Erweiterte AMS-Anwendungsbereitstellungsoptionen

AMS-Leitfaden für fortgeschrittene Anwendungsentwickler

Version September 13, 2024

AMS-Leitfaden für fortgeschrittene Anwendungsentwickler: Erweiterte AMS-Anwendungsbereitstellungsoptionen

Table of Contents

Onboarding von Anwendungen

Willkommen beim AMS-Betriebsplan von AWS Managed Services (AMS). In diesem Dokument werden die verschiedenen Methoden beschrieben, die Sie beim Onboarding Ihrer Anwendungen in AMS verwenden können, sobald die anfängliche Netzwerk- und Zugriffsverwaltung eingerichtet wurde, sowie die Aspekte, die Sie bei der Auswahl dieser Methoden berücksichtigen sollten.

Dieses Dokument richtet sich an Systemintegratoren und Anwendungsentwickler, um sie bei der Festlegung und Gestaltung von Anwendungsprozessen für neue AMS-Kunden zu unterstützen.

Was ist Anwendungs-Onboarding?

Das Onboarding von AMS-Anwendungen bezieht sich auf die Bereitstellung von Ressourcen und Anwendungen nach Bedarf in Ihrer AMS-Infrastruktur. Die Architektur von Anwendungen und Infrastruktur auf der AMS-Plattform ist der Architektur auf nativer Ebene sehr ähnlich. AWS Wenn Sie sich an bewährte Methoden für das AWS Anwendungs- und Infrastrukturdesign halten und gleichzeitig die von AMS bereitgestellten Funktionen berücksichtigen, werden Sie leistungsfähige und funktionsfähige Anwendungen erhalten, die in der AMS-Umgebung gehostet werden.

Note

- USA Ost (Virginia)
- USA West (Nordkalifornien)
- USA West (Oregon)
- USA Ost (Ohio)
- Kanada (Zentral)
- Südamerika (São Paulo)
- EU (Irland)
- EU (Frankfurt)
- EU (London)
- EU West (Paris)
- Asien-Pazifik (Mumbai)
- Asia Pacific (Seoul)
- Asien-Pazifik (Singapur)

- Asien-Pazifik (Sydney)
- Asien-Pazifik (Tokio)

Neue Regionen werden häufig hinzugefügt. Weitere Informationen finden Sie unter AWS-Regionen Availability Zones.

Was wir tun, was wir nicht tun

AMS bietet Ihnen einen standardisierten Ansatz für die Bereitstellung der AWS-Infrastruktur und bietet das erforderliche kontinuierliche Betriebsmanagement. Eine vollständige Beschreibung der Rollen, Verantwortlichkeiten und unterstützten Services finden Sie in der Servicebeschreibung.

> **Note**
>
> Um AMS aufzufordern, einen zusätzlichen AWS-Service bereitzustellen, reichen Sie eine Serviceanfrage ein. Weitere Informationen finden Sie unter Serviceanfragen stellen.

- Was wir tun:

  Nachdem Sie das Onboarding abgeschlossen haben, können Sie in der AMS-Umgebung Änderungsanfragen (RFCs), Incidents und Serviceanfragen entgegennehmen. Ihre Interaktion mit dem AMS-Service dreht sich um den Lebenszyklus eines Anwendungsstapels. Neue Stacks werden anhand einer vorkonfigurierten Liste von Vorlagen bestellt, in bestimmten Virtual Private Cloud (VPC) -Subnetzen gestartet, während ihrer Betriebsdauer durch Änderungsanforderungen (RFCs) geändert und rund um die Uhr auf Ereignisse und Vorfälle überwacht.

  Aktive Anwendungs-Stacks werden von AMS überwacht und verwaltet, einschließlich Patches, und erfordern während der gesamten Lebensdauer des Stacks keine weiteren Maßnahmen, es sei denn, eine Änderung ist erforderlich oder der Stack wird außer Betrieb genommen. Von AMS festgestellte Vorfälle, die den Zustand und die Funktion des Stacks beeinträchtigen, wird eine Benachrichtigung generiert. Möglicherweise müssen Sie Maßnahmen ergreifen, um diese zu beheben oder zu überprüfen. Fragen zu Anleitungen und andere Anfragen können gestellt werden, indem Sie eine Serviceanfrage einreichen.

Darüber hinaus können Sie mit AMS kompatible AWS-Services aktivieren, die nicht von AMS verwaltet werden. Informationen zu AWS-AMS-kompatiblen Diensten finden Sie unter Self-Service-Bereitstellungsmodus.

- Was wir NICHT tun:

  AMS vereinfacht zwar die Anwendungsbereitstellung durch die Bereitstellung einer Reihe manueller und automatisierter Optionen, Sie sind jedoch für die Entwicklung, das Testen, die Aktualisierung und das Management Ihrer Anwendung verantwortlich. AMS bietet Unterstützung bei der Behebung von Infrastrukturproblemen, die sich auf Anwendungen auswirken, AMS kann jedoch nicht auf Ihre Anwendungskonfigurationen zugreifen oder diese validieren.

AMS Amazon-Maschinenbilder (AMIs)

AMS erstellt jeden Monat aktualisierte Amazon Machine Images (AMIs) für AMS-unterstützte Betriebssysteme. Darüber hinaus produziert AMS auch sicherheitsoptimierte Images (AMIs), die auf dem CIS Level 1-Benchmark für eine Untergruppe der von AMS unterstützten Betriebssysteme basieren. Informationen darüber, für welche Betriebssysteme ein Image mit verbesserter Sicherheit verfügbar ist, finden Sie im AMS Security User Guide, das auf der Seite AWS Artifact -> Berichte (finden Sie die Option Berichte im linken Navigationsbereich) verfügbar ist und nach AWS Managed Services gefiltert ist. Um auf AWS Artifact zuzugreifen, können Sie sich an Ihren CSDM wenden, um Anweisungen zu erhalten, oder gehen Sie zu Erste Schritte mit AWS Artifact.

Um Benachrichtigungen zu erhalten, wenn neue AMS veröffentlicht AMIs werden, können Sie ein Amazon Simple Notification Service (Amazon SNS) -Benachrichtigungsthema namens „AMS AMI“ abonnieren. Einzelheiten finden Sie unter AMS-AMI-Benachrichtigungen mit SNS.

Die AMS-AMI-Namenskonvention lautet:`customer-ams-<operating system>-<release date> - <version>`. (zum Beispiel`customer-ams-rhel6-2018.11-3`)

Verwenden Sie nur AMS AMIs , die mit beginnen`customer`.

AMS empfiehlt, immer das neueste AMI zu verwenden. Das neueste finden Sie auf einer der AMIs folgenden Arten:

- Schauen Sie in der AMS-Konsole auf der AMIsSeite nach.

- Anzeige der neuesten AMS AMI-CSV-Datei, verfügbar in Ihrem CSDM oder über diese ZIP-Datei: AMS 11.2024 AMI-Inhalt und CSV-Datei in einer ZIP-Datei.

  Frühere AMI-ZIP-Dateien finden Sie in der Dokumentenhistorie.

- Diesen `SKMS` AMS-Befehl ausführen (AMS SKMS SDK erforderlich):

```
aws amsskms list-amis --vpc-id VPC_ID --query "Amis.sort_by(@,&Name)[?
 starts_with(Name,'customer')].[Name,AmiId,CreationTime]" --output table
```

AMS AMI-Inhalt zur Basis hinzugefügt AWS AMIs, nach Betriebssystem (OS)

- Linux AMIs:
  - AWS CLI-Tools
  - NTP
  - Trend Micro Endpoint Protection Service Agent
  - Code verteilen
  - PBIS//AD Bridge ist mehr als vertrauenswürdig
  - SSM-Agent
  - Yum Upgrade für kritische Patches
  - Benutzerdefinierte AMS-Skripts/Verwaltungssoftware (Steuerung von Start, AD-Join, Überwachung, Sicherheit und Protokollierung)
- Windows-Server AMIs:
  - Microsoft.NET Framework 4.5
  - PowerShell 5.1
  - AWS Tools für Windows PowerShell
  - PowerShell AMS-Module zur Steuerung von Start, AD-Join, Überwachung, Sicherheit und Protokollierung
  - Trend Micro Endpoint Protection Service Agent
  - SSM-Agent
  - CloudWatch Kundendienstmitarbeiter
  - EC2Konfigurationsdienst (über Windows Server 2012 R2)
  - EC2Starten (Windows Server 2016 und Windows Server 2019)
  - EC2LaunchV2 (Windows Server 2022 und höher)

Linux-basiert: AMIs

- Amazon Linux 2023 (neueste Nebenversion) (Minimales AMI wird nicht unterstützt)
- Amazon Linux 2 (neueste Nebenversion)
- Amazon Linux (2ARM64)
- Red Hat Enterprise 7 (neueste Nebenversion)
- Red Hat Enterprise 8 (neueste Nebenversion)
- Red Hat Enterprise 9 (neueste Nebenversion)
- SUSE Linux Enterprise Server 15 SP6
- Ubuntu Linux 18.04
- Ubuntu Linux 20.04
- Ubuntu Linux 22.04
- Ubuntu Linux 24.04
- Amazon Linux: Eine Produktübersicht, Preisinformationen, Nutzungsinformationen und Supportinformationen finden Sie unter Amazon Linux AMI (HVM/64-Bit) und Amazon Linux 2.

  Weitere Informationen finden Sie unter Amazon Linux FAQs 2.
- RedHat Enterprise Linux (RHEL): Eine Produktübersicht, Preisinformationen, Nutzungsinformationen und Supportinformationen finden Sie unter Red Hat Enterprise Linux (RHEL) 7 (HVM).
- Ubuntu Linux 18.04: Eine Produktübersicht, Preisinformationen, Nutzungsinformationen und Supportinformationen finden Sie unter Ubuntu 18.04 LTS — Bionic.
- SUSE Linux Enterprise Server für SAP-Anwendungen 15: SP6
  - Führen Sie die folgenden Schritte einmal pro Konto aus:
    1. Navigieren Sie zur AWS Marketplace.
    2. Suchen Sie nach dem SUSE 15 SAP-Produkt.
    3. Wählen Sie Weiter, um ein Abonnement abzuschließen.
    4. Wählen Sie Bedingungen akzeptieren aus.
  - Führen Sie jedes Mal, wenn Sie eine neue SUSE Linux Enterprise Server for SAP Applications SP6 15-Instanz starten müssen, die folgenden Schritte aus:
    1. Notieren Sie sich die AMI-ID für das abonnierte SUSE Linux Enterprise Server for SAP Applications 15 AMI.

2. Eine Bereitstellung erstellen | Erweiterte Stack-Komponenten | EC2 Stack | Änderungstyp erstellen ct-14027q0sjyt1h RFC. *`InstanceAmiId`*Ersetzen Sie es durch die AWS Marketplace AMI-ID, die Sie abonniert haben.

Windows-basiert AMIs:

Microsoft Windows Server (2016, 2019 und 2022), basierend auf dem neuesten Windows AMIs.

Beispiele für die Erstellung AMIs finden Sie unter Create AMI.

AMS AMIs auslagern:

AMS entzieht Ihnen während des Offboardings keine AMIs Informationen, um Auswirkungen auf Ihre Abhängigkeiten zu vermeiden. Wenn Sie AMS AMIs aus Ihrem Konto entfernen möchten, können Sie die `cancel-image-launch-permission` API verwenden, um bestimmte Informationen zu verbergen. AMIs Sie können beispielsweise das folgende Skript verwenden, um alle AMS auszublenden AMIs , die zuvor mit Ihrem Konto geteilt wurden:

```
for ami in $(aws ec2 describe-images --executable-users self --owners 027415890775 --
query 'Images[].ImageId' --output text) ;
    do
    aws ec2 cancel-image-launch-permission --image-id $ami ;
    done
```

Sie müssen die AWS CLI v2 installiert haben, damit das Skript fehlerfrei ausgeführt werden kann. Die Installationsschritte für AWS CLI finden Sie unter Installation oder Aktualisierung der neuesten Version der AWS-CLI. Einzelheiten zu dem `cancel-image-launch-permission` Befehl finden Sie unter `cancel-image-launch-permission`.

Verbesserte Sicherheit AMIs

AMS stellt für eine Untergruppe der von AMS unterstützten Betriebssysteme sicherheitsoptimierte Images (AMIs) bereit, die auf dem CIS Level 1-Benchmark basieren. Informationen darüber, für welche Betriebssysteme ein Image mit verbesserter Sicherheit verfügbar ist, finden Sie im Kundensicherheitshandbuch für AWS Managed Services (AMS). Um auf dieses Handbuch zuzugreifen, öffnen Sie es AWS Artifact, wählen Sie im linken Navigationsbereich Berichte aus und filtern Sie dann nach AWS Managed Services. Anweisungen zum Zugriff erhalten AWS Artifact Sie bei Ihrem CSDM oder unter Erste Schritte mit AWS Artifact.

Die wichtigsten Begriffe von AMS

- AMS Advanced: Die im Abschnitt „Servicebeschreibung“ der AMS Advanced-Dokumentation beschriebenen Dienste. Siehe Servicebeschreibung.
- AMS Advanced-Konten: AWS Konten, die jederzeit alle Anforderungen der AMS Advanced Onboarding Requirements erfüllen. Informationen zu den Vorteilen von AMS Advanced, Fallstudien und zur Kontaktaufnahme mit einem Vertriebsmitarbeiter finden Sie unter AWS Managed Services.
- AMS Accelerate-Konten: AWS Konten, die jederzeit alle Anforderungen der AMS Accelerate Onboarding-Anforderungen erfüllen. Siehe Erste Schritte mit AMS Accelerate.
- AWS Managed Services: AMS und/oder AMS Accelerate.
- AWS Managed Services Services-Konten: Die AMS-Konten und/oder AMS Accelerate-Konten.
- Kritische Empfehlung: Eine Empfehlung, die im AWS Rahmen einer Serviceanfrage ausgesprochen wurde und Sie darüber informiert, dass Ihre Maßnahmen zum Schutz vor potenziellen Risiken oder Störungen Ihrer Ressourcen oder der AWS-Services Wenn Sie sich entscheiden, einer kritischen Empfehlung bis zum angegebenen Datum nicht zu folgen, tragen Sie die alleinige Verantwortung für alle Schäden, die sich aus Ihrer Entscheidung ergeben.
- Vom Kunden angeforderte Konfiguration: Jede Software, Dienste oder andere Konfigurationen, die nicht identifiziert sind in:
  - Beschleunigen: Unterstützte Konfigurationen oder AMS Accelerate; Servicebeschreibung.
  - AMS Advanced: Unterstützte Konfigurationen oder AMS Advanced; Servicebeschreibung.
- Kommunikation mit einem Vorfall: AMS teilt Ihnen einen Incident mit oder Sie beantragen einen Incident bei AMS über einen Incident, der im Support Center für AMS Accelerate und in der AMS Console for AMS erstellt wurde. Die AMS Accelerate Console bietet eine Zusammenfassung der Vorfälle und Serviceanfragen auf dem Dashboard sowie Links zum Support Center für weitere Informationen.
- Verwaltete Umgebung: Die AMS Advanced-Konten und/oder die AMS Accelerate-Konten, die von AMS betrieben werden.

  Für AMS Advanced gehören dazu Multi-Account-Landingzone-Konten (MALZ) und Single-Account-Landingzone-Konten (SALZ).
- Startdatum der Abrechnung: Der nächste Werktag nach Erhalt AWS der in der AWS Managed Services Onboarding-E-Mail angeforderten Informationen. Die AWS Managed Services Onboarding-E-Mail bezieht sich auf die E-Mail, die von Ihnen gesendet wird AWS , um die

Informationen zu sammeln, die für die Aktivierung von AWS Managed Services auf Ihren Konten erforderlich sind.

Für Konten, die später von Ihnen registriert wurden, ist das Startdatum der Rechnungsstellung der nächste Tag, an dem AWS Managed Services eine Benachrichtigung über die Aktivierung von AWS Managed Services für das registrierte Konto gesendet hat. Eine Benachrichtigung zur Aktivierung von AWS Managed Services erfolgt in folgenden Fällen:

1. Sie gewähren Zugriff auf ein kompatibles AWS Konto und übergeben es an AWS Managed Services.
2. AWS Managed Services entwirft und erstellt das AWS Managed Services Services-Konto.

- Kündigung des Dienstes: Sie können die AWS Managed Services für alle AWS Managed Services Services-Konten oder für ein bestimmtes AWS Managed Services Services-Konto aus beliebigem Grund kündigen, indem Sie eine Serviceanfrage mit einer Frist von AWS mindestens 30 Tagen angeben. Am Tag der Kündigung des Service können Sie entweder:
  1. AWS übergibt Ihnen die Kontrolle über alle AWS Managed Services Services-Konten oder gegebenenfalls die angegebenen AWS Managed Services Services-Konten oder
  2. Die Parteien entfernen die AWS Identity and Access Management Rollen, die AWS Zugriff gewähren, von allen AWS Managed Services Services-Konten oder den angegebenen AWS Managed Services Services-Konten, sofern zutreffend.
- Kündigungsdatum des Dienstes: Das Kündigungsdatum des Dienstes ist der letzte Tag des Kalendermonats, der auf das Ende der 30-tägigen Kündigungsfrist folgt. Fällt das Ende der erforderlichen Kündigungsfrist nach dem 20. Tag des Kalendermonats, ist das Kündigungsdatum der letzte Tag des folgenden Kalendermonats. Im Folgenden finden Sie Beispielszenarien für Kündigungstermine.
  - Wenn die Kündigung am 12. April erfolgt, endet die Kündigungsfrist von 30 Tagen am 12. Mai. Das Kündigungsdatum des Dienstes ist der 31. Mai.
  - Wenn am 29. April eine Kündigungsfrist erteilt wird, endet die Kündigungsfrist von 30 Tagen am 29. Mai. Das Kündigungsdatum des Dienstes ist der 30. Juni.
- Bereitstellung von AWS Managed Services: AWS stellt Ihnen die AWS Managed Services zur Verfügung und Sie können ab dem Startdatum des Services für jedes AWS Managed Services-Konto auf AWS Managed Services zugreifen und diese nutzen.
- Kündigung für bestimmte AWS Managed Services-Konten: Sie können die AWS Managed Services für ein bestimmtes AWS Managed Services Services-Konto aus beliebigem Grund kündigen, indem AWS Sie dies durch eine Serviceanfrage („Anfrage zur Kündigung eines AMS-Kontos“) mitteilen.

Bedingungen für das Incident-Management:

- Ereignis: Eine Änderung in Ihrer AMS-Umgebung.
- Warnung: Immer wenn ein unterstütztes Ereignis einen Schwellenwert AWS-Service überschreitet und einen Alarm auslöst, wird eine Warnung erstellt und eine Benachrichtigung an Ihre Kontaktliste gesendet. Darüber hinaus wird ein Vorfall in Ihrer Incident-Liste erstellt.
- Vorfall: Eine ungeplante Unterbrechung oder Leistungsverschlechterung Ihrer AMS-Umgebung oder AWS Managed Services, die zu einer von AWS Managed Services oder Ihnen gemeldeten Beeinträchtigung führt.
- Problem: Eine gemeinsame Grundursache für einen oder mehrere Vorfälle.
- Lösung eines Vorfalls oder Lösung eines Vorfalls:
  - AMS hat alle nicht verfügbaren AMS-Dienste oder -Ressourcen im Zusammenhang mit diesem Vorfall wieder in den verfügbaren Zustand versetzt, oder
  - AMS hat festgestellt, dass nicht verfügbare Stacks oder Ressourcen nicht auf einen verfügbaren Zustand zurückgesetzt werden können, oder
  - AMS hat eine von Ihnen autorisierte Wiederherstellung der Infrastruktur eingeleitet.
- Reaktionszeit bei Vorfällen: Der Zeitunterschied zwischen dem Zeitpunkt, zu dem Sie einen Vorfall erstellen, und dem Zeitpunkt, zu dem AMS eine erste Antwort über die Konsole, E-Mail, das Service Center oder das Telefon bereitstellt.
- Zeit zur Behebung eines Vorfalls: Der Zeitunterschied zwischen dem Zeitpunkt, zu dem entweder AMS oder Sie einen Vorfall auslösen, und dem Zeitpunkt, zu dem der Vorfall behoben wird.
- Priorität des Vorfalls: Wie Vorfälle von AMS oder von Ihnen als „Niedrig“, „Mittel“ oder „Hoch“ priorisiert werden.
  - Niedrig: Ein unkritisches Problem mit Ihrem AMS-Service.
  - Medium: Ein AWS-Service in Ihrer verwalteten Umgebung ist verfügbar, funktioniert aber nicht wie vorgesehen (gemäß der entsprechenden Servicebeschreibung).
  - Hoch: Entweder (1) die AMS-Konsole oder ein oder mehrere AMS APIs in Ihrer verwalteten Umgebung sind nicht verfügbar; oder (2) ein oder mehrere AMS-Stacks oder Ressourcen in Ihrer verwalteten Umgebung sind nicht verfügbar und die Nichtverfügbarkeit verhindert, dass Ihre Anwendung ihre Funktion erfüllt.

  AMS kann Vorfälle gemäß den oben genannten Richtlinien neu kategorisieren.
- Wiederherstellung der Infrastruktur: Erneutes Bereitstellen vorhandener Stacks auf der Grundlage von Vorlagen der betroffenen Stacks und Initiierung einer Datenwiederherstellung auf der

Grundlage des letzten bekannten Wiederherstellungspunkts, sofern von Ihnen nicht anders angegeben, wenn eine Behebung des Vorfalls nicht möglich ist.

Bedingungen für die Infrastruktur:

- Verwaltete Produktionsumgebung: Ein Kundenkonto, in dem sich die Produktionsanwendungen des Kunden befinden.
- Verwaltete Nichtproduktionsumgebung: Ein Kundenkonto, das nur Anwendungen enthält, die nicht zur Produktion gehören, z. B. Anwendungen für Entwicklung und Tests.
- AMS-Stack: Eine Gruppe von einer oder mehreren AWS Ressourcen, die von AMS als eine Einheit verwaltet werden.
- Unveränderliche Infrastruktur: Ein für Amazon EC2 Auto Scaling Scaling-Gruppen (ASGs) typisches Infrastrukturwartungsmodell, bei dem aktualisierte Infrastrukturkomponenten (im AWS AMI) bei jeder Bereitstellung ersetzt werden, anstatt direkt aktualisiert zu werden. Die Vorteile einer unveränderlichen Infrastruktur bestehen darin, dass alle Komponenten synchron bleiben, da sie immer auf derselben Basis generiert werden. Die Unveränderlichkeit ist unabhängig von einem Tool oder Workflow zum Erstellen des AMI.
- Veränderbare Infrastruktur: Ein Infrastrukturwartungsmodell, das typisch für Stacks ist, die keine Amazon EC2 Auto Scaling Scaling-Gruppen sind und eine einzelne Instance oder nur wenige Instances enthalten. Dieses Modell entspricht am ehesten der traditionellen, hardwarebasierten Systembereitstellung, bei der ein System zu Beginn seines Lebenszyklus bereitgestellt wird und dann im Laufe der Zeit Updates auf dieses System übertragen werden. Alle Aktualisierungen des Systems werden einzeln auf die Instanzen angewendet und können aufgrund von Anwendungs- oder Systemneustarts zu Systemausfällen führen (abhängig von der Stack-Konfiguration).
- Sicherheitsgruppen: Virtuelle Firewalls für Ihre Instance zur Steuerung des ein- und ausgehenden Datenverkehrs. Sicherheitsgruppen wirken auf der Instance-Ebene und nicht auf der Subnetzebene. Daher kann jeder Instance in einem Subnetz in Ihrer VPC ein anderer Satz von Sicherheitsgruppen zugewiesen werden.
- Service Level Agreements (SLAs): Teil der AMS-Verträge mit Ihnen, die das erwartete Serviceniveau definieren.
- SLA nicht verfügbar und nicht verfügbar:
  - Eine von Ihnen eingereichte API-Anfrage, die zu einem Fehler führt.
  - Eine von Ihnen eingereichte Konsolenanfrage, die zu einer 5xx-HTTP-Antwort führt (der Server ist nicht in der Lage, die Anfrage auszuführen).

    - Alle AWS-Service Angebote, die Stacks oder Ressourcen in Ihrer von AMS verwalteten Infrastruktur bilden, befinden sich im Status „Serviceunterbrechung“, wie im Service Health Dashboard angezeigt.
    - Die Nichtverfügbarkeit, die direkt oder indirekt auf einen AMS-Ausschluss zurückzuführen ist, wird bei der Entscheidung über die Inanspruchnahme von Servicegutschriften nicht berücksichtigt. Dienste gelten als verfügbar, sofern sie nicht die Kriterien für die Nichtverfügbarkeit erfüllen.
- Servicelevel-Ziele (SLOs): Teil der AMS-Verträge mit Ihnen, in denen spezifische Serviceziele für AMS-Services definiert werden.

Bedingungen für das Patchen:

- Obligatorische Patches: Wichtige Sicherheitsupdates zur Behebung von Problemen, die den Sicherheitsstatus Ihrer Umgebung oder Ihres Kontos gefährden könnten. Ein „Kritisches Sicherheitsupdate“ ist ein Sicherheitsupdate, das vom Hersteller eines AMS-unterstützten Betriebssystems als „Kritisch“ eingestuft wird.
- Angekündigte oder veröffentlichte Patches: Patches werden in der Regel nach einem bestimmten Zeitplan angekündigt und veröffentlicht. Neue Patches werden angekündigt, wenn festgestellt wird, dass der Patch benötigt wird. In der Regel wird der Patch bald danach veröffentlicht.
- Patch-Add-on: Tag-basiertes Patchen für AMS-Instances, das die AWS Systems Manager (SSM) -Funktionalität nutzt, sodass Sie Instances taggen und diese Instances mithilfe einer Baseline und eines von Ihnen konfigurierten Fensters patchen lassen können.
- Patch-Methoden:
    - In-Place-Patching: Patchen, das durch Ändern vorhandener Instanzen erfolgt.
    - AMI-Ersatz-Patching: Patching, das durch Ändern des AMI-Referenzparameters einer vorhandenen EC2 Auto Scaling Scaling-Gruppenstartkonfiguration erfolgt.
- Patch-Anbieter (Betriebssystemanbieter, Drittanbieter): Patches werden vom Anbieter oder vom zuständigen Gremium der Anwendung bereitgestellt.
- Patch-Typen:
    - Kritisches Sicherheitsupdate (CSU): Ein Sicherheitsupdate, das vom Hersteller eines unterstützten Betriebssystems als „Kritisch“ eingestuft wurde.
    - Wichtiges Update (IU): Ein Sicherheitsupdate, das vom Hersteller eines unterstützten Betriebssystems als „wichtig“ eingestuft wurde, oder ein nicht sicherheitsrelevantes Update, das vom Hersteller eines unterstützten Betriebssystems als „Kritisch“ eingestuft wurde.

  - Anderes Update (OU): Ein Update des Herstellers eines unterstützten Betriebssystems, bei dem es sich nicht um eine CSU oder eine IU handelt.
- Unterstützte Patches: AMS unterstützt Patches auf Betriebssystemebene. Upgrades werden vom Anbieter veröffentlicht, um Sicherheitslücken oder andere Fehler zu beheben oder die Leistung zu verbessern. Eine Liste der derzeit unterstützten OSs Konfigurationen finden Sie unter Support-Konfigurationen.

Sicherheitsbedingungen:

- Detective Controls: Eine Bibliothek mit von AMS erstellten oder aktivierten Monitoren, die einen kontinuierlichen Überblick über vom Kunden verwaltete Umgebungen und Workloads für Konfigurationen bieten, die nicht den Sicherheits-, Betriebs- oder Kundenkontrollen entsprechen, und Maßnahmen ergreifen, indem sie Eigentümer benachrichtigen, Ressourcen proaktiv ändern oder beenden.

Bedingungen für Serviceanfragen:

- Serviceanfrage: Eine Anfrage von Ihnen nach einer Maßnahme, die AMS in Ihrem Namen ergreifen soll.
- Warnmeldung: Eine Benachrichtigung, die von AMS auf Ihrer Seite mit der Liste der Serviceanfragen veröffentlicht wird, wenn eine AMS-Warnung ausgelöst wird. Der für Ihr Konto konfigurierte Kontakt wird ebenfalls über die konfigurierte Methode (z. B. E-Mail) benachrichtigt. Wenn Ihre Instanzen/Ressourcen Kontakt-Tags enthalten und Ihrem Cloud Service Delivery Manager (CSDM) Ihre Zustimmung für tagbasierte Benachrichtigungen erteilt haben, werden die Kontaktinformationen (Schlüsselwert) im Tag auch bei automatisierten AMS-Benachrichtigungen benachrichtigt.
- Servicebenachricht: Eine Mitteilung von AMS, die auf Ihrer Seite mit der Liste Ihrer Serviceanfragen veröffentlicht wird.

Verschiedene Bedingungen:

- AWS Managed Services-Schnittstelle: Für AMS: Die AWS Managed Services Advanced Console, AMS CM API und Support API. Für AMS Accelerate: Die Support Konsole und die Support API.

- Kundenzufriedenheit (CSAT): AMS CSAT verfügt über umfassende Analysen, darunter Bewertungen der Fallkorrespondenz zu jedem Fall oder jeder etwaigen Korrespondenz, vierteljährliche Umfragen usw.
- DevOps: DevOps ist eine Entwicklungsmethodik, die sich nachdrücklich für Automatisierung und Überwachung aller Schritte einsetzt. DevOps zielt auf kürzere Entwicklungszyklen, eine höhere Bereitstellungshäufigkeit und zuverlässigere Releases ab, indem die traditionell getrennten Funktionen von Entwicklung und Betrieb auf einer Grundlage der Automatisierung zusammengeführt werden. Wenn Entwickler den Betrieb verwalten können und der Betrieb die Entwicklung beeinflusst, können Probleme und Probleme schneller entdeckt und gelöst werden, und Geschäftsziele lassen sich leichter erreichen.
- ITIL: Die Information Technology Infrastructure Library (ITIL genannt) ist ein ITSM-Framework, mit dem der Lebenszyklus von IT-Services standardisiert werden soll. ITIL ist in fünf Phasen unterteilt, die den IT-Servicelebenszyklus abdecken: Servicestrategie, Servicedesign, Serviceübergang, Servicebetrieb und Serviceverbesserung.
- IT-Servicemanagement (ITSM): Eine Reihe von Praktiken, die IT-Services auf die Bedürfnisse Ihres Unternehmens abstimmen.
- Managed Monitoring Services (MMS): AMS betreibt sein eigenes Überwachungssystem, den Managed Monitoring Service (MMS), das AWS Gesundheitsereignisse verarbeitet und CloudWatch Amazon-Daten sowie Daten von anderen aggregiert und AMS-Betreiber (rund um die Uhr online) über alle Alarme informiert AWS-Services, die über ein Amazon Simple Notification Service (Amazon SNS) -Thema ausgelöst wurden.
- Namespace: Wenn Sie IAM-Richtlinien erstellen oder mit Amazon Resource Names (ARNs) arbeiten, identifizieren Sie einen AWS-Service mithilfe eines Namespace. Sie verwenden Namespaces bei der Identifikation von Aktionen und Ressourcen.

Was ist mein Betriebsmodell?

Als AMS-Kunde hat sich Ihr Unternehmen dafür entschieden, Anwendungs- und Infrastrukturbetrieb zu trennen und AMS für den Infrastrukturbetrieb zu verwenden. AMS arbeitet mit Ihrem Anwendungsdesign- und Entwicklungsteam sowie Ihrem Infrastrukturdesignteam zusammen, um sicherzustellen, dass Ihr Infrastrukturbetrieb reibungslos abläuft. Die folgende Grafik veranschaulicht dieses Konzept:

AMS übernimmt die Verantwortung für Ihren AWS Infrastrukturbetrieb, während Ihre Teams für den Betrieb Ihrer Anwendungen verantwortlich sind. Als Team für Anwendungs- und Infrastrukturdesign

müssen Sie wissen, wer die Anwendung betreiben wird, sobald sie in der AMS-Infrastruktur in der Produktion eingesetzt wurde. Dieser Leitfaden behandelt gängige Ansätze für das Infrastrukturdesign in Bezug auf die Bereitstellung und Wartung von Anwendungen.

Servicemanagement in AWS Managed Services

Themen

So funktioniert der AMS-Service für Sie.

Kontoverwaltung in AWS Managed Services

In diesem Abschnitt wird die AMS-Kontoverwaltung behandelt.

