



Implementierung der Infrastruktur als Produkt (IAP) auf AWS

# AWS Präskriptive Leitlinien



# AWS Präskriptive Leitlinien: Implementierung der Infrastruktur als Produkt (IAP) auf AWS

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 Marken, die nicht im Besitz von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise mit Amazon verbunden sind oder von Amazon gesponsert werden.

# Table of Contents

Einführung .....	1
Warum Infrastruktur als Produkte verwalten? .....	1
Gezielte Geschäftsergebnisse .....	1
Verwendung AWS Service Catalog zur Verwaltung von iAP .....	3
Support für Modularität und Wiederverwendung von Code .....	4
Programmierungsoptionen für die Definition von Produkten im Service Catalog .....	5
CloudFormation Skripting .....	6
Programmatischer Ansatz mit dem AWS CDK .....	6
Integration mit externen Bereitstellungsprozessen und Workflows .....	7
Spezifikationen für die Produktbereitstellung .....	8
DevSecOps Lebenszyklus .....	8
Maßgeschneiderte Wiederverwendung und kontospezifische Bereitstellung .....	9
Definition und Verwaltung von Service Catalog-Produktressourcen als Anwendungen .....	9
Inventory .....	10
AWS Service Catalog Tools verwenden .....	11
Service Catalog Puppet .....	11
Support für die Bereitstellung von Workflows .....	12
Bereitstellungsmodi .....	12
Caching .....	14
DevSecOps Unterstützung für den Lebenszyklus Support .....	15
Reife, Vollständigkeit und Unterstützung .....	15
Service Catalog .....	16
Zusammenfassung und nächste Schritte .....	17
Ressourcen .....	18
Dokumentverlauf .....	19
Glossar .....	20
# .....	20
A .....	21
B .....	24
C .....	26
D .....	30
E .....	34
F .....	36
G .....	38

---

H .....	39
I .....	40
L .....	43
M .....	44
O .....	48
P .....	51
Q .....	54
R .....	54
S .....	57
T .....	61
U .....	62
V .....	63
W .....	64
Z .....	65
.....	lxvi

# Implementierung von IaP auf AWS

Kirsten Kissmeyer, Amazon Web Services (AWS)

Januar 2023 ([Dokumentengeschichte](#))

In diesem Leitfaden werden Ansätze für die Verwaltung Ihrer AWS Infrastruktur als Produkt (IaP) beschrieben. IaP bietet ein höheres Maß an Abstraktion und Kontrolle als Infrastructure as Code (IaC), verwendet jedoch IaC-Methoden, um seine Ziele zu erreichen. Der Leitfaden befasst sich auch mit Tools für die Verwaltung von IaP und zeigt auf, wie jedes Tool Ihre Ziele bei der Verwaltung Ihrer Infrastruktur unterstützen kann. Die Informationen in diesem Leitfaden basieren auf Erkenntnissen aus einer AWS Service Catalog Enablement-Initiative für ein sehr großes Unternehmen der Finanzbranche.

Dieses Handbuch richtet sich an Benutzer, die funktionale AWS Cloud Infrastrukturdienste entwickeln möchten, die nach Bedarf für verschiedene Unternehmensbenutzer, Geschäftsbereiche und Dritte einfach zugewiesen und autorisiert werden können.

## Warum Infrastruktur als Produkte verwalten?

Der Vorteil der Verwaltung Ihrer Infrastrukturressourcen als Produkte besteht darin, dass Sie die Funktionen Ihrer Nutzer in Form von Ressourcen mit standardisierten Definitionen und Konfigurationen zusammenfassen können. Produkte bieten Unternehmen eine bequeme Möglichkeit, zu verwalten und zu kontrollieren, wie AWS Kapazitäten zugewiesen und genutzt werden. Ein Produkt kann nur bestimmten [Organisationseinheiten \(OUs\) oder](#) Personen vorbehalten sein, die diese funktionalen Fähigkeiten benötigen. Ein Produkt kann auch auf bestimmte AWS-Regionen beschränkt werden.

Ein Produktbereitstellungsmodell ermöglicht es Ihnen auch, die Definition eines Produkts von einem zentralen Standort aus zu kapseln und zu aktualisieren. Anschließend können Sie Produktupdates einmalig oder geplant verteilen, da sich die Implementierung im Laufe der Zeit ändert.

## Gezielte Geschäftsergebnisse

Organizations suchen immer nach besseren Möglichkeiten, ihre AWS Infrastruktur zu verwalten und bereitzustellen. Zu Ihren Zielen könnten gehören:

- Erzielung eines hohen Maßes an Agilität, Zuverlässigkeit, Fehlertoleranz und zentraler Steuerung, bei der einzelne Konfigurationen die Einhaltung der sich entwickelnden internen und externen Standards gewährleisten.
- Ein Mechanismus, der auf Knopfdruck oder per Knopfdruck die Infrastruktur zentral verteilt und gleichzeitig den Self-Service-Zugriff für bestimmte Teams oder Einzelpersonen ermöglicht, wenn dies erforderlich ist.
- Die Fähigkeit, AWS Infrastruktur und Dienste für interne Mitarbeiter, Kundenkonten und OU-Konten von Partnern bereitzustellen. Möglicherweise möchten Sie auch kontrollieren, welche Organisationseinheiten oder Organisationen Zugriff auf bestimmte Infrastrukturkomponenten in bestimmten Regionen haben.
- Wenn Sie Tools von Drittanbietern (z. B. ServiceNow) oder benutzerdefinierte Tools verwenden, um Anfragen für den Zugriff auf und die Bereitstellung Ihrer Unternehmensressourcen und Infrastruktur zu verwalten, ist eine einfache Integration zwischen Ihrer AWS Infrastruktur und diesen Tools erforderlich.
- Die Fähigkeit, AWS Infrastruktur für Dutzende oder sogar Hunderte von Zielkonten gleichzeitig bereitzustellen.
- Support für die Bereitstellung mehrerer AWS Ressourcen zur Bereitstellung einer einzigen Funktion.
- Die Möglichkeit, innerhalb eines engen Zeitplans neue Konten mit der erforderlichen Infrastruktur zu erstellen.
- Zugriff auf ein Inventar der Infrastruktur, die Sie bereitgestellt haben, und die Möglichkeit, Infrastrukturkomponenten zu aktualisieren oder zu entfernen.
- Ansätze und Technologien, die den Bereitstellungs- und Wartungsprozess einfacher, schneller, sicherer und zuverlässiger machen.

# Verwendung AWS Service Catalog zur Verwaltung von iAP

AWS stellt einen Dienst namens bereit [AWS Service Catalog](#), der die Verwaltung und Bereitstellung der AWS Infrastruktur als Produkt unterstützt. Mit Service Catalog können Sie schnell die Infrastruktur definieren, die Sie für die Bereitstellung in Form einer Reihe von Produkten benötigen, gewünschten Parteien die Genehmigung für diese Produkte erteilen und die für einzelne Produkte erforderlichen Bereitstellungs- und Aktualisierungsmuster implementieren.

Der Service Catalog wird unterstützt von [AWS CloudFormation](#). Service Catalog-Portfolios, Produkte und ihre Bereitstellungsvorlagen werden als CloudFormation Stapel verwaltet. Sie können diese Stapel auf vier Arten definieren:

- Durch die Verwendung von CloudFormation Standardvorlagen.
- Indem Sie die [AWS Cloud Development Kit \(AWS CDK\)](#) und die [Service Catalog Construct Library](#) mit einer unterstützten Programmiersprache verwenden, die Sie bevorzugen.
- Durch die Verwendung eines Frameworks, das von einem Drittanbieter-Tool bereitgestellt wird, um die CloudFormation Stackdefinitionen aus deklarativen Metadaten zu generieren, die die Stacks beschreiben.
- Mithilfe der [Service Catalog API](#). Diese API bietet Methoden für alles außer für die Erstellung des Produkts. Sie können Produkte zu Portfolios hinzufügen, Produkte aus Portfolios entfernen, Produkte und Portfolios taggen, administrative und betriebliche Produktservice-Aktionen definieren sowie Portfolio- und Produktdefinitionen durchsuchen und danach suchen.

Im Kern besteht ein Service Catalog-Produkt aus einer oder mehreren AWS Ressourcen, die so konfiguriert sind, dass sie eine kollektive, anpassbare Funktion (durch Parametrisierung) bereitstellen. So können Sie zum Beispiel ein Service Catalog definieren, um einen privaten Amazon Simple Storage Service (Amazon S3) -Bucket in einem Ziel-Konto bereitzustellen. Der S3-Bucket ist ein Produkt, das Eingabeparameter wie den Bucket-Namen, einen Internet-Adressbereich für den Zugriff, eine Gruppe von Benutzern, die auf den Bucket zugreifen können, eine Lifecycle-Tiering-Richtlinie oder eine Bucket-Versionierungsspezifikation haben kann. Sie können auch eine AWS Identity and Access Management (IAM-) Rolle definieren, um den Zugriff auf den Bucket als Teil des Produkts zu ermöglichen.

Sie können ein Service Catalog-Produkt zu einem oder mehreren Portfolios hinzufügen. Ein Service Catalog-Portfolio ist eine Sammlung von Produkten, die in Gruppen zusammengefasst

sind, im Allgemeinen, weil sie einem ähnlichen Zweck dienen (z. B. Analyse, Entwicklung, Kundenzugriffsdienste, Partnerzugriffsdienste usw.).

Sie gewähren einem Benutzer, einer Gruppe oder einer Rolle die Zugriffsrechte für die Bereitstellung eines Produkts auf Portfolioebene. Für die Bereitstellung werden Produkte entweder mit einer Start-IAM-Rolle verknüpft (um das Produkt im Self-Service-Modus für jeden zu starten, der die Rolle übernehmen kann) oder einem [Stack-Set](#), das ein oder mehrere Konten definiert, für die das Produkt bereitgestellt werden kann. Um ein Stack-Set zu verwenden, müssen Sie eine Service Catalog-Administratorrolle im Service Catalog-Hub-Konto und eine Service Catalog-Produktbereitstellungsausführungsrolle in jedem Zielkonto des Stacksets definieren.

In den folgenden Abschnitten wird die Service Catalog iAP-Funktionalität ausführlicher beschrieben.

Themen

- [Support für Modularität und Wiederverwendung von Code](#)
- [Programmieroptionen für die Definition von Produkten im Service Catalog](#)
- [Integration mit externen Bereitstellungsprozessen und Workflows](#)
- [Spezifikationen für die Produktbereitstellung](#)
- [DevSecOps LebensCory](#)
- [Maßgeschneiderte Wiederverwendung und kontospezifische Bereitstellung](#)
- [Definition und Verwaltung von Service Catalog-Produktressourcen als Anwendungen](#)
- [Inventory](#)

## Support für Modularität und Wiederverwendung von Code

Sie können ein Produkt aus vielen verschiedenen AWS Ressourcen oder sogar aus anderen Produkten zusammenstellen. Idealerweise definieren Sie Ressourcen modular, sodass Sie sie in mehreren Produkten wiederverwenden können. Durch die Wiederverwendung auf Ressourcenebene können Sie future Änderungen an einem Ort vornehmen und nicht für jedes Produkt, das diesen Ressourcentyp verwendet.

Service Catalog bietet eine Funktion namens Chaining, um die Wiederverwendbarkeit auf Produktebene zu unterstützen. Sie können ein Produkt mit einem oder mehreren anderen Produkten verknüpfen. Beispielsweise möchten Sie vielleicht ein S3-Logging-Bucket-Produkt mit einem übergeordneten Überwachungsprodukt verknüpfen. Die Verkettung unterstützt zwar die Modularität, bringt jedoch einige betriebliche Komplexitäten mit sich, da Sie Abhängigkeiten verwalten müssen.



Service Catalog verwaltet nicht automatisch die Versionierung zwischen verketteten Produkten und kann daher nicht sicherstellen, dass Änderungen an einem Produkt keine Auswirkungen auf andere Produkte haben, die davon abhängen. Gehen Sie vorsichtig mit Verkettung um und entwickeln Sie Ihre eigenen Mechanismen, um die Versionierung sicherzustellen und Abhängigkeiten zu pflegen.

Service Catalog verwendet CloudFormation nativ, um eine Vorlage für die Produktbereitstellung als CloudFormation Stapel bereitzustellen. Der Service Catalog weist jedoch einige Einschränkungen bei der CloudFormation Bereitstellung des Produktstapels auf. Insbesondere unterstützt die Service Catalog-Bereitstellung das CloudFormation `include` Makro zum Einfügen wiederverwendbarer Skriptsegmente oder zum Verweisen auf verschachtelte CloudFormation Skripts (oder Stapel) auf mehr als einer Ebene nicht. Diese Servicekatalog-Einschränkungen schränken die Möglichkeit ein, Produkte anhand wiederverwendbarer CloudFormation Vorlagen oder Komponenten zu definieren. Dies ist eine bewährte Standardmethode, wenn Sie Stacks nativ in definieren CloudFormation.

#### Note

Mit Service Catalog können Sie erfolgreich Produkte mit Bereitstellungsvorlagen definieren, die diese CloudFormation Konstrukte verwenden. Wenn Sie das `include` Makro verwenden oder mehrere Skriptebenen in einer CloudFormation Servicekatalogvorlage verschachteln, treten jedoch Fehler bei der Bereitstellung auf.

Diese Einschränkungen könnten die Implementierung modularer und wiederverwendbarer Produkte im Service Catalog erschweren. Wenn Modularität eine Anforderung ist, könnten Sie [die verwenden](#), AWS CDK um Ihre Produkte und deren Bereitstellungsvorlagen zu implementieren, oder die Bereitstellungsworkflows und die Bereitstellungs-Engine im [Projekt AWS Labs Service Catalog Tools](#) verwenden. Beide Alternativen werden weiter in diesem Handbuch beschrieben.

## Programmieroptionen für die Definition von Produkten im Service Catalog

Zwei Programmieroptionen für die Verwendung von Service Catalog zur Bereitstellung der AWS Infrastruktur sind CloudFormation Vorlagen oder die AWS CDK. Derzeit gibt es keine deklarativen oder No-Code-Mechanismen zur Definition eines Service Catalog-Produkts.

## CloudFormation Skripting

AWS CloudFormation ist eine bewährte native IaC-Skriptsprache für die Bereitstellung der AWS Infrastruktur. Sie können ein CloudFormation Skript in der AWS Management Console oder entwickeln, indem Sie ein Entwicklungstool wie Visual Studio Code (oder einen einfachen Texteditor) und die AWS Command Line Interface (AWS CLI) verwenden.

Weitere Informationen finden Sie in der [CloudFormation-Dokumentation](#). Weitere Informationen zur Verwendung einer CloudFormation Vorlage zur Angabe eines Service Catalog-Produkts finden Sie in der CloudFormation Dokumentation in der [AWS::ServiceCatalog::CloudFormationProduktressource](#).

## Programmatischer Ansatz mit dem AWS CDK

Das AWS CDK bietet ein elegantes und leistungsstarkes objektorientiertes Programmierframework für die Definition und Wartung der AWS Infrastruktur unter Verwendung einer Auswahl von Programmiersprachen. Sie können das verwenden, AWS CDK um objektorientierte, feinkörnige Anpassungen und Erweiterungen des AWS Klassenrahmens zu entwickeln. Das AWS CDK richtet sich an Benutzer, die sich an anspruchsvollere Infrastrukturanforderungen anpassen AWS-Services möchten und über die erforderlichen Programmierkenntnisse und -erfahrung verfügen.

Um Service Catalog-Lösungen mithilfe von zu implementieren AWS CDK, verwenden Sie die integrierten Service Catalog-Klassen, um Ihre Produkte und Portfolios zu definieren. Diese Klassen werden vom [Modul AWS CDK aws-cdk-lib .aws\\_servicecatalog](#) bereitgestellt.

Sie können die verwenden AWS CDK, um Produkte auf vielfältige Weise zu implementieren. Um zu vermeiden, dass Sie die Bereitstellungsvorlage für ein Produkt schreiben müssen CloudFormation und um die Wiederverwendbarkeit zu gewährleisten, empfehlen wir, die AWS CDK [ProductStack Klasse](#) zu verwenden, die die Bereitstellungsvorlage darstellt. Eine ProductStack Instanz ist ein AWS CDK Stack, dem Sie programmgesteuert Ressourcen hinzufügen. So können Sie zum Beispiel einen S3-Bucket, IAM-Rollen oder ein CloudWatch Amazon-Protokoll hinzufügen. Wenn Sie die ProductStack Instance durch einen Aufruf zu einer definierten `servicecatalog.CloudFormationProduct` Instance als Bereitstellungsvorlage hinzufügen `servicecatalog.CloudFormationTemplate.fromProductStack (<ProductStack instance>)`, generiert der AWS CDK automatisch die CloudFormation Vorlage.

Hier ist ein Beispiel für die ProductStack Java-Implementierung für ein Amazon S3 S3-Produkt.

```
import * as s3 from 'aws-cdk-lib/aws-s3';
import * as cdk from 'aws-cdk-lib';
```

```
class S3BucketProduct extends servicecatalog.ProductStack {
  constructor(scope: Construct, id: string) {
    super(scope, id);

    new s3.Bucket(this, 'BucketProduct');
  }
}

const product = new servicecatalog.CloudFormationProduct(this, 'Product', {
  productName: "My Product",
  owner: "Product Owner",
  productVersions: [
    {
      productVersionName: "v1",
      cloudFormationTemplate:
        servicecatalog.CloudFormationTemplate.fromProductStack(new S3BucketProduct(this,
          'S3BucketProduct')),
    },
  ],
});
```

Das AWS CDK bietet integrierte Continuous Integration und Continuous Deployment (CI/CD). Sie können diese integrierten Pipelines und SDLC-Prozesse (Software Development Lifecycle) an Ihre eigenen Prozessstandards und -ziele anpassen.

