutzung zu untersuchen und Budgets zu erstellen. Der Deadline Cloud-Monitor ist sowohl als Webbrowser als auch als Desktop-Anwendung verfügbar.
- AWS SDK und AWS CLI — Verwenden Sie AWS Command Line Interface (AWS CLI), um die Deadline Cloud-API-Operationen von der Befehlszeile auf Ihrem lokalen System aus aufzurufen. Weitere Informationen finden Sie unter [Eine Entwickler-Workstation einrichten](#).

Zugehörige Services

Deadline Cloud funktioniert mit den folgenden Komponenten AWS-Services:

- Amazon CloudWatch — Mit CloudWatch können Sie Ihre Projekte und die zugehörigen AWS Ressourcen überwachen. Weitere Informationen finden Sie im [CloudWatchAmazon-Benutzerhandbuch](#).
- Amazon EC2 — Dies AWS-Service bietet virtuelle Server, auf denen Ihre Anwendungen in der Cloud ausgeführt werden. Sie können Ihre Projekte so konfigurieren, dass sie Amazon EC2 EC2-Instances für Ihre Workloads verwenden. Weitere Informationen finden Sie unter [Amazon EC2 EC2-Instances](#).
- Amazon EC2 Auto Scaling — Mit Auto Scaling können Sie die Anzahl der Instances automatisch erhöhen oder verringern, wenn sich die Nachfrage nach Ihren Instances ändert. Auto Scaling

hilft sicherzustellen, dass Sie die gewünschte Anzahl von Instances ausführen, auch wenn eine Instance ausfällt. Wenn Sie Auto Scaling mit Deadline Cloud aktivieren, werden Instances, die von Auto Scaling gestartet werden, automatisch beim Workload registriert. Ebenso werden Instances, die durch Auto Scaling beendet wurden, automatisch vom Workload abgemeldet.

Weitere Informationen finden Sie im [Amazon EC2 Auto Scaling Scaling-Benutzerhandbuch](#).

- **AWS PrivateLink**— AWS PrivateLink bietet private Konnektivität zwischen Virtual Private Clouds (VPCs) und Ihren lokalen Netzwerken AWS-Services, ohne dass Ihr Datenverkehr dem öffentlichen Internet ausgesetzt wird. AWS PrivateLink macht es einfach, Dienste über verschiedene Konten und VPCs hinweg zu verbinden. Weitere Informationen finden Sie unter [AWS PrivateLink](#).
- **Amazon S3** — Amazon S3 ist ein Objektspeicherservice. Deadline Cloud verwendet Amazon S3 S3-Buckets zum Speichern von Job-Anhängen.
- **IAM Identity Center** — Im IAM Identity Center können Sie Benutzern von einem zentralen AWS-Service Ort aus Single-Sign-On-Zugriff auf alle ihnen zugewiesenen Konten und Anwendungen gewähren. Sie können den Zugriff mehrerer Konten und die Benutzerberechtigungen für alle Ihre Konten auch zentral verwalten. AWS Organizations Weitere Informationen finden Sie unter [AWS IAM Identity Center : Häufig gestellte Fragen](#).

So funktioniert Deadline Cloud

Mit Deadline Cloud können Sie Rendering-Projekte und -Jobs direkt über Pipelines und Workstations zur Erstellung digitaler Inhalte (DCC) erstellen und verwalten.

Sie reichen Jobs mit dem AWS SDK, AWS Command Line Interface (AWS CLI) oder den Deadline Cloud-Job-Einreichern an Deadline Cloud ein. Deadline Cloud unterstützt die Open Job Description (OpenJD) für die Spezifikation von Jobvorlagen. Weitere Informationen finden Sie unter [Stellenbeschreibung öffnen](#) auf der GitHub Website.

Deadline Cloud stellt Jobeinreicher zur Verfügung. Ein Job Submitter ist ein DCC-Plugin zum Senden von Renderjobs über eine DCC-Schnittstelle eines Drittanbieters, z. B. oder. Maya Nuke Mit einem Einreicher können Künstler Rendereaufträge über eine Schnittstelle eines Drittanbieters an Deadline Cloud einreichen, wo Projektressourcen verwaltet und Jobs überwacht werden — alles von einem Ort aus.

Mit einer Deadline Cloud-Farm können Sie Warteschlangen und Flotten erstellen, Benutzer verwalten und die Nutzung und Kosten von Projektressourcen verwalten. Eine Farm besteht aus Warteschlangen und Flotten. In einer Warteschlange befinden sich eingereichte Jobs, deren Rendern geplant ist. Eine Flotte ist eine Gruppe von Worker-Knoten, die Aufgaben ausführen, um

Jobs abzuschließen. Eine Warteschlange muss einer Flotte zugeordnet werden, damit die Jobs gerendert werden können. Eine einzelne Flotte kann mehrere Warteschlangen unterstützen, und eine Warteschlange kann von mehreren Flotten unterstützt werden.

Jobs bestehen aus Schritten, und jeder Schritt besteht aus bestimmten Aufgaben. Mit dem Deadline Cloud-Monitor können Sie auf Status, Protokolle und andere Kennzahlen zur Fehlerbehebung für Jobs, Schritte und Aufgaben zugreifen.

Berechtigungen in Deadline Cloud

Deadline Cloud unterstützt Folgendes:

- Verwaltung des Zugriffs auf seine API-Operationen mithilfe von AWS Identity and Access Management (IAM)
- Verwaltung des Zugriffs von Workforce-Benutzern mithilfe einer Integration mit AWS IAM Identity Center

Bevor jemand an einem Projekt arbeiten kann, muss er Zugriff auf dieses Projekt und die zugehörige Farm haben. Deadline Cloud ist in IAM Identity Center integriert, um die Authentifizierung und Autorisierung von Mitarbeitern zu verwalten. Benutzer können direkt zu IAM Identity Center hinzugefügt werden, oder es kann mit Ihrem bestehenden Identity Provider (IdP) wie Okta oder verbunden werden. Active Directory IT-Administratoren können Benutzern und Gruppen auf verschiedenen Ebenen Zugriffsberechtigungen gewähren. Jede nachfolgende Ebene umfasst die Berechtigungen für die vorherigen Ebenen. In der folgenden Liste werden die vier Zugriffsebenen von der niedrigsten bis zur höchsten Ebene beschrieben:

- Zuschauer — Berechtigung zum Anzeigen von Ressourcen in den Farmen, Warteschlangen, Flotten und Aufträgen, auf die sie Zugriff haben. Ein Zuschauer kann keine Jobs einreichen oder Änderungen daran vornehmen.
- Mitwirkender — Identisch mit einem Betrachter, aber mit der Erlaubnis, Jobs an eine Warteschlange oder Farm zu senden.
- Manager — Identisch mit dem Mitwirkenden, aber mit der Berechtigung, Jobs in Warteschlangen zu bearbeiten, auf die er Zugriff hat, und Berechtigungen für Ressourcen zu erteilen, auf die er Zugriff hat.
- Besitzer — Identisch mit dem Manager, kann jedoch Budgets anzeigen und erstellen und deren Nutzung einsehen.

Note

Diese Berechtigungen gewähren Benutzern keinen Zugriff auf die Deadline Cloud-Infrastruktur AWS Management Console oder die Erlaubnis, sie zu ändern.

Benutzer müssen Zugriff auf eine Farm haben, bevor sie auf die zugehörigen Warteschlangen und Flotten zugreifen können. Der Benutzerzugriff wird Warteschlangen und Flotten innerhalb einer Farm separat zugewiesen.

Sie können Benutzer als Einzelpersonen oder als Teil einer Gruppe hinzufügen. Das Hinzufügen von Gruppen zu einer Farm, Flotte oder Warteschlange kann die Verwaltung von Zugriffsberechtigungen für große Personengruppen vereinfachen. Wenn Sie beispielsweise ein Team haben, das an einem bestimmten Projekt arbeitet, können Sie jedes Teammitglied zu einer Gruppe hinzufügen. Anschließend können Sie der gesamten Gruppe Zugriffsberechtigungen für die entsprechende Farm, Flotte oder Warteschlange gewähren.

Softwareunterstützung mit Deadline Cloud

Deadline Cloud funktioniert mit jeder Softwareanwendung, die über eine Befehlszeilenschnittstelle ausgeführt und mithilfe von Parameterwerten gesteuert werden kann. Deadline Cloud unterstützt die OpenJD Spezifikation zur Beschreibung von Arbeit als Jobs mit Softwareskriptschritten, die zu Aufgaben parametrisiert sind (z. B. über einen Frame-Bereich). Stellen Sie OpenJD Arbeitsanweisungen mit Tools und Funktionen von Deadline Cloud zu Auftragspaketen zusammen, um die Schritte aus einer Softwareanwendung eines Drittanbieters zu erstellen, auszuführen und zu lizenzieren.

Zum Rendern von Jobs ist eine Lizenz erforderlich. Deadline Cloud bietet nutzungsbasierte Lizenzen (UBL) für eine Auswahl von Softwareanwendungslizenzen an, die je nach Nutzung stundenweise in Minutenschritten abgerechnet werden. Mit Deadline Cloud können Sie auch Ihre eigenen Softwarelizenzen verwenden, wenn Sie möchten. Wenn ein Job nicht auf eine Lizenz zugreifen kann, wird er nicht gerendert und es wird ein Fehler ausgegeben, der im Aufgabenprotokoll im Deadline Cloud-Monitor angezeigt wird.

Erste Schritte mit Deadline Cloud

Um eine Farm in AWS Deadline Cloud zu erstellen, können Sie entweder die [Deadline Cloud-Konsole](#) oder die AWS Command Line Interface (AWS CLI) verwenden. Verwenden Sie die Konsole für eine geführte Erfahrung bei der Erstellung der Farm, einschließlich Warteschlangen und Flotten. Verwenden Sie den AWS CLI, um direkt mit dem Service zu arbeiten oder um Ihre eigenen Tools zu entwickeln, die mit Deadline Cloud funktionieren.

Um eine Farm zu erstellen und den Deadline Cloud-Monitor zu verwenden, richten Sie Ihr Konto für Deadline Cloud ein. Sie müssen die Deadline Cloud-Monitor-Infrastruktur nur einmal pro Konto einrichten. Von Ihrer Farm aus können Sie Ihr Projekt verwalten, einschließlich des Benutzerzugriffs auf Ihre Farm und ihre Ressourcen.

Um eine Farm zu erstellen, ohne die Deadline Cloud-Monitorinfrastruktur einzurichten, richten Sie eine Entwickler-Workstation für Deadline Cloud ein.

Um eine Farm mit minimalen Ressourcen für die Annahme von Jobs zu erstellen, wählen Sie auf der Startseite der Konsole Schnellstart aus. [Richten Sie den Deadline Cloud-Monitor ein](#) führt Sie durch diese Schritte. Diese Farmen beginnen mit einer Warteschlange und einer Flotte, die automatisch zugeordnet werden. Dieser Ansatz ist eine bequeme Möglichkeit, Farmen im Sandbox-Stil zum Experimentieren zu erstellen.

Themen

- [Einrichten Ihres AWS-Kontos](#)
- [Richten Sie den Deadline Cloud-Monitor ein](#)
- [Einrichtung einer Entwickler-Workstation für Deadline Cloud](#)
- [Richten Sie Deadline Cloud-Einreicher ein](#)
- [Benutze die Farm](#)

Einrichten Ihres AWS-Kontos

Richten Sie Ihr System AWS-Konto für die Verwendung von AWS Deadline Cloud ein.

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/signup>.
2. Folgen Sie den Online-Anweisungen.

Bei der Anmeldung müssen Sie auch einen Telefonanruf entgegennehmen und einen Verifizierungscode über die Telefontasten eingeben.

Wenn Sie sich für eine anmelden AWS-Konto, Root-Benutzer des AWS-Kontos wird eine erstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Aus Sicherheitsgründen sollten Sie einem Benutzer Administratorzugriff zuweisen und nur den Root-Benutzer verwenden, um [Aufgaben auszuführen, für die Root-Benutzerzugriff erforderlich](#) ist.

Wenn Sie zum ersten Mal eine 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. Der Zugriff erfolgt, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben.

Important

Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen und verwenden Sie diese, 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.

Richten Sie den Deadline Cloud-Monitor ein

Um zu beginnen, müssen Sie Ihre Deadline Cloud-Monitor-Infrastruktur erstellen und Ihre Farm definieren. Sie können auch zusätzliche, optionale Schritte ausführen, darunter das Hinzufügen von Gruppen und Benutzern, die Auswahl einer Servicerolle und das Hinzufügen von Tags zu Ihren Ressourcen.

Schritt 1: Richten Sie Ihren Monitor ein

Der Deadline Cloud-Monitor wird verwendet AWS IAM Identity Center , um Benutzer zu autorisieren. Die IAM Identity Center-Instanz, die Sie für Deadline Cloud verwenden, muss sich in derselben

Konfiguration AWS-Region wie der Monitor befinden. Wenn Ihre Konsole bei der Erstellung des Monitors eine andere Region verwendet, werden Sie daran erinnert, zur IAM Identity Center-Region zu wechseln.

Die Infrastruktur Ihres Monitors besteht aus den folgenden Komponenten:

- **Monitor-Anzeigename:** Anhand des Monitor-Anzeigenamens können Sie Ihren Monitor identifizieren, z. B. AnyCompany Monitor. Der Name Ihres Monitors bestimmt auch Ihre Monitor-URL.

 **Important**

Sie können den Anzeigenamen des Monitors nicht mehr ändern, nachdem Sie die Einrichtung abgeschlossen haben.

- **Monitor-URL:** Sie können über die Monitor-URL auf Ihren Monitor zugreifen. Die URL basiert auf dem Anzeigenamen des Monitors — zum Beispiel <https://anycompanymonitor.awsapps.com>.

 **Important**

Sie können die Monitor-URL nicht ändern, nachdem Sie die Einrichtung abgeschlossen haben.

- **AWS-Region:** Das AWS-Regionist der physische Standort für eine Sammlung von AWS Rechenzentren. Wenn Sie Ihren Monitor einrichten, wird als Region standardmäßig der Standort ausgewählt, der Ihnen am nächsten liegt. Wir empfehlen, die Region so zu ändern, dass sie Ihren Benutzern am nächsten ist. Dies reduziert die Verzögerung und verbessert die Datenübertragungsgeschwindigkeit. AWS IAM Identity Center muss genauso AWS-Region wie Deadline Cloud aktiviert sein.

 **Important**

Sie können Ihre Region nicht ändern, nachdem Sie die Einrichtung von Deadline Cloud abgeschlossen haben.

Führen Sie die Aufgaben in diesem Abschnitt aus, um die Infrastruktur Ihres Monitors zu konfigurieren.

Um die Infrastruktur Ihres Monitors zu konfigurieren

1. Melden Sie sich bei der an AWS Management Console, um das Welcome to Deadline Cloud-Setup zu starten, und wählen Sie dann Weiter.
2. Geben Sie zum Beispiel den Anzeigenamen des Monitors ein **AnyCompany Monitor**.
3. (Optional) Um den Monitornamen zu ändern, wählen Sie „URL bearbeiten“.
4. (Optional) Um den AWS-Regionso zu ändern, dass er Ihren Benutzern am nächsten kommt, wählen Sie „Region ändern“.
 - a. Wählen Sie die Region aus, die Ihren Benutzern am nächsten ist.
 - b. Wählen Sie „Region anwenden“.
 - (Optional) Um Gruppen und Benutzer hinzuzufügen, wählen Sie [\(Optional\) Fügen Sie Gruppen und Benutzer hinzu](#).
 - (Optional) Um Ihre Monitoreinstellung weiter anzupassen, wählen Sie [Zusätzliche Einstellungen](#).
5. Wenn Sie dazu bereit sind [Schritt 2: Definieren Sie die Farmdetails](#), wählen Sie Weiter.

(Optional) Fügen Sie Gruppen und Benutzer hinzu

Bevor Sie die Einrichtung des Deadline Cloud-Monitors abschließen, können Sie Monitor-Benutzer hinzufügen und sie einer Gruppe hinzufügen.

Nach Abschluss der Einrichtung können Sie neue Benutzer und Gruppen erstellen und Benutzer verwalten, um ihnen beispielsweise Gruppen, Berechtigungen und Anwendungen zuzuweisen oder Benutzer von Ihrem Monitor zu löschen.

Zusätzliche Einstellungen

Die Einrichtung von Deadline Cloud umfasst zusätzliche Einstellungen. Mit diesen Einstellungen können Sie alle Änderungen einsehen, die das Deadline Cloud-Setup an Ihnen vornimmt AWS-Konto, Ihre Monitor-Benutzerrolle konfigurieren und den Typ Ihres Verschlüsselungsschlüssels ändern.

AWS IAM Identity Center

AWS IAM Identity Center ist ein cloudbasierter Single-Sign-On-Service zur Verwaltung von Benutzern und Gruppen. IAM Identity Center kann auch in den Single Sign-On (SSO) -Anbieter Ihres

Unternehmens integriert werden, sodass sich Benutzer mit ihrem Unternehmenskonto anmelden können.

Deadline Cloud aktiviert IAM Identity Center standardmäßig und ist für die Einrichtung und Verwendung von Deadline Cloud erforderlich. Die IAM Identity Center-Instanz, die Sie für Deadline Cloud verwenden, muss sich in derselben Umgebung AWS-Region wie der Monitor befinden. Weitere Informationen finden Sie unter [Was ist AWS IAM Identity Center](#).

Konfigurieren Sie die Dienstzugriffsrolle

Ein AWS Dienst kann eine Servicerolle übernehmen, um Aktionen in Ihrem Namen auszuführen. Deadline Cloud benötigt eine Monitor-Benutzerrolle, damit Benutzer auf Ressourcen in Ihrem Monitor zugreifen können.

Sie können verwaltete AWS Identity and Access Management (IAM) -Richtlinien an die Benutzerrolle „Monitor“ anhängen. Die Richtlinien ermöglichen es Benutzern, bestimmte Aktionen auszuführen, z. B. das Erstellen von Jobs in einer bestimmten Deadline Cloud-Anwendung. Da Anwendungen von bestimmten Bedingungen in der verwalteten Richtlinie abhängen, funktioniert die Anwendung möglicherweise nicht wie erwartet, wenn Sie die verwalteten Richtlinien nicht verwenden.

Sie können die Rolle des Monitor-Benutzers nach Abschluss der Installation jederzeit ändern. Weitere Informationen zu Benutzerrollen finden Sie unter [IAM-Rollen](#).

Die folgenden Registerkarten enthalten Anweisungen für zwei verschiedene Anwendungsfälle. Um eine neue Servicerolle zu erstellen und zu verwenden, wählen Sie die Registerkarte Neue Servicerolle. Um eine bestehende Servicerolle zu verwenden, wählen Sie die Registerkarte Bestehende Servicerolle.

New service role

Um eine neue Servicerolle zu erstellen und zu verwenden

1. Wählen Sie Neue Servicerolle erstellen und verwenden aus.
2. (Optional) Geben Sie einen Namen für die Dienstbenutzerrolle ein.
3. Wählen Sie Berechtigungsdetails anzeigen aus, um weitere Informationen zur Rolle zu erhalten.

Existing service role

Um eine bestehende Servicerolle zu verwenden

1. Wählen Sie Bestehende Servicerolle verwenden aus.
2. Öffnen Sie die Dropdownliste, um eine bestehende Servicerolle auszuwählen.
3. (Optional) Wählen Sie In der IAM-Konsole anzeigen aus, um weitere Informationen zur Rolle zu erhalten.

Schritt 2: Definieren Sie die Farmdetails

Gehen Sie zurück zur Deadline Cloud-Konsole und führen Sie die folgenden Schritte aus, um die Farmdetails zu definieren.

1. Fügen Sie in den Farmdetails einen Namen für die Farm hinzu.
2. Geben Sie unter Beschreibung die Farmbeschreibung ein. Eine klare Beschreibung kann Ihnen helfen, den Zweck Ihrer Farm schnell zu ermitteln.
3. (Optional) Standardmäßig werden Ihre Daten mit einem Schlüssel verschlüsselt, der zu Ihrer eigenen Sicherheit AWS gehört und verwaltet wird. Sie können Verschlüsselungseinstellungen anpassen (erweitert) wählen, um einen vorhandenen Schlüssel zu verwenden oder einen neuen, von Ihnen verwalteten Schlüssel zu erstellen.

Wenn Sie die Verschlüsselungseinstellungen über das Kontrollkästchen anpassen möchten, geben Sie einen AWS KMS ARN ein oder erstellen Sie einen neuen, AWS KMS indem Sie Neuen KMS-Schlüssel erstellen wählen.

4. (Optional) Wählen Sie Neues Tag hinzufügen aus, um Ihrer Farm ein oder mehrere Tags hinzuzufügen.
5. Wählen Sie eine der folgenden Optionen:
 - Wählen Sie „Zur Überprüfung springen“ und „Erstellen“, um [Ihre Farm zu überprüfen und zu erstellen](#).
 - Wählen Sie Weiter, um mit weiteren optionalen Schritten fortzufahren.

(Optional) Schritt 3: Definieren Sie die Warteschlangendetails

Die Warteschlange ist dafür verantwortlich, den Fortschritt zu verfolgen und die Arbeit für Ihre Jobs zu planen.

1. Geben Sie zunächst in den Warteschlangendetails einen Namen für die Warteschlange ein.

2. Geben Sie unter Beschreibung die Beschreibung der Warteschlange ein. Eine klare Beschreibung kann Ihnen helfen, den Zweck Ihrer Warteschlange schnell zu identifizieren.
3. Für Job-Anhänge können Sie entweder einen neuen Amazon S3 S3-Bucket erstellen oder einen vorhandenen Amazon S3 S3-Bucket auswählen. Wenn Sie noch keinen Amazon S3 S3-Bucket haben, müssen Sie einen erstellen.
 - a. Um einen neuen Amazon S3 S3-Bucket zu erstellen, wählen Sie Neuen Job-Bucket erstellen. Sie können den Namen des Job-Buckets im Feld Root-Präfix definieren. Wir empfehlen, den Bucket aufzurufen **deadlinecloud-job-attachments-[MONITORNAME]**.

Sie können nur Kleinbuchstaben und Bindestriche verwenden. Keine Leerzeichen oder Sonderzeichen.
 - b. Um nach einem vorhandenen Amazon S3 S3-Bucket zu suchen und diesen auszuwählen, wählen Sie Aus vorhandenem Amazon S3 S3-Bucket auswählen. Suchen Sie anschließend nach einem vorhandenen Bucket, indem Sie „S3 durchsuchen“ wählen. Wenn die Liste Ihrer verfügbaren Amazon S3 S3-Buckets angezeigt wird, wählen Sie den Amazon S3 S3-Bucket aus, den Sie für Ihre Warteschlange verwenden möchten.
4. Wenn Sie vom Kunden verwaltete Flotten verwenden, wählen Sie Zuordnung zu kundenverwalteten Flotten aktivieren aus.
 - Fügen Sie für vom Kunden verwaltete Flotten einen für die Warteschlange konfigurierten Benutzer hinzu und legen Sie dann die POSIX- und/oder Windows-Anmeldeinformationen fest. Alternativ können Sie die Run-As-Funktionalität umgehen, indem Sie das Kontrollkästchen aktivieren.
5. Ihre Warteschlange benötigt die Erlaubnis, in Ihrem Namen auf Amazon S3 zuzugreifen. Wir empfehlen Ihnen, für jede Warteschlange eine neue Servicerolle zu erstellen.
 - a. Führen Sie für eine neue Rolle die folgenden Schritte aus.
 - i. Wählen Sie Neue Servicerolle erstellen und verwenden aus.
 - ii. Geben Sie einen Rollennamen für Ihre Warteschlangenrolle ein oder verwenden Sie den angegebenen Rollennamen.
 - iii. (Optional) Fügen Sie eine Beschreibung der Warteschlangenrolle hinzu.
 - iv. Sie können die IAM-Berechtigungen für die Warteschlangenrolle anzeigen, indem Sie Berechtigungsdetails anzeigen wählen.
 - b. Alternativ können Sie eine vorhandene Servicerolle auswählen.

6. (Optional) Fügen Sie mithilfe von Namens- und Wertepaaren Umgebungsvariablen für die Warteschlangenumgebung hinzu.
7. (Optional) Fügen Sie mithilfe von Schlüssel- und Wertepaaren Tags für die Warteschlange hinzu.

Nachdem Sie alle Warteschlangendetails eingegeben haben, wählen Sie Weiter aus.

(Optional) Schritt 4: Definieren Sie Flottendetails

Eine Flotte weist Mitarbeitern Mitarbeiter zu, die Ihre Rendering-Aufgaben ausführen. Wenn Sie eine Flotte für Ihre Renderaufgaben benötigen, aktivieren Sie das Kontrollkästchen Flotte erstellen.

1. Einzelheiten zur Flotte
 - a. Geben Sie sowohl einen Namen als auch eine optionale Beschreibung für Ihre Flotte an.
 - b. Wählen Sie aus, wie Ihre Rechenressourcen skaliert werden sollen. Mit der Option Service-Managed kann Deadline Cloud Ihre Rechenressourcen auto skalieren. Bei der vom Kunden verwalteten Option haben Sie die Kontrolle über Ihre eigene Rechenskalierung.
2. Wählen Sie im Abschnitt Instanzoption entweder Spot oder On-Demand aus. Amazon EC2 On-Demand-Instances bieten eine schnellere Verfügbarkeit und Amazon EC2 Spot-Instances eignen sich besser zur Kosteneinsparung.
3. Wählen Sie für die automatische Skalierung der Anzahl der Instances in Ihrer Flotte sowohl eine Mindestanzahl von Instances als auch eine Maximale Anzahl von Instances.

Wir empfehlen dringend, immer die Mindestanzahl an Instanzen festzulegen, **0** um zusätzliche Kosten zu vermeiden.

4. Ihre Flotte benötigt die Erlaubnis, in Ihrem Namen CloudWatch an Sie zu schreiben. Wir empfehlen Ihnen, für jede Flotte eine neue Servicerolle zu erstellen.
 - a. Führen Sie für eine neue Rolle die folgenden Schritte aus.
 - i. Wählen Sie Neue Servicerolle erstellen und verwenden aus.
 - ii. Geben Sie einen Rollennamen für Ihre Flottenrolle ein oder verwenden Sie den angegebenen Rollennamen.
 - iii. (Optional) Fügen Sie eine Beschreibung der Flottenrolle hinzu.
 - iv. Sie können die IAM-Berechtigungen für die Flottenrolle anzeigen, indem Sie Berechtigungsdetails anzeigen wählen.
 - b. Alternativ können Sie eine vorhandene Servicerolle verwenden.

5. (Optional) Fügen Sie mithilfe von Schlüssel- und Wertepaaren Tags für die Flotte hinzu.

Nachdem Sie alle Flottendetails eingegeben haben, wählen Sie Weiter aus.

(Optional) Schritt 5: Konfigurieren Sie die Mitarbeiteranforderungen

Definieren Sie die Anforderungen für Ihre Worker-Instanzen.

1. Prüfen Sie die Einstellungen des Betriebssystems (OS) und der CPU-Architektur auf Informationen.
2. Aktualisieren Sie die Mindest- und Höchstanzahl an vCPUs für Ihre Hardwareanforderungen.
3. Aktualisieren Sie die minimale und maximale Speicheranzahl (GiB) für Ihre Hardwareanforderungen.
4. Sie können Instance-Typen filtern, indem Sie Typen von Worker-Instances entweder zulassen oder ausschließen. In beiden Filteroptionen können Sie bis zu 10 Amazon EC2 EC2-Instance-Typen filtern.
5. Unter Zusätzliche Anforderungen (optional) können Sie das EBS-Stammvolumen nach Größe (GiB), IOPS und Durchsatz (MiB/s) definieren.
6. Nachdem alle Worker-Anforderungen festgelegt wurden, wählen Sie Weiter, um die Zugriffsebene für Ihre Gruppen zu definieren.

(Optional) Schritt 6: Definieren Sie Zugriffsebenen

Wenn Sie Gruppen mit Ihrem Monitor verbunden haben, können Sie deren Zugriffsebene definieren. Die Erlaubnis zur Nutzung der Funktionen von Deadline Cloud wird nach Zugriffsebenen verwaltet. Sie können Benutzergruppen unterschiedliche Zugriffsebenen zuweisen.

1. Verwenden Sie das Menü mit den Zugriffsebenen der Deadline Cloud-Farm, um die Berechtigungsstufe für die Gruppe auszuwählen.
2. Wählen Sie Weiter, um fortzufahren und alle eingegebenen Farmdetails zu überprüfen.

Schritt 7: Überprüfen und erstellen

Überprüfen Sie alle Informationen, die Sie zur Erstellung Ihrer Farm eingegeben haben. Wenn Sie bereit sind, wählen Sie Create Farm aus.

Der Fortschritt der Erstellung Ihrer Farm wird auf der Seite Farmen angezeigt. Eine Erfolgsmeldung wird angezeigt, wenn Ihre Farm betriebsbereit ist.

Einrichtung einer Entwickler-Workstation für Deadline Cloud

In diesem Tutorial werden Sie verwenden, AWS CloudShell um eine einfache Entwicklerfarm zu erstellen und den Worker-Agent auszuführen. Anschließend können Sie einen einfachen Job mit Parametern und Anhängen einreichen und ausführen, eine vom Service verwaltete Flotte hinzufügen und Ihre Farmressourcen bereinigen, wenn Sie fertig sind.

In den folgenden Abschnitten lernen Sie die verschiedenen Funktionen von Deadline Cloud kennen und erfahren, wie sie funktionieren und zusammenarbeiten. Das Befolgen dieser Schritte ist nützlich, um neue Workloads und Anpassungen zu entwickeln und zu testen.

Themen

- [Schritt 1: Erstellen Sie eine Deadline Cloud-Farm](#)
- [Schritt 2: Führen Sie den Worker Agent im Entwicklermodus in Deadline Cloud aus](#)
- [Schritt 3: Jobs mit Deadline Cloud einreichen und ausführen](#)
- [Schritt 4: Jobs mit Job-Anhängen in Deadline Cloud ausführen](#)
- [Schritt 5: Fügen Sie Ihrer Entwicklerfarm in Deadline Cloud eine serviceverwaltete Flotte hinzu](#)
- [Schritt 6: Bereinigen Sie Ihre Farmressourcen in Deadline Cloud](#)

Schritt 1: Erstellen Sie eine Deadline Cloud-Farm

Um Ihre Entwicklerfarm und Warteschlangenressourcen in AWS Deadline Cloud zu erstellen, verwenden Sie die AWS Command Line Interface (AWS CLI), wie im folgenden Verfahren gezeigt. Außerdem erstellen Sie eine AWS Identity and Access Management (IAM) -Rolle und eine vom Kunden verwaltete Flotte (CMF) und ordnen die Flotte Ihrer Warteschlange zu. Anschließend können Sie das konfigurieren AWS CLI und sicherstellen, dass Ihre Farm wie angegeben eingerichtet ist und funktioniert.

Sie können diese Farm verwenden, um die Funktionen von Deadline Cloud zu erkunden und anschließend neue Workloads, Anpassungen und Pipeline-Integrationen zu entwickeln und zu testen.

Um eine Farm zu erstellen

1. Installieren und konfigurieren Sie AWS Command Line Interface (AWS CLI), falls Sie dies noch nicht getan haben. Weitere Informationen finden [Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI](#).
2. Erstellen Sie einen Namen für Ihre Farm und fügen Sie diesen Farmnamen hinzu `~/.bashrc`. Dadurch wird es für andere Terminalsitzungen verfügbar.

```
echo "DEV_FARM_NAME=DeveloperFarm" >> ~/.bashrc
source ~/.bashrc
```

3. Erstellen Sie die Farmressource und fügen Sie ihre Farm-ID hinzu `~/.bashrc`.

```
aws deadline create-farm \
  --display-name "$DEV_FARM_NAME"

echo "DEV_FARM_ID=$(aws deadline list-farms \
  --query \"farms[?displayName=='\${DEV_FARM_NAME}'].farmId \
  | [0]\" --output text)" >> ~/.bashrc
source ~/.bashrc
```

4. Erstellen Sie die Warteschlangenressource und fügen Sie ihre Warteschlangen-ID hinzu `~/.bashrc`.

```
aws deadline create-queue \
  --farm-id $DEV_FARM_ID \
  --display-name "$DEV_FARM_NAME Queue" \
  --job-run-as-user '{"posix": {"user": "job-user", "group": "job-group"},
  "runAs": "QUEUE_CONFIGURED_USER"}'

echo "DEV_QUEUE_ID=$(aws deadline list-queues \
  --farm-id \${DEV_FARM_ID} \
  --query \"queues[?displayName=='\${DEV_FARM_NAME} Queue'].queueId \
  | [0]\" --output text)" >> ~/.bashrc
source ~/.bashrc
```

5. Erstellen Sie eine IAM-Rolle für die Flotte. Mit dieser Rolle erhalten Worker-Hosts in Ihrer Flotte die erforderlichen Sicherheitsanmeldedaten, um Jobs aus Ihrer Warteschlange auszuführen.

```
aws iam create-role \
  --role-name "${DEV_FARM_NAME}FleetRole" \
```

```
--assume-role-policy-document \  
  '{  
    "Version": "2012-10-17",  
    "Statement": [  
      {  
        "Effect": "Allow",  
        "Principal": {  
          "Service": "credentials.deadline.amazonaws.com"  
        },  
        "Action": "sts:AssumeRole"  
      }  
    ]  
  }'  
aws iam put-role-policy \  
  --role-name "${DEV_FARM_NAME}FleetRole" \  
  --policy-name WorkerPermissions \  
  --policy-document \  
  '{  
    "Version": "2012-10-17",  
    "Statement": [  
      {  
        "Effect": "Allow",  
        "Action": [  
          "deadline:AssumeFleetRoleForWorker",  
          "deadline:UpdateWorker",  
          "deadline>DeleteWorker",  
          "deadline:UpdateWorkerSchedule",  
          "deadline:BatchGetJobEntity",  
          "deadline:AssumeQueueRoleForWorker"  
        ],  
        "Resource": "*",  
        "Condition": {  
          "StringEquals": {  
            "aws:PrincipalAccount": "${aws:ResourceAccount}"  
          }  
        }  
      },  
      {  
        "Effect": "Allow",  
        "Action": [  
          "logs:CreateLogStream"  
        ],  
        "Resource": "arn:aws:logs:*:*:*:/aws/deadline/*",  
        "Condition": {
```

```

        "StringEquals": {
            "aws:PrincipalAccount": "${aws:ResourceAccount}"
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "logs:PutLogEvents",
            "logs:GetLogEvents"
        ],
        "Resource": "arn:aws:logs:*:*:*:/aws/deadline/*",
        "Condition": {
            "StringEquals": {
                "aws:PrincipalAccount": "${aws:ResourceAccount}"
            }
        }
    }
]
}'

```

6. Erstellen Sie die vom Kunden verwaltete Flotte (CMF) und fügen Sie deren Flotten-ID hinzu~/ .bashrc.

```

FLEET_ROLE_ARN="arn:aws:iam::$(aws sts get-caller-identity \
    --query "Account" --output text):role/${DEV_FARM_NAME}FleetRole"
aws deadline create-fleet \
    --farm-id $DEV_FARM_ID \
    --display-name "$DEV_FARM_NAME CMF" \
    --role-arn $FLEET_ROLE_ARN \
    --max-worker-count 5 \
    --configuration \
        '{
            "customerManaged": {
                "mode": "NO_SCALING",
                "workerCapabilities": {
                    "vCpuCount": {"min": 1},
                    "memoryMiB": {"min": 512},
                    "osFamily": "linux",
                    "cpuArchitectureType": "x86_64"
                }
            }
        }
}'

```

```
echo "DEV_CMF_ID=$(aws deadline list-fleets \
  --farm-id \ $DEV_FARM_ID \
  --query \"fleets[?displayName=='\ $DEV_FARM_NAME CMF'].fleetId \
  | [0]\" --output text)" >> ~/.bashrc
source ~/.bashrc
```

7. Stellen Sie sicher, dass Sie auf Deadline Cloud zugreifen können.

```
pip install deadline
```

8. Ordnen Sie das CMF Ihrer Warteschlange zu.

```
aws deadline create-queue-fleet-association \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --fleet-id $DEV_CMF_ID
```

9. Verwenden Sie den folgenden Befehl, um die Standardfarm auf die Farm-ID und die Warteschlange auf die Warteschlangen-ID festzulegen, die Sie zuvor erstellt haben.

```
deadline config set defaults.farm_id $DEV_FARM_ID
deadline config set defaults.queue_id $DEV_QUEUE_ID
```

10. (Optional) Verwenden Sie die folgenden Befehle, um zu überprüfen, ob Ihre Farm gemäß Ihren Spezifikationen eingerichtet wurde:

- Alle Farmen auflisten — **deadline farm list**
- Alle Warteschlangen in der Standardfarm auflisten — **deadline queue list**
- Alle Flotten in der Standardfarm auflisten — **deadline fleet list**
- Holen Sie sich die Standardfarm — **deadline farm get**
- Holen Sie sich die Standardwarteschlange — **deadline queue get**
- Ruft alle Flotten ab, die der Standardwarteschlange zugeordnet sind — **deadline fleet get**

Schritt 2: Führen Sie den Worker Agent im Entwicklermodus in Deadline Cloud aus

Bevor Sie die Jobs, die Sie an die Warteschlange auf Ihrer Entwicklerfarm einreichen, ausführen können, müssen Sie den AWS Deadline Cloud-Worker-Agent im Entwicklermodus auf einem Worker-Host ausführen.

Im weiteren Verlauf dieses Tutorials werden Sie mithilfe von zwei AWS CloudShell Tabs AWS CLI Operationen auf Ihrer Entwicklerfarm ausführen. Auf der ersten Registerkarte können Sie Jobs einreichen. Auf der zweiten Registerkarte können Sie den Worker Agent ausführen.

Note

Wenn Sie Ihre CloudShell Sitzung länger als 20 Minuten inaktiv lassen, kommt es zu einem Timeout und der Worker-Agent wird gestoppt. Folgen Sie den Anweisungen im folgenden Verfahren, um den Worker-Agent neu zu starten.

Um den Worker Agent im Entwicklermodus auszuführen

1. Installieren und konfigurieren Sie AWS Command Line Interface (AWS CLI), falls Sie dies noch nicht getan haben. Weitere Informationen finden [Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI](#).
2. Wenn Ihre Farm immer noch auf der ersten CloudShell Registerkarte geöffnet ist, öffnen Sie eine zweite CloudShell Registerkarte und erstellen Sie dann die `demoenv-persist` Verzeichnisse `demoenv-logs` und.

```
mkdir ~/demoenv-logs
mkdir ~/demoenv-persist
```

3. Laden Sie die Deadline Cloud Worker Agent-Pakete von PyPI herunter und installieren Sie sie:

Note

Bei Aktivierung ist es erforderlich, dass die Agentendateien im globalen Site-Packages-Verzeichnis von Python installiert werden. Virtuelle Python-Umgebungen werden derzeit nicht unterstützt.

```
python -m pip install deadline-cloud-worker-agent
```

4. Damit der Worker-Agent die temporären Verzeichnisse für die Ausführung von Jobs erstellen kann, erstellen Sie ein Verzeichnis:

```
sudo mkdir /sessions
sudo chmod 750 /sessions
sudo chown cloudshell-user /sessions
```

5. Führen Sie den Deadline Cloud-Worker-Agent im Entwicklermodus mit den Variablen DEV_FARM_ID aus DEV_CMF_ID, die Sie dem hinzugefügt haben ~/ .bashrc.

```
deadline-worker-agent \
  --farm-id $DEV_FARM_ID \
  --fleet-id $DEV_CMF_ID \
  --run-jobs-as-agent-user \
  --logs-dir ~/demoenv-logs \
  --persistence-dir ~/demoenv-persist
```

Während der Worker-Agent den UpdateWorkerSchedule API-Vorgang initialisiert und dann abfragt, wird die folgende Ausgabe angezeigt:

```
INFO    Worker Agent starting
[2024-03-27 15:51:01,292][INFO    ] # Worker Agent starting
[2024-03-27 15:51:01,292][INFO    ] AgentInfo
Python Interpreter: /usr/bin/python3
Python Version: 3.9.16 (main, Sep  8 2023, 00:00:00) - [GCC 11.4.1 20230605 (Red
Hat 11.4.1-2)]
Platform: linux
...
[2024-03-27 15:51:02,528][INFO    ] # API.Resp # [deadline:UpdateWorkerSchedule]
(200) params={'assignedSessions': {}, 'cancelSessionActions': {},
'updateIntervalSeconds': 15} ...
[2024-03-27 15:51:17,635][INFO    ] # API.Resp # [deadline:UpdateWorkerSchedule]
(200) params=(Duplicate removed, see previous response) ...
[2024-03-27 15:51:32,756][INFO    ] # API.Resp # [deadline:UpdateWorkerSchedule]
(200) params=(Duplicate removed, see previous response) ...
...
```

- Wählen Sie Ihre erste CloudShell Registerkarte aus und listen Sie dann die Mitarbeiter in der Flotte auf.

```
deadline worker list --fleet-id $DEV_CMF_ID
```

Es wird eine Ausgabe wie die folgende angezeigt:

```
Displaying 1 of 1 workers starting at 0

- workerId: worker-8c9af877c8734e89914047111f
  status: STARTED
  createdAt: 2023-12-13 20:43:06+00:00
```

In einer Produktionskonfiguration erfordert der Deadline Cloud-Worker-Agent die Einrichtung mehrerer Benutzer und Konfigurationsverzeichnisse als Administratorbenutzer auf dem Host-Computer. Sie können diese Einstellungen überschreiben, da Sie Jobs in Ihrer eigenen Entwicklungsfarm ausführen, auf die nur Sie zugreifen können.

Schritt 3: Jobs mit Deadline Cloud einreichen und ausführen

Gehen Sie wie folgt vor, um AWS Deadline Cloud zur Ausführung von Jobs zu verwenden. Verwenden Sie die erste AWS CloudShell Registerkarte, um Jobs an Ihre Entwicklerfarm zu senden. Verwenden Sie die zweite CloudShell Registerkarte, um die Ergebnisse des Worker-Agents anzuzeigen.

Themen

- [Reichen Sie die `simple_job` Probe ein](#)
- [Reichen Sie eine `simple_job` mit einem Parameter ein](#)
- [Erstellen Sie ein `simple_file_job`-Job-Paket mit Datei-I/O](#)

Reichen Sie die `simple_job` Probe ein

Nachdem Sie eine Farm erstellt und den Worker Agent ausgeführt haben, können Sie das `simple_job` Beispiel an Deadline Cloud senden.

Um das simple_job Beispiel an Deadline Cloud zu senden

1. Installieren und konfigurieren Sie AWS Command Line Interface (AWS CLI), falls Sie dies noch nicht getan haben. Weitere Informationen finden [Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI](#).

2. Laden Sie das Beispiel von herunter GitHub.

```
cd ~  
git clone https://github.com/aws-deadline/deadline-cloud-samples.git
```

3. Wählen Sie Ihre erste CloudShell Registerkarte und navigieren Sie dann zum Verzeichnis mit den Job-Bundle-Beispielen.

```
cd ~/deadline-cloud-samples/job_bundles/
```

4. Reichen Sie das simple_job Beispiel ein.

```
deadline bundle submit simple_job
```

5. Wählen Sie Ihre zweite CloudShell Registerkarte, um die Protokollausgabe zum AufrufenBatchGetJobEntities, Aufrufen einer Sitzung und Ausführen einer Sitzungsaktion anzuzeigen.

```
...  
[2024-03-27 16:00:21,846][INFO    ] # Session.Starting  
# [session-053d77cef82648fe2] Starting new Session.  
[queue-3ba4ff683ff54db09b851a2ed8327d7b/job-d34cc98a6e234b6f82577940ab4f76c6]  
[2024-03-27 16:00:21,853][INFO    ] # API.Req # [deadline:BatchGetJobEntity]  
resource={'farm-id': 'farm-3e24cfc9bbcd423e9c1b6754bc1',  
  'fleet-id': 'fleet-246ee60f46d44559b6cce010d05', 'worker-id':  
  'worker-75e0fce9c3c344a69bff57fcd83'} params={'identifiers': [{'jobDetails':  
  {'jobId': 'job-d34cc98a6e234b6f82577940ab4'}}]} request_url=https://  
scheduling.deadline.us-west-2.amazonaws.com/2023-10-12/farms/  
farm-3e24cfc9bbcd423e /fleets/fleet-246ee60f46d44559b1 /workers/worker-  
75e0fce9c3c344a69b /batchGetJobEntity  
[2024-03-27 16:00:22,013][INFO    ] # API.Resp # [deadline:BatchGetJobEntity](200)  
params={'entities': [{'jobDetails': {'jobId': 'job-d34cc98a6e234b6f82577940ab6',  
  'jobRunAsUser': {'posix': {'user': 'job-user', 'group': 'job-group'}},  
  'runAs': 'QUEUE_CONFIGURED_USER'}, 'logGroupName': '/aws/deadline/  
farm-3e24cfc9bbcd423e9c1b6754bc1/queue-3ba4ff683ff54db09b851a2ed83', 'parameters':
```

```
'*REDACTED*', 'schemaVersion': 'jobtemplate-2023-09']}]}, 'errors': []}
request_id=a3f55914-6470-439e-89e5-313f0c6
[2024-03-27 16:00:22,013][INFO    ] # Session.Add #
[session-053d77cef82648fea9c69827182] Appended new SessionActions.
(ActionIds: ['sessionaction-053d77cef82648fea9c69827182-0'])
[queue-3ba4ff683ff54db09b851a2ed8b/job-d34cc98a6e234b6f82577940ab6]
[2024-03-27 16:00:22,014][WARNING ] # Session.User #
[session-053d77cef82648fea9c69827182] Running as the Worker Agent's
user. (User: cloudshell-user) [queue-3ba4ff683ff54db09b851a2ed8b/job-
d34cc98a6e234b6f82577940ac6]
[2024-03-27 16:00:22,015][WARNING ] # Session.AWSCreds #
[session-053d77cef82648fea9c69827182] AWS Credentials are not available: Queue has
no IAM Role. [queue-3ba4ff683ff54db09b851a2ed8b/job-d34cc98a6e234b6f82577940ab6]
[2024-03-27 16:00:22,026][INFO    ] # Session.Logs #
[session-053d77cef82648fea9c69827182] Logs streamed to: AWS CloudWatch
Logs. (LogDestination: /aws/deadline/farm-3e24cfc9bbcd423e9c1b6754bc1/
queue-3ba4ff683ff54db09b851a2ed83/session-053d77cef82648fea9c69827181)
[queue-3ba4ff683ff54db09b851a2ed83/job-d34cc98a6e234b6f82577940ab4]
[2024-03-27 16:00:22,026][INFO    ] # Session.Logs #
[session-053d77cef82648fea9c69827182] Logs streamed to: local
file. (LogDestination: /home/cloudshell-user/demoenv-logs/
queue-3ba4ff683ff54db09b851a2ed8b/session-053d77cef82648fea9c69827182.log)
[queue-3ba4ff683ff54db09b851a2ed83/job-d34cc98a6e234b6f82577940ab4]
...
```

Note

Es werden nur die Protokollausgaben des Worker-Agenten angezeigt. Es gibt ein separates Protokoll für die Sitzung, in der der Job ausgeführt wird.

6. Wählen Sie Ihre erste Registerkarte und überprüfen Sie dann die Protokolldateien, die der Worker-Agent schreibt.
 - a. Navigieren Sie zum Protokollverzeichnis des Worker-Agenten und sehen Sie sich dessen Inhalt an.

```
cd ~/demoenv-logs
ls
```

- b. Drucken Sie die erste Protokolldatei, die der Worker-Agent erstellt.

```
cat worker-agent-bootstrap.log
```

Diese Datei enthält die Ergebnisse des Worker-Agents darüber, wie er die Deadline Cloud-API aufgerufen hat, um eine Worker-Ressource in Ihrer Flotte zu erstellen, und dann die Flottenrolle übernommen hat.

- c. Drückt die Protokolldatei, die ausgegeben wird, wenn der Worker Agent der Flotte beitrifft.

```
cat worker-agent.log
```

Dieses Protokoll enthält Ausgaben über alle Aktionen, die der Worker-Agent ausführt, aber keine Ausgaben über die Warteschlangen, in denen er Jobs ausführt, mit Ausnahme der IDs dieser Ressourcen.

- d. Drückt die Protokolldateien für jede Sitzung in einem Verzeichnis, das den gleichen Namen wie die Ressourcen-ID der Warteschlange hat.

```
cat $DEV_QUEUE_ID/session-*.log
```

Wenn der Auftrag erfolgreich ist, sieht die Ausgabe der Protokolldatei wie folgt aus:

```
cat $DEV_QUEUE_ID/$(ls -t $DEV_QUEUE_ID | head -1)
2024-03-27 16:00:22,026 WARNING Session running with no AWS Credentials.
2024-03-27 16:00:22,404 INFO
2024-03-27 16:00:22,405 INFO =====
2024-03-27 16:00:22,405 INFO ----- Running Task
2024-03-27 16:00:22,405 INFO =====
2024-03-27 16:00:22,406 INFO -----
2024-03-27 16:00:22,406 INFO Phase: Setup
2024-03-27 16:00:22,406 INFO -----
2024-03-27 16:00:22,406 INFO Writing embedded files for Task to disk.
2024-03-27 16:00:22,406 INFO Mapping: Task.File.runScript -> /sessions/
session-053d77cef82648fea9c698271812a/embedded_files/gj55_/tmp2u9yqtsz
2024-03-27 16:00:22,406 INFO Wrote: runScript -> /sessions/
session-053d77cef82648fea9c698271812a/embedded_files/gj55_/tmp2u9yqtsz
2024-03-27 16:00:22,407 INFO -----
2024-03-27 16:00:22,407 INFO Phase: Running action
2024-03-27 16:00:22,407 INFO -----
2024-03-27 16:00:22,407 INFO Running command /sessions/
session-053d77cef82648fea9c698271812a/tmpzuzxpslm.sh
```

```
2024-03-27 16:00:22,414 INFO Command started as pid: 471
2024-03-27 16:00:22,415 INFO Output:
2024-03-27 16:00:22,420 INFO Welcome to AWS Deadline Cloud!
2024-03-27 16:00:22,571 INFO
2024-03-27 16:00:22,572 INFO =====
2024-03-27 16:00:22,572 INFO ----- Session Cleanup
2024-03-27 16:00:22,572 INFO =====
2024-03-27 16:00:22,572 INFO Deleting working directory: /sessions/
session-053d77cef82648fea9c698271812a
```

7. Drückt Informationen über den Job.

```
deadline job get
```

Wenn Sie den Job weiterleiten, speichert das System ihn als Standard, sodass Sie die Job-ID nicht eingeben müssen.

Reichen Sie eine `simple_job` mit einem Parameter ein

Sie können Jobs mit Parametern einreichen. Im folgenden Verfahren bearbeiten Sie die `simple_job` Vorlage so, dass sie eine benutzerdefinierte Nachricht enthält, senden die und drucken dann die Sitzungsprotokolldatei `ausimple_job`, um die Nachricht anzuzeigen.

Um das `simple_job` Beispiel mit einem Parameter einzureichen

1. Wählen Sie Ihre erste CloudShell Registerkarte aus und navigieren Sie dann zum Verzeichnis mit den Job-Bundle-Beispielen.

```
cd ~/deadline-cloud-samples/job_bundles/
```

2. Drucken Sie den Inhalt der `simple_job` Vorlage aus.

```
cat simple_job/template.yaml
```

Der `parameterDefinitions` Abschnitt mit dem `Message` Parameter sollte wie folgt aussehen:

```
parameterDefinitions:
- name: Message
  type: STRING
  default: Welcome to AWS Deadline Cloud!
```

3. Senden Sie das `simple_job` Beispiel mit einem Parameterwert und warten Sie dann, bis die Ausführung des Jobs abgeschlossen ist.

```
deadline bundle submit simple_job \  
-p "Message=Greetings from the developer getting started guide."
```

4. Die benutzerdefinierte Meldung finden Sie in der letzten Sitzungsprotokolldatei.

```
cd ~/demoenv-logs  
cat $DEV_QUEUE_ID/$(ls -t $DEV_QUEUE_ID | head -1)
```

Erstellen Sie ein `simple_file_job`-Job-Paket mit Datei-I/O

Ein Renderjob muss die Szenendefinition lesen, daraus ein Bild rendern und dieses Bild dann in einer Ausgabedatei speichern. Sie können diese Aktion simulieren, indem Sie den Job den Hash der Eingabe berechnen lassen, anstatt ein Bild zu rendern.