Ihnen wird ein Cloud Service Delivery Manager (CSDM) zugewiesen, der Ihnen bei AMS beratend zur Seite steht und über ein detailliertes Verständnis Ihres Anwendungsfalls und Ihrer Technologiearchitektur für die verwaltete Umgebung verfügt. CSDMs arbeiten mit Kundenbetreuern, technischen Kundenbetreuern, AWS Managed Services Services-Cloudarchitekten (CAs) und AWS-Lösungsarchitekten (SAs) zusammen, um bei der Einführung neuer Projekte zu helfen und Empfehlungen zu bewährten Methoden während der gesamten Softwareentwicklungs- und Betriebsprozesse zu geben. Das CSDM ist die wichtigste Anlaufstelle für AMS. Die wichtigsten Aufgaben Ihres CSDM sind:

- Organisieren und leiten Sie monatliche Besprechungen zur Servicebewertung mit Kunden.
- Geben Sie Einzelheiten zur Sicherheit, zu Softwareupdates für die Umgebung und zu Optimierungsmöglichkeiten an.
- Machen Sie sich für Ihre Anforderungen, einschließlich Funktionsanfragen für AMS, stark.
- Beantworten und lösen Sie Anfragen zur Abrechnung und Serviceberichterstattung.
- Bieten Sie Einblicke für Empfehlungen zur Finanz- und Kapazitätsoptimierung.

Beginn des Dienstes in AWS Managed Services

Beginn des Service: Das Datum des Servicebeginns für ein AWS Managed Services Services-Konto ist der erste Tag des ersten Kalendermonats, nach dem AWS Sie darüber informiert, dass die in den Onboarding-Anforderungen für dieses AWS Managed Services Services-Konto festgelegten Aktivitäten abgeschlossen wurden. Sofern AWS eine solche Benachrichtigung nach dem 20. Tag eines Kalendermonats sendet, ist das Datum des Servicebeginns der erste Tag des zweiten Kalendermonats nach dem Datum dieser Benachrichtigung.

Beginn des Dienstes

- R steht für verantwortliche Partei, die die Arbeit zur Erfüllung der Aufgabe erledigt.
- I steht für informiert, d. h. eine Partei, die über den Fortschritt informiert wird, oft erst, wenn die Aufgabe oder das Ergebnis abgeschlossen ist.

Beginn der Dienstleistung

Schritt #	Titel des Schritts	Beschreibung	Custome	AMS
1.	Übergabe des AWS-Kontos an Kunden	Der Kunde erstellt ein neues AWS-Konto und übergibt es an AWS Managed Services	R	I
2.	AWS Managed Services Services-Konto — Design	Finalisieren Sie den Entwurf des AWS Managed Services Services-Kontos	I	R
3.	AWS Managed Services Services-Konto — erstellen	Ein AWS Managed Services Services-Konto wird gemäß dem Design in Schritt 2 erstellt	I	R

Kundenbeziehungsmanagement (CRM)

AWS Managed Services (AMS) bietet einen CRM-Prozess (Customer Relationship Management), um sicherzustellen, dass eine klar definierte Beziehung zu Ihnen aufgebaut und aufrechterhalten wird. Die Grundlage dieser Beziehung bildet das Wissen von AMS über Ihre Geschäftsanforderungen. Der CRM-Prozess ermöglicht ein genaues und umfassendes Verständnis von:

- Ihre Geschäftsanforderungen und wie Sie diese Bedürfnisse erfüllen können
- Ihre Fähigkeiten und Einschränkungen
- AMS und Ihre unterschiedlichen Verantwortlichkeiten und Pflichten

Der CRM-Prozess ermöglicht es AMS, einheitliche Methoden zur Erbringung von Dienstleistungen für Sie und zur Steuerung Ihrer Beziehung zu AMS zu verwenden. Der CRM-Prozess umfasst:

- Identifizierung Ihrer wichtigsten Stakeholder
- Aufbau eines Führungsteams
- Durchführung und Dokumentation von Besprechungen zur Leistungsbeurteilung mit Ihnen
- Bereitstellung eines formellen Servicebeschwerdeverfahrens mit einem Eskalationsverfahren
- Implementierung und Überwachung Ihres Zufriedenheits- und Feedback-Prozesses
- Verwaltung Ihres Vertrags

CRM-Prozess

Der CRM-Prozess umfasst die folgenden Aktivitäten:

- Identifizieren und Verstehen Ihrer Geschäftsprozesse und Bedürfnisse. Ihre Vereinbarung mit AMS identifiziert Ihre Stakeholder.
- Definition der Dienstleistungen, die gemäß Ihren Bedürfnissen und Anforderungen erbracht werden sollen.
- Treffen mit Ihnen im Rahmen der Serviceüberprüfung, um alle Änderungen des AMS-Serviceumfangs, der SLA, des Vertrags und Ihrer Geschäftsanforderungen zu besprechen. Es können vorläufige Treffen mit Ihnen abgehalten werden, um Leistungen, Erfolge, Probleme und Aktionspläne zu besprechen.
- Überwachen Sie Ihre Zufriedenheit mithilfe unserer Kundenzufriedenheitsumfrage und des Feedbacks, das Sie bei Besprechungen erhalten.

- Berichterstattung über die Leistung in monatlichen, intern gemessenen Leistungsberichten.
- Wir besprechen den Service gemeinsam mit Ihnen, um Verbesserungsmöglichkeiten zu ermitteln. Dazu gehört auch die regelmäßige Kommunikation mit Ihnen über das Niveau und die Qualität des erbrachten AMS-Dienstes.

CRM-Besprechungen

Die AMS Cloud Service Delivery Manager (CSDMs) führen regelmäßig Treffen mit Ihnen durch, um die einzelnen Servicebereiche (Betrieb, Sicherheit und Produktinnovationen) und die Führungsqualitäten (SLA-Berichte, Zufriedenheitsmaßnahmen und Änderungen Ihrer Geschäftsanforderungen) zu besprechen.

Meeting	Zweck	Mode	Teilnehmer
Wöchentliche Statusüberprüfung (optional)	Ausstehende Probleme oder Vorfälle, Patches, Sicherheitsereignisse, Problemaufzeichnungen Betriebstrend über 12 Wochen (+/- 6) Bedenken des Anwendungsbetreibers Zeitplan für das Wochenende	Kunde vor Ort location/ Telecom/Chime	AMS: CSDM und Cloud-Architekt (CA) Vom Kunden zugewiesene Teammitglieder (z. B.: Cloud-/ Infrastruktur-, Anwendungssupport-, Architekturteams usw.)
Monatlicher Geschäftsbericht	Überprüfen Sie die Leistung auf Serviceniveau (Berichte, Analysen und Trends) Finanzielle Analyse Produkt-Roadmap BESETZUNG	Kunde vor Ort location/ Telecom/Chime	AMS: CSDM, Cloud-Architekt (CA), AMS-Kundenbetreuer, Technischer Produktmanager (TPM) von AMS (optional), AMS

Meeting	Zweck	Mode	Teilnehmer
			OPS-Manager (optional) Sie: Vertreter des Anwendungsbetreibers
Vierteljährlicher Geschäftsbericht	Leistung und Trends der Scorecard und des Service Level Agreements (SLA) (6 Monate) Künftige Pläne/Migrationen für die kommenden 3/6/9/12 Monate Risiko und Risikominderung Wichtige Verbesserungsinitiativen Artikel der Produkt-Roadmap Auf die zukünftige Ausrichtung ausgerichtete Möglichkeiten Finanzdaten Initiativen zur Kosteneinsparung Geschäftsoptimierung	Standort des Kunden vor Ort	AMS: CSDM, Cloud-Architekt, AMS-Kundenbetreuer, AMS-Serviceleiter, AMS-Betriebsleiter Sie: Vertreter des Anwendungsbetreibers, Servicemitarbeiter, Serviceleiter

Vereinbarungen für CRM-Treffen

Das AMS CSDM ist für die Dokumentation des Treffens verantwortlich, einschließlich:

- Erstellung der Tagesordnung, einschließlich Aktionspunkten, Themen und Teilnehmerlisten.
- Erstellung der Liste der Aktionspunkte, die bei jeder Sitzung überprüft werden, um sicherzustellen, dass die Punkte termingerecht abgeschlossen und gelöst werden.

- Verteilung des Sitzungsprotokolls und der Liste der Aktionspunkte per E-Mail an die Besprechungsteilnehmer innerhalb eines Werktages nach der Besprechung.
- Speichern von Besprechungsprotokollen im entsprechenden Dokumentenspeicher.

In Ermangelung des CSDM erstellt und verteilt der AMS-Vertreter, der die Sitzung leitet, das Protokoll.

> **Note**
>
> Ihr CSDM arbeitet mit Ihnen zusammen, um Ihre Kontoverwaltung festzulegen.

Monatliche CRM-Berichte

Ihr AMS CSDM bereitet monatliche Präsentationen zur Serviceleistung vor und versendet sie. Die Präsentationen enthalten Informationen zu folgenden Themen:

- Datum des Berichts
- Zusammenfassung und Einblicke:
  - Wichtige Hinweise: Gesamtzahl und Anzahl der aktiven Stacks, Status der Stack-Patches, Onboarding-Status des Accounts (nur beim Onboarding), Zusammenfassungen der kundenspezifischen Probleme
  - Leistung: Statistiken zur Behebung von Vorfällen, zu Warnmeldungen, Patches, Änderungsanfragen (RFCs), Serviceanfragen sowie zur Verfügbarkeit von Konsole und API
  - Probleme, Herausforderungen, Bedenken und Risiken: Status der kundenspezifischen Probleme
  - Künftige Themen: Kundenspezifische Onboarding- oder Problemlösungspläne
- Verwaltete Ressourcen: Grafiken und Kreisdiagramme von Stapeln
- AMS-Metriken: Überwachungs- und Ereignismetriken, Vorfallkennzahlen, AMS-SLA-Einhaltungskennzahlen, Serviceanforderungskennzahlen, Change-Management-Metriken, Speichermetriken, Kontinuitätskennzahlen, Trusted Advisor Advisor-Metriken und Kostenübersichten (auf verschiedene Arten dargestellt). Anfragen zu Funktionen. Kontaktinformationen.

> **Note**
>
> Zusätzlich zu den beschriebenen Informationen informiert Sie Ihr CSDM auch über jede wesentliche Änderung des Geltungsbereichs oder der Bedingungen, einschließlich der Inanspruchnahme von Subunternehmern durch AMS für betriebliche Aktivitäten.
> AMS generiert Berichte über Patches und Backups, die Ihr CSDM in Ihren monatlichen Bericht aufnimmt. Als Teil des Systems zur Berichtsgenerierung fügt AMS Ihrem Konto einige Infrastrukturen hinzu, auf die Sie nicht zugreifen können:
>
> - Ein S3-Bucket mit den gemeldeten Rohdaten
> - Eine Athena-Instanz mit Abfragedefinitionen zum Abfragen der Daten
> - Ein Glue Crawler zum Lesen der Rohdaten aus dem S3-Bucket

Kostenoptimierung in AWS Managed Services

AWS Managed Services stellt Ihnen im Rahmen Ihrer monatlichen Geschäftsberichte jeden Monat detaillierte Berichte zur Kostenauslastung und zu Einsparungen bereit (MBRs).

AMS folgt einer Reihe von Standardprozessen und -mechanismen, um Möglichkeiten zur Kosteneinsparung in Ihren verwalteten Konten zu ermitteln und Sie bei der Planung und Einführung der Änderungen zur Optimierung Ihrer AWS-Ausgaben zu unterstützen.

> **Note**
>
> AMS entwickelt derzeit ein Video, das Ihnen bei der Kostenoptimierung helfen soll. Der erste Schritt besteht darin, Ihnen ein PDF und eine Excel-Tabelle mit bewährten Methoden zur Kostenoptimierung zur Verfügung zu stellen. Um auf diese Ressourcen zuzugreifen, öffnen Sie die ZIP-Datei mit der Kurzanleitung zur Kostenoptimierung.

Framework zur Kostenoptimierung

AMS verfolgt gemeinsam mit Ihnen einen dreistufigen Ansatz, um Ihre AWS-Kosten zu optimieren:

1. Identifizieren Sie Möglichkeiten zur Kostenoptimierung in Ihrer verwalteten Umgebung
2. Stellen Sie Ihnen einen Plan zur Kostenoptimierung vor
3. Unterstützen Sie uns dabei, die Kostenoptimierung auf messbare Weise zu erreichen

Identifizieren Sie Möglichkeiten zur Kostenoptimierung in der verwalteten Umgebung

AMS verwendet AWS native Tools wie Cost Explorer und Trusted Advisor und nutzt gleichzeitig über 20 Kosteneinsparungsmuster in den Bereichen Architekturoptimierung, EC2 instanzielle und AWS kundenorientierte Optimierung, um maßgeschneiderte Kosteneinsparungsempfehlungen für Sie zu erstellen.

Einige der Optimierungsempfehlungen beinhalten die folgenden.

Empfehlungen zur architektonischen Optimierung:

- Optimale Nutzung der S3-Speicherklasse: Amazon S3 bietet eine Reihe von Speicherklassen, um verschiedene Workload-Anforderungen in Bezug auf Datenzugriff, Ausfallsicherheit und Kosten zu erfüllen. S3 Intelligent-Tiering und S3-Speicherklassenanalysen auf der Grundlage der Workload-Anforderungen ermöglichen es Ihnen, die S3-Kosten effizient zu verwalten.
- Verwendung von Caching-Architekturen: Durch die Nutzung von Cache-Instanzen können Sie gegebenenfalls einige Datenbankinstanzen ersetzen und gleichzeitig Ihre IOPS-Anforderungen erfüllen.
- Einsparungen beim EBS-Upgrade: Die Migration Ihrer EBS-Volumes von gp2 auf gp3 bietet Kosteneinsparungen von bis zu 20% und Sie können unabhängig von der Volume-Größe von einer vorhersehbaren Basisleistung von 3.000 IOPS und 125 MiB/s profitieren.
- Einsatz von Elastizität: Die Funktionen zur auto-scaling AWS ermöglichen eine effektive Ressourcennutzung und Möglichkeiten zur Kostenoptimierung. Durch die regelmäßige Überprüfung und Aktualisierung der Richtlinien zur Instanzskalierung je nach Bedarf können weitere Kosten eingespart werden.

EC2 instanzorientierte Empfehlungen

- Richtige Dimensionierung von Instanzen: Die Empfehlungen konzentrierten sich auf die Dimensionierung der Instanzen und die optimale Konfiguration auf der Grundlage der Nutzung. Zu den Empfehlungen gehören auch die Nutzung der Amazon EC2 Auto Scaling Scaling-Funktion und das Ersetzen von EC2 Instances, sofern zutreffend, durch statische Webinhalte auf Amazon S3 usw. AWS Lambda
- Instance-Planung: Die Verwendung von AMS Resource Scheduler zum automatischen Starten und Stoppen von Instances auf der Grundlage eines Zeitplans trägt dazu bei, die Kosten einzudämmen, insbesondere für Instances, die nicht zur Produktion gehören und außerhalb der Geschäftszeiten nicht genutzt werden.

- Sparpläne abonnieren: Der Sparplan ist die einfachste Möglichkeit, bei der Nutzung zu sparen. AWS Die EC2 Instance-Sparpläne bieten bis zu 72% Einsparungen bei der Nutzung Ihrer Amazon EC2 Instances im Vergleich zu On-Demand-Preisen. Die Amazon SageMaker AI-Sparpläne bieten bis zu 64% Ersparnis bei der Nutzung Ihrer Amazon SageMaker AI-Services. AMS gibt Ihnen auf der Grundlage Ihrer AWS Ressourcennutzung entsprechende Empfehlungen zu Sparplänen.
- Hinweise zur Nutzung und Nutzung von Reserved Instances (RI): Amazon EC2 Reserved Instances (RI) bieten einen erheblichen discount (bis zu 75%) im Vergleich zu On-Demand-Preisen und ermöglichen eine Kapazitätsreservierung, wenn sie in einer bestimmten Availability Zone verwendet werden.
- Nutzung von Spot-Instances: Fehlertolerante Workloads können Spot-Instances nutzen und die Preise um bis zu 90% senken.
- Kündigung von Instances im Leerlauf: Identifizierung und Meldung von Instances, die sich im Leerlauf befinden oder nur eine geringe Auslastung aufweisen und beendet werden können.

Kundenorientierte Empfehlungen

- Kontobereinigung: Auf Kontoebene identifiziert AMS auch nicht genutzte EBS-Volumes, doppelte CloudTrail Pfade, leere Konten mit ungenutzten Ressourcen usw. und gibt Empfehlungen für die Bereinigung.
- SLA-Empfehlungen: Darüber hinaus überprüft AMS Ihre Plus- und Premium-Konten regelmäßig und empfiehlt, die richtige SLA-Stufe für die Konten zu wählen.
- Optimierung der AMS-Automatisierung: AMS optimiert kontinuierlich die AMS-Automatisierung und die Infrastruktur, die für die Bereitstellung von AMS-Diensten verwendet wird.

Präsentieren Sie Kunden und unterstützen Sie sie bei der Planung

AMS führt monatliche Geschäftsüberprüfungen (MBRs) mit den wichtigsten Kundenakteuren durch und stellt die identifizierten Möglichkeiten, Mechanismen und Empfehlungen zur Kosteneinsparung sowie mögliche Kosteneinsparungen vor. Wir arbeiten weiter mit Ihnen zusammen, um die erforderlichen Änderungen zu planen.

Unterstützung bei der Umsetzung der Empfehlungen und Messung der Kostenauswirkungen

AMS hilft bei der Erzielung und Messung von Kostenauswirkungen und Optimierungsänderungen.

Sie bewerten die Auswirkungen, Risiken und Erfolgskriterien der empfohlenen Änderungen auf die Anwendung und stellen über die AMS-Konsole die entsprechenden Änderungsanträge (RFCs). AMS arbeitet mit Ihnen zusammen und implementiert die Änderungen im Zusammenhang mit der Kostenoptimierung in Ihren verwalteten Konten. AMS misst die Auswirkungen auf die Kosten und bezieht die erzielten Einsparungen in die monatlichen Geschäftsberichte ein (MBRs).

Verantwortungsmatrix für die Kostenoptimierung

Verantwortlichkeiten bei der AMS-Kostenoptimierung.

Kostenoptimierung RACI

Aktivität	Customer	AMS
Zusammentellung von Empfehlungen zur Kosteneinsparung und Erstellung des Berichts	I	R
Vorlage des Kosteneinsparungsberichts	C	R
Planungsänderungen im Zusammenang mit Kostenein	R	C

Aktivität	Customer	AMS
sparungen		
Bewertung der Auswirkungen und des Risikos der Änderung	R	C
Erhebung RFCs zur Umsetzung der Änderungen	R	C
Überprüfung RFCs und Umsetzung der Änderungen	C	R
Testen der Anwendung und Validierung der Änderungsimplementierung	R	C

Aktivität	Customer	AMS
Messung der Kostenauswirkungen nach der Änderung und Präsentation beim Kunden	I	R

Servicezeiten in AWS Managed Services

Funktion	AMS für Fortgeschrittene
	Premium-Stufe
Serviceanfrage	24/7
Verwaltung von Zwischenfällen (P2-P3)	Rund um die
Sicherung und Wiederherstellung	24/7
Patch-Management	24/7
Überwachen und Warnen	24/7
Automatisierte Änderungsanforderung (RFC)	24/7
Nicht automatisierter Änderungsantrag (RFC)	24/7
Manager für die Bereitstellung von Cloud-Diensten (CSDM)	Montag bis Freitag: 08:00 — 17:00 Uhr, lokale Geschäftszeiten

Hilfe zu AWS Managed Services erhalten

AMS unterstützt Sie mit Incident Management, Service Request Management und Change Management 24 Stunden am Tag, 7 Tage die Woche, 365 Tage im Jahr (gemäß dem für das Konto geltenden AMS Service Level Agreement).

Um ein Leistungsproblem mit dem AWS- oder AMS-Service zu melden, das sich auf Ihre verwaltete Umgebung auswirkt, verwenden Sie die AMS-Konsole und reichen Sie einen Vorfallbericht ein. Einzelheiten finden Sie unter Einen Vorfall melden. Allgemeine Informationen zum AMS Incident Management finden Sie unter Reaktion auf Vorfälle.

Verwenden Sie die AMS-Konsole und reichen Sie eine Serviceanfrage ein, um Informationen oder Ratschläge zu erhalten oder zusätzliche Services von AMS in Anspruch zu nehmen. Einzelheiten finden Sie unter Serviceanfrage erstellen. Allgemeine Informationen zu AMS-Serviceanfragen finden Sie unter Verwaltung von Serviceanfragen.

Anwendungsentwicklung

Prozesse und Praktiken für die Anwendungsentwicklung, die ein effektives Design und die Bereitstellung von Anwendungen in einer AWS Managed Services (AMS) -Umgebung ermöglichen. AMS führt Sie durch den folgenden allgemeinen Prozess:

1. Stellen Sie sich eine Anwendung vor, die entwickelt oder in Ihre AMS-verwaltete Umgebung integriert werden soll, und entwerfen Sie sie. Einige Überlegungen:
    a. Wie werden Sie Ihre Anwendung bereitstellen? Mit Automatisierung mithilfe eines Bereitstellungstools wie Ansible oder manuell durch direktes Hochladen der benötigten Dateien?
    b. Wie werden Sie Ihre Anwendung aktualisieren? Mit einem veränderbaren Ansatz, bei dem jede Instanz separat aktualisiert wird, oder mit einem unveränderlichen Ansatz, bei dem jede Instanz mit einem einzigen, aktualisierten AMI in einer Auto Scaling Scaling-Gruppe aktualisiert wird?
2. Planen und gestalten Sie die Infrastruktur, die zum Hosten der Anwendung verwendet wird, mithilfe von AWS Architekturbibliotheken, AWS „Well-Architected“ -Anleitungen sowie AMS und anderen Experten für Cloud-Architektur. Die folgenden Abschnitte dieses Leitfadens enthalten Informationen, die Ihnen dabei helfen können.
3. Wählen Sie einen Ansatz zur Bereitstellung der Infrastruktur aus:
    a. Full Stack: Alle Infrastrukturkomponenten werden gleichzeitig und zusammen bereitgestellt.
    b. Stufe und Stufe: Infrastrukturbereitstellungen werden separat bereitgestellt und anschließend mit Änderungen an Sicherheitsgruppen verknüpft. Diese Art der Bereitstellung wird auch durch eine serielle Konfiguration von Stack-Komponenten erreicht, die aufeinander aufbauen. Beispielsweise wird der Load Balancer angegeben, den Sie zuvor erstellt haben, als Sie eine Auto Scaling Scaling-Gruppe erstellt haben.
    c. Welche Umgebungen, wie Dev, Staging und Prod, werden Sie einsetzen?
4. Wählen Sie AMS-Änderungstypen (CTs), die die erforderlichen Stacks oder Stufen bereitstellen, und bereiten Sie die erforderlichen Änderungsanforderungen vor (). RFCs
5. Senden Sie das RFCs , um die Bereitstellung der Infrastruktur in der entsprechenden Umgebung auszulösen.
6. Stellen Sie die Anwendung mithilfe des ausgewählten Ansatzes zur Anwendungsbereitstellung bereit.

7. Überarbeiten Sie die Infrastruktur und die Anwendungen nach Bedarf.
8. Stellen Sie Infrastruktur und Anwendungen in geeigneten Folgeumgebungen bereit, vorausgesetzt, Ihre erste Bereitstellung erfolgt in einer Nicht-Produktionsumgebung.
9. Die laufende Wartung erfolgt durch AMS, das die zugrunde liegende Infrastruktur betreibt, und Ihre Betriebsteams betreiben die Anwendungsinfrastrukturen.
10. Um eine Anwendung außer Betrieb zu nehmen, beenden Sie die entsprechende AMS-Infrastruktur.

Gut strukturiert sein

AWS Wir bei sind der Ansicht, dass gut konzipierte Systeme die Wahrscheinlichkeit eines Geschäftserfolgs erheblich erhöhen. Das AWS Architecture Center bietet fachkundige Beratung zur Architektur in der. AWS Cloud

Wir empfehlen die folgenden Artikel und Whitepapers, um Ihnen zu helfen, die Vor- und Nachteile der Entscheidungen zu verstehen, die Sie beim Aufbau von Systemen treffen müssen. AWS

Sind Sie Well-Architected? : Stellt das AWS Well-Architected Framework vor, das auf sechs Säulen basiert:

- Operative Exzellenz: Der Schwerpunkt Operational Excellence konzentriert sich auf den Betrieb und die Überwachung von Systemen, um einen Mehrwert für das Unternehmen zu erzielen, sowie auf die kontinuierliche Verbesserung von Prozessen und Verfahren. Zu den wichtigsten Themen gehören die Verwaltung und Automatisierung von Änderungen, die Reaktion auf Ereignisse und die Definition von Standards für eine erfolgreiche Verwaltung des täglichen Betriebs.
- Sicherheit: Der Schwerpunkt der Sicherheitssäule liegt auf dem Schutz von Informationen und Systemen. Zu den wichtigsten Themen gehören die Vertraulichkeit und Integrität von Daten, die Identifizierung und Verwaltung, wer was mit der Rechteverwaltung tun kann, der Schutz von Systemen und die Einrichtung von Kontrollen zur Erkennung von Sicherheitsereignissen.
- Zuverlässigkeit: Der Schwerpunkt der Zuverlässigkeit liegt auf der Fähigkeit, Ausfälle zu verhindern und diese schnell zu beheben, um den Geschäfts- und Kundenanforderungen gerecht zu werden. Zu den wichtigsten Themen gehören grundlegende Elemente rund um die Einrichtung, projektübergreifende Anforderungen, Wiederherstellungsplanung und unser Umgang mit Veränderungen.
- Leistungseffizienz: Der Schwerpunkt der Leistungseffizienz liegt auf der effizienten Nutzung von IT- und Computerressourcen. Zu den wichtigsten Themen gehören die Auswahl der richtigen

Ressourcentypen und -größen je nach Workload-Anforderungen, die Überwachung der Leistung und das Treffen fundierter Entscheidungen, um die Effizienz aufrechtzuerhalten, wenn sich die Geschäftsanforderungen weiterentwickeln.

- Kostenoptimierung: Der Schwerpunkt der Kostenoptimierung liegt auf der Vermeidung unnötiger Kosten. Zu den wichtigsten Themen gehören das Verständnis und die Kontrolle darüber, wofür Geld ausgegeben wird, die Auswahl der geeignetsten und richtigen Anzahl von Ressourcentypen, die Analyse der Ausgaben im Zeitverlauf und die Skalierung, um Geschäftsanforderungen zu erfüllen, ohne dabei zu hohe Ausgaben zu tätigen.
- Nachhaltigkeit: Die Nachhaltigkeit konzentriert sich auf die Fähigkeit, die Auswirkungen auf die Nachhaltigkeit kontinuierlich zu verbessern, indem der Energieverbrauch gesenkt und die Effizienz aller Komponenten einer Arbeitslast erhöht wird, indem der Nutzen der bereitgestellten Ressourcen maximiert und der Gesamtbedarf an Ressourcen minimiert wird.

AWS Well-Architected Framework: Beschreibt, AWS wie Kunden ihre Cloud-basierten Architekturen bewerten und verbessern und die geschäftlichen Auswirkungen ihrer Designentscheidungen besser verstehen können. Es befasst sich mit allgemeinen Entwurfsprinzipien sowie spezifischen Best Practices und Leitlinien in sechs konzeptionellen Bereichen, die als die Säulen des Well-Architected Framework AWS definiert werden.

Verantwortlichkeiten auf Anwendungsebene im Vergleich zu Verantwortlichkeiten auf Infrastrukturebene in AMS

Durch den Einsatz von AMS werden Ihre Infrastruktur und alles, was sie für Wartung und Wachstum benötigt, von AMS verwaltet. Was auch immer Sie für line-of-business Anwendungen oder Produktanwendungen benötigen, wird jedoch von Ihnen entwickelt, bereitgestellt und gewartet.

Mithilfe von Tools zur Anwendungsbereitstellung wie CodeDeploy AND oder Chef AWS CloudFormation, Puppet, Ansible oder Saltstack kann Ihre Anwendungsbereitstellung in Ihrer von AMS verwalteten Infrastruktur vollständig automatisiert werden.

Einzelheiten darüber, was AMS tut und was nicht, finden Sie unter. Was wir tun, was wir nicht tun

Veränderbarkeit von EC2 Amazon-Instances in AMS

Sie und AMS können die Amazon Elastic Compute Cloud (Amazon EC2) -Instances in Ihrer Infrastruktur auf zwei Arten verwalten:

- Unveränderlich: Dieses Modell verwendet Amazon Machine Images (AMIs), die mit den erforderlichen Funktionen gebacken (erstellt) wurden. Bei der Bereitstellung eines Updates werden die vorhandenen Instances heruntergefahren und vollständig durch neue ersetzt, die aus einem aktualisierten AMI erstellt wurden. Um Ausfallzeiten zu minimieren, bleiben bei diesem fortlaufenden Prozess einige Instanzen nicht aktualisiert und zugänglich, während andere aktualisiert werden, bis die neue Änderung schließlich vollständig implementiert ist.
- Veränderbar: In diesem Modell wird die Infrastruktur aktualisiert, indem neuer Code auf bestehenden Systemen in der Cloud bereitgestellt wird. Bei diesem Modell handelt es sich um eine Mischung aus manuellem Upload von Updates und Verwendung infrastructure-as-code zur Bereitstellung von Updates. Es ist nicht auf neue AMIs Updates angewiesen.

Diese Wartungsmodelle werden in späteren Abschnitten dieses Handbuchs ausführlicher behandelt.

AWS Secrets Manager mit AMS-Ressourcen verwenden

Es gibt viele Fälle, in denen Sie möglicherweise Geheimnisse mit AMS teilen müssen, zum Beispiel:

- Zurücksetzen des Master-Passworts für die RDS-Instanz
- Zertifikate für Load Balancer
- Beschaffung langlebiger Anmeldeinformationen für IAM-Benutzer von AMS

Der sicherste Weg, vertrauliche Informationen mit AMS zu teilen, ist der AWS Secrets Manager. Gehen Sie wie folgt vor:

1. Melden Sie sich mit Ihrem Verbundzugriff und der CustomerReadOnly Rolle für Single-Account-Landingzone (SALZ) bei der AWS Konsole an. Verwenden Sie eine dieser Rollen, AWSManaged ServicesSecurityOpsRole AWSManagedServicesAdminRole, und AWSManaged ServicesChangeManagementRole für Multi-Account-Landingzone (MALZ).
2. Navigieren Sie zur AWS Secrets Manager-Konsole und klicken Sie auf Neues Geheimnis speichern.
3. Wählen Sie „Andere Arten von Geheimnissen“ aus.
4. Geben Sie den geheimen Wert als Klartext ein und klicken Sie auf Weiter.
5. Geben Sie den geheimen Namen und die Beschreibung ein. Der Name sollte immer mit "customer-shared/*" beginnen. Zum Beispiel "customer-shared/license-2018“. Wenn Sie fertig sind, fahren Sie fort, indem Sie auf Weiter klicken.

6. Verwenden Sie die standardmäßige KMS-Verschlüsselung.
7. Lassen Sie die automatische Rotation deaktiviert und klicken Sie auf Weiter.
8. Überprüfen Sie es und klicken Sie auf Speichern, um das Geheimnis zu speichern.
9. Antworten Sie uns in einer AMS-Serviceanfrage mit dem geheimen Namen und der ARN, damit wir das Geheimnis identifizieren und abrufen können. Informationen zum Erstellen von Serviceanfragen finden Sie unter Beispiele für Serviceanfragen.

Anwendungsbereitstellung in AMS

Während des Onboardings ermittelt AWS Managed Services (AMS) gemeinsam mit Ihnen die Infrastruktur, die Sie benötigen.

Die grundlegende Infrastruktur umfasst eine AWS Virtual Private Cloud (VPC), Kommunikationssicherheit über einen ADFS-Forest-Trust, die grundlegenden Subnetze (DMZ, Shared Services und Private), die über zwei Availability Zones gespiegelt und mit einem verwalteten NAT, Bastionen, öffentlichen Load Balancern (DX) und der erforderlichen Sicherheit konfiguriert sind. AWS Direct Connect Ihre Anwendungsressourcen werden in Ihrem privaten Subnetz oder Subnetz für Kundenanwendungen bereitgestellt. Weitere Informationen zu einer typischen AMS-Architektur finden Sie im AWS Managed Services Services-Benutzerhandbuch.

Die Infrastruktur, die Sie bereitstellen, sobald die Grundlagen fertig sind, sollte alle Komponenten für Ihre Anwendungen und die Anwendungsentwicklung enthalten.

Funktionen zur Anwendungsbereitstellung in AMS

Einige der Möglichkeiten, Anwendungen in AMS bereitzustellen. Einzelheiten zu den einzelnen Methoden folgen.