Benutzerdefinierte AWS CDK Klassen können von anderen Klassen erben, um spezielle Funktionen bereitzustellen, und eine Klasse kann aus Instanzen anderer Klassen bestehen. Wenn Sie Shared AWS CDK Class-Frameworks verwenden, um mehrere Service Catalog-Produkte zu implementieren, sollten Sie alle Auswirkungen auf Versionierung oder Kompatibilität berücksichtigen, insbesondere zwischen mehreren Entwicklungsteams. Sie müssen sicherstellen, dass Änderungen abwärtskompatibel sind oder dass Sie ein Versionierungsschema haben, das befolgt wird, damit Klassenänderungen, die Sie für ein Produkt vornehmen, kein anderes Produkt beschädigen.

Weitere Informationen finden Sie in der [AWS CDK-Dokumentation](#).

## Integration mit externen Bereitstellungsprozessen und Workflows

Sie können mit Service Catalog-Komponenten interagieren, indem Sie AWS SDK-APIs oder die [AWS CLI](#) verwenden. Sie können die [AWS SDK Service Catalog API verwenden, um Service Catalog-Produkte](#) von jedem Tool aus zu verwalten, das Service Catalog-API-Aufrufe integrieren kann. Die

API deckt alle Aspekte der Erstellung und Verwaltung von Servicekatalogen ab. Terraform unterstützt beispielsweise den Start (die Bereitstellung) von Service Catalog-Produkten, indem es die AWS SDK Service Catalog API in seinem Launch Wizard aufruft. Weitere Informationen finden Sie in der AWS Dokumentation unter [AWS Service Catalog Produkte mit Terraform starten](#).

Sie können auch die AWS CLI Service Catalog-Befehle aufrufen, um Aktionen für Service Catalog auszuführen. Weitere Informationen zu unterstützten Befehlen finden Sie unter [servicecatalog](#) in der AWS CLI Befehlsreferenz.

## Spezifikationen für die Produktbereitstellung

Service Catalog initiiert den Bereitstellungsprozess als CloudFormation Stack-Set-Bereitstellung der Ressourcen, die in der CloudFormation Bereitstellungsvorlage angegeben sind. (Die Vorlage kann direkt im AWS CDK `ProductStack` Konstrukt erstellt werden oder von diesem generiert werden.) Die Bereitstellung von Service Catalog-Produkten ist ein geschlossener Prozess. Sie können ihn nicht anpassen, um Vor- oder Nachverarbeitungsschritte hinzuzufügen oder ihn zu optimieren. Sie können die Bereitstellungsvorlage jedoch ändern, um Schritte in Form von CloudFormation Ressourcenspezifikationen hinzuzufügen. Dies können von AWS Lambda oder von Lambda unterstützte benutzerdefinierte Ressourcen sein, die vorbereitende Schritte (z. B. benutzerdefiniertes Bootstrapping zum Einrichten eines Bastion-Hosts, der während der Bereitstellung verwendet wird) und nachfolgende Schritte (wie das Herunterfahren des Bastion-Hosts) ausführen. Diese Methode zur Implementierung der Schritte vor und nach der Bereitstellung unterliegt denselben `include` und verschachtelten Stack-Einschränkungen wie die Bereitstellungsvorlage.

Sie können Zielkonten als einzelne Konten und nicht als Organisationseinheiten (OUs) angeben. Sie können eine benutzerdefinierte Ressource oder Funktion schreiben, um diese Einschränkung zu umgehen. Die meisten Unternehmen stellen Produktportfolios für Organisationseinheiten und nicht für einzelne Konten bereit, da sie die Generierung von Konten automatisieren und Kontenlisten nicht manuell verwalten möchten.

## DevSecOps Lebenszyklus

Derzeit bieten Produkte, die mit Service CloudFormation Catalog-Skripten bereitgestellt werden, keine integrierte Unterstützung für CI/CD-Prozesse. Wir empfehlen, einen CI/CD-Prozess in AWS CodePipeline oder anderen DevOps Tools zu erstellen, um ein Produkt in Lebenszyklusumgebungen wie Entwicklung, Test, Phase und Produktion zu entwickeln, zu testen und freizugeben.

Das AWS CDK bietet integrierte CI/CD-Unterstützung für Produkte, wie bereits in diesem Handbuch beschrieben.

## Maßgeschneiderte Wiederverwendung und kontospezifische Bereitstellung

Produkte sollten für so viele verschiedene kundenspezifische Zwecke wie möglich wiederverwendbar sein. Service Catalog unterstützt die Wiederverwendbarkeit durch Produktparameter. Sie können diese Parameter bei der Bereitstellung als Eingabe für ein Produkt angeben.

Sie können diese Parameter auch als AWS Systems Manager Parameter Store-Werte auf CloudFormation Vorlagenebene angeben, um kontospezifische und OU-spezifische Werte anzuwenden. Dies ist eine bewährte Methode für die CloudFormation Bereitstellung von Vorlagendesigns. Der Wert des benannten Parameters innerhalb des Zielkontos wird angewendet, wenn das Produkt bereitgestellt wird. Sie können beispielsweise einen Subnetzparameter als Parameter-Speicherwert angeben und dieses Subnetz bei der Produktbereitstellung für ein bestimmtes OU-Konto anwenden. Weitere Informationen zum Speichern von Parametern als CloudFormation Vorlagenparameter finden Sie in der AWS CloudFormation Dokumentation [unter Verwenden dynamischer Referenzen zur Angabe von Vorlagenwerten](#).

## Definition und Verwaltung von Service Catalog-Produktressourcen als Anwendungen

AWS Service Catalog AppRegistry bietet zentrale Funktionen für die Suche, Berichterstattung und Verwaltung von Anwendungen. Eine AppRegistry Anwendung kann einen oder mehrere bereitgestellte Produktstapel sowie CloudFormation Stacks enthalten, die vom Service Catalog unabhängig sind. Sie können alle Ihre Anwendungsressourcensammlungen innerhalb der AWS-Konten, die Sie als Bereitstellungsziele definieren, gruppieren und anzeigen. Diese Konten können Ihre Entwicklungs-, Test- und Produktions-Konto sein.

Sie können es auch verwenden AppRegistry , um Metadatenattribute mit einer Anwendung zu verknüpfen. Sie können wiederverwendbare Attributgruppen zuweisen, die Attributsätze enthalten. Anschließend können Sie Anwendungsressourcen mit den angegebenen Attributen suchen und darauf reagieren, indem Sie AppRegistry unsere integrierten Dienste verwenden. Zu diesen integrierten Diensten gehören:

- [Application Manager](#) zur Untersuchung und Behebung von AWS Systems Manager Problemen mit AWS Ressourcen im Kontext Ihrer Anwendungen und Cluster im Kontext Ihrer Anwendungen und Cluster im Kontext Ihrer Anwendungen und Cluster im Kontext Ihrer Anwendungen und Cluster im Kontext Ihrer Anwendungen und Cluster im Kontext Ihrer Anwendungen und Cluster
- [AWS Resource Access Manager](#), um Anwendungen und Attributgruppen mit AWS Organisationsprinzipien zu teilen
- [AWS-Services, die mit AWS Resource Groups](#) funktionieren
- [AWS Resilience Hub](#) für die Entdeckung der Produktstruktur und die Bewertung der Resilienz
- [AWS Service Management Connector](#) um Verbindungen zu JIRA und anderen gängigen Tools zu ServiceNow deklarieren und einzurichten

## AppRegistry

- [AppRegistry Administratorhandbuch](#).
- [Verbessern Sie die Sichtbarkeit und Steuerung von Anwendungen mithilfe eines AWS Service Catalog AppRegistry](#) Blogbeitrags. Dieser Artikel bietet einen Überblick über die Verwendung AppRegistry in der Infrastrukturverwaltung und enthält Befehlszeilenbeispiele für die Registrierung Ihrer Infrastruktur als Anwendungen in AppRegistry.
- [Steuern Sie Ihre Anwendungen zentral mithilfe eines AppRegistry Application Manager-Blogbeitrags](#). Dieser Artikel bietet eine Übersicht mit einem Tutorial darüber, wie Sie die Registrierung einer LAMP-Webanwendung auf der beantragten AppRegistry AWS Management Console und diese mithilfe des Application Managers verwalten können.

## Inventory

Service Catalog verfügt über eine eigene interne Inventarverwaltungsfunktion, mit der Produkte registriert werden, wenn sie über Product Sharing und Self-Service bereitgestellt werden. Wir empfehlen jedoch, dass Sie [AWS Config](#) oder [AppRegistry](#) und verwandte Dienste verwenden, um Ihre vom Produkt bereitgestellten Ressourcen zu verwalten. Diese Tools bieten einen umfassenderen und integrierteren Ansatz für die Verwaltung Ihrer bereitgestellten Service Catalog-Produkte mit dem Rest Ihrer AWS Infrastruktur. AWS Config ermöglicht es Ihnen, bereitgestellte Produkte auf der Konsole oder mithilfe der AWS SDK-API zu inventarisieren und Aktionen für diese auszuführen. AppRegistry, das in Application Manager integriert ist, bietet auch die Inventarverwaltung für im Service Catalog bereitgestellte Produkte.

# AWS Service Catalog Tools verwenden

Wenn Sie Ihre IaC-Produkte mit benutzerdefinierten Bereitstellungsworkflows auf deklarativere Weise bereitstellen möchten, möchten Sie möglicherweise Teile der Service Catalog-Funktionalität erweitern. AWS stellt verschiedene Tools zur Unterstützung dieser Anforderungen bereit. Im AWS Labs-Projekt werden zwei beliebte Tools bereitgestellt: Service Catalog Puppet und Service Catalog Factory.

## Themen

- [Service Catalog Puppet](#)
- [Service Catalog](#)

## Service Catalog Puppet

Service Catalog Puppet wird in Python mithilfe der AWS Boto3-API implementiert. Dieses Tool bietet mehrere leistungsstarke Funktionen für die Konfiguration und Bereitstellung von Service Catalog-Produkten. Entwickler können Service Catalog-Produkt- und Portfolio-Bereitstellungsinformationen konfigurieren, indem sie YAML-Vorlagen verwenden, die als Manifeste dienen. Die Bereitstellungsworkflows von Service Catalog Puppet unterstützen Produkte, die komplexere Bereitstellungsprozesse erfordern als Service Catalog. Sie unterstützen auch Leistungsoptimierungen, um Produkte in großem Maßstab innerhalb aggressiver Zeitfenster bereitzustellen.

Service Catalog Puppet greift bei der Bereitstellung auf die Service CloudFormation Catalog-Vorlagen für die Produktbereitstellung zu. Es ruft CloudFormation direkt die Bereitstellung des Bereitstellungsvorlagenstapels für ein Produkt auf und umgeht die Einschränkungen, die durch den eigenen Stackset-Bereitstellungsprozess von Service Catalog auferlegt werden. Wenn die Bereitstellungsvorlage Makros verwendet, um andere CloudFormation Skripts einzubinden, oder verschachtelte CloudFormation Skripts verwendet, müssen Sie im Bootstrapping-Teil des Bereitstellungsworkflows Zugriff auf diese Skripts im Zielkonto gewähren.

## Weitere Informationen:

- Weitere Informationen finden Sie in der Service Catalog [Puppet-Dokumentation](#) und im [GitHubRepository](#).

- Wenn Sie das Service Catalog Puppet SDK verwenden möchten, um programmgesteuert mit dem Tool zu interagieren und die Produkt- und Portfoliobereitstellung zu initiieren, finden Sie weitere Informationen in der [SDK-Dokumentation](#).
- [GitOps](#) ist der Standardmechanismus für die Verwaltung der Service Catalog Puppet-Umgebung.

Service Catalog Puppet ist für Entwickler relativ einfach zu erlernen. Es erfordert Vertrautheit mit CloudFormation der Implementierung von Vorlagen für die Produktbereitstellung und mit YAML-Vorlagen zur Implementierung von Manifesten. Es gibt gute Workshops, um neue Entwickler auf den neuesten Stand zu bringen, z. B. [Workshops zum Selbststudium](#).

## Support für die Bereitstellung von Workflows

Service Catalog Puppet verwendet die Python Luigi Task Orchestration Engine, um Bootstrapping- und Provisioning-Workflows zu implementieren. Alle Schritte in diesen Workflows werden als Luigi-Workflow-Aufgaben implementiert. Einen Überblick über Luigi und seinen Vergleich zu anderen gängigen Workflow-Tools finden Sie im Data Revenue Blog unter [Airflow vs. Luigi vs. Argo vs. MLFlow vs. KubeFlow](#)

Luigi ermöglicht es Service Catalog Puppet, die Anzahl der mit Workflow-Aufgaben verbundenen Mitarbeiter zu kontrollieren und andere Aspekte der Workflows zu kontrollieren, um eine bessere Skalierung und Leistung zu erzielen. Service Catalog Puppet bietet auch einen [depends\\_on-Mechanismus](#) für die Verwaltung von Produkt- und Schrittabhängigkeiten sowie für die Orchestrierung der Produktbereitstellung. Diese Funktion hilft Ihnen bei der Implementierung und operativen Verwaltung detaillierter Produktdefinitionen und komplexer Abhängigkeiten.

## Bereitstellungsmodi

Service Catalog Puppet unterstützt drei Ausführungsmodi: [Hub, Spoke und Async](#). Alle drei Modi stellen Produkte innerhalb von Portfolios bereit, die bereits im Service Catalog definiert sind. Sie verlassen sich auf die gemeinsame Nutzung von Service Catalog-Produkten an die Zielkonten und verwenden Service Catalog-Administrator- und Startrollen, um die Bereitstellung in diesen Zielen zu realisieren. Service Catalog Puppet führt die Bootstrapping-Schritte innerhalb derselben Organisation auf der Grundlage der Rollenkonfigurationen aus, die in den YAML-Konfigurationsdateien bereitgestellt werden. Das Tool unterstützt auch die Bereitstellung für mehrere Organisationen von einem einzigen Hub-Konto aus. In diesem Szenario muss das Bootstrapping in den externen Organisationen manuell durchgeführt werden, damit Service Catalog Puppet die erforderlichen Bereitstellungsaktionen in den Konten der externen Organisation durchführen kann.



In allen Bereitstellungsmodi implementiert Service Catalog Puppet die Produktbereitstellung direkt, ohne den Bereitstellungsprozess von Service Catalog aufzurufen. Sie können ein Bereitstellungsmanifest so konfigurieren, dass es die Rollen- und Zielkontospezifikationen in einer vorhandenen Service Catalog-Stack-Set-Einschränkung verwendet. Service Catalog Puppet verwendet diese Informationen, um die Luigi-Workflows selbst bereitzustellen.

Sie können Ziele für die Bereitstellung von Produktportfolios auf der Grundlage eines Account-Tagging-Ansatzes definieren und zusätzlich Organisationseinheiten oder Konten direkt angeben. Bei der kontotagbasierten Bereitstellung wird ein Portfolioprodukt für alle Konten bereitgestellt, die alle Tags im angegebenen Tag-Set für die Manifestbereitstellung enthalten. Wenn Sie beispielsweise ein Portfolioprodukt an alle institutionellen Produktionskonten in den Regionen USA Ost ausgeben möchten, können Sie die Stichwörtertype:prod und angebenscope:institutional-client.partition:us-east angeben. Sie können auch Konten- und Organisationsausschlüsse deklarieren, um die Bereitstellung für Organisationseinheiten oder Konten mit den von Ihnen angegebenen Tags oder für Konten, die Mitglieder der von der Organisationseinheit angegebenen Ziele sind, zu untersagen. Weitere Informationen zur Kontokennzeichnung finden Sie in der [Dokumentation zu den Service Catalog Tools](#).

## Hub Hub-Hub-

Im Hub-Bereitstellungsmodus werden alle Luigi-Workflows für die Spoke-Konten vom angegebenen zentralen Hub-Konto aus verwaltet. Das Hub-Konto nimmt eine IAM-Rolle an, die es ihm ermöglicht, Aktionen in einem Spoke-Konto auszuführen. Die Verwaltung der Aufgaben erfolgt jedoch vom Hub-Konto aus. Das Hub-Konto wartet synchron, bis alle Aufgaben zur Bereitstellung von Spoke-Konten abgeschlossen sind, entweder erfolgreich oder erfolglos. Anschließend wird der endgültige Status gemeldet. Der Hub-Kontomodus ist der älteste und ausgereifteste Bereitstellungsmodus. Viele Benutzer sind jedoch auf den Spoke-Provisioning-Modus umgestiegen, um eine höhere Parallelität und Geschwindigkeit der Bereitstellung zu erreichen.

## Für den Spoke-

Im Spoke-Modus initiiert das Service Catalog-Hub-Konto die Luigi-Workflows, sodass sie in den angegebenen Bootstrap-Spoke-Konten ausgeführt werden. Das Hub-Konto wird benachrichtigt, wenn die Spoke-Workflows abgeschlossen sind. Fehler in einem Spoke-Konto werden auf das Hub-Konto übertragen. Das Hub-Konto fragt das Spoke-Konto ab, um festzustellen, ob es fertig ist, und um den Status zu ermitteln.