Um ein `simple_file_job`-Job-Paket mit Datei-I/O zu erstellen

1. Wählen Sie Ihre erste CloudShell Registerkarte aus und navigieren Sie dann zum Verzeichnis mit den Job-Bundle-Beispielen.

```
cd ~/deadline-cloud-samples/job_bundles/
```

2. Erstellen Sie eine Kopie von `simple_job` mit dem neuen Namens `simple_file_job`.

```
cp -r simple_job simple_file_job
```

3. Bearbeiten Sie die Jobvorlage wie folgt:

Note

Wir empfehlen Ihnen, nano für diese Schritte zu verwenden. Wenn Sie lieber verwenden möchten Vim, müssen Sie den Einfügemodus über einstellen: `set paste`.

- a. Öffnen Sie die Vorlage in einem Texteditor.

```
nano simple_file_job/template.yaml
```

- b. Fügen Sie Folgendes hinzu: `responseObjectType`, und `dataFlowparameterDefinitions`.

```
- name: InFile
  type: PATH
  objectType: FILE
  dataFlow: IN
- name: OutFile
  type: PATH
  objectType: FILE
  dataFlow: OUT
```

- c. Fügen Sie den folgenden bash Skriptbefehl am Ende der Datei hinzu, die aus der Eingabedatei liest und in die Ausgabedatei schreibt.

```
# hash the input file, and write that to the output
sha256sum "{{Param.InFile}}" > "{{Param.OutFile}}"
```

Die aktualisierte `template.yaml` Version sollte genau dem Folgenden entsprechen:

```
specificationVersion: 'jobtemplate-2023-09'
name: Simple File Job Bundle Example
parameterDefinitions:
- name: Message
  type: STRING
  default: Welcome to AWS Deadline Cloud!
- name: InFile
  type: PATH
  objectType: FILE
  dataFlow: IN
- name: OutFile
  type: PATH
  objectType: FILE
  dataFlow: OUT
steps:
- name: WelcomeToDeadlineCloud
  script:
    actions:
      onRun:
        command: '{{Task.File.runScript}}'
    embeddedFiles:
      - name: runScript
        type: TEXT
```

```
runnable: true
data: |
  #!/usr/bin/env bash
  echo "{{Param.Message}}"

  # hash the input file, and write that to the output
  sha256sum "{{Param.InFile}}" > "{{Param.OutFile}}"
```

 Note

Wenn Sie den Abstand in der anpassen möchten, stellen Sie `sichertemplate.yaml`, dass Sie Leerzeichen anstelle von Einzügen verwenden.

- d. Speichern Sie die Datei und beenden Sie den Texteditor.
4. Geben Sie Parameterwerte für die Eingabe- und Ausgabedateien an, um den `simple_file_job` zu senden.

```
deadline bundle submit simple_file_job \
  -p "InFile=simple_job/template.yaml" \
  -p "OutFile=hash.txt"
```

5. Drückt Informationen über den Job.

```
deadline job get
```

- Sie werden eine Ausgabe wie die folgende sehen:

```
parameters:
  Message:
    string: Welcome to AWS Deadline Cloud!
  InFile:
    path: /local/home/cloudshell-user/BundleFiles/JobBundle-Examples/simple_job/
template.yaml
  OutFile:
    path: /local/home/cloudshell-user/BundleFiles/JobBundle-Examples/hash.txt
```

- Sie haben zwar nur relative Pfade angegeben, für die Parameter ist jedoch der vollständige Pfadsatz festgelegt. Der AWS CLI verknüpft das aktuelle Arbeitsverzeichnis mit allen Pfaden, die als Parameter bereitgestellt werden, wenn die Pfade den folgenden Typ haben `PATH`.

- Der Worker-Agent, der im anderen Terminalfenster ausgeführt wird, nimmt den Job auf und führt ihn aus. Diese Aktion erstellt die `hash.txt` Datei, die Sie mit dem folgenden Befehl anzeigen können.

```
cat hash.txt
```

Dieser Befehl druckt eine Ausgabe, die der folgenden ähnelt.

```
eea2df5d34b54be5ac34c56a24a8c237b8487231a607eaf530a04d76b89c9cd3 /local/home/  
cloudshell-user/BundleFiles/JobBundle-Examples/simple_job/template.yaml
```

Schritt 4: Jobs mit Job-Anhängen in Deadline Cloud ausführen

Viele Farmen verwenden gemeinsam genutzte Dateisysteme, um Dateien zwischen den Hosts, die Jobs einreichen, und denen, die Jobs ausführen, gemeinsam zu nutzen. Im vorherigen Beispiel wird das lokale Dateisystem `simple_file_job` beispielsweise von den AWS CloudShell Terminalfenstern gemeinsam genutzt, die auf Registerkarte eins, wo Sie den Job einreichen, und Registerkarte zwei, wo Sie den Worker-Agent ausführen, ausgeführt werden.

Ein gemeinsam genutztes Dateisystem ist vorteilhaft, wenn sich die Worker-Hosts und die Worker-Hosts im selben lokalen Netzwerk befinden. Wenn Sie Ihre Daten vor Ort in der Nähe der Workstations speichern, die darauf zugreifen, bedeutet die Verwendung einer cloudbasierten Farm, dass Sie Ihre Dateisysteme über ein VPN mit hoher Latenz gemeinsam nutzen oder Ihre Dateisysteme in der Cloud synchronisieren müssen. Keine dieser Optionen ist einfach einzurichten oder zu bedienen.

AWS Deadline Cloud bietet eine einfache Lösung mit Job-Anhängen, die E-Mail-Anhängen ähneln. Mit Job-Anhängen hängen Sie Daten an Ihren Job an. Anschließend kümmert sich Deadline Cloud um die Details der Übertragung und Speicherung Ihrer Auftragsdaten in Amazon Simple Storage Service (Amazon S3) -Buckets.

Workflows zur Erstellung von Inhalten sind oft iterativ, was bedeutet, dass ein Benutzer Jobs mit einer kleinen Teilmenge modifizierter Dateien einreicht. Da Amazon S3 S3-Buckets Auftragsanhänge in einem inhaltsadressierbaren Speicher speichern, basiert der Name jedes Objekts auf dem Hash der Objektdaten, und der Inhalt eines Verzeichnisbaums wird in einem Manifest-Dateiformat gespeichert, das an einen Auftrag angehängt ist.

Gehen Sie wie folgt vor, um Jobs mit Job-Anhängen auszuführen.

Themen

- [Fügen Sie Ihrer Warteschlange eine Konfiguration für Jobanhänge hinzu](#)
- [simple_file_jobMit Stellenanhängen einreichen](#)
- [Verstehen, wie Jobanhänge in Amazon S3 gespeichert werden](#)

Fügen Sie Ihrer Warteschlange eine Konfiguration für Jobanhänge hinzu

Um Jobanhänge in Ihrer Warteschlange zu aktivieren, fügen Sie der Warteschlangenressource in Ihrem Konto eine Konfiguration für Jobanhänge hinzu.

Um Ihrer Warteschlange eine Konfiguration für Jobanhänge hinzuzufügen

1. Installieren und konfigurieren Sie AWS Command Line Interface (AWS CLI), falls Sie dies noch nicht getan haben. Weitere Informationen finden [Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI](#).
2. Wählen Sie Ihre erste CloudShell Registerkarte und geben Sie dann einen der folgenden Befehle ein, um einen Amazon S3 S3-Bucket für Jobanhänge zu verwenden.
 - Wenn Sie noch keinen privaten Amazon S3 S3-Bucket haben, können Sie einen neuen S3-Bucket erstellen und verwenden.

```
DEV_FARM_BUCKET=$(echo $DEV_FARM_NAME \  
  | tr '[:upper:]' '[:lower:]')-$(xxd -l 16 -p /dev/urandom)  
if [ "$AWS_REGION" == "us-east-1" ]; then LOCATION_CONSTRAINT=  
else LOCATION_CONSTRAINT="--create-bucket-configuration \  
  LocationConstraint=${AWS_REGION}"  
fi  
aws s3api create-bucket \  
  $LOCATION_CONSTRAINT \  
  --acl private \  
  --bucket ${DEV_FARM_BUCKET}
```

- Wenn Sie bereits einen privaten Amazon S3 S3-Bucket haben, können Sie ihn verwenden, indem Sie ihn *MY_BUCKET_NAME* durch den Namen Ihres Buckets ersetzen.

```
DEV_FARM_BUCKET=MY_BUCKET_NAME
```

3. Nachdem Sie Ihren Amazon S3 S3-Bucket erstellt oder ausgewählt haben, fügen Sie den Bucket-Namen hinzu, ~/ .bashrc um den Bucket für andere Terminalsitzungen verfügbar zu machen.

```
echo "DEV_FARM_BUCKET=${DEV_FARM_BUCKET}" >> ~/.bashrc
```

4. Erstellen Sie eine AWS Identity and Access Management (IAM-) Rolle für die Warteschlange.

```
aws iam create-role --role-name "${DEV_FARM_NAME}QueueRole" \
  --assume-role-policy-document \
    '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
            "Service": "credentials.deadline.amazonaws.com"
          },
          "Action": "sts:AssumeRole"
        }
      ]
    }'
```

```
aws iam put-role-policy \
  --role-name "${DEV_FARM_NAME}QueueRole" \
  --policy-name S3BucketsAccess \
  --policy-document \
    '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Action": [
            "s3:GetObject*",
            "s3:GetBucket*",
            "s3:List*",
            "s3:DeleteObject*",
            "s3:PutObject",
            "s3:PutObjectLegalHold",
            "s3:PutObjectRetention",
            "s3:PutObjectTagging",
            "s3:PutObjectVersionTagging",
            "s3:Abort*"
          ],
          "Resource": [
            "arn:aws:s3:::'$DEV_FARM_BUCKET'",
            "arn:aws:s3:::'$DEV_FARM_BUCKET'/*"
          ]
        }
      ]
    }'
```

```

        "Effect": "Allow"
      }
    ]
  }'

```

5. Aktualisieren Sie Ihre Warteschlange so, dass sie die Einstellungen für Jobanhänge und die IAM-Rolle enthält.

```

QUEUE_ROLE_ARN="arn:aws:iam::$(aws sts get-caller-identity \
  --query "Account" --output text):role/${DEV_FARM_NAME}QueueRole"
aws deadline update-queue \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --role-arn $QUEUE_ROLE_ARN \
  --job-attachment-settings \
  '{
    "s3BucketName": "'$DEV_FARM_BUCKET'",
    "rootPrefix": "JobAttachments"
  }'

```

6. Vergewissern Sie sich, dass Sie Ihre Warteschlange aktualisiert haben.

```
deadline queue get
```

Es wird eine Ausgabe wie die folgende angezeigt:

```

...
jobAttachmentSettings:
  s3BucketName: DEV_FARM_BUCKET
  rootPrefix: JobAttachments
roleArn: arn:aws:iam::ACCOUNT_NUMBER:role/DeveloperFarmQueueRole
...

```

simple_file_job Mit Stellenanhängen einreichen

Wenn Sie Job-Anhänge verwenden, müssen Job-Bundles Deadline Cloud genügend Informationen liefern, um den Datenfluss des Jobs zu bestimmen, z. B. anhand von PATH Parametern. Im Fall von haben Sie die `template.yaml` Datei bearbeitet `simple_file_job`, um Deadline Cloud mitzuteilen, dass sich der Datenfluss in der Eingabe- und Ausgabedatei befindet.

Nachdem Sie die Konfiguration für Jobanhänge zu Ihrer Warteschlange hinzugefügt haben, können Sie das `simple_file_job`-Beispiel mit Jobanhängen einreichen. Nachdem Sie dies getan haben, können Sie sich die Protokollierung und die Jobausgabe ansehen, um sich zu vergewissern, dass das `simple_file_job` mit den Stellenanhängen funktioniert.

Um das `simple_file_job`-Job-Paket mit Job-Anhängen einzureichen

1. Wählen Sie Ihre erste CloudShell Registerkarte und öffnen Sie dann das Verzeichnis `JobBundle-Samples`

2.

```
cd ~/AmazonDeadlineCloud-DocumentationAndSamples/JobBundle-Samples
```

3. Reichen Sie `simple_file_job` in die Warteschlange ein. Wenn Sie aufgefordert werden, den Upload zu bestätigen, geben Sie ein. `y`

```
deadline bundle submit simple_file_job \  
  -p InFile=simple_job/template.yaml \  
  -p OutFile=hash-jobattachments.txt
```

4. Wählen Sie Ihre zweite CloudShell Registerkarte, um die Ausgabe des Sitzungsprotokolls der Jobanhänge für die Datenübertragung anzuzeigen.

```
JOB_ID=$(deadline config get defaults.job_id)  
SESSION_ID=$(aws deadline list-sessions \  
  --farm-id $DEV_FARM_ID \  
  --queue-id $DEV_QUEUE_ID \  
  --job-id $JOB_ID \  
  --query "sessions[0].sessionId" \  
  --output text)  
cat ~/demoenv-logs/$DEV_QUEUE_ID/$SESSION_ID.log
```

5. Listet die Sitzungsaktionen auf, die innerhalb der Sitzung ausgeführt wurden.

```
aws deadline list-session-actions \  
  --farm-id $DEV_FARM_ID \  
  --queue-id $DEV_QUEUE_ID \  
  --job-id $JOB_ID \  
  --session-id $SESSION_ID
```

Es wird eine Ausgabe wie die folgende angezeigt:

```
{
  "sessionactions": [
    {
      "sessionId": "sessionaction-123-0",
      "status": "SUCCEEDED",
      "startedAt": "<timestamp>",
      "endedAt": "<timestamp>",
      "progressPercent": 100.0,
      "definition": {
        "syncInputJobAttachments": {}
      }
    },
    {
      "sessionId": "sessionaction-123-1",
      "status": "SUCCEEDED",
      "startedAt": "<timestamp>",
      "endedAt": "<timestamp>",
      "progressPercent": 100.0,
      "definition": {
        "taskRun": {
          "taskId": "task-abc-0",
          "stepId": "step-def"
        }
      }
    }
  ]
}
```

Bei der ersten Sitzungsaktion wurden die Eingabeauftragsanhänge heruntergeladen, während bei der zweiten Aktion die Aufgabe wie zuvor ausgeführt und anschließend die Anlagen des Ausgabeauftrags hochgeladen wurden.

6. Listet das Ausgabeverzeichnis auf.

```
ls *.txt
```

Ausgabe, wie sie gezeigt wird, zeigt `hash.txt` existiert, `hash-jobattachments.txt` existiert aber nicht.

7. Laden Sie die Ausgabe des letzten Jobs herunter.

```
deadline job download-output
```

8. Sehen Sie sich die Ausgabe der heruntergeladenen Datei an.

```
cat hash-jobattachments.txt
```

Es wird eine Ausgabe wie die folgende angezeigt:

```
eea2df5d34b54be5ac34c56a24a8c237b8487231a607eaf530a04d76b89c9cd3 /tmp/openjd/  
session-123/assetroot-abc/simple_job/template.yaml
```

Verstehen, wie Jobanhänge in Amazon S3 gespeichert werden

Sie können die AWS Command Line Interface (AWS CLI) verwenden, um Daten für Jobanhänge hoch- oder herunterzuladen, die in Amazon S3 S3-Buckets gespeichert sind. Wenn Sie wissen, wie Deadline Cloud Jobanhänge auf Amazon S3 speichert, hilft Ihnen das bei der Entwicklung von Workloads und Pipeline-Integrationen.

Um zu überprüfen, wie Deadline Cloud-Jobanhänge in Amazon S3 gespeichert werden

1. Wählen Sie Ihren ersten CloudShell Tab und öffnen Sie dann das Verzeichnis mit den Job-Bundle-Beispielen.

```
cd ~/AmazonDeadlineCloud-DokumentationAndSamples/JobBundle-Samples
```

2. Untersuchen Sie die Jobeigenschaften.

```
deadline job get
```

Es wird eine Ausgabe wie die folgende angezeigt:

```
parameters:  
  Message:  
    string: Welcome to Amazon Deadline Cloud!  
  InFile:  
    path: /home/cloudshell-user/AmazonDeadlineCloud-DokumentationAndSamples/  
JobBundle-Samples/simple_job/template.yaml  
  OutFile:  
    path: /home/cloudshell-user/AmazonDeadlineCloud-DokumentationAndSamples/  
JobBundle-Samples/hash-jobattachments.txt  
attachments:
```

```

manifests:
- rootPath: /home/cloudshell-user/AmazonDeadlineCloud-DokumentationAndSamples/
  JobBundle-Samples
  rootPathFormat: posix
  outputRelativeDirectories:
  - .
  inputManifestPath: farm-3040c59a5b9943d58052c29d907a645d/queue-
  cde9977c9f4d4018a1d85f3e6c1a4e6e/Inputs/
  f46af01ca8904cd8b514586671c79303/0d69cd94523ba617c731f29c019d16e8_input.xxh128
  inputManifestHash: f95ef91b5dab1fc1341b75637fe987ee
  fileSystem: COPIED

```

Das Feld „Anlagen“ enthält eine Liste von Manifeststrukturen, die Eingabe- und Ausgabedatenpfade beschreiben, die der Job bei seiner Ausführung verwendet. Sehen Sie `rootPath` sich den lokalen Verzeichnispfad auf dem Computer an, der den Job eingereicht hat. Das Amazon S3 S3-Objektsuffix, das eine Manifestdatei enthält, finden Sie unter `inputManifestFile`. Die Manifestdatei enthält Metadaten für einen Verzeichnisbaum-Snapshot der Eingabedaten des Jobs.

3. Drucken Sie das Amazon S3 S3-Manifest-Objekt hübsch aus, um die Eingabeverzeichnisstruktur für den Job zu sehen.

```

MANIFEST_SUFFIX=$(aws deadline get-job \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --job-id $JOB_ID \
  --query "attachments.manifests[0].inputManifestPath" \
  --output text)
aws s3 cp s3://$DEV_FARM_BUCKET/JobAttachments/Manifests/$MANIFEST_SUFFIX - | jq .

```

Es wird eine Ausgabe wie die folgende angezeigt:

```

{
  "hashAlg": "xxh128",
  "manifestVersion": "2023-03-03",
  "paths": [
    {
      "hash": "2ec297b04c59c4741ed97ac8fb83080c",
      "mtime": 1698186190000000,
      "path": "simple_job/template.yaml",
      "size": 445
    }
  ]
}

```

```

    ],
    "totalSize": 445
  }

```

4. Konstruieren Sie das Amazon S3 S3-Präfix, das die Manifeste für die Anlagen des Ausgabeauftrags enthält, und listen Sie das Objekt darunter auf.

```

SESSION_ACTION=$(aws deadline list-session-actions \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --job-id $JOB_ID \
  --session-id $SESSION_ID \
  --query "sessionActions[?definition.taskRun != null] | [0]")
STEP_ID=$(echo $SESSION_ACTION | jq -r .definition.taskRun.stepId)
TASK_ID=$(echo $SESSION_ACTION | jq -r .definition.taskRun.taskId)
TASK_OUTPUT_PREFIX=JobAttachments/Manifests/$DEV_FARM_ID/$DEV_QUEUE_ID/$JOB_ID/
$STEP_ID/$TASK_ID/
aws s3api list-objects-v2 --bucket $DEV_FARM_BUCKET --prefix $TASK_OUTPUT_PREFIX

```

Auf die Ausgabeauftragsanhänge wird nicht direkt von der Jobressource verwiesen, sondern sie werden stattdessen auf der Grundlage der Farm-Ressourcen-IDs in einem Amazon S3 S3-Bucket platziert.

5. Rufen Sie den neuesten Manifestobjektschlüssel für die spezifische Sitzungsaktions-ID ab und drucken Sie dann die Manifestobjekte hübsch aus.

```

SESSION_ACTION_ID=$(echo $SESSION_ACTION | jq -r .sessionId)
MANIFEST_KEY=$(aws s3api list-objects-v2 \
  --bucket $DEV_FARM_BUCKET \
  --prefix $TASK_OUTPUT_PREFIX \
  --query "Contents[*].Key" --output text \
  | grep $SESSION_ACTION_ID \
  | sort | tail -1)
MANIFEST_OBJECT=$(aws s3 cp s3://$DEV_FARM_BUCKET/$MANIFEST_KEY -)
echo $MANIFEST_OBJECT | jq .

```

In der Ausgabe werden Eigenschaften der Datei `hash-jobattachments.txt` wie die folgenden angezeigt:

```

{
  "hashAlg": "xxh128",
  "manifestVersion": "2023-03-03",

```

```
"paths": [  
  {  
    "hash": "f60b8e7d0fabf7214ba0b6822e82e08b",  
    "mtime": 1698785252554950,  
    "path": "hash-jobattachments.txt",  
    "size": 182  
  }  
],  
"totalSize": 182  
}
```

Ihr Job wird nur ein einziges Manifest-Objekt pro Aufgabenausführung haben, aber im Allgemeinen ist es möglich, mehr Objekte pro Aufgabenausführung zu haben.

6. Zeigen Sie die für den Inhalt adressierbare Amazon S3 S3-Speicherausgabe unter dem Präfix an. Data

```
FILE_HASH=$(echo $MANIFEST_OBJECT | jq -r .paths[0].hash)  
FILE_PATH=$(echo $MANIFEST_OBJECT | jq -r .paths[0].path)  
aws s3 cp s3://$DEV_FARM_BUCKET/JobAttachments/Data/$FILE_HASH -
```

Es wird eine Ausgabe wie die folgende angezeigt:

```
eea2df5d34b54be5ac34c56a24a8c237b8487231a607eaf530a04d76b89c9cd3 /tmp/openjd/  
session-123/assetroot-abc/simple_job/template.yaml
```

Schritt 5: Fügen Sie Ihrer Entwicklerfarm in Deadline Cloud eine serviceverwaltete Flotte hinzu

AWS CloudShell bietet nicht genügend Rechenkapazität, um größere Workloads zu testen. Es ist auch nicht für Jobs konfiguriert, die Aufgaben auf mehrere Worker-Hosts verteilen.

Anstatt zu verwenden CloudShell, können Sie Ihrer Entwicklerfarm eine Auto Scaling Service Managed Fleet (SMF) hinzufügen. Ein SMF bietet ausreichend Rechenkapazität für größere Workloads und kann Jobs bewältigen, bei denen Jobaufgaben auf mehrere Worker-Hosts verteilt werden müssen. Der Scheduler verwendet sowohl den SMF- als auch den CMF-Worker, um Jobs auszuführen, sofern Sie den CMF-Worker nicht herunterfahren.

So fügen Sie Ihrer Entwicklerfarm eine vom Service verwaltete Flotte hinzu

1. Installieren und konfigurieren Sie AWS Command Line Interface (AWS CLI), falls Sie dies noch nicht getan haben. Weitere Informationen finden [Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI](#).
2. Wählen Sie Ihre erste AWS CloudShell Registerkarte, erstellen Sie dann die vom Service verwaltete Flotte und fügen Sie deren Flotten-ID hinzu `.bashrc`. Diese Aktion macht es für andere Terminalansichten verfügbar.

```
FLEET_ROLE_ARN="arn:aws:iam::$(aws sts get-caller-identity \
  --query "Account" --output text):role/${DEV_FARM_NAME}FleetRole"
aws deadline create-fleet \
  --farm-id $DEV_FARM_ID \
  --display-name "$DEV_FARM_NAME SMF" \
  --role-arn $FLEET_ROLE_ARN \
  --max-worker-count 5 \
  --configuration \
    '{
      "serviceManagedEc2": {
        "instanceCapabilities": {
          "vCpuCount": {
            "min": 2,
            "max": 4
          },
          "memoryMiB": {
            "min": 512
          },
          "osFamily": "linux",
          "cpuArchitectureType": "x86_64"
        },
        "instanceMarketOptions": {
          "type": "spot"
        }
      }
    }'
```

```
echo "DEV_SMF_ID=$(aws deadline list-fleets \
  --farm-id $DEV_FARM_ID \
  --query "fleets[?displayName=='$DEV_FARM_NAME SMF'].fleetId \
  | [0]" --output text)" >> ~/.bashrc
source ~/.bashrc
```

3. Ordnen Sie das SMF Ihrer Warteschlange zu.

```
aws deadline create-queue-fleet-association \  
  --farm-id $DEV_FARM_ID \  
  --queue-id $DEV_QUEUE_ID \  
  --fleet-id $DEV_SMF_ID
```

4.

Note

Der Scheduler verwendet sowohl den SMF- als auch den CMF-Worker, um Jobs auszuführen, sofern Sie den CMF-Worker nicht herunterfahren.

In die Warteschlange einreichensimple_file_job. Wenn Sie aufgefordert werden, den Upload zu bestätigen, geben Sie einy.

```
deadline bundle submit simple_file_job \  
  -p InFile=simple_job/template.yaml \  
  -p OutFile=hash-jobattachments.txt
```

5. Vergewissern Sie sich, dass das SMF ordnungsgemäß funktioniert.

```
deadline fleet get
```

- Es kann einige Minuten dauern, bis der Mitarbeiter anfängt.
- Die queueFleetAssociationsStatus für Ihren Kunden verwaltete Flotte und die vom Service verwaltete Flotte werden es seinACTIVE.
- Das SMF autoScalingStatus wird von GROWING bis STEADY geändert.

Ihr Status wird etwa wie folgt aussehen:

```
fleetId: fleet-2cc78e0dd3f04d1db427e7dc1d51ea44  
farmId: farm-63ee8d77cdab4a578b685be8c5561c4a  
displayName: DeveloperFarm SMF  
description: ''  
status: ACTIVE  
autoScalingStatus: STEADY  
targetWorkerCount: 0  
workerCount: 0
```

```
minWorkerCount: 0
maxWorkerCount: 5
```

6. Sehen Sie sich das Protokoll für den Job an, den Sie eingereicht haben. Dieses Protokoll wird in einem Protokoll in Amazon CloudWatch Logs gespeichert, nicht im CloudShell Dateisystem.

```
JOB_ID=$(deadline config get defaults.job_id)
SESSION_ID=$(aws deadline list-sessions \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --job-id $JOB_ID \
  --query "sessions[0].sessionId" \
  --output text)
aws logs tail /aws/deadline/$DEV_FARM_ID/$DEV_QUEUE_ID \
  --log-stream-names $SESSION_ID
```

Schritt 6: Bereinigen Sie Ihre Farmressourcen in Deadline Cloud

Um neue Workloads und Pipeline-Integrationen zu entwickeln und zu testen, können Sie weiterhin die Deadline Cloud-Entwicklerfarm verwenden, die Sie für dieses Tutorial erstellt haben. Wenn Sie Ihre Entwicklerfarm nicht mehr benötigen, können Sie ihre Ressourcen, einschließlich Farm-, Flotten-, Warteschlangen-, AWS Identity and Access Management (IAM-) Rollen und Logs, in Amazon CloudWatch Logs löschen. Nachdem Sie diese Ressourcen gelöscht haben, müssen Sie das Tutorial erneut beginnen, um die Ressourcen verwenden zu können. Weitere Informationen finden Sie unter [Einrichtung einer Entwickler-Workstation für Deadline Cloud](#).

Um die Ressourcen der Entwicklerfarm zu bereinigen

1. Installieren und konfigurieren Sie AWS Command Line Interface (AWS CLI), falls Sie dies noch nicht getan haben. Weitere Informationen finden [Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI](#).
2. Wählen Sie Ihre erste CloudShell Registerkarte und beenden Sie dann alle Verbindungen zwischen Warteschlange und Flotte für Ihre Warteschlange.

```
FLEETS=$(aws deadline list-queue-fleet-associations \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --query "queueFleetAssociations[].fleetId" \
  --output text)
```

```
for FLEET_ID in $FLEETS; do
  aws deadline update-queue-fleet-association \
    --farm-id $DEV_FARM_ID \
    --queue-id $DEV_QUEUE_ID \
    --fleet-id $FLEET_ID \
    --status STOP_SCHEDULING_AND_CANCEL_TASKS
done
```

3. Listet die Flottenzuordnungen der Warteschlange auf.

```
aws deadline list-queue-fleet-associations \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID
```

Möglicherweise müssen Sie den Befehl erneut ausführen, bis die Ausgabe meldet "status": "STOPPED", dann können Sie mit dem nächsten Schritt fortfahren. Dieser Vorgang kann mehrere Minuten in Anspruch nehmen.

```
{
  "queueFleetAssociations": [
    {
      "queueId": "queue-abcdefgh01234567890123456789012id",
      "fleetId": "fleet-abcdefgh01234567890123456789012id",
      "status": "STOPPED",
      "createdAt": "2023-11-21T20:49:19+00:00",
      "createdBy": "arn:aws:sts::123456789012:assumed-role/RoleToBeAssumed/MySessionName",
      "updatedAt": "2023-11-21T20:49:38+00:00",
      "updatedBy": "arn:aws:sts::123456789012:assumed-role/RoleToBeAssumed/MySessionName"
    },
    {
      "queueId": "queue-abcdefgh01234567890123456789012id",
      "fleetId": "fleet-abcdefgh01234567890123456789012id",
      "status": "STOPPED",
      "createdAt": "2023-11-21T20:32:06+00:00",
      "createdBy": "arn:aws:sts::123456789012:assumed-role/RoleToBeAssumed/MySessionName",
      "updatedAt": "2023-11-21T20:49:39+00:00",
      "updatedBy": "arn:aws:sts::123456789012:assumed-role/RoleToBeAssumed/MySessionName"
    }
  ]
}
```

```
]
}
```

4. Löschen Sie alle Verknüpfungen zwischen Warteschlange und Flotte für Ihre Warteschlange.

```
for FLEET_ID in $FLEETS; do
    aws deadline delete-queue-fleet-association \
        --farm-id $DEV_FARM_ID \
        --queue-id $DEV_QUEUE_ID \
        --fleet-id $FLEET_ID
done
```

5. Löschen Sie alle Flotten, die Ihrer Warteschlange zugeordnet sind.

```
for FLEET_ID in $FLEETS; do
    aws deadline delete-fleet \
        --farm-id $DEV_FARM_ID \
        --fleet-id $FLEET_ID
done
```

6. Löscht die Warteschlange.

```
aws deadline delete-queue \
    --farm-id $DEV_FARM_ID \
    --queue-id $DEV_QUEUE_ID
```

7. Löschen Sie die Farm.

```
aws deadline delete-farm \
    --farm-id $DEV_FARM_ID
```

8. Löschen Sie andere AWS Ressourcen für Ihre Farm.

- a. Löschen Sie die Flottenrolle AWS Identity and Access Management (IAM).

```
aws iam delete-role-policy \
    --role-name "${DEV_FARM_NAME}FleetRole" \
    --policy-name WorkerPermissions
aws iam delete-role \
    --role-name "${DEV_FARM_NAME}FleetRole"
```

- b. Löschen Sie die Warteschlangen-IAM-Rolle.

```
aws iam delete-role-policy \  
  --role-name "${DEV_FARM_NAME}QueueRole" \  
  --policy-name S3BucketsAccess  
aws iam delete-role \  
  --role-name "${DEV_FARM_NAME}QueueRole"
```

- c. Löschen Sie die Amazon CloudWatch Logs-Protokollgruppen. Jede Warteschlange und Flotte hat ihre eigene Protokollgruppe.

```
aws logs delete-log-group \  
  --log-group-name "/aws/deadline/$DEV_FARM_ID/$DEV_QUEUE_ID"  
aws logs delete-log-group \  
  --log-group-name "/aws/deadline/$DEV_FARM_ID/$DEV_CMF_ID"  
aws logs delete-log-group \  
  --log-group-name "/aws/deadline/$DEV_FARM_ID/$DEV_SMF_ID"
```

Richten Sie Deadline Cloud-Einreicher ein

Dieser Prozess richtet sich an Administratoren und Künstler, die den AWS Deadline Cloud Submitter installieren, einrichten und starten möchten. Ein Deadline Cloud-Einreicher ist ein DCC-Plugin (Digital Content Creation). Künstler verwenden es, um Jobs über eine DCC-Schnittstelle eines Drittanbieters einzureichen, mit der sie vertraut sind.

Note

Dieser Vorgang muss auf allen Workstations abgeschlossen sein, die Künstler für das Einreichen von Renderings verwenden werden.

Themen

- [Schritt 1: Installieren Sie den Deadline Cloud Submitter](#)
- [Schritt 2: Deadline Cloud Monitor installieren und einrichten](#)
- [Schritt 3: Starten Sie den Deadline Cloud Submitter](#)

Schritt 1: Installieren Sie den Deadline Cloud Submitter

Die folgenden Abschnitte führen Sie durch die Schritte zur Installation des Deadline Cloud-Einreichers.

Laden Sie das Installationsprogramm für den Submitter herunter

Bevor Sie den Deadline Cloud Submitter installieren können, müssen Sie den Installer für den Submitter herunterladen. Derzeit unterstützt das Deadline Cloud-Installationsprogramm für Submitter nur und. Windows Linux

1. [Melden Sie sich bei der Deadline Cloud-Konsole an AWS Management Console und öffnen Sie sie.](#)
2. Wählen Sie im seitlichen Navigationsbereich die Option Downloads aus.
3. Suchen Sie den Installationsbereich Deadline Cloud Submitter.
4. Wählen Sie das Installationsprogramm für das Betriebssystem Ihres Computers aus und klicken Sie dann auf Herunterladen.

(Optional) Überprüfen Sie die Echtheit der heruntergeladenen Software

Um zu überprüfen, ob die heruntergeladene Software authentisch ist, verwenden Sie das folgende Verfahren für entweder Windows oderLinux.

Note

Sie können diese Anweisungen verwenden, um zuerst das Installationsprogramm zu verifizieren und dann den Deadline Cloud-Monitor zu verifizieren, nachdem Sie ihn im nächsten Abschnitt heruntergeladen haben (Schritt 2).

Windows

Gehen Sie wie folgt vor, um die Echtheit Ihrer heruntergeladenen Dateien zu überprüfen.

1. Ersetzen Sie den Befehl im folgenden Befehl *file* durch die Datei, die Sie überprüfen möchten. z. B. **C:\PATH\TO\MY\DeadlineCloudSubmitter-windows-x64-installer.exe** . Ersetzen Sie es außerdem *signtool-sdk-version* durch die Version des installierten SignTool SDK. z. B. **10.0.22000.0**.

```
"C:\Program Files (x86)\Windows Kits\10\bin\signtool-sdk-  
version\x86\signtool.exe" verify /vfile
```

2. Sie können beispielsweise die Installationsdatei für den Deadline Cloud-Absender überprüfen, indem Sie den folgenden Befehl ausführen:

```
"C:\Program Files (x86)\Windows Kits\10\bin  
\10.0.22000.0\x86\signtool.exe" verify /v DeadlineCloudSubmitter-  
windows-x64-installer.exe
```

Linux

Verwenden Sie das gpg Befehlszeilentool, um die Echtheit Ihrer heruntergeladenen Dateien zu überprüfen.

1. Importieren Sie den OpenPGP Schlüssel für das Deadline Cloud Submitter-Installationsprogramm, indem Sie den folgenden Befehl ausführen:

```
gpg --import --armor <<EOF  
-----BEGIN PGP PUBLIC KEY BLOCK-----  
  
mQINBGX6GQsBEADduUtJgqSXI+q7606fsFwEYKmbnlyL0xKvlq32EZuyv0otZo5L  
le4m5Gg52AzrvPvDiUTLooAlvYeozaYyirIGsK08Ydz0Ftdjroiuh/mw9JSJDJRI  
rnRn5yKet1JFezkjopA3pjsTBP6lW/mb1bDBDEwwwtH0x91V7A03FJ9T7Uzu/qSh  
q0/UYdkafro3cPASvkkqgDt2tCvURfBcUCAjZVfCfLZcVD5iwXacxvKsxxS/e7kuVv  
I1+VGT8Hj8XzWYhjCZx0LZk/fvpYPMYEEujN0fYUp6RtMIXve0C9awwMCy5nBG2J  
eE2015DsCpTaBd4Fdr3LWcSs8JFA/YfP9auL3Ncz0ozPoVJt+fw8CB1VIX00J715  
hvHDjcC+5v0wxqAlMG6+f/SX7CT8FXK+L3i0J5gBYUNXqHSxUdv8kt76/KVmQa1B  
Ak1+MPKpMq+1hw++S3G/1XqwWaDNQbRRw7dSZHymQVXvPp1nscq3hV7K10M+6s6g  
1g4mvFY4lf6DhptwZLWYQXU8rBQpojvQfiSmDFrFPWFi5BexesuVnkGIo1Qok1Kx  
AVUSdJPVEJCteyy7td4FPhBaSqT5vW3+ANbr9b/uoRYWJvn17dN0cc9HuRh/Ai+I  
nkfECo2WUDLZ0fEKGjGyFX+todWvJXjvc5kmE9Ty5vJp+M9Vvb8jd6t+mwARAQAB  
tCxBV1MgRGVhZGxpbnUgQ2xvdWQgPGF3cy1kZWFKbGluZUBhbWF6b24uY29tPokC  
VwQTAQgAQRyhBLhAwIwpqQeWoHH6pfbNP0a3bzzvBQJ1+hkLAXsvBAUJA8JnAAUL  
CQgHAgIiAgYVCgkICwIDFgIBAh4HAheAAAoJEPbNP0a3bzzvKswQAjXzKSAY8sY8  
F6Eas2oYwIDDDurs8FiEnFghjUE06MTt9AykF/jw+CQg2UzFtEy0bHBymhgmhXE  
3buVeom96tgM3ZDfZu+sxi5pGX6oAQnZ6riztN+VpkpQmLgwtMGpSML13KLwnv2k  
WK8mrR/fPMkfdaewB7A6RIUYiW33GAL4KfMIs8/vIwIJw99NxHpZQVoU6dFpuDtE  
10uxGcCqGJ7mAmo6H/YawSNp2Ns80gyqIKYo7o3LJ+WRroIRlQyctq8gnR9JvYXX  
42ASqLq5+0XKo4qh81b1XKYqtc176BbbSNFjWnzIQgKDgNiHFZCdc0VgqDhw015r  
NICbqqwwNLj/Fr2kecYx180Ktp10j00w5I0yh3bf3MVGWnYRdjvA1v+/CO+55N4g
```

```

z0kf50Lcdu5RtqV10XBCifn28pecqPaSdYcssYSR15DLiFktGbNzTGcZZwITTKQc
af8PPdTGtnnb6P+cdbW3bt9MvtN5/dgSHLThnS8MPEuNCtkTnpXshuVuBGgwBMdb
qUC+HjqvhZzbwns8dr5WI+6HWNBFgGANN6ageY158vVp0UkuNP8wcWjRARciHXZx
ku6W2jPTHDWGNrBQ02Fx7fd2QYJheIPPAShHcfJ0+XgWcof45D0vAxAJ8gGg9Eq+
gFWhsx4NSHn2gh1gDZ410u/4exJ11wPM
=uVaX
-----END PGP PUBLIC KEY BLOCK-----
EOF

```

2. Stellen Sie fest, ob Sie dem Schlüssel vertrauen möchten OpenPGP. Bei der Entscheidung, ob dem oben genannten Schlüssel vertraut werden soll, sollten Sie unter anderem die folgenden Faktoren berücksichtigen:
 - Die Internetverbindung, mit der Sie den GPG-Schlüssel von dieser Website abgerufen haben, ist sicher.
 - Das Gerät, mit dem Sie auf diese Website zugreifen, ist sicher.
 - AWS hat Maßnahmen ergriffen, um das Hosting des OpenPGP öffentlichen Schlüssels auf dieser Website zu sichern.
3. Wenn Sie sich dafür entscheiden, dem OpenPGP Schlüssel zu vertrauen, bearbeiten Sie den Schlüssel so, dass er vertrauenswürdig ist. Gehen Sie dabei wie im folgenden Beispiel vor:

```

$ gpg --edit-key 0xB840C08C29A90796A071FAA5F6CD3CE6B76F3CEF

gpg (GnuPG) 2.0.22; Copyright (C) 2013 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

pub 4096R/4BF0B8D2  created: 2023-06-23  expires: 2025-06-22  usage: SCEA
                        trust: unknown      validity: unknown
[ unknown] (1). AWS Deadline Cloud example@example.com

gpg> trust
pub 4096R/4BF0B8D2  created: 2023-06-23  expires: 2025-06-22  usage: SCEA
                        trust: unknown      validity: unknown
[ unknown] (1). AWS Deadline Cloud aws-deadline@amazon.com

Please decide how far you trust this user to correctly verify other users'
keys

```

```
(by looking at passports, checking fingerprints from different sources,
etc.)
```

```
1 = I don't know or won't say
2 = I do NOT trust
3 = I trust marginally
4 = I trust fully
5 = I trust ultimately
m = back to the main menu
```

```
Your decision? 5
```

```
Do you really want to set this key to ultimate trust? (y/N) y
```

```
pub 4096R/4BF0B8D2  created: 2023-06-23  expires: 2025-06-22  usage: SCEA
                        trust: ultimate      validity: unknown
```

```
[ unknown] (1). AWS Deadline Cloud aws-deadline@amazon.com
```

```
Please note that the shown key validity is not necessarily correct
unless you restart the program.
```

```
gpg> quit
```

4. Überprüfen Sie das Installationsprogramm

Gehen Sie wie folgt vor, um das Installationsprogramm zu verifizieren:

- a. Kehren Sie zur Download-Seite der Deadline [Cloud-Konsole](#) zurück und laden Sie die Signaturdatei für das Deadline Cloud-Installationsprogramm für den Absender herunter.
- b. Überprüfen Sie die Signatur des Deadline Cloud-Installationsprogramms für Submitter, indem Sie Folgendes ausführen:

```
gpg --verify ./DeadlineCloudSubmitter-linux-x64-
installer.run.sig ./DeadlineCloudSubmitter-linux-x64-
installer.run
```

5. Überprüfen Sie den Deadline Cloud-Monitor

Note

Sie können den Download des Deadline Cloud-Monitors mithilfe von Signaturdateien oder plattformspezifischen Methoden überprüfen. Plattformspezifische Methoden

finden Sie auf der Linux (DEB) Registerkarte oder auf der Linux (Applmage) Registerkarte, die auf Ihrem heruntergeladenen Dateityp basiert.

Gehen Sie wie folgt vor, um die Desktop-Anwendung Deadline Cloud Monitor anhand von Signaturdateien zu verifizieren:

- a. Kehren Sie zur Downloadseite der Deadline [Cloud-Konsole](#) zurück, laden Sie die entsprechende SIG-Datei herunter und führen Sie sie dann aus

Für .deb:

```
gpg --verify ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.deb.sig ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.deb
```

Für. Applmage:

```
gpg --verify ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.AppImage.sig ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.AppImage
```

- b. Vergewissern Sie sich, dass die Ausgabe wie folgt aussieht:

```
gpg: Signature made Mon Apr 1 21:10:14 2024 UTC
```

```
gpg: using RSA key B840C08C29A90796A071FAA5F6CD3CE6B7
```

Wenn die Ausgabe den Ausdruck enthält, bedeutet dies `Good signature from "AWS Deadline Cloud"`, dass die Signatur erfolgreich verifiziert wurde und Sie das Installationsskript für den Deadline Cloud-Monitor ausführen können.

Linux (DEB)

Um Pakete zu verifizieren, die Linux eine .deb-Binärdatei verwenden, führen Sie zunächst die Schritte 1—3 Linux auf der Registerkarte aus.

dpkg ist das zentrale Paketverwaltungswerkzeug in den meisten debian basierten Linux Distributionen. Sie können die .deb-Datei mit dem Tool überprüfen.

1. Laden Sie von der Downloadseite der Deadline [Cloud-Konsole](#) die .deb-Datei für den Deadline Cloud-Monitor herunter.
2. **<APP_VERSION>** Ersetzen Sie sie durch die Version der .deb-Datei, die Sie verifizieren möchten.

```
dpkg-sig --verify deadline-cloud-monitor_<APP_VERSION>_amd64.deb
```

3. Die Ausgabe wird wie folgt aussehen:

```
Processing deadline-cloud-monitor_1.1.1_amd64.deb... GOODSIG
_gpgbuilder B840C08C29A90796A071FAA5F6CD3C 171200
```

4. Um die .deb-Datei zu überprüfen, stellen Sie sicher, dass sie in der Ausgabe vorhanden GOODSIG ist.

Linux (AppImage)

Um Pakete zu verifizieren, die eine Linux verwenden. AppImage Binär, führen Sie zuerst die Schritte 1 bis 3 auf der Linux Registerkarte aus.

1. Laden Sie von der Downloadseite der Deadline [Cloud-Konsole](#) den Deadline Cloud-Monitor herunter. AppImage datei.
2. Um <APP_VERSION> durch die Version von zu ersetzen. AppImage Führen Sie die folgenden Schritte durch, um die Datei zu verifizieren:

- a. Schreiben Sie die Signatur von. AppImage Datei in eine .sig-Datei.

```
./deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage
--appimage-signature > ./deadline-cloud-
monitor_<APP_VERSION>_amd64_.AppImage.sig
```

- b. Verwenden Sie die generierte .sig-Datei, um dies mit dem folgenden Befehl zu überprüfen.

```
gpg --verify ./deadline-cloud-
monitor_<APP_VERSION>_amd64.AppImage.sig
```

- c. (Optional) Wenn der Fehler „Zugriff verweigert“ angezeigt wird, verwenden Sie den folgenden Befehl, um die Ausführungsberechtigung hinzuzufügen.

```
chmod +x ./deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage
```

- d. Vergewissern Sie sich, dass die Ausgabe etwa wie folgt aussieht:

```
gpg: Signature made Mon Apr 1 21:10:14 2024 UTC
```

```
gpg: using RSA key B840C08C29A90796A071FAA5F6CD3CE6B7
```

Wenn die Ausgabe den Ausdruck enthält, bedeutet dies `Good signature from "AWS Deadline Cloud"`, dass die Signatur erfolgreich verifiziert wurde und Sie das Installationsskript für den Deadline Cloud-Monitor ausführen können.

Installieren Sie den Deadline Cloud-Absender

Sie können einen Deadline Cloud-Einreicher mit oder installieren. Windows Linux Mit dem Installer können Sie die folgenden Submitter installieren:

- Mai 2024
- Atombombe 14,0 - 15,0
- Houdini 19,5
- Schlüsselschuss 12
- Mixer 3.6
- Unreal Engine 5

Windows

1. Navigieren Sie in einem Dateibrowser zu dem Ordner, in den das Installationsprogramm heruntergeladen wurde, und wählen Sie `DeadlineCloudSubmitter-windows-x64-installer.exe` dann.
 - a. Wenn ein Popup-Fenster mit Windows-Schutz für Ihren PC angezeigt wird, wählen Sie `Weitere Informationen` aus.
 - b. Wählen Sie `„Trotzdem ausführen“`.
2. Nachdem der AWS Deadline Cloud Submitter Setup Wizard geöffnet wurde, wählen Sie `Weiter`.
3. Wählen Sie den Umfang der Installation aus, indem Sie einen der folgenden Schritte ausführen:

- Um nur für den aktuellen Benutzer zu installieren, wählen Sie Benutzer.
- Um für alle Benutzer zu installieren, wählen Sie System.

Wenn Sie System wählen, müssen Sie das Installationsprogramm beenden und es als Administrator erneut ausführen, indem Sie die folgenden Schritte ausführen:

- a. Klicken Sie mit der rechten Maustaste auf **DeadlineCloudSubmitter-windows-x64-installer.exe** und wählen Sie dann Als Administrator ausführen.
 - b. Geben Sie Ihre Administratoranmeldedaten ein und wählen Sie dann Ja.
 - c. Wählen Sie System als Installationsbereich aus.
4. Nachdem Sie den Installationsbereich ausgewählt haben, wählen Sie Weiter.
 5. Wählen Sie erneut Weiter, um das Installationsverzeichnis zu akzeptieren.
 6. Wählen Sie Integrated Submitter für oder den SubmitterNuke, den Sie installieren möchten.
 7. Wählen Sie Weiter aus.
 8. Überprüfen Sie die Installation und wählen Sie Weiter.
 9. Wählen Sie erneut Weiter und dann Fertig stellen.

Linux

Note

Das integrierte Deadline Nuke Cloud-Installationsprogramm für Linux und der Deadline Cloud-Monitor können nur auf Linux Distributionen mit mindestens GLIBC 2.31 installiert werden.

1. Öffnen Sie ein Terminal-Fenster.
2. Um eine Systeminstallation des Installers durchzuführen, geben Sie den Befehl ein **sudo -i** und drücken Sie die Eingabetaste, um root zu werden.
3. Navigieren Sie zu dem Verzeichnis, in das Sie das Installationsprogramm heruntergeladen haben.

z. B. **cd /home/*USER*/Downloads.**
4. Geben Sie ein, um das Installationsprogramm ausführbar zu machen **chmod +x DeadlineCloudSubmitter-linux-x64-installer.run.**

5. Geben Sie ein, um das Deadline Cloud Submitter-Installationsprogramm auszuführen. **./DeadlineCloudSubmitter-linux-x64-installer.run**
6. Wenn das Installationsprogramm geöffnet wird, folgen Sie den Anweisungen auf Ihrem Bildschirm, um den Einrichtungsassistenten abzuschließen.

Sie können andere Einreicher installieren, die hier nicht aufgeführt sind. Wir verwenden Deadline Cloud-Bibliotheken, um Einreicher zu erstellen. [Den Quellcode für diese Bibliotheken und Einreicher finden Sie in der AWS-Deadline-Organisation. GitHub](#)

Schritt 2: Deadline Cloud Monitor installieren und einrichten

Sie können die Desktop-Anwendung Deadline Cloud Monitor mit Windows oder installierenLinux.

Windows

1. Falls Sie es noch nicht getan haben, melden Sie sich bei der Deadline [Cloud-Konsole](#) an AWS Management Console und öffnen Sie sie.
2. Wählen Sie im linken Navigationsbereich Downloads aus.
3. Wählen Sie im Bereich Deadline Cloud Monitor die Datei für das Betriebssystem Ihres Computers aus.
4. Um den Deadline Cloud-Monitor herunterzuladen, wählen Sie Herunterladen.

Linux

Um Deadline Cloud Monitor Appliance auf RPM-Distributionen zu installieren

1. Laden Sie den neuesten Deadline Cloud-Monitor Appliance herunter.
2. Geben Sie ein, um die Appliance ausführbare Datei zu erstellen **chmod a+x deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage**.
3. Geben **sudo ln -sf /etc/ssl/certs/ca-bundle.crt /etc/ssl/certs/ca-certificates.crt** Sie ein, um den richtigen SSL-Zertifikatspfad einzurichten.

Um Deadline Cloud Monitor Appliance auf Debian-Distributionen zu installieren

1. Laden Sie den neuesten Deadline Cloud-Monitor Appliance herunter.

2.

Note

Dieser Schritt gilt für Ubuntu 22 und höher. Für andere Versionen von Ubuntu überspringen Sie diesen Schritt.

Um libfuse2 zu installieren, geben Sie ein **sudo apt update**

sudo apt install libfuse2.

3. Geben Sie ein, um die AppImage ausführbare Datei zu erstellen. **chmod a+x deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage**

Um Deadline Cloud zu installieren, überwachen Sie das Debian-Paket auf Debian-Distributionen

1. Laden Sie das neueste Debian-Paket für den Deadline Cloud Monitor herunter.

2.

Note

Dieser Schritt gilt für Ubuntu 22 und höher. Für andere Versionen von Ubuntu überspringen Sie diesen Schritt.

Um libssl1.1 zu installieren, geben Sie ein **wget http://nz2.archive.ubuntu.com/ubuntu/pool/main/o/openssl/libssl1.<APP_VERSION>.1f-1ubuntu2.22_amd64.deb**

sudo dpkg -i libssl1.<APP_VERSION>.1f-1ubuntu2.22_amd64.deb.

3. Um das Debian-Paket Deadline Cloud Monitor zu installieren, geben Sie ein **sudo apt update**

sudo apt install ./deadline-cloud-monitor_<APP_VERSION>_amd64.deb.