Beispiele für Funktionen zur Anwendungsbereitstellung

Methodenname	Bereitstellung der Infrastruktur	AMI oder Schlüssel element (e)	Anwendung installie ren
Veränderbare Anwendungen, AMS AMI			
Manuelle Anwendung sbereitstellung	Vollständiger Stack, CT oder Tier und Tier CTs	Von AMS bereitges telltes AMI	Senden Sie das Access Managemen t CT und installieren Sie die Anwendung manuell.
UserData Anwendung sbereitstellung mit einem Anwendung			Verwenden Sie Provisioning CT mit UserData Scripting, das einen Anwendung

Methodenname	Bereitstellung der Infrastruktur	AMI oder Schlüssel element (e)	Anwendung installie ren
sagenten (d. h. Chef, Puppet usw.)			sagenten installie rt und der die Anwendung script/ag ent installiert.
UserData Anwendung sbereitstellung ohne Agenten (d. h. Ansible, Salt SSH usw.)			Senden Sie das Access Managemen t CT und installieren Sie den Anwendung sagenten. Stellen Sie die Anwendung mit Tools zur Anwendung sbereitstellung bereit.
Veränderbare Anwendungen, benutzerdefiniertes AMI			
Bereitstellung benutzerdefinierter AMI-Anwendungen (ohne ASG)	Vollständiger Stack-CT oder Tier-and-Tie CTs	Benutzerdefinierte s AMI. AMS AMI -> mit dem Tooling-Agenten für die Anwendungsbereitst ellung anpassen -> EC2 Instanz erstellen (CT) -> AMI erstellen (CT).	Tools zur Anwendung sbereitstellung (d. h. Chef), Nutzung von Agenten, Bereitste llung der Anwendung.
Bereitstellung von AWS Database Migration Service (DMS) -Anwendungen	AWS DMS-Synch ronisierung mit vorhandenem AMS Relational Database-Stack.	Benutzerdefiniertes AMI	Kunde oder Partner verwendet AWS Database Migration Service; AMS verifizie rt AMS-Komponenten beim Start

Methodenname	Bereitstellung der Infrastruktur	AMI oder Schlüssel element (e)	Anwendung installie ren
Bereitstellung der Workload Ingest-An wendung	Workload Ingest CT, von einem Partner migriert instance/ AMI und vom Kunden initiiert.		Der Partner migriert die Instanz und erstellt AMI in der vom Kunden verwaltet en VPC. Der Kunde verwendet Workload Ingest CT, um den Stack in AMS zu starten. Details hierzu finden Sie unter AMS-Arbei tslastaufnahme (WIGS).
Unveränderliche Anwendungen			
Bereitstellung benutzerdefinierter AMI-Anwendungen (ASG)	Vollständiger Stack-CT oder Tier-and-Tie CTs	AMS AMI -> anpassen -> EC2 Instanz erstellen (CT) -> AMI erstellen (CT) -> Auto Scaling Scaling-Gruppe erstellen.	Auto Scaling stellt die Anwendung mit dem benutzerdefinierten AMI bereit Details hierzu finden Sie unter Tier-and-Tie-App-Bereitstel lungen in AMS.
Veränderliche oder unveränderliche Anwendungen			

Methodenname	Bereitstellung der Infrastruktur	AMI oder Schlüsselelement (e)	Anwendung installieren
Bereitstellung benutzerdefinierter CloudFormation Vorlagenanwendungen	CloudFormation Vorlage	CloudFormation AWS-Vorlage -> customize/prepare für AMS -> Bereitstellung | Aufnahme | Stapel aus CloudFormation Vorlage | Erstellen (ct-36cn2avfrrj9v).	AMS stellt Ihre Anwendung mithilfe Ihrer benutzerdefinierten Vorlage in Ihrem Konto bereit und validiert die Anwendungsbereitstellung. CloudFormation Details hierzu finden Sie unter CloudFormation AMS-Aufnahme.
SQL-Datenbank-Import	AMS-Operationen (Andere | Andere CT)	Lokale SQL-Datenbank -> .bak-Datei -> AMS RDS-SQL-Datenbank -> Verwaltung | Andere | Andere | Erstellen Sie (ct-1e1xtak34nx76) für den Import.	AMS importiert Ihre lokale Datenbank in Ihre von AMS verwaltete RDS-Datenbank. Details hierzu finden Sie unter Datenbankimport (DB) nach AMS RDS für Microsoft SQL Server.
Database Migration Service (DMS)	AMS-Operationen (mehrere CTs)	Lokale Datenbank -> DMS-Replikationsinstanz -> DMS-Replikationssubnetzgruppe -> DMS-Zielendpunkt -> DMS-Quellendpunkt -> DMS-Replikationsaufgabe.	AMS importiert Ihre lokale Datenbank in Ihre von AMS verwaltete S3- oder RDS-Zieldatenbank. Details hierzu finden Sie unter AWS Database Migration Service (AWS DMS).

Methodenname	Bereitstellung der Infrastruktur	AMI oder Schlüsselelement (e)	Anwendung installieren
CodeDeploy Bereitstellung von Anwendungen	CodeDeploy	Anwendung -> CodeDeploy Anwendung -> CodeDeploy Bereitstellungsgruppe -> CodeDeploy Bereitstellung.	Je nach Nutzung, Direktbereitstellung oder Blue/Green Anwendungsbereitstellung. Details hierzu finden Sie unter CodeDeploy Anfragen.

Planung Ihrer Anwendungsbereitstellung in AMS

Eine Reihe von empfohlenen Fragen, die beantwortet werden müssen, um Anwendungsbereitstellungen zu ermöglichen, finden Sie unterAnhang: Fragebogen zum Onboarding von Bewerbungen. Zu den Fragen gehören die Beschreibung Ihrer:

- Zusammenfassung der Bereitstellung
- Komponenten für die Infrastrukturbereitstellung
- Plattform für das Hosten von Anwendungen
- Modell zur Anwendungsbereitstellung
- Abhängigkeiten von Anwendungen
- SSL-Zertifikate für Produktanwendungen

AMS-Arbeitslastaufnahme (WIGS)

Themen

- Migration von Workloads: Voraussetzungen für Linux und Windows
- Wie Migration Ihre Ressourcen verändert
- Migration von Workloads: Standardprozess
- Migration von Workloads: CloudEndure landing zone (SALZ)
- AMS Tools-Konto (Workloads migrieren)

- Migration von Workloads: Linux-Validierung vor der Datenaufnahme
- Migration von Workloads: Windows-Validierung vor der Erfassung
- Workload Ingest Stack: Erstellen

Verwenden Sie den AMS Workload Ingest Change Type (CT) mit einem AMS-Cloud-Migrationspartner, um Ihre vorhandenen Workloads in eine von AMS verwaltete VPC zu verschieben. Mithilfe von AMS Workload Ingest können Sie ein benutzerdefiniertes AMS-AMI erstellen, nachdem Sie migrierte Instances auf AMS verschoben haben. In diesem Abschnitt werden der Prozess, die Voraussetzungen und die Schritte beschrieben, die Ihr Migrationspartner und Sie selbst für die Aufnahme von AMS-Workloads ergreifen.

> **Important**
>
> Das Betriebssystem muss von AMS Workload Ingest unterstützt werden. Informationen zu unterstützten Betriebssystemen finden Sie unterMigration von Workloads: Voraussetzungen für Linux und Windows.
> Jeder Workload und jedes Konto ist anders. AMS wird mit Ihnen zusammenarbeiten, um sich auf ein erfolgreiches Ergebnis vorzubereiten.

Das folgende Diagramm zeigt den Prozess zur Erfassung von AMS-Workloads.

Migration von Workloads: Voraussetzungen für Linux und Windows

Bevor Sie eine Kopie einer lokalen Instance in AWS Managed Services (AMS) aufnehmen können, müssen bestimmte Voraussetzungen erfüllt sein. Dies sind die Voraussetzungen, einschließlich der Voraussetzungen, die sich zwischen Windows- und Linux-Betriebssystemen unterscheiden.

> **Note**
>
> Um den Prozess der Feststellung zu vereinfachen, ob die Instanzen für die Aufnahme bereit sind, wurden Validierungstools für Windows und Linux entwickelt. Diese Tools können heruntergeladen und direkt auf Ihren lokalen Servern sowie auf EC2 Instanzen in AWS ausgeführt werden. Linux Pre-WIGS Validation.zip, Windows Pre-WIGS Validation.zip.

BEVOR SIE BEGINNEN, für Linux und Windows:

- Führen Sie einen vollständigen Virenscan durch.
- Die Instanz muss über das `customer-mc-ec2-instance-profile` Instanzprofil verfügen.
- Installieren Sie den Amazon EC2 Systems Manager (SSM) -Agenten und stellen Sie sicher, dass der SSM-Agent betriebsbereit ist.
- Für die Ausführung von AMS Workload Ingest (WIGS) werden mindestens 10 GB freier Festplattenspeicher auf dem Root-Volume empfohlen. In betrieblicher Hinsicht empfiehlt AMS eine Festplattenauslastung von weniger als 75% und gibt eine Warnmeldung aus, wenn die Festplattenauslastung 85% erreicht.
- Legen Sie mit Ihrem Migrationspartner einen Zeitrahmen für die Datenaufnahme fest.
- Das benutzerdefinierte AMI ist als EC2 Instanz im AMS-Zielproduktionskonto vorhanden (dies liegt in der Verantwortung des Migrationspartners).

Important

Das Betriebssystem muss von AMS-Workload Ingest unterstützt werden.

- Die folgenden Betriebssysteme werden unterstützt:
  - Microsoft Windows Server: 2008 R2, 2012, 2012 R2, 2016, 2019 und 2022
  - Linux: Amazon Linux 2023, Amazon Linux 2 und Amazon Linux, CentOS 7.x, CentOS 6.5-6.10, Oracle Linux 7: Nebenversionen 7.5 und höher, Oracle Linux 8: Nebenversionen bis 8.3, RHEL 8.x, RHEL 7.x, RHEL 6.5-6.10, SUSE Linux Enterprise Server 15 SP3 und SAP-spezifische Versionen, SUSE Linux Enterprise Server 12, Ubuntu 18.04 SP4 SP5
- Folgendes wird nicht unterstützt: AMIs
  - Minimales AMI für Amazon Linux 2023.

Note

Die AMS-Endpunkte API/CLI (amscm und amsskms) befinden sich in der AWS-Region Nord-Virginia,. `us-east-1` Je nachdem, wie Ihre Authentifizierung eingerichtet ist und in welcher AWS-Region sich Ihr Konto und Ihre Ressourcen befinden, müssen Sie `--region us-east-1` bei der Ausgabe von Befehlen möglicherweise zusätzliche Informationen

hinzufügen. Möglicherweise müssen Sie auch angeben`--profile saml`, ob dies Ihre Authentifizierungsmethode ist.

LINUX-Voraussetzungen

Beachten Sie die unter aufgeführten Anforderungen Migration von Workloads: Voraussetzungen für Linux und Windows und stellen Sie Folgendes sicher, bevor Sie einen WIGS-RFC einreichen:

- Die neuesten erweiterten Netzwerktreiber sind installiert; siehe Enhanced Networking unter Linux.
- Softwarekomponenten von Drittanbietern, die zu Konflikten mit AMS-Komponenten führen könnten, wurden entfernt:
  - Antiviren-Clients
  - Backup-Clients
  - Virtualisierungssoftware (wie VM Tools oder Hyper-V-Integrationsdienste)
  - Zugriffsverwaltungssoftware (wie SSSD, Centrify oder PBIS)
- Stellen Sie sicher, dass SSH richtig konfiguriert ist — Dadurch wird vorübergehend die Authentifizierung mit privatem Schlüssel für SSH aktiviert. AMS verwendet dies mit unserem Konfigurationsmanagement-Tool. Verwenden Sie diese Befehle:

```
sudo grep -q "^PubkeyAuthentication" /etc/ssh/sshd_config && sudo sed "s/
^PubkeyAuthentication=.*/PubkeyAuthentication yes/" -i /etc/ssh/sshd_config || sudo
 sed "$ a\PubkeyAuthentication yes" -i /etc/ssh/sshd_config
```

```
sudo grep -q "^AuthorizedKeysFile" /etc/ssh/sshd_config && sudo sed "s/
^AuthorizedKeysFile=.*/AuthorizedKeysFile %h\/.ssh\/authorized_keys/" -i /etc/ssh/
sshd_config || sudo sed "$ a\AuthorizedKeysFile %h/.ssh/authorized_keys" -i /etc/ssh/
sshd_config
```

- Stellen Sie sicher, dass Yum richtig konfiguriert ist. Für die Nutzung der Yum-Repositorys ist RedHat eine Lizenzierung erforderlich. Die Instanz muss über einen Satellitenserver oder RedHat Cloud-Server lizenziert werden. Verwenden Sie einen dieser Links, wenn eine Lizenzierung erforderlich ist:
  - Red Hat Satellite
  - RedHat Cloud-Zugang

- Wenn Sie Red Hat Satellite verwenden, erfordert WIGS das Hinzufügen von Red Hat Software Collections (RHSCL). Das WIGS-System verwendet RHSCL, um neben der Konfiguration auf dem System auch einen Python3.6-Interpreter hinzuzufügen. Um diese Lösung zu unterstützen, müssen die folgenden Repositorys verfügbar sein:
  - rhel-server-rhscl
  - rhel-server-releases-optional

Windows -Voraussetzungen

Beachten Sie die unter aufgeführten Anforderungen Migration von Workloads: Voraussetzungen für Linux und Windows und stellen Sie Folgendes sicher, bevor Sie einen WIGS-RFC einreichen:

- Powershell Version 3 oder höher ist installiert.
- AWS EC2 Config ist auf der Instance mit dem Workload installiert, den Sie migrieren werden.
- Installieren Sie die AWS-Treiber, die die Instance-Typen der neuesten Generation unterstützen: PV, ENA und NVMe. Sie können die Informationen in diesen Links verwenden:
  - Aktualisieren von PV-Treibern auf Ihren Windows-Instances
  - Verbessertes Networking unter Windows
  - NVMe AWS-Treiber für Windows-Instances
  - Teil 3: Aktualisieren von NVMe AWS-Treibern
  - Teil 5: Installation des seriellen Port-Treibers für Bare-Metal-Instances
  - Teil 6: Aktualisierung der Energieverwaltungseinstellungen
- (Optional, aber empfohlen) Kritische Dienste deaktivieren — Stellen Sie wichtige Anwendungsdienste wie Datenbanken auf Deaktiviert ein, stellen Sie jedoch sicher, dass alle Änderungen dokumentiert werden, damit sie während der Phase der Anwendungsüberprüfung in den ursprünglichen Startmodus zurückgesetzt werden können.
- (Optional, aber empfohlen) Erstellen Sie ein Failsafe-AMI aus der vorbereiteten Instanz:
  - Verwenden Sie das Deployment | Erweiterte Stack-Komponenten | AMI | Erstellen
  - Fügen Sie während der Erstellung das Tag Key=Name, Value=Application-ID_ hinzu IngestReady
  - Warten Sie, bis AMI erstellt wurde, bevor Sie fortfahren
- Softwarekomponenten von Drittanbietern, die zu Konflikten mit AMS-Komponenten führen könnten, wurden entfernt:

- Antiviren-Clients
- Backup-Clients
- Virtualisierungssoftware (wie VM Tools oder Hyper-V-Integrationsdienste)

> **Note**
>
> Das End-of-Support Migrationsprogramm für Windows Server (EMP) umfasst Tools zur Migration Ihrer Legacy-Anwendungen von Windows Server 2003, 2008 und 2008 R2 auf neuere, unterstützte Versionen auf AWS, ohne dass ein Refactoring erforderlich ist.

Wie Migration Ihre Ressourcen verändert

Der in diesem Abschnitt beschriebene Ingestion-RFC umfasst den nächsten Schritt, nämlich das Hinzufügen von Konfigurationen zur Instance, sobald sie zu Ihrem AMS-Konto migriert wurde, sodass AMS sie verwalten kann.

Die hinzugefügten Konfigurationen sind wie folgt AMS-spezifisch.

Änderungen, die an aufgenommenen Linux-Instances vorgenommen wurden:

- Software, die installiert ist:
  - Cloud Init: Wird verwendet, um private Schlüssel für Jarvis Access zu konfigurieren.
  - Python 3 (Skriptsprache) für alle unterstützten Betriebssysteme (außer CentOS 6, RHEL 8, OracleLinux 7).
  - AWS CloudFormation Python Helper Scripts: AWS CloudFormation stellt Skripts zur Verfügung, die zur Installation von Software und zum Starten von Services auf EC2 Amazon-Instances verwendet werden.
  - AWS CLI: Die AWS-CLI ist ein Open-Source-Tool, das auf dem AWS-SDK für Python (Boto) aufbaut und Befehle für die Interaktion mit AWS-Services bereitstellt.
  - AWS SSM Agent: Der SSM-Agent verarbeitet Anfragen vom Systems Manager Manager-Service und konfiguriert den Computer wie in der Anfrage angegeben.
  - AWS CloudWatch Logs Agent: Sendet Protokolle an CloudWatch.
  - AWS CodeDeploy: Ein Bereitstellungsservice, der Anwendungsbereitstellungen für EC2 Amazon-Instances, lokale Instances oder serverlose Lambda-Funktionen automatisiert.

  - Ruby: Erforderlich für CodeDeploy
  - System Performance Tools (sysstat): Sysstat enthält verschiedene Hilfsprogramme zur Überwachung der Systemleistung und der Nutzungsaktivität.
  - AD Bridge (ehemals PowerBroker Identity Services): Verbindet Hosts, die nicht von Microsoft stammen, mit Active Directory-Domänen.
  - Trend Micro Deep Security Agent: Antivirensoftware.
- Software, die geändert wurde:
  - Die Instanzen sind so konfiguriert, dass sie die UTC-Zeitzone verwenden.

Änderungen, die an aufgenommenen Windows-Instanzen vorgenommen wurden:

- Software, die installiert ist:
  - AWS-Tools für Windows PowerShell: Mit den AWS-Tools PowerShell für können Entwickler und Administratoren ihre AWS-Services und -Ressourcen in der PowerShell Skriptumgebung verwalten.
  - Trend Micro Deep Security Agent: Antiviren-Schutz
  - PowerShell AMS-Module, die PowerShell Code zur Steuerung von Start, Active Directory-Verbindung, Überwachung, Sicherheit und Protokollierung enthalten.
- Software, die geändert wurde:
  - Server Message Block (SMB) Version 1 ist deaktiviert.
  - Windows Remote Management (WinRM) ist aktiviert und für die Überwachung von Port 5986 konfiguriert. Eine Firewallregel, die diesen eingehenden Port zulässt, wird ebenfalls erstellt.
- Software, die möglicherweise installiert oder geändert wurde:
  - Microsoft.Net Framework 4.5 (Entwicklerplattform), wenn eine niedrigere Version als .Net Framework 4.5 erkannt wird.
  - Für Windows 2012 und Windows 2012R2 führen wir ein Upgrade auf PowerShell 5.1 durch.

Migration von Workloads: Standardprozess

> **Note**
>
> Da für diesen Prozess zwei Parteien erforderlich sind, werden in diesem Abschnitt die jeweiligen Aufgaben beschrieben: ein AMS Cloud-Migrationspartner (Migrationspartner) und ein Anwendungseigentümer (Sie).

1. Migrationspartner, Einrichtung:
   a. Der Migrationspartner sendet eine Serviceanfrage an AMS für eine IAM-Rolle, um Ihre Instanz zu migrieren. Einzelheiten zum Einreichen von Serviceanfragen finden Sie unter Beispiele für Serviceanfragen.
   b. Der Migrationspartner reicht eine Administratorzugriffsanfrage ein. Das AMS Operations Team gewährt dem Migrationspartner über die angeforderte IAM-Rolle Zugriff auf Ihr Konto.
2. Migrationspartner, Migrieren einzelner Workloads:
   a. Der Migrationspartner migriert Ihre AWS Nicht-Instance über native Amazon- EC2 oder andere Migrationstools mit dem `customer-mc-ec2-instance-profile` IAM-Instance-Profil (muss sich im Konto befinden) in ein Subnetz in Ihrem AMS-Konto.
   b. Der Migrationspartner reicht einen RFC mit der vom Migrationspartner migrierten Instanz Deployment | Ingestion | Stack from Migration Partner | Create CT (ct-257p9zjk14ija) ein. Weitere Informationen zur Erstellung und Einreichung dieses RFC finden Sie unter. Workload Ingest Stack: Erstellen

      Die Ausführungsausgabe des RFC gibt eine Instanz-ID, eine IP-Adresse und eine AMI-ID zurück.

      Der Migrationspartner stellt Ihnen die Instanz-ID des Workloads zur Verfügung, der in Ihrem Konto erstellt wurde.
3. Sie, greifen auf die Migration zu und validieren sie:
   a. Senden Sie anhand der vom Migrationspartner bereitgestellten Ausführungsausgabe (AMI-ID, Instanz-ID und IP-Adresse) einen Zugriffs-RFC, melden Sie sich beim neu erstellten

AMS-Stack an und überprüfen Sie, ob Ihre Anwendung ordnungsgemäß funktioniert. Einzelheiten finden Sie unter Instanzzugriff beantragen.

b. Wenn Sie zufrieden sind, können Sie die gestartete Instance weiterhin als 1-Tier-Stack and/or verwenden. Verwenden Sie das AMI, um zusätzliche Stacks zu erstellen, einschließlich Auto Scaling Scaling-Gruppen.

c. Wenn Sie mit der Migration nicht zufrieden sind, reichen Sie eine Serviceanfrage ein und verweisen Sie auf den Stack und den RFC. AMS wird mit Ihnen zusammenarbeiten IDs, um Ihre Bedenken auszuräumen.

CloudEndure Der Prozess zur Erfassung von Landingzone-Workloads wird als Nächstes beschrieben.

Migration von Workloads: CloudEndure landing zone (SALZ)

Dieser Abschnitt enthält Informationen zur Einrichtung einer Zwischenmigration mit einem einzigen Konto landing zone) für CloudEndure (CE) -Cutover-Instances, die für einen Workload Ingest (WIGS) -RFC verfügbar sind.

Weitere Informationen finden Sie unter Migration. CloudEndure CloudEndure

> **Note**
>
> Dabei handelt es sich um ein vordefiniertes, sicherheitsverstärktes Migrationsschema und Muster.

Voraussetzungen:

- Ein AMS-Kundenkonto
- Netzwerk- und Zugriffsintegration zwischen dem AMS-Konto und dem Kunden vor Ort
- Ein Konto CloudEndure
- Ein Workflow zur Vorabgenehmigung für eine AMS-Sicherheitsprüfung und -freigabe, der mit Ihrem CA and/or CSDM ausgeführt wird (z. B. bietet der Missbrauch der permanenten Anmeldeinformationen des IAM-Benutzers die Möglichkeit, Instances und Sicherheitsgruppen zu create/delete verwenden)

> **Note**
>
> Spezifische Vorbereitungs- und Migrationsprozesse werden in diesem Abschnitt beschrieben.

Vorbereitung: Sie und der AMS-Betreiber:

1. Bereiten Sie einen Änderungsantrag (Request for Change, RFC) mit dem Management | Sonstige | Andere | Aktualisiere den Änderungstyp für AMS für die folgenden Ressourcen und Updates vor. Sie können ein separates Update | Anderes Update RFCs oder ein einzelnes Update einreichen. Einzelheiten zu diesem RFC/CT finden Sie unter Sonstiges | Anderes Update mit diesen Anfragen:

   a. Weisen Sie Ihrer AMS-VPC einen sekundären CIDR-Block zu, einen temporären CIDR-Block, der nach Abschluss der Migration entfernt wird. Stellen Sie sicher, dass der Block nicht mit bestehenden Routen zurück zu Ihrem lokalen Netzwerk kollidiert. Wenn Ihr AMS VPC CIDR beispielsweise 10.0.0.0/16 ist und es eine Route zurück zu Ihrem lokalen Netzwerk von 10.1.0.0/16 gibt, dann könnte das temporäre sekundäre CIDR 10.255.255.0/24 sein. Informationen zu AWS CIDR-Blöcken finden Sie unter VPC and Subnet Sizing.

   b. Erstellen Sie ein neues, privates Subnetz innerhalb der AMS-VPC mit Initial-Garden. Beispielname:. `migration-temp-subnet`

   c. Erstellen Sie eine neue Routentabelle für das Subnetz mit nur lokalen VPC- und NAT-Routen (Internet), um Konflikte mit dem Quellserver während der Instanzübernahme und mögliche Ausfälle zu vermeiden. Stellen Sie sicher, dass ausgehender Datenverkehr ins Internet für Patch-Downloads zugelassen ist und dass die Voraussetzungen für AMS WIGS heruntergeladen und installiert werden können.

   d. Aktualisieren Sie Ihre Managed AD-Sicherheitsgruppe, um eingehenden und ausgehenden Datenverkehr zuzulassen. to/from `migration-temp-subnet` Fordern Sie außerdem an, dass Ihre EPS-Load Balancer (ELB) -Sicherheitsgruppe (z. B.`mc-eps-McEpsElbPrivateSecurityGroup-M790XBZEEX74`) aktualisiert wird, um das neue, private Subnetz (d. h.) zuzulassen. `migration-temp-subnet` Wenn der Datenverkehr aus dem dedizierten Subnetz CloudEndure (CE) nicht auf allen drei TCP-Ports zulässig ist, schlägt die WIGS-Erfassung fehl.

e. Fordern Sie abschließend eine neue IAM-Richtlinie und einen neuen CloudEndure IAM-Benutzer an. <Customer Application Subnet (s) + Temp Migration Subnet>Für die Richtlinie ist Ihre korrekte Kontonummer erforderlich, und das Subnetz IDs in der `RunInstances` Abrechnung sollte wie folgt lauten: Ihr.

   Um eine vorab von AMS genehmigte CloudEndure IAM-Richtlinie einzusehen: Entpacken Sie die WIGS Cloud Endure Landing Zone-Beispieldatei und öffnen Sie die. `customer_cloud_endure_policy.json`

   > **Note**
   >
   > Wenn Sie eine restriktivere Richtlinie wünschen, besprechen Sie mit Ihrem Ansprechpartner, was Sie benötigen, CloudArchitect/CSDM und lassen Sie sich, falls erforderlich, von einem AMS Security Review und Signoff überzeugen, bevor Sie einen RFC zur Implementierung der Richtlinie einreichen.

2. Ihre Vorbereitungsschritte CloudEndure für die Erfassung von AMS-Workloads sind abgeschlossen, und wenn Ihr Migrationspartner seine Vorbereitungsschritte abgeschlossen hat, kann die Migration durchgeführt werden. Der WIGS-RFC wird von Ihrem Migrationspartner eingereicht.

> **Note**
>
> IAM-Benutzerschlüssel werden nicht direkt geteilt, sondern müssen vom AMS-Mitarbeiter in einer Bildschirmfreigabesitzung in die CloudEndure Managementkonsole eingegeben werden.

Vorbereitung: Migrationspartner und AMS-Betreiber:

1. CloudEndure Migrationsprojekt erstellen.

   a. Lassen Sie AMS während der Projekterstellung die IAM-Benutzeranmeldedaten in Screensharing-Sitzungen eingeben.

   b. Wählen Sie unter Replikationseinstellungen -> Wählen Sie das Subnetz aus, in dem die Replikationsserver gestartet werden sollen, und wählen Sie das Subnetz aus. customer-application-x

   c. Wählen Sie unter Replikationseinstellungen -> Wählen Sie die Sicherheitsgruppen aus, die auf die Replikationsserver angewendet werden sollen, beide Sentinel-Sicherheitsgruppen (Nur privat und). EgressAll

2. Definieren Sie die Umtauschoptionen für die Maschinen (Instanzen).

   a. Subnetz:. migration-temp-subnet

   b. Sicherheitsgruppe: Beide „Sentinel“ -Sicherheitsgruppen (Nur privat und). EgressAll

      Cutover-Instances müssen in der Lage sein, mit dem AMS Managed AD und mit öffentlichen AWS-Endpunkten zu kommunizieren.

   c. Elastische IP: Keine

   d. Öffentliche IP: nein

   e. IAM-Rolle: customer-mc-ec 2-Instanzen-Profil

      Die IAM-Rolle muss die SSM-Kommunikation ermöglichen. Es ist besser, die AMS-Standardeinstellung zu verwenden.

   f. Stellen Sie die Tags gemäß der Konvention ein.

Migration: Migrationspartner:

1. Erstellen Sie einen Dummy-Stack auf AMS. Sie verwenden die Stack-ID, um Zugriff auf die Bastionen zu erhalten.
2. Installieren Sie den CloudEndure (CE) -Agenten auf dem Quellserver. Einzelheiten finden Sie unter Installation der Agents.
3. Erstellen Sie lokale Administratoranmeldedaten auf dem Quellserver.
4. Planen Sie ein kurzes Umstellungsfenster ein und klicken Sie auf Übernahme, wenn Sie bereit sind. Damit ist die Migration abgeschlossen und die Benutzer werden zur AWS-Zielregion weitergeleitet.
5. Beantragen Sie den Zugriff eines Stack-Administrators auf den Dummy-Stack, siehe Anfrage für Administratorzugriff.
6. Melden Sie sich mit den von Ihnen erstellten lokalen Administratoranmeldedaten bei der Bastion und dann bei der Cutover-Instance an.
7. Erstellen Sie ein ausfallsicheres AMI. Einzelheiten zur Erstellung finden Sie AMIs unter AMI Create.

8. Bereiten Sie die Instance für die Aufnahme vor, siehe. Migration von Workloads: Voraussetzungen für Linux und Windows
9. Führen Sie WIGS RFC für die Instanz aus, siehe. Workload Ingest Stack: Erstellen

AMS Tools-Konto (Workloads migrieren)

Ihr Konto für Landing Zone-Tools mit mehreren Konten (mit VPC) beschleunigt die Migration, verbessert Ihre Sicherheitsposition, reduziert Kosten und Komplexität und standardisiert Ihr Nutzungsmuster.

Ein Tools-Konto bietet Folgendes:

- Eine klar definierte Grenze für den Zugriff auf Replikationsinstanzen für Systemintegratoren außerhalb Ihrer Produktions-Workloads.
- Ermöglicht die Einrichtung einer isolierten Kammer, um einen Workload auf Malware oder unbekannte Netzwerkrouten zu überprüfen, bevor er einem Konto mit anderen Workloads zugeordnet wird.
- Da es sich um ein definiertes Konto handelt, bietet es eine schnellere Einarbeitung und Einrichtung für die Migration von Workloads.
- Isolierte Netzwerkrouten zur Sicherung des Datenverkehrs vor Ort -> -> Tools-Konto CloudEndure -> AMS-aufgenommenes Image. Sobald ein Image aufgenommen wurde, können Sie es über einen AMS Management | Advanced Stack Components | AMI | Share (ct-1eiczxw8ihc18) -RFC für das Zielkonto freigeben.

Übergeordnetes Architekturdiagramm:

Verwenden Sie den Änderungstyp Deployment | Managed landing zone | Management account | Tools account (with VPC) erstellen (ct-2j7q1hgf26x5c), um schnell ein Tools-Konto bereitzustellen und einen Workload Ingestion-Prozess in einer Multi-Account-Landingzone-Umgebung zu instanziieren. Siehe Verwaltungskonto, Tools-Konto: Erstellen (mit VPC).

> **Note**
>
> Wir empfehlen, zwei Verfügbarkeitszonen (AZs) einzurichten, da es sich um einen Migrationshub handelt.

Standardmäßig erstellt AMS die folgenden zwei Sicherheitsgruppen (SGs) in jedem Konto. Vergewissern Sie sich, dass diese beiden vorhanden SGs sind. Wenn sie nicht vorhanden sind, öffnen Sie bitte eine neue Serviceanfrage beim AMS-Team, um sie anzufordern.

- SentinelDefaultSecurityGroupPrivateOnlyEgressAll
- InitialGarden-SentinelDefaultSecurityGroupPrivateOnly

Stellen Sie sicher, dass CloudEndure Replikationsinstanzen im privaten Subnetz erstellt werden, in dem es Routen zurück zum Standort gibt. Sie können dies überprüfen, indem Sie sicherstellen, dass die Routing-Tabellen für das private Subnetz über eine Standardroute zurück zu TGW verfügen. Bei einem CloudEndure Computer-Cutover sollte jedoch das „isolierte“ private Subnetz verwendet werden, in dem es keine Route zurück zum lokalen Standort gibt, sondern nur ausgehender Internetverkehr zulässig ist. Es ist wichtig sicherzustellen, dass die Umstellung im isolierten Subnetz erfolgt, um mögliche Probleme mit den Ressourcen vor Ort zu vermeiden.