Im Spoke-Modus ist es am wenigsten wahrscheinlich, dass AWS-Service Quotenerhöhungen erforderlich sind, da fast alles in den separaten Spoke-Konten läuft. Der Spoke-Modus bietet

auch eine weitaus größere Parallelität als der Hub-Modus, wobei die zentrale Steuerung erhalten bleibt. Es kann die Bereitstellungsgeschwindigkeit gegenüber dem Hub-Modus um 800 Prozent verbessern. Der Spoke-Modus unterstützt die Verkettung von Produkten durch `DependsOn` Beziehungen zwischen Produkten, wodurch sichergestellt wird, dass ein Produkt, von dem abhängig ist, bereits bereitgestellt wurde. Ein Produkt, das verkettete Produkte umfasst, kann auch ein komponentengekettetes Produkt bereitstellen. Sie können auch spezielle AWS Lambda Funktionsaufrufe verwenden, um die erforderlichen Schritte auszuführen. Fehler in einer Speicher sind von anderen Speichern isoliert.

Der Spoke-Modus wird von Unternehmen mit über 980 Konten in bis zu 7 Regionen verwendet. Diese Unternehmen sind in der Regel in der Lage, innerhalb einer Stunde ein Produkt für alle Regionen und Konten in ihrer Infrastruktur bereitzustellen.

#### Note

Diese Ergebnisse können je nach Faktoren wie der Netzwerkinfrastruktur, der Arbeitslast und den für die Hub- und Spoke-Konten der AWS Organisation geltenden Kontingente variieren. Sie hängen auch von den Produktressourcen ab, die bereitgestellt werden, ihren inhärenten Erstellungszeiten und ihrer Abhängigkeit von anderen Ressourcen.

## Für den Async-Modus

Der asynchrone Modus initiiert Provisioning-Workflows in Spoke-Konten, wartet jedoch nicht auf Abschlussantworten der Spokes und empfängt diese auch nicht.

## Caching

Ein weiterer Mechanismus, den Service Catalog Puppet verwendet, um die Geschwindigkeit von Workflows zu optimieren, besteht darin, allgemeine Aufgaben zwischenspeichern, die Schritte im Workflow darstellen. Wenn eine zwischengespeicherte Aufgabe abgeschlossen ist, schreibt sie ihre Ausgaben in Amazon Simple Storage Service (Amazon S3). Wenn die Aufgabe das nächste Mal in derselben Sitzung mit denselben Parametern aufgerufen wird, verwendet Service Catalog Puppet die zwischengespeicherten Werte, anstatt die Aufgabe erneut auszuführen. Weitere Informationen finden Sie unter [Caching](#) in der Service Catalog.

## DevSecOps Unterstützung für den LebenszyklusSupport

Service Catalog Puppet beinhaltet Unterstützung für die Verwaltung der DevSecOps Pipeline. Sie können die Aktionen der Service Catalog Tools (wie in der [Übersicht über Service Catalog Puppet](#) dargestellt) verwenden, um Tests zu automatisieren und Produkte in Ihren AWS Lifecycle-Konten zu bewerben, einschließlich des empfohlenen Canary-Kontos. Weitere Informationen finden Sie in der Service Catalog Puppet-Dokumentation unter [Verwalten Ihrer Umgebungen](#).

Um sicherzustellen, dass alle Probleme im Zusammenhang mit einer Produktänderung erkannt werden, bevor sie in großem Umfang in der Produktion eingesetzt werden, benötigt Service Catalog Puppet für die erste Bereitstellung mindestens ein Canary-Konto. Nachdem Sie die neue Version getestet und Vertrauen in sie gewonnen haben, können Sie sie für Kunden bewerben, die nicht auf Canary produzieren. Wenn Sie Probleme feststellen, können Sie die Version rückgängig machen und sie wieder einführen, sobald die Probleme behoben sind. Wenn Sie diesen Ansatz verwenden, können Produktionsprobleme auftreten, wenn Sie eine Canary-Version veröffentlichen, bei der ein Problem mit den Produktionskonten besteht. Alternativ können Sie für jede Produktänderung vollständige Regressionstests durchführen, bevor Sie die Änderung für die Produktion freigeben. Dies führt zu zusätzlichem Aufwand für den CI/CD-Prozess, hilft jedoch, Produktionsprobleme zu vermeiden. Es liegt in der Verantwortung des DevSecOps Administrators, die besten Feature-Release-Szenarien und -Ansätze für seine Entwicklungsteams zu ermitteln.

Service Catalog Puppet ermöglicht es mehreren Teams, die Bereitstellung von Service Catalog-Produktlösungen gleichzeitig zu entwickeln und zu testen. Es hat sich bewährt, ein Produkt nicht von mehreren Entwicklern gleichzeitig zu ändern. Stattdessen können Sie Produkte für separate, gleichzeitige Änderungen in feinkörnige Komponenten aufteilen.

Service Catalog Puppet hilft auch bei der Automatisierung von Tests durch eine Assertion-Anweisung, die statische und Komponententestfunktionen bereitstellt. Sie können Service Control Policies (SCP) und IAM-Richtlinien testen, indem Sie RichtlinienSimulatoren verwenden. Dies sind technische end-to-end Tests, können aber in Systemintegrationstestumgebungen (SIT) verwendet werden. Weitere Informationen finden Sie unter [Verwenden von Richtlinienimulationen](#) und [Anwenden von Service Control-Richtlinien](#) in der Service Catalog Puppet-Dokumentation.

## Reife, Vollständigkeit und Unterstützung

Service Catalog Puppet wird zwar nicht offiziell unterstützt AWS-Service, hat aber breite Akzeptanz gefunden. Dieses Tool wurde in den letzten Jahren von großen Organisationen verwendet, um Produkte für Hunderte von OU-Konten innerhalb der gewünschten Bereitstellungszeitfenster

erfolgreich und zentral bereitzustellen. Es hat sich gezeigt, dass es eine fehlertolerante Produktbereitstellung in großem Maßstab bietet. Benutzer, die auf Probleme mit Service Catalog Puppet stoßen, können diese im [GitHub Repository](#) anmelden, damit sie von den Mitwirkenden an dieser AWS Labs-Lösung behoben werden können.

## Service Catalog

Service Catalog Factory ist ein weiteres von AWS Labs bereitgestelltes Tool. Es ist ähnlich wie AWS Control Tower —es generiert Konten und ruft Service Catalog (möglicherweise über Puppet) auf, um iAP innerhalb dieser Konten bereitzustellen. Es verwendet viele der gleichen Mechanismen wie Service Catalog Puppet, um seine Funktionen zu implementieren. Service Catalog Factory kann Service Catalog oder Service Catalog Puppet anrufen, um die Infrastruktur für Produkte in einem Konto bereitzustellen. Dieses Tool unterstützt auch die Kontogenerierung in mehreren AWS-Regionen und Organisationen. Weitere Informationen finden Sie in der [Dokumentation](#) und im [GitHub Repository](#) von Service Catalog Factory.

## Zusammenfassung und nächste Schritte

Service Catalog hilft Ihnen, Ihre Infrastruktur schnell und zuverlässig als Produkt bereitzustellen. Sie können die Infrastruktur aus einem definierten Produktkatalog selbst verwalten oder Produkte an bestimmte Zielkonten in einem hub-and-spoke Modell weiterleiten. Sie können Service Catalog-Produkte und ihre Bereitstellungsvorlagen mithilfe von CloudFormation Skripten oder mithilfe von definierten AWS CDK. Bei beiden Ansätzen stellt Service Catalog ein Produkt bereit, indem CloudFormation es zur Bereitstellung eines Stacks aufruft, der die Bereitstellungsvorlage des Produkts darstellt. Der Stack wird für alle angegebenen Zielkonten innerhalb eines CloudFormation Stacksets bereitgestellt.

Der AWS CDK Ansatz für die Entwicklung von Service Catalog unterstützt eine stärkere Modularisierung und Wiederverwendung als CloudFormation, weil Sie Produkte und deren Ressourcen definieren können, indem Sie vordefinierte Service Catalog-Produkt- und Portfolioklassen sowie vordefinierte Ressourcentypen verwenden. Eine AWS CDK Implementierung erfordert fortgeschrittenere Programmierkenntnisse. Dies könnte gerechtfertigt sein, wenn Ihr Unternehmen ein eigenes wiederverwendbares Produktrahmen mit standardisierten Ressourcenkonfigurationen und -verhalten als Grundlage für die Entwicklung Ihrer AWS Infrastruktur einrichten möchte.

Sie können Service Catalog Puppet und Service Catalog Factory verwenden, um die Service Catalog-Funktionalität zu erweitern, hauptsächlich für die Bereitstellung. Service Catalog Puppet bietet deklarative und tagbasierte Produktbereitstellungsspezifikationen, integrierte, anpassbare, leistungsstarke und speziell entwickelte Bereitstellungsworkflows sowie integrierte, anpassbare, aktionsbasierte CI/CD- und SDLC-Pipelines. Mithilfe des Workflow-Abhängigkeitsmanagements und der integrierten Funktionen zur Testautomatisierung können Sie Service Catalog-Produkte mit geringerem Betriebsrisiko verketteten. Service Catalog Puppet hilft Ihnen dabei, Produkte für Hunderte von Konten innerhalb aggressiver Zeitfenster zuverlässig bereitzustellen. Service Catalog Factory ist ähnlich wie AWS Control Tower. Es generiert Konten und ruft Service Catalog auf, um iAP innerhalb dieser Konten bereitzustellen.

Service Catalog und Service Catalog Tools bieten umfangreiche Funktionen, mit denen Sie iAP auf verwalten können AWS. Der Service Catalog und diese Tools werden ständig verbessert. Die neuesten Funktionen finden Sie unter [AWS Service Catalog Funktionen](#) und im [AWS Service Catalog Produkt-Repository](#).

# Ressourcen

## Verweise

- [Dokumentation zum Service Catalog](#)
- [Servicekatalog-API](#)
- [AppRegistry](#)
- [AWS CloudFormation-Dokumentation](#)
- [AWS CloudFormationStapelsätze](#)
- [AWS::ServiceCatalog::CloudFormationProduktressource](#)
- [AWS Service CatalogProdukte mit Terraform auf den Markt bringen](#)
- [AWS Cloud Development Kit \(AWS CDK\)](#)
- [Service Catalog Construct Library](#)
- [AWS CDK ProductStackKlasse](#)
- [AWS Organizations-Dokumentation](#)

## Tools

- [Service Catalog Puppet-Dokumentation](#)
- [GitHubPuppet-Repository für Service Catalog](#)
- [Service Catalog Werksdokumentation](#)
- [Service Catalog GitHub Factory-Repository](#)

## AWSPräskriptive Leitbilder

- [AWS Service CatalogProdukte in mehreren verwaltenAWS-Konten undAWS-Regionen](#)
- [Kopieren SieAWS Service Catalog Produkte zwischen verschiedenenAWS-Konten undAWS-Regionen](#)
- [Automatisieren Sie denAWS Service Catalog Portfolio- und Produkteinsatz mithilfe vonAWS CDK](#)

# Dokumentverlauf

In der folgenden Tabelle werden die wichtigsten Änderungen an diesem Handbuch beschrieben. Wenn Sie über future Updates informiert werden möchten, können Sie einen [RSS-Feed](#) abonnieren.

Änderung	Beschreibung	Datum
<a href="#">Erste Veröffentlichung</a>	—	30. Januar 2023

# AWS Glossar zu präskriptiven Leitlinien

Im Folgenden finden Sie häufig verwendete Begriffe in Strategien, Leitfäden und Mustern, die von Prescriptive Guidance bereitgestellt AWS werden. Um Einträge vorzuschlagen, verwenden Sie bitte den Link Feedback geben am Ende des Glossars.

## Zahlen

### 7 Rs

Sieben gängige Migrationsstrategien für die Verlagerung von Anwendungen in die Cloud. Diese Strategien bauen auf den 5 Rs auf, die Gartner 2011 identifiziert hat, und bestehen aus folgenden Elementen:

- Faktorwechsel/Architekturwechsel – Verschieben Sie eine Anwendung und ändern Sie ihre Architektur, indem Sie alle Vorteile cloudnativer Feature nutzen, um Agilität, Leistung und Skalierbarkeit zu verbessern. Dies beinhaltet in der Regel die Portierung des Betriebssystems und der Datenbank. Beispiel: Migrieren Sie Ihre On-Premises-Oracle-Datenbank zu der PostgreSQL-kompatible Amazon-Aurora-Edition.
- Plattformwechsel (Lift and Reshape) – Verschieben Sie eine Anwendung in die Cloud und führen Sie ein gewisses Maß an Optimierung ein, um die Cloud-Funktionen zu nutzen. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank zu Amazon Relational Database Service (Amazon RDS) für Oracle in der AWS Cloud.
- Neukauf (Drop and Shop) – Wechseln Sie zu einem anderen Produkt, indem Sie typischerweise von einer herkömmlichen Lizenz zu einem SaaS-Modell wechseln. Beispiel: Migrieren Sie Ihr CRM-System (Customer Relationship Management) zu Salesforce.com.
- Hostwechsel (Lift and Shift) – Verschieben Sie eine Anwendung in die Cloud, ohne Änderungen vorzunehmen, um die Cloud-Funktionen zu nutzen. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank auf einer EC2-Instance in der Cloud zu Oracle. AWS
- Verschieben (Lift and Shift auf Hypervisor-Ebene) – Verlagern Sie die Infrastruktur in die Cloud, ohne neue Hardware kaufen, Anwendungen umschreiben oder Ihre bestehenden Abläufe ändern zu müssen. Dieses Migrationsszenario ist spezifisch für VMware Cloud on AWS, das die Kompatibilität mit virtuellen Maschinen (VM) und die Workload-Portabilität zwischen Ihrer lokalen Umgebung und unterstützt. AWS Sie können die VMware-Cloud-Foundation-Technologien von Ihren On-Premises-Rechenzentren aus verwenden, wenn Sie



Ihre Infrastruktur zu VMware Cloud in AWS migrieren. Beispiel: Verlagern Sie den Hypervisor, der Ihre Oracle-Datenbank hostet, zu VMware Cloud on. AWS

- Beibehaltung (Wiederaufgreifen) – Bewahren Sie Anwendungen in Ihrer Quellumgebung auf. Dazu können Anwendungen gehören, die einen umfangreichen Faktorwechsel erfordern und die Sie auf einen späteren Zeitpunkt verschieben möchten, sowie ältere Anwendungen, die Sie beibehalten möchten, da es keine geschäftliche Rechtfertigung für ihre Migration gibt.
- Außerbetriebnahme – Dekommissionierung oder Entfernung von Anwendungen, die in Ihrer Quellumgebung nicht mehr benötigt werden.

## A

### ABAC

Siehe [attributbasierte Zugriffskontrolle](#).

### abstrahierte Dienste

Siehe [Managed Services](#).

### ACID

Siehe [Atomarität, Konsistenz, Isolierung und Haltbarkeit](#).

### Aktiv-Aktiv-Migration

Eine Datenbankmigrationsmethode, bei der die Quell- und Zieldatenbanken synchron gehalten werden (mithilfe eines bidirektionalen Replikationstools oder dualer Schreibvorgänge) und beide Datenbanken Transaktionen von miteinander verbundenen Anwendungen während der Migration verarbeiten. Diese Methode unterstützt die Migration in kleinen, kontrollierten Batches, anstatt einen einmaligen Cutover zu erfordern. Es ist flexibler, erfordert aber mehr Arbeit als eine [aktiv-passive](#) Migration.

### Aktiv-Passiv-Migration

Eine Datenbankmigrationsmethode, bei der die Quell- und Zieldatenbanken synchron gehalten werden, aber nur die Quelldatenbank Transaktionen von verbindenden Anwendungen verarbeitet, während Daten in die Zieldatenbank repliziert werden. Die Zieldatenbank akzeptiert während der Migration keine Transaktionen.

## Aggregatfunktion

Eine SQL-Funktion, die mit einer Gruppe von Zeilen arbeitet und einen einzelnen Rückgabewert für die Gruppe berechnet. Beispiele für Aggregatfunktionen sind SUM und MAX.

## AI

Siehe [künstliche Intelligenz](#).

## AIOps

Siehe [Operationen mit künstlicher Intelligenz](#).

## Anonymisierung

Der Prozess des dauerhaften Löschens personenbezogener Daten in einem Datensatz. Anonymisierung kann zum Schutz der Privatsphäre beitragen. Anonymisierte Daten gelten nicht mehr als personenbezogene Daten.

## Anti-Muster

Eine häufig verwendete Lösung für ein wiederkehrendes Problem, bei dem die Lösung kontraproduktiv, ineffektiv oder weniger wirksam als eine Alternative ist.

## Anwendungssteuerung

Ein Sicherheitsansatz, bei dem nur zugelassene Anwendungen verwendet werden können, um ein System vor Schadsoftware zu schützen.

## Anwendungsportfolio

Eine Sammlung detaillierter Informationen zu jeder Anwendung, die von einer Organisation verwendet wird, einschließlich der Kosten für die Erstellung und Wartung der Anwendung und ihres Geschäftswerts. Diese Informationen sind entscheidend für [den Prozess der Portfoliofindung und -analyse](#) und hilft bei der Identifizierung und Priorisierung der Anwendungen, die migriert, modernisiert und optimiert werden sollen.

## künstliche Intelligenz (KI)

Das Gebiet der Datenverarbeitungswissenschaft, das sich der Nutzung von Computertechnologien zur Ausführung kognitiver Funktionen widmet, die typischerweise mit Menschen in Verbindung gebracht werden, wie Lernen, Problemlösen und Erkennen von Mustern. Weitere Informationen finden Sie unter [Was ist künstliche Intelligenz?](#)

## Operationen mit künstlicher Intelligenz (AIOps)

Der Prozess des Einsatzes von Techniken des Machine Learning zur Lösung betrieblicher Probleme, zur Reduzierung betrieblicher Zwischenfälle und menschlicher Eingriffe sowie zur Steigerung der Servicequalität. Weitere Informationen zur Verwendung von AIOps in der AWS - Migrationsstrategie finden Sie im [Leitfaden zur Betriebsintegration](#).