4. Falls die Installation bei Paketen fehlschlägt, deren Abhängigkeiten noch nicht erfüllt sind, reparieren Sie die defekten Pakete und führen Sie dann die folgenden Befehle aus.

sudo apt --fix-missing update

sudo apt update

sudo apt install -f

Nachdem Sie den Download abgeschlossen haben, können Sie die Authentizität der heruntergeladenen Software überprüfen. Weitere Informationen finden Sie unter Überprüfen der Authentizität der heruntergeladenen Software in Schritt 1.

Nachdem Sie den Deadline Cloud-Monitor heruntergeladen und die Authentizität überprüft haben, richten Sie den Deadline Cloud-Monitor wie folgt ein.

So richten Sie den Deadline Cloud-Monitor ein

1. Öffnen Sie den Deadline Cloud-Monitor.
2. Wenn Sie aufgefordert werden, ein neues Profil zu erstellen, führen Sie die folgenden Schritte aus.
 - a. Geben Sie Ihre Monitor-URL in die URL-Eingabe ein, die wie folgt aussieht **https://MY-MONITOR.deadlinecloud.amazonaws.com/**
 - b. Geben Sie einen Profilnamen ein.
 - c. Wählen Sie Profil erstellen.

Ihr Profil wurde erstellt und Ihre Anmeldeinformationen werden nun mit jeder Software geteilt, die den von Ihnen erstellten Profilnamen verwendet.

3. Nachdem Sie das Deadline Cloud-Monitorprofil erstellt haben, können Sie den Profilnamen oder die Studio-URL nicht mehr ändern. Wenn Sie Änderungen vornehmen müssen, gehen Sie stattdessen wie folgt vor:
 - a. Lösche das Profil. Wählen Sie im linken Navigationsbereich Deadline Cloud Monitor, Einstellungen, Löschen aus.
 - b. Erstellen Sie ein neues Profil mit den gewünschten Änderungen.
4. Verwenden Sie im linken Navigationsbereich die Option >Deadline Cloud Monitor, um Folgendes zu tun:
 - Ändern Sie das Deadline Cloud-Monitorprofil, um sich bei einem anderen Monitor anzumelden.
 - Aktivieren Sie Autologin, damit Sie Ihre Monitor-URL bei nachfolgenden Öffnungen des Deadline Cloud-Monitors nicht eingeben müssen.
5. Schließen Sie das Deadline Cloud-Monitorfenster. Es läuft weiterhin im Hintergrund und synchronisiert Ihre Anmeldeinformationen alle 15 Minuten.
6. Führen Sie für jede DCC-Anwendung (Digital Content Creation), die Sie für Ihre Renderprojekte verwenden möchten, die folgenden Schritte aus:

- a. Öffnen Sie in Ihrem Deadline Cloud-Absender die Konfiguration der Deadline Cloud-Workstation.
- b. Wählen Sie in der Workstation-Konfiguration das Profil aus, das Sie im Deadline Cloud-Monitor erstellt haben. Ihre Deadline Cloud-Anmeldeinformationen werden jetzt mit diesem DCC geteilt und Ihre Tools sollten wie erwartet funktionieren.

Schritt 3: Starten Sie den Deadline Cloud Submitter

Die folgenden Abschnitte führen Sie durch die Schritte zum Starten des Deadline Cloud-Einreicher-Plug-ins in Blender, und Nuke Maya Houdini

So starten Sie den Deadline Cloud-Einreicher in Blender

Note

Support für Blender wird mithilfe der Conda Umgebung für servicemanagerte Flotten bereitgestellt. Weitere Informationen finden Sie unter [CondaStandard-Warteschlangenumgebung](#).

1. Öffnen Sie Blender.
2. Öffnen Sie eine Blender Szene mit Abhängigkeiten, die im Stammverzeichnis der Ressource vorhanden sind.
3. Wählen Sie im Menü „Rendern“ das Dialogfeld „Deadline Cloud“ aus.
 - a. Wenn Sie im Deadline Cloud-Absender noch nicht authentifiziert sind, wird im Anmeldestatus NEEDS_LOGIN angezeigt.
 - b. Wählen Sie Login (Anmelden) aus.
 - c. Ein Browserfenster für die Anmeldung wird angezeigt. Melden Sie sich mit Ihren Benutzeranmeldedaten an.
 - d. Wählen Sie Zulassen. Sie sind jetzt angemeldet und der Status der Anmeldeinformationen wird als AUTHENTIFIZIERT angezeigt.
4. Wählen Sie Absenden aus.

Um den Deadline Cloud Submitter zu starten in Foundry Nuke

Note

Support für Nuke wird mithilfe der Conda Umgebung für servicemanagierte Flotten bereitgestellt. Weitere Informationen finden Sie unter [CondaStandard-Warteschlangen Umgebung](#).

1. Öffnen Sie Nuke.
2. Öffnen Sie ein Nuke Skript mit Abhängigkeiten, die im Stammverzeichnis der Ressource vorhanden sind.
3. Wählen Sie Thinkbox und dann Submit to Deadline Cloud, um den Submitter zu starten.
 - a. Wenn Sie noch nicht im Deadline Cloud-Absender authentifiziert sind, wird der Anmeldeinformationsstatus als NEEDS_LOGIN angezeigt.
 - b. Wählen Sie Login (Anmelden) aus.
 - c. Melden Sie sich im Anmeldefenster des Browsers mit Ihren Benutzeranmeldedaten an.
 - d. Wählen Sie Zulassen. Sie sind jetzt angemeldet und der Status der Anmeldeinformationen wird als AUTHENTIFIZIERT angezeigt.
4. Wählen Sie Absenden aus.

Um den Deadline Cloud Submitter zu starten in Maya

Note

Support für Maya und Arnold for Maya(MtoA) wird mithilfe der Conda Umgebung für servicemanagierte Flotten bereitgestellt. Weitere Informationen finden Sie unter [CondaStandard-Warteschlangen Umgebung](#).

1. Öffnen Sie Maya.
2. Legen Sie Ihr Projekt fest und öffnen Sie eine Datei, die sich im Stammverzeichnis der Ressource befindet.
3. Wählen Sie Windows → Einstellungen/Einstellungen → Plugin-Manager.
4. Suchen Sie nach Submitter. DeadlineCloud

5. Um das Deadline Cloud-Absender-Plugin zu laden, wählen Sie Geladen aus.
 - a. Wenn Sie noch nicht im Deadline Cloud-Absender authentifiziert sind, wird der Anmeldeinformationsstatus als NEEDS_LOGIN angezeigt.
 - b. Wählen Sie Login (Anmelden) aus.
 - c. Ein Anmeldefenster im Browser wird angezeigt. Melden Sie sich mit Ihren Benutzeranmeldedaten an.
 - d. Wählen Sie Zulassen. Sie sind jetzt angemeldet und der Status der Anmeldeinformationen wird als AUTHENTIFIZIERT angezeigt.
6. (Optional) Um das Deadline Cloud-Absender-Plugin bei jedem Öffnen zu laden, wählen Sie Automatisch laden.
7. Wählen Sie das Deadline Cloud-Regal aus und klicken Sie dann auf die grüne Schaltfläche, um den Submitter zu starten.

So starten Sie den Deadline Cloud-Absender in Houdini

 Note

Support für Houdini wird mithilfe der Conda Umgebung für servicemanagierte Flotten bereitgestellt. Weitere Informationen finden Sie unter [CondaStandard-Warteschlangenumgebung](#).

1. Öffnen Sie Houdini.
2. Wählen Sie im Netzwerk-Editor das /out-Netzwerk aus.
3. Drücken Sie die Tabulatortaste und dann die Eingabetaste **deadline**.
4. Wählen Sie die Option Deadline Cloud und verbinden Sie sie mit Ihrem vorhandenen Netzwerk.
5. Doppelklicken Sie auf den Deadline Cloud-Knoten.

Um den Deadline Cloud-Absender in zu starten KeyShot

Dies setzt voraus, dass Sie Deadline Cloud und PySide 2 bereits heruntergeladen haben.

1. Kopieren oder verknüpfen Sie die Datei Deadline-Cloud-for-Keyshot/Keyshot_Script/Submit to AWS Deadline Cloud.py in den Ordner scripts. KeyShot

Zum Windows Beispiel wäre on der Speicherort des Skript-Ordners. **C:/Users/*USER*/Documents/KeyShot 12/Scripts**

2. Legen Sie die folgenden Umgebungsvariablen fest.
 - a. Legen Sie die Umgebungsvariable **DEADLINE_PYTHON** als Pfad zur Python-Installation fest, in der sich deadline-cloud und PySide 2 befinden.

Wenn Sie Windows beispielsweise Python 3.10 verwenden, könnte der Befehl lautenset **DEADLINE_PYTHON=C:/Users/*USER*/AppData/Local/Programs/Python/Python310/python.**

- b. Legen Sie die Umgebungsvariable **DEADLINE_KEYSHOT** als Pfad zum Ordner keyshot_submitter fest.

Wenn sich die Quelle beispielsweise auf Ihrem Desktop befindetWindows, könnte der Befehl aktiviert sein. **set DEADLINE_KEYSHOT=C:/Users/*USER*/Desktop/deadline-cloud-for-keyshot/src/deadline/keyshot_submitter**

3. Starten Sie, wenn die Umgebungsvariablen gesetzt sind KeyShot.
4. Um den Submitter von zu starten KeyShot, wählen Sie Scripting console Windows, Submit to AWS Deadline Cloud und Run.

So starten Sie den Deadline Cloud-Absender in Unreal Engine

Dies setzt voraus, dass Sie Deadline Cloud bereits heruntergeladen haben.

1. Erstellen oder öffnen Sie den Ordner, den Sie für Ihre Unreal Engine Projekte verwenden.
2. Öffnen Sie die Befehlszeile und führen Sie die folgenden Befehle aus:
 - `git clone https://github.com/aws-deadline/deadline-cloud-for-unreal-engine`
 - `cd deadline-cloud-for-unreal/test_projects`
 - `git lfs fetch -all`
3. Um das Plugin für herunterzuladenUnreal Engine, öffnen Sie den Unreal Engine Projektordner und starten Sie `deadline-cloud-forunreal/test_projects/pull_ue_plugin.bat`.

Dadurch werden die Plugin-Dateien in `C:/LocalProjectsUnrealDeadlineCloudTest/Plugins/` abgelegt UnrealDeadline CloudService.

4. Um den Absender herunterzuladen, öffnen Sie den Ordner und führen Sie ihn aus UnrealDeadlineCloudService . **deadline-cloud-forunreal/ test_projects/Plugins/ UnrealDeadlineCloudService/install_unreal_submitter.bat**
5. Gehen Sie wie folgt vor, um den Absender von aus Unreal Engine zu starten:
 - a. Wählen Sie Bearbeiten > Projekteinstellungen.
 - b. Geben Sie im Suchfeld **movie render pipeline** ein.
 - c. Passen Sie die folgenden Einstellungen für die Movie Render-Pipeline an:
 - i. Geben Sie für Default Remote Executor ein. **MoviePipelineDeadlineCloudRemote Executor**
 - ii. Geben Sie für Default Executor Job Folgendes ein **MoviePipelineDeadlineCloudExecutorJob**
 - iii. Wählen Sie für Standardklassen für Jobeinstellungen das Pluszeichen aus, und geben Sie dann die Eingabetaste ein **DeadlineCloudRenderStepSetting**.

Mit diesen Einstellungen können Sie das Deadline Cloud-Plugin unter auswählen Unreal Engine.

Benutze die Farm

Wenn Sie alle Anweisungen für die ersten Schritte befolgt haben, haben Sie alles eingerichtet, was Sie benötigen, um Jobs von Ihrer lokalen Arbeitsstation an Ihre Farm zu senden und diese Jobs und Ressourcen anschließend zu überwachen. Weitere Informationen zum Senden aller Arten von Aufträgen oder zur Überwachung finden Sie in den entsprechenden Themen weiter unten.

- [Aufträge](#)
- [Verwenden des Monitors](#)

Den Deadline Cloud-Monitor verwenden

Der AWS Deadline Cloud-Monitor bietet Ihnen einen Gesamtüberblick über Ihre visuellen Rechenjobs. Sie können ihn verwenden, um Jobs zu überwachen und zu verwalten, die Mitarbeiteraktivitäten in Flotten einzusehen, Budgets und Nutzung zu verfolgen und die Ergebnisse eines Jobs herunterzuladen.

Jede Warteschlange verfügt über einen Jobmonitor, der Ihnen den Status von Aufträgen, Schritten und Aufgaben anzeigt. Der Monitor bietet Möglichkeiten, Jobs direkt vom Monitor aus zu verwalten. Sie können Änderungen an der Priorisierung vornehmen, Jobs stornieren und Jobs in eine Warteschlange stellen.

Der Deadline Cloud-Monitor enthält eine Tabelle, in der der Übersichtsstatus für einen Job angezeigt wird. Sie können auch einen Job auswählen, um detaillierte Aufgabenprotokolle anzuzeigen, die bei der Behebung von Problemen mit einem Job helfen.

Sie können den Deadline Cloud-Monitor verwenden, um die Ergebnisse an den Speicherort auf Ihrer Workstation herunterzuladen, der bei der Erstellung des Jobs angegeben wurde.

Der Deadline Cloud-Monitor hilft Ihnen auch dabei, die Nutzung zu überwachen und die Kosten zu verwalten. Weitere Informationen finden Sie unter [Verwaltung von Budgets und Nutzung für Deadline Cloud](#).

Themen

- [Teilen Sie die URL des Deadline Cloud-Monitors](#)
- [Öffnen Sie den Deadline Cloud-Monitor](#)
- [Warteschlangen- und Flottendetails in Deadline Cloud anzeigen](#)
- [Jobs, Schritte und Aufgaben in Deadline Cloud anzeigen und verwalten](#)
- [Jobdetails in Deadline Cloud anzeigen](#)
- [Einen Schritt in Deadline Cloud anzeigen](#)
- [Eine Aufgabe in Deadline Cloud anzeigen](#)
- [Logs in Deadline Cloud anzeigen](#)
- [Laden Sie die fertige Ausgabe in Deadline Cloud herunter](#)

Teilen Sie die URL des Deadline Cloud-Monitors

Wenn Sie den Deadline Cloud-Dienst einrichten, erstellen Sie standardmäßig eine URL, die den Deadline Cloud-Monitor für Ihr Konto öffnet. Verwenden Sie diese URL, um den Monitor in Ihrem Browser oder auf Ihrem Desktop zu öffnen. Teilen Sie die URL mit anderen Benutzern, damit diese auf den Deadline Cloud-Monitor zugreifen können.

Bevor ein Benutzer den Deadline Cloud-Monitor öffnen kann, müssen Sie dem Benutzer Zugriff gewähren. Um Zugriff zu gewähren, fügen Sie den Benutzer entweder der Liste der autorisierten Benutzer für den Monitor hinzu oder fügen Sie ihn einer Gruppe mit Zugriff auf den Monitor hinzu. Weitere Informationen finden Sie unter [Benutzer in Deadline Cloud verwalten](#).

Um die Monitor-URL zu teilen

1. Öffnen Sie die [Deadline Cloud-Konsole](#).
2. Wählen Sie unter Erste Schritte die Option Gehe zum Deadline Cloud-Dashboard.
3. Wählen Sie im Navigationsbereich Dashboard aus.
4. Wählen Sie im Abschnitt Kontoübersicht die Option Kontodetails aus.
5. Kopieren Sie die URL und senden Sie sie dann sicher an alle Personen, die auf den Deadline Cloud-Monitor zugreifen müssen.

Öffnen Sie den Deadline Cloud-Monitor

Sie können den Deadline Cloud-Monitor auf eine der folgenden Arten öffnen:

- Konsole — Melden Sie sich bei der Deadline Cloud-Konsole an AWS Management Console und öffnen Sie sie.
- Web — Rufen Sie die Monitor-URL auf, die Sie bei der Einrichtung von Deadline Cloud erstellt haben.
- Monitor — Verwenden Sie den Desktop-Monitor von Deadline Cloud.

Wenn Sie die Konsole verwenden, müssen Sie in der Lage sein, sich AWS mit einer AWS Identity and Access Management Identität anzumelden und sich dann mit AWS IAM Identity Center Anmeldeinformationen am Monitor anzumelden. Wenn Sie nur über IAM Identity Center-Anmeldeinformationen verfügen, müssen Sie sich mit der Monitor-URL oder der Desktop-Anwendung anmelden.

Um den Deadline Cloud-Monitor (Web) zu öffnen

1. Öffnen Sie mit einem Browser die Monitor-URL, die Sie bei der Einrichtung von Deadline Cloud erstellt haben.
2. Melden Sie sich mit Ihren Benutzeranmeldedaten an.

Um den Deadline Cloud-Monitor (Konsole) zu öffnen

1. Öffnen Sie die [Deadline Cloud-Konsole](#).
2. Wählen Sie im Navigationsbereich Farmen aus.
3. Wählen Sie eine Farm aus und wählen Sie dann Jobs verwalten, um die Deadline Cloud-Monitorseite zu öffnen.
4. Melden Sie sich mit Ihren Benutzeranmeldedaten an.

Um den Deadline Cloud-Monitor (Desktop) zu öffnen

1. Öffnen Sie die [Deadline Cloud-Konsole](#).

–oder–

Öffnen Sie den Deadline Cloud-Monitor — Web über die Monitor-URL.

2. • Gehen Sie in der Deadline Cloud-Konsole wie folgt vor:
 1. Wählen Sie im Monitor Gehe zum Deadline Cloud-Dashboard und dann im linken Menü die Option Downloads aus.
 2. Wählen Sie im Deadline Cloud-Monitor die Monitorversion für Ihren Desktop aus.
 3. Wählen Sie Herunterladen aus.
- Gehen Sie auf dem Deadline Cloud-Monitor — Web wie folgt vor:
 - Wählen Sie im linken Menü die Option Workstation-Setup. Wenn das Workstation-Setup-Element nicht sichtbar ist, verwenden Sie den Pfeil, um das linke Menü zu öffnen.
 - Wählen Sie Herunterladen aus.
 - Wählen Sie unter Betriebssystem auswählen Ihr Betriebssystem aus.
3. Laden Sie den Deadline Cloud-Monitor für den Desktop herunter.
4. Nachdem Sie den Monitor heruntergeladen und installiert haben, öffnen Sie ihn auf Ihrem Computer.

- Wenn Sie den Deadline Cloud-Monitor zum ersten Mal öffnen, müssen Sie die Monitor-URL angeben und einen Profilnamen erstellen. Als Nächstes melden Sie sich mit Ihren Deadline Cloud-Anmeldeinformationen beim Monitor an.
- Nachdem Sie ein Profil erstellt haben, öffnen Sie den Monitor, indem Sie ein Profil auswählen. Möglicherweise müssen Sie Ihre Deadline Cloud-Anmeldeinformationen eingeben.

Warteschlangen- und Flottendetails in Deadline Cloud anzeigen

Sie können den Deadline Cloud-Monitor verwenden, um die Konfiguration der Warteschlangen und Flotten in Ihrer Farm einzusehen. Sie können den Monitor auch verwenden, um eine Liste der Jobs in einer Warteschlange oder der Arbeiter in einer Flotte anzuzeigen.

Sie müssen VIEWING berechtigt sein, Warteschlangen- und Flottendetails einzusehen. Wenn die Details nicht angezeigt werden, wenden Sie sich an Ihren Administrator, um die richtigen Berechtigungen zu erhalten.

Um die Details der Warteschlange einzusehen

1. [Öffnen Sie den Deadline Cloud-Monitor.](#)
2. Wählen Sie aus der Liste der Farmen die Farm aus, die die Warteschlange enthält, an der Sie interessiert sind.
3. Wählen Sie in der Liste der Warteschlangen eine Warteschlange aus, um deren Details anzuzeigen. Um die Konfiguration von zwei oder mehr Warteschlangen zu vergleichen, aktivieren Sie mehr als ein Kontrollkästchen.
4. Um eine Liste der Jobs in der Warteschlange anzuzeigen, wählen Sie den Namen der Warteschlange aus der Liste der Warteschlangen oder aus dem Detailbereich aus.

Wenn der Monitor bereits geöffnet ist, können Sie die Warteschlange aus der Warteschlangenliste im linken Navigationsbereich auswählen.

So zeigen Sie Flottendetails an:

1. [Öffnen Sie den Deadline Cloud-Monitor.](#)
2. Wählen Sie aus der Liste der Farmen die Farm aus, die die Flotte enthält, an der Sie interessiert sind.
3. Wählen Sie unter Farmressourcen die Option Flotten aus.

4. Wählen Sie in der Liste der Flotten eine Flotte aus, um deren Details anzuzeigen. Um die Konfiguration von zwei oder mehr Flotten zu vergleichen, aktivieren Sie mehr als ein Kontrollkästchen.
5. Um eine Liste der Mitarbeiter in der Flotte zu sehen, wählen Sie den Flottennamen aus der Flottenliste oder aus dem Detailbereich aus.

Wenn der Monitor bereits geöffnet ist, können Sie die Flotte aus der Flottenliste im linken Navigationsbereich auswählen.

Jobs, Schritte und Aufgaben in Deadline Cloud anzeigen und verwalten

Wenn Sie eine Warteschlange auswählen, werden Ihnen im Bereich Job Monitor des Deadline Cloud-Monitors die Jobs in dieser Warteschlange, die Schritte im Job und die Aufgaben in jedem Schritt angezeigt. Wenn Sie einen Job, einen Schritt oder eine Aufgabe auswählen, können Sie die einzelnen Jobs, Schritte oder Aufgaben über das Aktionsmenü verwalten.

Um den Auftragsmonitor zu öffnen, folgen Sie den Schritten zum Anzeigen einer Warteschlange [Warteschlangen- und Flottendetails in Deadline Cloud anzeigen](#), und wählen Sie dann den Job, Schritt oder die Aufgabe aus, mit dem Sie arbeiten möchten.

Für Jobs, Schritte und Aufgaben können Sie wie folgt vorgehen:

- Ändern Sie den Status in „In Warteschlange“, „Erfolgreich“, „Fehlgeschlagen“ oder „Storniert“.
- Laden Sie die verarbeitete Ausgabe des Jobs, Schritts oder der Aufgabe herunter.
- Kopieren Sie die ID des Jobs, Schritts oder der Aufgabe.

Für den ausgewählten Job können Sie:

- Archivieren Sie den Job.
- Ändern Sie die Auftragseigenschaften, z. B. indem Sie die Priorisierung ändern oder die Abhängigkeiten von Schritt zu Schritt anzeigen.
- Mithilfe der Jobparameter können Sie zusätzliche Details anzeigen.

Weitere Informationen finden Sie unter [Jobdetails in Deadline Cloud anzeigen](#).

Für jeden Schritt können Sie:

- Die Abhängigkeiten für den Schritt anzeigen. Die Abhängigkeiten für einen Schritt müssen abgeschlossen sein, bevor der Schritt ausgeführt wird.

Details hierzu finden Sie unter [Einen Schritt in Deadline Cloud anzeigen](#).

Für jede Aufgabe können Sie:

- Protokolle für die Aufgabe anzeigen.
- Aufgabenparameter anzeigen.

Weitere Informationen finden Sie unter [Eine Aufgabe in Deadline Cloud anzeigen](#).

Jobdetails in Deadline Cloud anzeigen

Die Seite Job Monitor im Deadline Cloud Monitor bietet Ihnen Folgendes:

- Ein Gesamtüberblick über den Fortschritt eines Jobs.
- Ein Überblick über die Schritte und Aufgaben, aus denen sich der Job zusammensetzt.

Wählen Sie einen Job aus der Liste aus, um eine Liste der Schritte für den Job anzuzeigen, und wählen Sie dann einen Schritt aus der Liste der Schritte aus, um die Aufgaben für den Job anzuzeigen. Nachdem Sie ein Element ausgewählt haben, können Sie das Aktionsmenü für dieses Element verwenden, um Details anzuzeigen.

Um Jobdetails anzuzeigen

1. Folgen Sie den Schritten, um eine Warteschlange in anzuzeigen [Warteschlangen- und Flottendetails in Deadline Cloud anzeigen](#).
2. Wählen Sie im Navigationsbereich die Warteschlange aus, in der Sie Ihren Job eingereicht haben.
3. Wählen Sie einen Job mit einer der folgenden Methoden aus:
 - a. Wählen Sie aus der Jobliste einen Job aus, um dessen Details anzuzeigen.

- b. Geben Sie im Suchfeld den Text ein, der mit dem Job verknüpft ist, z. B. den Jobnamen oder den Benutzer, der den Job erstellt hat. Wählen Sie aus den angezeigten Ergebnissen den Job aus, den Sie anzeigen möchten.

Zu den Details eines Jobs gehören die Schritte im Job und die Aufgaben in jedem Schritt. Sie können das Menü Aktionen verwenden, um Folgendes zu tun:

- Ändern Sie den Status des Jobs.
- Die Eigenschaften eines Jobs anzeigen und ändern. Sie können die Abhängigkeiten zwischen den Schritten im Job anzeigen und die Priorität des Jobs ändern. In der Regel werden Jobs mit einer höheren Priorität früher abgeschlossen.
- Sehen Sie sich die Parameter für den Job an, die beim Absenden des Jobs festgelegt wurden.
- Laden Sie die Ausgabe eines Jobs herunter. Wenn Sie die Ausgabe eines Jobs herunterladen, enthält sie die gesamte Ausgabe, die durch die Schritte und Aufgaben im Job generiert wurde.

Einen Schritt in Deadline Cloud anzeigen

Verwenden Sie den AWS Deadline Cloud-Monitor, um sich die Schritte in Ihren Verarbeitungsjobs anzusehen. Im Job-Monitor zeigt die Liste der Schritte die Liste der Schritte, aus denen sich der ausgewählte Job zusammensetzt. Wenn Sie einen Schritt auswählen, werden in der Aufgabenliste die Aufgaben des Schritts angezeigt.

Um einen Schritt anzuzeigen

1. Folgen Sie den Schritten unter [Jobdetails in Deadline Cloud anzeigen](#), um eine Liste der Jobs anzuzeigen.
2. Wählen Sie einen Auftrag aus der Liste Jobs (Aufträge).
3. Wählen Sie einen Schritt aus der Schrittliste aus.

Sie können das Aktionsmenü verwenden, um Folgendes zu tun:

- Ändern Sie den Status des Schritts.
- Laden Sie die Ausgabe des Schritts herunter. Wenn Sie die Ausgabe eines Schritts herunterladen, enthält sie die gesamte Ausgabe, die von den Aufgaben in diesem Schritt generiert wurde.

- Sehen Sie sich die Abhängigkeiten eines Schritts an. Die Tabelle mit den Abhängigkeiten enthält eine Liste der Schritte, die abgeschlossen sein müssen, bevor der ausgewählte Schritt gestartet wird, sowie eine Liste der Schritte, die noch auf den Abschluss dieses Schritts warten.

Eine Aufgabe in Deadline Cloud anzeigen

Verwenden Sie den AWS Deadline Cloud-Monitor, um sich die Aufgaben in Ihren Verarbeitungsjobs anzusehen. Im Job-Monitor werden in der Aufgabenliste die Aufgaben angezeigt, aus denen der in der Schrittliste ausgewählte Schritt besteht.

Um eine Aufgabe anzusehen

1. Folgen Sie den Schritten unter [Jobdetails in Deadline Cloud anzeigen](#), um eine Liste von Jobs anzuzeigen.
2. Wählen Sie einen Auftrag aus der Liste Jobs (Aufträge).
3. Wählen Sie einen Schritt aus der Schrittliste aus.
4. Wählen Sie eine Aufgabe aus der Aufgabenliste aus.

Sie können das Aktionsmenü verwenden, um Folgendes zu tun:

- Ändern Sie den Status der Aufgabe.
- Aufgabenprotokolle anzeigen. Weitere Informationen finden Sie unter [Logs in Deadline Cloud anzeigen](#).
- Sehen Sie sich die Parameter an, die bei der Erstellung der Aufgabe festgelegt wurden.
- Laden Sie die Ausgabe der Aufgabe herunter. Wenn Sie die Ausgabe einer Aufgabe herunterladen, enthält sie nur die Ausgabe, die von der ausgewählten Aufgabe generiert wurde.

Logs in Deadline Cloud anzeigen

Logs liefern Ihnen detaillierte Informationen über den Status und die Bearbeitung von Aufgaben. Im AWS Deadline Cloud-Monitor können Sie die folgenden zwei Arten von Protokollen sehen:

- Sitzungsprotokolle beschreiben detailliert den Zeitplan der Aktionen, darunter:
 - Einrichtungsaktionen, z. B. das Synchronisieren von Anhängen und das Laden der Softwareumgebung

- Ausführen einer Aufgabe oder einer Reihe von Aufgaben
- Aktionen zum Schließen, z. B. das Herunterfahren der Umgebung eines Mitarbeiters

Eine Sitzung umfasst die Bearbeitung von mindestens einer Aufgabe und kann mehrere Aufgaben umfassen. Sitzungsprotokolle enthalten auch Informationen über den Instanztyp, die vCPU und den Arbeitsspeicher von Amazon Elastic Compute Cloud (Amazon EC2). Sitzungsprotokolle enthalten auch einen Link zum Protokoll für den in der Sitzung verwendeten Worker.

- Arbeitsprotokolle enthalten Details zum Zeitplan der Aktionen, die ein Mitarbeiter während seines Lebenszyklus ausführt. Arbeitsprotokolle können Informationen über mehrere Sitzungen enthalten.

Sie können Sitzungs- und Worker-Protokolle herunterladen, um sie offline zu überprüfen.

Um Sitzungsprotokolle einzusehen

1. Folgen Sie den Schritten unter [Jobdetails in Deadline Cloud anzeigen](#), um eine Liste der Jobs anzuzeigen.
2. Wählen Sie einen Auftrag aus der Liste Jobs (Aufträge).
3. Wählen Sie einen Schritt aus der Schrittliste aus.
4. Wählen Sie eine Aufgabe aus der Aufgabenliste aus.
5. Wählen Sie im Menü Aktionen die Option Protokolle anzeigen aus.

Im Abschnitt Zeitpläne wird eine Zusammenfassung der Aktionen für die Aufgabe angezeigt. Wenn Sie mehr in der Sitzung ausgeführte Aufgaben und die Aktionen zum Herunterfahren der Sitzung anzeigen möchten, wählen Sie Protokolle für alle Aufgaben anzeigen.

Um die Worker-Logs einer Aufgabe einzusehen

1. Folgen Sie den Anweisungen unter [Jobdetails in Deadline Cloud anzeigen](#), um eine Liste der Jobs anzuzeigen.
2. Wählen Sie einen Auftrag aus der Liste Jobs (Aufträge).
3. Wählen Sie einen Schritt aus der Schrittliste aus.
4. Wählen Sie eine Aufgabe aus der Aufgabenliste aus.
5. Wählen Sie im Menü Aktionen die Option Protokolle anzeigen aus.
6. Wählen Sie Sitzungsinformationen aus.
7. Wählen Sie „Mitarbeiterprotokoll anzeigen“.

Um Mitarbeiterprotokolle anhand der Flottendetails anzuzeigen

1. Folgen Sie den Anweisungen unter [Warteschlangen- und Flottendetails in Deadline Cloud anzeigen](#), um eine Flotte anzuzeigen.
2. Wählen Sie eine Mitarbeiter-ID aus der Mitarbeiterliste aus.
3. Wählen Sie im Menü „Aktionen“ die Option „Mitarbeiterprotokolle anzeigen“.

Laden Sie die fertige Ausgabe in Deadline Cloud herunter

Nachdem ein Job abgeschlossen ist, können Sie den AWS Deadline Cloud-Monitor verwenden, um die Ergebnisse auf Ihre Workstation herunterzuladen. Die Ausgabedatei wird mit dem Namen und dem Speicherort gespeichert, den Sie bei der Erstellung des Jobs angegeben haben.

Ausgabedateien werden auf unbestimmte Zeit gespeichert. Um die Speicherkosten zu senken, sollten Sie erwägen, eine S3-Lifecycle-Konfiguration für den Amazon S3 S3-Bucket Ihrer Warteschlange zu erstellen. Weitere Informationen finden Sie unter [Verwaltung Ihres Speicherlebenszyklus](#) im Amazon Simple Storage Service-Benutzerhandbuch.

Um die fertige Ausgabe eines Jobs, Schritts oder einer Aufgabe herunterzuladen

1. Folgen Sie den Schritten unter [Jobdetails in Deadline Cloud anzeigen](#), um eine Liste der Jobs anzuzeigen.
2. Wählen Sie den Job, Schritt oder die Aufgabe aus, für den Sie die Ausgabe herunterladen möchten.
 - Wenn Sie einen Job auswählen, können Sie die gesamte Ausgabe für alle Aufgaben in allen Schritten für diesen Job herunterladen.
 - Wenn Sie einen Schritt auswählen, können Sie die gesamte Ausgabe für alle Aufgaben in diesem Schritt herunterladen.
 - Wenn Sie eine Aufgabe auswählen, können Sie die Ausgabe für diese einzelne Aufgabe herunterladen.
3. Wählen Sie im Menü Aktionen die Option Ausgabe herunterladen.
4. Die Ausgabe wird an den Speicherort heruntergeladen, der beim Absenden des Jobs festgelegt wurde.

 Note

Das Herunterladen der Ausgabe über das Menü wird derzeit nur für Windows und unterstützt Linux. Wenn Sie eine haben Mac und den Menüpunkt Ausgabe herunterladen wählen, wird in einem Fenster der AWS CLI Befehl angezeigt, mit dem Sie die gerenderte Ausgabe herunterladen können.

Deadline Cloud-Farmen

Eine Farm ist ein Container für Warteschlangen, in denen Jobs und Flotten von Rechenressourcen verwaltet werden, die Aufgaben ausführen.

Themen

- [Erstellen Sie eine Farm](#)
- [Löschen Sie eine Farm](#)
- [Bearbeiten Sie eine Farm](#)

Erstellen Sie eine Farm

1. Wählen Sie in der [Deadline Cloud-Konsole](#) die Option Gehe zum Dashboard aus.
2. Wählen Sie im Bereich Farmen des Deadline Cloud-Dashboards Aktionen → Farm erstellen aus.
 - Alternativ können Sie in der linken Seitenleiste Farmen und andere Ressourcen und dann Create Farm auswählen.
3. Füge einen Namen für deine Farm hinzu.
4. Geben Sie unter Beschreibung die Farmbeschreibung ein. Eine klare Beschreibung kann Ihnen helfen, den Zweck Ihrer Farm schnell zu ermitteln.
5. (Optional) Standardmäßig werden Ihre Daten mit einem Schlüssel verschlüsselt, der zu Ihrer eigenen Sicherheit AWS gehört und verwaltet wird. Sie können Verschlüsselungseinstellungen anpassen (erweitert) wählen, um einen vorhandenen Schlüssel zu verwenden oder einen neuen, von Ihnen verwalteten Schlüssel zu erstellen.

Wenn Sie die Verschlüsselungseinstellungen über das Kontrollkästchen anpassen möchten, geben Sie einen AWS KMS ARN ein oder erstellen Sie einen neuen, AWS KMS indem Sie Neuen KMS-Schlüssel erstellen wählen.

6. (Optional) Wählen Sie Neues Tag hinzufügen aus, um Ihrer Farm ein oder mehrere Tags hinzuzufügen.
7. Wählen Sie Farm erstellen aus. Nach der Erstellung wird Ihre Farm angezeigt.

Löschen Sie eine Farm

1. Wählen Sie im Deadline Cloud-Dashboard die Option Farmen und andere Ressourcen aus.
2. Wählen Sie in der Farmenliste die Farm oder Farmen aus, die Sie löschen möchten, und wählen Sie dann Löschen aus.

Bearbeiten Sie eine Farm

1. Wählen Sie im Deadline Cloud-Dashboard die Option Farmen und andere Ressourcen aus.
2. Wählen Sie in der Farmenliste die Farm oder Farmen aus, die Sie löschen möchten, und klicken Sie dann auf Bearbeiten.
3. Ändern Sie im daraufhin angezeigten Bearbeitungsfenster den Namen oder die Beschreibung der Farm und wählen Sie dann Änderungen speichern aus.

Deadline Cloud-Warteschlangen

Eine Warteschlange ist eine Farmressource, die Jobs verwaltet und verarbeitet.

Um mit Warteschlangen arbeiten zu können, sollten Sie bereits einen Monitor und eine Farm eingerichtet haben.

Themen

- [Erstellen einer Warteschlange](#)
- [Erstellen Sie eine Warteschlangenumgebung](#)
- [Löschen einer Warteschlange](#)
- [Bearbeiten einer Warteschlange](#)
- [Ordnen Sie eine Warteschlange und eine Flotte zu](#)

Erstellen einer Warteschlange

1. Wählen Sie im Dashboard der [Deadline Cloud-Konsole](#) die Farm aus, für die Sie eine Warteschlange erstellen möchten.
 - Sie können auch auf der linken Seite Farmen und andere Ressourcen auswählen und dann die Farm auswählen, für die Sie eine Warteschlange erstellen möchten.
2. Wählen Sie auf der Registerkarte Warteschlangen die Option Warteschlange erstellen aus.
3. Geben Sie einen Namen für Ihre Warteschlange ein.
4. Geben Sie unter Beschreibung die Beschreibung der Warteschlange ein. Eine Beschreibung hilft Ihnen dabei, den Zweck Ihrer Warteschlange zu identifizieren.
5. Für Job-Anhänge können Sie entweder einen neuen Amazon S3 S3-Bucket erstellen oder einen vorhandenen Amazon S3 S3-Bucket auswählen.
 - a. Um einen neuen Amazon S3 S3-Bucket zu erstellen
 - i. Wählen Sie Neuen Job-Bucket erstellen aus.
 - ii. Geben Sie einen Namen für den Bucket ein. Wir empfehlen, dem Bucket einen Namen zu gebendeadlinecloud-job-attachments-[MONITORNAME].
 - iii. Geben Sie ein Root-Präfix ein, um den Stammspeicherort Ihrer Warteschlange zu definieren oder zu ändern.

- b. Um einen vorhandenen Amazon S3 S3-Bucket auszuwählen
 - i. Wählen Sie „Bestehenden S3-Bucket auswählen“ > „S3 durchsuchen“.
 - ii. Wählen Sie den S3-Bucket für Ihre Warteschlange aus der Liste der verfügbaren Buckets aus.
6. (Optional) Um Ihre Warteschlange einer vom Kunden verwalteten Flotte zuzuordnen, wählen Sie Zuordnung zu kundenverwalteten Flotten aktivieren aus.
7. Wenn Sie die Zuordnung zu kundenverwalteten Flotten aktivieren, müssen Sie die folgenden Schritte ausführen.

 **Important**

Es wird dringend empfohlen, Benutzer und Gruppen für die Run-as-Funktionalität anzugeben. Wenn Sie dies nicht tun, wird die Sicherheitslage Ihrer Farm beeinträchtigt, da die Jobs dann alles tun können, was der Agent des Arbeiters tun kann. Weitere Informationen zu den potenziellen Sicherheitsrisiken finden Sie unter [Jobs als Benutzer und Gruppen ausführen](#).

- a. Für Als Benutzer ausführen:

Um Anmeldeinformationen für die Jobs der Warteschlange anzugeben, wählen Sie In Warteschlange konfigurierter Benutzer aus.

Oder wählen Sie Worker Agent-Benutzer aus, wenn Sie nicht möchten, dass Sie Ihre eigenen Anmeldeinformationen festlegen und Jobs als Worker Agent-Benutzer ausführen.

- b. (Optional) Geben Sie für Als Benutzeranmeldedaten ausführen einen Benutzernamen und einen Gruppennamen ein, um die Anmeldeinformationen für die Jobs der Warteschlange bereitzustellen.

Wenn Sie eine Windows Flotte verwenden, müssen Sie ein AWS Secrets Manager Geheimnis erstellen, das das Passwort für „Als Benutzer ausführen“ enthält. Folgen Sie diesen Anweisungen, um das Geheimnis zu erstellen. Ersetzen Sie *jobuser* durch den Namen des `jobRunAsUser`.

- i. Öffnen Sie PowerShell oder eine Eingabeaufforderung als Administrator.
- ii. Erstellen Sie den Benutzer.

```
net user jobuser /add
```

- iii. Stellen Sie das Passwort ein.

```
net user jobuser *
```

- iv. Erstellen Sie ein lokales Profil und ein Home-Verzeichnis für den Benutzer. Führen Sie den folgenden Befehl aus und geben Sie das Passwort für den Benutzer ein, wenn Sie dazu aufgefordert werden.

```
runas /profile /user:jobuser "cmd.exe /C"
```

8. Wenn Sie ein Budget angeben, können Sie die Kosten für Ihre Warteschlange besser verwalten. Wählen Sie entweder Kein Budget erforderlich oder Budget erforderlich aus.
9. Ihre Warteschlange benötigt die Erlaubnis, in Ihrem Namen auf Amazon S3 zuzugreifen. Sie können eine neue Servicerolle erstellen oder eine bestehende Servicerolle verwenden. Wenn Sie noch keine Servicerolle haben, erstellen und verwenden Sie eine neue Servicerolle.
 - a. Um eine bestehende Servicerolle zu verwenden, wählen Sie eine Servicerolle auswählen und wählen Sie dann eine Rolle aus der Dropdownliste aus.
 - b. Um eine neue Servicerolle zu erstellen, wählen Sie Neue Servicerolle erstellen und verwenden aus und geben Sie dann einen Rollennamen und eine Beschreibung ein.
10. (Optional) Um Umgebungsvariablen für die Warteschlangenumgebung hinzuzufügen, wählen Sie Neue Umgebungsvariable hinzufügen und geben Sie dann einen Namen und einen Wert für jede hinzugefügte Variable ein.
11. (Optional) Wählen Sie Neues Tag hinzufügen, um Ihrer Warteschlange ein oder mehrere Tags hinzuzufügen.
12. Um eine standardmäßige Conda Warteschlangenumgebung zu erstellen, lassen Sie das Kontrollkästchen aktiviert. Weitere Informationen zu Warteschlangenumgebungen finden Sie unter [Eine Warteschlangenumgebung erstellen](#). Wenn Sie eine Warteschlange für eine vom Kunden verwaltete Flotte erstellen, deaktivieren Sie das Kontrollkästchen.
13. Wählen Sie Create queue (Warteschlange erstellen) aus.

Erstellen Sie eine Warteschlangenumgebung

Eine Warteschlangenumgebung besteht aus einer Reihe von Umgebungsvariablen und Befehlen, mit denen Flottenarbeiter eingerichtet werden. Sie können Warteschlangenumgebungen verwenden, um Softwareanwendungen, Umgebungsvariablen und andere Ressourcen für Jobs in der Warteschlange bereitzustellen.

Wenn Sie eine Warteschlange erstellen, haben Sie die Möglichkeit, eine Conda Standard-Warteschlangenumgebung zu erstellen. Diese Umgebung bietet vom Service verwalteten Flotten Zugriff auf Pakete für DCC-Anwendungen und Renderer von Partnern. Weitere Informationen finden Sie unter [CondaStandard-Warteschlangenumgebung](#).

Sie können Warteschlangenumgebungen mithilfe der Konsole hinzufügen oder indem Sie die JSON- oder YAML-Vorlage direkt bearbeiten. In diesem Verfahren wird beschrieben, wie Sie mit der Konsole eine Umgebung erstellen.

1. Um einer Warteschlange eine Warteschlangenumgebung hinzuzufügen, navigieren Sie zu der Warteschlange und wählen Sie die Registerkarte Warteschlangenumgebungen aus.
2. Wählen Sie „Aktionen“ und dann „Neues mit Formular erstellen“.
3. Geben Sie einen Namen und eine Beschreibung für die Warteschlangenumgebung ein.
4. Wählen Sie Neue Umgebungsvariable hinzufügen und geben Sie dann für jede hinzugefügte Variable einen Namen und einen Wert ein.
5. (Optional) Geben Sie eine Priorität für die Warteschlangenumgebung ein. Die Priorität gibt die Reihenfolge an, in der diese Warteschlangenumgebung auf dem Worker ausgeführt wird. Warteschlangenumgebungen mit höherer Priorität werden zuerst ausgeführt.
6. Wählen Sie Warteschlangenumgebung erstellen aus.

CondaStandard-Warteschlangenumgebung

Wenn Sie eine Warteschlange für eine vom Service verwaltete Flotte erstellen, haben Sie die Möglichkeit, eine Standard-Warteschlangenumgebung hinzuzufügen, die das Herunterladen und Installieren von Paketen in einer virtuellen Umgebung für Ihre Jobs unterstützt [Conda](#).

Conda stellt Pakete aus Kanälen bereit. Ein Kanal ist ein Ort, an dem Pakete gespeichert werden. Deadline Cloud stellt einen Kanal `deadline-cloud`, der Pakete hostet, die DCC-Anwendungen und Renderer von Partnern unterstützen. Die Pakete sind:

- Mixer
 - `blender=3.6`

- `blender-openjd`
- Houdini
 - `houdini=19.5`
 - `houdini-openjd`
- Maya
 - `maya=2024`
 - `maya-mtoa=2024.5.3`
 - `maya-openjd`
- Atombombe
 - `nuke=15`
 - `nuke-openjd`

Wenn Sie einen Job an eine Warteschlange mit der Conda Standardumgebung senden, fügt die Umgebung dem Job zwei Parameter hinzu. Diese Parameter geben die Conda Pakete und Kanäle an, die zur Konfiguration der Auftragsumgebung verwendet werden sollen, bevor die Aufgaben verarbeitet werden. Die Parameter sind:

- `CondaPackages`— eine durch Leerzeichen getrennte Liste von [Paketspezifikationen](#), wie z. B. `blender=3.6` oder `numpy>1.22`. Die Standardeinstellung ist leer, um die Erstellung einer virtuellen Umgebung zu überspringen.
- `CondaChannels`— eine durch Leerzeichen getrennte Liste von [CondaKanälen](#) wie `deadline-cloudconda-forge`, oder `s3://DOC-EXAMPLE-BUCKET/conda/channel`. Die Standardeinstellung ist ein Kanal `deadline-cloud`, der für vom Service verwaltete Flotten verfügbar ist und DCC-Anwendungen und Renderer von Partnern bereitstellt.

Wenn Sie einen integrierten Einreicher verwenden, um einen Job von Ihrem DCC an Deadline Cloud zu senden, füllt der Einreicher den Wert des Parameters auf der Grundlage der DCC-Anwendung und des Absenders aus. `CondaPackages` Wenn Sie beispielsweise Blender verwenden, ist der Parameter auf `blender=3.6.* blender-openjd=0.4.*` eingestellt.

Löschen einer Warteschlange

Warning

Sie können die Jobs in einer Warteschlange nicht wiederherstellen, wenn Sie die Warteschlange löschen. Durch das Löschen der Warteschlange werden auch die Jobs in dieser Warteschlange gelöscht.

1. Wählen Sie im Deadline Cloud-Dashboard die Option Farmen und andere Ressourcen aus.
2. Wählen Sie in der Farmenliste die Farm aus, die die zu löschende Warteschlange enthält.
3. Wählen Sie die Warteschlange aus und klicken Sie dann auf Löschen.
4. Wählen Sie im Bestätigungsfenster Delete. Ihre Warteschlange und alle Jobs in der Warteschlange werden gelöscht.

Bearbeiten einer Warteschlange

1. Wählen Sie im Deadline Cloud-Dashboard die Option Farmen und andere Ressourcen aus.
2. Wählen Sie in der Farmenliste die Farm aus, die die zu bearbeitende Warteschlange enthält.
3. Wählen Sie die Warteschlange aus und klicken Sie dann auf Bearbeiten.
4. Sie können den Namen, die Beschreibung, die Budgetanforderung, die Option Als Benutzer ausführen und die zugewiesene Servicerolle bearbeiten. Sie können Ihrer Warteschlange auch eine bestehende Flotte zuordnen.
5. Wählen Sie Änderungen speichern aus.

Ordnen Sie eine Warteschlange und eine Flotte zu

1. Wählen Sie die Warteschlange aus, die Sie einer Flotte zuordnen möchten.
2. Um eine Flotte auszuwählen, die Sie Ihrer Warteschlange zuordnen möchten, wählen Sie Flotten zuordnen.
3. Wählen Sie das Drop-down-Menü „Flotten auswählen“. Eine Liste der verfügbaren Flotten wird angezeigt.
4. Wählen Sie in der Liste der verfügbaren Flotten das Kontrollkästchen neben der Flotte oder den Flotten aus, die Sie Ihrer Warteschlange zuordnen möchten.

5. Wählen Sie Associate aus. Der Flottenzuordnungsstatus sollte jetzt Assoziiert lauten.

Deadline Cloud-Flotten verwalten

In diesem Abschnitt wird erklärt, wie Sie serviceverwaltete Flotten (SMF) und kundenverwaltete Flotten (CMF) für Deadline Cloud verwalten.

Sie können zwei Arten von Deadline Cloud-Flotten einrichten:

- Serviceverwaltete Flotten sind Flotten von Mitarbeitern, deren Standardeinstellungen von diesem Service, Deadline Cloud, bereitgestellt werden. Diese Standardeinstellungen sind so konzipiert, dass sie effizient und kostengünstig sind.
- Kundenverwaltete Flotten (CMFs) sind Flotten von Mitarbeitern, die Sie verwalten. Ein CMF kann sich innerhalb der AWS Infrastruktur, vor Ort oder in einem Rechenzentrum an einem anderen Standort befinden. Ein CMF bietet die volle Kontrolle und Verantwortung für die Flotte. Dazu gehören die Bereitstellung, der Betrieb, die Verwaltung und die Stilllegung der Mitarbeiter der Flotte.

Themen

- [Verwalten Sie vom Service verwaltete Flotten von Deadline Cloud](#)
- [Kundenverwaltete Flotten von Deadline Cloud verwalten](#)

Verwalten Sie vom Service verwaltete Flotten von Deadline Cloud

Serviceverwaltete Flotten sind Flotten von Mitarbeitern, deren Standardeinstellungen von Deadline Cloud bereitgestellt werden. Diese Standardeinstellungen sind so konzipiert, dass sie effizient und kostengünstig sind.

1. Um eine Service-Managed Fleet (SMF) zu erstellen, navigieren Sie zu der Farm, in der Sie die Flotte erstellen möchten.
2. Wählen Sie die Registerkarte Flotten aus.
3. Wählen Sie Create fleet (Flotte erstellen) aus.
4. Geben Sie einen Namen für Ihre Flotte ein.
5. Geben Sie eine Description ein. Eine klare Beschreibung kann Ihnen helfen, den Zweck Ihrer Flotte schnell zu erkennen.
6. Wählen Sie den Typ „Serviceverwaltete Flotte“ aus.

7. Wählen Sie für Ihre Flotte entweder die Option „Spot“ oder „On-Demand-Instance-Markt“. Spot-Instances sind unreservierte Kapazität, die Sie zu einem vergünstigten Preis nutzen können, die jedoch durch On-Demand-Anfragen unterbrochen werden kann. On-Demand-Instances werden sekundengenau berechnet, sind jedoch nicht langfristig gebunden und werden nicht unterbrochen. Standardmäßig verwenden Flotten Spot-Instances.
8. Optional: Legen Sie die maximale Anzahl von Instances fest, um die Flotte so zu skalieren, dass Kapazität für die Jobs in der Warteschlange verfügbar ist. Wir empfehlen, die Mindestanzahl an Instances beizubehalten, 0 um sicherzustellen, dass die Flotte alle Instances freigibt, wenn sich keine Jobs in der Warteschlange befinden.
9. Wählen Sie für den Servicezugriff für Ihre Flotte eine bestehende Rolle aus oder erstellen Sie eine neue Rolle. Eine Servicerolle stellt Anmeldeinformationen für Instanzen in der Flotte bereit und gewährt ihnen die Erlaubnis, Jobs zu verarbeiten, sowie für Benutzer im Monitor, sodass sie Protokollinformationen lesen können.
10. Wählen Sie Weiter aus.
11. Geben Sie die minimalen und maximalen vCPUs ein, die Sie für Ihre Flotte benötigen.
12. Geben Sie den minimalen und maximalen Arbeitsspeicher ein, den Sie für Ihre Flotte benötigen.
13. Optional: Sie können bestimmte Instance-Typen zulassen oder aus Ihrer Flotte ausschließen, um sicherzustellen, dass nur diese Instance-Typen für diese Flotte verwendet werden.
14. Optional Sie können die Größe des GP3-Volumes von Amazon Elastic Block Store (Amazon EBS) angeben, das den Mitarbeitern in dieser Flotte zugewiesen wird. Weitere Informationen finden Sie im [EBS-Benutzerhandbuch](#).
15. Wählen Sie Weiter aus.
16. Optional: Definieren Sie benutzerdefinierte Worker-Anforderungen, die die Funktionen dieser Flotte definieren. Diese können mit benutzerdefinierten Hostanforderungen kombiniert werden, die bei der Auftragseingabe angegeben werden. Ein Beispiel ist ein bestimmter Lizenztyp, wenn Sie Ihre Flotte mit Ihrem eigenen Lizenzserver verbinden möchten.
17. Wählen Sie Weiter aus.
18. Optional Um Ihre Flotte einer Warteschlange zuzuordnen, wählen Sie eine Warteschlange aus der Dropdownliste aus. Wenn die Warteschlange mit der Conda Standard-Warteschlangenumgebung eingerichtet ist, wird Ihre Flotte automatisch mit Paketen versorgt, die DCC-Anwendungen und -Renderer von Partnern unterstützen. Eine Liste der bereitgestellten Pakete finden Sie unter [CondaStandard-Warteschlangenumgebung](#)
19. Wählen Sie Weiter aus.