Voraussetzungen:

1. Entweder Plus - oder Premium-Supportstufe.
2. Das Anwendungskonto IDs für den KMS-Schlüssel, auf dem AMIs sie bereitgestellt werden.
3. Das Tools-Konto, das wie zuvor beschrieben erstellt wurde.

AWS Service zur Anwendungsmigration (AWS MGN)

AWS Der Application Migration Service (AWS MGN) kann in Ihrem MALZ Tools-Konto über die `AWSManagedServicesMigrationRole` IAM-Rolle verwendet werden, die bei der Bereitstellung des Tools-Kontos automatisch erstellt wird. Sie können AWS MGN verwenden, um Anwendungen und Datenbanken zu migrieren, die auf unterstützten Versionen von Windows- und Linux-Betriebssystemen laufen.

Die meisten up-to-date Informationen zum AWS-Region Support finden Sie in der Liste der AWS regionalen Dienste.

Falls Ihr bevorzugtes AWS-Region Produkt derzeit nicht von AWS MGN unterstützt wird oder das Betriebssystem, auf dem Ihre Anwendungen ausgeführt werden, derzeit nicht von AWS MGN

unterstützt wird, sollten Sie in Erwägung ziehen, stattdessen die CloudEndure Migration in Ihrem Tools-Konto zu verwenden.

Beantragen Sie die MGN-Initialisierung AWS

AWS MGN muss vor der ersten Verwendung von AMS initialisiert werden. Um dies für ein neues Tools-Konto zu beantragen, reichen Sie einen RFC Management | Other | Other aus dem Tools-Konto mit den folgenden Angaben ein:

```
RFC Subject=Please initialize AWS MGN in this account
RFC Comment=Please click 'Get started' on the MGN welcome page here:

    https://console.aws.amazon.com/mgn/home?region=MALZ_PRIMARY_REGION#/welcome using
 all default values
    to 'Create template' and complete the initialization process.
```

Sobald AMS den RFC erfolgreich abgeschlossen und AWS MGN in Ihrem Tools-Konto initialisiert hat, können Sie die Standardvorlage `AWSManagedServicesMigrationRole` für Ihre Anforderungen bearbeiten.

Zugriff auf das neue AMS Tools-Konto aktivieren

Sobald das Tools-Konto erstellt wurde, stellt AMS Ihnen eine Konto-ID zur Verfügung. Ihr nächster Schritt besteht darin, den Zugriff auf das neue Konto zu konfigurieren. Dazu gehen Sie wie folgt vor:

1. Aktualisieren Sie die entsprechenden Active Directory-Gruppen auf das entsprechende Konto IDs.

   Neuen, von AMS erstellten Konten wird die ReadOnly Rollenrichtlinie sowie eine Rolle zugewiesen, über die Benutzer Dateien ablegen können. RFCs

   Für das Tools-Konto stehen außerdem eine zusätzliche IAM-Rolle und ein zusätzlicher Benutzer zur Verfügung:

   - IAM-Rolle: `AWSManagedServicesMigrationRole`
   - IAM-Benutzer: `customer_cloud_endure_user`
2. Fordern Sie Richtlinien und Rollen an, damit die Mitglieder des Serviceintegrationsteams Tools der nächsten Generation einrichten können.

   Navigieren Sie zur AMS-Konsole und legen Sie Folgendes ab RFCs:

a. KMS-Schlüssel erstellen. Verwenden Sie entweder KMS-Schlüssel erstellen (auto) oder KMS-Schlüssel erstellen (Überprüfung erforderlich).

   Wenn Sie KMS zum Verschlüsseln aufgenommener Ressourcen verwenden, bietet die Verwendung eines einzigen KMS-Schlüssels, der mit den übrigen Landingzone-Anwendungskonten für mehrere Konten gemeinsam genutzt wird, Sicherheit für aufgenommene Bilder, sodass sie im Zielkonto entschlüsselt werden können.

b. Teilen Sie den KMS-Schlüssel.

   Verwenden Sie den Änderungstyp Verwaltung | Erweiterte Stack-Komponenten | KMS-Schlüssel | Teilen (Überprüfung erforderlich) (ct-05yb337abq3x5), um anzufordern, dass der neue KMS-Schlüssel für Ihre Anwendungskonten freigegeben wird, in denen die Daten gespeichert werden. AMIs

Beispielgrafik einer endgültigen Kontoeinrichtung:

Beispiel für eine vorab genehmigte IAM-Richtlinie von CloudEndure AMS

Um eine vorab von AMS genehmigte CloudEndure IAM-Richtlinie einzusehen:
Entpacken Sie die WIGS Cloud Endure Landing Zone-Beispieldatei und öffnen Sie die.
`customer_cloud_endure_policy.json`

Testen der Konnektivität und end-to-end Einrichtung des AMS Tools-Kontos

1. Beginnen Sie mit der Konfiguration CloudEndure und Installation des CloudEndure Agenten auf einem Server, der auf AMS repliziert wird.
2. Erstellen Sie ein Projekt in CloudEndure.
3. Geben Sie die AWS Anmeldeinformationen ein, die Sie bei der Erfüllung der Voraussetzungen über den Secrets Manager gemeinsam genutzt haben.
4. In den Replikationseinstellungen:
   a. Wählen Sie für die Option Wählen Sie die Sicherheitsgruppen aus, die auf die Replikationsserver angewendet werden sollen, beide AMS- „Sentinel“ -Sicherheitsgruppen (Nur privat und EgressAll) aus.
   b. Definieren Sie Umtauschoptionen für die Maschinen (Instanzen). Weitere Informationen finden Sie in Schritt 5. Überschneiden
   c. Subnetz: Privates Subnetz.

5. Sicherheitsgruppe:
   a. Wählen Sie beide AMS-Sicherheitsgruppen „Sentinel“ aus (Nur privat und EgressAll).
   b. Cutover-Instanzen müssen mit dem von AMS verwalteten Active Directory (MAD) und mit öffentlichen Endpunkten kommunizieren: AWS
      i. Elastische IP: Keine
      ii. Öffentliche IP: nein
      iii. IAM-Rolle: customer-mc-ec 2-Instanzen-Profil
   c. Legen Sie Tags gemäß Ihrer internen Tagging-Konvention fest.
6. Installieren Sie den CloudEndure Agenten auf dem Computer und suchen Sie in der EC2 Konsole nach der Replikationsinstanz, die in Ihrem AMS-Konto angezeigt werden soll.

Der AMS-Aufnahmeprozess:

AMS Tools Kontohygiene

Sie sollten aufräumen, nachdem Sie mit dem Konto fertig sind, das AMI gemeinsam genutzt haben und die replizierten Instances nicht mehr benötigen:

- Nach der Inanspruchnahme der WIGs Instance:
  - Cutover-Instance: Stoppen oder beenden Sie diese Instanz mindestens, nachdem die Arbeit abgeschlossen ist, über die AWS-Konsole
  - AMI-Backups vor der Ingestion: Wird entfernt, sobald die Instance aufgenommen und die On-Premise-Instance beendet wurde
  - AMS-Ingested Instances: Schalten Sie den Stack aus oder beenden Sie ihn, sobald das AMI gemeinsam genutzt wurde
  - AMS-Ingested AMIs: Wird gelöscht, sobald die gemeinsame Nutzung mit dem Zielkonto abgeschlossen ist
- Bereinigung am Ende der Migration: Dokumentieren Sie die Ressourcen, die im Entwicklermodus bereitgestellt wurden, um sicherzustellen, dass die Bereinigung regelmäßig erfolgt, zum Beispiel:
  - Sicherheitsgruppen
  - Ressourcen, die über Cloud-Formation erstellt wurden
  - Netzwerk ACK
  - Subnetz

- VPC
- Routing-Tabelle
- Rollen
- Benutzer und Konten

Migration im großen Maßstab - Migration Factory

Weitere Informationen finden Sie unter Einführung in die AWS CloudEndure Migration Factory-Lösung.

Migration von Workloads: Linux-Validierung vor der Datenaufnahme

Sie können überprüfen, ob Ihre Instance für die Aufnahme in Ihr AMS-Konto bereit ist. Bei der Validierung vor der Aufnahme von Workload Ingest (WIGS) werden u. a. der Betriebssystemtyp, der verfügbare Festplattenspeicher, das Vorhandensein von widersprüchlicher Drittanbietersoftware usw. geprüft. Bei der Ausführung erzeugt die WIGS-Validierung vor der Datenaufnahme eine Tabelle auf dem Bildschirm mit einer optionalen Protokolldatei. Die Ergebnisse enthalten einen pass/fail Status für jede Validierungsprüfung sowie den Grund für etwaige Fehler. Darüber hinaus können Sie die Validierungstests an Ihre Bedürfnisse anpassen.

Häufig gestellte Fragen:

- Wie verwende ich die Linux WIGS Pre-Ingestion-Validierung?

  Gehen Sie wie folgt vor, um die AMS Linux WIGS-Validierungsskripte vor der Datenaufnahme herunterzuladen und zu verwenden:

  1. Laden Sie eine ZIP-Datei mit den Validierungsskripten herunter

     ZIP-Datei zur Validierung vor der Installation von Linux WIGS.
  2. Entpacken Sie die angehängten Regeln in ein Verzeichnis Ihrer Wahl.
  3. Folgen Sie den Anweisungen in der Datei readme.md.
- Welche Validierungen werden im Rahmen der Linux WIGS-Validierung vor der Datenaufnahme durchgeführt?

  Die AMS Linux WIGS-Lösung zur Validierung vor der Datenaufnahme validiert Folgendes:

  1. Auf dem Startvolume sind mindestens 5 Gigabyte frei.
  2. Das Betriebssystem wird von AMS unterstützt.

3. Die Instanz hat ein bestimmtes Instanzprofil.
4. Die Instanz enthält keine Antiviren- oder Virtualisierungssoftware.
5. SSH ist ordnungsgemäß konfiguriert.
6. Die Instanz hat Zugriff auf Yum-Repositories.
7. Verbesserte Netzwerktreiber sind installiert.
8. Die Instanz hat den SSM-Agenten und er läuft.

- Warum wird eine benutzerdefinierte Konfigurationsdatei unterstützt?

  Die Skripts sind so konzipiert, dass sie sowohl auf physischen Servern vor Ort als auch auf EC2 AWS-Instanzen ausgeführt werden können. Wie in der obigen Liste gezeigt, schlagen einige Tests jedoch fehl, wenn sie vor Ort ausgeführt werden. Beispielsweise hätte ein physischer Server in einem Rechenzentrum kein Instanzprofil. In solchen Fällen können Sie die Konfigurationsdatei bearbeiten, um den Test des Instanzprofils zu überspringen, um Verwirrung zu vermeiden.

- Wie stelle ich sicher, dass ich die neueste Version des Skripts habe?

  Eine up-to-date Version der Linux WIGS Pre-Ingestion Validation Solution wird im Abschnitt AMS Helper Files auf der Hauptdokumentationsseite verfügbar sein.

- Ist das Skript schreibgeschützt?

  Das Skript ist mit Ausnahme der erstellten Protokolldateien so konzipiert, dass es schreibgeschützt ist. Es sollten jedoch bewährte Methoden befolgt werden, um das Skript in einer Umgebung außerhalb der Produktionsumgebung auszuführen.

- Ist die WIGS-Validierung vor der Datenaufnahme für Windows verfügbar?

  Ja. Sie ist im Abschnitt AMS-Hilfsdateien auf der Hauptdokumentationsseite verfügbar.

Migration von Workloads: Windows-Validierung vor der Erfassung

Sie können das WIGs Pre-Validator-Skript verwenden, um zu überprüfen, ob Ihre Instance für die Aufnahme in Ihr AMS-Konto bereit ist. Bei der Validierung vor der Aufnahme von Workload Ingest (WIGS) werden u. a. der Betriebssystemtyp, der verfügbare Festplattenspeicher, das Vorhandensein von widersprüchlicher Drittanbietersoftware usw. geprüft. Wenn sie ausgeführt wird, erzeugt die WIGS-Validierung vor der Datenaufnahme eine Bildschirmtabelle und eine optionale Protokolldatei. Die Ergebnisse enthalten einen pass/fail Status für jede Validierungsprüfung zusammen mit der Fehlerursache. Darüber hinaus können Sie die Validierungstests anpassen.

Häufig gestellte Fragen:

- Wie verwende ich die Windows WIGS-Validierung vor der Datenaufnahme?

  Sie können die Validierung über eine grafische Benutzeroberfläche und einen Webbrowser ausführen, oder Sie können Windows PowerShell, SSM Run Command oder SSM Session Manager verwenden.

  Option 1: Von einer GUI und einem Webbrowser aus ausführen

  Gehen Sie wie folgt vor, um die WIGs Windows-Vorvalidierung von einer GUI und einem Webbrowser aus auszuführen:

  1. Laden Sie eine ZIP-Datei mit den Überprüfungsskripten herunter:

     ZIP-Datei zur Überprüfung vor der Erfassung von Windows WIGS.

  2. Entpacken Sie die angehängten Regeln in ein Verzeichnis Ihrer Wahl.
  3. Folgen Sie den Anweisungen in der Datei README.md.

  Option 2: Unter Windows PowerShell, SSM Run Command oder SSM Session Manager ausführen

  Windows 2016 und höher

  1. Laden Sie die ZIP-Datei mit den Überprüfungsskripten herunter.

     ```
     $DestinationFile = "$env:TEMP\WIGValidation.zip"

     $Bucket = 'https://docs.aws.amazon.com/managedservices/latest/appguide/samples/
     windows-prewigs-validation.zip'
     $DestinationFile = "$env:TEMP\WIGValidation.zip"
     $ScriptFolder = "$env:TEMP\AWSManagedServices.PreWigs.Validation"
     ```

  2. Entfernen Sie vorhandene Dateien von`C:\Users\AppData\Local\Temp\AWSManagedServices.PreWigs.Validation`.

     ```
     Remove-Item $scriptFolder -Recurse -Force -ErrorAction Ignore
     ```

  3. Rufen Sie das Skript auf.

     ```
     Invoke-WebRequest -Uri $bucket -OutFile $DestinationFile
     Add-Type -Assembly "system.io.compression.filesystem"
     ```

4. Entpacken Sie angehängte Dateien in ein Verzeichnis Ihrer Wahl.

```
[io.compression.zipfile]::ExtractToDirectory($DestinationFile, $env:TEMP)
```

5. Führen Sie das Überprüfungsskript interaktiv aus und sehen Sie sich die Ergebnisse an.

```
Import-Module .\AWSManagedServices.PreWigs.Validation.psm1 -force
Invoke-PreWIGsValidation -RunWithoutExitCodes
```

6. (Optional) Um die im Abschnitt Exit-Codes aufgelisteten Fehlercodes zu erfassen, führen Sie das Skript ohne die `RunWithoutExitCodes` Option aus. Beachten Sie, dass dieser Befehl die aktive PowerShell Sitzung beendet.

```
Import-Module .\AWSManagedServices.PreWigs.Validation.psm1 -force
Invoke-PreWIGsValidation
```

Windows 2012 R2 und früher

Wenn Sie Windows Server 2012 R2 oder niedriger ausführen, müssen Sie TLS einrichten, bevor Sie die ZIP-Datei herunterladen. Gehen Sie wie folgt vor, um TLS einzurichten:

1. Laden Sie die ZIP-Datei mit den Überprüfungsskripten herunter.

```
$DestinationFile = "$env:TEMP\WIGValidation.zip"

$Bucket = 'https://docs.aws.amazon.com/managedservices/latest/appguide/samples/
windows-prewigs-validation.zip'
$DestinationFile = "$env:TEMP\WIGValidation.zip"
$ScriptFolder = "$env:TEMP\AWSManagedServices.PreWigs.Validation"
```

2. Wenn Validierungsdateien vorhanden sind, entfernen Sie diese.

```
Remove-Item $scriptFolder -Recurse -Force -ErrorAction Ignore
```

3. Stellen Sie die TLS-Version ein.

```
[System.Net.ServicePointManager]::SecurityProtocol = 'TLS12'
```

4. Laden Sie die WIG-Validierung herunter.

```
Invoke-WebRequest -Uri $bucket -OutFile $DestinationFile
```

```
Add-Type -Assembly "system.io.compression.filesystem"
```

5. Entpacken Sie die angehängten Regeln in ein Verzeichnis Ihrer Wahl.

```
[io.compression.zipfile]::ExtractToDirectory($DestinationFile, $env:TEMP)
```

6. Führen Sie das Überprüfungsskript interaktiv aus und sehen Sie sich die Ergebnisse an.

```
Import-Module .\AWSManagedServices.PreWigs.Validation.psm1 -force
Invoke-PreWIGsValidation -RunWithoutExitCodes
```

7. (Optional) Um die im Abschnitt Exit-Codes aufgelisteten Fehlercodes zu erfassen, führen Sie das Skript ohne die RunWithoutExitCodes Option aus. Beachten Sie, dass dieser Befehl die aktive PowerShell Sitzung beendet.

```
Import-Module .\AWSManagedServices.PreWigs.Validation.psm1 -force
Invoke-PreWIGsValidation
```

> **Note**
>
> Sie können die PowerShell Skripts herunterladen und ausführen. Laden Sie dazu die pre-wigs-validation-powershellDatei -scripts.zip herunter.

- Welche Validierungen werden mit der Windows WIGS Pre-Ingestion Validation durchgeführt?

  Die AMS Windows WIGS-Lösung zur Validierung vor der Datenaufnahme validiert Folgendes:

  1. Auf dem Startvolume sind mindestens 10 Gigabyte frei.
  2. Das Betriebssystem wird von AMS unterstützt.
  3. Die Instanz hat ein bestimmtes Instanzprofil.
  4. Die Instanz enthält keine Antiviren- oder Virtualisierungssoftware.
  5. DHCP ist auf mindestens einem Netzwerkadapter aktiviert.
  6. Die Instanz ist bereit für Sysprep.
     - Für 2008 R2 und 2012 Base und R2 überprüft Sysprep, dass:
       - Es gibt eine Datei unattend.xml
       - Die Datei sppnp.dll (falls vorhanden) ist nicht beschädigt
       - Das Betriebssystem wurde nicht aktualisiert

    - Sysprep wurde nicht öfter als die maximale Anzahl von Malen gemäß Microsoft-Richtlinien ausgeführt.
  - Für 2016 und höher werden alle oben genannten Prüfungen übersprungen, da sie für das jeweilige Betriebssystem keine Probleme bereiten
7. Das WMI-Subsystem (Windows Management Instrumentation) ist fehlerfrei.
8. Die erforderlichen Treiber sind installiert.
9. Der SSM-Agent ist installiert und läuft.
10. Es wird eine Warnung ausgegeben, um zu überprüfen, ob sich der Computer aufgrund der RDS-Lizenzkonfiguration in der Übergangszeit befindet.
11. Die erforderlichen Registrierungsschlüssel sind ordnungsgemäß festgelegt. Weitere Informationen finden Sie in der README-Datei in der ZIP-Datei „Pre-Ingestion Validation“.

- Warum wird eine benutzerdefinierte Konfigurationsdatei unterstützt?

  Die Skripts sind so konzipiert, dass sie sowohl auf physischen Servern vor Ort als auch auf EC2 AWS-Instanzen ausgeführt werden können. Wie in der obigen Liste gezeigt, schlagen einige Tests jedoch fehl, wenn sie vor Ort ausgeführt werden. Beispielsweise hätte ein physischer Server in einem Rechenzentrum kein Instanzprofil. In solchen Fällen können Sie die Konfigurationsdatei bearbeiten, um den Test des Instanzprofils zu überspringen, um Verwirrung zu vermeiden.
- Wie stelle ich sicher, dass ich die neueste Version des Skripts habe?

  Eine up-to-date Version der Windows WIGS-Lösung zur Validierung vor der Datenaufnahme wird im Abschnitt AMS Helper Files auf der Hauptdokumentationsseite verfügbar sein.
- Ist das Skript schreibgeschützt?

  Das Skript ist mit Ausnahme der erstellten Protokolldateien so konzipiert, dass es schreibgeschützt ist. Es sollten jedoch bewährte Methoden befolgt werden, um das Skript in einer Umgebung außerhalb der Produktionsumgebung auszuführen.
- Ist WIGS Pre-Ingestion Validation für Linux verfügbar?

  Ja. Die Linux-Version wurde am 31. Oktober 2019 veröffentlicht. Sie ist im Abschnitt AMS Helper Files auf der Hauptdokumentationsseite verfügbar.

Workload Ingest Stack: Erstellen

Migrieren Sie eine Instanz mit der Konsole zu einem AMS-Stack

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.
3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.
4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.

5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Brechen Sie optional den RFC ab oder erstellen Sie eine Kopie davon mit den Optionen oben auf der Seite.

> **Note**
>
> Wenn der RFC abgelehnt wird, enthält die Ausführungsausgabe einen Link zu CloudWatch Amazon-Protokollen. AMS Workload Ingest (WIGS) RFCs werden abgelehnt, wenn die Anforderungen nicht erfüllt sind, z. B. wenn auf der Instance Antivirensoftware erkannt wird. Die CloudWatch Protokolle enthalten Informationen über die fehlgeschlagene Anforderung und die zur Behebung zu ergreifenden Maßnahmen.

Migrieren einer Instanz zu einem AMS-Stack mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--`

notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}" Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

Sie können die AMS-CLI verwenden, um eine AMS-Instanz aus einer Nicht-AMS-Instanz zu erstellen, die auf ein AMS-Konto migriert wurde.

> **Note**
>
> Stellen Sie sicher, dass Sie die Voraussetzungen erfüllt haben. Weitere Informationen finden Sie unter Migration von Workloads: Voraussetzungen für Linux und Windows.

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm create-rfc --change-type-id "ct-257p9zjk14ija" --change-type-version "2.0" --
title "AMS-WIG-TEST-NO-ACTION" --execution-parameters "{\"InstanceId\":\"INSTANCE_ID\",
\"TargetVpcId\":\"VPC_ID\",\"TargetSubnetId\":\"SUBNET_ID\",\"TargetInstanceType\":
\"t2.large\",\"ApplyInstanceValidation\":true,\"Name\":\"WIG-TEST\",\"Description\":
\"WIG-TEST\",\"EnforceIMDSV2\":\"false\"}"
```

VORLAGE ERSTELLEN:

1. 0Geben Sie das JSON-Schema der Ausführungsparameter für diesen Änderungstyp in eine Datei aus. Beispiel nennt sie MigrateStackParams .json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-257p9zjk14ija" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > MigrateStackParams.json
   ```

2. Ändern und speichern Sie die JSON-Datei mit den Ausführungsparametern. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"InstanceId":          "MIGRATED_INSTANCE_ID",
"TargetVpcId":         "VPC_ID",
"TargetSubnetId":      "SUBNET_ID",
"Name":                "Migrated-Stack",
"Description":         "Create-Migrated-Stack",
"EnforceIMDSV2":       "false"
}
```

3. Gibt die JSON-Datei mit der RFC-Vorlage aus. Das Beispiel nennt sie MigrateStackRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > MigrateStackRfc.json
```

4. Ändern und speichern Sie die MigrateStackRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeId":         "ct-257p9zjk14ija",
"ChangeTypeVersion":    "2.0",
"Title":                "Migrate-Stack-RFC"
}
```

5. Erstellen Sie den RFC und geben Sie die MigrateStackRfc Datei und die MigrateStackParams Datei an:

```
aws amscm create-rfc --cli-input-json file://MigrateStackRfc.json  --execution-
parameters file://MigrateStackParams.json
```

   Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

   Die neue Instanz wird in der Instanzliste für das Konto des Anwendungsbesitzers für die entsprechende VPC angezeigt.

6. Sobald der RFC erfolgreich abgeschlossen wurde, benachrichtigen Sie den Anwendungsbesitzer, damit er sich bei der neuen Instanz anmelden und überprüfen kann, ob der Workload betriebsbereit ist.

> **Note**
>
> Wenn der RFC abgelehnt wird, enthält die Ausführungsausgabe einen Link zu CloudWatch Amazon-Protokollen. AMS Workload Ingest (WIGS) RFCs werden abgelehnt, wenn die Anforderungen nicht erfüllt sind, z. B. wenn auf der Instance Antivirensoftware erkannt wird. Die CloudWatch Protokolle enthalten Informationen über die fehlgeschlagene Anforderung und die zur Behebung zu ergreifenden Maßnahmen.

Tipps

> **Note**
>
> Vergewissern Sie sich, dass Sie die Voraussetzungen erfüllt haben. Weitere Informationen finden Sie unter Migration von Workloads: Voraussetzungen für Linux und Windows.

> **Note**
>
> Wenn ein Tag auf der zu migrierenden Instanz denselben Schlüssel hat wie ein im RFC bereitgestelltes Tag, schlägt der RFC fehl.

> **Note**
>
> Sie können bis zu vier Ziel- IDs, Ports- und Availability Zones angeben.

> **Note**
>
> Wenn der RFC abgelehnt wird, enthält die Ausführungsausgabe einen Link zu CloudWatch Amazon-Protokollen. AMS Workload Ingest (WIGS) RFCs werden abgelehnt, wenn die Anforderungen nicht erfüllt sind, z. B. wenn auf der Instance Antivirensoftware erkannt wird. Die CloudWatch Protokolle enthalten Informationen über die fehlgeschlagene Anforderung und die zur Behebung zu ergreifenden Maßnahmen.

> **Note**
>
> Wenn der RFC abgelehnt wird, enthält die Ausführungsausgabe einen Link zu CloudWatch Amazon-Protokollen. AMS Workload Ingest (WIGS) RFCs werden abgelehnt, wenn die Anforderungen nicht erfüllt sind, z. B. wenn auf der Instance Antivirensoftware erkannt wird. Die CloudWatch Protokolle enthalten Informationen über die fehlgeschlagene Anforderung und die zur Behebung zu ergreifenden Maßnahmen.

Falls erforderlich, finden Sie weitere Informationen unter Fehler bei der Workload Ingestion (WIGS).

CloudFormation AMS-Aufnahme

Mit dem AMS AWS CloudFormation Ingest Change Type (CT) können Sie Ihre vorhandenen CloudFormation Vorlagen mit einigen Änderungen verwenden, um benutzerdefinierte Stacks in einer von AMS verwalteten VPC bereitzustellen.

Themen

- AWS CloudFormation Richtlinien, bewährte Methoden und Einschränkungen für die Datenaufnahme
- AWS CloudFormation Ingest: Beispiele
- Erstellen Sie einen Ingest-Stack CloudFormation
- AWS CloudFormation Aktualisieren Sie den Ingest-Stack
- CloudFormation Genehmigen Sie einen Changeset für den Ingest-Stack
- Kündigungsschutz für AWS CloudFormation Update-Stacks
- Automatisierte IAM-Bereitstellungen mithilfe von CFN-Ingest oder Stack-Update in AMS CTs

Der AWS CloudFormation AMS-Ingest-Prozess umfasst Folgendes:

- Bereiten Sie Ihre benutzerdefinierte CloudFormation Vorlage vor und laden Sie sie in einen S3-Bucket hoch, oder stellen Sie die Vorlage direkt bei der Erstellung des RFC bereit. Wenn Sie einen S3-Bucket mit einer vorsignierten URL verwenden, finden Sie weitere Informationen unter presign.
- Senden Sie den Typ der CloudFormation Ingest-Änderung in einem RFC an AMS. Eine exemplarische Vorgehensweise für den CFN-Ingest-Änderungstyp finden Sie unter. Erstellen

Sie einen Ingest-Stack CloudFormation Beispiele für CFN-Ingest finden Sie unter. AWS CloudFormation Ingest: Beispiele

- Sobald Ihr Stack erstellt ist, können Sie ihn aktualisieren und Abweichungen beheben. Sollte das Update fehlschlagen, können Sie das Update außerdem explizit genehmigen und implementieren. All diese Verfahren werden in diesem Abschnitt beschrieben.

  Informationen zur CFN-Drifterkennung finden Sie unter Neu — CloudFormation Drifterkennung.

> **Note**
>
> - Dieser Änderungstyp hat jetzt eine Version 2.0. Version 2.0 ist automatisiert und wird nicht manuell ausgeführt. Dadurch kann die CT-Ausführung schneller vonstatten gehen. Mit dieser Version werden zwei neue Parameter eingeführt: CloudFormationTemplate, mit denen Sie eine benutzerdefinierte CloudFormation Vorlage in den RFC einfügen können, und VpcId, wodurch CloudFormation Ingest mit der AMS-Landingzone für mehrere Konten verwendet werden kann.
> - Version 1.0 ist ein manueller Änderungstyp. Das bedeutet, dass ein AMS-Betreiber einige Maßnahmen ergreifen muss, bevor der Änderungstyp erfolgreich abgeschlossen werden kann. Es ist mindestens eine Überprüfung erforderlich. Für diese Version muss der CloudFormationTemplateS3Endpoint-Parameterwert außerdem eine vorsignierte URL sein.

AWS CloudFormation Richtlinien, bewährte Methoden und Einschränkungen für die Datenaufnahme

Damit AMS Ihre CloudFormation Vorlage verarbeiten kann, gibt es einige Richtlinien und Einschränkungen.

Richtlinien

Beachten Sie die folgenden Richtlinien, um AWS CloudFormation Fehler beim AWS CloudFormation Ingest zu reduzieren:

- Betten Sie keine Anmeldeinformationen oder andere vertrauliche Informationen in die Vorlage ein — Die CloudFormation Vorlage ist in der AWS CloudFormation Konsole sichtbar, sodass Sie keine Anmeldeinformationen oder vertraulichen Daten in die Vorlage einbetten möchten. Die Vorlage darf

keine vertraulichen Informationen enthalten. Die folgenden Ressourcen sind nur zulässig, wenn Sie AWS Secrets Manager für den Wert verwenden:

- `AWS::RDS::DBInstance` - [MasterUserPassword,TdeCredentialPassword]
- `AWS::RDS::DBCluster` - [MasterUserPassword]
- `AWS::ElastiCache::ReplicationGroup` - [AuthToken]

> **Note**
>
> Informationen zur Verwendung eines AWS Secrets Manager-Geheimnisses in einer Ressourceneigenschaft finden Sie unter Erstellen und Abrufen von in AWS Secrets Manager verwalteten Geheimnissen mithilfe von CloudFormation AWS-Vorlagen und Verwenden dynamischer Referenzen zur Angabe von Vorlagenwerten.

- Verwenden Sie Amazon RDS-Snapshots, um RDS-DB-Instances zu erstellen — Auf diese Weise müssen Sie keine angeben. MasterUserPassword
- Wenn die von Ihnen eingereichte Vorlage ein IAM-Instance-Profil enthält, muss ihr das Präfix „Kunde“ vorangestellt werden. Die Verwendung eines Instanzprofils mit dem Namen " example-instance-profile führt beispielsweise zu einem Fehler. Verwenden Sie stattdessen ein Instanzprofil mit dem Namen 'customer-example-instance-profile'.
- Nehmen Sie keine sensiblen Daten in **`AWS::EC2::Instance`** - [UserData] auf. UserData sollte keine Passwörter, API-Schlüssel oder andere sensible Daten enthalten. Diese Art von Daten kann verschlüsselt und in einem S3-Bucket gespeichert und mithilfe von UserData.
- Die Erstellung von IAM-Richtlinien mithilfe von CloudFormation Vorlagen wird mit Einschränkungen unterstützt — IAM-Richtlinien müssen von AMS geprüft und genehmigt werden. SecOps Derzeit unterstützen wir nur die Bereitstellung von IAM-Rollen mit Inline-Richtlinien, die vorab genehmigte Berechtigungen enthalten. In anderen Fällen können IAM-Richtlinien nicht mithilfe von CloudFormation Vorlagen erstellt werden, da dies den AMS-Prozess außer Kraft setzen würde. SecOps
- SSH werden KeyPairs nicht unterstützt — EC2 Amazon-Instances müssen über das AMS Access Management System aufgerufen werden. Der AMS RFC-Prozess authentifiziert Sie. Sie können keine SSH-Schlüsselpaare in CloudFormation Vorlagen aufnehmen, da Sie nicht über die erforderlichen Berechtigungen verfügen, um SSH-Schlüsselpaare zu erstellen und das AMS-Zugriffsverwaltungsmodell zu überschreiben.

- Die Regeln für den Zugriff auf Sicherheitsgruppen sind eingeschränkt — Sie können keinen Quell-CIDR-Bereich von 0.0.0.0/0 oder einen öffentlich routbaren Adressraum mit einem TCP-Port verwenden, der etwas anderes als 80 oder 443 ist.
- Beachten Sie beim Schreiben von CloudFormation Ressourcenvorlagen die AWS CloudFormation Richtlinien — Stellen Sie sicher, dass Sie den richtigen type/property Datennamen für die Ressource verwenden, indem Sie im Benutzerhandbuch für diese Ressource nachschlagen.AWS CloudFormation Der Datentyp einer SecurityGroupIds Eigenschaft in einer AWS::EC2::Instance Ressource ist beispielsweise „Liste der Zeichenkettenwerte“, sodass ["sg-aaaaaaaa"] in Ordnung ist (mit Klammern), aber „sg-aaaaaaaa“ nicht (ohne Klammern).

  Weitere Informationen finden Sie unter AWS-Referenz zu Ressourcen- und Eigenschaftstypen.
- Konfigurieren Sie Ihre benutzerdefinierten CloudFormation Vorlagen so, dass sie im CloudFormation AMS-Ingest-CT definierte Parameter verwenden — Wenn Sie Ihre CloudFormation Vorlage so konfigurieren, dass sie im CloudFormation AMS-Ingest-CT definierte Parameter verwendet, können Sie die CloudFormation Vorlage wiederverwenden, um ähnliche Stapel zu erstellen, indem Sie sie mit geänderten Parameterwerten in der CT-Eingabe über Management | Benutzerdefinierter Stack | Stack aus CloudFormation Vorlage | Update CT (ct-361tlo1k7339x) einreichen. Ein Beispiel finden Sie unter AWS CloudFormation Ingest-Beispiele: Ressourcen definieren.
- Amazon S3 S3-Bucket-Endpunkte mit einer vorsignierten URL können nicht abgelaufen sein — Wenn Sie einen Amazon S3 S3-Bucket-Endpunkt mit einer vorsignierten URL verwenden, stellen Sie sicher, dass die vorsignierte Amazon S3 S3-URL nicht abgelaufen ist. Ein CloudFormation Ingest-RFC, der mit einer abgelaufenen, vorsignierten Amazon S3 S3-Bucket-URL eingereicht wurde, wird abgelehnt.
- Wait Condition erfordert Signallogik — Wait Condition wird verwendet, um die Erstellung von Stack-Ressourcen mit Konfigurationsaktionen zu koordinieren, die außerhalb der Stack-Erstellung liegen. Wenn Sie die Ressource AWS CloudFormation Wait Condition in der Vorlage verwenden, wartet sie auf ein Erfolgssignal und markiert die Stackerstellung als fehlgeschlagen, wenn die Anzahl der Erfolgssignale nicht gegeben wird. Sie benötigen eine Logik für das Signal, wenn Sie die Ressource Wait Condition verwenden. Weitere Informationen finden Sie unter Wartebedingungen in einer Vorlage erstellen.

Bewährte Methoden

Im Folgenden finden Sie einige bewährte Methoden, mit denen Sie Ressourcen mithilfe des AWS CloudFormation AMS-Ingest-Prozesses migrieren können:

- Senden Sie IAM und andere richtlinienbezogene Ressourcen in einem CT — Wenn Sie automatisierte Funktionen CTs wie CloudFormation Ingest für die Bereitstellung von IAM-Rollen verwenden können, empfehlen wir Ihnen, dies zu tun. In anderen Fällen empfiehlt AMS, dass Sie alle IAM- oder anderen richtlinienbezogenen Ressourcen zusammenfassen und sie in einem einzigen Bereich unter Verwaltung | Andere | Andere | Änderungstyp erstellen einreichen (ct-1e1xtak34nx76). Kombinieren Sie beispielsweise alle benötigten IAM-Rollen, EC2 IAM-Amazon-Instance-Profile, IAM-Richtlinienaktualisierungen für bestehende IAM-Rollen, Amazon S3-Bucket-Richtlinien, Amazon SNS/Amazon SQS-Richtlinien usw. und reichen Sie einen ct-1e1xtak34nx76-RFC ein, sodass diese bereits vorhandenen Ressourcen einfach in den future Ingest-Vorlagen referenziert werden können. CloudFormation
- EC2 Instances werden gebootet und erfolgreich mit der Domain verknüpft — dies wird als bewährte Methode automatisch durchgeführt. Um sicherzustellen, dass die über einen CloudFormation Ingest-Stack gestarteten EC2 Amazon-Instances gebootet werden und der Domain erfolgreich beitreten, enthält AMS eine CreationPolicy und eine UpdatePolicy für eine Auto Scaling Scaling-Gruppenressource (sofern diese Richtlinien noch nicht existieren).
- Der Amazon RDS-DB-Instance-Parameter muss angegeben werden — Wenn Sie eine Amazon RDS-Datenbank per AWS CloudFormation Ingest erstellen, müssen Sie den `DBSnapshotIdentifier` Parameter angeben, um eine Wiederherstellung aus einem früheren DB-Snapshot durchzuführen. Dies ist erforderlich, da AWS CloudFormation Ingest derzeit keine sensiblen Daten verarbeitet.

Ein Beispiel für die Verwendung einer CloudFormation Vorlage für die Aufnahme von CloudFormation AMS-Vorlagen finden Sie unter. AWS CloudFormation Ingest: Beispiele

Validierung von Vorlagen

Sie können Ihre CloudFormation Vorlage selbst validieren, bevor Sie sie an AMS senden.

Bei AMS AWS CloudFormation Ingest eingereichte Vorlagen werden validiert, um sicherzustellen, dass sie sicher in einem AMS-Konto bereitgestellt werden können. Der Validierungsprozess überprüft Folgendes:

- Unterstützte Ressourcen — Es werden nur Ressourcen verwendet, die AWS CloudFormation von AMS Ingest unterstützt werden. Weitere Informationen finden Sie unter Unterstützte Ressourcen.
- Unterstützt AMIs — Das AMI in der Vorlage ist ein AMS-unterstütztes AMI. Informationen zu AMS AMIs finden Sie unter. AMS Amazon-Maschinenbilder (AMIs)

- AMS Shared Services-Subnetz — Die Vorlage versucht nicht, Ressourcen in das AMS Shared Services-Subnetz zu starten.
- Ressourcenrichtlinien — Es gibt keine übermäßig freizügigen Ressourcenrichtlinien, wie z. B. eine öffentlich lesbare oder beschreibbare S3-Bucket-Richtlinie. AMS erlaubt keine öffentlich lesbaren oder beschreibbaren S3-Buckets in. AWS-Konten

Mit Linter validieren AWS CloudFormation

Sie können Ihre CloudFormation Vorlage mit dem Linter-Tool selbst validieren, bevor Sie sie an AMS senden. AWS CloudFormation

Das AWS CloudFormation Linter-Tool ist die beste Methode, um Ihre CloudFormation Vorlage zu validieren, da es die Validierung von resource/property Namen, Datentypen und Funktionen ermöglicht. Weitere Informationen finden Sie unter cfn-python-lintaws-cloudformation/.

Die AWS CloudFormation Linter-Ausgabe der zuvor gezeigten Vorlage sieht wie folgt aus:

```
$ cfn-lint -t ./testtmpl.json
E3002 Invalid Property Resources/SNSTopic/Properties/Name
./testtmpl.json:6:9
```

Um die Offline-Validierung von CloudFormation Vorlagen zu unterstützen, hat AMS eine Reihe von austauschbaren benutzerdefinierten Validierungsregeln für das AWS CloudFormation Linter-Tool entwickelt. Sie befinden sich auf der Seite Developers Resources der AMS-Konsole.

Gehen Sie wie folgt vor, um AWS CloudFormation Validierungsskripten vor der Aufnahme zu verwenden:

1. Installieren Sie das Linter-Tool AWS CloudFormation . Installationsanweisungen finden Sie unter aws-cloudformation /cfn-lint.
2. Laden Sie eine ZIP-Datei mit Überprüfungsskripten herunter:

   Benutzerdefinierte CFN Lint-Regeln.
3. Entpacken Sie die angehängten Regeln in ein Verzeichnis Ihrer Wahl.
4. Bestätigen Sie Ihre CloudFormation Vorlage, indem Sie den folgenden Befehl ausführen:

```
cfn-lint --template {TEMPLATE_FILE} --append-rules {DIRECTORY_WITH_CUSTOM_RULES}
```

CloudFormation Ingest-Stack: Beispiele für CFN-Validatoren

Diese Beispiele können Ihnen helfen, Ihre Vorlage für eine erfolgreiche Aufnahme vorzubereiten.

Validierung des Formats

Stellen Sie sicher, dass die Vorlage einen Abschnitt „Ressourcen“ enthält und dass alle darunter definierten Ressourcen den Wert „Typ“ haben.

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Description" : "Create a SNS topic",
  "Resources": {
    "SnsTopic": {
      "Type": "AWS::SNS::Topic"
    }
  }
}
```

Stellen Sie sicher, dass die Stammschlüssel der Vorlage zulässig sind. Zulässige Stammschlüssel sind:

```
[
  "AWSTemplateFormatVersion",
  "Description",
  "Mappings",
  "Parameters",
  "Conditions",
  "Resources",
  "Rules",
  "Outputs",
  "Metadata"
]
```

Eine manuelle Überprüfung erforderte eine Validierung

Wenn die Vorlage die folgenden Ressourcen enthält, schlägt die automatische Validierung fehl und Sie müssen manuell überprüft werden.

Die aufgeführten Richtlinien stellen aus Sicherheitsgründen Bereiche mit hohem Risiko dar.
Eine S3-Bucket-Richtlinie, die es jedem außer bestimmten Benutzern oder Gruppen ermöglicht,

Objekte zu erstellen oder Schreibberechtigungen zu erteilen, ist beispielsweise extrem gefährlich. Daher validieren wir die Richtlinien und genehmigen oder verweigern sie auf der Grundlage des Inhalts, und diese Richtlinien können nicht automatisch erstellt werden. Wir untersuchen mögliche Lösungsansätze für dieses Problem.

Derzeit verfügen wir nicht über eine automatisierte Validierung der folgenden Ressourcen.

```
[
    "S3::BucketPolicy",
    "SNS::TopicPolicy",
    "SQS::QueuePolicy"
]
```

Parametervalidierung

Überprüfen Sie, ob für einen Vorlagenparameter ein Standardwert angegeben wurde, wenn für ihn kein Wert angegeben wurde.

Validierung von Ressourcenattributen

Erforderliche Attributprüfung: Bestimmte Attribute müssen für bestimmte Ressourcentypen existieren.

- "VPCOptions" muss existieren in `AWS::OpenSearch::Domain`
- "CludsterSubnetGroupName" muss existieren in `AWS::Redshift::Cluster`

```
{
    "AWS::OpenSearch::Domain": [
      "VPCOptions"
    ],
    "AWS::Redshift::Cluster": [
      "ClusterSubnetGroupName"
    ]
}
```

Überprüfung unzulässiger Attribute: Bestimmte Attribute darfen*nicht* für bestimmte Ressourcentypen existieren.

- "SecretString" darf nicht existieren in "“ AWS::SecretsManager::Secret
- "MongoDbSettings" darf nicht in "AWS::DMS::Endpoint“ existieren

```
{
  "AWS::SecretsManager::Secret": [
    "SecretString"
  ],
  "AWS::DMS::Endpoint": [
    "MongoDbSettings"
  ]
}
```

SSM-Parameterprüfung: Für Attribute in der folgenden Liste müssen Werte über Secrets Manager oder Systems Manager Parameter Store (Secure String Parameter) angegeben werden:

```
{
  "RDS::DBInstance": [
    "MasterUserPassword",
    "TdeCredentialPassword"
  ],
  "RDS::DBCluster": [
    "MasterUserPassword"
  ],
  "ElastiCache::ReplicationGroup": [
    "AuthToken"
  ],
  "DMS::Certificate": [
    "CertificatePem",
    "CertificateWallet"
  ],
  "DMS::Endpoint": [
    "Password"
  ],
  "CodePipeline::Webhook": {
    "AuthenticationConfiguration": [
        "SecretToken"
    ]
  },
  "DocDB::DBCluster": [
    "MasterUserPassword"
  ]
},
```

Einige Attribute müssen bestimmten Mustern entsprechen. Beispielsweise dürfen Profilnamen von IAM-Instanzen nicht mit reservierten AMS-Präfixen beginnen, und der Attributwert muss der spezifischen Regex entsprechen, wie hier gezeigt:

```
{
    "AWS::EC2::Instance": {
      "IamInstanceProfile": [
        "^(?!arn:aws:iam|ams|Ams|AMS|AWSManagedServices|Managed_Services|mc|Mc|MC|
sentinel|Sentinel).+",
        "arn:aws:iam::(\\$\\{AWS::AccountId\\}|[0-9]+):instance-profile/(?!ams|Ams|AMS|
AWSManagedServices|Managed_Services|mc|Mc|MC|sentinel|Sentinel).+"
      ]
    },
    "AWS::AutoScaling::LaunchConfiguration": {
      "IamInstanceProfile": [
        "^(?!arn:aws:iam|ams|Ams|AMS|AWSManagedServices|Managed_Services|mc|Mc|MC|
sentinel|Sentinel).+",
        "arn:aws:iam::(\\$\\{AWS::AccountId\\}|[0-9]+):instance-profile/(?!ams|Ams|AMS|
AWSManagedServices|Managed_Services|mc|Mc|MC|sentinel|Sentinel).+"
      ]
    },
    "AWS::EC2::LaunchTemplate": {
      "LaunchTemplateData.IamInstanceProfile.Name": [
        "^(?!ams|Ams|AMS|AWSManagedServices|Managed_Services|mc|Mc|MC|sentinel|
Sentinel).+"
      ],
      "LaunchTemplateData.IamInstanceProfile.Arn": [
        "arn:aws:iam::(\\$\\{AWS::AccountId\\}|[0-9]+):instance-profile\/(?!ams|Ams|
AMS|AWSManagedServices|Managed_Services|mc|Mc|MC|sentinel|Sentinel).+"
      ]
    }
}
```

Validierung von Ressourcen

In der Vorlage können nur Ressourcen angegeben werden, die auf der Zulassungsliste stehen. Diese Ressourcen werden unter beschrieben. Unterstützte Ressourcen

EC2 Stacks und Auto Scaling Scaling-Gruppen (ASGs) sind aufgrund von Patching-Einschränkungen nicht im selben Stack zulässig.

Überprüfung der Regeln für eingehenden Zugriff auf Sicherheitsgruppen

- Für Anfragen, die aus den CT-Änderungstypen CFN Ingest Create oder Stack Update CT stammen:
  - Wenn (`IpProtocol`ist tcp oder 6) UND (Port ist 80 oder 443), gibt es keine Einschränkungen in Bezug auf den Wert `CidrIP`
  - Andernfalls kann er `CidrIP` nicht 0.0.0.0/0 sein
- Für Anfragen aus Service Catalog (Service Catalog-Produkte):
  - Zusätzlich zur Überprüfung des CT-Änderungstyps CFN Ingest Create oder Stack Update CT `ip_protocols` kann auf den Port `management_ports` mit dem eingegebenen Protokoll nur zugegriffen werden über: `allowed_cidrs`

```
{
      "ip_protocols": ["tcp", "6", "udp", "17"],
      "management_ports": [22, 23, 389, 636, 1494, 1604, 2222, 3389, 5900, 5901,
 5985, 5986],
      "allowed_cidrs": ["10.0.0.0/8", "100.64.0.0/10", "172.16.0.0/12",
 "192.168.0.0/16"]
  }
```

Einschränkungen

Die folgenden Merkmale und Funktionen werden derzeit vom AWS CloudFormation AMS-Ingest-Prozess nicht unterstützt.

- YAML — Nicht unterstützt. Es werden nur JSON-basierte CloudFormation Vorlagen unterstützt.
- Verschachtelte Stacks — Richten Sie Ihre Anwendungsinfrastruktur stattdessen so ein, dass sie eine einzige Vorlage verwendet. Alternativ können Sie stapelübergreifende Referenzierung verwenden, um Ressourcen auf mehrere Stapel zu verteilen, wobei eine Ressource von einer anderen abhängig ist. Weitere Informationen finden Sie unter Exemplarische Vorgehensweise: Siehe Ressourcenausgaben in einem anderen CloudFormation AWS-Stack.
- CloudFormation Stack-Sets — Aus Sicherheitsgründen nicht unterstützt.
- Erstellung von IAM-Ressourcen mithilfe von CloudFormation Vorlagen — Aus Sicherheitsgründen werden nur IAM-Rollen unterstützt.
- Sensible Daten — Nicht unterstützt. Nehmen Sie keine sensiblen Daten in die Vorlage oder in die Parameterwerte auf. Wenn Sie auf sensible Daten verweisen müssen, verwenden Sie Secrets

Manager, um diese Werte zu speichern und abzurufen. Informationen zur Verwendung von AWS Secrets Manager-Geheimnissen in einer Ressourceneigenschaft finden Sie unter Erstellen und Abrufen von in AWS Secrets Manager verwalteten Geheimnissen mithilfe von CloudFormation AWS-Vorlagen und Verwenden dynamischer Referenzen zur Angabe von Vorlagenwerten.

Unterstützte Ressourcen

Die folgenden AWS-Ressourcen werden im AWS CloudFormation AMS-Ingest-Prozess unterstützt.

CloudFormation Ingest Stack: Unterstützte Ressourcen

Das Instance-Betriebssystem muss von AMS Workload Ingestion unterstützt werden. Nur die hier aufgeführten AWS-Ressourcen werden unterstützt.

- Amazon API Gateway
  - AWS::ApiGateway::Account
  - AWS::ApiGateway::ApiKey
  - AWS::ApiGateway::Authorizer
  - AWS::ApiGateway::BasePathKartierung
  - AWS::ApiGateway::ClientCertificate
  - AWS::ApiGateway::Deployment
  - AWS::ApiGateway::DocumentationPart
  - AWS::ApiGateway::DocumentationVersion
  - AWS::ApiGateway::DomainName
  - AWS::ApiGateway::GatewayResponse
  - AWS::ApiGateway::Method
  - AWS::ApiGateway::Model
  - AWS::ApiGateway::RequestValidator
  - AWS::ApiGateway::Resource
  - AWS::ApiGateway::RestApi
  - AWS::ApiGateway::Stage
  - AWS::ApiGateway::UsagePlan
  - AWS::ApiGateway::UsagePlanSchlüssel
  - AWS::ApiGateway::VpcLink

- Amazon API Gateway V2
  - AWS::ApiGatewayV2::Api
  - AWS::ApiGatewayV2::ApiGatewayManagedOverrides
  - AWS::ApiGatewayV2::ApiMapping
  - AWS::ApiGatewayV2::Authorizer
  - AWS::ApiGatewayV2::Deployment
  - AWS::ApiGatewayV2::DomainName
  - AWS::ApiGatewayV2::Integration
  - AWS::ApiGatewayV2::IntegrationResponse
  - AWS::ApiGatewayV2::Model
  - AWS::ApiGatewayV2::Route
  - AWS::ApiGatewayV2::RouteResponse
  - AWS::ApiGatewayV2::Stage
  - AWS::ApiGatewayV2::VpcLink
- AWS AppSync
  - AWS::AppSync::ApiCache
  - AWS::AppSync::ApiKey
  - AWS::AppSync::DataSource
  - AWS::AppSync::FunctionConfiguration
  - AWS::AppSync::GraphQLApi
  - AWS::AppSync::GraphQLSchema
  - AWS::AppSync::Resolver
- Amazon Athena
  - AWS::Athena::NamedQuery
  - AWS::Athena::WorkGroup
- AWS Backup
  - AWS::Backup::BackupVault
- Amazon CloudFront
  - AWS::CloudFront::Distribution
  - AWS::CloudFront::CloudFrontOriginAccessIdentity

  - AWS::CloudFront::StreamingDistribution
- Amazon CloudWatch
  - AWS::CloudWatch::Alarm
  - AWS::CloudWatch::AnomalyDetector
  - AWS::CloudWatch::CompositeAlarm
  - AWS::CloudWatch::Dashboard
  - AWS::CloudWatch::InsightRule
- CloudWatch Amazon-Protokolle
  - AWS::Logs::LogGroup
  - AWS::Logs::LogStream
  - AWS::Logs::MetricFilter
  - AWS::Logs::SubscriptionFilter
- Amazon Cognito
  - AWS::Cognito::IdentityPool
  - AWS::Cognito::IdentityPoolRoleAttachment
  - AWS::Cognito::UserPool
  - AWS::Cognito::UserPool-Client
  - AWS::Cognito::UserPoolDomäne
  - AWS::Cognito::UserPoolGruppe
  - AWS::Cognito::UserPoolIdentityProvider
  - AWS::Cognito::UserPoolResourceServer
  - AWS::Cognito::UserPoolRiskConfigurationAttachment
  - AWS::Cognito::UserPoolUICustomizationAnlage
  - AWS::Cognito::UserPoolBenutzer
  - AWS::Cognito::UserPoolUserToGroupAttachment
- Amazon DocumentDB
  - AWS::DocDB: DBCluster
  - AWS::DocDB:: DBCluster ParameterGroup
  - AWS::DocDB:: DBInstance
  - AWS::DocDB: DBSubnet Gruppe

- Amazon-DynamoDB
  - AWS::DynamoDB::Table
- Amazon EC2
  - AWS::EC2::Volume
  - AWS::EC2::VolumeAttachment
  - AWS::EC2::Instance
  - AWS::EC2: EIP
  - AWS:EC2:: EIPAssociation
  - AWS::EC2::NetworkInterface
  - AWS::EC2::NetworkInterfaceAnlage
  - AWS::EC2::SecurityGroup
  - AWS::EC2::SecurityGroupEintritt
  - AWS::EC2::SecurityGroupAustritt
  - AWS::EC2::LaunchTemplate
- AWS Batch
  - AWS::Batch::ComputeEnvironment
  - AWS::Batch::JobDefinition
  - AWS::Batch::JobQueue
- Amazon Elastic Container Registry (ECR)
  - AWS::ECR::Repository
- Amazon Elastic Container Service (ECS) (Fargate)
  - AWS::ECS::CapacityProvider
  - AWS::ECS::Cluster
  - AWS::ECS::PrimaryTaskSatz
  - AWS::ECS::Service
  - AWS::ECS::TaskDefinition
  - AWS::ECS::TaskSet
- Amazon Elastic File System (EFS)
  - AWS::EFS::FileSystem

  - AWS::EFS::MountTarget

- Amazon ElastiCache
  - AWS::ElastiCache::CacheCluster
  - AWS::ElastiCache::ParameterGroup
  - AWS::ElastiCache::ReplicationGroup
  - AWS::ElastiCache::SecurityGroup
  - AWS::ElastiCache::SecurityGroupZutritt
  - AWS::ElastiCache::SubnetGroup
- Amazon EventBridge
  - AWS::Events::EventBus
  - AWS::Events::EventBusPolitik
  - AWS::Events::Rule
- Amazon FSx
  - AWS::FSx::FileSystem
- Amazon Inspector
  - AWS::Inspector::AssessmentTarget
  - AWS::Inspector::AssessmentTemplate
  - AWS::Inspector::ResourceGroup
- Amazon Kinesis Data Analytics
  - AWS::KinesisAnalytics::Application
  - AWS::KinesisAnalytics::ApplicationOutput
  - AWS::KinesisAnalytics::ApplicationReferenceDataSource
- Amazon Kinesis Data Firehose
  - AWS::KinesisFirehose::DeliveryStream
- Amazon Kinesis Data Streams
  - AWS::Kinesis::Stream
  - AWS::Kinesis::StreamConsumer
- Amazon MQ
  - AWS::AmazonMQ::Broker
  - AWS::AmazonMQ::Configuration

  - AWS::AmazonMQ::ConfigurationAssociation

- Amazon OpenSearch
  - AWS::OpenSearchService::Domain
- Amazon Relational Database Service (RDS)
  - AWS: :RDS:: DBCluster
  - AWS: :RDS:: DBCluster ParameterGroup
  - AWS: :RDS:: DBInstance
  - AWS: :RDS:: Gruppe DBParameter
  - AWS: :RDS:: Gruppe DBSubnet
  - AWS::RDS::EventSubscription
  - AWS::RDS::OptionGroup
- Amazon Route 53
  - AWS::Route53::HealthCheck
  - AWS::Route53::HostedZone
  - AWS::Route53::RecordSet
  - AWS::Route53::RecordSetGruppe
  - AWS::Route53Resolver::ResolverRule
  - AWS::Route53Resolver::ResolverRuleVerband
- Amazon S3
  - AWS::S3::Bucket
- Amazon Sagemaker
  - AWS::SageMaker::CodeRepository
  - AWS::SageMaker::Endpoint
  - AWS::SageMaker::EndpointConfig
  - AWS::SageMaker::Model
  - AWS::SageMaker::NotebookInstance
  - AWS::SageMaker::NotebookInstanceLifecycleConfig
  - AWS::SageMaker::Workteam
- Amazon Simple Email Service (SES)
  - AWS::SES::ConfigurationSet
  - AWS::SES::ConfigurationSetEventDestination

  - AWS::SES::ReceiptFilter
  - AWS::SES::ReceiptRule
  - AWS::SES::ReceiptRuleEinstellen
  - AWS::SES::Template
- Amazon SimpleDB
  - AWS::SDB::Domain
- Amazon SNS
  - AWS::SNS::Subscription
  - AWS::SNS::Topic
- Amazon SQS
  - AWS::SQS::Queue
- Amazon WorkSpaces
  - AWS::WorkSpaces::Workspace
- Anwendung AutoScaling
  - AWS::ApplicationAutoScaling::ScalableTarget
  - AWS::ApplicationAutoScaling::ScalingPolicy
- Amazon EC2 AutoScaling
  - AWS::AutoScaling::AutoScalingGruppe
  - AWS::AutoScaling::LaunchConfiguration
  - AWS::AutoScaling::LifecycleHook
  - AWS::AutoScaling::ScalingPolicy
  - AWS::AutoScaling::ScheduledAction
- AWS Certificate Manager
  - AWS::CertificateManager::Certificate
- AWS CloudFormation
  - AWS::CloudFormation::CustomResource
  - AWS::CloudFormation::Designer
  - AWS::CloudFormation::WaitCondition
  - AWS::CloudFormation::WaitConditionGriff
- AWS CodeBuild

  - AWS::CodeBuild::Project
  - AWS::CodeBuild::ReportGroup
  - AWS::CodeBuild::SourceCredential
- AWS CodeCommit
  - AWS::CodeCommit::Repository
- AWS CodeDeploy
  - AWS::CodeDeploy::Application
  - AWS::CodeDeploy::DeploymentConfig
  - AWS::CodeDeploy::DeploymentGroup
- AWS CodePipeline
  - AWS::CodePipeline::CustomActionTyp
  - AWS::CodePipeline::Pipeline
  - AWS::CodePipeline::Webhook
- AWS Database Migration Service (DMS)
  - AWS::DMS::Certificate
  - AWS::DMS::Endpoint
  - AWS::DMS::EventSubscription
  - AWS::DMS::ReplicationInstance
  - AWS::DMS::ReplicationSubnetGruppe
  - AWS::DMS::ReplicationTask

  Die MongoDbSettings Eigenschaft in der AWS::DMS::Endpoint Ressource ist nicht zulässig.

  Die folgenden Eigenschaften sind nur zulässig, wenn sie von AWS Secrets Manager aufgelöst werden: CertificatePem und CertificateWallet Eigenschaften in der AWS::DMS::Certificate Ressource und die Eigenschaft Password in der AWS::DMS::Endpoint Ressource.
- AWS Elastic Load Balancing — Anwendungs-Load-Balancer//Network Load Balancer
  - AWS::ElasticLoadBalancingV2::Listener
  - AWS::ElasticLoadBalancingV2::ListenerCertificate
  - AWS::ElasticLoadBalancingV2::ListenerRule
  - AWS::ElasticLoadBalancingV2::LoadBalancer
  - AWS::ElasticLoadBalancingV2::TargetGroup

- AWS Elastic Load Balancing — Classic Load Balancer
  - AWS::ElasticLoadBalancing::LoadBalancer
- AWS Elemental MediaConvert
  - AWS::MediaConvert::JobTemplate
  - AWS::MediaConvert::Preset
  - AWS::MediaConvert::Queue
- AWS Elemental MediaStore
  - AWS::MediaStore::Container
- AWS Identity and Access Management (ICH BIN)
  - AWS::IAM::Role
- AWS-verwaltetes Managed Streaming for Apache Kafka (MSK)
  - AWS::MSK::Cluster
- AWS Glue
  - AWS::Glue::Classifier
  - AWS::Glue::Connection
  - AWS::Glue::Crawler
  - AWS::Glue::Database
  - AWS::Glue::DataCatalogEncryptionSettings
  - AWS::Glue::DevEndpoint
  - AWS::Glue::Job
  - AWS::Glue::MLTransform
  - AWS::Glue::Partition
  - AWS::Glue::SecurityConfiguration
  - AWS::Glue::Table
  - AWS::Glue::Trigger
  - AWS::Glue::Workflow
- AWS Key Management Service (KMS)
  - AWS::KMS::Key
  - AWS::KMS::Alias

- AWS Lake Formation

  - AWS::LakeFormation::DataLakeEinstellungen
  - AWS::LakeFormation::Permissions
  - AWS::LakeFormation::Resource
- AWS Lambda
  - AWS::Lambda::Alias
  - AWS::Lambda::EventInvokeConfig
  - AWS::Lambda::EventSourceKartografie
  - AWS::Lambda::Function
  - AWS::Lambda::LayerVersion
  - AWS::Lambda::LayerVersionErlaubnis
  - AWS::Lambda::Permission
  - AWS::Lambda::Version
- Amazon Redshift
  - AWS::Redshift::Cluster
  - AWS::Redshift::ClusterParameterGruppe
  - AWS::Redshift::ClusterSubnetGruppe
- AWS Secrets Manager
  - AWS::SecretsManager::ResourcePolicy
  - AWS::SecretsManager::RotationSchedule
  - AWS::SecretsManager::Secret
  - AWS::SecretsManager::SecretTargetAnlage
- AWS Security Hub
  - AWS::SecurityHub::Hub
- AWS Step Functions
  - AWS::StepFunctions::Activity
  - AWS::StepFunctions::StateMachine
- AWS Systems Manager (SSM)
  - AWS::SSM::Parameter
- Amazon CloudWatch Synthetics
  - AWS::Synthetics::Canary

- AWS Transfer Family
  - AWS::Transfer::Server
  - AWS::Transfer::User
- AWS WAF
  - AWS::WAF::ByteMatchSet
  - AWS: :WAF: IPSet
  - AWS::WAF::Rule
  - AWS::WAF::SizeConstraintSatz
  - AWS::WAF::SqlInjectionMatchSet
  - AWS::WAF::WebACL
  - AWS::WAF::XssMatchSatz
- AWS WAF Regional
  - AWS::WAFRegional::ByteMatchSatz
  - AWS::WAFRegional::GeoMatchSatz
  - AWS:WAFRegional:: IPSet
  - AWS::WAFRegional::RateBasedRegel
  - AWS::WAFRegional::RegexPatternSatz
  - AWS::WAFRegional::Rule
  - AWS::WAFRegional::SizeConstraintSatz
  - AWS::WAFRegional::SqlInjectionMatchSet
  - AWS::WAFRegional::WebACL
  - AWS::WAFRegional::WebACLAssociation
  - AWS::WAFRegional::XssMatchSatz
- AWS WAFv2
  - AWS:WAFv2:: IPSet
  - AWS::WAFv2::RegexPatternSatz
  - AWS::WAFv2::RuleGroup
  - AWS::WAFv2::WebACL
  - AWS::WAFv2::WebACLAssociation

AWS CloudFormation Ingest: Beispiele

Hier finden Sie einige detaillierte Beispiele für die Verwendung des Change-Typs Create Stack with CloudFormation Template.

Eine Reihe von CloudFormation Beispielvorlagen pro können Sie AWS-Region unter Beispielvorlagen herunterladen.

Referenzinformationen zu AWS CloudFormation Ressourcen finden Sie unter AWS-Referenz zu Ressourcen- und Eigenschaftstypen. AMS unterstützt jedoch einen kleineren Satz von Ressourcen, die unter beschrieben werden CloudFormation AMS-Aufnahme.

> **Note**
>
> AMS empfiehlt Ihnen, alle IAM- oder anderen richtlinienbezogenen Ressourcen zu sammeln und sie in einem einzigen Bereich unter Verwaltung | Andere | Andere | Änderungstyp erstellen (ct-1e1xtak34nx76) einzureichen. Kombinieren Sie beispielsweise alle benötigten IAM-Rollen, IAM-Instanzprofile, IAM-Richtlinienaktualisierungen für bestehende IAM-Rollen, S3-Bucket-Richtlinien, SNS/SQS Richtlinien usw. und reichen Sie dann einen ct-1e1xtak34nx76-RFC ein, damit diese bereits vorhandenen Ressourcen in den future CFN-Ingest-Templates referenziert werden können.