## Asymmetrische Verschlüsselung

Ein Verschlüsselungsalgorithmus, der ein Schlüsselpaar, einen öffentlichen Schlüssel für die Verschlüsselung und einen privaten Schlüssel für die Entschlüsselung verwendet. Sie können den öffentlichen Schlüssel teilen, da er nicht für die Entschlüsselung verwendet wird. Der Zugriff auf den privaten Schlüssel sollte jedoch stark eingeschränkt sein.

## Atomizität, Konsistenz, Isolierung, Haltbarkeit (ACID)

Eine Reihe von Softwareeigenschaften, die die Datenvalidität und betriebliche Zuverlässigkeit einer Datenbank auch bei Fehlern, Stromausfällen oder anderen Problemen gewährleisten.

## Attributbasierte Zugriffskontrolle (ABAC)

Die Praxis, detaillierte Berechtigungen auf der Grundlage von Benutzerattributen wie Abteilung, Aufgabenrolle und Teamname zu erstellen. Weitere Informationen finden Sie unter [ABAC AWS](#) in der AWS Identity and Access Management (IAM-) Dokumentation.

## autoritative Datenquelle

Ein Ort, an dem Sie die primäre Version der Daten speichern, die als die zuverlässigste Informationsquelle angesehen wird. Sie können Daten aus der maßgeblichen Datenquelle an andere Speicherorte kopieren, um die Daten zu verarbeiten oder zu ändern, z. B. zu anonymisieren, zu redigieren oder zu pseudonymisieren.

## Availability Zone

Ein bestimmter Standort innerhalb einer AWS-Region, der vor Ausfällen in anderen Availability Zones geschützt ist und kostengünstige Netzwerkkonnektivität mit niedriger Latenz zu anderen Availability Zones in derselben Region bietet.

## AWS Framework für die Einführung der Cloud (AWS CAF)

Ein Framework mit Richtlinien und bewährten Verfahren, das Unternehmen bei der Entwicklung eines effizienten und effektiven Plans für den erfolgreichen Umstieg auf die Cloud unterstützt. AWS CAF unterteilt die Leitlinien in sechs Schwerpunktbereiche, die als Perspektiven bezeichnet werden: Unternehmen, Mitarbeiter, Unternehmensführung, Plattform, Sicherheit und

Betrieb. Die Perspektiven Geschäft, Mitarbeiter und Unternehmensführung konzentrieren sich auf Geschäftskompetenzen und -prozesse, während sich die Perspektiven Plattform, Sicherheit und Betriebsabläufe auf technische Fähigkeiten und Prozesse konzentrieren. Die Personalperspektive zielt beispielsweise auf Stakeholder ab, die sich mit Personalwesen (HR), Personalfunktionen und Personalmanagement befassen. Aus dieser Perspektive bietet AWS CAF Leitlinien für Personalentwicklung, Schulung und Kommunikation, um das Unternehmen auf eine erfolgreiche Cloud-Einführung vorzubereiten. Weitere Informationen finden Sie auf der [AWS -CAF-Webseite](#) und dem [AWS -CAF-Whitepaper](#).

### AWS Workload-Qualifizierungsrahmen (AWS WQF)

Ein Tool, das Workloads bei der Datenbankmigration bewertet, Migrationsstrategien empfiehlt und Arbeitsschätzungen bereitstellt. AWS WQF ist in () enthalten. AWS Schema Conversion Tool AWS SCT Es analysiert Datenbankschemas und Codeobjekte, Anwendungscode, Abhängigkeiten und Leistungsmerkmale und stellt Bewertungsberichte bereit.

## B

### schlechter Bot

Ein [Bot](#), der Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen soll.

### BCP

Siehe [Planung der Geschäftskontinuität](#).

### Verhaltensdiagramm

Eine einheitliche, interaktive Ansicht des Ressourcenverhaltens und der Interaktionen im Laufe der Zeit. Sie können ein Verhaltensdiagramm mit Amazon Detective verwenden, um fehlgeschlagene Anmeldeversuche, verdächtige API-Aufrufe und ähnliche Vorgänge zu untersuchen. Weitere Informationen finden Sie unter [Daten in einem Verhaltensdiagramm](#) in der Detective-Dokumentation.

### Big-Endian-System

Ein System, welches das höchstwertige Byte zuerst speichert. Siehe auch [Endianness](#).

### Binäre Klassifikation

Ein Prozess, der ein binäres Ergebnis vorhersagt (eine von zwei möglichen Klassen). Beispielsweise könnte Ihr ML-Modell möglicherweise Probleme wie „Handelt es sich bei dieser

E-Mail um Spam oder nicht?“ vorhersagen müssen oder „Ist dieses Produkt ein Buch oder ein Auto?“

### Bloom-Filter

Eine probabilistische, speichereffiziente Datenstruktur, mit der getestet wird, ob ein Element Teil einer Menge ist.

### Blau/Grün-Bereitstellung

Eine Bereitstellungsstrategie, bei der Sie zwei separate, aber identische Umgebungen erstellen. Sie führen die aktuelle Anwendungsversion in einer Umgebung (blau) und die neue Anwendungsversion in der anderen Umgebung (grün) aus. Mit dieser Strategie können Sie schnell und mit minimalen Auswirkungen ein Rollback durchführen.

### Bot

Eine Softwareanwendung, die automatisierte Aufgaben über das Internet ausführt und menschliche Aktivitäten oder Interaktionen simuliert. Manche Bots sind nützlich oder nützlich, wie z. B. Webcrawler, die Informationen im Internet indexieren. Einige andere Bots, sogenannte böartige Bots, sollen Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen.

### Botnetz

Netzwerke von [Bots](#), die mit [Malware](#) infiziert sind und unter der Kontrolle einer einzigen Partei stehen, die als Bot-Herder oder Bot-Operator bezeichnet wird. Botnetze sind der bekannteste Mechanismus zur Skalierung von Bots und ihrer Wirkung.

### branch

Ein containerisierter Bereich eines Code-Repositorys. Der erste Zweig, der in einem Repository erstellt wurde, ist der Hauptzweig. Sie können einen neuen Zweig aus einem vorhandenen Zweig erstellen und dann Feature entwickeln oder Fehler in dem neuen Zweig beheben. Ein Zweig, den Sie erstellen, um ein Feature zu erstellen, wird allgemein als Feature-Zweig bezeichnet. Wenn das Feature zur Veröffentlichung bereit ist, führen Sie den Feature-Zweig wieder mit dem Hauptzweig zusammen. Weitere Informationen finden Sie unter [Über Branches](#) (GitHub Dokumentation).

### Zugang durch Glasbruch

Unter außergewöhnlichen Umständen und im Rahmen eines genehmigten Verfahrens ist dies eine schnelle Methode für einen Benutzer, auf einen Bereich zuzugreifen AWS-Konto, für den er in der Regel keine Zugriffsrechte besitzt. Weitere Informationen finden Sie unter dem Indikator [Implementation break-glass procedures](#) in den AWS Well-Architected-Leitlinien.

## Brownfield-Strategie

Die bestehende Infrastruktur in Ihrer Umgebung. Wenn Sie eine Brownfield-Strategie für eine Systemarchitektur anwenden, richten Sie sich bei der Gestaltung der Architektur nach den Einschränkungen der aktuellen Systeme und Infrastruktur. Wenn Sie die bestehende Infrastruktur erweitern, könnten Sie Brownfield- und [Greenfield](#)-Strategien mischen.

## Puffer-Cache

Der Speicherbereich, in dem die am häufigsten abgerufenen Daten gespeichert werden.

## Geschäftsfähigkeit

Was ein Unternehmen tut, um Wert zu generieren (z. B. Vertrieb, Kundenservice oder Marketing). Microservices-Architekturen und Entwicklungsentscheidungen können von den Geschäftskapazitäten beeinflusst werden. Weitere Informationen finden Sie im Abschnitt [Organisiert nach Geschäftskapazitäten](#) des Whitepapers [Ausführen von containerisierten Microservices in AWS](#).

## Planung der Geschäftskontinuität (BCP)

Ein Plan, der die potenziellen Auswirkungen eines störenden Ereignisses, wie z. B. einer groß angelegten Migration, auf den Betrieb berücksichtigt und es einem Unternehmen ermöglicht, den Betrieb schnell wieder aufzunehmen.

# C

## CAF

Weitere Informationen finden Sie unter [Framework für die AWS Cloud-Einführung](#).

## Bereitstellung auf Kanaren

Die langsame und schrittweise Veröffentlichung einer Version für Endbenutzer. Wenn Sie sich sicher sind, stellen Sie die neue Version bereit und ersetzen die aktuelle Version vollständig.

## CCoE

Weitere Informationen finden Sie [im Cloud Center of Excellence](#).

## CDC

Siehe [Erfassung von Änderungsdaten](#).

## Erfassung von Datenänderungen (CDC)

Der Prozess der Nachverfolgung von Änderungen an einer Datenquelle, z. B. einer Datenbanktabelle, und der Aufzeichnung von Metadaten zu der Änderung. Sie können CDC für verschiedene Zwecke verwenden, z. B. für die Prüfung oder Replikation von Änderungen in einem Zielsystem, um die Synchronisation aufrechtzuerhalten.

## Chaos-Technik

Absichtliches Einführen von Ausfällen oder Störungsereignissen, um die Widerstandsfähigkeit eines Systems zu testen. Sie können [AWS Fault Injection Service \(AWS FIS\)](#) verwenden, um Experimente durchzuführen, die Ihre AWS Workloads stress, und deren Reaktion zu bewerten.

## CI/CD

Siehe [Continuous Integration und Continuous Delivery](#).

## Klassifizierung

Ein Kategorisierungsprozess, der bei der Erstellung von Vorhersagen hilft. ML-Modelle für Klassifikationsprobleme sagen einen diskreten Wert voraus. Diskrete Werte unterscheiden sich immer voneinander. Beispielsweise muss ein Modell möglicherweise auswerten, ob auf einem Bild ein Auto zu sehen ist oder nicht.

## clientseitige Verschlüsselung

Lokale Verschlüsselung von Daten, bevor das Ziel sie AWS-Service empfängt.

## Cloud-Kompetenzzentrum (CCoE)

Ein multidisziplinäres Team, das die Cloud-Einführung in der gesamten Organisation vorantreibt, einschließlich der Entwicklung bewährter Cloud-Methoden, der Mobilisierung von Ressourcen, der Festlegung von Migrationszeitplänen und der Begleitung der Organisation durch groß angelegte Transformationen. Weitere Informationen finden Sie in den [CCoE-Beiträgen](#) im AWS Cloud Enterprise Strategy Blog.

## Cloud Computing

Die Cloud-Technologie, die typischerweise für die Ferndatenspeicherung und das IoT-Gerätemanagement verwendet wird. Cloud Computing ist häufig mit [Edge-Computing-Technologie](#) verbunden.

## Cloud-Betriebsmodell

In einer IT-Organisation das Betriebsmodell, das zum Aufbau, zur Weiterentwicklung und Optimierung einer oder mehrerer Cloud-Umgebungen verwendet wird. Weitere Informationen finden Sie unter [Aufbau Ihres Cloud-Betriebsmodells](#).

## Phasen der Einführung der Cloud

Die vier Phasen, die Unternehmen normalerweise durchlaufen, wenn sie zur AWS Cloud migrieren:

- Projekt – Durchführung einiger Cloud-bezogener Projekte zu Machbarkeitsnachweisen und zu Lernzwecken
- Fundament – Grundlegende Investitionen tätigen, um Ihre Cloud-Einführung zu skalieren (z. B. Einrichtung einer Landing Zone, Definition eines CCoE, Einrichtung eines Betriebsmodells)
- Migration – Migrieren einzelner Anwendungen
- Neuentwicklung – Optimierung von Produkten und Services und Innovation in der Cloud

Diese Phasen wurden von Stephen Orban im Blogbeitrag [The Journey Toward Cloud-First & the Stages of Adoption](#) im AWS Cloud Enterprise Strategy-Blog definiert. Informationen darüber, wie sie mit der AWS Migrationsstrategie zusammenhängen, finden Sie im Leitfaden zur Vorbereitung der [Migration](#).

## CMDB

Siehe [Datenbank für das Konfigurationsmanagement](#).

## Code-Repository

Ein Ort, an dem Quellcode und andere Komponenten wie Dokumentation, Beispiele und Skripts gespeichert und im Rahmen von Versionskontrollprozessen aktualisiert werden. Zu den gängigen Cloud-Repositorys gehören GitHub oder AWS CodeCommit. Jede Version des Codes wird Zweig genannt. In einer Microservice-Struktur ist jedes Repository einer einzelnen Funktionalität gewidmet. Eine einzelne CI/CD-Pipeline kann mehrere Repositorien verwenden.

## Kalter Cache

Ein Puffer-Cache, der leer oder nicht gut gefüllt ist oder veraltete oder irrelevante Daten enthält. Dies beeinträchtigt die Leistung, da die Datenbank-Instance aus dem Hauptspeicher oder der Festplatte lesen muss, was langsamer ist als das Lesen aus dem Puffercache.



## Kalte Daten

Daten, auf die selten zugegriffen wird und die in der Regel historisch sind. Bei der Abfrage dieser Art von Daten sind langsame Abfragen in der Regel akzeptabel. Durch die Verlagerung dieser Daten auf leistungsschwächere und kostengünstigere Speicherstufen oder -klassen können Kosten gesenkt werden.

## Computer Vision (CV)

Ein Bereich der [KI](#), der maschinelles Lernen nutzt, um Informationen aus visuellen Formaten wie digitalen Bildern und Videos zu analysieren und zu extrahieren. AWS Panorama Bietet beispielsweise Geräte an, die CV zu lokalen Kameranetzwerken hinzufügen, und Amazon SageMaker stellt Bildverarbeitungsalgorithmen für CV bereit.

## Drift in der Konfiguration

Bei einer Arbeitslast eine Änderung der Konfiguration gegenüber dem erwarteten Zustand. Dies kann dazu führen, dass der Workload nicht mehr richtlinienkonform wird, und zwar in der Regel schrittweise und unbeabsichtigt.

## Verwaltung der Datenbankkonfiguration (CMDB)

Ein Repository, das Informationen über eine Datenbank und ihre IT-Umgebung speichert und verwaltet, inklusive Hardware- und Softwarekomponenten und deren Konfigurationen. In der Regel verwenden Sie Daten aus einer CMDB in der Phase der Portfolioerkennung und -analyse der Migration.

## Konformitätspaket

Eine Sammlung von AWS Config Regeln und Abhilfemaßnahmen, die Sie zusammenstellen können, um Ihre Konformitäts- und Sicherheitsprüfungen individuell anzupassen. Mithilfe einer YAML-Vorlage können Sie ein Conformance Pack als einzelne Entität in einer AWS-Konto AND-Region oder unternehmensweit bereitstellen. Weitere Informationen finden Sie in der Dokumentation unter [Conformance Packs](#). AWS Config

## Kontinuierliche Bereitstellung und kontinuierliche Integration (CI/CD)

Der Prozess der Automatisierung der Quell-, Build-, Test-, Staging- und Produktionsphasen des Softwareveröffentlichungsprozesses. CI/CD wird allgemein als Pipeline beschrieben. CI/CD kann Ihnen helfen, Prozesse zu automatisieren, die Produktivität zu steigern, die Codequalität zu verbessern und schneller zu liefern. Weitere Informationen finden Sie unter [Vorteile der kontinuierlichen Auslieferung](#). CD kann auch für kontinuierliche Bereitstellung stehen. Weitere

Informationen finden Sie unter [Kontinuierliche Auslieferung im Vergleich zu kontinuierlicher Bereitstellung](#).

CV

Siehe [Computer Vision](#).

D

Daten im Ruhezustand

Daten, die in Ihrem Netzwerk stationär sind, z. B. Daten, die sich im Speicher befinden.

Datenklassifizierung

Ein Prozess zur Identifizierung und Kategorisierung der Daten in Ihrem Netzwerk auf der Grundlage ihrer Kritikalität und Sensitivität. Sie ist eine wichtige Komponente jeder Strategie für das Management von Cybersecurity-Risiken, da sie Ihnen hilft, die geeigneten Schutz- und Aufbewahrungskontrollen für die Daten zu bestimmen. Die Datenklassifizierung ist ein Bestandteil der Sicherheitssäule im AWS Well-Architected Framework. Weitere Informationen finden Sie unter [Datenklassifizierung](#).

Datendrift

Eine signifikante Variation zwischen den Produktionsdaten und den Daten, die zum Trainieren eines ML-Modells verwendet wurden, oder eine signifikante Änderung der Eingabedaten im Laufe der Zeit. Datendrift kann die Gesamtqualität, Genauigkeit und Fairness von ML-Modellvorhersagen beeinträchtigen.

Daten während der Übertragung

Daten, die sich aktiv durch Ihr Netzwerk bewegen, z. B. zwischen Netzwerkressourcen.

Datennetz

Ein architektonisches Framework, das verteilte, dezentrale Dateneigentum mit zentraler Verwaltung und Steuerung ermöglicht.

Datenminimierung

Das Prinzip, nur die Daten zu sammeln und zu verarbeiten, die unbedingt erforderlich sind. Durch Datenminimierung im AWS Cloud können Datenschutzrisiken, Kosten und der CO2-Fußabdruck Ihrer Analysen reduziert werden.