20. Optional Um Ihrer Flotte ein Tag hinzuzufügen, wählen Sie Neues Schlagwort hinzufügen und geben Sie dann den Schlüssel und den Wert für dieses Tag ein.
21. Wählen Sie Weiter aus.
22. Überprüfen Sie Ihre Flotteneinstellungen und wählen Sie dann Flotte erstellen. Nach der Erstellung wird Ihre Flotte angezeigt.

VFX Reference Platform-Kompatibilität

Das VFX Reference Platform ist eine gemeinsame Zielplattform für die VFX-Branche. Um die standardmäßige Amazon EC2 EC2-Instance für servicemanagierte Flotte zu verwenden, auf der Amazon Linux 2023 ausgeführt wird, mit SoftwareVFX Reference Platform, die das unterstützt, sollten Sie bei der Verwendung einer servicemanagierten Flotte die folgenden Überlegungen berücksichtigen.

Das wird jährlich VFX Reference Platform aktualisiert. Diese Überlegungen zur Nutzung eines AL2023, einschließlich der vom Service verwalteten Flotten von Deadline Cloud, basieren auf den Referenzplattformen für die Kalenderjahre (CY) 2022 bis 2024. Weitere Informationen finden Sie unter [VFX Reference Platform](#).

Note

Wenn Sie eine benutzerdefinierte Amazon Machine Image (AMI) für eine vom Kunden verwaltete Flotte erstellen, können Sie diese Anforderungen hinzufügen, wenn Sie die Amazon EC2 EC2-Instance vorbereiten.

Beachten Sie Folgendes, um VFX Reference Platform unterstützte Software auf einer AL2023 Amazon EC2 EC2-Instance zu verwenden:

- Die mit AL2023 installierte Glibc-Version ist für die Runtime-Nutzung kompatibel, jedoch nicht für die Erstellung von Software, die mit CY2024 oder früher kompatibel ist. VFX Reference Platform
- Python 3.9 und 3.11 sind mit der servicemanagierten Flotte ausgestattet, sodass sie mit VFX Reference Platform CY2022 und CY2024 kompatibel sind. Python 3.7 und 3.10 sind in der Service-Managed-Flotte nicht enthalten. Software, die sie benötigt, muss die Python-Installation in der Warteschlangen- oder Jobumgebung bereitstellen.
- Bei einigen Komponenten der Boost-Bibliothek, die in der vom Service verwalteten Flotte enthalten sind, handelt es sich um Version 1.75, die nicht kompatibel ist mit der. VFX Reference Platform

Wenn Ihre Anwendung Boost verwendet, müssen Sie aus Kompatibilitätsgründen Ihre eigene Version der Bibliothek bereitstellen.

- Intel TBB Update 3 ist Teil der Service-Managed-Flotte. Dies ist kompatibel mit VFX Reference Platform CY2022, CY2023 und CY2024.
- Andere Bibliotheken, deren Versionen von spezifiziert VFX Reference Platform sind, werden nicht von der vom Service verwalteten Flotte bereitgestellt. Sie müssen der Bibliothek alle Anwendungen zur Verfügung stellen, die in einer vom Service verwalteten Flotte verwendet werden. Eine Liste der Bibliotheken finden Sie [auf der Referenzplattform](#).

Kundenverwaltete Flotten von Deadline Cloud verwalten

In diesem Abschnitt wird erklärt, wie Sie eine vom Kunden verwaltete Flotte (CMF) für Deadline Cloud verwalten.

CMFs sind Flotten von Mitarbeitern, die Sie verwalten. Ein CMF kann sich innerhalb der AWS Infrastruktur, vor Ort oder in einem Rechenzentrum an einem anderen Standort befinden. Ein CMF bietet die volle Kontrolle und Verantwortung für die Flotte. Dazu gehören die Bereitstellung, der Betrieb, die Verwaltung und die Stilllegung der Mitarbeiter der Flotte.

Themen

- [Erstellen Sie eine vom Kunden verwaltete Flotte](#)
- [Einrichtung und Konfiguration des Worker-Hosts](#)
- [Zugriff auf Windows-Job-Benutzergeheimnisse verwalten](#)
- [Installation und Konfiguration der für Jobs erforderlichen Software](#)
- [AWS Anmeldeinformationen konfigurieren](#)
- [Erstellen eines Amazon Machine Image](#)
- [Erstellen Sie eine Flotteninfrastruktur mit einer Amazon EC2 Auto Scaling Scaling-Gruppe](#)
- [Vom Kunden verwaltete Flotten mit einem Lizenzendpunkt Connect](#)

Erstellen Sie eine vom Kunden verwaltete Flotte

Gehen Sie wie folgt vor, um eine kundenverwaltete Flotte (CMF) zu erstellen.

Deadline Cloud console

Um mit der Deadline Cloud-Konsole eine vom Kunden verwaltete Flotte zu erstellen

1. Öffnen Sie die Deadline [Cloud-Konsole](#).
2. Wählen Sie Farmen aus. Eine Liste der verfügbaren Farmen wird angezeigt.
3. Wählen Sie den Namen der Farm aus, in der Sie arbeiten möchten.
4. Wählen Sie die Registerkarte Flotten aus.
5. Wählen Sie Create fleet (Flotte erstellen) aus.
6. Geben Sie einen Namen für Ihre Flotte ein.
7. (Optional) Geben Sie eine Beschreibung für Ihre Flotte ein.
8. Wählen Sie als Flottenart die Option Vom Kunden verwaltet aus.
9. Wählen Sie einen Auto Scaling-Typ aus. Weitere Informationen finden Sie unter [Verwendung EventBridge zur Behandlung von Auto Scaling Scaling-Ereignissen](#).
 - Keine Skalierung: Sie erstellen eine Flotte vor Ort und möchten Deadline Cloud Auto Scaling deaktivieren.
 - Empfehlungen zur Skalierung: Sie erstellen eine Amazon Elastic Compute Cloud (Amazon EC2) -Flotte.
10. Wählen Sie den Servicezugang Ihrer Flotte aus.
 - a. Wir empfehlen, für jede Flotte die Option Neue Servicerolle erstellen und verwenden zu verwenden, um die Berechtigungen detaillierter steuern zu können. Diese Option ist standardmäßig ausgewählt.
 - b. Sie können auch eine bestehende Servicerolle verwenden, indem Sie eine Servicerolle auswählen auswählen.
11. Überprüfen Sie Ihre Auswahl und wählen Sie dann Weiter.
12. Wählen Sie ein Betriebssystem für Ihre Flotte aus. Alle Mitarbeiter einer Flotte müssen über ein gemeinsames Betriebssystem verfügen.
13. Wählen Sie die Host-CPU-Architektur aus.
14. Wählen Sie die folgenden Hardwareanforderungen für die Worker-Hosts in dieser Flotte aus.
 - a. Wählen Sie die minimalen und maximalen vCPU- und Speicher-Hardwareanforderungen aus, um den Workload-Anforderungen Ihrer Flotten gerecht zu werden.
 - b. (Optional) Wählen Sie GPU-Anforderung aus und geben Sie dann die minimalen und maximalen GPUs ein.
15. Überprüfen Sie Ihre Auswahl und wählen Sie dann Weiter.
16. (Optional) Definieren Sie benutzerdefinierte Worker-Anforderungen.

17. Wählen Sie in der Dropdownliste eine oder mehrere Warteschlangen aus, die Sie der Flotte zuordnen möchten.

 Note

Wir empfehlen, eine Flotte nur Warteschlangen zuzuordnen, die sich alle innerhalb derselben Vertrauensgrenze befinden. Dadurch wird eine starke Sicherheitsgrenze zwischen der Ausführung von Aufträgen auf demselben Worker gewährleistet.

18. Überprüfen Sie die Warteschlangenzuordnungen und wählen Sie dann Weiter aus.
19. (Optional) Für die standardmäßige Conda-Warteschlangenumgebung erstellen wir eine Umgebung für Ihre Warteschlange, in der die von Jobs angeforderten Conda-Pakete installiert werden.

 Note

Die Conda-Warteschlangenumgebung wird verwendet, um Conda-Pakete zu installieren, die von Jobs angefordert werden. Normalerweise sollten Sie die Conda-Warteschlangenumgebung für Warteschlangen deaktivieren, die mit CMFs verknüpft sind, da bei CMFs die erforderlichen Conda-Befehle standardmäßig nicht installiert sind.

20. (Optional) Fügen Sie Ihrem CMF Tags hinzu. Weitere Informationen finden Sie unter [Taggen Ihrer AWS Ressourcen](#).
21. Überprüfen Sie Ihre Flottenkonfiguration und nehmen Sie etwaige Änderungen vor.
22. Wählen Sie Create fleet (Flotte erstellen) aus.
23. Wählen Sie die Registerkarte Flotten aus und notieren Sie sich die Flotten-ID.

AWS CLI

Um den zu verwenden AWS CLI , um eine vom Kunden verwaltete Flotte zu erstellen

1. Öffnen Sie das AWS CLI.
2. Bearbeiten `fleet-trust-policy.json`.
 - a. Fügen Sie die folgende IAM-Richtlinie hinzu und ersetzen Sie den ***KURSIV GEDRUCKTEN*** Text durch Ihre AWS Konto-ID und Deadline Cloud-Farm-ID.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "credentials.deadline.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "ACCOUNT_ID"
        },
        "ArnEquals": {
          "aws:SourceArn":
            "arn:aws:deadline:*:ACCOUNT_ID:farm/FARM_ID"
        }
      }
    }
  ]
}
```

- b. Speichern Sie Ihre Änderungen.
3. Bearbeiten `create-cmf-fleet.json`.
 - a. Fügen Sie die folgende IAM-Richtlinie hinzu.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "deadline:AssumeFleetRoleForWorker",
        "deadline:UpdateWorker",
        "deadline>DeleteWorker",
        "deadline:UpdateWorkerSchedule",
        "deadline:BatchGetJobEntity",
        "deadline:AssumeQueueRoleForWorker"
      ],
      "Resource": "*",
      "Condition": {
```

```

        "StringEquals": {
            "aws:PrincipalAccount": "${aws:ResourceAccount}"
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "logs:CreateLogStream"
        ],
        "Resource": "arn:aws:logs:*:*:*:/aws/deadline/*",
        "Condition": {
            "StringEquals": {
                "aws:PrincipalAccount": "${aws:ResourceAccount}"
            }
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "logs:PutLogEvents",
            "logs:GetLogEvents"
        ],
        "Resource": "arn:aws:logs:*:*:*:/aws/deadline/*",
        "Condition": {
            "StringEquals": {
                "aws:PrincipalAccount": "${aws:ResourceAccount}"
            }
        }
    }
]
}

```

b. Speichern Sie Ihre Änderungen.

4. Fügen Sie eine IAM-Rolle hinzu, die die Mitarbeiter in Ihrer Flotte verwenden können.

```

aws iam create-role --role-name FleetWorkerRoleName --assume-role-policy-
document file://fleet-trust-policy.json
aws iam put-role-policy --role-name FleetWorkerRoleName --policy-name
FleetWorkerPolicy --policy-document file://fleet-policy.json

```

5. Bearbeiten `create-fleet-request.json`.

- a. Fügen Sie die folgende IAM-Richtlinie hinzu und ersetzen Sie den KURSIV gedruckten Text durch die Werte Ihres CMF.

 Note

*Sie finden den **ROLE_ARN** in der `create-cmf-fleet.json`*
Für die **OS_FAMILY** müssen Sie eines von, oder wählen. `linux macos windows`

```
{
  "farmId": "FARM_ID",
  "displayName": "FLEET_NAME",
  "description": "FLEET_DESCRIPTION",
  "roleArn": "ROLE_ARN",
  "minWorkerCount": 0,
  "maxWorkerCount": 10,
  "configuration": {
    "customerManaged": {
      "mode": "NO_SCALING",
      "workerCapabilities": {
        "vCpuCount": {
          "min": 1,
          "max": 4
        },
        "memoryMiB": {
          "min": 1024,
          "max": 4096
        },
        "osFamily": "OS_FAMILY",
        "cpuArchitectureType": "x86_64",
      },
    },
  },
}
```

- b. Speichern Sie Ihre Änderungen.
6. Erstellen Sie Ihre Flotte.

```
aws deadline create-fleet --cli-input-json file://create-fleet-request.json
```

Einrichtung und Konfiguration des Worker-Hosts

Ein Worker-Host bezieht sich auf einen Host-Computer, auf dem ein Deadline Cloud-Worker ausgeführt wird. In diesem Abschnitt wird erklärt, wie Sie den Worker-Host einrichten und für Ihre spezifischen Bedürfnisse konfigurieren. Jeder Worker-Host führt ein Programm aus, das als Worker-Agent bezeichnet wird. Der Worker-Agent ist verantwortlich für:

- Verwaltung des Lebenszyklus des Arbeitnehmers.
- Synchronisieren der zugewiesenen Arbeit, ihres Fortschritts und ihrer Ergebnisse.
- Überwachung laufender Arbeiten.
- Logs an konfigurierte Ziele weiterleiten.

Wir empfehlen Ihnen, den mitgelieferten Deadline Cloud-Worker-Agent zu verwenden. Der Worker-Agent ist Open Source und wir freuen uns über Funktionsanfragen, aber Sie können ihn auch entwickeln und an Ihre Bedürfnisse anpassen.

Um die Aufgaben in den folgenden Abschnitten ausführen zu können, benötigen Sie Folgendes:

Linux

- Eine Linux basierte Amazon Elastic Compute Cloud (Amazon EC2) -Instance. Wir empfehlen Amazon Linux 2023.
- sudoPrivilegien.
- Python 3.9 oder höher.

Windows

- Eine Windows basierte Amazon Elastic Compute Cloud (Amazon EC2) -Instance. Wir empfehlen Windows Server 2022.
- Administratorzugriff auf den Worker-Host
- Python 3.9 oder höher für alle Benutzer installiert

Erstellen und konfigurieren Sie eine virtuelle Python-Umgebung

Sie können eine virtuelle Python-Umgebung erstellen, Linux wenn Sie Python 3.9 oder höher installiert und in Ihrem platziert habenPATH.

So erstellen und aktivieren Sie eine virtuelle Python-Umgebung

1. Öffnen Sie das AWS CLI.
2. Erstellen und aktivieren Sie eine virtuelle Python-Umgebung.

```
python3 -m venv /opt/deadline/worker
source /opt/deadline/worker/bin/activate
pip install --upgrade pip
```

Installieren Sie den Deadline Cloud Worker Agent

Nachdem Sie Ihr Python eingerichtet und eine virtuelle Umgebung erstellt habenLinux, installieren Sie die Python-Pakete für den Deadline Cloud Worker Agent.

Um die Python-Pakete für den Worker Agent zu installieren

1. Öffnen Sie ein -Terminalfenster.
 - a. Öffnen Sie ein Terminal als root Benutzer (oder verwenden Siesudo/su) Linux
 - b. EinWindows, öffnen Sie eine Administrator-Befehlszeile oder ein PowerShell Terminal.
2. Laden Sie die Deadline Cloud Worker Agent-Pakete von PyPI herunter und installieren Sie sie:

Note

Bei Aktivierung Windows müssen die Agentendateien im globalen Site-Packages-Verzeichnis von Python installiert werden. Virtuelle Python-Umgebungen werden derzeit nicht unterstützt.

```
python -m pip install deadline-cloud-worker-agent
```

Konfigurieren Sie den Deadline Cloud Worker Agent

Sie können die Deadline Cloud-Worker-Agent-Einstellungen auf drei Arten konfigurieren. Wir empfehlen Ihnen, das über eingerichtete Betriebssystem zu verwenden `install-deadline-worker`.

Befehlszeilenargumente — Sie können Argumente angeben, wenn Sie den Deadline Cloud-Worker-Agent von der Befehlszeile aus ausführen. Einige Konfigurationseinstellungen sind nicht über Befehlszeilenargumente verfügbar. Um alle verfügbaren Befehlszeilenargumente `deadline-worker-agent --help` zu sehen, geben Sie ein, um alle verfügbaren Befehlszeilenargumente zu sehen.

Umgebungsvariablen — Sie können den Deadline Cloud-Worker-Agent konfigurieren, indem Sie die Umgebungsvariable festlegen, die mit `beginnt DEADLINE_WORKER_`. Sie können dies beispielsweise verwenden, `export DEADLINE_WORKER_VERBOSE=true` um die Ausgabe des Worker-Agents auf ausführlich zu setzen. Weitere Beispiele und Informationen finden Sie unter `/etc/amazon/deadline/worker.toml.example` on Linux oder `C:\ProgramData\Amazon\Deadline\Config\worker.toml.example` on Windows.

Konfigurationsdatei — Wenn Sie den Worker-Agent installieren, erstellt er eine Konfigurationsdatei, die sich unter `/etc/amazon/deadline/worker.toml` on Linux oder `C:\ProgramData\Amazon\Deadline\Config\worker.toml` on befindet Windows. Der Worker-Agent lädt diese Konfigurationsdatei, wenn er gestartet wird. Sie können die Beispielkonfigurationsdatei (aktiviert Linux oder `/etc/amazon/deadline/worker.toml.example` C:\ProgramData\Amazon\Deadline\Config\worker.toml.example aktiviert Windows) verwenden, um die Standard-Worker-Agent-Konfigurationsdatei an Ihre spezifischen Bedürfnisse anzupassen.

Schließlich empfehlen wir Ihnen, das `auto` Herunterfahren für den Worker Agent zu aktivieren. Auf diese Weise kann die Worker-Flotte bei Bedarf hochskaliert und heruntergefahren werden, wenn der Rendering-Job abgeschlossen ist. Durch die automatische Skalierung wird sichergestellt, dass Sie Ressourcen nur bei Bedarf verwenden.

Um die `auto` Abschaltung zu aktivieren

Als **root** Benutzer:

- Installieren Sie den Worker Agent mit Parametern **--allow-shutdown**.

Linux

Geben Sie ein:

```
/opt/deadline/worker/bin/install-deadline-worker \  
  --farm-id FARM_ID \  
  --fleet-id FLEET_ID \  
  --region REGION \  
  --allow-shutdown
```

Windows

Geben Sie ein:

```
install-deadline-worker ^  
  --farm-id FARM_ID ^  
  --fleet-id FLEET_ID ^  
  --region REGION ^  
  --allow-shutdown
```

Job-Benutzer und Gruppen erstellen

In diesem Abschnitt wird die erforderliche Benutzer- und Gruppenbeziehung zwischen dem Agent-Benutzer und den in Ihren Warteschlangen `jobRunAsUser` definierten Benutzern beschrieben.

Der Deadline Cloud-Worker-Agent sollte als dedizierter agentenspezifischer Benutzer auf dem Host ausgeführt werden. Sie sollten die `jobRunAsUser` Eigenschaft von Deadline Cloud-Warteschlangen so konfigurieren, dass Mitarbeiter die Warteschlangenjobs als ein bestimmter Betriebssystembenutzer und eine bestimmte Gruppe ausführen. Das bedeutet, dass Sie die gemeinsamen Dateisystemberechtigungen für Ihre Jobs kontrollieren können. Es stellt auch eine wichtige Sicherheitsgrenze zwischen Ihren Jobs und dem Worker-Agent-Benutzer dar.

LinuxJobbenutzer und Gruppen

Um Ihren Agent-Benutzer einzurichten und stellen Sie sicher `jobRunAsUser`, dass Sie die folgenden Anforderungen erfüllen:

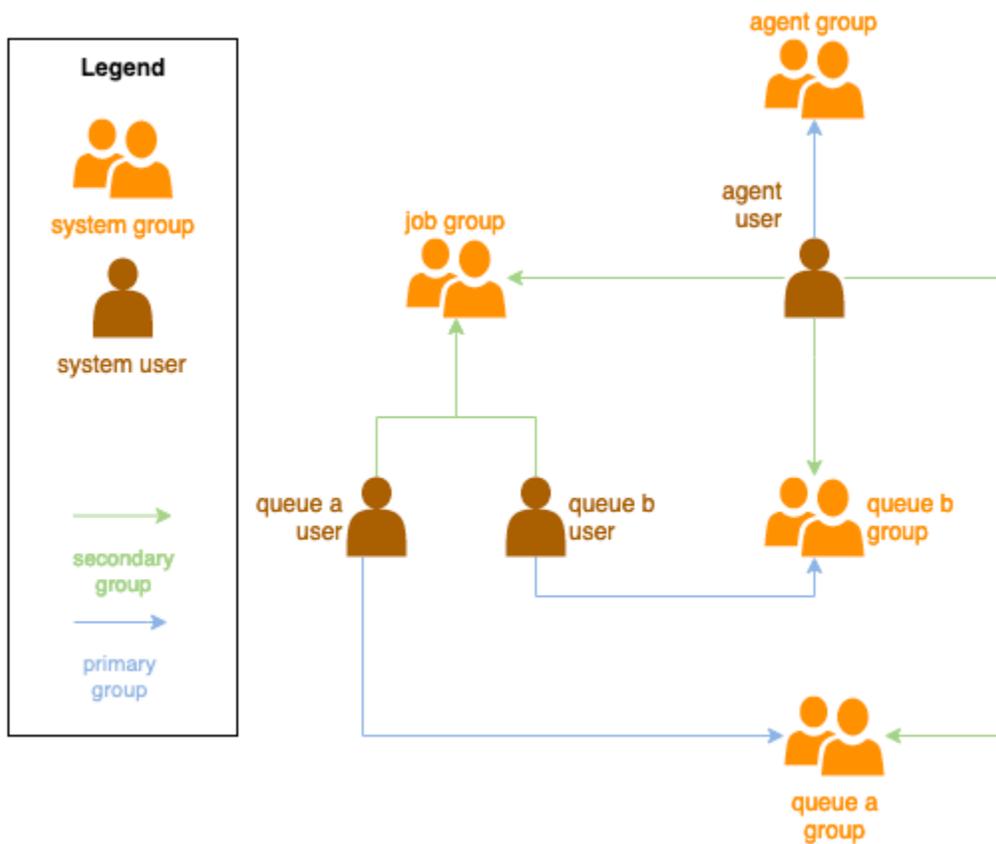
- Für jede Gruppe gibt es eine Gruppe `jobRunAsUser`, und diese ist die primäre Gruppe für die entsprechenden Gruppen. `jobRunAsUser`

- Der Agent-Benutzer gehört zur primären Gruppe der `jobRunAsUser` Warteschlangen, in denen der Mitarbeiter Arbeit erhält. Aus Sicherheitsgründen empfehlen wir, dies als sekundäre Gruppe der Agent-Benutzer zu verwenden. Diese gemeinsam genutzte Gruppe ermöglicht es dem Worker-Agent, Dateien für den Job verfügbar zu machen, während dieser ausgeführt wird.
- `A` gehört `jobRunAsUser` nicht zur primären Gruppe des Agent-Benutzers. Bewährte Sicherheitsmethoden finden Sie unter:
 - Vertrauliche Dateien, die vom Worker-Agenten geschrieben wurden, gehören der primären Gruppe des Agenten.
 - Wenn `a` zu dieser Gruppe `jobRunAsUser` gehört, können auf Dateien, die der Worker-Agent schreibt, die Jobs zugreifen, die an die Warteschlange weitergeleitet werden, die auf dem Worker ausgeführt wird.
- Die AWS Standardregion sollte der Region der Farm entsprechen, zu der der Worker gehört. Weitere Informationen finden Sie unter [Einstellungen für die Konfiguration und die Anmeldeinformationsdatei](#).

Dies sollte angewendet werden auf:

- Der Agent-Benutzer
- Alle `jobRunAsUser` Warteschlangenkonto auf dem Worker
- Der Agent-Benutzer kann `sudo` Befehle wie folgt ausführen. `jobRunAsUser`

Das folgende Diagramm veranschaulicht die Beziehung zwischen dem Agent-Benutzer und den `jobRunAsUser` Benutzern und Gruppen für Warteschlangen, die der Flotte zugeordnet sind.



Windows-Benutzer

Um einen Windows Benutzer als Benutzer verwenden zu können `jobRunAsUser`, muss er die folgenden Anforderungen erfüllen:

- Alle `jobRunAsUser` Warteschlangenbenutzer müssen vorhanden sein.
- Ihre Passwörter müssen mit dem Wert des Geheimnisses übereinstimmen, das im `JobRunAsUser` Feld ihrer Warteschlange angegeben ist. Anweisungen finden Sie in Schritt 7 unter [Erstellen einer Warteschlange](#).
- Der Agent-Benutzer muss sich als dieser Benutzer anmelden können.

Zugriff auf Windows-Job-Benutzergeheimnisse verwalten

Wenn Sie eine Warteschlange mit einem Windows konfigurieren `jobRunAsUser`, müssen Sie ein AWS Secrets Manager Manager-Geheimnis angeben. Es wird erwartet, dass der Wert dieses Geheimnisses ein JSON-kodiertes Objekt der folgenden Form ist:

```
{  
  "password": "JOB_USER_PASSWORD"  
}
```

Damit Worker Jobs so ausführen können, wie die Warteschlange konfiguriert ist `jobRunAsUser`, muss die IAM-Rolle der Flotte über die erforderlichen Berechtigungen verfügen, um den Wert des Geheimnisses abzurufen. Wenn das Geheimnis mit einem vom Kunden verwalteten KMS-Schlüssel verschlüsselt wird, muss die IAM-Rolle der Flotte auch über Berechtigungen zur Entschlüsselung mit dem KMS-Schlüssel verfügen.

Es wird dringend empfohlen, bei diesen Geheimnissen das Prinzip der geringsten Rechte einzuhalten. Das bedeutet, dass der Zugriff zum Abrufen des geheimen Werts von `jobRunAsUser` → `→ windows` einer Warteschlange wie folgt sein sollte: `passwordArn`

- wird einer Flottenrolle zugewiesen, wenn zwischen der Flotte und der Warteschlange eine Verbindung zwischen Warteschlange und Flotte erstellt wird
- wird einer Flottenrolle entzogen, wenn zwischen der Flotte und der Warteschlange eine Flottenverbindung zwischen der Flotte und der Warteschlange gelöscht wird

Außerdem sollte das AWS Secrets Manager Manager-Geheimnis, das das `jobRunAsUser` Passwort enthält, gelöscht werden, wenn es nicht mehr verwendet wird.

Gewähren Sie Zugriff auf ein geheimes Passwort

Deadline Cloud-Flotten benötigen Zugriff auf das `jobRunAsUser` Passwort, das im Passwortgeheimnis der Warteschlange gespeichert ist, wenn die Warteschlange und die Flotte verknüpft werden. Wir empfehlen, die AWS Secrets Manager Manager-Ressourcenrichtlinie zu verwenden, um Zugriff auf die Flottenrollen zu gewähren. Wenn Sie sich strikt an diese Richtlinie halten, ist es einfacher festzustellen, welche Flottenrollen Zugriff auf den geheimen Schlüssel haben.

Um Zugriff auf das Geheimnis zu gewähren

1. Öffnen Sie die AWS Secret Manager-Konsole für das Geheimnis.
2. Fügen Sie im Abschnitt „Ressourcenberechtigungen“ eine Richtlinienerklärung der folgenden Form hinzu:

```
{  
  "Version" : "2012-10-17",
```

```
"Statement" : [  
  // ...  
  {  
    "Effect" : "Allow",  
    "Principal" : {  
      "AWS" : "FLEET_ROLE_ARN"  
    },  
    "Action" : "secretsmanager:GetSecretValue",  
    "Resource" : "*"  
  }  
  // ...  
]  
}
```

Widerrufen Sie den Zugriff auf ein geheimes Passwort

Wenn eine Flotte keinen Zugriff mehr auf eine Warteschlange benötigt, entfernen Sie den Zugriff auf das geheime Passwort für die Warteschlange `jobRunAsUser`. Wir empfehlen, die AWS Secrets Manager Manager-Ressourcenrichtlinie zu verwenden, um Zugriff auf die Flottenrollen zu gewähren. Wenn Sie sich strikt an diese Richtlinie halten, ist es einfacher festzustellen, welche Flottenrollen Zugriff auf den geheimen Schlüssel haben.

Um den Zugriff auf das Geheimnis zu entziehen

1. Öffnen Sie die AWS Secret Manager-Konsole für das Geheimnis.
2. Entfernen Sie im Abschnitt Ressourcenberechtigungen die Richtlinienerklärung in der folgenden Form:

```
{  
  "Version" : "2012-10-17",  
  "Statement" : [  
    // ...  
    {  
      "Effect" : "Allow",  
      "Principal" : {  
        "AWS" : "FLEET_ROLE_ARN"  
      },  
      "Action" : "secretsmanager:GetSecretValue",  
      "Resource" : "*"  
    }  
    // ...  
  ]  
}
```

```
]
}
```

Installation und Konfiguration der für Jobs erforderlichen Software

Nachdem Sie den Deadline Cloud-Worker-Agent eingerichtet haben, können Sie den Worker-Host mit jeder Software vorbereiten, die für die Ausführung von Jobs erforderlich ist.

Wenn Sie einen Job an eine Warteschlange senden, der ein zugeordneter Job zugeordnet ist `jobRunAsUser`, wird der Job als dieser Benutzer ausgeführt. Alle Befehle müssen für diesen Benutzer verfügbar sein. `PATH`

Unter Linux können Sie das `PATH` für einen Benutzer in einer der folgenden Optionen angeben:

- ihr `~/ .bashrc` oder `~/ .bash_profile`
- Systemkonfigurationsdateien wie `/etc/profile.d/*` und `/etc/profile`
- Shell-Startskripte: `/etc/bashrc`.

Unter Windows können Sie das `PATH` für einen Benutzer in einer der folgenden Optionen angeben:

- ihre benutzerspezifischen Umgebungsvariablen
- die systemweiten Umgebungsvariablen

Installieren Sie die Adapter für Tools zur Erstellung digitaler Inhalte

Deadline Cloud bietet DCC-Anwendungen (Digital Content Creation) mit Integrationsunterstützung von Erstanbietern. Um diese Integrationen in einer vom Kunden verwalteten Flotte zu verwenden, müssen Sie die DCC-Software und die Adapter installieren.

Um DCC-Adapter in einer vom Kunden verwalteten Flotte zu installieren

1. Öffnen Sie das A-Terminal.
 - a. Öffnen Sie unter Linux ein Terminal als `root` Benutzer (oder verwenden `Siesudo/su`)
 - b. Öffnen Sie unter Windows eine Administrator-Befehlszeile oder ein PowerShell Terminal.
2. Installieren Sie die Deadline Cloud-Adapterpakete.

```
pip install deadline deadline-cloud-for-maya deadline-cloud-for-nuke deadline-  
cloud-for-blender
```

AWS Anmeldeinformationen konfigurieren

In diesem Abschnitt wird erklärt, wie AWS Anmeldeinformationen konfiguriert werden.

Bei dieser Anfangsphase des Lebenszyklus eines Mitarbeiters handelt es sich um Bootstrapping. In dieser Phase erstellt die Worker-Agent-Software einen Worker in Ihrer Flotte und bezieht die AWS Anmeldeinformationen für den weiteren Betrieb von der Rolle Ihres Fuhrparks.

AWS credentials for Amazon EC2

So konfigurieren Sie AWS Anmeldeinformationen für Amazon EC2

1. Öffnen Sie die IAM-Konsole unter <https://console.aws.amazon.com/iam/>.
2. Wählen Sie im Navigationsbereich Rollen und anschließend Rolle erstellen aus.
3. Wählen Sie AWS Dienst aus.
4. Wählen Sie EC2 als Dienst oder Anwendungsfall aus und wählen Sie dann Weiter aus.
5. Hängen Sie die `AWSDeadlineCloud-WorkerHost` AWS verwaltete Richtlinie an.

On-premise AWS credentials

Um AWS lokale Anmeldeinformationen zu konfigurieren

1. Öffnen Sie die IAM-Konsole unter <https://console.aws.amazon.com/iam/>.
2. Wählen Sie im Navigationsbereich Rollen und anschließend Rolle erstellen aus.
3. Wählen Sie AWS-Kontound dann Weiter aus.
4. Hängen Sie die `AWSDeadlineCloud-WorkerHost` AWS verwaltete Richtlinie an.
5. Generieren Sie AWS IAM-Zugriffs- und geheime Schlüssel für den IAM-Benutzer:
 - a. Informationen zu IAM Role Anywhere finden Sie unter [IAM](#) Roles Anywhere.
 - b. Informationen zur sichersten Methode zum Einrichten von Anmeldeinformationen auf dem Host finden Sie unter [Abrufen temporärer Sicherheitsanmeldedaten von AWS Identity and Access Management Roles Anywhere](#).

- c. Sie können CLI auch als alternative Authentifizierung verwenden. Weitere Informationen finden Sie unter [Mit IAM-Benutzeranmeldeinformationen authentifizieren](#).
6. Speichern Sie diese Schlüssel in der Datei mit den AWS Anmeldeinformationen des Agent-Benutzers auf dem Worker-Host-Dateisystem.
 - a. Unter Linux befindet sich dies unter `~/.aws/credentials`
 - b. Unter Windows befindet sich dies unter `%USERPROFILE%\aws\credentials`

 Note

Auf die Anmeldeinformationen sollte nur der Betriebssystem-Benutzername (`deadline-worker-agent`) zugreifen können, der den Worker-Agent installiert hat.

```
# Replace keys below
[default]
aws_access_key_id=ACCESS_KEY_ID
aws_secret_access_key=SECRET_ACCESSS_KEY
```

7. Ändern Sie den `deadline-worker-agent` Besitzer und die Berechtigungen.

 Note

Wenn Sie den Betriebssystem-Benutzernamen (`deadline-worker-agent`) bei der Installation des Worker Agents geändert haben, verwenden Sie stattdessen diesen Namen.

Erstellen eines Amazon Machine Image

Um ein Amazon Machine Image (AMI) für die Verwendung in einer kundenverwalteten Amazon Elastic Compute Cloud (Amazon EC2) -Flotte (CMF) zu erstellen, führen Sie die Aufgaben in diesem Abschnitt aus. Sie müssen eine Amazon EC2 EC2-Instance erstellen, bevor Sie fortfahren können. Weitere Informationen finden Sie unter [Starten Ihrer Instance](#) im Amazon EC2 EC2-Benutzerhandbuch für Linux-Instances.

⚠ Important

Beim Erstellen und Erstellen AMI eines Snapshots der angehängten Volumes der Amazon EC2 EC2-Instance. Jegliche auf der Instance installierte Software bleibt bestehen, also Instances, die wiederverwendet werden, wenn Sie Instances von der aus starten. AMI Wir empfehlen, eine Patch-Strategie zu verfolgen und alle neuen Patches regelmäßig AMI mit aktualisierter Software zu aktualisieren, bevor Sie sie auf Ihre Flotte anwenden.

Bereiten Sie die Amazon EC2 EC2-Instance vor

Bevor Sie eine erstellenAMI, müssen Sie den Worker-Status löschen. Der Worker-Status bleibt auch zwischen den Starts des Worker-Agents bestehen. Bleibt dieser Status auf dem bestehenAMI, haben alle Instances, die von dort aus gestartet werden, denselben Status.

Wir empfehlen außerdem, alle vorhandenen Protokolldateien zu löschen. Protokolldateien können auf einer Amazon EC2 EC2-Instance verbleiben, wenn Sie das AMI vorbereiten. Durch das Löschen dieser Dateien wird Verwirrung bei der Diagnose möglicher Probleme in Arbeiterflotten, die das AMI verwenden, minimiert.

Sie sollten auch den Worker-Agent-Systemdienst aktivieren, damit der Deadline Cloud-Worker-Agent gestartet wird, wenn Amazon EC2 gestartet wird.

Schließlich empfehlen wir Ihnen, das auto Herunterfahren des Worker-Agents zu aktivieren. Auf diese Weise kann die Worker-Flotte bei Bedarf hochskaliert und heruntergefahren werden, wenn der Rendering-Job abgeschlossen ist. Diese auto Skalierung stellt sicher, dass Sie Ressourcen nur nach Bedarf verwenden.

So bereiten Sie die Amazon EC2 EC2-Instance vor

1. Öffnen Sie die Amazon EC2-Konsole.
2. Starten Sie eine Amazon-EC2-Instance. Weitere Informationen finden Sie unter [Starten Sie Ihre Instance](#).
3. Richten Sie den Host so ein, dass er eine Verbindung zu Ihrem Identity Provider (IdP) herstellt, und mounten Sie dann jedes gemeinsam genutzte Dateisystem, das er benötigt.
4. Folgen Sie den Anleitungen zu [Installieren Sie den Deadline Cloud Worker AgentWorker Agent konfigurieren](#), dann und. [Job-Benutzer und Gruppen erstellen](#)

5. Wenn Sie eine auf Amazon Linux 2023 AMI basierende Software für die Ausführung vorbereiten, die mit der VFX-Referenzplattform kompatibel ist, müssen Sie mehrere Anforderungen aktualisieren. Weitere Informationen finden Sie unter [VFX Reference Platform-Kompatibilität](#).
6. Öffnen Sie ein -Terminalfenster.
 - a. Öffnen Sie unter Linux ein Terminal als `root` Benutzer (oder verwenden Sie `sudo/su`)
 - b. Öffnen Sie unter Windows eine Administrator-Befehlszeile oder ein PowerShell Terminal.
7. Stellen Sie sicher, dass der Worker-Dienst nicht läuft und nicht so konfiguriert ist, dass er beim Booten gestartet wird:

- a. Führen Sie unter Linux Folgendes aus

```
systemctl stop deadline-worker  
systemctl enable deadline-worker
```

- b. Führen Sie unter Windows Folgendes aus

```
sc.exe stop DeadlineWorker  
sc.exe config DeadlineWorker start= auto
```

8. Löschen Sie den Arbeiterstatus.

- a. Führen Sie unter Linux Folgendes aus

```
rm -rf /var/lib/deadline/*
```

- b. Führen Sie unter Windows Folgendes aus

```
del /Q /S %PROGRAMDATA%\Amazon\Deadline\Cache\*
```

9. Löschen Sie die Protokolldateien.

- a. Führen Sie unter Linux Folgendes aus

```
rm -rf /var/log/amazon/deadline/*
```

- b. Führen Sie unter Windows Folgendes aus

```
del /Q /S %PROGRAMDATA%\Amazon\Deadline\Logs\*
```

10. Unter Windows wird empfohlen, die Anwendung Amazon EC2Launch Settings im Startmenü auszuführen, um die letzte Vorbereitung und das Herunterfahren der Instance auf dem Host abzuschließen.

Note

Sie MÜSSEN Shutdown ohne Sysprep wählen und dürfen niemals Shutdown with Sysprep wählen. Beim Herunterfahren mit Sysprep werden alle lokalen Benutzer unbrauchbar. Weitere Informationen finden Sie im [Abschnitt Bevor Sie beginnen des Themas Erstellen eines benutzerdefinierten AMIs im Benutzerhandbuch für Windows-Instances](#).

Erstellen Sie das AMI

Um das zu bauen AMI

1. Öffnen Sie die Amazon EC2-Konsole.
2. Wählen Sie im Navigationsbereich Instances und dann Ihre Instance aus.
3. Wählen Sie Instanzstatus und dann Instanz beenden aus.
4. Nachdem die Instance gestoppt wurde, wählen Sie Actions aus.
5. Wählen Sie „Bild und Vorlagen“ und anschließend „Bild erstellen“.
6. Geben Sie einen Bildnamen ein.
7. (Optional) Geben Sie eine Beschreibung für Ihr Bild ein.
8. Wählen Sie Create Image (Image erstellen) aus.

Erstellen Sie eine Flotteninfrastruktur mit einer Amazon EC2 Auto Scaling Scaling-Gruppe

In diesem Abschnitt wird erklärt, wie Sie eine Amazon EC2 Auto Scaling Scaling-Flotte erstellen.

Verwenden Sie die folgende AWS CloudFormation YAML-Vorlage, um eine Amazon EC2 Auto Scaling (Auto Scaling) -Gruppe, eine Amazon Virtual Private Cloud (Amazon VPC) mit zwei Subnetzen, einem Instance-Profil und einer Instance-Zugriffsrolle zu erstellen. Diese sind erforderlich, um die Instance mithilfe von Auto Scaling in den Subnetzen zu starten.

Sie sollten die Liste der Instance-Typen überprüfen und aktualisieren, damit sie Ihren Rendering-Anforderungen entspricht.

So erstellen Sie eine Amazon EC2 Auto Scaling Scaling-Flotte

1. Öffnen Sie die AWS CloudFormation Konsole unter <https://console.aws.amazon.com/cloudformation>.
2. Erstellen Sie eine CloudFormation Vorlage mit den Parametern Farm ID, Fleet ID, und AMI ID.

```

AWSTemplateFormatVersion: 2010-09-09
Description: Amazon Deadline Cloud customer-managed fleet
Parameters:
  FarmId:
    Type: String
    Description: Farm ID
  FleetId:
    Type: String
    Description: Fleet ID
  AMIID:
    Type: String
    Description: AMI ID for launching Workers
Resources:
  deadlineVPC:
    Type: 'AWS::EC2::VPC'
    Properties:
      CidrBlock: 100.100.0.0/16
  deadlineWorkerSecurityGroup:
    Type: 'AWS::EC2::SecurityGroup'
    Properties:
      GroupDescription: !Join
        - ' '
        - - Security Group created for deadline workers in fleet
          - !Ref FleetId
      GroupName: !Join
        - ' '
        - - deadlineWorkerSecurityGroup-
          - !Ref FleetId
      SecurityGroupEgress:
        - CidrIp: 0.0.0.0/0
          IpProtocol: '-1'
      SecurityGroupIngress: []

```

```
VpcId: !Ref deadlineVPC
deadlineIGW:
  Type: 'AWS::EC2::InternetGateway'
  Properties: {}
deadlineVPCGatewayAttachment:
  Type: 'AWS::EC2::VPCGatewayAttachment'
  Properties:
    VpcId: !Ref deadlineVPC
    InternetGatewayId: !Ref deadlineIGW
deadlinePublicRouteTable:
  Type: 'AWS::EC2::RouteTable'
  Properties:
    VpcId: !Ref deadlineVPC
deadlinePublicRoute:
  Type: 'AWS::EC2::Route'
  Properties:
    RouteTableId: !Ref deadlinePublicRouteTable
    DestinationCidrBlock: 0.0.0.0/0
    GatewayId: !Ref deadlineIGW
  DependsOn:
    - deadlineIGW
    - deadlineVPCGatewayAttachment
deadlinePublicSubnet0:
  Type: 'AWS::EC2::Subnet'
  Properties:
    VpcId: !Ref deadlineVPC
    CidrBlock: 100.100.16.0/22
    AvailabilityZone: !Join
      - ''
      - - !Ref 'AWS::Region'
      - a
deadlineSubnetRouteTableAssociation0:
  Type: 'AWS::EC2::SubnetRouteTableAssociation'
  Properties:
    RouteTableId: !Ref deadlinePublicRouteTable
    SubnetId: !Ref deadlinePublicSubnet0
deadlinePublicSubnet1:
  Type: 'AWS::EC2::Subnet'
  Properties:
    VpcId: !Ref deadlineVPC
    CidrBlock: 100.100.20.0/22
    AvailabilityZone: !Join
      - ''
      - - !Ref 'AWS::Region'
```

```

    - c
deadlineSubnetRouteTableAssociation1:
  Type: 'AWS::EC2::SubnetRouteTableAssociation'
  Properties:
    RouteTableId: !Ref deadlinePublicRouteTable
    SubnetId: !Ref deadlinePublicSubnet1
deadlineInstanceAccessAccessRole:
  Type: 'AWS::IAM::Role'
  Properties:
    RoleName: !Join
      - '-'
      - - deadline
        - InstanceAccess
        - !Ref FleetId
    AssumeRolePolicyDocument:
      Statement:
        - Effect: Allow
          Principal:
            Service: ec2.amazonaws.com
          Action:
            - 'sts:AssumeRole'
      Path: /
    ManagedPolicyArns:
      - 'arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy'
      - 'arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore'
      - 'arn:aws:iam::aws:policy/AWSDeadlineCloud-WorkerHost'
deadlineInstanceProfile:
  Type: 'AWS::IAM::InstanceProfile'
  Properties:
    Path: /
    Roles:
      - !Ref deadlineInstanceAccessAccessRole
deadlineLaunchTemplate:
  Type: 'AWS::EC2::LaunchTemplate'
  Properties:
    LaunchTemplateName: !Join
      - ''
      - - deadline-LT-
        - !Ref FleetId
    LaunchTemplateData:
      NetworkInterfaces:
        - DeviceIndex: 0
          AssociatePublicIpAddress: true
      Groups:

```

```
    - !Ref deadlineWorkerSecurityGroup
      DeleteOnTermination: true
    ImageId: !Ref AMIID
    InstanceInitiatedShutdownBehavior: terminate
    IamInstanceProfile:
      Arn: !GetAtt
        - deadlineInstanceProfile
        - Arn
    MetadataOptions:
      HttpTokens: required
      HttpEndpoint: enabled

deadlineAutoScalingGroup:
  Type: 'AWS::AutoScaling::AutoScalingGroup'
  Properties:
    AutoScalingGroupName: !Join
      - ''
      - - deadline-ASG-autoscalable-
        - !Ref FleetId
    MinSize: 0
    MaxSize: 10
    VPCZoneIdentifier:
      - !Ref deadlinePublicSubnet0
      - !Ref deadlinePublicSubnet1
    NewInstancesProtectedFromScaleIn: true
    MixedInstancesPolicy:
      InstancesDistribution:
        OnDemandBaseCapacity: 0
        OnDemandPercentageAboveBaseCapacity: 0
        SpotAllocationStrategy: capacity-optimized
        OnDemandAllocationStrategy: lowest-price
    LaunchTemplate:
      LaunchTemplateSpecification:
        LaunchTemplateId: !Ref deadlineLaunchTemplate
        Version: !GetAtt
          - deadlineLaunchTemplate
          - LatestVersionNumber
    Overrides:
      - InstanceType: m5.large
      - InstanceType: m5d.large
      - InstanceType: m5a.large
      - InstanceType: m5ad.large
      - InstanceType: m5n.large
      - InstanceType: m5dn.large
```

```
- InstanceType: m4.large
- InstanceType: m3.large
- InstanceType: r5.large
- InstanceType: r5d.large
- InstanceType: r5a.large
- InstanceType: r5ad.large
- InstanceType: r5n.large
- InstanceType: r5dn.large
- InstanceType: r4.large
MetricsCollection:
  - Granularity: 1Minute
  Metrics:
    - GroupMinSize
    - GroupMaxSize
    - GroupDesiredCapacity
    - GroupInServiceInstances
    - GroupTotalInstances
    - GroupInServiceCapacity
    - GroupTotalCapacity
```

3. Nachdem Sie die IAM-Rollen erstellt haben, müssen Sie Folgendes bestätigen:
 - Anmeldeinformationen aus der IAM-Rolle, die an die Amazon EC2 EC2-Instance Ihres Workers angehängt sind, sind für alle Prozesse verfügbar, die auf diesem Worker ausgeführt werden, einschließlich Jobs. Der Mitarbeiter sollte die geringsten Betriebsberechtigungen haben: `und deadline:CreateWorker deadline:AssumeFleetRoleForWorker`.
 - Der Worker-Agent ruft die Anmeldeinformationen für die Warteschlangenrolle ab und konfiguriert sie für die Verwendung bei der Ausführung von Jobs. Die Amazon EC2 EC2-Instance-Profilrolle sollte keine Berechtigungen enthalten, die für Ihre Jobs erforderlich sind.

Skalieren Sie Ihre Amazon EC2 EC2-Flotte automatisch mit der Deadline Cloud-Funktion für Skalierungsempfehlungen

Deadline Cloud nutzt eine Amazon EC2 Auto Scaling (Auto Scaling) -Gruppe, um die vom Kunden verwaltete Amazon EC2 EC2-Flotte (CMF) automatisch zu skalieren. Sie müssen den Flottenmodus konfigurieren und die erforderliche Infrastruktur in Ihrem Konto bereitstellen, damit Ihre Flotte auto skaliert werden kann. Die von Ihnen bereitgestellte Infrastruktur funktioniert für alle Flotten, sodass Sie sie nur einmal einrichten müssen.

Der grundlegende Arbeitsablauf ist: Sie konfigurieren Ihren Flottenmodus so, dass er auto skaliert, und dann sendet Deadline Cloud ein EventBridge Ereignis für diese Flotte aus, wenn sich die empfohlene Flottengröße ändert (ein Ereignis enthält die Flotten-ID, die empfohlene Flottengröße und andere Metadaten). Sie werden eine EventBridge Regel haben, um die relevanten Ereignisse zu filtern, und ein Lambda, um sie zu verarbeiten. Das Lambda wird in Amazon EC2 Auto Scaling integriert, AutoScalingGroup um die Amazon EC2 EC2-Flotte automatisch zu skalieren.

Stellen Sie den Flottenmodus auf **EVENT_BASED_AUTO_SCALING**

Konfigurieren Sie Ihren Flottenmodus auf **EVENT_BASED_AUTO_SCALING**. Sie können dazu die Konsole verwenden oder die verwenden, AWS CLI um die `CreateFleet` `UpdateFleet` OR-API direkt aufzurufen. Nachdem der Modus konfiguriert wurde, beginnt Deadline Cloud mit dem Senden von EventBridge Ereignissen, sobald sich die empfohlene Flottengröße ändert.

- `UpdateFleet` Beispielbefehl:

```
aws deadline update-fleet \  
  --farm-id FARM_ID \  
  --fleet-id FLEET_ID \  
  --configuration file://configuration.json
```

- `CreateFleet` Beispielbefehl:

```
aws deadline create-fleet \  
  --farm-id FARM_ID \  
  --display-name "Fleet name" \  
  --max-worker-count 10 \  
  --configuration file://configuration.json
```

Das Folgende ist ein Beispiel für die `configuration.json` Verwendung in den obigen CLI-Befehlen (`--configuration file://configuration.json`).

- Um Auto Scaling für Ihre Flotte zu aktivieren, sollten Sie den Modus auf `EVENT_BASED_AUTO_SCALING` einstellen.
- Dies `workerCapabilities` sind die Standardwerte, die dem CMF bei der Erstellung zugewiesen wurden. Sie können diese Werte ändern, wenn Sie mehr Ressourcen benötigen, die Ihrem CMF zur Verfügung stehen.

Nachdem Sie den Flottenmodus konfiguriert haben, sendet Deadline Cloud Ereignisse mit Empfehlungen zur Flottengröße für diese Flotte aus.

```
{
  "customerManaged": {
    "mode": "EVENT_BASED_AUTO_SCALING",
    "workerCapabilities": {
      "vCpuCount": {
        "min": 1,
        "max": 4
      },
      "memoryMiB": {
        "min": 1024,
        "max": 4096
      },
      "osFamily": "linux",
      "cpuArchitectureType": "x86_64",
    }
  }
}
```

Stellen Sie den Auto Scaling Scaling-Stack mithilfe der AWS CloudFormation Vorlage bereit

Sie können eine EventBridge Regel zum Filtern von Ereignissen, ein Lambda zum Verwerten der Ereignisse und zur Steuerung von Auto Scaling und eine SQS-Warteschlange zum Speichern unverarbeiteter Ereignisse einrichten. Verwenden Sie die folgende AWS CloudFormation Vorlage, um alles in einem Stack bereitzustellen. Nachdem Sie die Ressourcen erfolgreich bereitgestellt haben, können Sie einen Auftrag einreichen und die Flotte wird automatisch skaliert.