Themen

AWS CloudFormation Ingest-Beispiele: Ressourcen definieren

Wenn Sie AWS CloudFormation AMS-Ingest verwenden, passen Sie eine CloudFormation Vorlage an und senden sie in einem RFC mit dem CloudFormation Ingest-Änderungstyp (ct-36cn2avfrrj9v) an AMS. Um eine CloudFormation Vorlage zu erstellen, die mehrfach wiederverwendet werden kann, fügen Sie die Stack-Konfigurationsparameter der Eingabe für die Ausführung des CloudFormation Ingest-Änderungstyps hinzu, anstatt sie in der Vorlage fest zu codieren. CloudFormation Der größte Vorteil besteht darin, dass Sie die Vorlage wiederverwenden können.

Mit dem Eingabeschema für den CloudFormation AMS-Ingest-Änderungstyp können Sie bis zu sechzig Parameter in einer CloudFormation Vorlage auswählen und benutzerdefinierte Werte angeben.

Dieses Beispiel zeigt, wie eine Ressourceneigenschaft, die in einer Vielzahl von CloudFormation Vorlagen verwendet werden kann, als Parameter im CloudFormation AMS-Ingest-CT definiert wird. Die Beispiele in diesem Abschnitt zeigen speziell die Verwendung von SNS-Themen.

Themen

Beispiel 1: Hardcodieren Sie die AWS CloudFormation SNSTopic Ressourceneigenschaft **`TopicName`**

In diesem Beispiel codieren Sie die AWS CloudFormation SNSTopic `TopicName` Ressourceneigenschaft in der CloudFormation Vorlage fest. Beachten Sie, dass der `Parameters` Abschnitt leer ist.

Um über eine CloudFormation Vorlage zu verfügen, mit der Sie den Wert für den SNSTopic Namen eines neuen Stacks ändern können, ohne eine neue CloudFormation Vorlage erstellen zu müssen, können Sie den `Parameters` AMS-Abschnitt des Änderungstyps CloudFormation Ingest verwenden, um diese Konfiguration vorzunehmen. Auf diese Weise verwenden Sie später dieselbe CloudFormation Vorlage, um einen neuen Stack mit einem anderen `SNSTopic` Namen zu erstellen.

```
{
  "AWSTemplateFormatVersion" : "2010-09-09",
  "Description" : "My SNS Topic",
  "Parameters" : {
  },
  "Resources" : {
```

```
    "SNSTopic" : {
      "Type" : "AWS::SNS::Topic",
      "Properties" : {
        "TopicName" : "MyTopicName"
      }
    }
  }
}
```

Beispiel 2: Verwenden Sie eine SNSTopic Ressource, um auf einen Parameter im AMS-Änderungstyp zu verweisen

In diesem Beispiel verwenden Sie eine in der CloudFormation Vorlage definierte `SNSTopic` `TopicName` Ressourceneigenschaft, um auf a `Parameter` im AMS-Änderungstyp zu verweisen.

```
{
  "AWSTemplateFormatVersion" : "2010-09-09",
  "Description" : "My SNS Topic",
  "Parameters" : {
    "TopicName" : {
      "Type" : "String",
      "Description" : "Topic ID",
      "Default" : "MyTopicName"
    }
  },
  "Resources" : {
    "SNSTopic" : {
      "Type" : "AWS::SNS::Topic",
      "Properties" : {
        "TopicName" : { "Ref" : "TopicName"}
      }
    }
  }
}
```

Beispiel 3: Erstellen Sie ein SNS-Thema, indem Sie eine JSON-Ausführungsparameterdatei mit dem Änderungstyp AMS-Ingest einreichen

In diesem Beispiel reichen Sie eine JSON-Ausführungsparameterdatei mit dem AMS-Ingest-CT ein, mit dem das SNS-Thema erstellt wird. `TopicName` Das SNS-Thema muss in der CloudFormation Vorlage auf die in diesem Beispiel gezeigte änderbare Weise definiert werden.

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_PRESIGNED_URL",
  "VpcId": "VPC_ID",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
  "Parameters": [
    {"Name": "TopicName", "Value": "MyTopic1"}
  ],
  "TimeoutInMinutes": 60
}
```

Beispiel 4: Reichen Sie einen neuen Änderungstyp ein, der auf dieselbe Vorlage verweist CloudFormation

In diesem JSON-Beispiel wird der `TopicName` SNS-Wert geändert, ohne die CloudFormation Vorlage zu ändern. Stattdessen reichen Sie einen neuen Änderungstyp Deployment | Ingestion | Stack from CloudFormation Template | Create Change Type ein, der auf dieselbe CFN-Vorlage verweist.

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_PRESIGNED_URL",
  "VpcId": "VPC_ID",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
  "Parameters": [
    {"Name": "TopicName", "Value": "MyTopic2"}
  ],
  "TimeoutInMinutes": 60
}
```

Beispiel 5: Verwenden Sie die Standardparameterwerte in der Vorlage CloudFormation

In diesem Beispiel wird das SNS `TopicName` = 'MyTopicName' erstellt, weil im `Parameters` Ausführungsparameter kein `TopicName` Wert angegeben wurde. Wenn Sie keine `Parameters`

Definitionen angeben, werden die Standardparameterwerte in der CloudFormation Vorlage verwendet.

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_PRESIGNED_URL",
  "VpcId": "VPC_ID",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
  "TimeoutInMinutes": 60
}
```

CloudFormation Ingest-Beispiele: 3-stufige Webanwendung

Ingestieren Sie eine CloudFormation Vorlage für eine standardmäßige 3-Tier-Webanwendung.

Dazu gehören ein Application Load Balancer, eine Application Load Balancer Balancer-Zielgruppe, eine Auto Scaling Scaling-Gruppe, eine Startvorlage für Auto Scaling Scaling-Gruppen, Amazon Relational Database Service (RDS für SQL Server) mit einer MySQL-Datenbank, einem AWS SSM-Parameterspeicher und Secrets Manager. AWS Nehmen Sie sich 30-60 Minuten Zeit, um dieses Beispiel durchzugehen.

Voraussetzungen

- Erstellen Sie mit dem AWS Secrets Manager ein Geheimnis, das einen Benutzernamen und ein Passwort mit entsprechenden Werten enthält. Sie können auf diese JSON-Beispielvorlage (ZIP-Datei) verweisen, die den geheimen Namen enthält`ams-shared/myapp/dev/dbsecrets`, und sie durch Ihren geheimen Namen ersetzen. Hinweise zur Verwendung von AWS Secrets Manager mit AMS finden Sie unterAWS Secrets Manager mit AMS-Ressourcen verwenden.
- Richten Sie die erforderlichen Parameter im AWS SSM Parameter Store (PS) ein. In diesem Beispiel werden das `VPCId` und `Subnet-Id` der privaten und öffentlichen Subnetze im SSM PS in Pfaden wie`/app/DemoApp/PublicSubnet1a`,`PublicSubnet1c`, `PrivateSubnet1a` und gespeichert. `PrivateSubnet1c` `VPCCidr` Aktualisieren Sie die Pfade sowie die Parameternamen und Werte entsprechend Ihren Anforderungen.
- Erstellen Sie eine EC2 IAM-Amazon-Instance-Rolle mit Leseberechtigungen für die Pfade AWS Secrets Manager und SSM Parameter Store (die in diesen Beispielen erstellte und

verwendete IAM-Rolle ist). `customer-ec2_secrets_manager_instance_profile` Wenn Sie IAM-Standardrichtlinien wie die Instance-Profilrolle erstellen, muss der Rollenname mit beginnen. `customer-` Um eine neue IAM-Rolle zu erstellen (Sie können ihr einen Namen oder einen anderen Namen geben)`customer-ec2_secrets_manager_instance_profile`, verwenden Sie den AMS-Änderungstyp Management | Applications | IAM-Instanzprofil | Create (ct-0ixp4ch2tiu04) CT und fügen Sie die erforderlichen Richtlinien hinzu. Sie können die AMS IAM-Standardrichtlinien überprüfen `customer_secrets_manager_policy` und `customer_systemsmanager_parameterstore_policy` in der IAM-Konsole überprüfen, ob sie unverändert oder als Referenz verwendet werden können. AWS

Investieren Sie eine CloudFormation Vorlage für eine standardmäßige 3-Tier-Webanwendung

1. Laden Sie die angehängte CloudFormation JSON-Beispielvorlage als ZIP-Datei (tier-cfn-ingest3.zip) in einen S3-Bucket hoch und generieren Sie eine signierte S3-URL zur Verwendung im CFN Ingest RFC. Weitere Informationen finden Sie unter Presign. Die CFN-Vorlage kann auch im copy/pasted CFN-Ingest-RFC enthalten sein, wenn Sie den RFC über die AMS-Konsole einreichen.
2. Erstellen Sie einen CloudFormation Ingest-RFC (Deployment | Ingestion | Stack from CloudFormation template | Create (ct-36cn2avfrrj9v)), entweder über die AMS-Konsole oder die AMS-CLI. Bei der Automatisierung der CloudFormation Datenaufnahme wird die Vorlage validiert, um sicherzustellen, dass die CloudFormation Vorlage über gültige AMS-unterstützte Ressourcen verfügt und den Sicherheitsstandards entspricht.

   - Mithilfe der Konsole — Wählen Sie für den Änderungstyp Deployment -> Ingestion -> Stack from CloudFormation Template -> Create aus und fügen Sie dann als Beispiel die folgenden Parameter hinzu (beachten Sie, dass die Standardeinstellung für Multi falsch ist): AZDatabase

     ```
     CloudFormationTemplateS3Endpoint: "https://s3-ap-
     southeast-2.amazonaws.com/amzn-s3-demo-bucket/3-tier-cfn-ingest.json?
     AWSAccessKeyId=#{S3_ACCESS_KEY_ID}&Expires=#{EXPIRE_DATE}&Signature=#{SIGNATURE}"
     VpcId: "VPC_ID"
     TimeoutInMinutes: 120
     IAMEC2InstanceProfile: "customer_ec2_secrets_manager_instance_profile"
     MultiAZDatabase: "true"
     WebServerCapacity: "2"
     ```

   - Verwenden von AWS CLI — Weitere Informationen zum Erstellen RFCs mit dem finden Sie AWS CLI unter Erstellen. RFCs Führen Sie z. B. den folgenden Befehl aus:

```
aws --profile=saml amscm create-rfc  --change-type-id ct-36cn2avfrrj9v
 --change-type-version "2.0" --title "TEST_CFN_INGEST" --execution-
parameters "{\"CloudFormationTemplateS3Endpoint\":\"https://s3-
ap-southeast-2.amazonaws.com/my-bucket/3-tier-cfn-ingest.json?
AWSAccessKeyId=#{S3_ACCESS_KEY_ID}&Expires=#{EXPIRE_DATE}&Signature=#{SIGNATURE}\",
\"TimeoutInMinutes\":120,\"Description\":\"TEST\",\”VpcId”\”:\”VPC_ID\”,
\"Name\":\"MY_TEST\",\"Tags\":[{\"Key\":\"env\",\"Value\":\"test\"}],
\"Parameters\":[{\"Name\":\"IAMEC2InstanceProfile\",\"Value\":
\"customer_ec2_secrets_manager_instance_profile\"},{\"Name\":\"MultiAZDatabase\",
\"Value\":\"true\"},{\"Name\":\"VpcId\",\"Value\":\"VPC_ID\"},{\"Name\":
\"WebServerCapacity\",\"Value\":\"2\"}]}" --endpoint-url https://amscm.us-
east-1.amazonaws.com/operational/ --no-verify-ssl
```

Suchen Sie die Application Load Balancer Balancer-URL in der AWS CloudFormation RFC-Ausführungsausgabe, um auf die Website zuzugreifen. Informationen zum Zugriff auf Ressourcen finden Sie unter Zugreifen auf Instanzen.

Erstellen Sie einen Ingest-Stack CloudFormation

Einen Ingest-Stack mithilfe CloudFormation der Konsole erstellen

Um einen CloudFormation Ingest-Stack mit der Konsole zu erstellen

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf Create RFC.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.

- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Erstellen eines CloudFormation Ingest-Stacks mit der CLI

So erstellen Sie einen CloudFormation Ingest-Stack mit der CLI

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

1. Bereiten Sie die CloudFormation Vorlage vor, mit der Sie den Stack erstellen werden, und laden Sie sie in Ihren S3-Bucket hoch. Wichtige Informationen finden Sie unter AWS CloudFormation Ingest Guidelines, Best Practices and Limitations.
2. Erstellen Sie den RFC und reichen Sie ihn an AMS ein:

   - Erstellen und speichern Sie die JSON-Datei mit den Ausführungsparametern, einschließlich der gewünschten CloudFormation Vorlagenparameter. Im folgenden Beispiel wird sie „ CreateCfnParams.json“ genannt.

     Beispiel für eine CreateCfnParams .json-Datei für den Webanwendungsstapel:

     ```
     {
       "Name": "cfn-ingest",
       "Description": "CFNIngest Web Application Stack",
       "VpcId": "VPC_ID",
       "CloudFormationTemplateS3Endpoint": "$S3_URL",
       "TimeoutInMinutes": 120,
       "Tags": [
        {
         "Key":   "Enviroment Type"
         "Value": "Dev",
        },
        {
         "Key":   "Application"
         "Value": "PCS",
     ```

```
    }
  ],
  "Parameters": [
   {
    "Name": "Parameter-for-S3Bucket-Name",
    "Value":  "BUCKET-NAME"
   },
   {
    "Name": "Parameter-for-Image-Id",
    "Value":  "AMI-ID"
   }
  ],
}
```

Beispiel für eine CreateCfnParams .json-Datei mit einem SNS-Thema:

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_URL",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
  "Parameters": [
    {"Name": "TopicName", "Value": "MyTopic1"}
  ]
}
```

3. Erstellen und speichern Sie die JSON-Datei mit den RFC-Parametern mit dem folgenden Inhalt. Im folgenden Beispiel wird sie als CreateCfnRfc JSON-Datei bezeichnet:

```
{
   "ChangeTypeId": "ct-36cn2avfrrj9v",
   "ChangeTypeVersion": "2.0",
   "Title": "cfn-ingest"
}
```

4. Erstellen Sie den RFC und geben Sie die CreateCfnRfc Datei und die CreateCfnParams Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateCfnRfc.json  --execution-
parameters file://CreateCfnParams.json
```

In der Antwort erhalten Sie die ID des neuen RFC und können damit den RFC einreichen und überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

> **Note**
>
> Dieser Änderungstyp hat Version 2.0 und ist automatisiert (nicht manuell ausgeführt). Dadurch kann die CT-Ausführung schneller durchgeführt werden, und ein neuer Parameter ermöglicht es Ihnen CloudFormationTemplate, eine benutzerdefinierte CloudFormation Vorlage in den RFC einzufügen. Außerdem fügen wir in dieser Version die standardmäßigen AMS-Sicherheitsgruppen nicht hinzu, wenn Sie Ihre eigenen Sicherheitsgruppen angeben. Wenn Sie in der Anfrage keine eigenen Sicherheitsgruppen angeben, hängt AMS die AMS-Standardsicherheitsgruppen an. In CFN Ingest v1.0 haben wir immer die AMS-Standardsicherheitsgruppen angehängt, unabhängig davon, ob Sie Ihre eigenen Sicherheitsgruppen angegeben haben oder nicht.
> AMS hat 17 AMS Self-Provisioned Services für diesen Änderungstyp aktiviert. Informationen zu unterstützten Ressourcen finden Sie unter CloudFormation Ingest Stack: Unterstützte Ressourcen.

> **Note**
>
> Version 2.0 akzeptiert einen S3-Endpunkt, bei dem es sich nicht um eine vorsignierte URL handelt.
> Wenn Sie die vorherige Version dieses CT verwenden, muss der CloudFormationTemplateS3Endpoint-Parameterwert eine vorsignierte URL sein.
> Beispielbefehl zum Generieren einer vorsignierten S3-Bucket-URL (Mac/Linux):
>
> ```
> export S3_PRESIGNED_URL=$(aws s3 presign DASHDASHexpires-in 86400
>  s3://BUCKET_NAME/CFN_TEMPLATE.json)
> ```
>
> Beispielbefehl zum Generieren einer vorsignierten S3-Bucket-URL (Windows):

```
for /f %i in ('aws s3 presign DASHDASHexpires-in 86400
 s3://BUCKET_NAME/CFN_TEMPLATE.json') do set S3_PRESIGNED_URL=%i
```

Siehe auch Erstellen von vorsignierten URLs Buckets für Amazon S3.

> **Note**
>
> Wenn der S3-Bucket in einem AMS-Konto vorhanden ist, müssen Sie Ihre AMS-Anmeldeinformationen für diesen Befehl verwenden. Beispielsweise müssen Sie möglicherweise `--profile saml` nach Erhalt Ihrer AMS AWS Security Token Service (AWS STS) -Anmeldeinformationen eine Datei anhängen.

Verwandte Änderungstypen: CloudFormation Genehmigen Sie einen Changeset für den Ingest-Stack, AWS CloudFormation Aktualisieren Sie den Ingest-Stack

Weitere Informationen zu AWS finden Sie CloudFormation unter AWS CloudFormation. Um CloudFormation Vorlagen zu sehen, öffnen Sie die CloudFormation AWS-Vorlagenreferenz.

Ein Ingest wird validiert AWS CloudFormation

Die Vorlage wird validiert, um sicherzustellen, dass sie in einem AMS-Konto erstellt werden kann. Wenn sie die Validierung besteht, wird sie aktualisiert und enthält nun alle Ressourcen oder Konfigurationen, die für die AMS-Konformität erforderlich sind. Dazu gehört das Hinzufügen von Ressourcen wie CloudWatch Amazon-Alarmen, damit AMS Operations den Stack überwachen kann.

Der RFC wird abgelehnt, wenn eine der folgenden Bedingungen zutrifft:

- Die RFC-JSON-Syntax ist falsch oder folgt nicht dem angegebenen Format.
- Die angegebene vorsignierte URL für den S3-Bucket ist nicht gültig.
- Die Vorlage hat keine gültige AWS CloudFormation Syntax.
- Für die Vorlage sind keine Standardwerte für alle Parameterwerte festgelegt.
- Die Vorlage schlägt die AMS-Validierung fehl. Die Schritte zur AMS-Validierung finden Sie weiter unten in diesem Thema.

Der RFC schlägt fehl, wenn der CloudFormation Stack aufgrund eines Problems bei der Ressourcenerstellung nicht erstellt werden kann.

Weitere Informationen zur CFN-Validierung und -Validierung finden Sie unter Vorlagenvalidierung und CloudFormation Ingest-Stack: CFN-Validator-Beispiele.

AWS CloudFormation Aktualisieren Sie den Ingest-Stack

Aktualisierung eines CloudFormation Ingest-Stacks mithilfe der Konsole

Um einen CloudFormation Ingest-Stack mithilfe der Konsole zu aktualisieren

1. Navigieren Sie zur Seite „RFC erstellen“: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf Create RFC.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.
3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Aktualisierung eines CloudFormation Ingest-Stacks mit der CLI

So aktualisieren Sie einen CloudFormation Ingest-Stack mit der CLI

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

1. Bereiten Sie die AWS CloudFormation Vorlage vor, die Sie für die Aktualisierung des Stacks verwenden möchten, und laden Sie sie in Ihren S3-Bucket hoch. Wichtige Informationen finden Sie unter AWS CloudFormation Ingest Guidelines, Best Practices and Limitations.
2. Erstellen Sie den RFC und reichen Sie ihn an AMS ein:

   - Erstellen und speichern Sie die JSON-Datei mit den Ausführungsparametern, einschließlich der gewünschten CloudFormation Vorlagenparameter. In diesem Beispiel wird sie „ UpdateCfnParams.json“ genannt.

     Beispiel für eine UpdateCfnParams JSON-Datei mit Inline-Parameteraktualisierungen:

     ```
     {
       "StackId": "stack-yjjoo9aicjyqw4ro2",
       "VpcId": "VPC_ID",
       "CloudFormationTemplate": "{\"AWSTemplateFormatVersion\":\"2010-09-09\",
     \"Description\":\"Create a SNS topic\",\"Parameters\":{\"TopicName\":{\"Type
     \":\"String\"},\"DisplayName\":{\"Type\":\"String\"}},\"Resources\":{\"SnsTopic
     \":{\"Type\":\"AWS::SNS::Topic\",\"Properties\":{\"TopicName\":{\"Ref\":
     \"TopicName\"},\"DisplayName\":{\"Ref\":\"DisplayName\"}}}}}",
       "TemplateParameters": [
         {
           "Key": "TopicName",
           "Value": "TopicNameCLI"
         },
         {
           "Key": "DisplayName",
           "Value": "DisplayNameCLI"
         }
       ],
       "TimeoutInMinutes": 1440
     }
     ```

     Beispiel für eine UpdateCfnParams JSON-Datei mit einem S3-Bucket-Endpunkt, die eine aktualisierte CloudFormation Vorlage enthält:

     ```
     {
       "StackId": "stack-yjjoo9aicjyqw4ro2",
       "VpcId": "VPC_ID",
       "CloudFormationTemplateS3Endpoint": "s3_url",
       "TemplateParameters": [
         {
     ```

```
      "Key": "TopicName",
      "Value": "TopicNameCLI"
    },
    {
      "Key": "DisplayName",
      "Value": "DisplayNameCLI"
    }
  ],
  "TimeoutInMinutes": 1080
}
```

3. Erstellen und speichern Sie die JSON-Datei mit den RFC-Parametern mit dem folgenden Inhalt. In diesem Beispiel wird sie als UpdateCfnRfc JSON-Datei bezeichnet.

```
{
   "ChangeTypeId": "ct-361tlo1k7339x",
   "ChangeTypeVersion": "1.0",
   "Title": "cfn-ingest-template-update"
}
```

4. Erstellen Sie den RFC und geben Sie die UpdateCfnRfc Datei und die UpdateCfnParams Datei an:

```
aws amscm create-rfc --cli-input-json file://UpdateCfnRfc.json  --execution-
parameters file://UpdateCfnParams.json
```

   In der Antwort erhalten Sie die ID des neuen RFC und können damit den RFC einreichen und überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

- Dieser Änderungstyp ist jetzt in Version 2.0. Zu den Änderungen gehören das Entfernen des AutoApproveUpdateForResourcesParameters, der in Version 1.0 dieses CT verwendet wurde, und das Hinzufügen von zwei neuen Parametern: AutoApproveRiskyUpdatesund BypassDriftCheck.
- Wenn der S3-Bucket in einem AMS-Konto vorhanden ist, müssen Sie Ihre AMS-Anmeldeinformationen für diesen Befehl verwenden. Beispielsweise müssen Sie möglicherweise `--profile saml` nach Erhalt Ihrer AMS AWS Security Token Service (AWS STS) -Anmeldeinformationen eine Datei anhängen.

- Alle `Parameter` Werte für Ressourcen in der CloudFormation Vorlage müssen einen Wert haben, entweder durch einen Standardwert oder einen benutzerdefinierten Wert im Parameterbereich des CT. Sie können den Parameterwert überschreiben, indem Sie die CloudFormation Vorlagenressourcen so strukturieren, dass sie auf einen Parameterschlüssel verweisen. Beispiele, die zeigen, wie das geht, finden Sie unter CloudFormation Ingest Stack: CFN-Validator-Beispiele.

  WICHTIG: Fehlende Parameter, die nicht explizit im Formular angegeben wurden, verwenden standardmäßig die aktuell festgelegten Werte im vorhandenen Stack oder in der Vorlage.

- Eine Liste der selbst bereitgestellten Dienste, die Sie mithilfe von Ingest hinzufügen können, finden Sie unter AWS CloudFormation CloudFormation Ingest Stack: Unterstützte Ressourcen.

  Weitere Informationen finden Sie AWS CloudFormation unter AWS CloudFormation.

Eine Aufnahme validieren AWS CloudFormation

Die Vorlage wird validiert, um sicherzustellen, dass sie in einem AMS-Konto erstellt werden kann. Wenn sie die Validierung besteht, wird sie aktualisiert und enthält nun alle Ressourcen oder Konfigurationen, die für die AMS-Konformität erforderlich sind. Dazu gehört das Hinzufügen von Ressourcen wie CloudWatch Amazon-Alarmen, damit AMS Operations den Stack überwachen kann.

Der RFC wird abgelehnt, wenn eine der folgenden Bedingungen zutrifft:

- Die RFC-JSON-Syntax ist falsch oder folgt nicht dem angegebenen Format.
- Die angegebene vorsignierte URL für den S3-Bucket ist nicht gültig.
- Die Vorlage hat keine gültige AWS CloudFormation Syntax.
- Für die Vorlage sind keine Standardwerte für alle Parameterwerte festgelegt.
- Die Vorlage schlägt die AMS-Validierung fehl. Die Schritte zur AMS-Validierung finden Sie weiter unten in diesem Thema.

Der RFC schlägt fehl, wenn der CloudFormation Stack aufgrund eines Problems bei der Ressourcenerstellung nicht erstellt werden kann.

Weitere Informationen zur CFN-Validierung und -Validierung finden Sie unter Vorlagenvalidierung und CloudFormation Ingest-Stack: CFN-Validator-Beispiele.

CloudFormation Genehmigen Sie einen Changeset für den Ingest-Stack

Genehmigen und Aktualisieren eines CloudFormation Ingest-Stacks mithilfe der Konsole

Um einen CloudFormation Ingest-Stack mithilfe der Konsole zu genehmigen und zu aktualisieren

1. Navigieren Sie zur Seite „RFC erstellen“: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf Create RFC.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.
3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.
4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional

können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Genehmigen und Aktualisieren eines CloudFormation Ingest-Stacks mithilfe der CLI

So genehmigen und aktualisieren Sie einen CloudFormation Ingest-Stack mit der CLI

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *ID* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *ID*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

1. Geben Sie das JSON-Schema der Ausführungsparameter für diesen Änderungstyp in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird es „ CreateAsgParams.json“ genannt:

```
aws amscm create-rfc --change-type-id "ct-1404e21baa2ox" --change-
type-version "1.0" --title "Approve Update" --execution-parameters
 file://PATH_TO_EXECUTION_PARAMETERS --profile saml
```

2. Ändern und speichern Sie das Schema wie folgt:

```
{
  "StackId": "STACK_ID",
  "VpcId": "VPC_ID",
  "ChangeSetName": "UPDATE-ef81e2bc-03f6-4b17-a3c7-feb700e78faa",
  "TimeoutInMinutes": 1080
}
```

Tipps

> **Note**
>
> Wenn ein Stack mehrere Ressourcen enthält und Sie nur eine Teilmenge der Stack-Ressourcen löschen möchten, verwenden Sie das CloudFormation Update CT; siehe CloudFormation Ingest Stack: Updating. Sie können auch eine Serviceanfrage einreichen und die AMS-Techniker können Ihnen bei Bedarf bei der Erstellung des Changesets helfen.

Weitere Informationen AWS CloudFormation dazu finden Sie unter. AWS CloudFormation

Kündigungsschutz für AWS CloudFormation Update-Stacks

Aktualisierung eines AWS CloudFormation Terminierungsschutz-Stacks mit der Konsole

Im Folgenden wird dieser Änderungstyp in der AMS-Konsole dargestellt.

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.

- Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

  Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.
4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Aktualisierung eines AWS CloudFormation Stack-Termination-Schutzes mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.

2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *ID* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *ID*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

Geben Sie nur die Parameter an, die Sie ändern möchten. Fehlende Parameter behalten die vorhandenen Werte bei.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm create-rfc \
--change-type-id "ct-2uzbqr7x7mekd" \
--change-type-version "1.0" \
--title "Enable termination protection on CFN stack" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
ManageResourceTerminationProtection\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ResourceId\":[\"stack-psvnq6cupymio3enl\"],\"TerminationProtectionDesiredState\":
[\"enabled\"]}}"
```

VORLAGE ERSTELLEN:

1. Gibt die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei aus. In diesem Beispiel wird sie „ EnableTermProCFNParams.json“ genannt:

```
aws amscm get-change-type-version --change-type-id "ct-2uzbqr7x7mekd"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 EnableTermProCFNParams.json
```

2. Ändern und speichern Sie die EnableTermPro CFNParams Datei und behalten Sie dabei nur die Parameter bei, die Sie ändern möchten. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
  "DocumentName": "AWSManagedServices-ManageResourceTerminationProtection",
  "Region": "us-east-1",
  "Parameters": {
    "ResourceId": ["stack-psvnq6cupymio3enl"],
    "TerminationProtectionDesiredState": ["enabled"]
  }
}
```

3. Geben Sie die RFC-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. Dieses Beispiel nennt sie EnableTermPro CFNRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > EnableTermProCFNRfc.json
```

4. Ändern und speichern Sie die EnableTermPro CFNRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
    "ChangeTypeId": "ct-2uzbqr7x7mekd",
    "ChangeTypeVersion": "1.0",
    "Title": "Enable termination protection on CFN instance"
}
```

5. Erstellen Sie den RFC und geben Sie die EnableTermPro CFNRfc Datei und die EnableTermPro CFNParams Datei an:

```
aws amscm create-rfc --cli-input-json file://EnableTermProCFNRfc.json  --execution-
parameters file://EnableTermProCFNParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

> **Note**
>
> Es gibt ein verwandtes CT für Amazon EC2, EC2 Stack: Aktualisierung des Kündigungsschutzes.

Weitere Informationen zum Kündigungsschutz finden Sie unter Einen Stack vor dem Löschen schützen.

Automatisierte IAM-Bereitstellungen mithilfe von CFN-Ingest oder Stack-Update in AMS CTs

Sie können diese AMS-Änderungstypen verwenden, um IAM-Rollen (die `AWS::IAM::Role` Ressource) sowohl in der Multi-Account-Landingzone (MALZ) als auch in der Single-Account-Landingzone (SALZ) bereitzustellen:

- Bereitstellung | Aufnahme | Aus CloudFormation Vorlage stapeln | Erstellen (ct-36cn2avfrrj9v)
- Verwaltung | Benutzerdefinierter Stack | Stapel aus Vorlage | Update (ct-361tlo1k7339x) CloudFormation
- Verwaltung | Benutzerdefinierter Stack | Stapel aus Vorlage | Genehmigen und aktualisieren (ct-1404e21baa2ox) CloudFormation

Validierungen, die für die IAM-Rollen in Ihrer CFN-Vorlage durchgeführt wurden:

- ManagedPolicyArns: Das Attribut ManagedPolicyArnsdarf in nicht existieren. `AWS::IAM::Role` Die Validierung verhindert das Anhängen verwalteter Richtlinien an die bereitgestellte Rolle. Stattdessen können die Berechtigungen für die Rolle mithilfe der Inline-Richtlinie über die Eigenschaft Richtlinien verwaltet werden.
- PermissionsBoundary: Bei der Richtlinie, die zur Festlegung der Berechtigungsgrenze für die Rolle verwendet wurde, kann es sich nur um die von AMS angebotene verwaltete Richtlinie

handeln:`AWSManagedServices_IAM_PermissionsBoundary`. Diese Richtlinie dient als Leitplanke, die die AMS-Infrastrukturressourcen davor schützt, dass sie durch die bereitgestellte Rolle verändert werden. Mit dieser Standardberechtigungsgrenze bleiben die Sicherheitsvorteile, die AMS bietet, erhalten.

Die `AWSManagedServices_IAM_PermissionsBoundary` (Standardeinstellung) ist erforderlich. Andernfalls wird die Anfrage abgelehnt.