## Datenperimeter

Eine Reihe präventiver Schutzmaßnahmen in Ihrer AWS Umgebung, mit denen sichergestellt werden kann, dass nur vertrauenswürdige Identitäten auf vertrauenswürdige Ressourcen von erwarteten Netzwerken zugreifen. Weitere Informationen finden Sie unter [Aufbau eines Datenperimeters](#) auf AWS.

## Vorverarbeitung der Daten

Rohdaten in ein Format umzuwandeln, das von Ihrem ML-Modell problemlos verarbeitet werden kann. Die Vorverarbeitung von Daten kann bedeuten, dass bestimmte Spalten oder Zeilen entfernt und fehlende, inkonsistente oder doppelte Werte behoben werden.

## Herkunft der Daten

Der Prozess der Nachverfolgung des Ursprungs und der Geschichte von Daten während ihres gesamten Lebenszyklus, z. B. wie die Daten generiert, übertragen und gespeichert wurden.

## betreffene Person

Eine Person, deren Daten gesammelt und verarbeitet werden.

## Data Warehouse

Ein Datenverwaltungssystem, das Business Intelligence wie Analysen unterstützt. Data Warehouses enthalten in der Regel große Mengen historischer Daten und werden in der Regel für Abfragen und Analysen verwendet.

## Datenbankdefinitionssprache (DDL)

Anweisungen oder Befehle zum Erstellen oder Ändern der Struktur von Tabellen und Objekten in einer Datenbank.

## Datenbankmanipulationssprache (DML)

Anweisungen oder Befehle zum Ändern (Einfügen, Aktualisieren und Löschen) von Informationen in einer Datenbank.

## DDL

Siehe [Datenbankdefinitionssprache](#).

## Deep-Ensemble

Mehrere Deep-Learning-Modelle zur Vorhersage kombinieren. Sie können Deep-Ensembles verwenden, um eine genauere Vorhersage zu erhalten oder um die Unsicherheit von Vorhersagen abzuschätzen.

## Deep Learning

Ein ML-Teilbereich, der mehrere Schichten künstlicher neuronaler Netzwerke verwendet, um die Zuordnung zwischen Eingabedaten und Zielvariablen von Interesse zu ermitteln.

## defense-in-depth

Ein Ansatz zur Informationssicherheit, bei dem eine Reihe von Sicherheitsmechanismen und -kontrollen sorgfältig in einem Computernetzwerk verteilt werden, um die Vertraulichkeit, Integrität und Verfügbarkeit des Netzwerks und der darin enthaltenen Daten zu schützen. Wenn Sie diese Strategie anwenden AWS, fügen Sie mehrere Steuerelemente auf verschiedenen Ebenen der AWS Organizations Struktur hinzu, um die Ressourcen zu schützen. Ein defense-in-depth Ansatz könnte beispielsweise Multi-Faktor-Authentifizierung, Netzwerksegmentierung und Verschlüsselung kombinieren.

## delegierter Administrator

Ein kompatibler Dienst kann ein AWS Mitgliedskonto registrieren AWS Organizations, um die Konten der Organisation zu verwalten und die Berechtigungen für diesen Dienst zu verwalten. Dieses Konto wird als delegierter Administrator für diesen Service bezeichnet. Weitere Informationen und eine Liste kompatibler Services finden Sie unter [Services, die mit AWS Organizations funktionieren](#) in der AWS Organizations -Dokumentation.

## Bereitstellung

Der Prozess, bei dem eine Anwendung, neue Feature oder Codekorrekturen in der Zielumgebung verfügbar gemacht werden. Die Bereitstellung umfasst das Implementieren von Änderungen an einer Codebasis und das anschließende Erstellen und Ausführen dieser Codebasis in den Anwendungsumgebungen.

## Entwicklungsumgebung

Siehe [Umgebung](#).

## Detektivische Kontrolle

Eine Sicherheitskontrolle, die darauf ausgelegt ist, ein Ereignis zu erkennen, zu protokollieren und zu warnen, nachdem ein Ereignis eingetreten ist. Diese Kontrollen stellen eine zweite Verteidigungslinie dar und warnen Sie vor Sicherheitsereignissen, bei denen die vorhandenen präventiven Kontrollen umgangen wurden. Weitere Informationen finden Sie unter [Detektivische Kontrolle](#) in Implementierung von Sicherheitskontrollen in AWS.

## Abbildung des Wertstroms in der Entwicklung (DVSM)

Ein Prozess zur Identifizierung und Priorisierung von Einschränkungen, die sich negativ auf Geschwindigkeit und Qualität im Lebenszyklus der Softwareentwicklung auswirken. DVSM erweitert den Prozess der Wertstromanalyse, der ursprünglich für Lean-Manufacturing-Praktiken konzipiert wurde. Es konzentriert sich auf die Schritte und Teams, die erforderlich sind, um durch den Softwareentwicklungsprozess Mehrwert zu schaffen und zu steigern.

## digitaler Zwilling

Eine virtuelle Darstellung eines realen Systems, z. B. eines Gebäudes, einer Fabrik, einer Industrieanlage oder einer Produktionslinie. Digitale Zwillinge unterstützen vorausschauende Wartung, Fernüberwachung und Produktionsoptimierung.

## Maßtabelle

In einem [Sternschema](#) eine kleinere Tabelle, die Datenattribute zu quantitativen Daten in einer Faktentabelle enthält. Bei Attributen von Dimensionstabellen handelt es sich in der Regel um Textfelder oder diskrete Zahlen, die sich wie Text verhalten. Diese Attribute werden häufig zum Einschränken von Abfragen, zum Filtern und zur Kennzeichnung von Ergebnismengen verwendet.

## Katastrophe

Ein Ereignis, das verhindert, dass ein Workload oder ein System seine Geschäftsziele an seinem primären Einsatzort erfüllt. Diese Ereignisse können Naturkatastrophen, technische Ausfälle oder das Ergebnis menschlichen Handelns sein, z. B. unbeabsichtigte Fehlkonfigurationen oder ein Malware-Angriff.

## Disaster Recovery (DR)

Die Strategie und der Prozess, mit denen Sie Ausfallzeiten und Datenverluste aufgrund einer [Katastrophe](#) minimieren. Weitere Informationen finden Sie unter [Disaster Recovery von Workloads unter AWS: Wiederherstellung in der Cloud im AWS Well-Architected Framework](#).

## DML

Siehe Sprache zur [Datenbankmanipulation](#).

## Domainorientiertes Design

Ein Ansatz zur Entwicklung eines komplexen Softwaresystems, bei dem seine Komponenten mit sich entwickelnden Domains oder Kerngeschäftsziele verknüpft werden, denen jede Komponente dient. Dieses Konzept wurde von Eric Evans in seinem Buch Domaingesteuertes

Design: Bewältigen der Komplexität im Herzen der Software (Boston: Addison-Wesley Professional, 2003) vorgestellt. Informationen darüber, wie Sie domaingesteuertes Design mit dem Strangler-Fig-Muster verwenden können, finden Sie unter [Schrittweises Modernisieren älterer Microsoft ASP.NET \(ASMX\)-Webservices mithilfe von Containern und Amazon API Gateway](#).

## DR

Siehe [Disaster Recovery](#).

## Erkennung von Driften

Verfolgung von Abweichungen von einer Basiskonfiguration Sie können es beispielsweise verwenden, AWS CloudFormation um [Abweichungen bei den Systemressourcen zu erkennen](#), oder Sie können AWS Control Tower damit [Änderungen in Ihrer landing zone erkennen](#), die sich auf die Einhaltung von Governance-Anforderungen auswirken könnten.

## DVSM

Siehe [Abbildung der Wertströme in der Entwicklung](#).

## E

### EDA

Siehe [explorative Datenanalyse](#).

### Edge-Computing

Die Technologie, die die Rechenleistung für intelligente Geräte an den Rändern eines IoT-Netzwerks erhöht. Im Vergleich zu [Cloud Computing](#) kann Edge Computing die Kommunikationslatenz reduzieren und die Reaktionszeit verbessern.

### Verschlüsselung

Ein Rechenprozess, der Klartextdaten, die für Menschen lesbar sind, in Chiffretext umwandelt.

### Verschlüsselungsschlüssel

Eine kryptografische Zeichenfolge aus zufälligen Bits, die von einem Verschlüsselungsalgorithmus generiert wird. Schlüssel können unterschiedlich lang sein, und jeder Schlüssel ist so konzipiert, dass er unvorhersehbar und einzigartig ist.

## Endianismus

Die Reihenfolge, in der Bytes im Computerspeicher gespeichert werden. Big-Endian-Systeme speichern das höchstwertige Byte zuerst. Little-Endian-Systeme speichern das niedrigwertigste Byte zuerst.

## Endpunkt

[Siehe](#) Service-Endpunkt.

## Endpunkt-Services

Ein Service, den Sie in einer Virtual Private Cloud (VPC) hosten können, um ihn mit anderen Benutzern zu teilen. Sie können einen Endpunktdienst mit anderen AWS-Konten oder AWS Identity and Access Management (IAM AWS PrivateLink -) Prinzipalen erstellen und diesen Berechtigungen gewähren. Diese Konten oder Prinzipale können sich privat mit Ihrem Endpunktservice verbinden, indem sie Schnittstellen-VPC-Endpunkte erstellen. Weitere Informationen finden Sie unter [Einen Endpunkt-Service erstellen](#) in der Amazon Virtual Private Cloud (Amazon VPC)-Dokumentation.

## Unternehmensressourcenplanung (ERP)

Ein System, das wichtige Geschäftsprozesse (wie Buchhaltung, [MES](#) und Projektmanagement) für ein Unternehmen automatisiert und verwaltet.

## Envelope-Verschlüsselung

Der Prozess der Verschlüsselung eines Verschlüsselungsschlüssels mit einem anderen Verschlüsselungsschlüssel. Weitere Informationen finden Sie unter [Envelope-Verschlüsselung](#) in der AWS Key Management Service (AWS KMS) -Dokumentation.

## Umgebung

Eine Instance einer laufenden Anwendung. Die folgenden Arten von Umgebungen sind beim Cloud-Computing üblich:

- **Entwicklungsumgebung** – Eine Instance einer laufenden Anwendung, die nur dem Kernteam zur Verfügung steht, das für die Wartung der Anwendung verantwortlich ist. Entwicklungsumgebungen werden verwendet, um Änderungen zu testen, bevor sie in höhere Umgebungen übertragen werden. Diese Art von Umgebung wird manchmal als Testumgebung bezeichnet.
- **Niedrigere Umgebungen** – Alle Entwicklungsumgebungen für eine Anwendung, z. B. solche, die für erste Builds und Tests verwendet wurden.

- Produktionsumgebung – Eine Instance einer laufenden Anwendung, auf die Endbenutzer zugreifen können. In einer CI/CD-Pipeline ist die Produktionsumgebung die letzte Bereitstellungsumgebung.
- Höhere Umgebungen – Alle Umgebungen, auf die auch andere Benutzer als das Kernentwicklungsteam zugreifen können. Dies kann eine Produktionsumgebung, Vorproduktionsumgebungen und Umgebungen für Benutzerakzeptanztests umfassen.

## Epics

In der agilen Methodik sind dies funktionale Kategorien, die Ihnen helfen, Ihre Arbeit zu organisieren und zu priorisieren. Epics bieten eine allgemeine Beschreibung der Anforderungen und Implementierungsaufgaben. Zu den Sicherheitsthemen AWS von CAF gehören beispielsweise Identitäts- und Zugriffsmanagement, Detektivkontrollen, Infrastruktursicherheit, Datenschutz und Reaktion auf Vorfälle. Weitere Informationen zu Epics in der AWS - Migrationsstrategie finden Sie im [Leitfaden zur Programm-Implementierung](#).

## ERP

Siehe [Enterprise Resource Planning](#).

## Explorative Datenanalyse (EDA)

Der Prozess der Analyse eines Datensatzes, um seine Hauptmerkmale zu verstehen. Sie sammeln oder aggregieren Daten und führen dann erste Untersuchungen durch, um Muster zu finden, Anomalien zu erkennen und Annahmen zu überprüfen. EDA wird durchgeführt, indem zusammenfassende Statistiken berechnet und Datenvisualisierungen erstellt werden.

## F

### Faktentabelle

Die zentrale Tabelle in einem [Sternschema](#). Sie speichert quantitative Daten über den Geschäftsbetrieb. In der Regel enthält eine Faktentabelle zwei Arten von Spalten: Spalten, die Kennzahlen enthalten, und Spalten, die einen Fremdschlüssel für eine Dimensionstabelle enthalten.

### schnell scheitern

Eine Philosophie, die häufige und inkrementelle Tests verwendet, um den Entwicklungslebenszyklus zu verkürzen. Dies ist ein wichtiger Bestandteil eines agilen Ansatzes.



## Grenze zur Fehlerisolierung

Dabei handelt es sich um eine Grenze AWS Cloud, z. B. eine Availability Zone AWS-Region, eine Steuerungsebene oder eine Datenebene, die die Auswirkungen eines Fehlers begrenzt und die Widerstandsfähigkeit von Workloads verbessert. Weitere Informationen finden Sie unter [Grenzen zur AWS Fehlerisolierung](#).

## Feature-Zweig

Siehe [Zweig](#).

## Features

Die Eingabedaten, die Sie verwenden, um eine Vorhersage zu treffen. In einem Fertigungskontext könnten Feature beispielsweise Bilder sein, die regelmäßig von der Fertigungslinie aus aufgenommen werden.

## Bedeutung der Feature

Wie wichtig ein Feature für die Vorhersagen eines Modells ist. Dies wird in der Regel als numerischer Wert ausgedrückt, der mit verschiedenen Techniken wie Shapley Additive Explanations (SHAP) und integrierten Gradienten berechnet werden kann. Weitere Informationen finden Sie unter [Interpretierbarkeit von Modellen für maschinelles Lernen mit:AWS](#).

## Featuretransformation

Daten für den ML-Prozess optimieren, einschließlich der Anreicherung von Daten mit zusätzlichen Quellen, der Skalierung von Werten oder der Extraktion mehrerer Informationssätze aus einem einzigen Datenfeld. Das ermöglicht dem ML-Modell, von den Daten profitieren. Wenn Sie beispielsweise das Datum „27.05.2021 00:15:37“ in „2021“, „Mai“, „Donnerstag“ und „15“ aufschlüsseln, können Sie dem Lernalgorithmus helfen, nuancierte Muster zu erlernen, die mit verschiedenen Datenkomponenten verknüpft sind.

## FGAC

Siehe [detaillierte Zugriffskontrolle](#).

## Feinkörnige Zugriffskontrolle (FGAC)

Die Verwendung mehrerer Bedingungen, um eine Zugriffsanfrage zuzulassen oder abzulehnen.

## Flash-Cut-Migration

Eine Datenbankmigrationsmethode, bei der eine kontinuierliche Datenreplikation durch [Erfassung von Änderungsdaten](#) verwendet wird, um Daten in kürzester Zeit zu migrieren, anstatt einen schrittweisen Ansatz zu verwenden. Ziel ist es, Ausfallzeiten auf ein Minimum zu beschränken.

# G

## Geoblocking

Siehe [geografische Einschränkungen](#).

## Geografische Einschränkungen (Geoblocking)

Bei Amazon eine Option CloudFront, um zu verhindern, dass Benutzer in bestimmten Ländern auf Inhaltsverteilungen zugreifen. Sie können eine Zulassungsliste oder eine Sperrliste verwenden, um zugelassene und gesperrte Länder anzugeben. Weitere Informationen finden Sie in [der Dokumentation unter Beschränkung der geografischen Verteilung Ihrer Inhalte](#). CloudFront

## Gitflow-Workflow

Ein Ansatz, bei dem niedrigere und höhere Umgebungen unterschiedliche Zweige in einem Quellcode-Repository verwenden. Der Gitflow-Workflow gilt als veraltet, und der [Trunk-basierte Workflow](#) ist der moderne, bevorzugte Ansatz.

## Greenfield-Strategie

Das Fehlen vorhandener Infrastruktur in einer neuen Umgebung. Bei der Einführung einer Neuausrichtung einer Systemarchitektur können Sie alle neuen Technologien ohne Einschränkung der Kompatibilität mit der vorhandenen Infrastruktur auswählen, auch bekannt als [Brownfield](#). Wenn Sie die bestehende Infrastruktur erweitern, könnten Sie Brownfield- und Greenfield-Strategien mischen.

## Integritätsschutz

Eine allgemeine Regel, die dabei hilft, Ressourcen, Richtlinien und die Einhaltung von Vorschriften in allen Organisationseinheiten (OUs) zu regeln. Präventiver Integritätsschutz setzt Richtlinien durch, um die Einhaltung von Standards zu gewährleisten. Sie werden mithilfe von Service-Kontrollrichtlinien und IAM-Berechtigungsgrenzen implementiert. Detektivischer Integritätsschutz erkennt Richtlinienverstöße und Compliance-Probleme und generiert Warnmeldungen zur Abhilfe. Sie werden mithilfe von AWS Config, AWS Security Hub, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector und benutzerdefinierten AWS Lambda Prüfungen implementiert.

# H

## HEKTAR

Siehe [Hochverfügbarkeit](#).

### Heterogene Datenbankmigration

Migrieren Sie Ihre Quelldatenbank in eine Zieldatenbank, die eine andere Datenbank-Engine verwendet (z. B. Oracle zu Amazon Aurora). Eine heterogene Migration ist in der Regel Teil einer Neuarchitektur, und die Konvertierung des Schemas kann eine komplexe Aufgabe sein. [AWS bietet AWS SCT](#), welches bei Schemakonvertierungen hilft.