Resources:

AutoScalingLambda:

Type: 'AWS::Lambda::Function'

Properties:

Code:

ZipFile: |-
 """

This lambda is configured to handle "Fleet Size Recommendation Change" messages. It will handle all such events, and requires that the ASG is named based on the fleet id. It will scale up/down the fleet based on the recommended fleet size in the message.

Example EventBridge message:

```
{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "Fleet Size Recommendation Change",
  "source": "aws.deadline",
  "account": "111122223333",
  "time": "2017-12-22T18:43:48Z",
  "region": "us-west-1",
  "resources": [],
  "detail": {
    "farmId": "farm-12345678900000000000000000000000",
    "fleetId": "fleet-12345678900000000000000000000000",
    "oldFleetSize": 1,
    "newFleetSize": 5,
  }
}
```

```
import json
import boto3
import logging

logger = logging.getLogger()
logger.setLevel(logging.INFO)

auto_scaling_client = boto3.client("autoscaling")

def lambda_handler(event, context):
    logger.info(event)
    event_detail = event["detail"]
    fleet_id = event_detail["fleetId"]
    desired_capacity = event_detail["newFleetSize"]

    asg_name = f"deadline-ASG-autoscalable-{fleet_id}"
    auto_scaling_client.set_desired_capacity(
        AutoScalingGroupName=asg_name,
        DesiredCapacity=desired_capacity,
        HonorCooldown=False,
    )

    return {
        'statusCode': 200,
        'body': json.dumps(f'Successfully set desired_capacity for {asg_name}
to {desired_capacity}')
```

```
    }
    Handler: index.lambda_handler
    Role: !GetAtt
      - AutoScalingLambdaServiceRole
      - Arn
    Runtime: python3.11
    DependsOn:
      - AutoScalingLambdaServiceRoleDefaultPolicy
      - AutoScalingLambdaServiceRole
    AutoScalingEventRule:
      Type: 'AWS::Events::Rule'
      Properties:
        EventPattern:
          source:
            - aws.deadline
          detail-type:
            - Fleet Size Recommendation Change
        State: ENABLED
      Targets:
        - Arn: !GetAtt
          - AutoScalingLambda
          - Arn
        DeadLetterConfig:
          Arn: !GetAtt
            - UnprocessedAutoScalingEventQueue
            - Arn
        Id: Target0
        RetryPolicy:
          MaximumRetryAttempts: 15
    AutoScalingEventRuleTargetPermission:
      Type: 'AWS::Lambda::Permission'
      Properties:
        Action: 'lambda:InvokeFunction'
        FunctionName: !GetAtt
          - AutoScalingLambda
          - Arn
        Principal: events.amazonaws.com
        SourceArn: !GetAtt
          - AutoScalingEventRule
          - Arn
    AutoScalingLambdaServiceRole:
      Type: 'AWS::IAM::Role'
      Properties:
        AssumeRolePolicyDocument:
```

```
Statement:
  - Action: 'sts:AssumeRole'
    Effect: Allow
    Principal:
      Service: lambda.amazonaws.com
Version: 2012-10-17
ManagedPolicyArns:
  - !Join
    - ''
    - - 'arn:'
      - !Ref 'AWS::Partition'
      - ':iam::aws:policy/service-role/AWSLambdaBasicExecutionRole'
AutoScalingLambdaServiceRoleDefaultPolicy:
Type: 'AWS::IAM::Policy'
Properties:
  PolicyDocument:
    Statement:
      - Action: 'autoscaling:SetDesiredCapacity'
        Effect: Allow
        Resource: '*'
    Version: 2012-10-17
  PolicyName: AutoScalingLambdaServiceRoleDefaultPolicy
Roles:
  - !Ref AutoScalingLambdaServiceRole
UnprocessedAutoScalingEventQueue:
Type: 'AWS::SQS::Queue'
Properties:
  QueueName: deadline-unprocessed-autoscaling-events
  UpdateReplacePolicy: Delete
  DeletionPolicy: Delete
UnprocessedAutoScalingEventQueuePolicy:
Type: 'AWS::SQS::QueuePolicy'
Properties:
  PolicyDocument:
    Statement:
      - Action: 'sqs:SendMessage'
        Condition:
          ArnEquals:
            'aws:SourceArn': !GetAtt
              - AutoScalingEventRule
              - Arn
        Effect: Allow
        Principal:
          Service: events.amazonaws.com
```

```
Resource: !GetAtt
  - UnprocessedAutoScalingEventQueue
  - Arn
Version: 2012-10-17
Queues:
  - !Ref UnprocessedAutoScalingEventQueue
```

Vom Kunden verwaltete Flotten mit einem Lizenzendpunkt Connect

Der nutzungsbasierte Lizenzserver von AWS Deadline Cloud (Deadline Cloud) bietet On-Demand-Lizenzen für ausgewählte Produkte von Drittanbietern. Auf diese Weise können Sie nach Bedarf zahlen. Sie ändern sich nur für die Zeit, die Sie nutzen.

Der nutzungsbasierte Lizenzserver von Deadline Cloud kann mit jedem Flottentyp verwendet werden, sofern die Deadline Cloud-Mitarbeiter mit dem Lizenzserver kommunizieren können. Dies wird automatisch in serviceverwalteten Flotten eingerichtet. Dieses Setup ist nur für vom Kunden verwaltete Flotten erforderlich.

Um den Lizenzserver zu erstellen, benötigen Sie Folgendes:

- Eine Sicherheitsgruppe für die VPC Ihrer Farm, die Datenverkehr für Drittanbieterlizenzen zulässt.
- Eine AWS Identity and Access Management (IAM-) Rolle mit einer angehängten Richtlinie, die den Zugriff auf die Endpunktoperationen der Deadline Cloud-Lizenz ermöglicht.

Themen

- [Schritt 1: Erstellen Sie eine Sicherheitsgruppe](#)
- [Schritt 2: Richten Sie den Lizenzendpunkt ein](#)
- [Schritt 3: Eine Rendering-Anwendung mit einem Endpunkt Connect](#)

Schritt 1: Erstellen Sie eine Sicherheitsgruppe

Verwenden Sie die Amazon VPC-Konsole (<https://console.aws.amazon.com/vpc/>), um eine Sicherheitsgruppe für die VPC Ihrer Farm zu erstellen. Konfigurieren Sie die Sicherheitsgruppe so, dass sie die folgenden Regeln für eingehenden Datenverkehr zulässt:

- Autodesk Maya und Arnold — 2701 — 2702, TCP, IPv4
- Autodesk 3ds Max — 2704, TCP, IPv4
- Foundry Nuke — 6101, TCP, IPv4

- SideFX Houdini, Mantra und Karma — 1715 — 1717, TCP, IPv4

Die Quelle für jede eingehende Regel ist die Worker-Sicherheitsgruppe der Flotte.

Weitere Informationen zum Erstellen einer Sicherheitsgruppe finden Sie unter [Erstellen einer Sicherheitsgruppe](#) im Amazon Virtual Private Cloud Cloud-Benutzerhandbuch.

Schritt 2: Richten Sie den Lizenzendpunkt ein

Ein Lizenzendpunkt ermöglicht den Zugriff auf Lizenzserver für Produkte von Drittanbietern. Lizenzanfragen werden an den Lizenzendpunkt gesendet. Der Endpunkt leitet sie an den entsprechenden Lizenzserver weiter. Der Lizenzserver verfolgt Nutzungsbeschränkungen und Berechtigungen. Für jeden Lizenzendpunkt, den Sie erstellen, fallen Gebühren an. Weitere Informationen dazu finden Sie unter [Preise für Amazon VPC](#).

Sie können Ihren Lizenzendpunkt AWS Command Line Interface mit den entsprechenden Berechtigungen von aus erstellen. Die für die Erstellung eines Lizenzendpunkts erforderliche Richtlinie finden Sie unter [Richtlinie zur Zulassung der Erstellung eines Lizenzendpunkts](#).

Sie können die AWS CloudShell (<https://console.aws.amazon.com/cloudshell/>) oder eine andere AWS CLI Umgebung verwenden, um den Lizenzendpunkt mit den folgenden AWS Command Line Interface Befehlen zu konfigurieren.

1. Erstellen Sie den Lizenzendpunkt. Ersetzen Sie die Sicherheitsgruppen-ID, Subnetz-ID und VPC-ID durch die Werte, die Sie zuvor erstellt haben. Wenn Sie mehrere Subnetze verwenden, trennen Sie sie durch Leerzeichen.

```
aws deadline create-license-endpoint \  
  --security-group-id SECURITY_GROUP_ID \  
  --subnet-ids SUBNET_ID1 SUBNET_ID2 \  
  --vpc-id VPC_ID
```

2. Bestätigen Sie mit dem folgenden Befehl, dass der Endpunkt erfolgreich erstellt wurde. Merken Sie sich den DNS-Namen des VPC-Endpunkts.

```
aws deadline get-license-endpoint \  
  --license-endpoint-id LICENSE_ENDPOINT_ID
```

3. Sehen Sie sich eine Liste der verfügbaren Produkte mit Messgerät an:

```
aws deadline list-available-metered-products
```

4. Fügen Sie dem Lizenzendpunkt mit dem folgenden Befehl kostenpflichtige Produkte hinzu.

```
aws deadline put-metered-product \  
--license-endpoint-id LICENSE_ENDPOINT_ID \  
--product-id PRODUCT_ID
```

Sie können ein Produkt mit dem `remove-metered-product` folgenden Befehl von einem Lizenzendpunkt entfernen:

```
aws deadline remove-metered-product \  
--license-endpoint-id LICENSE_ENDPOINT_ID \  
--productId PRODUCT_ID
```

Sie können einen Lizenzendpunkt mit dem `delete-license-endpoint` folgenden Befehl löschen:

```
aws deadline delete-license-endpoint \  
--license-endpoint-id LICENSE_ENDPOINT_ID
```

Schritt 3: Eine Rendering-Anwendung mit einem Endpunkt Connect

Nachdem der Lizenzendpunkt eingerichtet wurde, verwenden Anwendungen ihn genauso wie einen Lizenzserver eines Drittanbieters. In der Regel konfigurieren Sie den Lizenzserver für die Anwendung, indem Sie eine Umgebungsvariable oder eine andere Systemeinstellung, z. B. einen Microsoft Windows-Registrierungsschlüssel, auf einen Lizenzserverport und eine Adresse festlegen.

Verwenden Sie den folgenden AWS CLI Befehl, um den DNS-Namen des Lizenzendpunkts abzurufen.

```
aws deadline get-license-endpoint
```

Oder Sie können die Amazon VPC-Konsole (<https://console.aws.amazon.com/vpc/>) verwenden, um den VPC-Endpunkt zu identifizieren, der im vorherigen Schritt von der Deadline Cloud API erstellt wurde.

Beispiele für Konfigurationen

Example — Autodesk Maya und Arnold

Setzen Sie die Umgebungsvariable `ADSKFLEX_LICENSE_FILE` auf:

```
2702@VPC_Endpoint_DNS_Name:2701@VPC_Endpoint_DNS_Name
```

Note

Verwenden Sie für Windows Worker ein Semikolon (;) anstelle eines Doppelpunkts (:), um die Endpunkte voneinander zu trennen.

Example — Autodesk 3ds Max

Stellen Sie die Umgebungsvariable wie folgt `ADSKFLEX_LICENSE_FILE` ein:

```
2704@VPC_Endpoint_DNS_Name
```

Example — Foundry Nuke

Setzen Sie die Umgebungsvariable `foundry_LICENSE` auf `Um 6101@VPC_Endpoint_DNS_Name` zu testen, ob die Lizenzierung ordnungsgemäß funktioniert, können Sie Nuke in einem Terminal ausführen:

```
~/nuke/Nuke14.0v5/Nuke14.0 -x
```

Example — SideFX Houdini, Mantra und Karma

Führen Sie den folgenden Befehl aus:

```
/opt/hfs19.5.640/bin/hserver -S  
"http://VPC_Endpoint_DNS_Name:1715;http://VPC_Endpoint_DNS_Name:1716;http://  
VPC_Endpoint_DNS_Name:1717;"
```

Um zu testen, ob die Lizenzierung ordnungsgemäß funktioniert, können Sie eine Houdini-Szene mit diesem Befehl rendern:

```
/opt/hfs19.5.640/bin/hython ~/forpentest.hip -c "hou.node('/out/mantra1').render()"
```

Benutzer in Deadline Cloud verwalten

AWS Deadline Cloud verwendet AWS IAM Identity Center, um Benutzer und Gruppen zu verwalten. IAM Identity Center ist ein cloudbasierter Single-Sign-On-Service, der in Ihren Enterprise-Single-Sign-On-Anbieter (SSO) integriert werden kann. Mit der Integration können sich Benutzer mit ihrem Unternehmenskonto anmelden.

Deadline Cloud aktiviert IAM Identity Center standardmäßig und ist für die Einrichtung und Verwendung von Deadline Cloud erforderlich. Weitere Informationen finden Sie unter [Ihre Identitätsquelle verwalten](#).

Ein Organisationsinhaber für Sie AWS Organizations ist für die Verwaltung der Benutzer und Gruppen verantwortlich, die Zugriff auf Ihren Deadline Cloud-Monitor haben. Sie können diese Benutzer und Gruppen mithilfe von IAM Identity Center oder der Deadline Cloud-Konsole erstellen und verwalten. Weitere Informationen finden Sie unter [Was ist AWS Organizations](#).

Sie erstellen und entfernen Benutzer und Gruppen, die den Monitor verwenden können, um Farmen, Warteschlangen und Flotten mithilfe der Deadline Cloud-Konsole zu verwalten. Wenn Sie einen Benutzer zu Deadline Cloud hinzufügen, muss er sein Passwort mithilfe von IAM Identity Center zurücksetzen, bevor er Zugriff erhält.

Themen

- [Benutzer und Gruppen für den Monitor verwalten](#)
- [Verwalten Sie Benutzer und Gruppen für Farmen, Warteschlangen und Flotten](#)

Benutzer und Gruppen für den Monitor verwalten

Ein Organisationsinhaber kann die Deadline Cloud-Konsole verwenden, um die Benutzer und Gruppen zu verwalten, die Zugriff auf den Deadline Cloud-Monitor haben. Sie können aus vorhandenen IAM Identity Center-Benutzern und -Gruppen wählen oder neue Benutzer und Gruppen über die Konsole hinzufügen.

1. Melden Sie sich bei der Deadline [Cloud-Konsole](#) an AWS Management Console und öffnen Sie sie. Wählen Sie auf der Hauptseite im Bereich Erste Schritte die Option Deadline Cloud einrichten oder Gehe zum Dashboard.
2. Wählen Sie im linken Navigationsbereich Benutzerverwaltung aus. Standardmäßig ist die Registerkarte Gruppen ausgewählt.

Wählen Sie je nach der auszuführenden Aktion entweder die Registerkarte Gruppen oder die Registerkarte Benutzer.

Monitor groups

So erstellen Sie eine Gruppe

1. Wählen Sie Create group (Gruppe erstellen) aus.
2. Geben Sie einen Gruppennamen ein. Der Name muss für alle Gruppen in Ihrer IAM Identity Center-Organisation eindeutig sein.

Um eine Gruppe zu entfernen

1. Wählen Sie die zu entfernende Gruppe aus.
2. Wählen Sie Remove (Entfernen) aus.
3. Wählen Sie im Bestätigungsdialogfeld die Option Gruppe entfernen aus.

Note

Sie entfernen die Gruppe aus IAM Identity Center. Gruppenmitglieder können sich nicht mehr bei der Deadline Cloud anmelden oder auf Farmressourcen zugreifen.

Monitor users

So fügen Sie Benutzer hinzu

1. Wählen Sie die Registerkarte Users.
2. Wählen Sie Add Users (Benutzer hinzufügen).
3. Geben Sie den Namen, die E-Mail-Adresse und den Benutzernamen für den neuen Benutzer ein.
4. Wählen Sie bei Bedarf eine oder mehrere IAM Identity Center-Gruppen aus, zu denen der neue Benutzer hinzugefügt werden soll.
5. Wählen Sie Einladung senden, um dem neuen Benutzer eine E-Mail mit Anweisungen zum Beitritt zu Ihrer IAM Identity Center-Organisation zu senden.

So entfernen Sie einen Benutzer:

1. Wählen Sie den Benutzer aus, den Sie von Ihrem Monitor entfernen möchten.
2. Wählen Sie Remove (Entfernen) aus.
3. Wählen Sie im Bestätigungsdialogfeld die Option Benutzer entfernen.

 Note

Sie entfernen den Benutzer aus IAM Identity Center. Der Benutzer kann sich nicht mehr beim Deadline Cloud-Monitor anmelden oder auf Farmressourcen zugreifen.

Verwalten Sie Benutzer und Gruppen für Farmen, Warteschlangen und Flotten

1. [Falls Sie es noch nicht getan haben, melden Sie sich bei der Deadline Cloud-Konsole an AWS Management Console und öffnen Sie sie.](#)
2. Wählen Sie im linken Navigationsbereich Farmen und andere Ressourcen aus.
3. Wählen Sie die Farm aus, die Sie verwalten möchten. Wählen Sie den Farmnamen, um die Detailseite zu öffnen. Sie können mit der Suchleiste nach der Farm suchen.
4. Um eine Warteschlange oder Flotte zu verwalten, wählen Sie die Registerkarte Warteschlangen oder Flotten und dann die Warteschlange oder Flotte aus, die Sie verwalten möchten.
5. Wählen Sie die Registerkarte Zugriffsverwaltung. Standardmäßig ist die Registerkarte Gruppen ausgewählt. Um Benutzer zu verwalten, verschieben Sie den Schalter auf Benutzer.

Wählen Sie je nach der auszuführenden Aktion entweder die Registerkarte Gruppen oder die Registerkarte Benutzer.

Definitionen der Zugriffsebenen finden Sie unter [Berechtigungen](#).

Groups

Um Gruppen hinzuzufügen

1. Wählen Sie den Schalter Gruppen aus.
2. Wählen Sie Add Group (Gruppe hinzufügen) aus.

3. Wählen Sie aus der Dropdownliste die Gruppen aus, die Sie hinzufügen möchten.
4. Wählen Sie für die Gruppenzugriffsebene eine der folgenden Optionen aus:
 - Betrachter
 - Beitragender
 - Manager
 - Eigentümer
5. Wählen Sie Hinzufügen aus.

Um Gruppen zu entfernen

1. Wählen Sie die zu entfernenden Gruppen aus.
2. Wählen Sie Remove (Entfernen) aus.
3. Wählen Sie im Bestätigungsdiaologfeld Remove aus.

Users

So fügen Sie Benutzer hinzu

1. Um einen Benutzer hinzuzufügen, wählen Sie Benutzer hinzufügen.
2. Wählen Sie aus der Dropdownliste die Benutzer aus, die Sie Ihrer Farm hinzufügen möchten.
3. Wählen Sie für die Benutzerzugriffsebene eine der folgenden Optionen aus:
 - Betrachter
 - Beitragender
 - Manager
 - Eigentümer
4. Wählen Sie Hinzufügen aus. Die Benutzer werden zu Ihrer Farm hinzugefügt.

Um Benutzer zu entfernen

1. Wählen Sie den Benutzer aus, den Sie entfernen möchten.
2. Wählen Sie im Bestätigungsdiaologfeld Entfernen die Option Entfernen aus. Der Benutzer wird dann aus der ausgewählten Farm entfernt.

Sie können Farmberechtigungen für Benutzer und Gruppen auch mithilfe der IAM Identity Center-Konsole unter <https://console.aws.amazon.com/singlesignon/> hinzufügen oder entfernen.

Frist für Cloud-Jobs

Ein Job ist eine Reihe von Anweisungen, die AWS Deadline Cloud verwendet, um Arbeiten an verfügbaren Mitarbeitern zu planen und auszuführen. Wenn Sie einen Job erstellen, wählen Sie die Farm und die Warteschlange aus, an die der Job gesendet werden soll. Sie stellen auch eine JSON- oder YAML-Datei bereit, die Anweisungen für die Verarbeitung durch die Mitarbeiter enthält. Deadline Cloud akzeptiert Jobvorlagen, die der Open Job Description (OpenJD) -Spezifikation zur Beschreibung von Jobs folgen. Weitere Informationen finden Sie in der [Dokumentation zur offenen Stellenbeschreibung](#) auf der GitHub Website.

Ein Job besteht aus:

- Schritte — Definiert das Skript, das auf Workern ausgeführt werden soll. Für Schritte können Anforderungen wie Mindestarbeitspeicher oder andere Schritte gelten, die zuerst abgeschlossen werden müssen. Jeder Schritt umfasst eine oder mehrere Aufgaben.
- Aufgaben — Eine Arbeitseinheit, die an einen Mitarbeiter zur Ausführung geschickt wird. Eine Aufgabe ist eine Kombination aus dem Skript eines Schritts und Parametern, wie z. B. der Frame-Nummer, die im Skript verwendet werden. Der Job ist abgeschlossen, wenn alle Aufgaben für alle Schritte abgeschlossen sind.
- Umgebungen — Richten Sie Anweisungen ein und entfernen Sie sie, wenn mehrere Schritte oder Aufgaben gemeinsam ausgeführt werden.

Sie können einen Job auf eine der folgenden Arten erstellen:

- Verwenden Sie einen Deadline Cloud-Einreicher.
- Erstellen Sie ein Job-Bundle und verwenden Sie die [Deadline Cloud-Befehlszeilenschnittstelle](#) (Deadline Cloud CLI).
- Verwenden Sie das AWS SDK.
- Verwenden Sie die AWS Command Line Interface (AWS CLI).

Ein Submitter ist ein Plugin für Ihre DCC-Software (Digital Content Creation), das die Erstellung eines Jobs in der Schnittstelle zu Ihrer DCC-Software verwaltet. Nachdem Sie den Job erstellt haben, verwenden Sie den Absender, um ihn zur Bearbeitung an Deadline Cloud zu senden. Hinter den Kulissen erstellt der Einreicher eine OpenJD-Jobvorlage, die den Job beschreibt. Gleichzeitig werden Ihre Asset-Dateien in einen Amazon Simple Storage Service (Amazon S3) -Bucket hochgeladen. Um

den Zeitaufwand für das Senden von Dateien zu reduzieren, werden nur Dateien, die sich seit dem letzten Hochladen von Dateien geändert haben, an Amazon S3 gesendet.

Um Ihre eigenen Skripts und Pipelines zum Senden von Jobs an Deadline Cloud zu erstellen, können Sie die Deadline Cloud-CLI, das AWS SDK oder die AWS CLI To-Call-Operationen verwenden, um Jobs zu erstellen, abzurufen, anzuzeigen und aufzulisten. In den folgenden Themen wird erklärt, wie Sie die Deadline Cloud CLI verwenden.

Die Deadline Cloud-CLI wird zusammen mit dem Deadline Cloud-Submitter installiert. Weitere Informationen finden Sie unter [Richten Sie Deadline Cloud-Einreicher ein](#).

Themen

- [Jobs mit der Deadline Cloud CLI einreichen](#)
- [Jobs in Deadline Cloud planen](#)
- [Job in der Deadline Cloud CLI](#)
- [Jobs in Deadline Cloud ändern](#)
- [Wie Deadline Cloud Jobs verarbeitet](#)
- [Fehlerbehebung bei Deadline Cloud-Jobs](#)

Jobs mit der Deadline Cloud CLI einreichen

Um einen Job über die Deadline Cloud-Befehlszeilenschnittstelle (Deadline Cloud CLI) einzureichen, verwenden Sie den `deadline bundle submit` Befehl.

Jobs werden in Warteschlangen eingereicht. Wenn Sie noch keine Farm und Warteschlange eingerichtet haben, verwenden Sie die Deadline Cloud-Konsole (<https://console.aws.amazon.com/https://console.aws.amazon.com/deadlinecloud/home>), um eine Farm und eine Warteschlange einzurichten und die Farm- und Warteschlangen-ID zu sehen. Weitere Informationen finden Sie unter [Farmdetails definieren](#) und [Warteschlangendetails definieren](#).

Verwenden Sie den folgenden Befehl, um die Standardfarm und die Warteschlange für die Deadline Cloud-CLI festzulegen. Wenn Sie die Standardeinstellungen festlegen, können Sie Deadline Cloud CLI-Befehle verwenden, ohne eine Farm oder Warteschlange anzugeben. Ersetzen Sie im folgenden Beispiel *farmId* und *queueId* durch Ihre eigenen Informationen:

```
deadline config set defaults.farm_id farmId
```

```
deadline config set defaults.queue_id queueId
```

Um die Schritte und Aufgaben in einem Job zu spezifizieren, erstellen Sie eine OpenJD-Jobvorlage. Weitere Informationen finden Sie unter [Template Schemas \[Version: 2023-09\]](#) im Open Job Description Specification Repository. GitHub

Das folgende Beispiel ist eine YAML-Jobvorlage. Es definiert einen Job mit zwei Schritten und fünf Aufgaben pro Schritt.

```
name: Sample Job
specificationVersion: jobtemplate-2023-09
steps:
- name: Sample Step 1
  parameterSpace:
    taskParameterDefinitions:
      - name: var
        range: 1-5
        type: INT
  script:
    actions:
      onRun:
        args:
          - '1'
        command: /usr/bin/sleep
- name: Sample Step 2
  parameterSpace:
    taskParameterDefinitions:
      - name: var
        range: 1-5
        type: INT
  script:
    actions:
      onRun:
        args:
          - '1'
        command: /usr/bin/sleep
```

Um einen Job zu erstellen, erstellen Sie einen neuen Ordner mit dem Namen `sample_job` und speichern Sie dann die Vorlagendatei im neuen Ordner unter `template.yaml`. Sie reichen den Job mit dem folgenden Deadline Cloud CLI-Befehl ein:

```
deadline bundle submit path/to/sample_job
```

Die Antwort des Befehls enthält eine Kennung für den Job. Merken Sie sich die ID, damit Sie den Status des Jobs später überprüfen können.

```
Submitting to Queue: test-queue
Waiting for Job to be created...
Submitted job bundle:
  sample_job
Job creation completed successfully
jobId
```

Es gibt zusätzliche Optionen, die Sie beim Einreichen eines Jobs verwenden können. Weitere Informationen finden Sie unter [Weitere Optionen zum Einreichen von Jobs mit der Deadline Cloud CLI](#).

Weitere Optionen zum Einreichen von Jobs mit der Deadline Cloud CLI

Der CLI-Befehl `deadline bundle submit` Deadline Cloud bietet Optionen, mit denen Sie zusätzliche Informationen für einen Job angeben können. In den nachstehenden Beispielen wird Folgendes veranschaulicht:

- Geben Sie die Parameter an, die bei der Verarbeitung der Jobvorlage verwendet werden.
- Hängen Sie Dateien und Ordner in einer gemeinsam genutzten Umgebung an einen Job an.
- Legen Sie die maximale Anzahl von Aufgabenfehlern fest, bevor ein Job abgebrochen wird.
- Legen Sie die maximale Anzahl von Wiederholungen für eine Aufgabe fest.

Auftragsparameter

Die `parameters` Option legt den Wert eines Job-Parameters fest, wenn Sie den Job erstellen. Die Jobvorlage definiert das Feld, und die `parameters` Option legt den Wert fest. Ein Parameter kann einen Standardwert haben. Wenn für den Parameter ein Wert angegeben wird, überschreibt der angegebene Wert den Standardwert.

Die folgende Jobvorlage definiert das `TestParameter` Feld:

```
name: Sample Job With Job Parameter
parameterDefinitions:
- default: test
  name: TestParameter
```

```
type: STRING
specificationVersion: jobtemplate-2023-09
steps:
- description: step description
  name: MyStep
  parameterSpace:
    taskParameterDefinitions:
    - name: var
      range: 1-5
      type: INT
  script:
    actions:
    onRun:
      args:
      - '1'
      command: /usr/bin/sleep
```

Der folgende Befehl setzt den Wert von TestParameter auf „Hello AWS“:

```
deadline bundle submit sample_job --parameter "TestParameter=Hello AWS"
```

Speicherprofile

Speicherprofile helfen beim Austausch von Dateien zwischen Mitarbeitern mit unterschiedlichen Betriebssystemen. Erstellen Sie mit der Deadline Cloud-Konsole ein Speicherprofil. Verwenden Sie dann den `storage-profile-id` Parameter, um das Speicherprofil zu verwenden. Weitere Informationen finden Sie unter [Gemeinsamer Speicher in Deadline Cloud](#).

Um das Speicherprofil für Jobübermittlungen mithilfe der Deadline Cloud-CLI festzulegen, verwenden Sie den folgenden Befehl, um den `storage-profile-id` Konfigurationsparameter festzulegen:

```
deadline config set settings.storage_profile_id storageProfileId
```

Maximale Anzahl fehlgeschlagener Aufgaben

Die `max-failed-tasks-count` Option legt die maximale Anzahl von Aufgaben fest, die fehlschlagen können, bevor der gesamte Job fehlschlägt und alle verbleibenden Aufgaben markiert werden `CANCELED`. Der Standardwert lautet 100.

```
deadline bundle submit sample_job --max-failed-tasks-count 10
```

Maximale Anzahl fehlgeschlagener Aufgabenwiederholungen

Die `max-retries-per-task` Option legt fest, wie oft eine Aufgabe maximal wiederholt wird, bevor sie fehlschlägt. Wenn eine Aufgabe erneut versucht wird, wird sie in den READY Status versetzt. Der Standardwert ist 5.

```
deadline bundle submit sample_job --max-retries-per-task 10
```

Jobs in Deadline Cloud planen

Nachdem ein Auftrag erstellt wurde, plant AWS Deadline Cloud, dass er in einer oder mehreren Flotten bearbeitet wird, die einer Warteschlange zugeordnet sind. Die Flotte, die eine bestimmte Aufgabe bearbeitet, wird auf der Grundlage der für die Flotte konfigurierten Funktionen und der Hostanforderungen eines bestimmten Schritts ausgewählt.

Jobs werden in der Reihenfolge der bestmöglichen Priorität geplant, von der höchsten zur niedrigsten Priorität. Wenn zwei Jobs dieselbe Priorität haben, wird der älteste Job zuerst geplant.

In den folgenden Abschnitten wird detailliert beschrieben, wie ein Job geplant wird.

Prüfen Sie die Flottenkompatibilität

Nachdem ein Job erstellt wurde, vergleicht Deadline Cloud die Hostanforderungen für jeden Schritt im Job mit den Fähigkeiten der Flotten, die mit der Warteschlange verknüpft sind, an die der Job übermittelt wurde. Wenn eine Flotte die Hostanforderungen erfüllt, wird der Job in den READY Status versetzt.

Wenn für einen Schritt des Jobs Anforderungen gelten, die von einer Flotte, die der Warteschlange zugeordnet ist, nicht erfüllt werden können, wird der Status des Schritts auf `gesetztNOT_COMPATIBLE`. Außerdem werden die restlichen Schritte des Jobs storniert.

Die Funktionen für eine Flotte werden auf Flottenebene festgelegt. Selbst wenn ein Mitarbeiter in einer Flotte die Anforderungen des Auftrags erfüllt, werden ihm keine Aufgaben aus dem Auftrag zugewiesen, wenn seine Flotte die Anforderungen des Auftrags nicht erfüllt.

Die folgende Jobvorlage enthält einen Schritt, der die Hostanforderungen für den Schritt spezifiziert:

```
name: Sample Job With Host Requirements
```

```
specificationVersion: jobtemplate-2023-09
steps:
- name: Step 1
  script:
    actions:
      onRun:
        args:
          - '1'
        command: /usr/bin/sleep
    hostRequirements:
      amounts:
        # Capabilities starting with "amount." are amount capabilities. If they start with
        "amount.worker.",
        # they are defined by the OpenJD specification. Other names are free for custom
        usage.
        - name: amount.worker.vcpu
          min: 4
          max: 8
      attributes:
        - name: attr.worker.os.family
          anyOf:
            - linux
```

Dieser Job kann für eine Flotte mit den folgenden Funktionen geplant werden:

```
{
  "vCpuCount": {"min": 4, "max": 8},
  "memoryMiB": {"min": 1024},
  "osFamily": "linux",
  "cpuArchitectureType": "x86_64"
}
```

Dieser Job kann nicht für eine Flotte mit einer der folgenden Funktionen geplant werden:

```
{
  "vCpuCount": {"min": 4},
  "memoryMiB": {"min": 1024},
  "osFamily": "linux",
  "cpuArchitectureType": "x86_64"
}
The vCpuCount has no maximum, so it exceeds the maximum vCPU host requirement.
{
```

```
"vCpuCount": {"max": 8},
"memoryMiB": {"min": 1024},
"osFamily": "linux",
"cpuArchitectureType": "x86_64"
}
```

The vCpuCount has no minimum, so it doesn't satisfy the minimum vCPU host requirement.

```
{
  "vCpuCount": {"min": 4, "max": 8},
  "memoryMiB": {"min": 1024},
  "osFamily": "windows",
  "cpuArchitectureType": "x86_64"
}
```

The osFamily doesn't match.

Skalierung der Flotte

Wenn ein Auftrag einer kompatiblen Service-Managed-Flotte zugewiesen wird, wird die Flotte auto skaliert. Die Anzahl der Mitarbeiter in der Flotte schwankt je nach der Anzahl der Aufgaben, die der Flotte zur Ausführung zur Verfügung stehen.

Wenn ein Auftrag einer vom Kunden verwalteten Flotte zugewiesen wird, sind Mitarbeiter möglicherweise bereits vorhanden oder können mithilfe von ereignisbasierter Autoskalierung erstellt werden. Weitere Informationen finden Sie unter [Verwenden EventBridge zur Behandlung von Auto Scaling-Ereignissen](#) im Amazon EC2 Auto Scaling-Benutzerhandbuch.

Sitzungen

Die Aufgaben in einem Job sind in eine oder mehrere Sitzungen aufgeteilt. Die Mitarbeiter führen die Sitzungen durch, um die Umgebung einzurichten, die Aufgaben auszuführen und dann die Umgebung zu zerstören. Jede Sitzung besteht aus einer oder mehreren Aktionen, die ein Mitarbeiter ausführen muss.

Wenn ein Mitarbeiter Abschnittsaktionen abschließt, können zusätzliche Sitzungsaktionen an den Mitarbeiter gesendet werden. Der Mitarbeiter verwendet in der Sitzung vorhandene Umgebungen und Jobanhänge wieder, um Aufgaben effizienter zu erledigen.

Jobanhänge werden von dem Einreicher, den Sie verwenden, als Teil Ihres Deadline Cloud CLI-Jobpakets erstellt. Sie können auch Jobanhänge erstellen, indem Sie die `--attachments`

Option für den `create-job` AWS CLI Befehl verwenden. Umgebungen werden an zwei Stellen definiert: Warteschlangenumgebungen, die an eine bestimmte Warteschlange angehängt sind, und Arbeitsschrittmgebungen, die in der Jobvorlage definiert sind.

Es gibt vier Arten von Sitzungsaktionen:

- `syncInputJobAttachments`— Lädt die Eingabe-Job-Anhänge an den Worker herunter.
- `envEnter`— Führt die `onEnter` Aktionen für eine Umgebung aus.
- `taskRun`— Führt die `onRun` Aktionen für eine Aufgabe aus.
- `envExit`— Führt die `onExit` Aktionen für eine Umgebung aus.

Die folgende Jobvorlage hat eine Schrittmgebung. Sie enthält eine `onEnter` Definition zum Einrichten der Schrittmgebung, eine `onRun` Definition, die die auszuführende Aufgabe definiert, und eine `onExit` Definition zum Abbau der Schrittmgebung. Die für diesen Job erstellten Sitzungen umfassen eine `envEnter` Aktion, eine oder mehrere `taskRun` Aktionen und dann eine `envExit` Aktion.

```
name: Sample Job with Maya Environment
specificationVersion: jobtemplate-2023-09
steps:
- name: Maya Step
  stepEnvironments:
  - name: Maya
    description: Runs Maya in the background.
    script:
      embeddedFiles:
      - name: initData
        filename: init-data.yaml
        type: TEXT
        data: |
          scene_file: MyAwesomeSceneFile
          renderer: arnold
          camera: persp
    actions:
      onEnter:
        command: MayaAdaptor
        args:
        - daemon
        - start
        - --init-data
```

```
    - file://{{Env.File.initData}}
  onExit:
    command: MayaAdaptor
    args:
      - daemon
      - stop
  parameterSpace:
    taskParameterDefinitions:
      - name: Frame
        range: 1-5
        type: INT
  script:
    embeddedFiles:
      - name: runData
        filename: run-data.yaml
        type: TEXT
        data: |
          frame: {{Task.Param.Frame}}
  actions:
    onRun:
      command: MayaAdaptor
      args:
        - daemon
        - run
        - --run-data
        - file://{{ Task.File.runData }}
```

Abhängigkeiten der einzelnen Schritte

Deadline Cloud unterstützt die Definition von Abhängigkeiten zwischen Schritten, sodass ein Schritt wartet, bis ein anderer Schritt abgeschlossen ist, bevor er gestartet wird. Sie können mehr als eine Abhängigkeit für einen Schritt definieren. Ein Schritt mit einer Abhängigkeit wird erst geplant, wenn alle Abhängigkeiten abgeschlossen sind.

Wenn die Jobvorlage eine zirkuläre Abhängigkeit definiert, wird der Job abgelehnt und der Jobstatus wird auf gesetztCREATE_FAILED.

Mit der folgenden Jobvorlage wird ein Job in zwei Schritten erstellt. StepBhängt davon abStepA. StepBwird erst ausgeführt, nachdem der StepA Vorgang erfolgreich abgeschlossen wurde.

Nachdem der Job erstellt wurde, StepA befindet er sich im READY Status und StepB befindet sich im PENDING Status. Wenn der StepA Vorgang abgeschlossen ist, StepB wechselt er in den READY

Status. StepA schlägt fehl oder wurde der StepA Vorgang abgebrochen, StepB wechselt er in den CANCELED Status.

Sie können eine Abhängigkeit von mehreren Schritten festlegen. Wenn beispielsweise von beiden StepA und StepC abhängt StepB, StepC wird erst gestartet, wenn die anderen beiden Schritte abgeschlossen sind.

```
name: Step-Step Dependency Test
specificationVersion: 'jobtemplate-2023-09'
steps:
- name: A
  script:
    actions:
      onRun:
        command: bash
        args: ['{{ Task.File.run }}']
    embeddedFiles:
      - name: run
        type: TEXT
        data: |
          #!/bin/env bash

          set -euo pipefail

          sleep 1
          echo Task A Done!
- name: B
  dependencies:
    - dependsOn: A # This means Step B depends on Step A
  script:
    actions:
      onRun:
        command: bash
        args: ['{{ Task.File.run }}']
    embeddedFiles:
      - name: run
        type: TEXT
        data: |
          #!/bin/env bash

          set -euo pipefail

          sleep 1
```

```
echo Task B Done!
```

Job in der Deadline Cloud CLI

In diesem Thema wird beschrieben, wie Sie die AWS Deadline Cloud-Befehlszeilenschnittstelle (Deadline Cloud CLI) verwenden, um den Status eines Jobs oder Schritts anzuzeigen. Wenn Sie den Deadline Cloud-Monitor verwenden möchten, um den Status von Jobs oder Schritten einzusehen, finden Sie weitere Informationen unter [Jobs, Schritte und Aufgaben in Deadline Cloud anzeigen und verwalten](#).

Sie können den Status eines Jobs mit dem Befehl `deadline job get --job-id` Deadline Cloud CLI anzeigen. Die Antwort auf die Befehle umfasst den Status des Jobs oder Schritts und die Anzahl der Aufgaben in jedem Verarbeitungsstatus.

Wenn Sie einen Job zum ersten Mal einreichen, lautet der Status `CREATE_IN_PROGRESS`. Wenn der Job die Validierungsprüfungen besteht, ändert sich sein Status in `CREATE_COMPLETE`. Wenn nicht, ändert sich der Status zu `CREATE_FAILED`.

Zu den möglichen Gründen, warum ein Job die Validierungsprüfungen nicht bestehen kann, gehören die folgenden:

- Die Jobvorlage entspricht nicht der OpenJD-Spezifikation.
- Der Job enthält zu viele Schritte.
- Der Job enthält insgesamt zu viele Aufgaben.

Um die Kontingente für die maximale Anzahl von Schritten und Aufgaben in einem Job zu sehen, verwenden Sie die Service-Kontingents-Konsole. Weitere Informationen finden Sie unter [Kontingente für Deadline Cloud](#).

Möglicherweise liegt auch ein interner Dienstfehler vor, der die Erstellung eines Auftrags verhindert. In diesem Fall lautet der Statuscode des Jobs `INTERNAL_ERROR` und das Feld mit der Statusmeldung enthält eine detailliertere Erklärung.

Verwenden Sie den folgenden Deadline Cloud CLI-Befehl, um die Details für einen Job anzuzeigen. Ersetzen Sie es im folgenden Beispiel `jobID` durch Ihre eigenen Informationen:

```
deadline job get --job-id jobId
```

Die Antwort des `deadline job get` Befehls lautet wie folgt:

```
jobId: jobId
name: Sample Job
lifecycleStatus: CREATE_COMPLETE
lifecycleStatusMessage: Job creation completed successfully
priority: 50
createdAt: 2024-03-26 18:11:19.065000+00:00
createdBy: Test User
startedAt: 2024-03-26 18:12:50.710000+00:00
taskRunStatus: STARTING
taskRunStatusCounts:
  PENDING: 0
  READY: 5
  RUNNING: 0
  ASSIGNED: 0
  STARTING: 0
  SCHEDULED: 0
  INTERRUPTING: 0
  SUSPENDED: 0
  CANCELED: 0
  FAILED: 0
  SUCCEEDED: 0
  NOT_COMPATIBLE: 0
maxFailedTasksCount: 100
maxRetriesPerTask: 5
```

Jede Aufgabe in einem Job oder Schritt hat einen Status. Die Aufgabenstatus werden kombiniert, um einen Gesamtstatus für Jobs und Schritte zu ergeben. Die Anzahl der Aufgaben in den einzelnen Bundesstaaten wird im `taskRunStatusCounts` Antwortfeld angegeben.

Der Status eines Jobs oder Schritts hängt vom Status seiner Aufgaben ab. Der Status wird durch die Aufgaben bestimmt, die diesen Status der Reihe nach haben. Der Status der Schritte wird genauso bestimmt wie der Auftragsstatus.

In der folgenden Liste werden die Status beschrieben:

NOT_COMPATIBLE

Der Auftrag ist nicht mit der Farm kompatibel, da es keine Flotten gibt, die eine der Aufgaben des Jobs ausführen können.

RUNNING

Ein oder mehrere Mitarbeiter führen Aufgaben aus dem Job aus. Solange mindestens eine laufende Aufgabe vorhanden ist, ist der Job markiert **RUNNING**.

ASSIGNED

Einem oder mehreren Arbeitskräften werden als nächste Aktion Aufgaben im Job zugewiesen. Die Umgebung, falls vorhanden, ist eingerichtet.

STARTING

Ein oder mehrere Mitarbeiter richten die Umgebung für die Ausführung von Aufgaben ein.

SCHEDULED

Aufgaben für den Job werden für einen oder mehrere Mitarbeiter als nächste Aktion des Arbeiters geplant.

READY

Mindestens eine Aufgabe für den Job ist zur Bearbeitung bereit.

INTERRUPTING

Mindestens eine Aufgabe im Job wird unterbrochen. Unterbrechungen können auftreten, wenn Sie den Status des Jobs manuell aktualisieren. Dies kann auch als Reaktion auf eine Unterbrechung aufgrund von Spot-Preisänderungen von Amazon Elastic Compute Cloud (Amazon EC2) geschehen.

FAILED

Eine oder mehrere Aufgaben im Job wurden nicht erfolgreich abgeschlossen.

CANCELED

Eine oder mehrere Aufgaben des Jobs wurden storniert.

SUSPENDED

Mindestens eine Aufgabe im Job wurde ausgesetzt.

PENDING

Eine Aufgabe im Job wartet auf die Verfügbarkeit einer anderen Ressource.

SUCCEDED

Alle Aufgaben im Job wurden erfolgreich verarbeitet.

Jobs in Deadline Cloud ändern

Sie können die folgenden update Befehle AWS Command Line Interface (AWS CLI) verwenden, um die Konfiguration eines Jobs zu ändern oder den Zielstatus eines Jobs, Schritts oder einer Aufgabe festzulegen:

- `aws deadline update-job`
- `aws deadline update-step`
- `aws deadline update-task`

Ersetzen Sie in den folgenden update Befehlsbeispielen jeden Befehl *user input placeholder* durch Ihre eigenen Informationen.

Sie können den Deadline Cloud-Monitor auch verwenden, um die Konfiguration eines Jobs zu ändern. Weitere Informationen finden Sie unter [Jobs, Schritte und Aufgaben in Deadline Cloud anzeigen und verwalten](#).

Example — Einen Job erneut in die Warteschlange stellen

Alle Aufgaben im Job wechseln in den READY Status, sofern es keine Schrittabhängigkeiten gibt. Schritte mit Abhängigkeiten wechseln zu entweder READY oder PENDING, wenn sie wiederhergestellt werden.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status PENDING
```

Example — Stornieren Sie einen Job

Alle Aufgaben im Job, die nicht den Status haben SUCCEEDED oder markiert FAILED sind CANCELED.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status CANCELED
```

Example — Einen Job als fehlgeschlagen markieren

Alle Aufgaben im Job, die den Status haben, SUCCEEDED bleiben unverändert. Alle anderen Aufgaben sind markiert FAILED.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status FAILED
```

Example — Markiere einen Job als erfolgreich

Alle Aufgaben im Job werden in den SUCCEEDED Bundesstaat übertragen.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status SUCCEEDED
```

Example — Einen Job aussetzen

Die Aufgaben des Jobs im FAILED Status SUCCEEDEDCANCELED, oder ändern sich nicht. Alle anderen Aufgaben sind markiert SUSPENDED.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status SUSPENDED
```

Example — Ändern Sie die Priorität eines Jobs

Aktualisiert die Priorität eines Jobs, um die Reihenfolge zu ändern, in der er geplant ist. Jobs mit höherer Priorität werden in der Regel zuerst geplant.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--priority 100
```

Example — Ändert die Anzahl der zulässigen fehlgeschlagenen Aufgaben

Aktualisiert die maximale Anzahl fehlgeschlagener Aufgaben, die der Job haben kann, bevor die verbleibenden Aufgaben abgebrochen werden.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--max-failed-tasks-count 200
```

Example — Ändert die Anzahl der zulässigen Aufgabenwiederholungen

Aktualisiert die maximale Anzahl von Wiederholungen für eine Aufgabe, bevor die Aufgabe fehlschlägt. Eine Aufgabe, die die maximale Anzahl von Wiederholungen erreicht hat, kann erst in die Warteschlange gestellt werden, wenn dieser Wert erhöht wird.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--max-retries-per-task 10
```

Example — Archivieren Sie einen Job

Aktualisiert den Lebenszyklusstatus des Jobs aufARCHIVED. Archivierte Jobs können nicht geplant oder geändert werden. Sie können nur einen Job archivieren, der sich im SUSPENDED Status FAILED,CANCELED,SUCCEEDED, oder befindet.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--lifecycle-status ARCHIVED
```

Example — Einen Schritt erneut in die Warteschlange stellen

Alle Aufgaben im Schritt wechseln in den READY Status, sofern keine Schrittabhängigkeiten bestehen. Aufgaben in Schritten mit Abhängigkeiten wechseln entweder zu READY oderPENDING, und die Aufgabe wird wiederhergestellt.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status PENDING
```

Example — Einen Schritt abbrechen

Alle Aufgaben in dem Schritt, die nicht den Status haben SUCCEEDED oder markiert FAILED sind CANCELED.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status CANCELED
```

Example — Einen Schritt als fehlgeschlagen markieren

Alle Aufgaben in dem Schritt, die den Status haben, SUCCEEDED bleiben unverändert. Alle anderen Aufgaben sind markiert FAILED.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status FAILED
```

Example — Markiere einen Schritt als erfolgreich

Alle Aufgaben in dem Schritt sind markiert SUCCEEDED.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status SUCCEEDED
```

Example — Einen Schritt aussetzen

Aufgaben im Schritt im FAILED Status SUCCEEDDCANCELED, oder ändern sich nicht. Alle anderen Aufgaben sind markiertSUSPENDED.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status SUSPENDED
```

Example — Den Status einer Aufgabe ändern

Wenn Sie den Befehl `update-task` Deadline Cloud CLI verwenden, wechselt die Aufgabe in den angegebenen Status.

```
aws deadline update-task \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--task-id taskID \  
--target-task-run-status SUCCEEDED | SUSPENDED | CANCELED | FAILED | PENDING
```

Wie Deadline Cloud Jobs verarbeitet

Um einen Job zu bearbeiten, verwendet AWS Deadline Cloud die Jobvorlage Open Job Description (OpenJD), um die benötigten Ressourcen zu ermitteln. Deadline Cloud wählt aus den Flotten, die zu Ihrer Warteschlange gehören, einen geeigneten Mitarbeiter für einen Schritt aus. Der ausgewählte Mitarbeiter erfüllt alle für den Schritt erforderlichen Fähigkeitsattribute.

Als Nächstes sendet Deadline Cloud Anweisungen an die Mitarbeiter, um eine Sitzung für den Schritt einzurichten. Die für den Schritt erforderliche Software muss auf der Worker-Instanz verfügbar sein, damit der Job ausgeführt werden kann. Der Service kann Sitzungen für mehrere Worker öffnen, sofern die Skalierungseinstellungen für die Flotte ausreichend sind.

Sie können die Software in einem Amazon Machine Image (AMI) einrichten, oder Ihr Mitarbeiter kann die Software zur Laufzeit aus einem Repository oder Paketmanager laden. Sie können

Warteschlangen-, Job- oder Schrittumgebungen verwenden, um die von Ihnen bevorzugte Software bereitzustellen.

Der Deadline Cloud-Dienst verwendet die OpenJD-Vorlage, um die für den Job erforderlichen Schritte und die für jeden Schritt erforderlichen Aufgaben zu ermitteln. Einige Schritte hängen von anderen Schritten ab, sodass Deadline Cloud die Reihenfolge festlegt, in der die Schritte abgeschlossen werden. Anschließend sendet Deadline Cloud die Aufgaben für jeden Schritt zur Bearbeitung an die Mitarbeiter. Wenn eine Aufgabe abgeschlossen ist, sendet der Service eine weitere Aufgabe in derselben Sitzung, oder der Mitarbeiter kann eine neue Sitzung starten.

Sie können den Fortschritt des Jobs im Deadline Cloud-Monitor, in der Deadline Cloud-Befehlszeilenschnittstelle (Deadline Cloud CLI) oder im verfolgten AWS CLI. Weitere Informationen zur Verwendung des Monitors finden Sie unter [Den Deadline Cloud-Monitor verwenden](#). Weitere Informationen zur Verwendung der Deadline Cloud-CLI finden Sie unter [Job in der Deadline Cloud CLI](#).

Nachdem alle Aufgaben in jedem Schritt abgeschlossen sind, ist der Job abgeschlossen und die Ausgabe kann auf Ihre Workstation heruntergeladen werden. Auch wenn der Job nicht abgeschlossen wurde, steht die Ausgabe aller abgeschlossenen Schritte und Aufgaben zum Herunterladen zur Verfügung.

Deadline Cloud entfernt Jobs 120 Tage nach ihrer Einreichung. Wenn ein Job entfernt wird, werden auch alle mit dem Job verknüpften Schritte und Aufgaben entfernt. Wenn Sie den Job erneut ausführen müssen, reichen Sie die OpenJD-Vorlage für den Job erneut ein.

Fehlerbehebung bei Deadline Cloud-Jobs

Informationen zu häufigen Problemen mit Jobs in AWS Deadline Cloud finden Sie in den folgenden Themen.

Themen

- [Warum ist die Erstellung meines Jobs fehlgeschlagen?](#)
- [Warum ist mein Job nicht kompatibel?](#)
- [Warum ist mein Job immer noch fertig?](#)
- [Warum ist mein Job gescheitert?](#)
- [Warum steht mein Schritt noch aus?](#)

Warum ist die Erstellung meines Jobs fehlgeschlagen?