- MaxSessionDuration: Die maximale Sitzungsdauer, die für die IAM-Rolle festgelegt werden kann, beträgt 1 bis 4 Stunden. Gemäß den technischen Standards von AMS muss der Kunde das Risiko eingehen, wenn eine Sitzung länger als 4 Stunden dauert.
- RoleName: Die folgenden Namespaces werden von AMS beibehalten und können nicht als Präfixe für IAM-Rollennamen verwendet werden:

```
AmazonSSMRole,
AMS,
Ams,
ams,
AWSManagedServices,
customer_developer_role,
customer-mc-,
Managed_Services,
MC,
Mc,
mc,
SENTINEL,
Sentinel,
sentinel,
StackSet-AMS,
StackSet-Ams,
StackSet-ams,
StackSet-AWS,
StackSet-MC,
StackSet-Mc,
StackSet-mc
```

- Richtlinien: Die in die IAM-Rolle eingebettete Inline-Richtlinie kann nur eine Reihe von IAM-Aktionen enthalten, die vorab von AMS genehmigt wurden. Dies ist die Obergrenze aller IAM-Aktionen, mit denen eine IAM-Rolle erstellt werden darf (Kontrollrichtlinie). Die Kontrollrichtlinie besteht aus:

- Alle Aktionen in der AWS verwalteten Richtlinie ReadOnlyAccess , die nur Lesezugriff auf alle AWS-Services Ressourcen bietet
- Die folgenden Aktionen, mit der Beschränkung auf kontoübergreifende S3-Aktionen, d. h. zulässige S3-Aktionen, können nur für Ressourcen ausgeführt werden, die sich in demselben Konto befinden wie die zu erstellende Rolle:

```
amscm:*,
amsskms:*,
lambda:InvokeFunction,
logs:CreateLogStream,
logs:PutLogEvents,
s3:AbortMultipartUpload,
s3:DeleteObject,
s3:DeleteObjectVersion,
s3:ObjectOwnerOverrideToBucketOwner,
s3:PutObject,
s3:ReplicateTags,
secretsmanager:GetRandomPassword,
sns:Publish
```

  Jede IAM-Rolle, die durch CFN-Ingest erstellt oder aktualisiert wurde, kann Aktionen zulassen, die in dieser Kontrollrichtlinie aufgeführt sind, oder Aktionen, die von den in der Kontrollrichtlinie aufgeführten Aktionen abweichen (weniger freizügig sind als). Derzeit sind diese sicheren IAM-Aktionen zulässig, die als schreibgeschützte Aktionen eingestuft werden können, sowie die oben genannten Aktionen ohne Lesezugriff, die nicht über den technischen Standard von AMS durchgeführt werden können und die vorab genehmigt wurden. CTs

- AssumeRolePolicyDocument: Die folgenden Entitäten wurden vorab genehmigt und können in die Vertrauensrichtlinie aufgenommen werden, sodass sie die zu erstellende Rolle übernehmen können:
  - Jede IAM-Entität (Rolle, Benutzer, Root-Benutzer, STS-Sitzung mit übernommener Rolle) in demselben Konto kann die Rolle übernehmen.
  - Folgende Personen AWS-Services können die Rolle übernehmen:

```
apigateway.amazonaws.com,
autoscaling.amazonaws.com,
cloudformation.amazonaws.com,
codebuild.amazonaws.com,
codedeploy.amazonaws.com,
```

```
codepipeline.amazonaws.com,
datapipeline.amazonaws.com,
datasync.amazonaws.com,
dax.amazonaws.com,
dms.amazonaws.com,
ec2.amazonaws.com,
ecs-tasks.amazonaws.com,
ecs.application-autoscaling.amazonaws.com,
elasticmapreduce.amazonaws.com,
es.amazonaws.com,
events.amazonaws.com,
firehose.amazonaws.com,
glue.amazonaws.com,
lambda.amazonaws.com,
monitoring.rds.amazonaws.com,
pinpoint.amazonaws.com,
rds.amazonaws.com,
redshift.amazonaws.com,
s3.amazonaws.com,
sagemaker.amazonaws.com,
servicecatalog.amazonaws.com,
sns.amazonaws.com,
ssm.amazonaws.com,
states.amazonaws.com,
storagegateway.amazonaws.com,
transfer.amazonaws.com,
vmie.amazonaws.com
```

- Der SAML-Anbieter im selben Konto kann die Rolle übernehmen. Derzeit ist der einzige unterstützte SAML-Anbietername. `customer-saml`

Wenn eine oder mehrere der Validierungen fehlschlagen, wird der RFC abgelehnt. Ein Beispiel für einen RFC-Ablehnungsgrund sieht wie folgt aus:

```
{"errorMessage":"[ 'LambdaRole: The maximum session duration (in seconds) should be a
 numeric value in the range 3600 to 14400 (i.e. 1 to 4 hours).', 'lambda-policy: Policy
 document is too permissive.']","errorType":"ClientError"}
```

Wenn Sie Hilfe bei einer fehlgeschlagenen RFC-Validierung oder -Ausführung benötigen, wenden Sie sich über die RFC-Korrespondenz an AMS. Anweisungen finden Sie unter RFC-Korrespondenz und Anhang (Konsole). Wenn Sie weitere Fragen haben, senden Sie eine Serviceanfrage. Eine Anleitung finden Sie unter Serviceanfrage erstellen.

> **Note**
>
> Derzeit setzen wir im Rahmen unserer IAM-Validierungen keine Best Practices für IAM durch. Bewährte Methoden für IAM finden Sie unter Bewährte Methoden zur Sicherheit in IAM.

IAM-Rollen mit freizügigeren Aktionen erstellen oder bewährte IAM-Praktiken durchsetzen

Erstellen Sie Ihre IAM-Entitäten mit den folgenden manuellen Änderungstypen:

- Bereitstellung | Erweiterte Stack-Komponenten | Identity and Access Management (IAM) | Entität oder Richtlinie erstellen (ct-3dpd8mdd9jn1r)
- Verwaltung | Erweiterte Stack-Komponenten | Identity and Access Management (IAM) | Entität oder Richtlinie aktualisieren (ct-27tuth19k52b4)

Wir empfehlen Ihnen, unsere technischen Standards zu lesen und zu verstehen, bevor Sie dieses Handbuch einreichen. RFCs Informationen zum Zugriff finden Sie unter So greifen Sie auf technische Standards zu.

> **Note**
>
> Jede IAM-Rolle, die direkt mit diesen manuellen Änderungstypen erstellt wurde, gehört zu ihrem eigenen individuellen Stack und befindet sich nicht in demselben Stack, in dem die anderen Infrastrukturressourcen über CFN Ingest CT erstellt werden.

Aktualisierung von IAM-Rollen, die mit CFN Ingest erstellt wurden, mithilfe manueller Änderungstypen, wenn Aktualisierungen nicht über automatisierte Änderungstypen durchgeführt werden können

Verwenden Sie den Änderungstyp Verwaltung | Erweiterte Stack-Komponenten | Identity and Access Management (IAM) | Entität oder Richtlinie aktualisieren (ct-27tuth19k52b4).

> **Important**
>
> Aktualisierungen von IAM-Rollen durch das manuelle CT spiegeln sich nicht in den CFN-Stack-Vorlagen wider und führen zu Stack-Drift. Sobald die Rolle durch eine manuelle Anfrage auf einen Status aktualisiert wurde, der unsere Validierungen nicht bestanden hat,

kann die Rolle nicht erneut mit dem Stack Update CT (ct-361tlo1k7339x) aktualisiert werden, solange sie weiterhin nicht unseren Validierungen entspricht. Das Update-CT kann nur verwendet werden, wenn die CFN-Stack-Vorlage unseren Validierungen entspricht. Der Stack kann jedoch weiterhin über das Stack Update CT (ct-361tlo1k7339x) aktualisiert werden, solange die IAM-Ressource, die unseren Validierungen nicht entspricht, nicht aktualisiert wird und die CFN-Vorlage unsere Validierungen besteht.

Löschen Ihrer AWS CloudFormation durch Ingest erstellten IAM-Rollen

Wenn Sie den gesamten Stack löschen möchten, verwenden Sie den folgenden automatisierten Änderungstyp „Stack löschen“. Anweisungen finden Sie unter Stapel löschen:

- Typ-ID ändern: ct-0q0bic0ywqk6c
- Klassifizierung: Verwaltung | Standard-Stacks | Stapel | Löschen und Verwalten | Erweiterte Stack-Komponenten | Stapel | Löschen

Wenn Sie eine IAM-Rolle löschen möchten, ohne den gesamten Stack zu löschen, können Sie die IAM-Rolle aus der CloudFormation Vorlage entfernen und die aktualisierte Vorlage als Eingabe für den automatisierten Änderungstyp Stack Update verwenden:

- Typ-ID ändern: ct-361tlo1k7339x
- Klassifizierung: Verwaltung | Benutzerdefinierter Stapel | Stapel aus Vorlage | Update CloudFormation

Anweisungen finden Sie unter AWS CloudFormation Ingest-Stack aktualisieren.

CodeDeploy Anfragen

Sie können AWS verwenden CodeDeploy , um Anwendungscontainer zu erstellen, die Sie dann über eine CodeDeploy Anwendungsgruppe bereitstellen können. Weitere Informationen CodeDeploy dazu finden Sie in der CodeDeploy AWS-Dokumentation.

Die Zusammenarbeit mit AWS CodeDeploy umfasst den folgenden Prozess:

1. Erstellen Sie eine CodeDeploy Anwendung. Die CodeDeploy Anwendung ist ein Name oder Container, der verwendet wird, CodeDeploy um sicherzustellen, dass während einer Bereitstellung auf die richtige Version, Bereitstellungskonfiguration und Bereitstellungsgruppe verwiesen wird.

2. Erstellen Sie eine CodeDeploy Bereitstellungsgruppe. Eine CodeDeploy Bereitstellungsgruppe definiert eine Reihe von einzelnen Instanzen, die für eine Bereitstellung vorgesehen sind. AMS hat einen separaten Änderungstyp für CodeDeploy Bereitstellungsgruppen für EC2.
3. Stellen Sie die CodeDeploy Anwendung über die CodeDeploy Bereitstellungsgruppe bereit.

CodeDeploy Anwendung

CodeDeploy Anwendungen erstellen oder bereitstellen.

CodeDeploy Anwendung erstellen

Eine CodeDeploy Anwendung mit der Konsole erstellen

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.
3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Eine CodeDeploy Anwendung mit der CLI erstellen

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com`

\"]}}" Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm create-rfc --change-type-id "ct-0ah3gwb9seqk2" --change-type-version "1.0"
 --title "Stack-Create-CD-App" --execution-parameters "{\"Description\":\"TestCdApp\",
\"VpcId\":\"VPC_ID\",\"StackTemplateId\":\"stm-sft6rv00000000000\",\"Name\":\"Test\",
\"TimeoutInMinutes\":60,\"Parameters\":{\"CodeDeployApplicationName\":\"Test\"}}"
```

VORLAGE ERSTELLEN:

1. Geben Sie das JSON-Schema der Ausführungsparameter für die CodeDeploy Anwendung CT in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie Create CDApp Params.json genannt:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ah3gwb9seqk2" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateCDAppParams.json
   ```

2. Ändern und speichern Sie die JSON-Datei wie folgt. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

   ```
   {
   "Description":                      "Create WP CodeDeploy App",
   "VpcId":                            "VPC_ID",
   "StackTemplateId":                  "stm-sft6rv00000000000",
   "Name":                             "WpCDApp",
   "TimeoutInMinutes":                 60,
   "Parameters":   {
       "CodeDeployApplicationName":    "WordPressCDApp"
       }
   }
   ```

3. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie Create CDApp rfc.Json genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDAppRfc.json
```

4. Ändern und speichern Sie die JSON-Datei wie folgt. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0ah3gwb9seqk2",
"Title":                "CD-App-Stack-RFC"
}
```

5. Erstellen Sie den RFC und geben Sie die Datei Create CDApp Rfc und die Datei mit den Ausführungsparametern an:

```
aws amscm create-rfc --cli-input-json file://CreateCDAppRfc.json --execution-
parameters file://CreateCDAppParams.json
```

   Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

Weitere Informationen zu AWS CodeDeploy finden Sie unter Erstellen einer Anwendung mit AWS CodeDeploy.

CodeDeploy Anwendung bereitstellen

Eine CodeDeploy Anwendung mit der Konsole bereitstellen

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.

- Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

  Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.
4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Bereitstellen einer CodeDeploy Anwendung mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.

2. Reichen Sie den `aws amscm submit-rfc --rfc-id ID` Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id ID`

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben) und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm create-rfc --change-type-id "ct-2edc3sd1sqmrb" --change-
type-version "2.0" --title "Stack-Deploy-CD-App" --execution-
parameters "{\"Description\":\"MyCDAppDeployTest\",\"VpcId\":
\"VPC_ID\",\"Name\":\"Test\",\"TimeoutInMinutes\":60,\"Parameters\":
{\"CodeDeployApplicationName\":\"TestCDApp\",\"CodeDeployDeploymentConfigName\":
\"CodeDeployDefault.OneAtATime\",\"CodeDeployDeploymentGroupName\":\"TestCDDepGroup\",
\"CodeDeployIgnoreApplicationStopFailures\":false,\"CodeDeployRevision\":
{\"RevisionType\":\"S3\",\"S3Location\":{\"S3Bucket\":\"amzn-s3-demo-bucket\",
\"S3BundleType\":\"tar\",\"S3Key\":\"TestKey\"}}}}"Test\"}}"
```

VORLAGE ERSTELLEN:

1. Gibt das JSON-Schema der Ausführungsparameter für das CodeDeploy Anwendungs-Deployment CT aus. In diesem Beispiel wird es Deploy CDApp Params.json genannt:

```
aws amscm get-change-type-version --change-type-id "ct-2edc3sd1sqmrb" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > DeployCDAppParams.json
```

2. Ändern Sie die JSON-Datei wie folgt. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":                      "Deploy WordPress CodeDeploy Application",
"VpcId":                            "VPC_ID",
"Name":                             "WP CodeDeploy Deployment Group",
"TimeoutInMinutes":                 360,
"Parameters":   {
    "CodeDeployApplicationName":        "WordPressCDApp",
    "CodeDeployDeploymentGroupName":    "WordPressCDDepGroup",
    "CodeDeployIgnoreApplicationStopFailures": false,
    "CodeDeployRevision": {
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket": "amzn-s3-demo-bucket",
        "S3BundleType": "zip",
        "S3Key": "wordpress.zip" }
        }
    }
}
```

3. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie Deploy CDApp rfc.Json genannt:

```
aws amscm create-rfc --generate-cli-skeleton > DeployCDAppRfc.json
```

4. Ändern und speichern Sie die Datei Deploy CDApp RFC.json. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-CD-APP-Stack-RFC"
}
```

5. Erstellen Sie den RFC und geben Sie dabei die Ausführungsparameterdatei und die CDApp Deploy-RFC-Datei an:

```
aws amscm create-rfc --cli-input-json file://DeployCDAppRfc.json  --execution-
parameters file://DeployCDAppParams.json
```

   Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

Weitere Informationen finden Sie unter Erstellen einer Bereitstellung mit CodeDeploy.

CodeDeploy Bereitstellungsgruppen

Erstellen Sie CodeDeploy Anwendungsgruppen.

CodeDeploy Bereitstellungsgruppe erstellen

Erstellen einer CodeDeploy Bereitstellungsgruppe mit der Konsole

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können

Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.

5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Erstellen einer CodeDeploy Bereitstellungsgruppe mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *ID* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *ID*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben) und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm create-rfc --change-type-id "ct-2gd0u847qd9d2" --change-type-version
 "1.0" --title "Stack-Create-CD-Dep-Group" --execution-parameters "{\"Description
\":\"TestCdDepGroupRfc\",\"VpcId\":\"VPC_ID\",\"StackTemplateId\":\"stm-
sp9lrk00000000000\",\"Name\":\"MyTestCDDepGroup\",\"TimeoutInMinutes\":60,\"Parameters
\":{\"CodeDeployApplicationName\":\"TestCDApp\",\"CodeDeployAutoScalingGroups\":
[\"TestASG\"],\"CodeDeployDeploymentConfigName\":\"CodeDeployDefault.OneAtATime\",
\"CodeDeployDeploymentGroupName\":\"Test\",\"CodeDeployServiceRoleArn\":
\"arn:aws:iam::000000000:role/aws-codedeploy-role\"}}"
```

VORLAGE ERSTELLEN:

1. Geben Sie das JSON-Schema der Ausführungsparameter in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie Create CDDep GroupParams .json genannt:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2gd0u847qd9d2"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateCDDepGroupParams.json
   ```

2. Ändern und speichern Sie die JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

   ```
   {
   ```

```
"Description":                          "CreateCDDeploymentGroup",
"VpcId":                                "VPC_ID",
"StackTemplateId":                      "stm-sp9lrk00000000000",
"Name":                                 "WordPressCDAppGroup",
"TimeoutInMinutes":                     60,
"Parameters":   {
    "CodeDeployApplicationName":        "WordPressCDApp",
    "CodeDeployAutoScalingGroups":      ["ASG_NAME"],
    "CodeDeployDeploymentConfigName":   "CodeDeployDefault.HalfAtATime",
    "CodeDeployDeploymentGroupName":    "UNIQUE_CDDepGroupNAME",
    "CodeDeployServiceRoleArn":         "arn:aws:iam::ACCOUNT_ID:role/aws-
codedeploy-role"
    }
}
```

3. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie Create CDDep GroupRfc .json genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDDepGroupRfc.json
```

4. Ändern und speichern Sie die JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2gd0u847qd9d2",
"Title":                "CD-Dep-Group-RFC"
}
```

5. Erstellen Sie den RFC, indem Sie die CDDep GroupRfc Datei Create und die Datei mit den Ausführungsparametern angeben:

```
aws amscm create-rfc --cli-input-json file://CreateCDDepGroupRfc.json --execution-
parameters file://CreateCDDepGroupParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

Weitere Informationen zu CodeDeploy AWS-Bereitstellungsgruppen finden Sie unter Erstellen einer Bereitstellungsgruppe mit AWS CodeDeploy.

Erstellen Sie eine CodeDeploy Bereitstellungsgruppe für EC2

Erstellen Sie eine CodeDeploy Bereitstellungsgruppe für EC2 mit der Konsole

So funktioniert es:

1. Navigieren Sie zur Seite „RFC erstellen“: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.
3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Erstellen einer CodeDeploy Bereitstellungsgruppe für EC2 mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm create-rfc --change-type-id "ct-00tlkda4242x7" --change-type-
version "1.0" --title "Stack-Create-CD-Ec2-Dep-Group" --execution-parameters
 "{\"Description\":\"MyTestCdDepEc2DepGroup\",\"VpcId\":\"VPC_ID\",\"Name\":
\"TestCDDepEc2Group\",\"StackTemplateId\":\"stm-n3hsoirgqeqqdbpk2\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"ApplicationName\":\"TestCDApp\",\"DeploymentConfigName\":
\"CodeDeployDefault.OneAtATime\",\"AutoRollbackEnabled\":\"False\",\"EC2FilterTag\":
\"Name=Test\",\"EC2FilterTag2\":\"\",\"EC2FilterTag3\":\"\",\"ServiceRoleArn\":\"\"}}"
```

VORLAGE ERSTELLEN:

1. Gibt das JSON-Schema der Ausführungsparameter in eine Datei aus. In diesem Beispiel wird sie Create CDDep GroupEc 2Params.json genannt:

   ```
   aws amscm get-change-type-version --change-type-id "ct-00tlkda4242x7"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateCDDepGroupEc2Params.json
   ```

2. Ändern und speichern Sie die JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

   ```
   {
   "Description":                    "CreateCDDepGroupEc2",
   "VpcId":                          "VPC_ID",
   "StackTemplateId":                "stm-n3hsoirgqeqqdbpk2",
   "Name":                           "CDAppGroupEc2",
   "TimeoutInMinutes":               60,
   "Parameters":   {
       "ApplicationName":        "CDAppEc2",
       "DeploymentConfigName":   "CodeDeployDefault.OneAtATime",
       "CodeDeployDeploymentGroupName":    "UNIQUE_CDDepGroupNAME",
       "CodeDeployServiceRoleArn":         "arn:aws:iam::ACCOUNT_ID:role/aws-
   codedeploy-role"
       }
   }
   ```

3. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie Create CDDep GroupEc 2RFC.json genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDDepGroupEc2Rfc.json
```

4. Ändern und speichern Sie die JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-00tlkda4242x7",
"Title":                "CD-Dep-Group-For-Ec2-Stack-RFC"
}
```

5. Erstellen Sie den RFC, indem Sie die Datei Create CDDep GroupEc 2Rfc und die Datei mit den Ausführungsparametern angeben:

```
aws amscm create-rfc --cli-input-json file://CreateCDDepGroupEc2Rfc.json --
execution-parameters file://CreateCDDepGroupEc2Params.json
```

   Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

Weitere Informationen zu CodeDeploy AWS-Bereitstellungsgruppen finden Sie unter Erstellen einer Bereitstellungsgruppe mit AWS CodeDeploy.

AWS Database Migration Service (AWS DMS)

AWS Database Migration Service (AWS DMS) hilft Ihnen dabei, Datenbanken einfach und sicher zu AMS zu migrieren. Sie können Ihre Daten von und zu den gängigsten kommerziellen und Open-Source-Datenbanken migrieren, wie z. B. Oracle, MySQL und PostgreSQL. Der Service unterstützt homogene Migrationen wie Oracle zu Oracle sowie heterogene Migrationen zwischen verschiedenen Datenbankplattformen wie Oracle zu PostgreSQL oder MySQL zu Oracle. AWS DMS ist ein AWS Service; das AMS CTs hilft Ihnen dabei, Ressourcen in Ihrem von AMS verwalteten Konto zu erstellen AWS DMS

Die folgende Grafik zeigt den Arbeitsablauf einer Datenbankmigration.

Themen

AWS Database Migration Service (AWS DMS), bevor du anfängst

Beachten Sie bei der Planung einer Datenbankmigration mit dem AMS AWS DMS Folgendes:

- Quell- und Zielendpunkte: Sie müssen wissen, welche Informationen und Tabellen in der Quelldatenbank in die Zieldatenbank migriert werden müssen. AMS AWS DMS unterstützt die grundlegende Schemamigration, einschließlich der Erstellung von Tabellen und Primärschlüsseln. AMS erstellt jedoch AWS DMS nicht automatisch Sekundärindizes, Fremdschlüssel, Konten usw. in der Zieldatenbank. Weitere Informationen finden Sie unter Quellen für die Datenmigration und Ziele für die Datenmigration.
- Schema-/Code-Migration: AMS führt AWS DMS keine Schema- oder Codekonvertierung durch. Sie können Ihr Schema mithilfe von Tools wie Oracle SQL Developer, MySQL Workbench oder pgAdmin III umwandeln. Wenn Sie ein vorhandenes Schema in eine andere Datenbank-Engine konvertieren möchten, können Sie das AWS Schema Conversion Tool verwenden. Dieses Tool kann ein Zielschema erstellen, aber auch ein ganzes Schema generieren und erstellen, wie Tabellen, Indizes, Ansichten und so weiter. Sie können das Tool auch verwenden, um PL/SQL oder TSQL in PgSQL und andere Formate zu konvertieren.
- Nicht unterstützte Datentypen: Einige Quelldatentypen müssen in die entsprechenden Datentypen für die Zieldatenbank konvertiert werden.

AWS DMS zu berücksichtigende Szenarien

Die folgenden dokumentierten Szenarien können Ihnen dabei helfen, Ihren eigenen Datenbankmigrationspfad zu erstellen.

- Migrieren Sie Daten von einem lokalen MySQL-Server zu Amazon RDS MySQL: Siehe AWS-Blogbeitrag Migrieren von lokalen MySQL-Daten zu Amazon RDS (und zurück)

- Daten von einer Oracle-Datenbank zur Amazon RDS Aurora PostgreSQL-Datenbank migrieren: Siehe AWS-Blogbeitrag Eine kurze Einführung in die Migration von einer Oracle-Datenbank zu einer Amazon Aurora PostgreSQL-Datenbank
- Migrieren Sie Daten von RDS MySQL zu S3: Siehe AWS-Blogbeitrag So archivieren Sie Daten aus relationalen Datenbanken mit AWS DMS in Amazon Glacier

Für eine Datenbankmigration müssen Sie die folgenden Schritte ausführen:

- Planen Sie Ihre Datenbankmigration, dazu gehört auch die Einrichtung einer Replikationssubnetzgruppe.
- Ordnen Sie eine Replikationsinstanz zu, die alle Prozesse für die Migration ausführt.
- Geben Sie einen Quell- und einen Zieldatenbank-Endpunkt an.
- Erstellen Sie eine oder mehrere Aufgaben, um festzulegen, welche Tabellen und Replikationsprozesse verwendet werden sollen.
- Erstellen Sie das AWS DMS IAM `dms-cloudwatch-logs-role` und die `dms-vpc-role` Rollen. Wenn Sie Amazon Redshift als Zieldatenbank verwenden, müssen Sie auch die IAM-Rolle erstellen und `dms-access-for-endpoint` zu Ihrem AWS-Konto hinzufügen. Weitere Informationen finden Sie unter Erstellen der IAM-Rollen zur Verwendung mit der AWS-CLI und der AWS DMS-API.

Diese exemplarischen Vorgehensweisen bieten ein Beispiel für die Verwendung der AMS-Konsole oder der AMS-CLI zur Erstellung eines AWS Database Migration Service ()AWS DMS. CLI-Befehle zum Erstellen der AWS DMS Replikationsinstanz, der Subnetzgruppe und der Aufgabe sowie eines AWS DMS Quellendpunkts und eines Zielendpunkts werden bereitgestellt.

Weitere Informationen zu AMS AWS DMS finden Sie unter allgemeine AWS Database Migration ServiceInformationen und Antworten AWS Database Migration Service FAQsauf häufig gestellte Fragen.

AWS DMS, erforderliche Daten für die Einrichtung

Für jede der folgenden AWS DMS exemplarischen Vorgehensweisen sind einige gemeinsame Daten erforderlich.

- `Description`: Aussagekräftige Informationen über die Ressource, getrennt von anderen `Description` Parameteroptionen.

- `VpcId`: Die zu verwendende VPC. Sie können dies herausfinden, indem Sie den ListVpcSummaries Betrieb der SKMS-API (`list-vpc-summaries`in der CLI) ausführen oder auf der VPCsSeite in der AMS-Konsole nachschauen. Die AMS SKMS API-Referenz finden Sie auf der Registerkarte Berichte in der AWS Artifact Console.
- `Name`: Ein Name für den Stack oder die Stack-Komponente; daraus wird der Stack-Name.
- `TimeoutInMinutes`: Wie viele Minuten sind für die Erstellung des Stacks vorgesehen, bevor der RFC fehlschlägt. Diese Einstellung verzögert die RFC-Ausführung nicht, Sie müssen jedoch genügend Zeit einplanen (z. B. nicht angeben`"5"`).
- `ChangeTypeId`,`ChangeTypeVersion`, und`StackTemplateId`: Diese sind erforderlich, variieren jedoch je nach CT und ihre Werte werden in den jeweiligen nachfolgenden Abschnitten angegeben.

AWS DMS Einrichtungsaufgaben

Richten Sie sich AWS DMS mit den folgenden exemplarischen Vorgehensweisen ein.

1: AWS DMS Replikationssubnetzgruppe: Erstellen

Sie können die AMS-Konsole verwenden oder API/CLI eine AWS DMS AMS-Replikationssubnetzgruppe erstellen.

Erstellen Sie eine AWS DMS Replizierungssubnetzgruppe

Erstellen einer AWS DMS Replikationssubnetzgruppe mit der Konsole

> **Note**
>
> Dieser CT schlägt fehl, wenn die `dms-vpc-role` IAM-Rolle im Konto nicht vorhanden ist.

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder in der Ansicht „Nach Kategorie auswählen“ einen CT aus.

- Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

  Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.

- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.

5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Erstellen einer AWS DMS Replikationssubnetzgruppe mit der CLI

> **Note**
>
> Dieser CT schlägt fehl, wenn die `dms-vpc-role` IAM-Rolle im Konto nicht vorhanden ist.

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *ID* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *ID*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben) und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-2q5azjd8p1ag5" --change-type-version "1.0" --title "TestDMSRepSG" --execution-
parameters "{\"Description\":\"DMSTestRepSG\",\"VpcId\":\"VPC-ID\",\"Name\":\"Test
 Stack\",\"Parameters\":{\"Description\":\"DESCRIPTION\",\"SubnetIds\":[\"SUBNET-ID\",
 \"SUBNET-ID\"]},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-j637f96ls1h4oy5fj
\"}"
```

VORLAGE ERSTELLEN:

1. Gibt die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei aus. Dieses Beispiel nennt sie CreateDmsRsgParams .json:

```
aws amscm get-change-type-version --change-type-id "ct-2q5azjd8p1ag5" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRsgParams.json
```

2. Ändern und speichern Sie die Ausführungsparameter in der CreateDmsRsgParams JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":          "DMSTestRepSG",
"VpcId":                "VPC_ID",
"TimeoutInMinutes":     60,
"StackTemplateId":      "stm-j637f96ls1h4oy5fj",
"Name":                 "Test RSG",
"Parameters":   {
    "Description":            "DESCRIPTION",
    "SubnetIds":              ["SUBNET_ID", "SUBNET_ID"]
    }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie „ CreateDmsRsgRfc.json“ genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRsgRfc.json
```

4. Ändern und speichern Sie die CreateDmsRsgRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2q5azjd8p1ag5",
"Title":                "DMS-RSG-Create-RFC"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die CreateDmsRsgRfc Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRsgRfc.json --execution-
parameters file://CreateDmsRsgParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

- Dieser CT schlägt fehl, wenn die `dms-vpc-role` IAM-Rolle im Konto nicht vorhanden ist.
- Sie können bis zu 50 Tags hinzufügen. Dazu müssen Sie jedoch die Ansicht Zusätzliche Konfiguration aktivieren.

  Weitere Informationen zu DMS-Replikationsinstanzen und Subnetzgruppen finden Sie unter Netzwerk für eine Replikationsinstanz einrichten.

2: AWS DMS Replikationsinstanz: Erstellen

Sie können die AMS-Konsole verwenden oder API/CLI eine AWS DMS AMS-Replikationsinstanz erstellen.

AWS DMS Replikationsinstanz erstellen

Eine AWS DMS Replikationsinstanz mit der Konsole erstellen

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.

- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.
4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Erstellen einer AWS DMS Replikationsinstanz mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-27apldkhqr0ol" --change-type-version "1.0" --title "TestDMSRepInstance" --
execution-parameters "{\"Description\":\"DMSTestRepInstance\",\"VpcId\":\"VPC-ID\",
\"Name\":\"REP-INSTANCE-NAME\",\"Parameters\":{\"InstanceClass\":\"dms.t2.micro\",
\"ReplicationSubnetGroupIdentifier\":\"TEST-REP-SG\",\"SecurityGroupIds\":\"SG-ID, SG-
ID\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-3n1j5hdrmiiiuqk6v\"}"
```

Während die Replikations-Instance erstellt wird, können Sie die Quell- und Zieldatenspeicher angeben. Die Quell- und Zieldatenspeicher können sich auf einer Amazon Elastic Compute Cloud (Amazon EC2) -Instance, einem AWS S3 Bucket, einer Amazon Relational Database Service (Amazon RDS) -DB-Instance oder einer lokalen Datenbank befinden.