### hohe Verfügbarkeit (HA)

Die Fähigkeit eines Workloads, im Falle von Herausforderungen oder Katastrophen kontinuierlich und ohne Eingreifen zu arbeiten. HA-Systeme sind so konzipiert, dass sie automatisch ein Failover durchführen, eine gleichbleibend hohe Leistung bieten und unterschiedliche Lasten und Ausfälle mit minimalen Leistungseinbußen bewältigen.

### historische Modernisierung

Ein Ansatz zur Modernisierung und Aufrüstung von Betriebstechnologiesystemen (OT), um den Bedürfnissen der Fertigungsindustrie besser gerecht zu werden. Ein Historian ist eine Art von Datenbank, die verwendet wird, um Daten aus verschiedenen Quellen in einer Fabrik zu sammeln und zu speichern.

### Homogene Datenbankmigration

Migrieren Sie Ihre Quelldatenbank zu einer Zieldatenbank, die dieselbe Datenbank-Engine verwendet (z. B. Microsoft SQL Server zu Amazon RDS für SQL Server). Eine homogene Migration ist in der Regel Teil eines Hostwechsels oder eines Plattformwechsels. Sie können native Datenbankserviceprogramme verwenden, um das Schema zu migrieren.

### heiße Daten

Daten, auf die häufig zugegriffen wird, z. B. Echtzeitdaten oder aktuelle Transaktionsdaten. Für diese Daten ist in der Regel eine leistungsstarke Speicherebene oder -klasse erforderlich, um schnelle Abfrageantworten zu ermöglichen.

## Hotfix

Eine dringende Lösung für ein kritisches Problem in einer Produktionsumgebung. Aufgrund seiner Dringlichkeit wird ein Hotfix normalerweise außerhalb des typischen DevOps Release-Workflows erstellt.

## Hypercare-Phase

Unmittelbar nach dem Cutover, der Zeitraum, in dem ein Migrationsteam die migrierten Anwendungen in der Cloud verwaltet und überwacht, um etwaige Probleme zu beheben. In der Regel dauert dieser Zeitraum 1–4 Tage. Am Ende der Hypercare-Phase überträgt das Migrationsteam in der Regel die Verantwortung für die Anwendungen an das Cloud-Betriebsteam.

## I

### IaC

Sehen Sie [Infrastruktur als Code](#).

### Identitätsbasierte Richtlinie

Eine Richtlinie, die einem oder mehreren IAM-Prinzipalen zugeordnet ist und deren Berechtigungen innerhalb der AWS Cloud Umgebung definiert.

### Leerlaufanwendung

Eine Anwendung mit einer durchschnittlichen CPU- und Arbeitsspeicherauslastung zwischen 5 und 20 Prozent über einen Zeitraum von 90 Tagen. In einem Migrationsprojekt ist es üblich, diese Anwendungen außer Betrieb zu nehmen oder sie On-Premises beizubehalten.

### IIoT

Siehe [Industrielles Internet der Dinge](#).

### unveränderliche Infrastruktur

Ein Modell, das eine neue Infrastruktur für Produktionsworkloads bereitstellt, anstatt die bestehende Infrastruktur zu aktualisieren, zu patchen oder zu modifizieren. [Unveränderliche Infrastrukturen sind von Natur aus konsistenter, zuverlässiger und vorhersehbarer als veränderliche Infrastrukturen](#). Weitere Informationen finden Sie in der Best Practice [Deploy using immutable infrastructure](#) im AWS Well-Architected Framework.

## Eingehende (ingress) VPC

In einer Architektur AWS mit mehreren Konten ist dies eine VPC, die Netzwerkverbindungen von außerhalb einer Anwendung akzeptiert, überprüft und weiterleitet. Die [AWS -Referenzarchitektur für die Sicherheit](#) empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet zu schützen.

## Inkrementelle Migration

Eine Cutover-Strategie, bei der Sie Ihre Anwendung in kleinen Teilen migrieren, anstatt eine einziges vollständiges Cutover durchzuführen. Beispielsweise könnten Sie zunächst nur einige Microservices oder Benutzer auf das neue System umstellen. Nachdem Sie sich vergewissert haben, dass alles ordnungsgemäß funktioniert, können Sie weitere Microservices oder Benutzer schrittweise verschieben, bis Sie Ihr Legacy-System außer Betrieb nehmen können. Diese Strategie reduziert die mit großen Migrationen verbundenen Risiken.

## Industrie 4.0

Ein Begriff, der 2016 von [Klaus Schwab](#) eingeführt wurde und sich auf die Modernisierung von Fertigungsprozessen durch Fortschritte in den Bereichen Konnektivität, Echtzeitdaten, Automatisierung, Analytik und KI/ML bezieht.

## Infrastruktur

Alle Ressourcen und Komponenten, die in der Umgebung einer Anwendung enthalten sind.

## Infrastructure as Code (IaC)

Der Prozess der Bereitstellung und Verwaltung der Infrastruktur einer Anwendung mithilfe einer Reihe von Konfigurationsdateien. IaC soll Ihnen helfen, das Infrastrukturmanagement zu zentralisieren, Ressourcen zu standardisieren und schnell zu skalieren, sodass neue Umgebungen wiederholbar, zuverlässig und konsistent sind.

## Industrielles Internet der Dinge (IIoT)

Einsatz von mit dem Internet verbundenen Sensoren und Geräten in Industriesektoren wie Fertigung, Energie, Automobilindustrie, Gesundheitswesen, Biowissenschaften und Landwirtschaft. Mehr Informationen finden Sie unter [Aufbau einer digitalen Transformationsstrategie für das industrielle Internet der Dinge \(IIoT\)](#).

## Inspektions-VPC

In einer Architektur AWS mit mehreren Konten eine zentralisierte VPC, die Inspektionen des Netzwerkverkehrs zwischen VPCs (in derselben oder unterschiedlichen AWS-Regionen), dem

Internet und lokalen Netzwerken verwaltet. Die [AWS -Referenzarchitektur für die Sicherheit](#) empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet zu schützen.

## Internet of Things (IoT)

Das Netzwerk verbundener physischer Objekte mit eingebetteten Sensoren oder Prozessoren, das über das Internet oder über ein lokales Kommunikationsnetzwerk mit anderen Geräten und Systemen kommuniziert. Weitere Informationen finden Sie unter [Was ist IoT?](#)

## Interpretierbarkeit

Ein Merkmal eines Modells für Machine Learning, das beschreibt, inwieweit ein Mensch verstehen kann, wie die Vorhersagen des Modells von seinen Eingaben abhängen. Weitere Informationen finden Sie unter [Interpretierbarkeit von Modellen für Machine Learning mit AWS](#).

## IoT

[Siehe Internet der Dinge.](#)

## IT information library (ITIL, IT-Informationsbibliothek)

Eine Reihe von bewährten Methoden für die Bereitstellung von IT-Services und die Abstimmung dieser Services auf die Geschäftsanforderungen. ITIL bietet die Grundlage für ITSM.

## T service management (ITSM, IT-Servicemanagement)

Aktivitäten im Zusammenhang mit der Gestaltung, Implementierung, Verwaltung und Unterstützung von IT-Services für eine Organisation. Informationen zur Integration von Cloud-Vorgängen mit ITSM-Tools finden Sie im [Leitfaden zur Betriebsintegration](#).

## BIS

Siehe [IT-Informationsbibliothek](#).

## ITSM

Siehe [IT-Servicemanagement](#).

## L

### Labelbasierte Zugangskontrolle (LBAC)

Eine Implementierung der Mandatory Access Control (MAC), bei der den Benutzern und den Daten selbst jeweils explizit ein Sicherheitslabelwert zugewiesen wird. Die Schnittmenge zwischen der Benutzersicherheitsbeschriftung und der Datensicherheitsbeschriftung bestimmt, welche Zeilen und Spalten für den Benutzer sichtbar sind.

### Landing Zone

Eine landing zone ist eine gut strukturierte AWS Umgebung mit mehreren Konten, die skalierbar und sicher ist. Dies ist ein Ausgangspunkt, von dem aus Ihre Organisationen Workloads und Anwendungen schnell und mit Vertrauen in ihre Sicherheits- und Infrastrukturmgebung starten und bereitstellen können. Weitere Informationen zu Landing Zones finden Sie unter [Einrichtung einer sicheren und skalierbaren AWS -Umgebung mit mehreren Konten.](#)

### Große Migration

Eine Migration von 300 oder mehr Servern.

### SCHWARZ

Siehe [Labelbasierte Zugriffskontrolle](#).

### Geringste Berechtigung

Die bewährte Sicherheitsmethode, bei der nur die für die Durchführung einer Aufgabe erforderlichen Mindestberechtigungen erteilt werden. Weitere Informationen finden Sie unter [Geringste Berechtigungen anwenden](#) in der IAM-Dokumentation.

### Lift and Shift

Siehe [7 Rs](#).

### Little-Endian-System

Ein System, welches das niedrigwertigste Byte zuerst speichert. Siehe auch [Endianness](#).

### Niedrigere Umgebungen

[Siehe Umwelt](#).

# M

## Machine Learning (ML)

Eine Art künstlicher Intelligenz, die Algorithmen und Techniken zur Mustererkennung und zum Lernen verwendet. ML analysiert aufgezeichnete Daten, wie z. B. Daten aus dem Internet der Dinge (IoT), und lernt daraus, um ein statistisches Modell auf der Grundlage von Mustern zu erstellen. Weitere Informationen finden Sie unter [Machine Learning](#).

### Hauptzweig

Siehe [Filiale](#).

## Malware

Software, die entwickelt wurde, um die Computersicherheit oder den Datenschutz zu gefährden. Malware kann Computersysteme stören, vertrauliche Informationen durchsickern lassen oder sich unbefugten Zugriff verschaffen. Beispiele für Malware sind Viren, Würmer, Ransomware, Trojaner, Spyware und Keylogger.

### verwaltete Dienste

AWS-Services für die die Infrastrukturebene, das Betriebssystem und die Plattformen AWS betrieben werden, und Sie greifen auf die Endgeräte zu, um Daten zu speichern und abzurufen. Amazon Simple Storage Service (Amazon S3) und Amazon DynamoDB sind Beispiele für Managed Services. Diese werden auch als abstrakte Dienste bezeichnet.

## Manufacturing Execution System (MES)

Ein Softwaresystem zur Nachverfolgung, Überwachung, Dokumentation und Steuerung von Produktionsprozessen, bei denen Rohstoffe in der Fertigung zu fertigen Produkten umgewandelt werden.

## MAP

Siehe [Migration Acceleration Program](#).

## Mechanismus

Ein vollständiger Prozess, bei dem Sie ein Tool erstellen, die Akzeptanz des Tools vorantreiben und anschließend die Ergebnisse überprüfen, um Anpassungen vorzunehmen. Ein Mechanismus ist ein Zyklus, der sich im Laufe seiner Tätigkeit selbst verstärkt und verbessert. Weitere Informationen finden Sie unter [Aufbau von Mechanismen](#) im AWS Well-Architected Framework.



## Mitgliedskonto

Alle AWS-Konten außer dem Verwaltungskonto, die Teil einer Organisation in sind. AWS Organizations Ein Konto kann jeweils nur einer Organisation angehören.

## DURCHEINANDER

Siehe [Manufacturing Execution System](#).

## Message Queuing-Telemetrietransport (MQTT)

[Ein leichtes machine-to-machine \(M2M\) -Kommunikationsprotokoll, das auf dem Publish/Subscribe-Muster für IoT-Geräte mit beschränkten Ressourcen basiert.](#)

## Microservice

Ein kleiner, unabhängiger Service, der über klar definierte APIs kommuniziert und in der Regel kleinen, eigenständigen Teams gehört. Ein Versicherungssystem kann beispielsweise Microservices beinhalten, die Geschäftsfunktionen wie Vertrieb oder Marketing oder Subdomains wie Einkauf, Schadenersatz oder Analytik zugeordnet sind. Zu den Vorteilen von Microservices gehören Agilität, flexible Skalierung, einfache Bereitstellung, wiederverwendbarer Code und Ausfallsicherheit. [Weitere Informationen finden Sie unter Integration von Microservices mithilfe serverloser Dienste. AWS](#)

## Microservices-Architekturen

Ein Ansatz zur Erstellung einer Anwendung mit unabhängigen Komponenten, die jeden Anwendungsprozess als Microservice ausführen. Diese Microservices kommunizieren über eine klar definierte Schnittstelle mithilfe einfacher APIs. Jeder Microservice in dieser Architektur kann aktualisiert, bereitgestellt und skaliert werden, um den Bedarf an bestimmten Funktionen einer Anwendung zu decken. Weitere Informationen finden Sie unter [Implementieren von Microservices auf AWS](#)

## Migration Acceleration Program (MAP)

Ein AWS Programm, das Beratung, Unterstützung, Schulungen und Services bietet, um Unternehmen dabei zu unterstützen, eine solide betriebliche Grundlage für die Umstellung auf die Cloud zu schaffen und die anfänglichen Kosten von Migrationen auszugleichen. MAP umfasst eine Migrationsmethode für die methodische Durchführung von Legacy-Migrationen sowie eine Reihe von Tools zur Automatisierung und Beschleunigung gängiger Migrationsszenarien.

## Migration in großem Maßstab

Der Prozess, bei dem der Großteil des Anwendungsportfolios in Wellen in die Cloud verlagert wird, wobei in jeder Welle mehr Anwendungen schneller migriert werden. In dieser Phase werden

die bewährten Verfahren und Erkenntnisse aus den früheren Phasen zur Implementierung einer Migrationsfabrik von Teams, Tools und Prozessen zur Optimierung der Migration von Workloads durch Automatisierung und agile Bereitstellung verwendet. Dies ist die dritte Phase der [AWS - Migrationsstrategie](#).

## Migrationsfabrik

Funktionsübergreifende Teams, die die Migration von Workloads durch automatisierte, agile Ansätze optimieren. Zu den Teams in der Migrationsabteilung gehören in der Regel Betriebsabläufe, Geschäftsanalysten und Eigentümer, Migrationsingenieure, Entwickler und DevOps Experten, die in Sprints arbeiten. Zwischen 20 und 50 Prozent eines Unternehmensanwendungsportfolios bestehen aus sich wiederholenden Mustern, die durch einen Fabrik-Ansatz optimiert werden können. Weitere Informationen finden Sie in [Diskussion über Migrationsfabriken](#) und den [Leitfaden zur Cloud-Migration-Fabrik](#) in diesem Inhaltssatz.

## Migrationsmetadaten

Die Informationen über die Anwendung und den Server, die für den Abschluss der Migration benötigt werden. Für jedes Migrationsmuster ist ein anderer Satz von Migrationsmetadaten erforderlich. Beispiele für Migrationsmetadaten sind das Zielsubnetz, die Sicherheitsgruppe und AWS das Konto.

## Migrationsmuster

Eine wiederholbare Migrationsaufgabe, in der die Migrationsstrategie, das Migrationsziel und die verwendete Migrationsanwendung oder der verwendete Migrationsservice detailliert beschrieben werden. Beispiel: Rehost-Migration zu Amazon EC2 mit AWS Application Migration Service.

## Migration Portfolio Assessment (MPA)

Ein Online-Tool, das Informationen zur Validierung des Geschäftsszenarios für die Migration in die Cloud bereitstellt. AWS MPA bietet eine detaillierte Portfoliobewertung (richtige Servergröße, Preisgestaltung, Gesamtbetriebskostenanalyse, Migrationskostenanalyse) sowie Migrationsplanung (Anwendungsdatenanalyse und Datenerfassung, Anwendungsgruppierung, Migrationspriorisierung und Wellenplanung). Das [MPA-Tool](#) (Anmeldung erforderlich) steht allen AWS Beratern und APN-Partnerberatern kostenlos zur Verfügung.

## Migration Readiness Assessment (MRA)

Der Prozess, bei dem mithilfe des AWS CAF Erkenntnisse über den Cloud-Bereitschaftsstatus eines Unternehmens gewonnen, Stärken und Schwächen identifiziert und ein Aktionsplan zur Schließung festgestellter Lücken erstellt wird. Weitere Informationen finden Sie

im [Benutzerhandbuch für Migration Readiness](#). MRA ist die erste Phase der [AWS - Migrationsstrategie](#).

## Migrationsstrategie

Der Ansatz, der verwendet wird, um einen Workload in die AWS Cloud zu migrieren. Weitere Informationen finden Sie im Eintrag [7 Rs](#) in diesem Glossar und unter [Mobilisieren Sie Ihr Unternehmen, um groß angelegte Migrationen zu beschleunigen](#).

## ML

[Siehe maschinelles Lernen](#).

## Modernisierung

Umwandlung einer veralteten (veralteten oder monolithischen) Anwendung und ihrer Infrastruktur in ein agiles, elastisches und hochverfügbares System in der Cloud, um Kosten zu senken, die Effizienz zu steigern und Innovationen zu nutzen. Weitere Informationen finden Sie unter [Strategie zur Modernisierung von Anwendungen in der AWS Cloud](#).

## Bewertung der Modernisierungsfähigkeit

Eine Bewertung, anhand derer festgestellt werden kann, ob die Anwendungen einer Organisation für die Modernisierung bereit sind, Vorteile, Risiken und Abhängigkeiten identifiziert und ermittelt wird, wie gut die Organisation den zukünftigen Status dieser Anwendungen unterstützen kann. Das Ergebnis der Bewertung ist eine Vorlage der Zielarchitektur, eine Roadmap, in der die Entwicklungsphasen und Meilensteine des Modernisierungsprozesses detailliert beschrieben werden, sowie ein Aktionsplan zur Behebung festgestellter Lücken. Weitere Informationen finden Sie unter [Bewertung der Modernisierungsbereitschaft von Anwendungen in der AWS -Cloud](#).