Zu den möglichen Gründen, warum ein Job die Gültigkeitsprüfungen nicht bestehen kann, gehören die folgenden:

- Die Jobvorlage entspricht nicht der OpenJD-Spezifikation.
- Der Job enthält zu viele Schritte.
- Der Job enthält insgesamt zu viele Aufgaben.
- Es ist ein interner Dienstfehler aufgetreten, der die Erstellung des Jobs verhindert hat.

Um die Kontingente für die maximale Anzahl von Schritten und Aufgaben in einem Job zu sehen, verwenden Sie die Service-Kontingents-Konsole. Weitere Informationen finden Sie unter [Kontingente für Deadline Cloud](#).

Warum ist mein Job nicht kompatibel?

Zu den häufigsten Gründen, warum Jobs nicht mit Warteschlangen kompatibel sind, gehören die folgenden:

- Der Warteschlange, an die der Job übermittelt wurde, sind keine Flotten zugeordnet. Öffnen Sie den Deadline Cloud-Monitor und überprüfen Sie, ob der Warteschlange Flotten zugeordnet sind. Weitere Informationen zum Anzeigen von Warteschlangen finden Sie unter [Warteschlangen- und Flottendetails in Deadline Cloud anzeigen](#)
- Für den Job gelten Hostanforderungen, die von keiner der Flotten erfüllt werden, die der Warteschlange zugeordnet sind. Vergleichen Sie zur Überprüfung den `hostRequirements` Eintrag in der Auftragsvorlage mit der Konfiguration der Flotten in Ihrer Farm. Stellen Sie sicher, dass eine der Flotten die Hostanforderungen erfüllt. Weitere Informationen zur Flottenkompatibilität finden Sie unter [Prüfen Sie die Flottenkompatibilität](#) Informationen zur Flottenkonfiguration finden Sie unter [Warteschlangen- und Flottendetails in Deadline Cloud anzeigen](#).

Warum ist mein Job immer noch fertig?

Zu den möglichen Gründen, warum Ihr Job im READY Bundesstaat festgefahren zu sein scheint, gehören die folgenden:

- Die maximale Anzahl von Mitarbeitern für Flotten, die der Warteschlange zugeordnet sind, ist auf Null gesetzt. Informationen zur Überprüfung finden Sie unter [Warteschlangen- und Flottendetails in Deadline Cloud anzeigen](#).
- In der Warteschlange befindet sich ein Job mit höherer Priorität. Informationen zur Überprüfung finden Sie unter [Warteschlangen- und Flottendetails in Deadline Cloud anzeigen](#).
- Überprüfen Sie für vom Kunden verwaltete Flotten die Auto Scaling-Konfiguration. Weitere Informationen finden Sie unter [Skalieren Sie Ihre Amazon EC2 EC2-Flotte automatisch mit der Deadline Cloud-Funktion für Skalierungsempfehlungen](#).

Warum ist mein Job gescheitert?

Ein Job kann aus vielen Gründen scheitern. Um nach dem Problem zu suchen, öffnen Sie den Deadline Cloud-Monitor und wählen Sie den fehlgeschlagenen Job aus. Wählen Sie eine Aufgabe aus, die fehlgeschlagen ist, und sehen Sie sich dann die Protokolle für die Aufgabe an. Anweisungen finden Sie unter [Logs in Deadline Cloud anzeigen](#).

- Wenn Sie Lizenzfehler sehen oder ein Wasserzeichen angezeigt wird, das angezeigt wird, weil die Software nicht über eine gültige Lizenz verfügt, stellen Sie sicher, dass der Worker eine Verbindung zum erforderlichen Lizenzserver herstellen kann. Weitere Informationen finden Sie unter [Vom Kunden verwaltete Flotten mit einem Lizenzendpunkt Connect](#).

Warum steht mein Schritt noch aus?

Schritte können im PENDING Status verbleiben, wenn eine oder mehrere ihrer Abhängigkeiten nicht abgeschlossen sind. Sie können den Status der Abhängigkeiten mithilfe des Deadline Cloud-Monitors überprüfen. Anweisungen finden Sie unter [Einen Schritt in Deadline Cloud anzeigen](#).

Dateispeicher für Deadline Cloud

Mitarbeiter müssen Zugriff auf die Speicherorte haben, die die für die Bearbeitung eines Auftrags erforderlichen Eingabedateien enthalten, sowie auf die Speicherorte, an denen die Ausgabe gespeichert wird. AWS Deadline Cloud bietet zwei Optionen für Speicherorte:

- Mit Job-Anhängen überträgt Deadline Cloud die Eingabe- und Ausgabedateien für Ihre Jobs zwischen einer Workstation und Deadline Cloud-Mitarbeitern hin und her. Um die Dateiübertragungen zu ermöglichen, verwendet Deadline Cloud einen Amazon Simple Storage Service (Amazon S3) -Bucket in Ihrem AWS-Konto.

Wenn Sie Auftragsanhänge mit einer vom Service verwalteten Flotte verwenden, können Sie ein virtuelles Dateisystem (VFS) in Ihrem virtuellen privaten Netzwerk (VPN) einrichten. Dann können Mitarbeiter Dateien nur bei Bedarf laden.

- Bei gemeinsam genutztem Speicher nutzen Sie die Dateifreigabe mit Ihrem Betriebssystem, um Zugriff auf Dateien zu gewähren.

Wenn Sie plattformübergreifenden gemeinsamen Speicher verwenden, können Sie ein Speicherprofil erstellen, sodass Mitarbeiter den Pfad Dateien zwischen zwei verschiedenen Betriebssystemen zuordnen können.

Themen

- [Jobanhänge in Deadline Cloud](#)
- [Gemeinsamer Speicher in Deadline Cloud](#)

Jobanhänge in Deadline Cloud

Job Job-Anhängen können Sie Dateien zwischen Ihrer Workstation und AWS Deadline Cloud hin und her übertragen. Mit Job-Anhängen müssen Sie keinen Amazon S3 S3-Bucket für Ihre Dateien manuell einrichten. Stattdessen wählen Sie beim Erstellen einer Warteschlange mit der Deadline Cloud-Konsole den Bucket für Ihre Jobanhänge aus.

Wenn Sie zum ersten Mal einen Job bei Deadline Cloud einreichen, werden alle Dateien für den Job in Deadline Cloud übertragen. Bei nachfolgenden Einreichungen werden nur die Dateien übertragen, die sich geändert haben, was sowohl Zeit als auch Bandbreite spart.

Nach Abschluss der Verarbeitung können Sie das Ergebnis von der Jobdetailseite oder mithilfe des Deadline Cloud `deadline job download-output` CLI-Befehls herunterladen.

Sie können denselben S3-Bucket für mehrere Warteschlangen verwenden. Legen Sie für jede Warteschlange ein anderes Root-Präfix fest, um die Anhänge im Bucket zu organisieren.

Wenn Sie mit der Konsole eine Warteschlange erstellen, können Sie entweder eine bestehende AWS Identity and Access Management (IAM-) Rolle auswählen oder die Konsole eine neue Rolle erstellen lassen. Wenn die Konsole die Rolle erstellt, legt sie Berechtigungen für den Zugriff auf den Bucket fest, der für die Warteschlange angegeben ist. Wenn Sie eine bestehende Rolle auswählen, müssen Sie der Rolle Berechtigungen für den Zugriff auf den S3-Bucket gewähren.

Verschlüsselung für S3-Buckets mit Stellenanhängen

Job-Anhangsdateien werden standardmäßig automatisch in Ihrem S3-Bucket verschlüsselt. Dieser Ansatz trägt dazu bei, Ihre Informationen vor unbefugtem Zugriff zu schützen. Sie müssen nichts tun, um Ihre Dateien mit Schlüsseln zu verschlüsseln, die von Deadline Cloud bereitgestellt werden. Weitere Informationen finden Sie unter [Amazon S3 verschlüsselt jetzt automatisch alle neuen Objekte](#) im Amazon S3 S3-Benutzerhandbuch.

Sie können Ihren eigenen, vom Kunden verwalteten AWS Key Management Service Schlüssel verwenden, um den S3-Bucket zu verschlüsseln, der Ihre Jobanhänge enthält. Dazu müssen Sie die IAM-Rolle für die Warteschlange ändern, die dem Bucket zugeordnet ist, um den Zugriff auf den zu ermöglichen. AWS KMS key

Um den IAM-Richtlinieneditor für die Warteschlangenrolle zu öffnen

1. Melden Sie sich bei der Deadline [Cloud-Konsole](#) an AWS Management Console und öffnen Sie sie. Wählen Sie auf der Hauptseite im Abschnitt Erste Schritte die Option Farmen anzeigen aus.
2. Wählen Sie aus der Liste der Farmen die Farm aus, die die zu ändernde Warteschlange enthält.
3. Wählen Sie aus der Liste der Warteschlangen die Warteschlange aus, die Sie ändern möchten.
4. Wählen Sie im Abschnitt Warteschlangendetails die Servicerolle aus, um die IAM-Konsole für die Servicerolle zu öffnen.

Führen Sie als Nächstes das folgende Verfahren aus.

Um die Rollenrichtlinie mit der Erlaubnis zu aktualisieren AWS KMS

1. Wählen Sie aus der Liste der Berechtigungsrichtlinien die Richtlinie für die Rolle aus.

2. Wählen Sie im Abschnitt „In dieser Richtlinie definierte Berechtigungen“ die Option Bearbeiten aus.
3. Wählen Sie Neue Aussage hinzufügen aus.
4. Kopieren Sie die folgende Richtlinie und fügen Sie sie in den Editor ein. Ändern Sie die *RegionaccountID*, und *keyID* in Ihre eigenen Werte.

```
{
  "Effect": "Allow",
  "Action": [
    "kms:Decrypt",
    "kms:DescribeKey",
    "kms:GenerateDataKey"
  ],
  "Resource": [
    "arn:aws:kms:Region:accountID:key/keyID"
  ]
}
```

5. Wählen Sie Weiter aus.
6. Überprüfen Sie die Änderungen an der Richtlinie, und wenn Sie damit zufrieden sind, wählen Sie Änderungen speichern aus.

Verwaltung von Job-Anhängen in S3-Buckets

Deadline Cloud speichert die für Ihren Job erforderlichen Dateien mit Anhängen von Jobs in einem S3-Bucket. Diese Dateien sammeln sich im Laufe der Zeit an, was zu erhöhten Amazon S3 S3-Kosten führt. Um die Kosten zu senken, können Sie eine S3-Lifecycle-Konfiguration auf Ihren S3-Bucket anwenden. Diese Konfiguration kann Dateien im Bucket automatisch löschen. Da sich der S3-Bucket in Ihrem Konto befindet, können Sie die S3-Lifecycle-Konfiguration jederzeit ändern oder entfernen. Weitere Informationen finden Sie unter [Beispiele für die Konfiguration von S3 Lifecycle](#) im Amazon S3 S3-Benutzerhandbuch.

Für eine detailliertere S3-Bucket-Verwaltungslösung können Sie festlegen, dass Ihre AWS-Konto Objekte in einem S3-Bucket auf der Grundlage des letzten Zugriffs ablaufen. Weitere Informationen finden Sie im AWS Architektur-Blog unter [Ablaufen von Amazon S3 S3-Objekten basierend auf dem Datum des letzten Zugriffs zur Kostensenkung](#).

Virtuelles Dateisystem Deadline Cloud

Die Unterstützung virtueller Dateisysteme für Jobanhänge in AWS Deadline Cloud ermöglicht es der Client-Software auf Mitarbeitern, direkt mit Amazon Simple Storage Service zu kommunizieren. Mitarbeiter können Dateien nur bei Bedarf laden, anstatt alle Dateien vor der Verarbeitung herunterzuladen. Dateien werden lokal gespeichert. Durch diesen Ansatz wird vermieden, dass Ressourcen heruntergeladen werden, die mehrmals als einmal verwendet wurden. Alle Dateien werden nach Abschluss des Jobs entfernt.

- Das virtuelle Dateisystem bietet eine erhebliche Leistungssteigerung für bestimmte Jobprofile. Im Allgemeinen bieten kleinere Teilmengen der gesamten Dateien mit einer größeren Mitarbeiterflotte den größten Nutzen. Eine geringe Anzahl von Dateien mit weniger Mitarbeitern hat ungefähr die gleiche Bearbeitungszeit.
- Die Unterstützung virtueller Dateisysteme ist nur für Linux Mitarbeiter in serviceverwalteten Flotten verfügbar.
- Das virtuelle Dateisystem von Deadline Cloud unterstützt die folgenden Operationen, ist jedoch nicht POSIX-konform:
 - `Datei create, delete, open, close, read, write, append, truncate, rename, move, copy, statfsync`, und `falloc`
 - `Verzeichnis create, delete, rename, move, copy`, und `stat`
- Das virtuelle Dateisystem wurde entwickelt, um die Datenübertragung zu reduzieren und die Leistung zu verbessern, wenn Ihre Aufgaben nur auf einen Teil eines großen Datensatzes zugreifen. Es ist nicht für alle Workloads optimiert. Sie sollten Ihre Arbeitslast testen, bevor Sie Produktionsjobs ausführen.

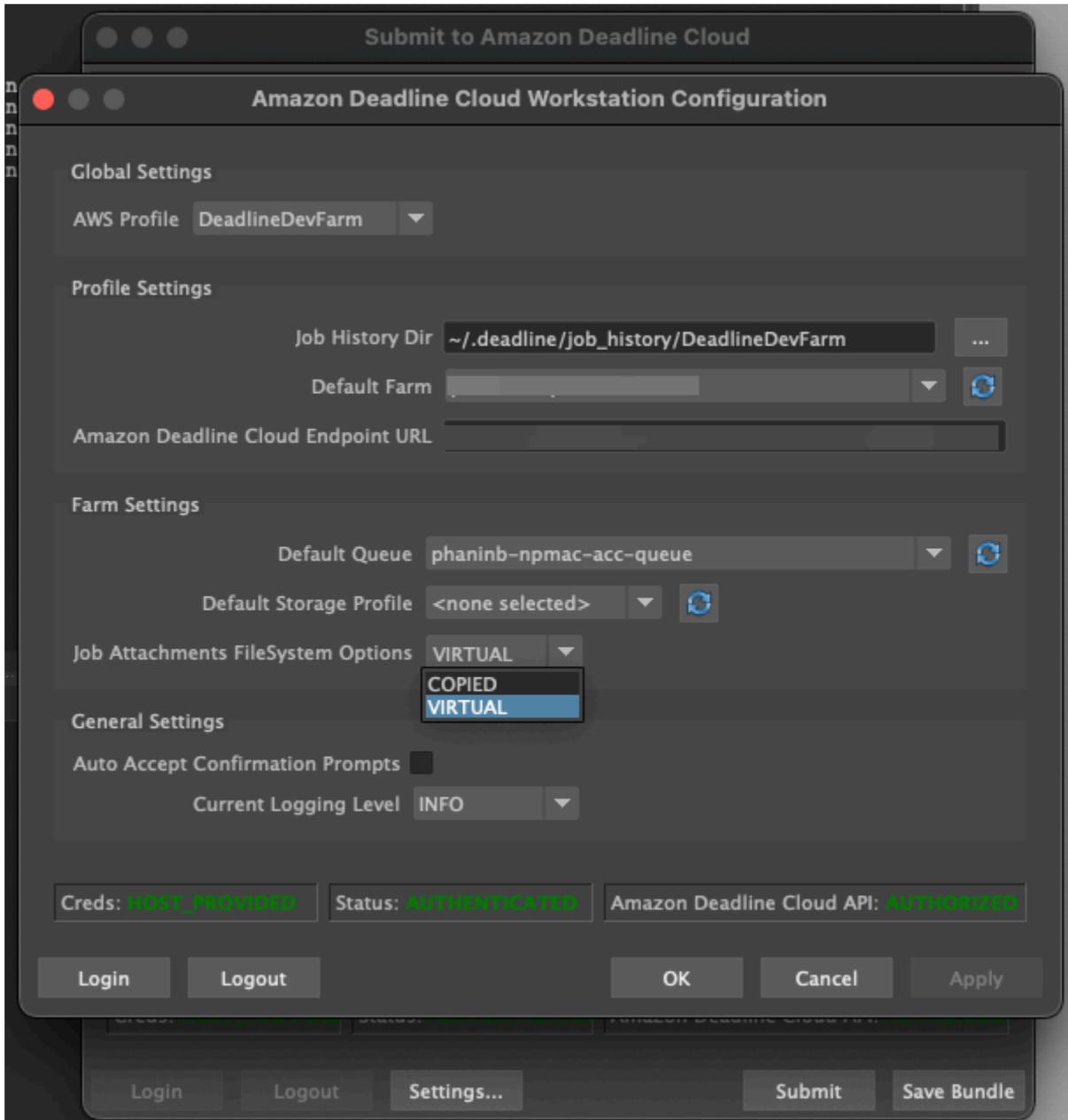
Aktivieren Sie die VFS-Unterstützung

Die Unterstützung virtueller Dateisysteme (VFS) ist für jeden Job aktiviert. In den folgenden Fällen greift ein Job auf das Standard-Framework für Jobanhänge zurück:

- Ein Worker-Instanzprofil unterstützt kein virtuelles Dateisystem.
- Probleme verhindern das Starten des virtuellen Dateisystemprozesses.
- Das virtuelle Dateisystem kann nicht gemountet werden.

Um die Unterstützung virtueller Dateisysteme mithilfe des Absenders zu aktivieren

1. Wenn Sie einen Job einreichen, wählen Sie die Schaltfläche Einstellungen, um das Konfigurationsfenster der AWS Deadline Cloud-Workstation zu öffnen.
2. Wählen Sie in der Dropdownliste Dateisystemoptionen für Jobanhänge die Option VIRTUELL aus.



3. Um Ihre Änderungen zu speichern, wählen Sie OK.

Um die Unterstützung virtueller Dateisysteme zu aktivieren, verwenden Sie AWS CLI

- Verwenden Sie den folgenden Befehl, wenn Sie einen gespeicherten Job einreichen:

```
deadline bundle submit-job --job-attachments-file-system VIRTUAL
```

Um zu überprüfen, ob das virtuelle Dateisystem für einen bestimmten Job erfolgreich gestartet wurde, überprüfen Sie Ihre Protokolle in Amazon CloudWatch Logs. Suchen Sie nach den folgenden Meldungen:

```
Using mount_point mount_point  
Launching vfs with command command  
Launched vfs as pid PID number
```

Wenn das Protokoll die folgende Meldung enthält, ist die Unterstützung für virtuelle Dateisysteme deaktiviert:

```
Virtual File System not found, falling back to COPIED for JobAttachmentsFileSystem.
```

Problembehandlung bei der Unterstützung virtueller Dateisysteme

Mit dem Deadline Cloud-Monitor können Sie Protokolle für Ihr virtuelles Dateisystem anzeigen. Anweisungen finden Sie unter [Logs in Deadline Cloud anzeigen](#).

Virtuelle Dateisystemprotokolle werden auch an die Gruppe CloudWatch Logs gesendet, die der Warteschlange zugeordnet ist, die gemeinsam mit der Ausgabe des Worker-Agents genutzt wird.

Gemeinsamer Speicher in Deadline Cloud

Um gemeinsam genutzten Speicher zu nutzen, nutzen Mitarbeiter das Filesharing-System des Betriebssystems für den Zugriff auf einen gemeinsam genutzten Speicherplatz für die Eingabe und Ausgabe Ihrer Jobs.

Die tatsächliche Methode, mit der Sie Dateien gemeinsam nutzen, hängt von Ihrem Betriebssystem und der Art und Weise ab, wie Sie gemeinsam genutzten Speicher in Ihrem Netzwerk implementieren. Sie sind dafür verantwortlich, wie Sie die Dateifreigabe konfigurieren und sicherstellen, dass sie Ihren Anforderungen entspricht.

Wenn Sie eine systemübergreifende Filesharing-Lösung verwenden, können Sie Speicherprofile verwenden, um Dateispeicherorte zwischen Dateisystemen Linux und Windows Dateisystemen zuzuordnen.

Speicherprofile in Deadline Cloud

Ein Speicherprofil ermöglicht es Ihnen, Farmen mit plattformübergreifendem gemeinsam genutztem Speicher einzurichten. Ein Speicherprofil ordnet betriebssystemübergreifende Pfade für Jobs zu, die auf Workern mit einem anderen Betriebssystem als der Workstation bearbeitet wurden, von der aus sie übermittelt wurden.

Speicherprofile sind erforderlich, wenn Sie eine vom Kunden verwaltete Flotte mit einer Mischung aus Betriebssystemen für Arbeitsstationen und Mitarbeiter verwenden. Speicherprofile werden auf vom Service verwalteten Flotten nicht unterstützt.

Nachdem Sie ein Speicherprofil erstellt haben, müssen Sie Zugriff auf die Warteschlangen und Flotten gewähren, die das Profil verwenden.

Um ein Speicherprofil zu erstellen

1. Öffnen Sie die [Deadline Cloud-Konsole](#).
2. Wählen Sie unter Erste Schritte die Option Gehe zum Deadline Cloud-Dashboard.
3. Wählen Sie eine Farm und dann den Tab Speicherprofile aus.
4. Wählen Sie Speicherprofil erstellen aus.
5. Wählen Sie ein Betriebssystem aus der Drop-down-Liste aus.
6. Geben Sie einen Namen für das Profil ein. Ein klarer Name hilft Ihnen bei der Auswahl des Speicherprofils, das Sie beim Senden von Jobs verwenden möchten.
7. Geben Sie als Pfadnamen den Stammspeicherort der Jobdaten auf der Workstation ein, von der aus Sie Jobs einreichen.
8. Wählen Sie einen Speichertyp:
 - Lokal bezieht sich auf Dateispeicherorte, die nicht vom Worker und der Workstation gemeinsam genutzt werden. Sie werden als Jobanhänge hochgeladen.
 - Shared bezieht sich auf Speicher, der von der Workstation und der Workstation gemeinsam genutzt wird. Dateien im gemeinsam genutzten Speicher werden nicht als Jobanhänge hochgeladen.

9. Geben Sie einen Pfad zum Speicherort des Dateisystems an. Dies ist das Stammverzeichnis für Ihre Jobdaten.
10. Wählen Sie Erstellen.

Nachdem Sie ein Speicherprofil erstellt haben, müssen Sie Ihre Warteschlangen und vom Kunden verwalteten Flotten ändern, um das neue Profil verwenden zu können. Gehen Sie nach Abschluss des vorherigen Verfahrens wie folgt vor, um Zugriff auf ein Speicherprofil zu gewähren.

Damit Warteschlangen und vom Kunden verwaltete Flotten ein Speicherprofil verwenden können

1. Wählen Sie entweder die Registerkarte Warteschlangen oder Flotten.
2. Wählen Sie die Warteschlange oder Flotte aus, die Sie ändern möchten.
3. Wählen Sie Speicherprofile ändern aus.
4. Wählen Sie das Speicherprofil, das Sie zulassen möchten, und die Dateisystemspeicherorte aus diesem Profil aus.
5. Wählen Sie Save Changes.

Verwaltung von Budgets und Nutzung für Deadline Cloud

Der Budgetmanager und der Usage Explorer von AWS Deadline Cloud sind Tools für das Kostenmanagement, die anhand verfügbarer Informationen zu Kostenvariablen die ungefähren Kosten für die Nutzung von Deadline Cloud ermitteln. Die Kostenmanagement-Tools garantieren nicht den Betrag, der Ihnen für Ihre tatsächliche Nutzung von Deadline Cloud und anderen AWS Diensten geschuldet wird.

Um Ihnen bei der Verwaltung der Kosten für Deadline Cloud zu helfen, können Sie die folgenden Funktionen verwenden:

- **Budgetmanager** — Mit dem Budgetmanager von Deadline Cloud können Sie Budgets erstellen und bearbeiten, um die Projektkosten zu verwalten.
- **Nutzungsexplorer** — Mit dem Deadline Cloud-Nutzungsexplorer können Sie sehen, wie viele AWS Ressourcen verwendet werden und wie hoch die geschätzten Kosten für diese Ressourcen sind.

Annahmen zu den Kosten

Die grundlegende Berechnung, die von den Kostenmanagement-Tools von Deadline Cloud verwendet wird, lautet:

```
Cost per job =  
  (CMF run time x CMF compute rate) +  
  (SMF run time x SMF compute rate) +  
  (License run time x license rate)
```

- Die Laufzeit ist die Summe aller Aufgaben in einem Job, von der Startzeit bis zur Endzeit.
- Die Rechenrate richtet sich nach den [AWS Deadline Cloud-Preisen](#) für serviceverwaltete Flotten. Für vom Kunden verwaltete Flotten wird der Rechenpreis auf 1 USD pro Arbeitsstunde geschätzt.
- Der Lizenztarif wird durch den Basislizenzpreis von Deadline Cloud bestimmt. Zusätzliche Stufen sind nicht enthalten. Weitere Informationen zu den Lizenzpreisen finden Sie unter [AWS Deadline Cloud-Preise](#).

Der Kostenvoranschlag der Kostenmanagement-Tools von Deadline Cloud kann aus verschiedenen Gründen von Ihren tatsächlichen Kosten abweichen. Zu den häufigsten Gründen gehören:

- Kundeneigene Ressourcen und deren Preisgestaltung. Sie können wählen, ob Sie Ihre eigenen Ressourcen mitbringen möchten, entweder von AWS oder extern von lokalen oder anderen Cloud-Anbietern. Die tatsächlichen Kosten dieser Ressourcen werden nicht berechnet.
- Kosten ungenutzter Arbeitskräfte. Bei Flotten mit einer Mindestanzahl von Instanzen über Null werden untätige Mitarbeiter in Berechnungen nicht berücksichtigt.
- Werbegutschriften, Rabatte und individuelle Preisvereinbarungen. Die Kostenmanagement-Tools berücksichtigen keine Werbegutschriften, private Preisvereinbarungen oder andere Rabatte. Möglicherweise haben Sie Anspruch auf andere Rabatte, die nicht Teil des Kostenvoranschlags sind.
- Aufbewahrung von Vermögenswerten. Die Speicherung von Ressourcen ist in den Kosten- und Nutzungsschätzungen nicht enthalten.
- Preisänderungen. AWS bietet pay-as-you-go Preise für die meisten Dienste an. Die Preise können sich im Laufe der Zeit ändern. Die Kostenmanagement-Tools verwenden die meisten öffentlich verfügbaren up-to-date Preise, aber es kann nach Änderungen zu Verzögerungen kommen.
- Steuern. Die Kostenmanagement-Tools beinhalten keine Steuern, die auf unseren Kauf der Dienstleistung erhoben werden.
- Rundung. Das Kostenmanagement-Tool führt eine mathematische Rundung von Preisdaten durch.
- Währung. Kostenschätzungen werden in US-Dollar vorgenommen. Die globalen Wechselkurse variieren im Laufe der Zeit. Wenn Sie Schätzungen anhand des aktuellen Wechselkurses in eine andere Währungsbasis umrechnen, wirken sich Wechselkursänderungen auf die Schätzung aus.
- Externe Lizenzierung. Wenn Sie sich dafür entscheiden, vorab gekaufte Lizenzen zu verwenden (bringen Sie Ihre eigene Lizenz mit), können die Kostenmanagement-Tools von Deadline Cloud diese Kosten nicht berücksichtigen.

Den Deadline Cloud Budget Manager verwenden

Der Deadline Cloud-Budgetmanager hilft Ihnen dabei, die Ausgaben für eine bestimmte Ressource zu kontrollieren, z. B. für eine Warteschlange, Flotte oder Farm. Sie können Budgetbeträge und -limits erstellen und automatisierte Aktionen festlegen, um zusätzliche Ausgaben im Rahmen des Budgets zu reduzieren oder zu verhindern.

In den folgenden Abschnitten finden Sie die Schritte zur Verwendung des Deadline Cloud-Budgetmanagers.

Themen

- [Voraussetzung](#)
- [Rufen Sie den Budgetmanager auf](#)
- [Budget erstellen](#)
- [Ein Budget anzeigen](#)
- [Bearbeiten Sie ein Budget](#)
- [Deaktivieren Sie ein Budget](#)

Voraussetzung

Um den Deadline Cloud-Budgetmanager verwenden zu können, benötigen Sie eine OWNER Zugriffsebene. Um die OWNER Genehmigung zu erteilen, folgen Sie den Schritten unter [Benutzer in Deadline Cloud verwalten](#).

Rufen Sie den Budgetmanager auf

Gehen Sie wie folgt vor, um auf den Deadline Cloud-Budgetmanager zuzugreifen.

1. Melden Sie sich bei der Deadline [Cloud-Konsole](#) an AWS Management Console und öffnen Sie sie.
2. Wählen Sie Farmen anzeigen.
3. Suchen Sie die Farm, über die Sie Informationen erhalten möchten, und wählen Sie dann Jobs verwalten aus. Der Deadline Cloud-Monitor wird auf einer neuen Registerkarte geöffnet.
4. Wählen Sie im Deadline Cloud-Monitor im linken Navigationsbereich Budgets aus.

Auf der Übersichtsseite des Budget-Managers wird eine Liste der aktiven und inaktiven Budgets angezeigt:

- Aktive Budgets werden anhand der ausgewählten Ressource (einer Warteschlange) erfasst.
- Inaktive Budgets sind entweder abgelaufen oder wurden von einem Benutzer storniert und die Kosten werden nicht mehr im Rahmen der Budgetgrenzen erfasst.

Nachdem Sie ein Budget ausgewählt haben, enthält die Seite mit der Budgetübersicht grundlegende Informationen zum Budget. Zu den bereitgestellten Informationen gehören der Budgetname, der Status, die Ressourcen, der verbleibende Prozentsatz, der verbleibende Betrag, das Gesamtbudget, das Startdatum und das Enddatum.

Budget erstellen

Gehen Sie wie folgt vor, um ein Budget zu erstellen.

1. Falls Sie dies noch nicht getan haben, melden Sie sich bei der an AWS Management Console, öffnen Sie die Deadline [Cloud-Konsole](#), wählen Sie eine Farm aus und wählen Sie dann Jobs verwalten aus.
2. Wählen Sie auf der Budget-Manager-Seite die Option Budget erstellen aus.
3. Geben Sie im Detailbereich eine Budgetbezeichnung für das Budget ein.
4. (Optional) Geben Sie im Beschreibungsfeld eine klare, kurze Beschreibung für das Budget ein.
5. Wählen Sie unter Ressource die Dropdownliste Warteschlange aus, um die Warteschlange zu suchen und auszuwählen, für die Sie ein Budget erstellen möchten.
6. Legen Sie unter Zeitraum das Start- und Enddatum für das Budget fest, indem Sie die folgenden Schritte ausführen:

- a. Geben Sie als Startdatum das erste Datum der Budgetverfolgung im Format YYYY/MM/DD ein, oder wählen Sie das Kalendersymbol und wählen Sie ein Datum aus.

Das Standard-Startdatum ist das Datum, an dem das Budget erstellt wird.

- b. Geben Sie als Enddatum das letzte Datum der Budgetverfolgung im Format YYYY/MM/DD ein oder wählen Sie das Kalendersymbol und wählen Sie ein Datum aus.

Das Standard-Enddatum liegt 120 Tage vom Startdatum entfernt.

7. Geben Sie unter Budgetbetrag den Dollarbetrag des Budgets ein.
8. (Optional) Wir empfehlen Ihnen, Limitwarnungen zu erstellen. Im Abschnitt Limitaktionen können Sie automatisierte Aktionen implementieren, die ausgelöst werden, wenn bestimmte Beträge im Budget verbleiben. Führen Sie dazu die folgenden Schritte aus:
 - a. Wählen Sie Neue Aktion hinzufügen aus.
 - b. Geben Sie unter Verbleibender Betrag den Dollarbetrag ein, mit dem Sie die Aktion starten möchten.
 - c. Wählen Sie in der Dropdownliste Aktion die gewünschte Aktion aus. Zu den Aktionen gehören:
 - Nach Abschluss der aktuellen Arbeit beenden — Alle Arbeiten, die derzeit ausgeführt werden, wenn der Schwellenwert erreicht ist, laufen weiter (und verursachen Kosten), bis sie abgeschlossen sind.

- Arbeit sofort beenden — Alle Arbeiten werden sofort abgebrochen, wenn der Schwellenwert erreicht ist.
- d. Um weitere Limit-Benachrichtigungen zu erstellen, wählen Sie Neue Aktion hinzufügen und wiederholen Sie die beiden vorherigen Schritte.
9. Wählen Sie Budget erstellen aus. Die Seite „Budgetmanager“ wird angezeigt. Das neu erstellte Budget wird auf der Registerkarte Aktive Budgets angezeigt.

Ein Budget anzeigen

Nachdem Sie ein Budget erstellt haben, können Sie es auf der Seite Budgetmanager einsehen. Von dort aus können Sie den Gesamtbetrag des Budgets und die dem jeweiligen Budget zugewiesenen Gesamtkosten einsehen.

Gehen Sie wie folgt vor, um ein Budget einzusehen.

1. Falls Sie dies noch nicht getan haben, melden Sie sich bei der an AWS Management Console, öffnen Sie die Deadline [Cloud-Konsole](#), wählen Sie eine Farm aus und klicken Sie dann auf Jobs verwalten.
2. Wählen Sie im linken Navigationsbereich Budgets aus. Die Seite Budget Manager wird angezeigt.
3. Um ein aktives Budget anzuzeigen, wählen Sie die Registerkarte Aktive Budgets und dann den Namen des Budgets, das Sie anzeigen möchten. Die Seite mit den Budgetdetails wird angezeigt.
4. Um die Budgetdetails für ein abgelaufenes Budget anzuzeigen, wählen Sie den Tab Inaktive Budgets. Wählen Sie dann den Namen des Budgets aus, das Sie anzeigen möchten. Die Seite mit den Budgetdetails wird angezeigt.

Bearbeiten Sie ein Budget

Sie können jedes aktive Budget bearbeiten. Gehen Sie wie folgt vor, um ein aktives Budget zu bearbeiten.

1. Falls Sie dies noch nicht getan haben, melden Sie sich bei der an AWS Management Console, öffnen Sie die Deadline [Cloud-Konsole](#), wählen Sie eine Farm aus und wählen Sie dann Jobs verwalten aus.
2. Wählen Sie auf der Seite Budget Manager auf der Registerkarte Aktive Budgets die Schaltfläche neben dem Budget aus, das Sie bearbeiten möchten.

3. Wählen Sie im Dropdownmenü Aktionen in der oberen rechten Ecke die Option Budget bearbeiten aus.
4. Nehmen Sie die gewünschten Änderungen vor und wählen Sie dann Budget aktualisieren.

Deaktivieren Sie ein Budget

Sie können jedes aktive Budget deaktivieren. Wenn Sie ein Budget deaktivieren, ändert sich sein Status von Aktiv in Inaktiv. Wenn ein Budget deaktiviert wird, entspricht es einer Ressource nicht mehr dem Betrag dieses Budgets.

Gehen Sie wie folgt vor, um ein Budget zu deaktivieren.

1. Falls Sie dies noch nicht getan haben, melden Sie sich bei der an AWS Management Console, öffnen Sie die Deadline [Cloud-Konsole](#), wählen Sie eine Farm aus und wählen Sie dann Jobs verwalten aus.
2. Wählen Sie auf der Budget-Manager-Seite auf der Registerkarte Aktive Budgets die Schaltfläche neben dem Budget aus, das Sie deaktivieren möchten.
3. Wählen Sie im Dropdownmenü Aktionen in der oberen rechten Ecke die Option Budget deaktivieren aus. In wenigen Augenblicken wird das ausgewählte Budget von Aktiv zu Inaktiv geändert und von der Registerkarte Aktive Budgets auf die Registerkarte Inaktive Budgets verschoben.

Verwenden des Deadline Cloud-Nutzungsexplorers

Mit dem Deadline Cloud-Nutzungsexplorer können Sie Echtzeit-Metriken zu den Aktivitäten auf jeder Farm einsehen. Sie können sich die Kosten der Farm anhand verschiedener Variablen wie Warteschlange, Auftrag, Lizenzprodukt oder Instanztyp ansehen. Wählen Sie verschiedene Zeitrahmen aus, um sich die Nutzung in einem bestimmten Zeitraum und die Nutzungstrends im Laufe der Zeit anzusehen. Sie können sich auch eine detaillierte Aufschlüsselung der ausgewählten Datenpunkte ansehen, sodass Sie sich die Kennzahlen genauer ansehen können. Die Nutzung kann nach Zeit (Minuten und Stunden) oder nach Kosten (USD) angezeigt werden.

In den folgenden Abschnitten werden die Schritte für den Zugriff auf und die Verwendung des Deadline Cloud-Nutzungsexplorers beschrieben.

Themen

- [Voraussetzung](#)

- [Öffnen Sie den Usage Explorer](#)
- [Verwenden Sie den Usage Explorer](#)

Voraussetzung

Um den Deadline Cloud-Nutzungsexplorer verwenden zu können, benötigen Sie MANAGER entweder OWNER Farmberechtigungen. Weitere Informationen finden Sie unter [Verwalten Sie Benutzer und Gruppen für Farmen, Warteschlangen und Flotten](#).

Öffnen Sie den Usage Explorer

Gehen Sie wie folgt vor, um den Deadline Cloud-Nutzungsexplorer zu öffnen.

1. Melden Sie sich bei der Deadline [Cloud-Konsole](#) an AWS Management Console und öffnen Sie sie.
2. Um alle verfügbaren Farmen zu sehen, wählen Sie Farmen anzeigen.
3. Suchen Sie die Farm, über die Sie Informationen abrufen möchten, und wählen Sie dann Jobs verwalten aus. Der Deadline Cloud-Monitor wird auf einer neuen Registerkarte geöffnet.
4. Wählen Sie im Deadline Cloud-Monitor im linken Menü die Option Nutzungsexplorer aus.

Verwenden Sie den Usage Explorer

Auf der Seite des Usage Explorers können Sie bestimmte Parameter auswählen, in denen die Daten angezeigt werden können. Standardmäßig wird die Gesamtnutzung in Zeit (Stunden und Minuten) innerhalb der letzten 7 Tage angezeigt. Sie können diese Parameter ändern, und die angezeigten Informationen ändern sich dynamisch entsprechend den Parametereinstellungen.

Sie können die Ergebnisse nach Warteschlange, Auftrag, Computernutzung, Instanztyp oder Lizenzprodukt gruppieren. Wenn Sie sich für ein Lizenzprodukt entscheiden, werden die Kosten für bestimmte Lizenzen berechnet. Für alle anderen Gruppen wird die Zeit berechnet, indem die für die Ausführung der einzelnen Aufgaben benötigte Zeit addiert wird.

Der Usage Explorer gibt auf der Grundlage der von Ihnen festgelegten Filterkriterien nur 100 Ergebnisse zurück. Die Ergebnisse werden in absteigender Reihenfolge nach dem Erstellungsdatum und dem Zeitstempel aufgelistet. Wenn es mehr als 100 Ergebnisse gibt, erhalten Sie eine Fehlermeldung. Sie können Ihre Abfrage verfeinern, um die Anzahl der Ergebnisse zu reduzieren:

- Wählen Sie einen kleineren Zeitraum
- Wählen Sie weniger Warteschlangen aus
- Wählen Sie eine andere Gruppierung, z. B. eine Gruppierung nach Warteschlange statt nach Job

Themen

- [Verwenden Sie visuelle Grafiken, um Daten zu überprüfen](#)
- [Sehen Sie sich eine Aufschlüsselung der Messwerte an](#)
- [Sehen Sie sich die ungefähre Laufzeit der Warteschlangen an](#)

Verwenden Sie visuelle Grafiken, um Daten zu überprüfen

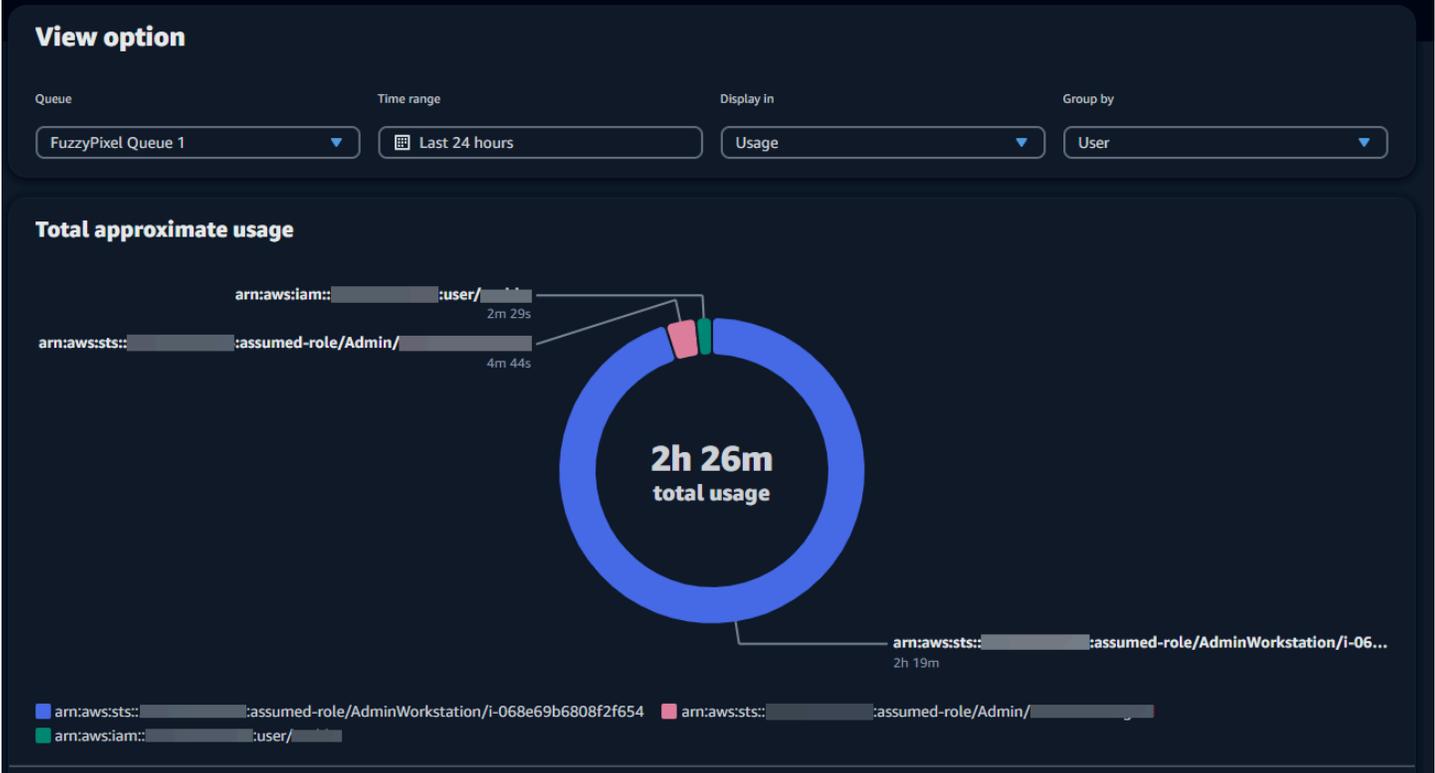
Sie können Daten in einem visuellen Format überprüfen, um Trends und potenzielle Bereiche zu identifizieren, die möglicherweise mehr Analyse oder Aufmerksamkeit erfordern. Der Usage Explorer bietet ein Kreisdiagramm, in dem der Gesamtverbrauch und die Kosten angezeigt werden. Es besteht die Möglichkeit, die Gesamtsummen in kleinere Zwischensummen zu gruppieren.

Note

In dem Diagramm werden nur die fünf besten Ergebnisse angezeigt, wobei andere Ergebnisse in einem Abschnitt „Andere“ zusammengefasst sind. Sie können alle Ergebnisse im Aufschlüsselungsbereich unter dem Diagramm einsehen.

Cost Explorer

Visualize and understand costs incurred in FuzzyPixelFarm-M8-1025. The numbers displayed here are estimation and may be different from the AWS Cost Explorer.



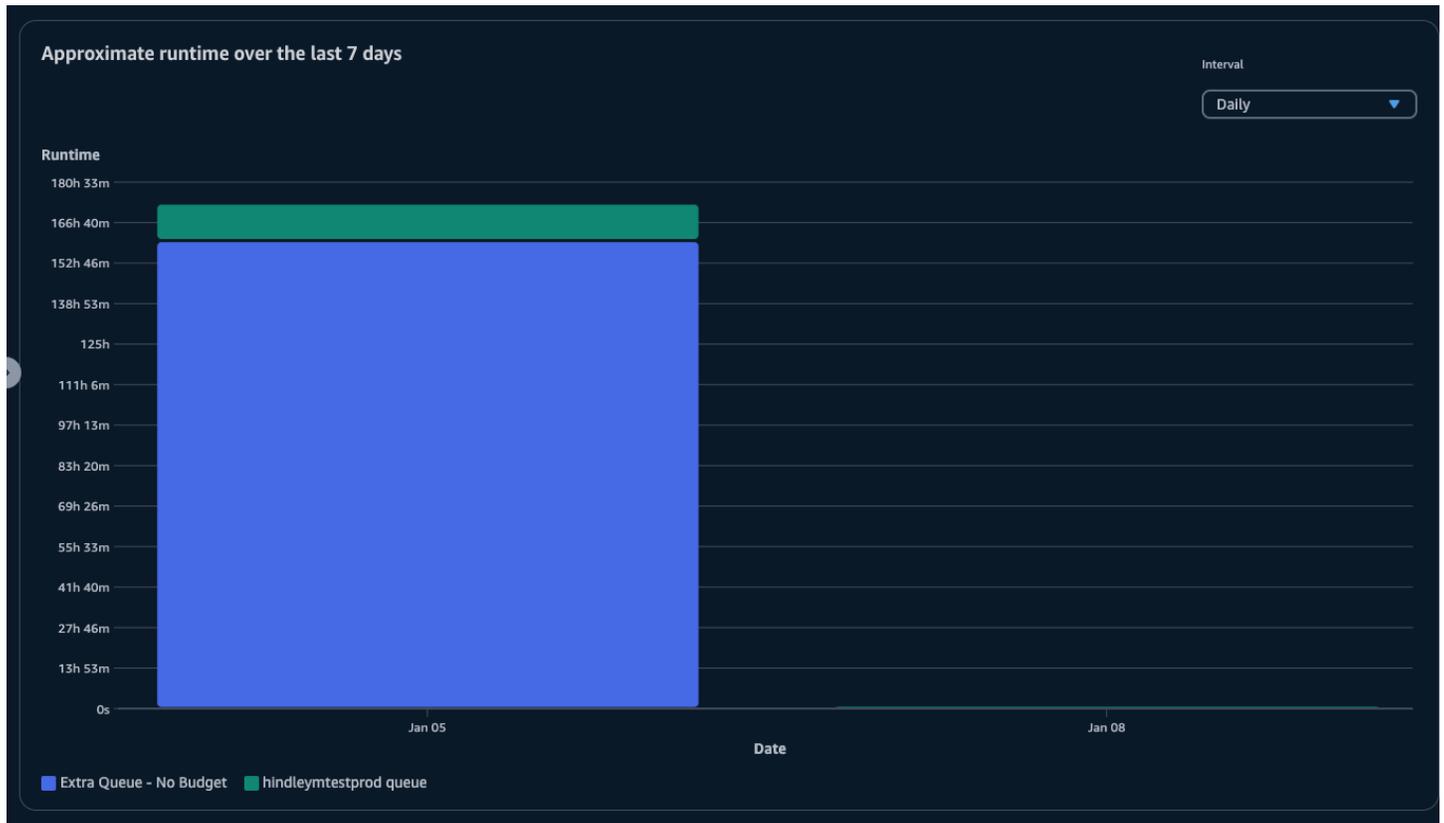
Sehen Sie sich eine Aufschlüsselung der Messwerte an

Unter dem Kreisdiagramm bietet der Usage Explorer eine detailliertere Aufschlüsselung bestimmter Metriken, die sich ändern, wenn sich die Parameter ändern. Standardmäßig werden im Usage Explorer fünf Ergebnisse angezeigt. Mithilfe der Seitennummerierungspfeile im Aufschlüsselungsbereich können Sie durch die Ergebnisse blättern.

Die Aufschlüsselung ist standardmäßig minimiert. Um die Ergebnisse zu erweitern und anzuzeigen, wählen Sie den Pfeil Alle Aufschlüsselung anzeigen aus. Um die Aufschlüsselung herunterzuladen, wählen Sie Daten herunterladen.

Sehen Sie sich die ungefähre Laufzeit der Warteschlangen an

Sie können auch die ungefähre Laufzeit Ihrer Warteschlangen anhand verschiedener von Ihnen festgelegter Intervalle anzeigen. Die Intervalloptionen sind stündlich, täglich, wöchentlich und monatlich. Nachdem Sie ein Intervall ausgewählt haben, zeigt das Diagramm die ungefähre Laufzeit Ihrer Warteschlangen an.



Kostenmanagement

AWS Deadline Cloud bietet Budgets und den Usage Explorer, mit dem Sie die Kosten für Ihre Jobs kontrollieren und visualisieren können. Deadline Cloud verwendet jedoch andere AWS Dienste wie Amazon S3. Die Kosten für diese Dienste sind nicht in den Budgets von Deadline Cloud oder im Usage Explorer enthalten und werden je nach Nutzung separat berechnet. Je nachdem, wie Sie Deadline Cloud konfigurieren, können Sie die folgenden und andere AWS Dienste nutzen:

Service	Seite mit der Preisgestaltung
CloudWatch Amazon-Protokolle	Preise für Amazon CloudWatch Logs
Amazon Elastic Compute Cloud	Preise für Amazon Elastic Compute Cloud
AWS Key Management Service	AWS Key Management Service -Preise
AWS PrivateLink	AWS PrivateLink -Preise
Amazon Simple Storage Service	Amazon Simple Storage Service – Preise

Service	Seite mit der Preisgestaltung
Amazon Virtual Private Cloud	Preise für Amazon Virtual Private Cloud

Bewährte Methoden für das Kostenmanagement

Mithilfe der folgenden bewährten Methoden können Sie Ihre Kosten bei der Verwendung von Deadline Cloud sowie die Kompromisse, die Sie zwischen Kosten und Effizienz eingehen können, besser verstehen und kontrollieren.

Note

Die endgültigen Kosten für die Nutzung von Deadline Cloud hängen von der Interaktion zwischen einer Reihe von AWS Diensten, dem Arbeitsaufwand, den Sie verarbeiten, und dem Ort ab, an AWS-Region dem Sie Ihre Jobs ausführen. Die folgenden bewährten Methoden sind Richtlinien und können die Kosten möglicherweise nicht wesentlich senken.

Bewährte Methoden für CloudWatch Protokolle

Deadline Cloud sendet Mitarbeiter- und CloudWatch Aufgabenprotokolle an Logs. Es wird Ihnen in Rechnung gestellt, diese Protokolle zu sammeln, zu speichern und zu analysieren. Sie können die Kosten senken, indem Sie nur die Mindestmenge an Daten protokollieren, die für die Überwachung Ihrer Aufgaben erforderlich sind.

Wenn Sie eine Warteschlange oder Flotte erstellen, erstellt Deadline Cloud eine CloudWatch Logs-Protokollgruppe mit den folgenden Namen:

- `aws/deadline/<FARM_ID>/<FLEET_ID>`
- `aws/deadline/<FARM_ID>/<QUEUE_ID>`

Standardmäßig laufen diese Protokolle nie ab. Sie können die Aufbewahrungsrichtlinie von Protokollgruppen anpassen, um alte Protokolle zu entfernen und die Speicherkosten zu senken. Sie können Protokolle auch nach Amazon S3 exportieren. Die Speicherkosten von Amazon S3 sind niedriger als die für CloudWatch. Weitere Informationen finden Sie unter [Exportieren von Protokolldaten zu Amazon S3](#).