## Monolithische Anwendungen (Monolithen)

Anwendungen, die als ein einziger Service mit eng gekoppelten Prozessen ausgeführt werden. Monolithische Anwendungen haben verschiedene Nachteile. Wenn ein Anwendungs-Feature stark nachgefragt wird, muss die gesamte Architektur skaliert werden. Das Hinzufügen oder Verbessern der Feature einer monolithischen Anwendung wird ebenfalls komplexer, wenn die Codebasis wächst. Um diese Probleme zu beheben, können Sie eine Microservices-Architektur verwenden. Weitere Informationen finden Sie unter [Zerlegen von Monolithen in Microservices](#).

## MPA

Siehe [Bewertung des Migrationsportfolios](#).

## MQTT

Siehe [Message Queuing-Telemetrietransport](#).

## Mehrklassen-Klassifizierung

Ein Prozess, der dabei hilft, Vorhersagen für mehrere Klassen zu generieren (wobei eines von mehr als zwei Ergebnissen vorhergesagt wird). Ein ML-Modell könnte beispielsweise fragen: „Ist dieses Produkt ein Buch, ein Auto oder ein Telefon?“ oder „Welche Kategorie von Produkten ist für diesen Kunden am interessantesten?“

## veränderbare Infrastruktur

Ein Modell, das die bestehende Infrastruktur für Produktionsworkloads aktualisiert und modifiziert. Für eine verbesserte Konsistenz, Zuverlässigkeit und Vorhersagbarkeit empfiehlt das AWS Well-Architected Framework die Verwendung einer [unveränderlichen Infrastruktur](#) als bewährte Methode.

## O

### OAC

Siehe [Origin Access Control](#).

### EICHE

Siehe [Zugriffsidentität von Origin](#).

### COM

Siehe [organisatorisches Change-Management](#).

## Offline-Migration

Eine Migrationsmethode, bei der der Quell-Workload während des Migrationsprozesses heruntergefahren wird. Diese Methode ist mit längeren Ausfallzeiten verbunden und wird in der Regel für kleine, unkritische Workloads verwendet.

## OI

Siehe [Betriebsintegration](#).

## OLA

Siehe Vereinbarung auf [operativer Ebene](#).

## Online-Migration

Eine Migrationsmethode, bei der der Quell-Workload auf das Zielsystem kopiert wird, ohne offline genommen zu werden. Anwendungen, die mit dem Workload verbunden sind, können während der Migration weiterhin funktionieren. Diese Methode beinhaltet keine bis minimale Ausfallzeit und wird in der Regel für kritische Produktionsworkloads verwendet.

## OPC-UA

Siehe [Open Process Communications — Unified Architecture](#).

## Offene Prozesskommunikation — Einheitliche Architektur (OPC-UA)

Ein machine-to-machine (M2M) -Kommunikationsprotokoll für die industrielle Automatisierung. OPC-UA bietet einen Interoperabilitätsstandard mit Datenverschlüsselungs-, Authentifizierungs- und Autorisierungsschemata.

## Vereinbarung auf Betriebsebene (OLA)

Eine Vereinbarung, in der klargestellt wird, welche funktionalen IT-Gruppen sich gegenseitig versprechen zu liefern, um ein Service Level Agreement (SLA) zu unterstützen.

## Überprüfung der Betriebsbereitschaft (ORR)

Eine Checkliste mit Fragen und zugehörigen bewährten Methoden, die Ihnen helfen, Vorfälle und mögliche Ausfälle zu verstehen, zu bewerten, zu verhindern oder deren Umfang zu reduzieren. Weitere Informationen finden Sie unter [Operational Readiness Reviews \(ORR\)](#) im AWS Well-Architected Framework.

## Betriebstechnologie (OT)

Hardware- und Softwaresysteme, die mit der physischen Umgebung zusammenarbeiten, um industrielle Abläufe, Ausrüstung und Infrastruktur zu steuern. In der Fertigung ist die Integration von OT- und Informationstechnologie (IT) -Systemen ein zentraler Schwerpunkt der [Industrie 4.0-Transformationen](#).

## Betriebsintegration (OI)

Der Prozess der Modernisierung von Abläufen in der Cloud, der Bereitschaftsplanung, Automatisierung und Integration umfasst. Weitere Informationen finden Sie im [Leitfaden zur Betriebsintegration](#).

## Organisationspfad

Ein Pfad, der von erstellt wird und in AWS CloudTrail dem alle Ereignisse für alle AWS-Konten in einer Organisation protokolliert werden. AWS Organizations Diese Spur wird in jedem AWS-

Konto, der Teil der Organisation ist, erstellt und verfolgt die Aktivität in jedem Konto. Weitere Informationen finden Sie in der CloudTrail Dokumentation unter [Einen Trail für eine Organisation erstellen](#).

### Organisatorisches Veränderungsmanagement (OCM)

Ein Framework für das Management wichtiger, disruptiver Geschäftstransformationen aus Sicht der Mitarbeiter, der Kultur und der Führung. OCM hilft Organisationen dabei, sich auf neue Systeme und Strategien vorzubereiten und auf diese umzustellen, indem es die Akzeptanz von Veränderungen beschleunigt, Übergangsprobleme angeht und kulturelle und organisatorische Veränderungen vorantreibt. In der AWS Migrationsstrategie wird dieses Framework aufgrund der Geschwindigkeit des Wandels, der bei Projekten zur Cloud-Einführung erforderlich ist, als Mitarbeiterbeschleunigung bezeichnet. Weitere Informationen finden Sie im [OCM-Handbuch](#).

### Ursprungszugriffskontrolle (OAC)

In CloudFront, eine erweiterte Option zur Zugriffsbeschränkung, um Ihre Amazon Simple Storage Service (Amazon S3) -Inhalte zu sichern. OAC unterstützt alle S3-Buckets insgesamt AWS-Regionen, serverseitige Verschlüsselung mit AWS KMS (SSE-KMS) sowie dynamische PUT und DELETE Anfragen an den S3-Bucket.

### Ursprungszugriffsidentität (OAI)

In CloudFront, eine Option zur Zugriffsbeschränkung, um Ihre Amazon S3 S3-Inhalte zu sichern. Wenn Sie OAI verwenden, CloudFront erstellt es einen Principal, mit dem sich Amazon S3 authentifizieren kann. Authentifizierte Principals können nur über eine bestimmte Distribution auf Inhalte in einem S3-Bucket zugreifen. CloudFront Siehe auch [OAC](#), das eine detailliertere und verbesserte Zugriffskontrolle bietet.

### ODER

Siehe [Überprüfung der Betriebsbereitschaft](#).

### NICHT

Siehe [Betriebstechnologie](#).

### Ausgehende (egress) VPC

In einer Architektur AWS mit mehreren Konten eine VPC, die Netzwerkverbindungen verarbeitet, die von einer Anwendung aus initiiert werden. Die [AWS -Referenzarchitektur für die Sicherheit](#) empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet zu schützen.

# P

## Berechtigungsgrenze

Eine IAM-Verwaltungsrichtlinie, die den IAM-Prinzipalen zugeordnet ist, um die maximalen Berechtigungen festzulegen, die der Benutzer oder die Rolle haben kann. Weitere Informationen finden Sie unter [Berechtigungsgrenzen](#) für IAM-Entitäts in der IAM-Dokumentation.

## persönlich identifizierbare Informationen (PII)

Informationen, die, wenn sie direkt betrachtet oder mit anderen verwandten Daten kombiniert werden, verwendet werden können, um vernünftige Rückschlüsse auf die Identität einer Person zu ziehen. Beispiele für personenbezogene Daten sind Namen, Adressen und Kontaktinformationen.

## Personenbezogene Daten

Siehe [persönlich identifizierbare Informationen](#).

## Playbook

Eine Reihe vordefinierter Schritte, die die mit Migrationen verbundenen Aufgaben erfassen, z. B. die Bereitstellung zentraler Betriebsfunktionen in der Cloud. Ein Playbook kann die Form von Skripten, automatisierten Runbooks oder einer Zusammenfassung der Prozesse oder Schritte annehmen, die für den Betrieb Ihrer modernisierten Umgebung erforderlich sind.

## PLC

Siehe [programmierbare Logiksteuerung](#).

## PLM

Siehe [Produktlebenszyklusmanagement](#).

## policy

Ein Objekt, das Berechtigungen definieren (siehe [identitätsbasierte Richtlinie](#)), Zugriffsbedingungen spezifizieren (siehe [ressourcenbasierte Richtlinie](#)) oder die maximalen Berechtigungen für alle Konten in einer Organisation definieren kann AWS Organizations (siehe [Dienststeuerungsrichtlinie](#)).

## Polyglotte Beharrlichkeit

Unabhängige Auswahl der Datenspeichertechnologie eines Microservices auf der Grundlage von Datenzugriffsmustern und anderen Anforderungen. Wenn Ihre Microservices über dieselbe

Datenspeichertechnologie verfügen, kann dies zu Implementierungsproblemen oder zu Leistungseinbußen führen. Microservices lassen sich leichter implementieren und erzielen eine bessere Leistung und Skalierbarkeit, wenn sie den Datenspeicher verwenden, der ihren Anforderungen am besten entspricht. Weitere Informationen finden Sie unter [Datenpersistenz in Microservices aktivieren](#).

## Portfoliobewertung

Ein Prozess, bei dem das Anwendungsportfolio ermittelt, analysiert und priorisiert wird, um die Migration zu planen. Weitere Informationen finden Sie in [Bewerten der Migrationsbereitschaft](#).

## predicate

Eine Abfragebedingung, die `true` oder zurückgibt `false`, was üblicherweise in einer Klausel vorkommt. WHERE

## Prädikat Pushdown

Eine Technik zur Optimierung von Datenbankabfragen, bei der die Daten in der Abfrage vor der Übertragung gefiltert werden. Dadurch wird die Datenmenge reduziert, die aus der relationalen Datenbank abgerufen und verarbeitet werden muss, und die Abfrageleistung wird verbessert.

## Präventive Kontrolle

Eine Sicherheitskontrolle, die verhindern soll, dass ein Ereignis eintritt. Diese Kontrollen stellen eine erste Verteidigungslinie dar, um unbefugten Zugriff oder unerwünschte Änderungen an Ihrem Netzwerk zu verhindern. Weitere Informationen finden Sie unter [Präventive Kontrolle](#) in Implementierung von Sicherheitskontrollen in AWS.

## Prinzipal

Eine Entität AWS, die Aktionen ausführen und auf Ressourcen zugreifen kann. Bei dieser Entität handelt es sich in der Regel um einen Root-Benutzer für eine AWS-Konto, eine IAM-Rolle oder einen Benutzer. Weitere Informationen finden Sie unter Prinzipal in [Rollenbegriffe und -konzepte](#) in der IAM-Dokumentation.

## Datenschutz durch Design

Ein Ansatz in der Systemtechnik, der den Datenschutz während des gesamten Engineering-Prozesses berücksichtigt.

## Privat gehostete Zonen

Ein Container, der Informationen darüber enthält, wie Amazon Route 53 auf DNS-Abfragen für eine Domain und ihre Subdomains innerhalb einer oder mehrerer VPCs reagieren soll.



Weitere Informationen finden Sie unter [Arbeiten mit privat gehosteten Zonen](#) in der Route-53-Dokumentation.

## proaktive Steuerung

Eine [Sicherheitskontrolle](#), die den Einsatz nicht richtlinienkonformer Ressourcen verhindern soll. Diese Steuerelemente scannen Ressourcen, bevor sie bereitgestellt werden. Wenn die Ressource nicht mit der Steuerung konform ist, wird sie nicht bereitgestellt. Weitere Informationen finden Sie im [Referenzhandbuch zu Kontrollen](#) in der AWS Control Tower Dokumentation und unter [Proaktive Kontrollen](#) unter Implementierung von Sicherheitskontrollen am AWS.

## Produktlebenszyklusmanagement (PLM)

Das Management von Daten und Prozessen für ein Produkt während seines gesamten Lebenszyklus, vom Design, der Entwicklung und Markteinführung über Wachstum und Reife bis hin zur Markteinführung und Markteinführung.

## Produktionsumgebung

Siehe [Umgebung](#).

## Speicherprogrammierbare Steuerung (SPS)

In der Fertigung ein äußerst zuverlässiger, anpassungsfähiger Computer, der Maschinen überwacht und Fertigungsprozesse automatisiert.

## Pseudonymisierung

Der Prozess, bei dem persönliche Identifikatoren in einem Datensatz durch Platzhalterwerte ersetzt werden. Pseudonymisierung kann zum Schutz der Privatsphäre beitragen. Pseudonymisierte Daten gelten weiterhin als personenbezogene Daten.

## veröffentlichen/abonnieren (pub/sub)

Ein Muster, das asynchrone Kommunikation zwischen Microservices ermöglicht, um die Skalierbarkeit und Reaktionsfähigkeit zu verbessern. In einem auf Microservices basierenden [MES](#) kann ein Microservice beispielsweise Ereignismeldungen in einem Kanal veröffentlichen, den andere Microservices abonnieren können. Das System kann neue Microservices hinzufügen, ohne den Veröffentlichungsservice zu ändern.

## Q

### Abfrageplan

Eine Reihe von Schritten, wie Anweisungen, die für den Zugriff auf die Daten in einem relationalen SQL-Datenbanksystem verwendet werden.

### Abfrageplanregression

Wenn ein Datenbankserviceoptimierer einen weniger optimalen Plan wählt als vor einer bestimmten Änderung der Datenbankumgebung. Dies kann durch Änderungen an Statistiken, Beschränkungen, Umgebungseinstellungen, Abfrageparameter-Bindungen und Aktualisierungen der Datenbank-Engine verursacht werden.

## R

### RACI-Matrix

Siehe [verantwortlich, rechenschaftspflichtig, konsultiert, informiert \(RACI\)](#).

### Ransomware

Eine bösartige Software, die entwickelt wurde, um den Zugriff auf ein Computersystem oder Daten zu blockieren, bis eine Zahlung erfolgt ist.

### RASCI-Matrix

Siehe [verantwortlich, rechenschaftspflichtig, konsultiert, informiert \(RACI\)](#).

### RCAC

Siehe [Zugriffskontrolle für Zeilen und Spalten](#).

### Read Replica

Eine Kopie einer Datenbank, die nur für Lesezwecke verwendet wird. Sie können Abfragen an das Lesereplikat weiterleiten, um die Belastung auf Ihrer Primärdatenbank zu reduzieren.

### neu strukturieren

Siehe [7 Rs](#).

## Recovery Point Objective (RPO)

Die maximal zulässige Zeitspanne seit dem letzten Datenwiederherstellungspunkt. Dies bestimmt, was als akzeptabler Datenverlust zwischen dem letzten Wiederherstellungspunkt und der Betriebsunterbrechung angesehen wird.

## Ziel der Wiederherstellungszeit (RTO)

Die maximal zulässige Verzögerung zwischen der Betriebsunterbrechung und der Wiederherstellung des Dienstes.

## Refaktorisierung

Siehe [7 Rs.](#)

## Region

Eine Sammlung von AWS Ressourcen in einem geografischen Gebiet. Jeder AWS-Region ist isoliert und unabhängig von den anderen, um Fehlertoleranz, Stabilität und Belastbarkeit zu gewährleisten. Weitere Informationen finden [Sie unter Geben Sie an, was AWS-Regionen Ihr Konto verwenden kann.](#)

## Regression

Eine ML-Technik, die einen numerischen Wert vorhersagt. Zum Beispiel, um das Problem „Zu welchem Preis wird dieses Haus verkauft werden?“ zu lösen Ein ML-Modell könnte ein lineares Regressionsmodell verwenden, um den Verkaufspreis eines Hauses auf der Grundlage bekannter Fakten über das Haus (z. B. die Quadratmeterzahl) vorherzusagen.

## rehosten

Siehe [7 Rs.](#)

## Veröffentlichung

In einem Bereitstellungsprozess der Akt der Förderung von Änderungen an einer Produktionsumgebung.

## umziehen

Siehe [7 Rs.](#)

## neue Plattform

Siehe [7 Rs.](#)

## Rückkauf

Siehe [7 Rs.](#)

## Ausfallsicherheit

Die Fähigkeit einer Anwendung, Störungen zu widerstehen oder sich von ihnen zu erholen. [Hochverfügbarkeit](#) und [Notfallwiederherstellung](#) sind häufig Überlegungen bei der Planung der Ausfallsicherheit in der. AWS Cloud Weitere Informationen finden Sie unter [AWS Cloud Resilienz](#).

## Ressourcenbasierte Richtlinie

Eine mit einer Ressource verknüpfte Richtlinie, z. B. ein Amazon-S3-Bucket, ein Endpunkt oder ein Verschlüsselungsschlüssel. Diese Art von Richtlinie legt fest, welchen Prinzipalen der Zugriff gewährt wird, welche Aktionen unterstützt werden und welche anderen Bedingungen erfüllt sein müssen.

## RACI-Matrix (verantwortlich, rechenschaftspflichtig, konsultiert, informiert)

Eine Matrix, die die Rollen und Verantwortlichkeiten für alle Parteien definiert, die an Migrationsaktivitäten und Cloud-Vorgängen beteiligt sind. Der Matrixname leitet sich von den in der Matrix definierten Zuständigkeitstypen ab: verantwortlich (R), rechenschaftspflichtig (A), konsultiert (C) und informiert (I). Der Unterstützungstyp (S) ist optional. Wenn Sie Unterstützung einbeziehen, wird die Matrix als RASCI-Matrix bezeichnet, und wenn Sie sie ausschließen, wird sie als RACI-Matrix bezeichnet.

## Reaktive Kontrolle

Eine Sicherheitskontrolle, die darauf ausgelegt ist, die Behebung unerwünschter Ereignisse oder Abweichungen von Ihren Sicherheitsstandards voranzutreiben. Weitere Informationen finden Sie unter [Reaktive Kontrolle](#) in Implementieren von Sicherheitskontrollen in AWS.

## Beibehaltung

Siehe [7 Rs.](#)

## zurückziehen

Siehe [7 Rs.](#)

## Drehung

Der Vorgang, bei dem ein [Geheimnis](#) regelmäßig aktualisiert wird, um es einem Angreifer zu erschweren, auf die Anmeldeinformationen zuzugreifen.

## Zugriffskontrolle für Zeilen und Spalten (RCAC)

Die Verwendung einfacher, flexibler SQL-Ausdrücke mit definierten Zugriffsregeln. RCAC besteht aus Zeilenberechtigungen und Spaltenmasken.

## RPO

Siehe [Recovery Point Objective](#).

## RTO

Siehe [Ziel der Wiederherstellungszeit](#).

## Runbook

Eine Reihe manueller oder automatisierter Verfahren, die zur Ausführung einer bestimmten Aufgabe erforderlich sind. Diese sind in der Regel darauf ausgelegt, sich wiederholende Operationen oder Verfahren mit hohen Fehlerquoten zu rationalisieren.

# S

## SAML 2.0

Ein offener Standard, den viele Identitätsanbieter (IdPs) verwenden. Diese Funktion ermöglicht föderiertes Single Sign-On (SSO), sodass sich Benutzer bei den API-Vorgängen anmelden AWS Management Console oder die AWS API-Operationen aufrufen können, ohne dass Sie einen Benutzer in IAM für alle in Ihrer Organisation erstellen müssen. Weitere Informationen zum SAML-2.0.-basierten Verbund finden Sie unter [Über den SAML-2.0-basierten Verbund](#) in der IAM-Dokumentation.

## SCADA

Siehe [Aufsichtskontrolle und Datenerfassung](#).

## SCP

Siehe [Richtlinie zur Dienstkontrolle](#).

## Secret

Interne AWS Secrets Manager, vertrauliche oder eingeschränkte Informationen, wie z. B. ein Passwort oder Benutzeranmeldedaten, die Sie in verschlüsselter Form speichern. Es besteht aus dem geheimen Wert und seinen Metadaten. Der geheime Wert kann binär, eine einzelne

Zeichenfolge oder mehrere Zeichenketten sein. Weitere Informationen finden Sie unter [Secret](#) in der Secrets Manager-Dokumentation.

### Sicherheitskontrolle

Ein technischer oder administrativer Integritätsschutz, der die Fähigkeit eines Bedrohungsakteurs, eine Schwachstelle auszunutzen, verhindert, erkennt oder einschränkt. Es gibt vier Haupttypen von Sicherheitskontrollen: [präventiv](#), [detektiv](#), [reaktionsschnell](#) und [proaktiv](#).

### Härtung der Sicherheit

Der Prozess, bei dem die Angriffsfläche reduziert wird, um sie widerstandsfähiger gegen Angriffe zu machen. Dies kann Aktionen wie das Entfernen von Ressourcen, die nicht mehr benötigt werden, die Implementierung der bewährten Sicherheitsmethode der Gewährung geringster Berechtigungen oder die Deaktivierung unnötiger Feature in Konfigurationsdateien umfassen.

### System zur Verwaltung von Sicherheitsinformationen und Ereignissen (security information and event management – SIEM)

Tools und Services, die Systeme für das Sicherheitsinformationsmanagement (SIM) und das Management von Sicherheitsereignissen (SEM) kombinieren. Ein SIEM-System sammelt, überwacht und analysiert Daten von Servern, Netzwerken, Geräten und anderen Quellen, um Bedrohungen und Sicherheitsverletzungen zu erkennen und Warnmeldungen zu generieren.

### Automatisierung von Sicherheitsreaktionen

Eine vordefinierte und programmierte Aktion, die darauf ausgelegt ist, automatisch auf ein Sicherheitsereignis zu reagieren oder es zu beheben. Diese Automatisierungen dienen als [detektive](#) oder [reaktionsschnelle](#) Sicherheitskontrollen, die Sie bei der Implementierung bewährter AWS Sicherheitsmethoden unterstützen. Beispiele für automatisierte Antwortaktionen sind das Ändern einer VPC-Sicherheitsgruppe, das Patchen einer Amazon EC2 EC2-Instance oder das Rotieren von Anmeldeinformationen.

### Serverseitige Verschlüsselung

Verschlüsselung von Daten am Zielort durch denjenigen AWS-Service, der sie empfängt.

### Service-Kontrollrichtlinie (SCP)

Eine Richtlinie, die eine zentrale Kontrolle über die Berechtigungen für alle Konten in einer Organisation in AWS Organizations ermöglicht. SCPs definieren Integritätsschutz oder legen Grenzwerte für Aktionen fest, die ein Administrator an Benutzer oder Rollen delegieren kann. Sie können SCPs als Zulassungs- oder Ablehnungslisten verwenden, um festzulegen, welche

Services oder Aktionen zulässig oder verboten sind. Weitere Informationen finden Sie in der AWS Organizations Dokumentation unter [Richtlinien zur Dienststeuerung](#).

### Service-Endpunkt

Die URL des Einstiegspunkts für einen AWS-Service. Sie können den Endpunkt verwenden, um programmgesteuert eine Verbindung zum Zielservice herzustellen. Weitere Informationen finden Sie unter [AWS-Service -Endpunkte](#) in der Allgemeine AWS-Referenz.

### Service Level Agreement (SLA)

Eine Vereinbarung, in der klargestellt wird, was ein IT-Team seinen Kunden zu bieten verspricht, z. B. in Bezug auf Verfügbarkeit und Leistung der Services.

### Service-Level-Indikator (SLI)

Eine Messung eines Leistungsaspekts eines Dienstes, z. B. seiner Fehlerrate, Verfügbarkeit oder Durchsatz.

### Service-Level-Ziel (SLO)

Eine Zielkennzahl, die den Zustand eines Dienstes darstellt, gemessen anhand eines [Service-Level-Indikators](#).

### Modell der geteilten Verantwortung

Ein Modell, das die Verantwortung beschreibt, mit der Sie gemeinsam AWS für Cloud-Sicherheit und Compliance verantwortlich sind. AWS ist für die Sicherheit der Cloud verantwortlich, wohingegen Sie für die Sicherheit in der Cloud verantwortlich sind. Weitere Informationen finden Sie unter [Modell der geteilten Verantwortung](#).

### SIEM

Siehe [Sicherheitsinformations- und Event-Management-System](#).

### Single Point of Failure (SPOF)

Ein Fehler in einer einzelnen, kritischen Komponente einer Anwendung, der das System stören kann.

### SLA

Siehe [Service Level Agreement](#).

### SLI

Siehe [Service-Level-Indikator](#).

## ALSO

Siehe [Service-Level-Ziel](#).

### split-and-seed Modell

Ein Muster für die Skalierung und Beschleunigung von Modernisierungsprojekten. Sobald neue Features und Produktversionen definiert werden, teilt sich das Kernteam auf, um neue Produktteams zu bilden. Dies trägt zur Skalierung der Fähigkeiten und Services Ihrer Organisation bei, verbessert die Produktivität der Entwickler und unterstützt schnelle Innovationen. Weitere Informationen finden Sie unter [Schrittweiser Ansatz zur Modernisierung von Anwendungen in der AWS Cloud](#)

## SPOTTEN

Siehe [Single Point of Failure](#).

### Sternschema

Eine Datenbank-Organisationsstruktur, die eine große Faktentabelle zum Speichern von Transaktions- oder Messdaten und eine oder mehrere kleinere dimensionale Tabellen zum Speichern von Datenattributen verwendet. Diese Struktur ist für die Verwendung in einem [Data Warehouse](#) oder für Business Intelligence-Zwecke konzipiert.

### Strangler-Fig-Muster

Ein Ansatz zur Modernisierung monolithischer Systeme, bei dem die Systemfunktionen schrittweise umgeschrieben und ersetzt werden, bis das Legacy-System außer Betrieb genommen werden kann. Dieses Muster verwendet die Analogie einer Feigenrebe, die zu einem etablierten Baum heranwächst und schließlich ihren Wirt überwindet und ersetzt. Das Muster wurde [eingeführt von Martin Fowler](#) als Möglichkeit, Risiken beim Umschreiben monolithischer Systeme zu managen. Ein Beispiel für die Anwendung dieses Musters finden Sie unter [Schrittweises Modernisieren älterer Microsoft ASP.NET \(ASMX\)-Webservices mithilfe von Containern und Amazon API Gateway](#).

### Subnetz

Ein Bereich von IP-Adressen in Ihrer VPC. Ein Subnetz muss sich in einer einzigen Availability Zone befinden.

### Aufsichtskontrolle und Datenerfassung (SCADA)

In der Fertigung ein System, das Hardware und Software zur Überwachung von Sachanlagen und Produktionsabläufen verwendet.



## Symmetrische Verschlüsselung

Ein Verschlüsselungsalgorithmus, der denselben Schlüssel zum Verschlüsseln und Entschlüsseln der Daten verwendet.

## synthetisches Testen

Testen eines Systems auf eine Weise, die Benutzerinteraktionen simuliert, um potenzielle Probleme zu erkennen oder die Leistung zu überwachen. Sie können [Amazon CloudWatch Synthetics](#) verwenden, um diese Tests zu erstellen.

# T

## tags

Schlüssel-Wert-Paare, die als Metadaten für die Organisation Ihrer Ressourcen dienen. AWS Mit Tags können Sie Ressourcen verwalten, identifizieren, organisieren, suchen und filtern. Weitere Informationen finden Sie unter [Markieren Ihrer AWS -Ressourcen](#).

## Zielvariable

Der Wert, den Sie in überwachtem ML vorhersagen möchten. Dies wird auch als Ergebnisvariable bezeichnet. In einer Fertigungsumgebung könnte die Zielvariable beispielsweise ein Produktfehler sein.

## Aufgabenliste

Ein Tool, das verwendet wird, um den Fortschritt anhand eines Runbooks zu verfolgen. Eine Aufgabenliste enthält eine Übersicht über das Runbook und eine Liste mit allgemeinen Aufgaben, die erledigt werden müssen. Für jede allgemeine Aufgabe werden der geschätzte Zeitaufwand, der Eigentümer und der Fortschritt angegeben.

## Testumgebungen

[Siehe Umgebung.](#)

## Training

Daten für Ihr ML-Modell bereitstellen, aus denen es lernen kann. Die Trainingsdaten müssen die richtige Antwort enthalten. Der Lernalgorithmus findet Muster in den Trainingsdaten, die die Attribute der Input-Daten dem Ziel (die Antwort, die Sie voraussagen möchten) zuordnen. Es gibt ein ML-Modell aus, das diese Muster erfasst. Sie können dann das ML-Modell verwenden, um Voraussagen für neue Daten zu erhalten, bei denen Sie das Ziel nicht kennen.

## Transit-Gateway

Ein Transit-Gateway ist ein Netzwerk-Transit-Hub, mit dem Sie Ihre VPCs und On-Premises-Netzwerke miteinander verbinden können. Weitere Informationen finden Sie in der AWS Transit Gateway Dokumentation unter [Was ist ein Transit-Gateway](#).

## Stammbasierter Workflow

Ein Ansatz, bei dem Entwickler Feature lokal in einem Feature-Zweig erstellen und testen und diese Änderungen dann im Hauptzweig zusammenführen. Der Hauptzweig wird dann sequentiell für die Entwicklungs-, Vorproduktions- und Produktionsumgebungen erstellt.

## Vertrauenswürdiger Zugriff

Gewährung von Berechtigungen für einen Dienst, den Sie angeben, um Aufgaben in Ihrer Organisation AWS Organizations und in deren Konten in Ihrem Namen auszuführen. Der vertrauenswürdige Service erstellt in jedem Konto eine mit dem Service verknüpfte Rolle, wenn diese Rolle benötigt wird, um Verwaltungsaufgaben für Sie auszuführen. Weitere Informationen finden Sie in der AWS Organizations Dokumentation [unter Verwendung AWS Organizations mit anderen AWS Diensten](#).

## Optimieren

Aspekte Ihres Trainingsprozesses ändern, um die Genauigkeit des ML-Modells zu verbessern. Sie können das ML-Modell z. B. trainieren, indem Sie einen Beschriftungssatz generieren, Beschriftungen hinzufügen und diese Schritte dann mehrmals unter verschiedenen Einstellungen wiederholen, um das Modell zu optimieren.

## Zwei-Pizzen-Team

Ein kleines DevOps Team, das Sie mit zwei Pizzen ernähren können. Eine Teamgröße von zwei Pizzen gewährleistet die bestmögliche Gelegenheit zur Zusammenarbeit bei der Softwareentwicklung.

# U

## Unsicherheit

Ein Konzept, das sich auf ungenaue, unvollständige oder unbekanntere Informationen bezieht, die die Zuverlässigkeit von prädiktiven ML-Modellen untergraben können. Es gibt zwei Arten von Unsicherheit: Epistemische Unsicherheit wird durch begrenzte, unvollständige Daten verursacht,

wohingegen aleatorische Unsicherheit durch Rauschen und Randomisierung verursacht wird, die in den Daten liegt. Weitere Informationen finden Sie im Leitfaden [Quantifizieren der Unsicherheit in Deep-Learning-Systemen](#).

## undifferenzierte Aufgaben

Diese Arbeit wird auch als Schwerstarbeit bezeichnet. Dabei handelt es sich um Arbeiten, die zwar für die Erstellung und den Betrieb einer Anwendung erforderlich sind, aber dem Endbenutzer keinen direkten Mehrwert bieten oder keinen Wettbewerbsvorteil bieten. Beispiele für undifferenzierte Aufgaben sind Beschaffung, Wartung und Kapazitätsplanung.

## höhere Umgebungen

Siehe [Umgebung](#).

# V

## Vacuuming

Ein Vorgang zur Datenbankwartung, bei dem die Datenbank nach inkrementellen Aktualisierungen bereinigt wird, um Speicherplatz zurückzugewinnen und die Leistung zu verbessern.

## Versionskontrolle

Prozesse und Tools zur Nachverfolgung von Änderungen, z. B. Änderungen am Quellcode in einem Repository.

## VPC-Peering

Eine Verbindung zwischen zwei VPCs, mit der Sie den Datenverkehr mithilfe von privaten IP-Adressen weiterleiten können. Weitere Informationen finden Sie unter [Was ist VPC-Peering?](#) in der Amazon-VPC-Dokumentation.

## Schwachstelle

Ein Software- oder Hardwarefehler, der die Sicherheit des Systems gefährdet.

# W

## Warmer Cache

Ein Puffer-Cache, der aktuelle, relevante Daten enthält, auf die häufig zugegriffen wird. Die Datenbank-Instance kann aus dem Puffer-Cache lesen, was schneller ist als das Lesen aus dem Hauptspeicher oder von der Festplatte.

## warme Daten

Daten, auf die selten zugegriffen wird. Bei der Abfrage dieser Art von Daten sind mäßig langsame Abfragen in der Regel akzeptabel.

## Fensterfunktion

Eine SQL-Funktion, die eine Berechnung für eine Gruppe von Zeilen durchführt, die sich in irgendeiner Weise auf den aktuellen Datensatz beziehen. Fensterfunktionen sind nützlich für die Verarbeitung von Aufgaben wie die Berechnung eines gleitenden Durchschnitts oder für den Zugriff auf den Wert von Zeilen auf der Grundlage der relativen Position der aktuellen Zeile.

## Workload

Ein Workload ist eine Sammlung von Ressourcen und Code, die einen Unternehmenswert bietet, wie z. B. eine kundenorientierte Anwendung oder ein Backend-Prozess.

## Workstream

Funktionsgruppen in einem Migrationsprojekt, die für eine bestimmte Reihe von Aufgaben verantwortlich sind. Jeder Workstream ist unabhängig, unterstützt aber die anderen Workstreams im Projekt. Der Portfolio-Workstream ist beispielsweise für die Priorisierung von Anwendungen, die Wellenplanung und die Erfassung von Migrationsmetadaten verantwortlich. Der Portfolio-Workstream liefert diese Komponenten an den Migrations-Workstream, der dann die Server und Anwendungen migriert.

## WURM

[Mal schreiben, viele lesen.](#)

## WQF

Weitere Informationen finden Sie unter [AWS Workload Qualification Framework.](#)

## einmal schreiben, viele lesen (WORM)

Ein Speichermodell, das Daten ein einziges Mal schreibt und verhindert, dass die Daten gelöscht oder geändert werden. Autorisierte Benutzer können die Daten so oft wie nötig lesen, aber sie können sie nicht ändern. Diese Datenspeicherinfrastruktur wird als [unveränderlich](#) angesehen.

## Z

### Zero-Day-Exploit

Ein Angriff, in der Regel Malware, der eine [Zero-Day-Sicherheitslücke](#) ausnutzt.

### Zero-Day-Sicherheitslücke

Ein unfehlbarer Fehler oder eine Sicherheitslücke in einem Produktionssystem. Bedrohungsakteure können diese Art von Sicherheitslücke nutzen, um das System anzugreifen. Entwickler werden aufgrund des Angriffs häufig auf die Sicherheitsanfälligkeit aufmerksam.

### Zombie-Anwendung

Eine Anwendung, deren durchschnittliche CPU- und Arbeitsspeichernutzung unter 5 Prozent liegt. In einem Migrationsprojekt ist es üblich, diese Anwendungen außer Betrieb zu nehmen.

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.