Bewährte Methoden für Amazon EC2

Sie können Amazon EC2 EC2-Instances sowohl für vom Service verwaltete als auch für kundenverwaltete Flotten verwenden. Es gibt drei Überlegungen:

- Bei servicemanagierten Flotten können Sie wählen, ob eine oder mehrere Instances jederzeit verfügbar sein sollen, indem Sie die Mindestanzahl an Mitarbeitern für die Flotte festlegen. Wenn Sie die Mindestanzahl an Arbeitskräften auf 0 setzen, sind in der Flotte immer so viele Mitarbeiter im Einsatz. Dadurch kann die Zeit reduziert werden, die Deadline Cloud benötigt, um mit der Verarbeitung von Jobs zu beginnen. Allerdings wird Ihnen die Leerlaufzeit der Instanz in Rechnung gestellt.
- Legen Sie für servicemanagierte Flotten eine maximale Größe für die Flotte fest. Dadurch wird die Anzahl der Instanzen begrenzt, auf die eine Flotte auto skaliert werden kann. Flotten werden diese Größe nicht überschreiten, selbst wenn mehr Jobs darauf warten, bearbeitet zu werden.
- Sowohl für vom Service verwaltete als auch für kundenverwaltete Flotten können Sie die Amazon EC2 EC2-Instance-Typen in Ihren Flotten angeben. Die Verwendung kleinerer Instances kostet weniger pro Minute, kann aber länger dauern, bis ein Auftrag abgeschlossen ist. Umgekehrt kostet eine größere Instanz mehr pro Minute, kann aber die Zeit bis zur Fertigstellung eines Jobs reduzieren. Wenn Sie die Anforderungen verstehen, die Ihre Jobs an eine Instanz stellen, können Sie Ihre Kosten senken.
- Wählen Sie nach Möglichkeit Amazon EC2 Spot-Instances für Ihre Flotte aus. Spot-Instances sind zu einem reduzierten Preis erhältlich, können jedoch durch On-Demand-Anfragen unterbrochen werden. On-Demand-Instances werden sekundengenau berechnet und nicht unterbrochen.

Bewährte Methoden für AWS KMS

Standardmäßig verschlüsselt Deadline Cloud Ihre Daten mit einem AWS eigenen Schlüssel. Dieser Schlüssel wird Ihnen nicht in Rechnung gestellt.

Sie können sich dafür entscheiden, einen vom Kunden verwalteten Schlüssel zur Verschlüsselung Ihrer Daten zu verwenden. Wenn Sie Ihren eigenen Schlüssel verwenden, wird Ihnen die Gebühr auf der Grundlage der Verwendung Ihres Schlüssels berechnet. Wenn Sie einen vorhandenen Schlüssel verwenden, fallen zusätzliche Kosten für die zusätzliche Nutzung an.

Bewährte Methoden für AWS PrivateLink

Sie können AWS PrivateLink verwenden, um mithilfe eines Schnittstellenendpunkts eine Verbindung zwischen Ihrer VPC und Deadline Cloud herzustellen. Wenn Sie eine Verbindung

herstellen, können Sie alle Deadline Cloud-API-Aktionen aufrufen. Für jeden Endpunkt, den Sie erstellen, wird Ihnen pro Stunde eine Gebühr berechnet. Wenn Sie verwenden PrivateLink, müssen Sie mindestens drei Endpunkte erstellen, und je nach Konfiguration benötigen Sie möglicherweise bis zu fünf.

Bewährte Methoden für Amazon S3

Deadline Cloud verwendet Amazon S3, um Ressourcen für die Verarbeitung, Jobanhänge, Ausgaben und Protokolle zu speichern. Um die mit Amazon S3 verbundenen Kosten zu senken, reduzieren Sie die Datenmenge, die Sie speichern. Einige Vorschläge:

- Speichern Sie nur Ressourcen, die derzeit verwendet werden oder in Kürze verwendet werden.
- Verwenden Sie eine [S3-Lifecycle-Konfiguration](#), um ungenutzte Dateien automatisch aus einem S3-Bucket zu löschen.

Bewährte Methoden für Amazon VPC

Wenn Sie die nutzungsbasierte Lizenzierung für Ihre vom Kunden verwaltete Flotte verwenden, erstellen Sie einen Deadline Cloud-Lizenzendpunkt, bei dem es sich um einen Amazon VPC-Endpunkt handelt, der in Ihrem Konto erstellt wurde. Dieser Endpunkt wird mit einem Stundensatz berechnet. Um die Kosten zu senken, entfernen Sie die Endgeräte, wenn Sie keine nutzungsbasierten Lizenzen verwenden.

Sicherheit in Deadline Cloud

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame AWS Verantwortung von Ihnen und Ihnen. Das [Modell der geteilten Verantwortung](#) beschreibt dies als Sicherheit der Cloud selbst und Sicherheit in der Cloud:

- **Sicherheit der Cloud** — AWS ist verantwortlich für den Schutz der Infrastruktur, die AWS-Services in der läuft AWS Cloud. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Externe Prüfer testen und verifizieren regelmäßig die Wirksamkeit unserer Sicherheitsmaßnahmen im Rahmen der [AWS](#) . Weitere Informationen zu den Compliance-Programmen, die für gelten AWS Deadline Cloud, finden Sie [AWS-Services unter Umfang nach Compliance-Programmen](#) AWS-Services und unter .
- **Sicherheit in der Cloud** — Ihre Verantwortung richtet sich nach dem AWS-Service , was Sie verwenden. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung anwenden können Deadline Cloud. In den folgenden Themen erfahren Sie, wie Sie die Konfiguration vornehmen Deadline Cloud , um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere verwenden können AWS-Services , die Ihnen bei der Überwachung und Sicherung Ihrer Deadline Cloud Ressourcen helfen.

Themen

- [Datenschutz in Deadline Cloud](#)
- [Identity and Access Management in Deadline Cloud](#)
- [Konformitätsprüfung für Deadline Cloud](#)
- [Resilienz in Deadline Cloud](#)
- [Sicherheit der Infrastruktur in Deadline Cloud](#)
- [Konfiguration und Schwachstellenanalyse in Deadline Cloud](#)
- [Serviceübergreifende Confused-Deputy-Prävention](#)
- [Zugriff AWS Deadline Cloud über einen Schnittstellenendpunkt \(AWS PrivateLink\)](#)

- [Bewährte Sicherheitsmethoden für Deadline Cloud](#)

Datenschutz in Deadline Cloud

Das [Modell der AWS gemeinsamen Verantwortung](#) und geteilter Verantwortung gilt für den Datenschutz in AWS Deadline Cloud. 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
- 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 140-2-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter [Federal Information Processing Standard \(FIPS\) 140-2](#).

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 API Deadline Cloud oder den SDKs arbeiten oder diese anderweitig AWS-Services verwenden. AWS CLI AWS 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.

Themen

- [Verschlüsselung im Ruhezustand](#)
- [Verschlüsselung während der Übertragung](#)
- [Schlüsselverwaltung](#)
- [Datenschutz für den Datenverkehr zwischen Netzwerken](#)
- [Abmelden](#)

Verschlüsselung im Ruhezustand

AWS Deadline Cloud schützt sensible Daten, indem sie im Ruhezustand mit den in [AWS Key Management Service \(AWS KMS\)](#) gespeicherten Verschlüsselungsschlüsseln verschlüsselt werden. Verschlüsselung im Ruhezustand ist überall verfügbar, AWS-Regionen wo sie verfügbar Deadline Cloud ist.

Die Verschlüsselung von Daten bedeutet, dass sensible Daten, die auf Festplatten gespeichert sind, für einen Benutzer oder eine Anwendung ohne gültigen Schlüssel nicht lesbar sind. Nur eine Partei mit einem gültigen verwalteten Schlüssel kann die Daten entschlüsseln.

Informationen darüber, wie Deadline Cloud Daten im Ruhezustand verschlüsselt werden, finden Sie unter. AWS KMS [Schlüsselverwaltung](#)

Verschlüsselung während der Übertragung

AWS Deadline Cloud verwendet Transport Layer Security (TLS) 1.2 oder 1.3 für die Verschlüsselung von Daten, die zwischen dem Dienst und den Workern gesendet werden. Wir benötigen TLS 1.2 und empfehlen TLS 1.3. Wenn Sie eine Virtual Private Cloud (VPC) verwenden, können Sie darüber hinaus eine private Verbindung zwischen Ihrer VPC und herstellen. AWS PrivateLink Deadline Cloud

Schlüsselverwaltung

Wenn Sie eine neue Farm erstellen, können Sie einen der folgenden Schlüssel zum Verschlüsseln Ihrer Farmdaten wählen:

- **AWS eigener KMS-Schlüssel** — Standardverschlüsselungstyp, wenn Sie beim Erstellen der Farm keinen Schlüssel angeben. Der KMS-Schlüssel gehört AWS Deadline Cloud. Sie können AWS eigene Schlüssel nicht anzeigen, verwalten oder verwenden. Sie müssen jedoch keine Maßnahmen ergreifen, um die Schlüssel zu schützen, mit denen Ihre Daten verschlüsselt werden. Weitere Informationen finden Sie [AWS im AWS Key Management Service Entwicklerhandbuch unter Eigene Schlüssel](#).
- **Kundenverwalteter KMS-Schlüssel** — Sie geben einen vom Kunden verwalteten Schlüssel an, wenn Sie eine Farm erstellen. Der gesamte Inhalt der Farm ist mit dem KMS-Schlüssel verschlüsselt. Der Schlüssel wird in Ihrem Konto gespeichert und wird von Ihnen erstellt, gehört und verwaltet. Es fallen AWS KMS Gebühren an. Sie haben die volle Kontrolle über den KMS-Schlüssel. Sie können folgende Aufgaben ausführen:
 - Festlegung und Aufrechterhaltung wichtiger Richtlinien
 - Festlegung und Aufrechterhaltung von IAM-Richtlinien und -Zuschüssen
 - Aktivieren und Deaktivieren wichtiger Richtlinien
 - Hinzufügen von Tags
 - Erstellen von Schlüsselaliasen

Sie können einen kundeneigenen Schlüssel, der in einer Deadline Cloud Farm verwendet wird, nicht manuell rotieren. Die automatische Rotation des Schlüssels wird unterstützt.

Weitere Informationen finden Sie im AWS Key Management Service Entwicklerhandbuch unter [Schlüssel, die dem Kunden gehören](#).

Um einen vom Kunden verwalteten Schlüssel zu erstellen, folgen Sie den Schritten [unter Erstellen symmetrischer kundenverwalteter Schlüssel](#) im AWS Key Management Service Entwicklerhandbuch.

Wie werden Deadline Cloud Zuschüsse verwendet AWS KMS

Deadline Cloud Für die Nutzung Ihres vom Kunden verwalteten Schlüssels ist ein [Zuschuss](#) erforderlich. Wenn Sie eine Farm erstellen, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt ist, erstellt Deadline Cloud das Programm in Ihrem Namen einen Zuschuss, indem es eine [CreateGrant](#) Anfrage an AWS KMS sendet, um Zugriff auf den von Ihnen angegebenen KMS-Schlüssel zu erhalten.

Deadline Cloud verwendet mehrere Zuschüsse. Jeder Grant wird von einem anderen Teil verwendet Deadline Cloud, der Ihre Daten ver- oder entschlüsseln muss. Deadline Cloud verwendet auch

Zuschüsse, um den Zugriff auf andere AWS Dienste zu ermöglichen, die zum Speichern von Daten in Ihrem Namen verwendet werden, wie Amazon Simple Storage Service, Amazon Elastic Block Store oder OpenSearch.

Zuschüsse, die Deadline Cloud die Verwaltung von Maschinen in einer vom `GranteePrincipal` Service verwalteten Flotte ermöglichen, beinhalten eine Deadline Cloud Kontonummer und eine Rolle als Service Principal. Dies ist zwar nicht üblich, aber notwendig, um Amazon EBS-Volumes für Mitarbeiter in serviceverwalteten Flotten mit dem für die Farm angegebenen kundenverwalteten KMS-Schlüssel zu verschlüsseln.

Kundenverwaltete CMK-Schlüsselrichtlinie

Schlüsselrichtlinien steuern den Zugriff auf den vom Kunden verwalteten Schlüssel. Jeder Schlüssel muss über genau eine Schlüsselrichtlinie verfügen, die Aussagen enthält, die festlegen, wer den Schlüssel verwenden darf und wie er verwendet werden darf. Wenn Sie Ihren vom Kunden verwalteten Schlüssel erstellen, können Sie eine Schlüsselrichtlinie angeben. Weitere Informationen finden Sie unter [Verwalten des Zugriffs auf kundenverwaltete Schlüssel](#) im Entwicklerhandbuch zum AWS Key Management Service .

Minimale IAM-Richtlinie für CreateFarm

Um Ihren vom Kunden verwalteten Schlüssel zum Erstellen von Farmen mithilfe der Konsole oder des [CreateFarm](#) API-Vorgangs zu verwenden, müssen die folgenden AWS KMS API-Operationen zugelassen sein:

- [kms:CreateGrant](#): Fügt einem kundenverwalteten Schlüssel eine Erteilung hinzu. Gewährt Konsolenzugriff auf einen angegebenen AWS KMS Schlüssel. Weitere Informationen finden Sie im AWS Key Management Service Entwicklerhandbuch unter [Using Grants](#).
- [kms:Decrypt](#)— Ermöglicht Deadline Cloud das Entschlüsseln von Daten in der Farm.
- [kms:DescribeKey](#)— Stellt dem Kunden verwaltete Schlüsseldetails zur Verfügung, damit Deadline Cloud der Schlüssel validiert werden kann.
- [kms:GenerateDataKey](#)— Ermöglicht Deadline Cloud die Verschlüsselung von Daten mit einem eindeutigen Datenschlüssel.

Die folgende Richtlinienerklärung gewährt die erforderlichen Berechtigungen für den `CreateFarm` Vorgang.

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "DeadlineCreateGrants",
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt",
      "kms:GenerateDataKey",
      "kms:CreateGrant",
      "kms:DescribeKey"
    ],
    "Resource": "arn:aws::kms:us-west-2:111122223333:key/1234567890abcdef0",
    "Condition": {
      "StringEquals": {
        "kms:ViaService": "deadline.us-west-2.amazonaws.com"
      }
    }
  }
]
```

Minimale IAM-Richtlinie für schreibgeschützte Operationen

Um Ihren vom Kunden verwalteten Schlüssel für schreibgeschützte Deadline Cloud Operationen zu verwenden, z. B. für das Abrufen von Informationen über Farmen, Warteschlangen und Flotten. Die folgenden AWS KMS API-Operationen müssen zulässig sein:

- [kms:Decrypt](#)— Ermöglicht Deadline Cloud das Entschlüsseln von Daten in der Farm.
- [kms:DescribeKey](#)— Stellt dem Kunden verwaltete Schlüsseldetails zur Verfügung, damit Deadline Cloud der Schlüssel validiert werden kann.

Die folgende Richtlinienerklärung gewährt die erforderlichen Berechtigungen für schreibgeschützte Operationen.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeadlineReadOnly",
      "Effect": "Allow",
      "Action": [
```

```

        "kms:Decrypt",
        "kms:DescribeKey"
    ],
    "Resource": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
    "Condition": {
        "StringEquals": {
            "kms:ViaService": "deadline.us-west-2.amazonaws.com"
        }
    }
}
]
}

```

Minimale IAM-Richtlinie für Lese- und Schreibvorgänge

Um Ihren vom Kunden verwalteten Schlüssel für Lese- und Deadline Cloud Schreibvorgänge wie das Erstellen und Aktualisieren von Farmen, Warteschlangen und Flotten zu verwenden. Die folgenden AWS KMS API-Operationen müssen zulässig sein:

- [kms:Decrypt](#)— Ermöglicht Deadline Cloud das Entschlüsseln von Daten in der Farm.
- [kms:DescribeKey](#)— Stellt dem Kunden verwaltete Schlüsseldetails zur Verfügung, damit Deadline Cloud der Schlüssel validiert werden kann.
- [kms:GenerateDataKey](#)— Ermöglicht Deadline Cloud die Verschlüsselung von Daten mit einem eindeutigen Datenschlüssel.

Die folgende Richtlinienerklärung gewährt die erforderlichen Berechtigungen für den CreateFarm Vorgang.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeadlineReadWrite",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:GenerateDataKey",
      ],
    }
  ],
}

```

```

    "Resource": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
    "Condition": {
      "StringEquals": {
        "kms:ViaService": "deadline.us-west-2.amazonaws.com"
      }
    }
  }
]
}

```

Überwachen Ihrer Verschlüsselungsschlüssel

Wenn Sie einen vom AWS KMS Kunden verwalteten Schlüssel für Ihre Deadline Cloud Farmen verwenden, können Sie [Amazon CloudWatch Logs](#) verwenden [AWS CloudTrail](#), um Anfragen zu verfolgen, die Deadline Cloud an gesendet AWS KMS werden.

CloudTrail Veranstaltung für Zuschüsse

Das folgende CloudTrail Beispiereignis tritt ein, wenn Zuschüsse erstellt werden, in der Regel, wenn Sie die CreateFleet Operation CreateFarmCreateMonitor, oder aufrufen.

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws::sts::111122223333:assumed-role/Admin/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE3",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE",
        "arn": "arn:aws::iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2024-04-23T02:05:26Z",
        "mfaAuthenticated": "false"
      }
    }
  }
}

```

```

    },
    "invokedBy": "deadline.amazonaws.com"
  },
  "eventTime": "2024-04-23T02:05:35Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "CreateGrant",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "deadline.amazonaws.com",
  "userAgent": "deadline.amazonaws.com",
  "requestParameters": {
    "operations": [
      "CreateGrant",
      "Decrypt",
      "DescribeKey",
      "Encrypt",
      "GenerateDataKey"
    ],
    "constraints": {
      "encryptionContextSubset": {
        "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
        "aws:deadline:accountId": "111122223333"
      }
    },
    "granteePrincipal": "deadline.amazonaws.com",
    "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "retiringPrincipal": "deadline.amazonaws.com"
  },
  "responseElements": {
    "grantId": "6bbe819394822a400fe5e3a75d0e9ef16c1733143fff0c1fc00dc7ac282a18a0",
    "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
  },
  "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
  "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
  "readOnly": false,
  "resources": [
    {
      "accountId": "AWS Internal",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE44444"
    }
  ],

```

```
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"eventCategory": "Management"
}
```

CloudTrail Ereignis für die Entschlüsselung

Das folgende CloudTrail Beispiereignis tritt ein, wenn Werte mithilfe des vom Kunden verwalteten KMS-Schlüssels entschlüsselt werden.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws::sts::111122223333:assumed-role/SampleRole/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE",
        "arn": "arn:aws::iam::111122223333:role/SampleRole",
        "accountId": "111122223333",
        "userName": "SampleRole"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2024-04-23T18:46:51Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "deadline.amazonaws.com"
  },
  "eventTime": "2024-04-23T18:51:44Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "Decrypt",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "deadline.amazonaws.com",
  "userAgent": "deadline.amazonaws.com",
  "requestParameters": {
    "encryptionContext": {
```

```

    "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
    "aws:deadline:accountId": "111122223333",
    "aws-crypto-public-key": "AotL+SAMPLEVALUEiOMEXAMPLEEaaqNOTREALaGTESTONLY
+p/5H+EuKd4Q=="
  },
  "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
  "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
},
"responseElements": null,
"requestID": "aaaaaaaa-bbbb-cccc-dddd-eeeeefffffff",
"eventID": "ffffffff-eeee-dddd-cccc-bbbbbbaaaaaa",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"eventCategory": "Management"
}

```

CloudTrail Ereignis für die Verschlüsselung

Das folgende CloudTrail Beispiereignis tritt ein, wenn Werte mit dem vom Kunden verwalteten KMS-Schlüssel verschlüsselt werden.

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws::sts::111122223333:assumed-role/SampleRole/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",

```

```

        "principalId": "AROAIKDTESTANDEXAMPLE",
        "arn": "arn:aws::iam::111122223333:role/SampleRole",
        "accountId": "111122223333",
        "userName": "SampleRole"
    },
    "webIdFederationData": {},
    "attributes": {
        "creationDate": "2024-04-23T18:46:51Z",
        "mfaAuthenticated": "false"
    }
},
"invokedBy": "deadline.amazonaws.com"
},
"eventTime": "2024-04-23T18:52:40Z",
"eventSource": "kms.amazonaws.com",
"eventName": "GenerateDataKey",
"awsRegion": "us-west-2",
"sourceIPAddress": "deadline.amazonaws.com",
"userAgent": "deadline.amazonaws.com",
"requestParameters": {
    "numberOfBytes": 32,
    "encryptionContext": {
        "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
        "aws:deadline:accountId": "111122223333",
        "aws-crypto-public-key": "AotL+SAMPLEVALUEiOMEXAMPLEEaaqNOTREALaGTESTONLY
+p/5H+EuKd4Q=="
    }
},
"keyId": "arn:aws::kms:us-
west-2:111122223333:key/abcdef12-3456-7890-0987-654321fedcba"
},
"responseElements": null,
"requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
"readOnly": true,
"resources": [
    {
        "accountId": "111122223333",
        "type": "AWS::KMS::Key",
        "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE33333"
    }
],
"eventType": "AwsApiCall",
"managementEvent": true,

```

```
"recipientAccountId": "111122223333",  
"eventCategory": "Management"  
}
```

Löschen eines vom Kunden verwalteten KMS-Schlüssels

Das Löschen eines vom Kunden verwalteten KMS-Schlüssels in AWS Key Management Service (AWS KMS) ist destruktiv und potenziell gefährlich. Dadurch werden das Schlüsselmaterial und alle mit dem Schlüssel verknüpften Metadaten unwiderruflich gelöscht. Nachdem ein vom Kunden verwalteter KMS-Schlüssel gelöscht wurde, können Sie die mit diesem Schlüssel verschlüsselten Daten nicht mehr entschlüsseln. Das bedeutet, dass die Daten nicht mehr wiederhergestellt werden können.

Aus diesem Grund AWS KMS haben Kunden eine Wartezeit von bis zu 30 Tagen, bevor der KMS-Schlüssel gelöscht wird. Die Standardwartezeit beträgt 30 Tage.

Über die Wartezeit

Da das Löschen eines vom Kunden verwalteten KMS-Schlüssels zerstörerisch und potenziell gefährlich ist, müssen Sie eine Wartezeit von 7–30 Tagen festlegen. Die Standardwartezeit beträgt 30 Tage.

Die tatsächliche Wartezeit kann jedoch bis zu 24 Stunden länger sein als der von Ihnen geplante Zeitraum. Verwenden Sie den [DescribeKey](#) Vorgang, um das tatsächliche Datum und die Uhrzeit der Löschung des Schlüssels zu ermitteln. Sie können das geplante Löschedatum eines Schlüssels auch in der [AWS KMS Konsole](#) auf der Detailseite des Schlüssels im Abschnitt Allgemeine Konfiguration sehen. Beachten Sie die Zeitzone.

Während der Wartezeit lautet der Status und der Schlüsselstatus des vom Kunden verwalteten Schlüssels **Ausstehende Löschung**.

- Ein vom Kunden verwalteter KMS-Schlüssel, dessen Löschung aussteht, kann für keine [kryptografischen Operationen](#) verwendet werden.
- AWS KMS [rotiert nicht die Backing-Schlüssel](#) von vom Kunden verwalteten KMS-Schlüsseln, deren Löschung noch aussteht.

Weitere Informationen zum Löschen eines vom Kunden verwalteten KMS-Schlüssels finden Sie unter [Löschen von Kundenhauptschlüsseln](#) im AWS Key Management Service Entwicklerhandbuch.

Datenschutz für den Datenverkehr zwischen Netzwerken

AWS Deadline Cloud unterstützt Amazon Virtual Private Cloud (Amazon VPC) zur Sicherung von Verbindungen. Amazon VPC bietet Funktionen, mit denen Sie die Sicherheit Ihrer Virtual Private Cloud (VPC) erhöhen und überwachen können.

Sie können eine vom Kunden verwaltete Flotte (CMF) mit Amazon Elastic Compute Cloud (Amazon EC2) -Instances einrichten, die in einer VPC ausgeführt werden. Durch die Bereitstellung von Amazon VPC-Endpunkten zur Nutzung AWS PrivateLink bleibt der Datenverkehr zwischen Workern in Ihrem CMF und dem Deadline Cloud Endpunkt innerhalb Ihrer VPC. Darüber hinaus können Sie Ihre VPC so konfigurieren, dass der Internetzugang auf Ihre Instances beschränkt wird.

In serviceverwalteten Flotten sind die Mitarbeiter nicht über das Internet erreichbar, aber sie haben Internetzugang und stellen über das Internet eine Verbindung zum Deadline Cloud Service her.

Abmelden

AWS Deadline Cloud sammelt bestimmte Betriebsinformationen, um uns bei der Entwicklung und Verbesserung zu unterstützen Deadline Cloud. Zu den gesammelten Daten gehören Dinge wie Ihre AWS Konto-ID und Benutzer-ID, sodass wir Sie korrekt identifizieren können, falls Sie ein Problem mit der haben Deadline Cloud. Wir erfassen auch Deadline Cloud spezifische Informationen wie Ressourcen-IDs (eine FarmID oder QueueID, falls zutreffend), den Produktnamen (z. B. JobAttachments WorkerAgent, und mehr) und die Produktversion.

Sie können diese Datenerfassung mithilfe der Anwendungskonfiguration deaktivieren. Jeder Computer Deadline Cloud, mit dem sowohl Client-Workstations als auch Flottenmitarbeiter interagiert, muss sich separat abmelden.

Deadline Cloud Monitor — Desktop

Deadline Cloud monitor — desktop sammelt Betriebsinformationen, z. B. wann Abstürze auftreten und wann die Anwendung geöffnet wird, damit wir wissen, wenn Sie Probleme mit der Anwendung haben. Um die Erfassung dieser Betriebsinformationen zu deaktivieren, deaktivieren Sie auf der Einstellungsseite die Option Datenerfassung aktivieren, um die Leistung von Deadline Cloud Monitor zu messen.

Nachdem Sie sich abmelden, sendet der Desktop-Monitor die Betriebsdaten nicht mehr. Alle zuvor gesammelten Daten werden gespeichert und können weiterhin zur Verbesserung des

Dienstes verwendet werden. Weitere Informationen finden Sie in den [Häufig gestellten Fragen zum Datenschutz](#).

AWS Deadline Cloud CLI und Tools

Die AWS Deadline Cloud CLI, die Einreicher und der Worker Agent sammeln alle Betriebsinformationen, z. B. wann Abstürze auftreten und wann Jobs eingereicht werden, damit wir wissen, wenn Sie Probleme mit diesen Anwendungen haben. Verwenden Sie eine der folgenden Methoden, um sich von der Erfassung dieser Betriebsinformationen abzumelden:

- Geben Sie im Terminal ein **deadline config set telemetry.opt_out true**.

Dadurch werden die CLI, die Einreicher und der Worker-Agent deaktiviert, wenn sie als aktueller Benutzer ausgeführt werden.

- Fügen Sie bei der Installation des Deadline Cloud Worker-Agenten das **--telemetry-opt-out** Befehlszeilenargument hinzu. z. B. **./install.sh --farm-id \$FARM_ID --fleet-id \$FLEET_ID --telemetry-opt-out**.
- Bevor Sie den Worker-Agent, die CLI oder den Submitter ausführen, legen Sie eine Umgebungsvariable fest: **DEADLINE_CLOUD_TELEMETRY_OPT_OUT=true**

Nach dem Abmelden senden die Deadline Cloud Tools keine Betriebsdaten mehr. Alle zuvor gesammelten Daten werden gespeichert und können weiterhin zur Verbesserung des Dienstes verwendet werden. Weitere Informationen finden Sie in den [Häufig gestellten Fragen zum Datenschutz](#).

Identity and Access Management in Deadline Cloud

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 Deadline Cloud-Ressourcen zu verwenden. IAM ist ein Programm AWS-Service, das Sie ohne zusätzliche Kosten nutzen können.

Themen

- [Zielgruppe](#)
- [Authentifizierung mit Identitäten](#)

- [Verwalten des Zugriffs mit Richtlinien](#)
- [So funktioniert Deadline Cloud mit IAM](#)
- [Beispiele für identitätsbasierte Richtlinien für Deadline Cloud](#)
- [AWS verwaltete Richtlinien für Deadline Cloud](#)
- [Fehlerbehebung bei AWS Deadline Cloud-Identität und -Zugriff](#)

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, die Sie in Deadline Cloud erledigen.

Dienstbenutzer — Wenn Sie den Deadline Cloud-Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die Anmeldeinformationen und Berechtigungen zur Verfügung, die Sie benötigen. Wenn Sie für Ihre Arbeit mehr Funktionen von Deadline Cloud verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Funktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern müssen. Wenn Sie auf eine Funktion in Deadline Cloud nicht zugreifen können, finden Sie weitere Informationen unter [Fehlerbehebung bei AWS Deadline Cloud-Identität und -Zugriff](#).

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für die Ressourcen von Deadline Cloud verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf Deadline Cloud. Es ist Ihre Aufgabe, zu bestimmen, auf welche Funktionen und Ressourcen von Deadline Cloud Ihre Servicebenutzer zugreifen sollen. Sie müssen dann Anträge an Ihren IAM-Administrator stellen, um die Berechtigungen Ihrer Servicenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM mit Deadline Cloud nutzen kann, finden Sie unter [So funktioniert Deadline Cloud mit IAM](#).

IAM-Administrator — Wenn Sie ein IAM-Administrator sind, möchten Sie vielleicht mehr darüber erfahren, wie Sie Richtlinien schreiben können, um den Zugriff auf Deadline Cloud zu verwalten. Beispiele für identitätsbasierte Richtlinien von Deadline Cloud, die Sie in IAM verwenden können, finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Deadline Cloud](#)

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 Zugangsportale 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 darauf 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, um Anfragen selbst zu [signieren, finden Sie im IAM-Benutzerhandbuch unter AWS API-Anfragen](#) signieren.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen angeben. 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 [Verwenden der Multi-Faktor-Authentifizierung \(MFA\) in AWS](#) im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie ein AWS-Konto erstellen, 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 und verwenden Sie diese, 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 einer Gruppe mit dem Namen IAMAdmins Berechtigungen zum Verwalten 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 [Erstellen eines IAM-Benutzers \(anstatt einer Rolle\)](#) im IAM-Benutzerhandbuch.

IAM-Rollen

Eine [IAM-Rolle](#) ist eine Identität innerhalb Ihres Unternehmens AWS-Konto, die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, ist aber nicht mit einer bestimmten Person verknüpft. Sie können vorübergehend eine IAM-Rolle in der übernehmen, AWS Management Console indem Sie die Rollen [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 [Verwenden von IAM-Rollen](#) 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](#) 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 zum Unterschied zwischen Rollen und ressourcenbasierten Richtlinien für den kontenübergreifenden Zugriff finden Sie unter [Kontenübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch](#).
- **Serviceübergreifender Zugriff** — Einige verwenden Funktionen in anderen. AWS-Services AWS-Services Wenn Sie beispielsweise einen Aufruf in einem Service tätigen, führt dieser Service häufig Anwendungen in Amazon-EC2 aus oder speichert Objekte in Amazon-S3. 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.
- **Anwendungen, die auf Amazon EC2 ausgeführt werden** — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2-Instance ausgeführt werden und API-Anfragen stellen AWS CLI . AWS Das ist eher zu empfehlen, als Zugriffsschlüssel innerhalb der EC2-Instance zu speichern. Um einer EC2-Instance eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instance-Profil, das an die Instance angehängt ist. Ein Instance-Profil enthält die Rolle und ermöglicht, dass Programme, die in der EC2-Instance ausgeführt werden, temporäre Anmeldeinformationen erhalten. Weitere Informationen finden Sie unter [Verwenden einer IAM-Rolle zum Erteilen von Berechtigungen für Anwendungen, die auf Amazon-EC2-Instances ausgeführt werden](#) im IAM-Benutzerhandbuch.

Informationen dazu, wann Sie IAM-Rollen oder IAM-Benutzer verwenden sollten, finden Sie unter [Erstellen einer IAM-Rolle \(anstatt eines Benutzers\)](#) im IAM-Benutzerhandbuch.

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. Berechtigungen in den Richtlinien bestimmen, ob die 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 auf was Zugriff hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

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 [Erstellen von IAM-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 eingebundenen Richtlinie

wählen, finden Sie unter [Auswahl 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.

Zugriffssteuerungslisten (ACLs)

Zugriffssteuerungslisten (ACLs) steuern, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) auf eine Ressource zugreifen können. ACLs sind ähnlich wie ressourcenbasierte Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3 und Amazon VPC sind Beispiele für Services, die ACLs unterstützen. AWS WAF Weitere Informationen“ zu ACLs finden Sie unter [Zugriffskontrollliste \(ACL\) – Übersicht](#) (Access Control List) im Amazon-Simple-Storage-Service-Entwicklerhandbuch.

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.

- **Service Control Policies (SCPs)** — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Objekte AWS-Konten, die Ihrem Unternehmen gehören. Wenn Sie innerhalb einer Organisation alle Features aktivieren, können Sie Service-Kontrollrichtlinien (SCPs) auf alle oder einzelne Ihrer Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Entitäten. Root-Benutzer des AWS-Kontos Weitere Informationen zu Organizations und SCPs finden Sie unter [Funktionsweise von SCPs](#) 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 funktioniert Deadline Cloud mit IAM

Bevor Sie IAM verwenden, um den Zugriff auf Deadline Cloud zu verwalten, sollten Sie sich darüber informieren, welche IAM-Funktionen mit Deadline Cloud verwendet werden können.

IAM-Funktionen, die Sie mit Deadline Cloud verwenden können AWS

IAM-Feature	Deadline Cloud-Unterstützung
Identitätsbasierte Richtlinien	Ja

IAM-Feature	Deadline Cloud-Unterstützung
Ressourcenbasierte Richtlinien	Nein
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Richtlinienbedingungsschlüssel (servicespezifisch)	Ja
ACLs	Nein
ABAC (Tags in Richtlinien)	Ja
Temporäre Anmeldeinformationen	Ja
Forward Access Sessions (FAS)	Ja
Servicerollen	Ja
Service-verknüpfte Rollen	Nein

Einen allgemeinen Überblick darüber, wie Deadline Cloud und andere mit den meisten IAM-Funktionen AWS-Services funktionieren, finden Sie im [IAM-Benutzerhandbuch unter AWS Dienste, die mit IAM funktionieren](#).

Identitätsbasierte Richtlinien für Deadline Cloud

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 [Erstellen von IAM-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 Deadline Cloud

Beispiele für identitätsbasierte Richtlinien von Deadline Cloud finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Deadline Cloud](#)

Ressourcenbasierte Richtlinien in Deadline Cloud

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 Prinzipalentsität (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 [Kontenübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

Richtlinienmaßnahmen für Deadline Cloud

Unterstützt Richtlinienaktionen Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

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 Deadline Cloud-Aktionen finden Sie unter [Von AWS Deadline Cloud definierte Aktionen](#) in der Serviceautorisierungsreferenz.

Richtlinienaktionen in Deadline Cloud verwenden vor der Aktion das folgende Präfix:

```
deadline
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [  
  "deadline:action1",  
  "deadline:action2"  
]
```

Beispiele für identitätsbasierte Richtlinien von Deadline Cloud finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Deadline Cloud](#)

Richtlinienressourcen für Deadline Cloud

Unterstützt Richtlinienressourcen	Ja
-----------------------------------	----

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das bedeutet die Festlegung, welcher Prinzipal Aktionen für welche Ressourcen 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 Deadline Cloud-Ressourcentypen und ihrer ARNs finden Sie unter [Von AWS Deadline Cloud definierte Ressourcen](#) in der Service Authorization Reference. Informationen dazu, mit welchen Aktionen Sie den ARN jeder Ressource angeben können, finden Sie unter [Von AWS Deadline Cloud definierte Aktionen](#).

Beispiele für identitätsbasierte Richtlinien von Deadline Cloud finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Deadline Cloud](#)

Bedingungsschlüssel für Richtlinien für Deadline Cloud

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 Bedingungs Schlüssel angeben, 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 Bedingungs Schlüssel und dienstspezifische Bedingungs Schlüssel. Eine Übersicht aller AWS globalen Bedingungs Schlüssel finden Sie unter [Kontextschlüssel für AWS globale Bedingungen](#) im IAM-Benutzerhandbuch.

Eine Liste der Deadline Cloud-Bedingungs Schlüssel finden Sie unter [Bedingungs Schlüssel für AWS Deadline Cloud](#) in der Service Authorization Reference. Informationen zu den Aktionen und Ressourcen, mit denen Sie einen Bedingungs Schlüssel verwenden können, finden Sie unter [Von AWS Deadline Cloud definierte Aktionen](#).

Beispiele für identitätsbasierte Richtlinien von Deadline Cloud finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Deadline Cloud](#)

ACLs in Deadline Cloud

Unterstützt ACLs

Nein

Zugriffssteuerungslisten (ACLs) steuern, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) auf eine Ressource zugreifen können. ACLs sind ähnlich wie ressourcenbasierte Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

ABAC mit Deadline Cloud

Unterstützt ABAC (Tags in Richtlinien)	Ja
--	----

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 Bedingungelement 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 [Was ist ABAC?](#) im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe [Attributbasierte Zugriffskontrolle \(ABAC\)](#) verwenden im IAM-Benutzerhandbuch.

Verwenden temporärer Anmeldeinformationen mit Deadline Cloud

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 zu einer 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 für Deadline Cloud weiterleiten

Unterstützt Forward Access Sessions (FAS)	Ja
---	----

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 Deadline Cloud

Unterstützt Servicerollen	Ja
---------------------------	----

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

Das Ändern der Berechtigungen für eine Servicerolle kann die Funktionalität von Deadline Cloud beeinträchtigen. Bearbeiten Sie Servicerollen nur, wenn Deadline Cloud Sie dazu anleitet.

Servicebezogene Rollen für Deadline Cloud

Unterstützt serviceverknüpfte Rollen	Nein
--------------------------------------	------

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.

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 Deadline Cloud

Standardmäßig sind Benutzer und Rollen nicht berechtigt, Deadline Cloud-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mithilfe der AWS Management Console, AWS Command Line Interface (AWS CLI) oder AWS API 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](#) im IAM-Benutzerhandbuch.

Einzelheiten zu den von Deadline Cloud definierten Aktionen und Ressourcentypen, einschließlich des Formats der ARNs für jeden Ressourcentyp, finden Sie unter [Aktionen, Ressourcen und Bedingungsschlüssel für AWS Deadline Cloud](#) in der Service Authorization Reference.

Themen

- [Bewährte Methoden für Richtlinien](#)
- [Verwenden Sie die Deadline Cloud-Konsole](#)
- [Richtlinie zum Einreichen von Jobs an eine Warteschlange](#)
- [Richtlinie, die die Erstellung eines Lizenzendpunkts ermöglicht](#)
- [Richtlinie, die die Überwachung einer bestimmten Farmwarteschlange ermöglicht](#)

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand Deadline Cloud-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 [Richtlinienvvalidierung zum 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 [Konfigurieren eines MFA-geschützten API-Zugriffs](#) im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter [Bewährte Methoden für die Sicherheit in IAM](#) im IAM-Benutzerhandbuch.

Verwenden Sie die Deadline Cloud-Konsole

Um auf die AWS Deadline Cloud-Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den Deadline Cloud-Ressourcen in Ihrem aufzulisten und einzusehen AWS-Konto. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Um sicherzustellen, dass Benutzer und Rollen die Deadline Cloud-Konsole weiterhin verwenden können, fügen Sie den Entitäten auch die Deadline Cloud *ConsoleAccess* oder die *ReadOnly* AWS verwaltete Richtlinie hinzu. Weitere Informationen finden Sie unter [Hinzufügen von Berechtigungen zu einem Benutzer](#) im IAM-Benutzerhandbuch.

Richtlinie zum Einreichen von Jobs an eine Warteschlange

In diesem Beispiel erstellen Sie eine Richtlinie mit eingeschränktem Geltungsbereich, die die Berechtigung zum Senden von Aufträgen an eine bestimmte Warteschlange in einer bestimmten Farm erteilt.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "SubmitJobsFarmAndQueue",
      "Effect": "Allow",
      "Action": "deadline:CreateJob",
      "Resource": "arn:aws:deadline:REGION:ACCOUNT_ID:farm/FARM_A/queue/QUEUE_B/job/*"
    }
  ]
}
```

Richtlinie, die die Erstellung eines Lizenzendpunkts ermöglicht

In diesem Beispiel erstellen Sie eine nach unten abgegrenzte Richtlinie, die die erforderlichen Berechtigungen zum Erstellen und Verwalten von Lizenzendpunkten gewährt. Verwenden Sie diese Richtlinie, um den Lizenzendpunkt für die VPC zu erstellen, die Ihrer Farm zugeordnet ist.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "SID": "CreateLicenseEndpoint",
    "Effect": "Allow",
    "Action": [
      "deadline:CreateLicenseEndpoint",
      "deadline>DeleteLicenseEndpoint",
      "deadline:GetLicenseEndpoint",
      "deadline:UpdateLicenseEndpoint",
      "deadline>ListLicenseEndpoints",
      "deadline:PutMeteredProduct",
      "deadline>DeleteMeteredProduct",
      "deadline>ListMeteredProducts",
      "deadline>ListAvailableMeteredProducts",
      "ec2:CreateVpcEndpoint",
      "ec2:DescribeVpcEndpoints",
    ]
  }]
}
```

```

        "ec2:DeleteVpcEndpoints"
    ],
    "Resource": "*"
}]
}

```

Richtlinie, die die Überwachung einer bestimmten Farmwarteschlange ermöglicht

In diesem Beispiel erstellen Sie eine Richtlinie mit eingeschränktem Geltungsbereich, die die Erlaubnis erteilt, Jobs in einer bestimmten Warteschlange für eine bestimmte Farm zu überwachen.

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "MonitorJobsFarmAndQueue",
    "Effect": "Allow",
    "Action": [
      "deadline:SearchJobs",
      "deadline:ListJobs",
      "deadline:GetJob",
      "deadline:SearchSteps",
      "deadline:ListSteps",
      "deadline:ListStepConsumers",
      "deadline:ListStepDependencies",
      "deadline:GetStep",
      "deadline:SearchTasks",
      "deadline:ListTasks",
      "deadline:GetTask",
      "deadline:ListSessions",
      "deadline:GetSession",
      "deadline:ListSessionActions",
      "deadline:GetSessionAction"
    ],
    "Resource": [
      "arn:aws:deadline:REGION:123456789012:farm/FARM_A/queue/QUEUE_B",
      "arn:aws:deadline:REGION:123456789012:farm/FARM_A/queue/QUEUE_B/*"
    ]
  }]
}

```

AWS verwaltete Richtlinien für Deadline Cloud

Eine AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von erstellt und verwaltet wird AWS. AWS Verwaltete Richtlinien sind so konzipiert, dass sie Berechtigungen für viele gängige Anwendungsfälle bereitstellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS verwaltete Richtlinien für Ihre speziellen Anwendungsfälle möglicherweise keine Berechtigungen mit den geringsten Rechten gewähren, da sie allen AWS Kunden zur Verfügung stehen. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie [kundenverwaltete Richtlinien](#) definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Sie können die in AWS verwalteten Richtlinien definierten Berechtigungen nicht ändern. Wenn die in einer AWS verwalteten Richtlinie definierten Berechtigungen AWS aktualisiert werden, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert eine AWS verwaltete Richtlinie höchstwahrscheinlich, wenn eine neue Richtlinie eingeführt AWS-Service wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter [Von AWS verwaltete Richtlinien](#) im IAM-Benutzerhandbuch.

AWS verwaltete Richtlinie: AWSDeadlineCloud-FleetWorker

Sie können die `AWSDeadlineCloud-FleetWorker` Richtlinie an Ihre AWS Identity and Access Management (IAM-) Identitäten anhängen.

Diese Richtlinie gewährt den Mitarbeitern dieser Flotte die Berechtigungen, die sie benötigen, um eine Verbindung mit dem Service herzustellen und Aufgaben vom Service zu empfangen.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen:

- `deadline`— Ermöglicht es Prinzipalen, Mitarbeiter in einer Flotte zu verwalten.

Eine JSON-Liste der Richtliniendetails finden Sie unter [AWSDeadlineCloud— FleetWorker](#) im Referenzhandbuch zu AWS Managed Policy.

AWS verwaltete Richtlinie: AWSDeadlineCloud-WorkerHost

Sie können die `AWSDeadlineCloud-WorkerHost`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie gewährt die Berechtigungen, die für die anfängliche Verbindung mit dem Dienst erforderlich sind. Es kann als Amazon Elastic Compute Cloud (Amazon EC2) Instance-Profil verwendet werden.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen:

- `deadline`— Ermöglicht es Prinzipalen, Worker zu erstellen.

Eine JSON-Liste der Richtliniendetails finden Sie unter [AWSDeadlineCloud— WorkerHost](#) im Referenzhandbuch zu AWS Managed Policy.

AWS verwaltete Richtlinie: AWSDeadlineCloud-UserAccessFarms

Sie können die `AWSDeadlineCloud-UserAccessFarms`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie ermöglicht Benutzern den Zugriff auf Farmdaten auf der Grundlage der Farmen, in denen sie Mitglied sind, und ihrer Mitgliedschaftsstufe.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen:

- `deadline`— Ermöglicht dem Benutzer den Zugriff auf Farmdaten.
- `ec2`— Ermöglicht Benutzern, Details zu Amazon EC2 EC2-Instance-Typen zu sehen.
- `identitystore`— Ermöglicht Benutzern, Benutzer- und Gruppennamen zu sehen.

Eine JSON-Liste der Richtliniendetails finden Sie unter [AWSDeadlineCloud— UserAccess Farmen](#) im Referenzhandbuch zu AWS Managed Policy.

AWS verwaltete Richtlinie: AWSDeadlineCloud-UserAccessFleets

Sie können die `AWSDeadlineCloud-UserAccessFleets`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie ermöglicht Benutzern den Zugriff auf Flottendaten auf der Grundlage der Farmen, in denen sie Mitglied sind, und ihrer Mitgliedschaftsstufe.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen:

- `deadline`— Ermöglicht dem Benutzer den Zugriff auf Farmdaten.
- `ec2`— Ermöglicht Benutzern, Details zu Amazon EC2 EC2-Instance-Typen zu sehen.
- `identitystore`— Ermöglicht Benutzern, Benutzer- und Gruppennamen zu sehen.

Eine JSON-Liste der Richtliniendetails finden Sie unter [AWSDeadlineCloud— UserAccess Flotten](#) im Referenzhandbuch zu AWS Managed Policy.

AWS verwaltete Richtlinie: AWSDeadlineCloud-UserAccessJobs

Sie können die `AWSDeadlineCloud-UserAccessJobs`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie ermöglicht Benutzern den Zugriff auf Auftragsdaten auf der Grundlage der Farmen, in denen sie Mitglied sind, und ihrer Mitgliedschaftsstufe.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen:

- `deadline`— Ermöglicht dem Benutzer den Zugriff auf Farmdaten.
- `ec2`— Ermöglicht Benutzern, Details zu Amazon EC2 EC2-Instance-Typen zu sehen.
- `identitystore`— Ermöglicht Benutzern, Benutzer- und Gruppennamen zu sehen.

Eine JSON-Liste der Richtliniendetails finden Sie unter [AWSDeadlineCloud— UserAccess Jobs](#) im Referenzhandbuch zu AWS Managed Policy.

AWS verwaltete Richtlinie: AWSDeadlineCloud-UserAccessQueues

Sie können die `AWSDeadlineCloud-UserAccessQueues`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie ermöglicht Benutzern den Zugriff auf Warteschlangendaten auf der Grundlage der Farmen, in denen sie Mitglied sind, und ihrer Mitgliedschaftsstufe.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen:

- `deadline`— Ermöglicht dem Benutzer den Zugriff auf Farmdaten.
- `ec2`— Ermöglicht Benutzern, Details zu Amazon EC2 EC2-Instance-Typen zu sehen.
- `identitystore`— Ermöglicht Benutzern, Benutzer- und Gruppennamen zu sehen.

Eine JSON-Liste der Richtliniendetails finden Sie unter [AWSDeadlineCloud-UserAccessQueues](#) im Referenzhandbuch zu AWS Managed Policy.

Deadline Cloud-Updates für AWS verwaltete Richtlinien

Sehen Sie sich Details zu Aktualisierungen der AWS verwalteten Richtlinien für Deadline Cloud an, seit dieser Dienst begonnen hat, diese Änderungen zu verfolgen. Um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten, abonnieren Sie den RSS-Feed auf der Deadline Cloud-Dokumentverlaufsseite.

Änderung	Beschreibung	Datum
Deadline Cloud hat damit begonnen, Änderungen nachzuverfolgen	Deadline Cloud begann, Änderungen an seinen AWS verwalteten Richtlinien nachzuverfolgen.	2. April 2024

Fehlerbehebung bei AWS Deadline Cloud-Identität und -Zugriff

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit Deadline Cloud und IAM auftreten können.

Themen

- [Ich bin nicht berechtigt, eine Aktion in Deadline Cloud durchzuführen](#)
- [Ich bin nicht berechtigt, iam auszuführen: PassRole](#)
- [Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine Deadline Cloud-Ressourcen ermöglichen](#)

Ich bin nicht berechtigt, eine Aktion in Deadline Cloud durchzuführen

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 `deadline:GetWidget`-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
deadline:GetWidget on resource: my-example-widget
```

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `deadline: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 eine Fehlermeldung erhalten, dass Sie nicht berechtigt sind, die `iam:PassRole` Aktion durchzuführen, müssen Ihre Richtlinien aktualisiert werden, damit Sie eine Rolle an Deadline Cloud übergeben 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 Deadline Cloud 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 Deadline Cloud-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. Im Fall von Diensten, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (Access Control Lists, ACLs) verwenden, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob Deadline Cloud diese Funktionen unterstützt, finden Sie unter [So funktioniert Deadline Cloud mit IAM](#).
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im [IAM-Benutzerhandbuch unter Zugriff 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 im IAM-Benutzerhandbuch unter [Kontenübergreifender Ressourcenzugriff in IAM](#).

Konformitätsprüfung für Deadline Cloud

Informationen darüber, ob AWS-Service ein [AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt](#), finden Sie unter [Umfang nach Compliance-Programm AWS-Services unter](#) . Wählen Sie dort das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter [AWS Compliance-Programme AWS](#) .

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter [Berichte herunterladen unter](#) .

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services 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 Bereitstellung von Basisumgebungen beschrieben AWS , bei denen Sicherheit und Compliance im Mittelpunkt stehen.
- [Architecting for HIPAA Security and Compliance on Amazon Web Services](#) — In diesem Whitepaper wird beschrieben, wie Unternehmen HIPAA-fähige Anwendungen erstellen AWS können.

 Note

AWS-Services Nicht alle sind HIPAA-fähig. Weitere Informationen finden Sie in der [Referenz für HIPAA-berechtigte Services](#).

- [AWS Compliance-Ressourcen](#) — Diese Sammlung von Arbeitsmapen und Leitfäden gilt möglicherweise für Ihre Branche und Ihren Standort.
- [AWS Leitfäden zur Einhaltung von Vorschriften für Kunden](#) — Verstehen Sie das Modell der gemeinsamen Verantwortung aus dem Blickwinkel der Einhaltung von Vorschriften. In den Leitfäden werden die bewährten Verfahren zur Sicherung zusammengefasst AWS-Services und die Leitlinien den Sicherheitskontrollen in verschiedenen Frameworks (einschließlich des National Institute of Standards and Technology (NIST), des Payment Card Industry Security Standards Council (PCI) und der International Organization for Standardization (ISO)) zugeordnet.
- [Evaluierung von Ressourcen anhand von Regeln](#) im AWS Config Entwicklerhandbuch — Der AWS Config Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.
- [AWS Security Hub](#) — Auf diese AWS-Service Weise erhalten Sie einen umfassenden Überblick über Ihren internen Sicherheitsstatus. AWS Security Hub verwendet Sicherheitskontrollen, um Ihre AWS -Ressourcen zu bewerten und Ihre Einhaltung von Sicherheitsstandards und bewährten Methoden zu überprüfen. Eine Liste der unterstützten Services und Kontrollen finden Sie in der [Security-Hub-Steuerungsreferenz](#).
- [Amazon GuardDuty](#) — Dies AWS-Service erkennt potenzielle Bedrohungen für Ihre Workloads AWS-Konten, Container und Daten, indem es Ihre Umgebung auf verdächtige und böswillige Aktivitäten überwacht. GuardDuty kann Ihnen helfen, verschiedene Compliance-Anforderungen

wie PCI DSS zu erfüllen, indem es die in bestimmten Compliance-Frameworks vorgeschriebenen Anforderungen zur Erkennung von Eindringlingen erfüllt.

- [AWS Audit Manager](#)— Auf diese AWS-Service Weise können Sie Ihre AWS Nutzung kontinuierlich überprüfen, um das Risikomanagement und die Einhaltung von Vorschriften und Industriestandards zu vereinfachen.

Resilienz in Deadline Cloud

Die AWS globale Infrastruktur basiert auf Availability AWS-Regionen Zones. AWS-Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Zonen ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu Availability Zones AWS-Regionen und Availability Zones finden Sie unter [AWS Globale](#) Infrastruktur.

AWS Deadline Cloud sichert keine Daten, die in Ihrem S3-Bucket für Jobanhänge gespeichert sind. Sie können Backups Ihrer Job-Anhangsdaten mit jedem standardmäßigen Amazon S3 S3-Backup-Mechanismus wie [S3 Versioning](#) oder [AWS Backup](#) aktivieren.

Sicherheit der Infrastruktur in Deadline Cloud

Als verwalteter Service ist AWS Deadline Cloud 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, um über das Netzwerk auf Deadline Cloud zuzugreifen. 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.

Deadline Cloud unterstützt die Verwendung von AWS PrivateLink Virtual Private Cloud (VPC) - Endpunktrichtlinien nicht. Es verwendet die AWS PrivateLink Standardrichtlinie, die vollen Zugriff auf den Endpunkt gewährt. Weitere Informationen finden Sie im AWS PrivateLink Benutzerhandbuch unter [Standard-Endpunktrichtlinie](#).

Konfiguration und Schwachstellenanalyse in Deadline Cloud

AWS kümmert sich um grundlegende Sicherheitsaufgaben wie das Patchen von Gastbetriebssystemen (OS) und Datenbanken, die Firewall-Konfiguration und die Notfallwiederherstellung. Diese Verfahren wurden von qualifizierten Dritten überprüft und zertifiziert. Weitere Informationen finden Sie in den folgenden Ressourcen:

- [Modell der übergreifenden Verantwortlichkeit](#)
- [Amazon Web Services: Übersicht über Sicherheitsverfahren](#) (Whitepaper)

AWS Deadline Cloud verwaltet Aufgaben auf vom Service oder vom Kunden verwalteten Flotten:

- Für vom Service verwaltete Flotten verwaltet Deadline Cloud das Gastbetriebssystem.
- Bei vom Kunden verwalteten Flotten sind Sie für die Verwaltung des Betriebssystems verantwortlich.

Weitere Informationen zur Konfiguration und Schwachstellenanalyse für AWS Deadline Cloud finden Sie unter

- [Bewährte Sicherheitsmethoden für Deadline Cloud](#)

Serviceübergreifende Confused-Deputy-Prävention

Das Confused-Deputy-Problem ist ein Sicherheitsproblem, bei dem eine juristische Stelle, die nicht über die Berechtigung zum Ausführen einer Aktion verfügt, eine privilegiertere juristische Stelle zwingen kann, die Aktion auszuführen. In AWS kann ein dienstübergreifendes Identitätswechsels zum Problem des verwirrten Stellvertreters führen. Ein dienstübergreifender Identitätswechsel

kann auftreten, wenn ein Dienst (der Anruf-Dienst) einen anderen Dienst anruft (den aufgerufenen Dienst). Der aufrufende Service kann manipuliert werden, um seine Berechtigungen zu verwenden, um Aktionen auf die Ressourcen eines anderen Kunden auszuführen, für die er sonst keine Zugriffsberechtigung haben sollte. Um dies zu verhindern, bietet AWS Tools, mit denen Sie Ihre Daten für alle Services mit Serviceprinzipalen schützen können, die Zugriff auf Ressourcen in Ihrem Konto erhalten haben.

Wir empfehlen, die Kontextschlüssel [aws:SourceArn](#) und die [aws:SourceAccount](#) globalen Bedingungsschlüssel in Ressourcenrichtlinien zu verwenden, um die Berechtigungen einzuschränken, die der AWS Deadline Cloud Ressource einen anderen Dienst gewähren. Verwenden Sie `aws:SourceArn`, wenn Sie nur eine Ressource mit dem betriebsübergreifenden Zugriff verknüpfen möchten. Verwenden Sie `aws:SourceAccount`, wenn Sie zulassen möchten, dass Ressourcen in diesem Konto mit der betriebsübergreifenden Verwendung verknüpft werden.

Der effektivste Weg, um sich vor dem Verwirrter-Stellvertreter-Problem zu schützen, ist die Verwendung des `aws:SourceArn` globalen Bedingungskontextschlüssels mit dem vollständigen Amazon-Ressourcenname (ARN) der Ressource. Wenn Sie den vollständigen ARN der Ressource nicht kennen oder wenn Sie mehrere Ressourcen angeben, verwenden Sie den globalen Kontextbedingungsschlüssel `aws:SourceArn` mit Platzhalterzeichen (*) für die unbekanntenen Teile des ARN. z. B. `arn:aws:deadline:*:123456789012:*`.

Wenn der `aws:SourceArn`-Wert die Konto-ID nicht enthält, z. B. einen Amazon-S3-Bucket-ARN, müssen Sie beide globale Bedingungskontextschlüssel verwenden, um Berechtigungen einzuschränken.

Das folgende Beispiel zeigt, wie Sie die Kontextschlüssel `aws:SourceArn` und die `aws:SourceAccount` globale Bedingung verwenden können, Deadline Cloud um das Problem des verwirrten Stellvertreters zu vermeiden.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "deadline.amazonaws.com"
    },
    "Action": "deadline:ActionName",
    "Resource": [
      "*"
    ]
  }
}
```

```
  ],
  "Condition": {
    "ArnLike": {
      "aws:SourceArn": "arn:aws:deadline:*:123456789012:*"
    },
    "StringEquals": {
      "aws:SourceAccount": "123456789012"
    }
  }
}
```

Zugriff AWS Deadline Cloud über einen Schnittstellenendpunkt (AWS PrivateLink)

Sie können AWS PrivateLink damit eine private Verbindung zwischen Ihrer VPC und AWS Deadline Cloud herstellen. Sie können darauf zugreifen, Deadline Cloud als ob es in Ihrer VPC wäre, ohne ein Internet-Gateway, ein NAT-Gerät, eine VPN-Verbindung oder AWS Direct Connect eine Verbindung zu verwenden. Instances in Ihrer VPC benötigen für den Zugriff Deadline Cloud keine öffentlichen IP-Adressen.

Sie stellen diese private Verbindung her, indem Sie einen Schnittstellen-Endpunkt erstellen, der von AWS PrivateLink unterstützt wird. Wir erstellen eine Endpunkt-Netzwerkschnittstelle in jedem Subnetz, das Sie für den Schnittstellen-Endpunkt aktivieren. Hierbei handelt es sich um vom Anforderer verwaltete Netzwerkschnittstellen, die als Eingangspunkt für den Datenverkehr dienen, der für Deadline Cloud bestimmt ist.

Weitere Informationen finden Sie unter [Zugriff auf AWS-Services über AWS PrivateLink](#) im AWS PrivateLink -Leitfaden.

Überlegungen zu Deadline Cloud

Bevor Sie einen Schnittstellenendpunkt für einrichten Deadline Cloud, finden Sie weitere Informationen unter [Zugreifen auf einen AWS-Service mithilfe eines Schnittstellen-VPC-Endpunkts](#) im AWS PrivateLink Handbuch.

Deadline Cloud unterstützt Aufrufe aller API-Aktionen über den Schnittstellenendpunkt.

Standardmäßig Deadline Cloud ist der vollständige Zugriff auf über den Schnittstellenendpunkt zulässig. Alternativ können Sie den Endpunkt-Netzwerkschnittstellen eine Sicherheitsgruppe zuordnen, um den Datenverkehr Deadline Cloud über den Schnittstellenendpunkt zu kontrollieren.

Deadline Cloud unterstützt keine VPC-Endpunktrichtlinien. Weitere Informationen finden Sie im Handbuch unter [Steuern des Zugriffs auf VPC-Endpunkte mithilfe von Endpunktrichtlinien](#).AWS PrivateLink

Deadline Cloud Endpunkte

Deadline Cloud verwendet zwei Endpunkte für den Zugriff auf den Dienst mithilfe von. AWS PrivateLink

Mitarbeiter verwenden den `com.amazonaws.region.deadline.scheduling` Endpunkt, um Aufgaben aus der Warteschlange abzurufen, ihnen den Fortschritt zu Deadline Cloud melden und die Aufgabenausgabe zurückzusenden. Wenn Sie eine vom Kunden verwaltete Flotte verwenden, ist der Terminplanungsendpunkt der einzige Endpunkt, den Sie erstellen müssen, es sei denn, Sie verwenden Verwaltungsoperationen. Wenn durch einen Auftrag beispielsweise mehr Jobs erstellt werden, müssen Sie den Verwaltungsendpunkt so einrichten, dass er den `CreateJob` Vorgang aufrufen kann.

Der Deadline Cloud Monitor verwendet den, `com.amazonaws.region.deadline.management` um die Ressourcen in Ihrer Farm zu verwalten, z. B. Warteschlangen und Flotten zu erstellen und zu ändern oder Listen mit Aufträgen, Schritten und Aufgaben abzurufen.

Deadline Cloud erfordert außerdem Endpunkte für die folgenden AWS Dienstendpunkte:

- Deadline Cloud verwendet AWS STS , um Mitarbeiter zu authentifizieren, sodass sie auf Arbeitsressourcen zugreifen können. Weitere Informationen zu AWS STS finden Sie unter [Temporäre Sicherheitsanmeldeinformationen in IAM](#) im AWS Identity and Access Management Benutzerhandbuch.
- Wenn Sie Ihre vom Kunden verwaltete Flotte in einem Subnetz ohne Internetverbindung einrichten, müssen Sie einen VPC-Endpunkt für Amazon CloudWatch Logs einrichten, damit Mitarbeiter Protokolle schreiben können. [Weitere Informationen finden Sie unter Überwachung mit CloudWatch](#)
- Wenn Sie Jobanhänge verwenden, müssen Sie einen VPC-Endpunkt für Amazon Simple Storage Service (Amazon S3) erstellen, damit Mitarbeiter auf die Anlagen zugreifen können. Weitere Informationen finden Sie unter [Jobanhänge in Deadline Cloud](#).

Erstellen Sie Endpunkte für Deadline Cloud

Sie können Schnittstellen-Endpunkte für die Deadline Cloud Verwendung entweder der Amazon VPC-Konsole oder der AWS Command Line Interface (AWS CLI) erstellen. Weitere Informationen finden Sie unter [Erstellen eines Schnittstellenendpunkts](#) im AWS PrivateLink -Leitfaden.

Erstellen Sie Verwaltungs- und Planungsendpunkte für die Deadline Cloud Verwendung der folgenden Servicenamen. Ersetzen Sie *Region* durch die Region AWS-Region, in der Sie bereitgestellt Deadline Cloud haben.

```
com.amazonaws.region.deadline.management
```

```
com.amazonaws.region.deadline.scheduling
```

Wenn Sie privates DNS für die Schnittstellenendpunkte aktivieren, können Sie API-Anfragen Deadline Cloud unter Verwendung des standardmäßigen regionalen DNS-Namens stellen. Zum Beispiel `worker.deadline.us-east-1.amazonaws.com` für Arbeitsoperationen oder `management.deadline.us-east-1.amazonaws.com` für alle anderen Operationen.

Sie müssen auch einen Endpunkt für die AWS STS Verwendung des folgenden Dienstnamens erstellen:

```
com.amazonaws.region.sts
```

Wenn sich Ihre vom Kunden verwaltete Flotte in einem Subnetz ohne Internetverbindung befindet, müssen Sie einen CloudWatch Logs-Endpunkt mit dem folgenden Dienstnamen erstellen:

```
com.amazonaws.region.logs
```

Wenn Sie Auftragsanhänge zum Übertragen von Dateien verwenden, müssen Sie einen Amazon S3 S3-Endpunkt mit dem folgenden Servicenamen erstellen:

```
com.amazonaws.region.s3
```

Bewährte Sicherheitsmethoden für Deadline Cloud

AWS Deadline Cloud (Deadline Cloud) bietet eine Reihe von Sicherheitsfunktionen, die Sie bei der Entwicklung und Implementierung Ihrer eigenen Sicherheitsrichtlinien berücksichtigen sollten. Die

folgenden bewährten Methoden sind allgemeine Richtlinien und keine vollständige Sicherheitslösung. Da diese bewährten Methoden für Ihre Umgebung möglicherweise nicht angemessen oder ausreichend sind, sollten Sie sie als hilfreiche Überlegungen und nicht als bindend ansehen.

Note

Weitere Informationen zur Bedeutung vieler Sicherheitsthemen finden Sie im [Modell der gemeinsamen Verantwortung](#).

Datenschutz

Aus Datenschutzgründen empfehlen wir, dass Sie Ihre AWS-Konto Anmeldeinformationen schützen und individuelle Konten mit 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
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie fortschrittliche verwaltete Sicherheitsdienste wie Amazon Macie, die Sie bei der Erkennung und Sicherung personenbezogener Daten unterstützen, die in Amazon Simple Storage Service (Amazon S3) gespeichert sind.
- Wenn Sie für den Zugriff auf AWS über eine Befehlszeilenschnittstelle oder über eine API FIPS 140-2-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter [Federal Information Processing Standard \(FIPS\) 140-2](#).

Wir empfehlen dringend, in Freitextfeldern wie z. B. im Feld Name keine sensiblen, identifizierenden Informationen wie Kontonummern von Kunden einzugeben. Dies gilt auch, wenn Sie mit AWS Deadline Cloud oder anderen Anwendungen arbeiten und die Konsole, die API oder AWS SDKs AWS-Services verwenden. AWS CLI Alle Daten, die Sie in Deadline Cloud oder andere Dienste eingeben, werden möglicherweise zur Aufnahme in Diagnoseprotokolle aufgenommen. Wenn Sie

eine URL für einen externen Server bereitstellen, schließen Sie keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL ein.

AWS Identity and Access Management Berechtigungen

Verwalten Sie den Zugriff auf AWS Ressourcen mithilfe von Benutzern und AWS Identity and Access Management (IAM-) Rollen und indem Sie Benutzern die geringsten Rechte gewähren. Richten Sie Richtlinien und Verfahren zur Verwaltung von Anmeldeinformationen für die Erstellung, Verteilung, Rotation und den Widerruf AWS von Zugangsdaten ein. Weitere Informationen finden Sie unter [Bewährte Methoden für IAM](#) im IAM-Benutzerhandbuch.

Führen Sie Jobs als Benutzer und Gruppen aus

Bei der Verwendung der Warteschlangenfunktion in Deadline Cloud hat es sich bewährt, einen Betriebssystembenutzer (OS) und seine primäre Gruppe anzugeben, sodass der Betriebssystembenutzer die geringsten Rechte für die Jobs der Warteschlange hat.

Wenn Sie die Option „Als Benutzer ausführen“ (und Gruppe) angeben, werden alle Prozesse für Jobs, die an die Warteschlange gesendet werden, mit diesem Betriebssystembenutzer ausgeführt und erben die zugehörigen Betriebssystemberechtigungen dieses Benutzers.

Die Kombination der Flotten- und Warteschlangenkonfigurationen sorgt für ein gewisses Maß an Sicherheit. Auf der Warteschlangenseite können die Rolle „Job wird als Benutzer ausgeführt“ und die IAM-Rolle angegeben werden, um das Betriebssystem und die AWS Berechtigungen für die Jobs der Warteschlange zu verwenden. Die Flotte definiert die Infrastruktur (Worker-Hosts, Netzwerke, bereitgestellter gemeinsam genutzter Speicher), über die Jobs innerhalb der Warteschlange ausgeführt werden, sofern sie einer bestimmten Warteschlange zugeordnet sind. Auf die auf den Worker-Hosts verfügbaren Daten müssen Jobs aus einer oder mehreren zugehörigen Warteschlangen zugreifen können. Die Angabe eines Benutzers oder einer Gruppe trägt dazu bei, die Daten in Jobs vor anderen Warteschlangen, anderer installierter Software oder anderen Benutzern mit Zugriff auf die Worker-Hosts zu schützen. Wenn es in einer Warteschlange keinen Benutzer gibt, wird sie als Agent-Benutzer ausgeführt, der sich als (sudo) beliebiger Warteschlangenbenutzer ausgeben kann. Auf diese Weise kann eine Warteschlange ohne Benutzer die Rechte an eine andere Warteschlange weiterleiten.

Netzwerk

Um zu verhindern, dass der Datenverkehr abgefangen oder umgeleitet wird, müssen Sie unbedingt sicherstellen, wie und wohin Ihr Netzwerkverkehr geleitet wird.

Wir empfehlen Ihnen, Ihre Netzwerkumgebung auf folgende Weise zu sichern:

- Sichere Subnetz-Routing-Tabellen für Amazon Virtual Private Cloud (Amazon VPC), um zu kontrollieren, wie der Datenverkehr auf IP-Ebene weitergeleitet wird.
- Wenn Sie Amazon Route 53 (Route 53) als DNS-Anbieter in Ihrem Farm- oder Workstation-Setup verwenden, sichern Sie den Zugriff auf die Route 53-API.
- Wenn Sie eine Verbindung zu Deadline Cloud außerhalb herstellen, AWS z. B. über lokale Workstations oder andere Rechenzentren, sichern Sie jede lokale Netzwerkinfrastruktur. Dazu gehören DNS-Server und Routing-Tabellen auf Routern, Switches und anderen Netzwerkgeräten.

Jobs und Jobdaten

Deadline Cloud-Jobs werden innerhalb von Sitzungen auf Worker-Hosts ausgeführt. In jeder Sitzung werden ein oder mehrere Prozesse auf dem Worker-Host ausgeführt. Für die Ausgabe müssen Sie in der Regel Daten eingeben.

Um diese Daten zu sichern, können Sie Betriebssystembenutzer mit Warteschlangen konfigurieren. Der Worker-Agent verwendet den Warteschlangen-OS-Benutzer, um Sitzungsunterprozesse auszuführen. Diese Unterprozesse erben die Berechtigungen des Queue-OS-Benutzers.

Wir empfehlen, dass Sie sich an bewährte Methoden halten, um den Zugriff auf die Daten, auf die diese Unterprozesse zugreifen, zu sichern. Weitere Informationen finden Sie unter [Modell der geteilten Verantwortung](#).

Struktur der Farm

Sie können Deadline Cloud-Flotten und Warteschlangen auf viele Arten anordnen. Bestimmte Vereinbarungen haben jedoch Auswirkungen auf die Sicherheit.

Eine Farm hat eine der sichersten Grenzen, da sie Deadline Cloud-Ressourcen nicht mit anderen Farmen teilen kann, einschließlich Flotten, Warteschlangen und Speicherprofilen. Sie können jedoch externe AWS Ressourcen innerhalb einer Farm gemeinsam nutzen, wodurch die Sicherheitsgrenze gefährdet wird.

Mit der entsprechenden Konfiguration können Sie auch Sicherheitsgrenzen zwischen Warteschlangen innerhalb derselben Farm einrichten.

Folgen Sie diesen bewährten Methoden, um sichere Warteschlangen in derselben Farm zu erstellen:

- Ordnen Sie eine Flotte nur Warteschlangen innerhalb derselben Sicherheitsgrenze zu. Beachten Sie Folgendes:
 - Nachdem der Job auf dem Worker-Host ausgeführt wurde, können Daten zurückbleiben, z. B. in einem temporären Verzeichnis oder im Home-Verzeichnis des Warteschlangenbenutzers.
 - Derselbe Betriebssystembenutzer führt alle Jobs auf einem firmeneigenen Fleet-Worker-Host aus, unabhängig davon, an welche Warteschlange Sie den Job senden.
 - Ein Job kann dazu führen, dass Prozesse auf einem Worker-Host ausgeführt werden, sodass Jobs aus anderen Warteschlangen andere laufende Prozesse beobachten können.
- Stellen Sie sicher, dass sich nur Warteschlangen innerhalb derselben Sicherheitsgrenze einen Amazon S3 S3-Bucket für Jobanhänge teilen.
- Stellen Sie sicher, dass nur Warteschlangen innerhalb derselben Sicherheitsgrenze denselben Betriebssystembenutzer verwenden.
- Sichern Sie alle anderen AWS Ressourcen, die in die Farm integriert sind, bis zur Grenze.

Warteschlangen für Arbeitsanhänge

Jobanhänge sind mit einer Warteschlange verknüpft, die Ihren Amazon S3 S3-Bucket verwendet.

- Auftragsanhänge schreiben in ein Root-Präfix im Amazon S3 S3-Bucket und lesen aus diesem. Sie geben dieses Root-Präfix im `CreateQueue` API-Aufruf an.
- Der Bucket hat ein entsprechendes `QueueRole`, das die Rolle spezifiziert, die Warteschlangenbenutzern Zugriff auf den Bucket und das Root-Präfix gewährt. Beim Erstellen einer Warteschlange geben Sie den `QueueRole` Amazon-Ressourcennamen (ARN) zusammen mit dem Bucket und dem Root-Präfix für Jobanhänge an.
- Autorisierte Aufrufe von `AssumeQueueRoleForRead`, `AssumeQueueRoleForUser`, und `AssumeQueueRoleForWorker` API-Operationen geben einen Satz temporärer Sicherheitsanmeldedaten für die `QueueRole`.

Wenn Sie eine Warteschlange erstellen und einen Amazon S3 S3-Bucket und ein Root-Präfix wiederverwenden, besteht die Gefahr, dass Informationen an Unbefugte weitergegeben werden. `QueueA` und `QueueB` verwenden beispielsweise denselben Bucket und dasselbe Root-Präfix. In einem sicheren Workflow hat `ArtistA` Zugriff auf `QueueA`, aber nicht auf `QueueB`. Wenn sich jedoch mehrere Warteschlangen einen Bucket teilen, kann `ArtistA` auf die Daten in `QueueB`-Daten zugreifen, da es denselben Bucket und dasselbe Root-Präfix wie `QueueA` verwendet.

Die Konsole richtet Warteschlangen ein, die standardmäßig sicher sind. Stellen Sie sicher, dass die Warteschlangen eine eindeutige Kombination aus Amazon S3 S3-Bucket und Root-Präfix haben, sofern sie nicht Teil einer gemeinsamen Sicherheitsgrenze sind.

Um Ihre Warteschlangen zu isolieren, müssen Sie das so konfigurieren, Queue Role dass nur der Warteschlangenzugriff auf den Bucket und das Root-Präfix zulässig ist. Ersetzen Sie im folgenden Beispiel jeden *Platzhalter* durch Ihre ressourcenspezifischen Informationen.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::JOB_ATTACHMENTS_BUCKET_NAME",
        "arn:aws:s3:::JOB_ATTACHMENTS_BUCKET_NAME/JOB_ATTACHMENTS_ROOT_PREFIX/*"
      ],
      "Condition": {
        "StringEquals": { "aws:ResourceAccount": "ACCOUNT_ID" }
      }
    },
    {
      "Action": ["logs:GetLogEvents"],
      "Effect": "Allow",
      "Resource": "arn:aws:logs:REGION:ACCOUNT_ID:log-group:/aws/deadline/FARM_ID/*"
    }
  ]
}
```

Sie müssen auch eine Vertrauensrichtlinie für die Rolle festlegen. Ersetzen Sie im folgenden Beispiel den *Platzhaltertext* durch Ihre ressourcenspezifischen Informationen.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```

    "Action": ["sts:AssumeRole"],
    "Effect": "Allow",
    "Principal": { "Service": "deadline.amazonaws.com" },
    "Condition": {
      "StringEquals": { "aws:SourceAccount": "ACCOUNT_ID" },
      "ArnEquals": {
        "aws:SourceArn": "arn:aws:deadline:REGION:ACCOUNT_ID:farm/FARM_ID"
      }
    }
  },
  {
    "Action": ["sts:AssumeRole"],
    "Effect": "Allow",
    "Principal": { "Service": "credentials.deadline.amazonaws.com" },
    "Condition": {
      "StringEquals": { "aws:SourceAccount": "ACCOUNT_ID" },
      "ArnEquals": {
        "aws:SourceArn": "arn:aws:deadline:REGION:ACCOUNT_ID:farm/FARM_ID"
      }
    }
  }
]
}

```

Amazon S3 S3-Buckets mit benutzerdefinierter Software

Sie können die folgende Anweisung zu Ihrem hinzufügen, Queue Role um auf benutzerdefinierte Software in Ihrem Amazon S3 S3-Bucket zuzugreifen. Ersetzen Sie im folgenden Beispiel *SOFTWARE_BUCKET_NAME* durch den Namen Ihres S3-Buckets.

```

"Statement": [
  {
    "Action": [
      "s3:GetObject",
      "s3:ListBucket"
    ],
    "Effect": "Allow",
    "Resource": [
      "arn:aws:s3::SOFTWARE_BUCKET_NAME",
      "arn:aws:s3::SOFTWARE_BUCKET_NAME/*"
    ]
  }
]

```

]

Weitere Informationen zu den bewährten Sicherheitsmethoden von Amazon S3 finden Sie unter [Bewährte Sicherheitsmethoden für Amazon S3](#) im Amazon Simple Storage Service-Benutzerhandbuch.

Worker-Hosts

Schützen Sie Worker-Hosts, um sicherzustellen, dass jeder Benutzer nur Operationen für die ihm zugewiesene Rolle ausführen kann.

Wir empfehlen die folgenden bewährten Methoden zur Sicherung von Worker-Hosts:

- Verwenden Sie nicht denselben `jobRunAsUser` Wert für mehrere Warteschlangen, es sei denn, an diese Warteschlangen übermittelte Jobs liegen innerhalb derselben Sicherheitsgrenze.
- Stellen Sie die Warteschlange nicht `jobRunAsUser` auf den Namen des Betriebssystembenutzers ein, unter dem der Worker-Agent ausgeführt wird.
- Gewähren Sie Warteschlangenbenutzern die Betriebssystemberechtigungen mit den geringsten Rechten, die für die vorgesehenen Warteschlangenworkloads erforderlich sind. Stellen Sie sicher, dass sie keine Dateisystem-Schreibberechtigungen für Work-Agent-Programmdateien oder andere gemeinsam genutzte Software haben.
- Stellen Sie sicher, dass nur der Root-Benutzer Linux und das Konto `Administrator` Eigentümer der Worker-Agent-Programmdateien sind und diese ändern können. Windows
- Auf Linux Worker-Hosts sollten Sie erwägen, einen `umask` Override-Vorgang zu konfigurieren/`etc/sudoers`, der es dem Worker-Agent-Benutzer ermöglicht, Prozesse als Warteschlangenbenutzer zu starten. Diese Konfiguration trägt dazu bei, dass andere Benutzer nicht auf Dateien zugreifen können, die in die Warteschlange geschrieben wurden.
- Gewähren Sie vertrauenswürdigen Personen den Zugriff auf Worker-Hosts mit den geringsten Rechten.
- Beschränken Sie die Berechtigungen auf lokale DNS-Override-Konfigurationsdateien (aktiviert Linux und aktiviert) sowie `/etc/hosts` `C:\Windows\system32\etc\hosts` auf Windows Routing-Tabellen auf Workstations und Worker-Host-Betriebssystemen.
- Beschränken Sie die Berechtigungen für die DNS-Konfiguration auf Workstations und Worker-Host-Betriebssystemen.

- Patchen Sie regelmäßig das Betriebssystem und die gesamte installierte Software. Dieser Ansatz umfasst Software, die speziell mit Deadline Cloud verwendet wird, wie z. B. Einreicher, Adapter, Worker Agents, OpenJD Pakete und andere.
- Verwenden Sie sichere Passwörter für die Warteschlange. Windows `jobRunAsUser`
- Wechseln Sie regelmäßig die Passwörter für Ihre Warteschlange `jobRunAsUser`.
- Sorgen Sie für den geringsten Zugriff auf die Windows Kennwortgeheimnisse und löschen Sie ungenutzte Geheimnisse.
- Erteilen Sie der Warteschlange nicht die `jobRunAsUser` Erlaubnis, die Befehle für den Zeitplan in der future auszuführen:
 - EinLinux, verweigern Sie diesen Konten den Zugriff auf `cron` und `at`.
 - Ist diese Windows Option aktiviert, wird diesen Konten der Zugriff auf den Windows Taskplaner verweigert.

Note

Weitere Informationen darüber, wie wichtig es ist, das Betriebssystem und die installierte Software regelmäßig zu patchen, finden Sie im Modell der [gemeinsamen Verantwortung](#).

Workstations

Es ist wichtig, Workstations mit Zugriff auf Deadline Cloud zu sichern. Dieser Ansatz trägt dazu bei, dass Jobs, die Sie an Deadline Cloud einreichen, keine beliebigen Workloads ausführen können, die Ihnen in Rechnung gestellt werden. AWS-Konto

Wir empfehlen die folgenden bewährten Methoden zur Sicherung von Künstler-Workstations. Weitere Informationen finden Sie unter [-Modell der geteilten Verantwortung](#).

- Sichern Sie alle dauerhaften Anmeldeinformationen, die Zugriff auf, einschließlich Deadline AWS Cloud, ermöglichen. Weitere Informationen finden Sie unter [Verwalten der Zugriffsschlüssel für IAM-Benutzer](#) im -IAM-Benutzerhandbuch.
- Installieren Sie nur vertrauenswürdige, sichere Software.
- Erfordern Sie, dass Benutzer sich mit einem Identitätsanbieter zusammenschließen, um AWS mit temporären Anmeldeinformationen zugreifen zu können.

- Verwenden Sie sichere Berechtigungen für die Programmdateien von Deadline Cloud-Absendern, um Manipulationen zu verhindern.
- Gewähren Sie vertrauenswürdigen Personen den Zugriff auf die Workstations von Künstlern mit den geringsten Rechten.
- Verwenden Sie nur Einreicher und Adapter, die Sie über den Deadline Cloud Monitor erhalten.
- Beschränken Sie die Berechtigungen für Workstations `/etc/hosts` und Worker-Host-Betriebssysteme und leiten Sie Tabellen weiter.
- Beschränken Sie die Berechtigungen `/etc/resolv.conf` auf Workstations und Worker-Host-Betriebssystemen.
- Patchen Sie regelmäßig das Betriebssystem und die gesamte installierte Software. Dieser Ansatz umfasst Software, die speziell mit Deadline Cloud verwendet wird, wie z. B. Einreicher, Adapter, Worker Agents, OpenJD Pakete und andere.

AWS Deadline Cloud überwachen

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Zuverlässigkeit, Verfügbarkeit und Leistung von AWS Deadline Cloud (Deadline Cloud) und Ihrer AWS Lösungen. Sammeln Sie Überwachungsdaten aus allen Teilen Ihrer AWS Lösung, damit Sie einen Fehler an mehreren Stellen leichter debuggen können, falls einer auftritt. Bevor Sie mit der Überwachung von Deadline Cloud beginnen, sollten Sie einen Überwachungsplan erstellen, der Antworten auf die folgenden Fragen enthält:

- Was sind Ihre Überwachungsziele?
- Welche Ressourcen möchten Sie überwachen?
- Wie oft werden diese Ressourcen überwacht?
- Welche Überwachungs-Tools möchten Sie verwenden?
- Wer soll die Überwachungsaufgaben ausführen?
- Wer soll benachrichtigt werden, wenn Fehler auftreten?

AWS und Deadline Cloud bieten Tools, mit denen Sie Ihre Ressourcen überwachen und auf potenzielle Vorfälle reagieren können. Einige dieser Tools übernehmen die Überwachung für Sie, andere Tools erfordern manuelles Eingreifen. Sie sollten die Überwachungsaufgaben so weit wie möglich automatisieren.

- Amazon CloudWatch überwacht Ihre AWS Ressourcen und die Anwendungen, auf denen Sie laufen, AWS in Echtzeit. Sie können Kennzahlen erfassen und verfolgen, benutzerdefinierte Dashboards erstellen und Alarme festlegen, die Sie benachrichtigen oder Maßnahmen ergreifen, wenn eine bestimmte Metrik einen von Ihnen festgelegten Schwellenwert erreicht. Sie können beispielsweise die CPU-Auslastung oder andere Kennzahlen Ihrer Amazon EC2 EC2-Instances CloudWatch verfolgen und bei Bedarf automatisch neue Instances starten. Weitere Informationen finden Sie im [CloudWatch Amazon-Benutzerhandbuch](#).

Deadline Cloud hat drei CloudWatch Metriken.

- Mit Amazon CloudWatch Logs können Sie Ihre Protokolldateien von Amazon EC2 EC2-Instances und anderen Quellen überwachen CloudTrail, speichern und darauf zugreifen. CloudWatch Logs können Informationen in den Protokolldateien überwachen und Sie benachrichtigen, wenn bestimmte Schwellenwerte erreicht werden. Sie können Ihre Protokolldaten auch in einem sehr

robusten Speicher archivieren. Weitere Informationen finden Sie im [Amazon CloudWatch Logs-Benutzerhandbuch](#).

- Amazon EventBridge kann verwendet werden, um Ihre AWS Services zu automatisieren und automatisch auf Systemereignisse wie Probleme mit der Anwendungsverfügbarkeit oder Ressourcenänderungen zu reagieren. Ereignisse aus AWS Services werden nahezu EventBridge in Echtzeit zugestellt. Sie können einfache Regeln schreiben, um anzugeben, welche Ereignisse für Sie interessant sind und welche automatisierten Aktionen ausgeführt werden sollen, wenn ein Ereignis mit einer Regel übereinstimmt. Weitere Informationen finden Sie im [EventBridge Amazon-Benutzerhandbuch](#).
- AWS CloudTrail erfasst API-Aufrufe und zugehörige Ereignisse, die von oder im Namen Ihres AWS Kontos getätigt wurden, und übermittelt die Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket. Sie können feststellen, welche Benutzer und Konten angerufen wurden AWS, von welcher Quell-IP-Adresse aus die Aufrufe getätigt wurden und wann die Aufrufe erfolgten. Weitere Informationen finden Sie im [AWS CloudTrail -Benutzerhandbuch](#).

Themen

- [Anrufe protokollieren mit CloudTrail](#)
- [Überwachung mit CloudWatch](#)
- [Auf EventBridge Ereignisse reagieren](#)

Anrufe protokollieren mit CloudTrail

AWS Deadline Cloud ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der von einem Benutzer, einer Rolle oder einer AWS-Service in Deadline Cloud ausgeführten Aktionen bereitstellt. CloudTrail erfasst alle API-Aufrufe für Deadline Cloud als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der Deadline Cloud-Konsole und Code-Aufrufe der Deadline Cloud-API-Operationen.

Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Bereitstellung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für Deadline Cloud. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Event-Verlauf einsehen. Anhand der von gesammelten Informationen können Sie die Anfrage CloudTrail, die an Deadline Cloud gestellt wurde, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details ermitteln.

Weitere Informationen CloudTrail dazu finden Sie im [AWS CloudTrail Benutzerhandbuch](#).

Deadline Cloud-Informationen finden Sie unter CloudTrail

CloudTrail ist auf Ihrem aktiviert AWS-Konto , wenn Sie das Konto erstellen. Wenn eine Aktivität in Deadline Cloud stattfindet, wird diese Aktivität zusammen mit anderen AWS-Service Ereignissen im CloudTrail Event-Verlauf in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem anzeigen, suchen und herunterladen AWS-Konto. Weitere Informationen finden Sie unter [Ereignisse mit CloudTrail Ereignisverlauf anzeigen](#).

CloudTrail zeichnet auch Ereignisse auf, wenn sich Benutzer beim Deadline Cloud-Monitor anmelden und AWS Anmeldeinformationen erhalten. Wenn sich ein Benutzer anmeldet, gibt es ein CloudTrail Ereignis mit der Quelle `signin.amazonaws.com` und dem Namen `UserAuthentication`. Es gibt ein zweites Ereignis, wenn der angemeldete Benutzer AWS Anmeldeinformationen aus der Quelle `sts.amazonaws.com` und dem Namen `AssumeRole` erhält. Die Benutzer-ID wird im zweiten Ereignis im Namen der Rollensitzung aufgezeichnet.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrer Datenbank AWS-Konto, einschließlich der Ereignisse für Deadline Cloud, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Wenn Sie einen Trail in der Konsole anlegen, gilt dieser für alle AWS-Regionen-Regionen. Der Trail protokolliert Ereignisse aus allen Regionen der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen Amazon S3 S3-Bucket. Darüber hinaus können Sie andere konfigurieren, AWS-Services um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren.

Weitere Informationen finden Sie hier:

[Übersicht zum Erstellen eines Trails](#)

[CloudTrail unterstützte Dienste und Integrationen](#)

[Konfiguration von Amazon SNS SNS-Benachrichtigungen für CloudTrail](#)

[Empfangen von CloudTrail Protokolldateien aus mehreren Regionen](#)

[Empfangen von CloudTrail Protokolldateien von mehreren Konten](#)

Deadline Cloud unterstützt die Protokollierung der folgenden Aktionen als Ereignisse in CloudTrail Protokolldateien:

- [associate-member-to-farm](#)
- [associate-member-to-fleet](#)
- [associate-member-to-job](#)
- [associate-member-to-queue](#)
- [assume-fleet-role-for-gelesen](#)
- [assume-fleet-role-for-Arbeiter](#)
- [assume-queue-role-for-gelesen](#)
- [assume-queue-role-for-Benutzer](#)
- [assume-queue-role-for-Arbeiter](#)
- [Budget erstellen](#)
- [Farm erstellen](#)
- [create-fleet](#)
- [create-license-endpoint](#)
- [Monitor erstellen](#)
- [Warteschlange erstellen](#)
- [create-queue-environment](#)
- [create-queue-fleet-association](#)
- [create-storage-profile](#)
- [Mitarbeiter erstellen](#)
- [Budget löschen](#)
- [Farm löschen](#)
- [delete-fleet](#)
- [delete-license-endpoint](#)
- [delete-metered-product](#)
- [Monitor löschen](#)
- [Warteschlange löschen](#)
- [delete-queue-environment](#)
- [delete-queue-fleet-association](#)

- [delete-storage-profile](#)
- [Mitarbeiter löschen](#)
- [disassociate-member-from-farm](#)
- [disassociate-member-from-fleet](#)
- [disassociate-member-from-job](#)
- [disassociate-member-from-queue](#)
- [get-application-version](#)
- [Holen Sie sich ein Budget](#)
- [Farm abrufen](#)
- [get-feature-map](#)
- [Holen Sie sich die Flotte](#)
- [get-license-endpoint](#)
- [Holen Sie sich einen Monitor](#)
- [Warteschlange abrufen](#)
- [get-queue-environment](#)
- [get-queue-fleet-association](#)
- [get-sessions-statistics-aggregation](#)
- [get-storage-profile](#)
- [get-storage-profile-for-Warteschlange](#)
- [list-available-metered-products](#)
- [Budgets auflisten](#)
- [list-farm-members](#)
- [Farmen auflisten](#)
- [list-fleet-members](#)
- [Flotten auflisten](#)
- [list-job-members](#)
- [list-license-endpoints](#)
- [list-metered-products](#)

- [Monitore auflisten](#)
- [list-queue-environments](#)
- [list-queue-fleet-associations](#)
- [list-queue-members](#)
- [Listen-Warteschlangen](#)
- [list-storage-profiles](#)
- [list-storage-profiles-for-Warteschlange](#)
- [list-tags-for-resource](#)
- [put-metered-product](#)
- [start-sessions-statistics-aggregation](#)
- [tag-resource](#)
- [untag-resource](#)
- [Budget aktualisieren](#)
- [Farm aktualisieren](#)
- [Flotte aktualisieren](#)
- [aktualisiere den Monitor](#)
- [Aktualisierungswarteschlange](#)
- [update-queue-environment](#)
- [update-queue-fleet-association](#)
- [update-storage-profile](#)
- [Aktualisiere Worker](#)

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 Root- oder AWS Identity and Access Management (IAM-) Benutzeranmeldedaten gestellt wurde.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anforderung aus einem anderen -Service gesendet wurde

Weitere Informationen finden Sie unter dem [Element CloudTrail Benutzeridentität](#).

Grundlegendes zu Deadline Cloud-Protokolldateieinträgen

Ein Trail ist eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere Protokolleinträge. Ein Ereignis stellt eine einzelne Anforderung aus einer beliebigen Quelle dar und enthält Informationen über die angeforderte Aktion, Datum und Uhrzeit der Aktion, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass sie nicht in einer bestimmten Reihenfolge angezeigt werden.

Dieses JSON-Beispiel zeigt das Protokoll, das durch einen **CreateFarm** API-Aufruf generiert wurde:

```
{
  "eventVersion": "0",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE-PrincipalID:EXAMPLE-Session",
    "arn": "arn:aws:sts::111122223333:assumed-role/EXAMPLE-UserName/EXAMPLE-Session",
    "accountId": "111122223333",
    "accessKeyId": "EXAMPLE-accessKeyId",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EXAMPLE-PrincipalID",
        "arn": "arn:aws:iam::111122223333:role/EXAMPLE-UserName",
        "accountId": "111122223333",
        "userName": "EXAMPLE-UserName"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-03-08T23:25:49Z"
      }
    }
  },
  "eventTime": "2021-03-08T23:25:49Z",
  "eventSource": "deadline.amazonaws.com",
  "eventName": "CreateFarm",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "EXAMPLE-userAgent",
```

```
"requestParameters": {
  "displayName": "example-farm",
  "kmsKeyArn": "arn:aws:kms:us-west-2:111122223333:key/111122223333",
  "X-Amz-Client-Token": "12abc12a-1234-1abc-123a-1a11bc1111a",
  "description": "example-description",
  "tags": {
    "purpose_1": "e2e"
    "purpose_2": "tag_test"
  }
},
"responseElements": {
  "farmId": "EXAMPLE-farmID"
},
"requestID": "EXAMPLE-requestID",
"eventID": "EXAMPLE-eventID",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333"
"eventCategory": "Management",
}
```

Das Beispiel zeigt die AWS Region, die IP-Adresse und andere "", wie requestParameters "" und "displayName", die Ihnen bei der Identifizierung des Ereignisses helfen können. kmsKeyArn

Überwachung mit CloudWatch

Amazon CloudWatch (CloudWatch) sammelt Rohdaten und verarbeitet sie zu lesbaren, nahezu in Echtzeit verfügbaren Metriken. Sie können die CloudWatch Konsole unter <https://console.aws.amazon.com/cloudwatch/> öffnen, um Deadline Cloud-Metriken anzusehen und zu filtern.

- In einer vom Kunden verwalteten Flotte von Deadline Cloud werden Ihnen zwei Messwerte CloudWatch gesendet UnhealthyWorkerCount undRecommendedFleetSize:
- Der Namespace für diese Metriken lautet AWS/DeadlineCloud.
- Sie können die Dimensionen farmID und fleetID zum Filtern von Metriken verwenden.
- Beide Metriken verwenden die Einheitcount.

Diese Statistiken werden 15 Monate lang aufbewahrt, sodass Sie auf historische Informationen zugreifen können, um einen besseren Überblick über die Leistung Ihrer Webanwendung oder Ihres

Dienstes zu erhalten. Sie können auch Alarme einrichten, die auf bestimmte Grenzwerte achten und Benachrichtigungen senden oder Aktivitäten auslösen, wenn diese Grenzwerte erreicht werden. Weitere Informationen finden Sie im [CloudWatch Amazon-Benutzerhandbuch](#).

Deadline Cloud hat zwei Arten von Protokollen — Aufgabenprotokolle und Worker-Logs. Ein Aufgabenprotokoll liegt vor, wenn Sie Ausführungsprotokolle als Skript oder während der Ausführung von DCC ausführen. In einem Aufgabenprotokoll können Ereignisse wie das Laden von Objekten, das Rendern von Kacheln oder das Nichtauffinden von Texturen angezeigt werden.

In einem Worker-Protokoll werden die Prozesse des Worker-Agents aufgeführt. Dazu können Dinge gehören, z. B. wann der Worker-Agent startet, sich selbst registriert, Fortschritte meldet, Konfigurationen lädt oder Aufgaben abschließt.

Bei Deadline Cloud laden Mitarbeiter diese CloudWatch Protokolle in Logs hoch. Standardmäßig laufen Protokolle nie ab. Wenn bei einem Auftrag eine große Datenmenge ausgegeben wird, können zusätzliche Kosten anfallen. Weitere Informationen finden Sie unter [CloudWatch Amazon-Preise](#).

Sie können die Aufbewahrungsrichtlinie für jede Protokollgruppe anpassen. Eine kürzere Aufbewahrung entfernt alte Protokolle und kann zur Senkung der Speicherkosten beitragen. Um Protokolle aufzubewahren, können Sie sie in Amazon Simple Storage Service archivieren, bevor Sie das Protokoll entfernen. Weitere Informationen finden Sie unter [Exportieren von Protokolldaten nach Amazon S3 mithilfe der Konsole](#) im CloudWatch Amazon-Benutzerhandbuch.

Note

CloudWatch Das Lesen von Protokollen ist begrenzt durch AWS. Wenn Sie planen, viele Künstler an Bord zu nehmen, empfehlen wir Ihnen, sich an den AWS Kundensupport zu wenden und eine Erhöhung des `GetLogEvents` Kontingents in zu beantragen CloudWatch. Außerdem empfehlen wir Ihnen, das Log-Tailing-Portal zu schließen, wenn Sie nicht debuggen.

Weitere Informationen finden Sie unter [CloudWatch Log-Kontingente](#) im CloudWatch Amazon-Benutzerhandbuch.

Auf EventBridge Ereignisse reagieren

Deadline Cloud sendet Ereignisse an Amazon EventBridge , um Sie über Änderungen am Status des Dienstes zu informieren. Sie können diese Ereignisse verwenden EventBridge , um Regeln zu

schreiben, die Maßnahmen ergreifen, z. B. Sie benachrichtigen, wenn sich Ihre Flotte ändert. Weitere Informationen finden Sie unter [Was ist Amazon EventBridge](#)

Änderung der Empfehlung zur Flottengröße

Wenn Sie Ihre Flotte so konfigurieren, dass sie ereignisbasiertes Auto Scaling verwendet, sendet Deadline Cloud Ereignisse, mit denen Sie Ihre Flotten verwalten können. Jedes dieser Ereignisse enthält Informationen über die aktuelle Größe und die angeforderte Größe einer Flotte. Ein Beispiel für die Verwendung eines EventBridge Ereignisses und eine Lambda-Beispielfunktion zur Behandlung des Ereignisses finden Sie unter [Skalieren Sie Ihre Amazon EC2 EC2-Flotte automatisch mit der Deadline Cloud-Funktion für Skalierungsempfehlungen](#).

Das Ereignis zur Änderung der Flottengrößenempfehlung wird gesendet, wenn Folgendes eintritt:

- Wenn sich die empfohlene Flottengröße ändert und `oldFleetSize` sich von `unterscheidetnewFleetSize`.
- Wenn der Service feststellt, dass die tatsächliche Flottengröße nicht der empfohlenen Flottengröße entspricht. Die tatsächliche Flottengröße können Sie `workerCount` der [GetFleet](#) Betriebsantwort entnehmen. Dies kann passieren, wenn sich eine aktive Amazon EC2 EC2-Instance nicht als Deadline Cloud-Worker registrieren kann.

Die Veranstaltung hat das folgende Format:

```
{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "Fleet Size Recommendation Change",
  "source": "aws.deadline",
  "account": "111122223333",
  "time": "2017-12-22T18:43:48Z",
  "region": "us-west-1",
  "resources": [],
  "detail": {
    "farmId": "farm-12345678900000000000000000000000",
    "fleetId": "fleet-12345678900000000000000000000000",
    "oldFleetSize": 1,
    "newFleetSize": 5,
  }
}
```

Die folgenden Felder definieren das Ereignismuster:

```
"source": "aws.deadline"
```

Identifiziert, dass die Quelle dieses Ereignisses Deadline Cloud ist.

```
"detail-type": "Fleet Size Recommendation Change"
```

Identifiziert den Ereignistyp.

```
"detail": { }
```

Enthält Informationen zu den empfohlenen Änderungen der Flottengröße.

```
"farmId": "farm-1234567890000000000000000000000000"
```

Die Kennung der Farm, in der sich die Flotte befindet.

```
"fleetId": "fleet-1234567890000000000000000000000000"
```

Die Kennung der Flotte, deren Größe geändert werden muss.

```
"oldFleetSize": 1
```

Die aktuelle Größe der Flotte.

```
"newFleetSize": 5
```

Die empfohlene neue Größe der Flotte.

Kontingente für Deadline Cloud

AWS Deadline Cloud stellt Ressourcen wie Farmen, Flotten und Warteschlangen bereit, die Sie zur Verarbeitung von Aufträgen verwenden können. Wenn Sie Ihr erstes AWS-Konto erstellen, legen wir für jede Ressource Standardkontingente für diese Ressourcen fest. AWS-Region

Service Quotas ist ein zentraler Ort, an dem Sie Ihre Kontingente für anzeigen und verwalten können AWS-Services. Sie können auch eine Erhöhung des Kontingents für viele der von Ihnen genutzten Ressourcen beantragen.

Um die Kontingente für anzuzeigen Deadline Cloud, öffnen Sie die [Konsole Service Quotas](#). Wählen Sie im Navigationsbereich AWS-Services und anschließend Deadline Cloud aus.

Informationen zur Erhöhung eines Kontingents finden Sie unter [Anfordern einer Kontingenterhöhung](#) im Service-Quotas-Benutzerhandbuch. Wenn das Kontingent noch nicht unter Servicekontingente verfügbar ist, verwenden Sie das [Formular zur Erhöhung der Service Quotas](#).

AWS Deadline Cloud-Ressourcen erstellen mit AWS CloudFormation

AWS Deadline Cloud ist integriert mit AWS CloudFormation, ein Service, der Ihnen hilft, Ihre AWS Ressourcen zu modellieren und einzurichten, sodass Sie weniger Zeit mit der Erstellung und Verwaltung Ihrer Ressourcen und Infrastruktur verbringen müssen. Sie erstellen eine Vorlage, die alle gewünschten AWS Ressourcen beschreibt (wie Farmen, Warteschlangen und Flotten) und diese Ressourcen für Sie AWS CloudFormation bereitstellt und konfiguriert.

Wenn Sie sie verwenden AWS CloudFormation, können Sie Ihre Vorlage wiederverwenden, um Ihre Deadline Cloud-Ressourcen konsistent und wiederholt einzurichten. Beschreiben Sie Ihre Ressourcen einmal und stellen Sie dann dieselben Ressourcen immer wieder in mehreren AWS-Konten Regionen bereit.

Deadline Cloud und AWS CloudFormation Vorlagen

Um Ressourcen für Deadline Cloud und verwandte Dienste bereitzustellen und zu konfigurieren, müssen Sie [AWS CloudFormation Vorlagen](#) verstehen. Vorlagen sind formatierte Textdateien in JSON oder YAML. Diese Vorlagen beschreiben die Ressourcen, die Sie in Ihren AWS CloudFormation Stacks bereitstellen möchten. Wenn Sie mit JSON oder YAML nicht vertraut sind, können Sie AWS CloudFormation Designer verwenden, um Ihnen die ersten Schritte mit Vorlagen zu erleichtern. Weitere Informationen finden Sie unter [Was ist AWS CloudFormation -Designer?](#) im AWS CloudFormation -Benutzerhandbuch.

Deadline Cloud unterstützt das Erstellen von Farmen, Warteschlangen und Flotten in. AWS CloudFormation Weitere Informationen, einschließlich Beispielen für JSON- und YAML-Vorlagen für Farmen, Warteschlangen und Flotten, finden Sie in der [AWS Deadline Cloud im Benutzerhandbuch](#). AWS CloudFormation

Erfahren Sie mehr über AWS CloudFormation

Weitere Informationen AWS CloudFormation dazu finden Sie in den folgenden Ressourcen:

- [AWS CloudFormation](#)
- [AWS CloudFormation Benutzerhandbuch](#)
- [AWS CloudFormation API Reference](#)

- [AWS CloudFormation Benutzerhandbuch für die Befehlszeilenschnittstelle](#)

Dokumentenverlauf für das Deadline Cloud- Benutzerhandbuch

In der folgenden Tabelle werden wichtige Änderungen in jeder Version des AWS Deadline Cloud-Benutzerhandbuchs beschrieben.

Änderung	Beschreibung	Datum
Erstversion	Dies ist die erste Version des Deadline Cloud-Benutzerhandbuchs.	2. April 2024

AWS Glossar

Die neueste AWS Terminologie finden Sie im [AWS Glossar](#) in der AWS-Glossar Referenz.

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.