VORLAGE ERSTELLEN:

1. Gibt die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei aus. In diesem Beispiel wird sie „ CreateDmsRiParams.json“ genannt:

```
aws amscm get-change-type-version --change-type-id "ct-27apldkhqr0ol" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRiParams.json
```

2. Ändern und speichern Sie die Ausführungsparameter in der CreateDmsRiParams JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":          "DMSTestRepInstance",
"VpcId":                "VPC_ID",
"Name":                 "Test RI",
"StackTemplateId":      "stm-3n1j5hdrmiiiuqk6v",
"TimeoutInMinutes":     60,
"Parameters":   {
    "Description":                      "DESCRIPTION",
    "InstanceClass":                    "dms.t2.micro",
    "ReplicationSubnetGroupIdentifier": "TEST-REP-SG",
    "SecurityGroupIds":                 ["SG-ID, SG-ID"}
    }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie „ CreateDmsRiRfc.json“ genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRiRfc.json
```

4. Ändern und speichern Sie die CreateDmsRiRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-27apldkhqr0ol",
"Title":                "DMS-RI-Create-RFC"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die CreateDmsRiRfc Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRiRfc.json --execution-
parameters file://CreateDmsRiParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

- Sie können bis zu 50 Tags hinzufügen. Dazu müssen Sie jedoch die Ansicht Zusätzliche Konfiguration aktivieren.
- Sie müssen eine Replikationsinstanz auf einer EC2 Instance in Ihrer AMS-VPC erstellen, die über ausreichend Speicher und Rechenleistung verfügt, um die von Ihnen zugewiesenen Aufgaben auszuführen und Daten von Ihrer Quelldatenbank zur Zieldatenbank zu migrieren. Die erforderliche Größe dieser Instance hängt von der Menge der Daten ab, die Sie migrieren müssen, sowie von den Aufgaben, die die Instance ausführen muss. Die Replikationsinstanz bietet Hochverfügbarkeit und Failover-Unterstützung mithilfe einer Multi-AZ-Bereitstellung, wenn Sie die Option auswählen. `MultiAZ` Weitere Informationen zu Replikationsinstanzen finden Sie unter Arbeiten mit einer AWS DMS-Replikationsinstanz.

3: AWS DMS Quellendpunkt: Erstellen, für Mongo DB erstellen, für S3 erstellen

Sie können die AMS-Konsole verwenden oder API/CLI um einen AMS DMS-Quellendpunkt für verschiedene Datenbanken zu erstellen. Wir stellen drei Beispiele zur Verfügung.

DMS-Quellendpunkt: erstellen

Einen DMS-Quellendpunkt mit der Konsole erstellen

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.

- Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

  Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.
4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Einen DMS-Quellendpunkt mit der CLI erstellen

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.

2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *ID* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *ID*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws --profile saml --region us-east-1 amscm create-rfc --title "MariaDB-DMS-
Source-Endpoint" --aws-account-id ACCOUNT-ID --change-type-id ct-0attesnjqy2cx --
change-type-version 1.0 --execution-parameters "{\"Description\":\"DESCRIPTION.\",
\"VpcId\":\"VPC-ID\",\"Name\":\"MariaDB-DMS-SE\",\"Parameters\":{\"EngineName\":
\"mariadb\",\"ServerName\":\"mariadb.db.example.com\",\"Port\":3306,\"Username\":
\"DB-USER\",\"Password\":\"DB-PW\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-
pud4ghhkp7395n9bc\"}"
```

VORLAGE ERSTELLEN:

1. Geben Sie die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei namens CreateDmsSeParams .json aus.

```
aws amscm get-change-type-version --change-type-id "ct-0attesnjqy2cx" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsSeParams.json
```

2. Ändern und speichern Sie die JSON-Datei mit den Ausführungsparametern. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":          "MariaDB-DMS-SE",
"VpcId":                "VPC_ID",
"Name":                 "Test SE",
"StackTemplateId":      "stm-pud4ghhkp7395n9bc",
"TimeoutInMinutes":     60,
"Parameters":   {
    "Description":      "DESCRIPTION",
    "EngineName":       "mariadb",
    "ServerName":       "mariadb.db.example.com",
    "Port":             "3306",
    "Username":         "DB-USER",
    "Password":         "DB-PW",}
    }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie „ CreateDmsSeRfc.json“ genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeRfc.json
```

4. Ändern und speichern Sie die CreateDmsSeRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0attesnjqy2cx",
"Title":                "MariaDB-DMS-Source-Endpoint"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die CreateDmsSeRfc Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeRfc.json --execution-
parameters file://CreateDmsSeParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

Bevor Sie den DMS-Endpunkt erstellen, stellen Sie sicher, dass Ihr Passwort keine Zeichen enthält, die nicht unterstützt werden. Weitere Informationen finden Sie im Benutzerhandbuch unter Quell- und Zielendpunkte erstellen.AWS Database Migration Service

Weitere Informationen finden Sie unter Quellen für die Datenmigration.

Informationen zu einem S3-Quellendpunkt finden Sie unterDMS-Quellendpunkt für S3: erstellen.

Informationen zu einem Mongo-DB-Quellendpunkt finden Sie unterDMS-Quellendpunkt für MongoDB: Erstellen.

DMS-Quellendpunkt für MongoDB: Erstellen

Einen DMS Mongo DB-Quellendpunkt mit der Konsole erstellen

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC

erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.

- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Einen DMS Mongo DB-Quellendpunkt mit der CLI erstellen

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm  --profile saml --region us-east-1 create-rfc --change-type-id
 "ct-2hxcllf1b4ey0" --change-type-version "1.0" --title 'DMS_Source_MongoDB'
 --description "DESCRIPTION"  --execution-parameters "{\"Description\":
\"DMS_MongoDB_Source_Endpoint\",\"VpcId\":\"VPC_ID\",\"Name\":\"DMS-Mongo-SE\",
\"StackTemplateId\":\"stm-pud4ghhkp7395n9bc\",\"TimeoutInMinutes\":60,\"Parameters\":
{\"DatabaseName\":\"mytestdb\",\"EngineName\":\"mongodb\",\"Port\":27017,\"ServerName
\":\"test.example.com\"}}"
```

VORLAGE ERSTELLEN:

1. Geben Sie die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei namens CreateDmsSeMongoParams .json aus.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2hxcllf1b4ey0"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateDmsSeMongoParams.json
   ```

2. Ändern und speichern Sie die JSON-Datei mit den Ausführungsparametern. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":          "MongoDB-DMS-SE",
"VpcId":                "VPC_ID",
"StackTemplateId":      "stm-pud4ghhkp7395n9bc",
"Name":                 "Test Mongo SE",
"TimeoutInMinutes":     60,
"Parameters":   {
    "Description":      "DESCRIPTION",
    "DatabaseName":       "mytestdb",
    "EngineName":       "mongodb",
    "ServerName":       "test.example.com",
    "Port":             "27017"
    }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie „ CreateDmsSeMongoRfc.json“ genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeMongoRfc.json
```

4. Ändern und speichern Sie die CreateDmsSeMongoRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2hxcllf1b4ey0",
"Title":                "DMS_Source_MongoDB"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die CreateDmsSeMongoRfc Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeMongoRfc.json --execution-
parameters file://CreateDmsSeMongoParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

> **Note**
>
> Sie können bis zu 50 Tags hinzufügen. Dazu müssen Sie jedoch die Ansicht Zusätzliche Konfiguration aktivieren.

AMS DMS kann Mongo oder einen beliebigen Relational Database Service (RDS) als Quellendpunkt verwenden. Informationen zu einem S3-Quellendpunkt finden Sie unter. DMS-Quellendpunkt für S3: erstellen

DMS-Quellendpunkt für S3: erstellen

Einen DMS S3-Quellendpunkt mit der Konsole erstellen

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.
3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht

„Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Einen DMS S3-Quellendpunkt mit der CLI erstellen

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile

dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws --profile saml --region us-east-1 amscm create-rfc --title "S3DMSSourceEndpoint" --
aws-account-id ACCOUNT-ID --change-type-id ct-2oxl37nphsrjz --change-type-version 1.0
 --execution-parameters "{\"Description\":\"TestS3DMS-SE\",\"VpcId\":\"VPC-ID\",\"Name
\":\"S3-DMS-SE\",\"Parameters\":{\"EngineName\":\"s3\",\"S3BucketName\":\"amzn-s3-
demo-bucket\",\"S3ExternalTableDefinition\":\"{\\\"TableCount\\\":\\\"1\\\",\\\"Tables
\\\":[{\\\"TableName\\\":\\\"employee\\\",\\\"TablePath\\\":\\\"hr/employee/\\\",\\
\"TableOwner\\\":\\\"hr\\\",\\\"TableColumns\\\":[{\\\"ColumnName\\\":\\\"Id\\\",\\
\"ColumnType\\\":\\\"INT8\\\",\\\"ColumnNullable\\\":\\\"false\\\",\\\"ColumnIsPk\\\":
\\\"true\\\"},{\\\"ColumnName\\\":\\\"LastName\\\",\\\"ColumnType\\\":\\\"STRING\\\",
\\\"ColumnLength\\\":\\\"20\\\"},{\\\"ColumnName\\\":\\\"FirstName\\\",\\\"ColumnType
\\\":\\\"STRING\\\",\\\"ColumnLength\\\":\\\"30\\\"},{\\\"ColumnName\\\":\\\"HireDate\
\\",\\\"ColumnType\\\":\\\"DATETIME\\\"},{\\\"ColumnName\\\":\\\"OfficeLocation\\\",\\
\"ColumnType\\\":\\\"STRING\\\",\\\"ColumnLength\\\":\\\"20\\\"}],\\\"TableColumnsTotal
\\\":\\\"5\\\"}]}\",\"S3ServiceAccessRoleArn\":\"arn:aws:iam::123456789101:role/ams-
ops-ct-authors-dms-s3-test-role\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-
pud4ghhkp7395n9bc\"}"
```

VORLAGE ERSTELLEN:

1. Geben Sie die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei mit dem Namen CreateDmsSe S3Params.json aus.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2oxl37nphsrjz" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsSeS3Params.json
   ```

2. Ändern und speichern Sie die JSON-Datei mit den Ausführungsparametern. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":          "TestS3DMS-SE",
"VpcId":                "VPC_ID",
"Name":                 "S3-DMS-SE",
"StackTemplateId":      "stm-pud4ghhkp7395n9bc",
"TimeoutInMinutes":     60,
"Parameters":   {
    "EngineName":               "s3",
    "S3BucketName":              "amzn-s3-demo-bucket",
    "S3ExternalTableDefinition":  "BUCKET-NAME",
    {"TableCount":                "1",
      "Tables":[{"TableName":"employee","TablePath":"hr/
employee/","TableOwner":"hr","TableColumns":
[{"ColumnName":"Id","ColumnType":"INT8","ColumnNullable":"false","ColumnIsPk":"true"},
{"ColumnName":"LastName","ColumnType":"STRING","ColumnLength":"20"},
{"ColumnName":"FirstName","ColumnType":"STRING","ColumnLength":"30"},
{"ColumnName":"HireDate","ColumnType":"DATETIME"},
{"ColumnName":"OfficeLocation","ColumnType":"STRING","ColumnLength":"20"}],"TableColumnsTot
    "S3ServiceAccessRoleArn":      "arn:aws:iam::123456789101:role/ams-ops-ct-
authors-dms-s3-test-role",
      }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie CreateDmsSe S3RFC.json genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeS3Rfc.json
```

4. Ändern und speichern Sie die Datei S3rfc.json. CreateDmsSe Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2oxl37nphsrjz",
"Title":                "DMS_Source_S3"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die CreateDmsSe S3RFC-Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeS3Rfc.json --execution-
parameters file://CreateDmsSeS3Params.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

> **Note**
>
> Sie können bis zu 50 Tags hinzufügen. Dazu müssen Sie jedoch die Ansicht Zusätzliche Konfiguration aktivieren.

AMS DMS kann S3 oder einen beliebigen Quellendpunkt des Relational Database Service (RDS) verwenden. Informationen zu einem Mongo-DB-Quellendpunkt finden Sie unter. DMS-Quellendpunkt für MongoDB: Erstellen

4: AWS DMS Zielendpunkt: Erstellen, für S3 erstellen

Sie können die AMS-Konsole verwenden oder API/CLI einen AMS DMS-Zielendpunkt für verschiedene Datenbanken erstellen. Wir stellen zwei Beispiele zur Verfügung.

DMS-Zielendpunkt: erstellen

AMS DMS kann S3 oder einen beliebigen Relational Database Service (RDS) mit MySQL, MariaDB, Oracle, Postgresql oder Microsoft SQL als Zielendpunkt verwenden.

Einen DMS-Zielendpunkt mit der Konsole erstellen

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite „RFC erstellen“: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.

2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.
3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.
4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Brechen Sie optional den RFC ab oder erstellen Sie eine Kopie davon mit den Optionen oben auf der Seite.

Einen DMS-Zielendpunkt mit der CLI erstellen

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.

2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *ID* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *ID*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-3gf8dolbo8x9p" --change-type-version "1.0" --title "TestDMSTargetEndpoint" --
execution-parameters "{\"Description\":\"TestTE\",\"VpcId\":\"VPC-ID\",\"Name\":
\"TE-NAME\",\"StackTemplateId\":\"stm-knghtmmgefafdq89u\",\"TimeoutInMinutes\":60,
\"Parameters\":{\"EngineName\":\"mysql\",\"Password\":\"testpw123\",\"Port\":\"3306\",
\"ServerName\":\"mytestdb.d5fga0rf2wpi.ap-southeast-2.rds.amazonaws.com\",\"Username\":
\"USERNAME\"}}"
```

VORLAGE ERSTELLEN:

1. Geben Sie die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei namens CreateDmsTeParams .json aus.

```
aws amscm get-change-type-version --change-type-id "ct-3gf8dolbo8x9p" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeParams.json
```

2. Ändern und speichern Sie die JSON-Datei mit den Ausführungsparametern. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":          "TestTE",
"VpcId":                "VPC_ID",
"StackTemplateId":      "stm-knghtmmgefafdq89u",
"Name":                 "TE-NAME",
"TimeoutInMinutes":     60,
"Parameters":   {
    "EngineName":       "mysql",
    "ServerName":       "sql.db.example.com",
    "Port":             "3306",
    "Username":         "DB-USER",
    "Password":         "DB-PW",}
    }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie „ CreateDmsTeRfc.json“ genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsTeRfc.json
```

4. Ändern und speichern Sie die CreateDmsTeRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3gf8dolbo8x9p",
"Title":                "DB-DMS-Target-Endpoint"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die CreateDmsTeRfc Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeRfc.json --execution-
parameters file://CreateDmsTeParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

- Dieser Änderungstyp ist jetzt in Version 2.0.
- AMS DMS kann S3 oder einen beliebigen Relational Database Service (RDS) mit MySQL, MariaDB, Oracle, Postgresql oder Microsoft SQL als Zielendpunkt verwenden. Informationen zu einem S3-Zielendpunkt finden Sie unter. DMS-Zielendpunkt für S3: erstellen
- Weitere Informationen finden Sie unter Ziele für die Datenmigration.
- Sie können bis zu 50 Tags hinzufügen. Dazu müssen Sie jedoch die Ansicht Zusätzliche Konfiguration aktivieren.

DMS-Zielendpunkt für S3: erstellen

Einen DMS S3-Zielendpunkt mit der Konsole erstellen

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite „RFC erstellen“: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC

erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.

- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Brechen Sie optional den RFC ab oder erstellen Sie eine Kopie davon mit den Optionen oben auf der Seite.

Einen DMS S3-Zielendpunkt mit der CLI erstellen

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-05muqzievnxk5" --change-type-version "1.0" --title "TestDMSTargetEndpointS3"
 --execution-parameters "{\"Description\":\"TestS3TE\",\"VpcId\":\"VPC-ID\",\"Name
\":\"S3TE-NAME\",\"StackTemplateId\":\"stm-knghtmmgefafdq89u\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"EngineName\":\"s3\",\"S3BucketName\":\"amzn-s3-demo-bucket\",
\"S3ServiceAccessRoleArn\":\"arn:aws:iam::123456789123:role/my-s3-role\"}}"
```

VORLAGE ERSTELLEN:

1. Gibt die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei aus. In diesem Beispiel wird sie CreateDmsTe S3Params.json genannt:

   ```
   aws amscm get-change-type-version --change-type-id "ct-05muqzievnxk5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeS3Params.json
   ```

2. Ändern und speichern Sie die Ausführungsparameter in der Datei S3Params.json. CreateDmsTe Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":          "TestS3DMS-TE",
"VpcId":                "VPC_ID",
"StackTemplateId":      "stm-knghtmmgefafdq89u",
"Name":                 "DMS-S3-TE",
"TimeoutInMinutes":     60,
"Parameters":   {
    "EngineName":       "s3",
    "S3BucketName":      "amzn-s3-demo-bucket",
    "S3ServiceAccessRoleArn":       "arn:aws:iam::123456789101:role/ams-ops-ct-
authors-dms-s3-test-role"
      }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie CreateDmsTe S3RFC.json genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsTeS3Rfc.json
```

4. Ändern und speichern Sie die Datei S3rfc.json. CreateDmsTe Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-05muqzievnxk5",
"Title":                "DMS_Target_S3"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die CreateDmsTe S3RFC-Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeS3Rfc.json --execution-
parameters file://CreateDmsTeS3Params.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

> **Note**
>
> Sie können bis zu 50 Tags hinzufügen. Dazu müssen Sie jedoch die Ansicht Zusätzliche Konfiguration aktivieren.

AMS bietet einen separaten Änderungstyp für die Erstellung eines Zielendpunkts für S3. Weitere Informationen finden Sie unter Verwenden von Amazon S3 als Ziel für AWS Database Migration Service und Zusätzliche Verbindungsattribute bei der Verwendung von Amazon S3 als Ziel für AWS DMS.

5: AWS DMS Replikationsaufgabe: Erstellen

Sie können die AMS-Konsole verwenden oder API/CLI um eine AWS DMS AMS-Replikationsaufgabe zu erstellen.

Erstellen Sie eine AWS DMS Replikationsaufgabe

Einen AWS DMS Replikations-Task mit der Konsole erstellen

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite „RFC erstellen“: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.

- Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Brechen Sie optional den RFC ab oder erstellen Sie eine Kopie davon mit den Optionen oben auf der Seite.

Erstellen einer AWS DMS Replikationsaufgabe mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-1d2fml15b9eth" --change-type-version "1.0" --title "TestDMSRepTask" --
execution-parameters "{\"Description\":\"TestRepTask\",\"VpcId\":\"VPC-ID\",\"Name
\":\"DMSRepTask\",\"Parameters\":{\"CdcStartTime\":\1533776569\"MigrationType\":
\"full-load\",\"ReplicationInstanceArn\":\"REP_INSTANCE_ARN\",\"SourceEndpointArn
\":\"SOURCE_ENDPOINT_ARN\",\"TableMappings\":\"{\\\"rules\\\": [{\\\"rule-type
\\\": \\\"selection\\\",\\\"rule-id\\\": \\\"1\\\",\\\"rule-name\\\": \\\"1\\
\",\\\"object-locator\\\": {\\\"schema-name\\\": \\\"Test\\\",\\\"table-name\\
\": \\\"%\\\"}, \\\"rule-action\\\": \\\"include\\\"}] }\",\"TargetEndpointArn
\":\"TARGET_ENDPOINT_ARN\"},\"StackTemplateId\":\"stm-eos7uq0usnmeggdet\",
\"TimeoutInMinutes\":60}"
```

VORLAGE ERSTELLEN:

1. Gibt die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei aus. In diesem Beispiel wird sie „ CreateDmsRtParams.json“ genannt:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1d2fml15b9eth" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRtParams.json
   ```

2. Ändern und speichern Sie die JSON-Datei mit den Ausführungsparametern. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"Description":          "DMSTestRepTask",
"VpcId":                "VPC_ID",
"StackTemplateId":      "stm-eos7uq0usnmeggdet",
"Name":                 "Test DMS RT",
"TimeoutInMinutes":     60,
"Parameters":   {
    "CdcStartTime":           "1533776569",
    "MigrationType":          "full-load",
    "ReplicationInstanceArn": "REP_INSTANCE_ARN",
    "SourceEndpointArn":      "SOURCE_ENDPOINT_ARN",
    "TargetEndpointArn":      "TARGET_ENDPOINT_ARN"
    "TableMappings":          {"rules": [{"rule-type": "selection","rule-id":
 "1","rule-name": "1","object-locator": {"schema-name": "Test","table-name": "%"},
 "rule-action": "include"}] }",
    }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie „ CreateDmsRtRfc.json“ genannt:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRtRfc.json
```

4. Ändern und speichern Sie die CreateDmsRtRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1d2fml15b9eth",
"Title":                "DMS-RI-Create-RFC"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die CreateDmsRtRfc Datei an:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRtRfc.json --execution-
parameters file://CreateDmsRtParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

Sie können eine AWS DMS Aufgabe erstellen, die drei verschiedene Arten von Änderungen oder Daten erfasst. Weitere Informationen finden Sie unter Arbeiten mit AWS DMS-Aufgaben, Erstellen einer Aufgabe und Erstellen von Aufgaben für die laufende Replikation mit AWS DMS.

AWS DMS Verwaltung

AWS DMS Beispiele für das Management.

Starten Sie die AWS DMS Replikationsaufgabe

Starten einer AWS DMS Replikationsaufgabe mit der Konsole

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können

Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.

3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.

5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Starten einer AWS DMS Replikationsaufgabe mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *ID* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *ID*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm create-rfc --change-type-id "ct-1yq7hhqse71yg" --change-type-version
 "1.0" --title "Start DMS Replication Task" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-StartDmsTask\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ReplicationTaskArn\":[\"TASK_ARN\"],\"StartReplicationTaskType\":[\"start-
replication\"],\"CdcStartPosition\":[\"\"],\"CdcStopPosition\":[\"\"]}}"
```

VORLAGE ERSTELLEN:

1. Gibt die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei aus. Dieses Beispiel nennt sie StartDmsRtParams .json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1yq7hhqse71yg" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StartDmsRtParams.json
   ```

2. Ändern und speichern Sie die JSON-Datei mit den Ausführungsparametern. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

   ```
   {
     "DocumentName": "AWSManagedServices-StartDmsTask",
     "Region": "us-east-1",
     "Parameters": {
   ```

```
      "ReplicationTaskArn": [
        "TASK_ARN"
      ],
      "StartReplicationTaskType": [
        "start-replication"
      ],
      "CdcStartPosition": [
        ""
      ],
      "CdcStopPosition": [
        ""
      ]
  }
}
```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie „ StartDmsRtRfc.json“ genannt:

```
aws amscm create-rfc --generate-cli-skeleton > StartDmsRtRfc.json
```

4. Ändern und speichern Sie die StartDmsRtRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
  "ChangeTypeId": "ct-1yq7hhqse71yg",
  "ChangeTypeVersion": "1.0",
  "Title": "Start DMS Replication Task"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die StartDmsRtRfc Datei an:

```
aws amscm create-rfc --cli-input-json file://StartDmsRtRfc.json --execution-
parameters file://StartDmsRtParams.json
```

Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

Sie können eine AWS DMS Replikationsaufgabe mit der AMS-Konsole oder der AMS-API/CLI starten. Weitere Informationen finden Sie unter Arbeiten mit AWS DMS-Aufgaben.

AWS DMS Replizierungsaufgabe beenden

Stoppen einer AWS DMS Replikationsaufgabe mit der Konsole

Screenshot dieses Änderungstyps in der AMS-Konsole:

So funktioniert es:

1. Navigieren Sie zur Seite RFC erstellen: Klicken Sie im linken Navigationsbereich der AMS-Konsole, um die RFCs Listenseite RFCszu öffnen, und klicken Sie dann auf RFC erstellen.
2. Wählen Sie in der Standardansicht „Änderungstypen durchsuchen“ einen beliebten Änderungstyp (CT) oder wählen Sie in der Ansicht „Nach Kategorie auswählen“ einen CT aus.
   - Nach Änderungstyp suchen: Sie können im Bereich Schnellerstellung auf ein beliebtes CT klicken, um sofort die Seite RFC ausführen zu öffnen. Beachten Sie, dass Sie mit Quick Create keine ältere CT-Version auswählen können.

     Verwenden Sie zum Sortieren CTs den Bereich Alle Änderungstypen in der Karten - oder Tabellenansicht. Wählen Sie in einer der Ansichten einen CT aus und klicken Sie dann auf RFC erstellen, um die Seite RFC ausführen zu öffnen. Falls zutreffend, wird neben der Schaltfläche „RFC erstellen“ die Option Mit älterer Version erstellen angezeigt.
   - Nach Kategorie auswählen: Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Daraufhin wird das Feld mit den CT-Details geöffnet. Dort können Sie gegebenenfalls die Option „Mit älterer Version erstellen“ auswählen. Klicken Sie auf RFC erstellen, um die Seite RFC ausführen zu öffnen.
3. Öffnen Sie auf der Seite RFC ausführen den Bereich CT-Name, um das Feld mit den CT-Details zu sehen. Ein Betreff ist erforderlich (dieser wird für Sie ausgefüllt, wenn Sie Ihr CT in der Ansicht „Änderungstypen durchsuchen“ auswählen). Öffnen Sie den Bereich Zusätzliche Konfiguration, um Informationen zum RFC hinzuzufügen.

   Verwenden Sie im Bereich Ausführungskonfiguration die verfügbaren Dropdownlisten oder geben Sie Werte für die erforderlichen Parameter ein. Um optionale Ausführungsparameter zu konfigurieren, öffnen Sie den Bereich Zusätzliche Konfiguration.

4. Wenn Sie fertig sind, klicken Sie auf Ausführen. Wenn keine Fehler vorliegen, wird die Seite mit dem RFC erfolgreich erstellt mit den übermittelten RFC-Details und der ersten Run-Ausgabe angezeigt.
5. Öffnen Sie den Bereich Run-Parameter, um die von Ihnen eingereichten Konfigurationen zu sehen. Aktualisieren Sie die Seite, um den RFC-Ausführungsstatus zu aktualisieren. Optional können Sie den RFC abbrechen oder eine Kopie davon mit den Optionen oben auf der Seite erstellen.

Stoppen einer AWS DMS Replikationsaufgabe mit der CLI

So funktioniert es:

1. Verwenden Sie entweder Inline Create (Sie geben einen `create-rfc` Befehl mit allen RFC- und Ausführungsparametern aus) oder Template Create (Sie erstellen zwei JSON-Dateien, eine für die RFC-Parameter und eine für die Ausführungsparameter) und geben Sie den `create-rfc` Befehl mit den beiden Dateien als Eingabe aus. Beide Methoden werden hier beschrieben.
2. Reichen Sie den `aws amscm submit-rfc --rfc-id` *`ID`* Befehl RFC: mit der zurückgegebenen RFC-ID ein.

   Überwachen Sie den RFC: -Befehl. `aws amscm get-rfc --rfc-id` *`ID`*

Verwenden Sie diesen Befehl, um die Version des Änderungstyps zu überprüfen:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> Sie können alle `CreateRfc` Parameter mit jedem RFC verwenden, unabhängig davon, ob sie Teil des Schemas für den Änderungstyp sind oder nicht. Um beispielsweise Benachrichtigungen zu erhalten, wenn sich der RFC-Status ändert, fügen Sie diese Zeile dem RFC-Parameter-Teil der Anfrage hinzu (nicht den Ausführungsparametern). `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` Eine Liste aller CreateRfc Parameter finden Sie in der AMS Change Management API-Referenz.

INLINE-ERSTELLUNG:

Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben), und senden Sie dann die zurückgegebene RFC-ID. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
aws amscm create-rfc --change-type-id "ct-1vd3y4ygbqmfk" --change-type-version
 "1.0" --title "Stop DMS Replication Task" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-StopDmsTask\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ReplicationTaskArn\":[\"TASK_ARN\"]}}"
```

VORLAGE ERSTELLEN:

1. Gibt die Ausführungsparameter für diesen Änderungstyp in eine JSON-Datei aus. Dieses Beispiel nennt sie StopDmsRtParams .json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1vd3y4ygbqmfk" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StopDmsRtParams.json
   ```

2. Ändern und speichern Sie die JSON-Datei mit den Ausführungsparametern. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

   ```
   {
     "DocumentName": "AWSManagedServices-StopDmsTask",
     "Region": "us-east-1",
     "Parameters": {
         "ReplicationTaskArn": [
           "TASK_ARN"
         ]
     }
   }
   ```

3. Geben Sie die JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. In diesem Beispiel wird sie „ StopDmsRtRfc.json“ genannt:

   ```
   aws amscm create-rfc --generate-cli-skeleton > StopDmsRtRfc.json
   ```

4. Ändern und speichern Sie die StopDmsRtRfc JSON-Datei. Sie können den Inhalt beispielsweise durch etwas Ähnliches ersetzen:

```
{
  "ChangeTypeId": "ct-1vd3y4ygbqmfk",
  "ChangeTypeVersion": "1.0",
  "Title": "Stop DMS Replication Task"
}
```

5. Erstellen Sie den RFC und geben Sie die Ausführungsparameterdatei und die StopDmsRtRfc Datei an:

```
aws amscm create-rfc --cli-input-json file://StopDmsRtRfc.json --execution-
parameters file://StopDmsRtParams.json
```

   Sie erhalten die ID des neuen RFC in der Antwort und können sie verwenden, um den RFC zu senden und zu überwachen. Bis Sie ihn abschicken, verbleibt der RFC im Bearbeitungszustand und startet nicht.

Tipps

Sie können eine DMS-Replikationsaufgabe mithilfe der AMS-Konsole oder der AMS-API/CLI beenden. Weitere Informationen finden Sie unter Arbeiten mit AWS DMS-Aufgaben.

Datenbankimport (DB) nach AMS RDS für Microsoft SQL Server

> **Note**
>
> Die AMS-Endpunkte API/CLI (amscm und amsskms) befinden sich in der AWS-Region Nord-Virginia,. `us-east-1` Je nachdem, wie Ihre Authentifizierung eingerichtet ist und in welcher AWS-Region sich Ihr Konto und Ihre Ressourcen befinden, müssen Sie `--region us-east-1` bei der Ausgabe von Befehlen möglicherweise zusätzliche Informationen hinzufügen. Möglicherweise müssen Sie auch angeben`--profile saml`, ob dies Ihre Authentifizierungsmethode ist.

Der DB-Import in AMS RDS for SQL Server basiert auf AMS-Änderungstypen (CTs), die als Änderungsanforderungen (RFCs) eingereicht wurden, und verwendet die Amazon RDS-API-Parameter als Eingabe. MicroSoft SQL Server ist ein relationales Datenbankmanagementsystem

(RDBMS). Weitere Informationen finden Sie auch unter Amazon Relational Database Service (Amazon RDS) und rds oder Amazon RDS API-Referenz.

> **Note**
>
> Stellen Sie sicher, dass jeder RFC erfolgreich abgeschlossen wurde, bevor Sie mit dem nächsten Schritt fortfahren.

Importschritte auf hoher Ebene:

1. Sichern Sie Ihre lokale MS SQL-Quelldatenbank in einer Bak-Datei (Backup)
2. Kopieren Sie die .bak-Datei in den Transit-Bucket (verschlüsselt) von Amazon Simple Storage Service (S3)
3. Importieren Sie die .bak-Datei in eine neue Datenbank auf Ihrer Amazon RDS MS SQL-Zielinstanz

Voraussetzungen:

- MS SQL RDS-Stack im AMS
- RDS-Stack mit Wiederherstellungsoption (`SQLSERVER_BACKUP_RESTORE`)
- S3-Bucket versenden
- IAM-Rolle mit Bucket-Zugriff, sodass Amazon RDS die Rolle übernehmen kann
- Eine EC2 Instance, auf der MS SQL Management Studio zur Verwaltung des RDS installiert ist (kann eine Workstation vor Ort sein)

Einrichten

Führen Sie diese Aufgaben aus, um den Importvorgang zu starten.

1. Senden Sie einen RFC, um einen RDS-Stack mithilfe von Deployment | Advanced Stack Components | RDS database stack | Create (ct-2z60dyvto9g6c) zu erstellen. Verwenden Sie nicht den Ziel-DB-Namen (*`RDSDBName`*Parameter) in der Erstellungsanforderung, die Ziel-DB wird während des Imports erstellt. Stellen Sie sicher, dass genügend Speicherplatz (`RDSAllocatedStorage`Parameter) vorhanden ist. Einzelheiten dazu finden Sie im AMS Change Management Guide RDS DB Stack | Create.

2. Senden Sie mithilfe von Deployment | Advanced Stack Components | S3 storage | Create (ct-1a68ck03fn98r) einen RFC, um den Transit-S3-Bucket zu erstellen (falls dieser noch nicht existiert). Einzelheiten dazu finden Sie im AMS Change Management Guide S3 Storage | Create.
3. Reichen Sie den RFC Management | Other | Other | Update (ct-1e1xtak34nx76) ein, um den mit diesen Details zu implementieren: `customer_rds_s3_role`

   In der Konsole:

   - Betreff: „Implementieren Sie die Implementierung unter diesem Konto, um den MS SQL Server-Datenbankimport zu unterstützen. `customer_rds_s3_role`
   - Name des Transit-S3-Buckets:*BUCKET_NAME*.
   - Kontaktinformationen:*EMAIL*.

   Mit einer ImportDbParams .json-Datei für die CLI:

   ```
   {
          "Comment": "{"Transit S3 bucket name":"BUCKET_NAME"}",
          "Priority": "High"
        }
   }
   ```

4. Senden Sie einen RFC für Management | Other | Other | Update und fordern Sie AMS auf, die `SQLSERVER_BACKUP_RESTORE` Option auf den in Schritt 1 erstellten RDS zu setzen (verwenden Sie in dieser Anfrage die Stack-ID aus der Ausgabe von Schritt 1 und die `customer_rds_s3_role` IAM-Rolle in dieser Anfrage).
5. Reichen Sie einen RFC ein, um eine EC2 Instanz zu erstellen (Sie können jede vorhandene EC2 oder lokale Workstation/Instanz verwenden) und installieren Sie Microsoft SQL Management Studio auf der Instanz.

Die Datenbank importieren

Gehen Sie folgendermaßen vor, um die Datenbank (DB) zu importieren.

1. Sichern Sie Ihre lokale Quelldatenbank mit MS SQL Native Backup and Restore (siehe Support für systemeigene Sicherung und Wiederherstellung in SQL Server). Als Ergebnis der Ausführung dieses Vorgangs sollten Sie über eine Bak-Datei (Sicherungsdatei) verfügen.

2. Laden Sie die .bak-Datei mithilfe der AWS S3-CLI oder der AWS S3-Konsole in einen vorhandenen Transit-S3-Bucket hoch. Informationen zu Transit-S3-Buckets finden Sie unter Daten durch Verschlüsselung schützen.
3. Importieren Sie die .bak-Datei in eine neue Datenbank auf Ihrer Ziel-RDS für SQL Server MS SQL-Instance (Einzelheiten zu den Typen finden Sie unter Amazon RDS for MySQL MySQL-Instance-Typen):

   a. Melden Sie sich bei der EC2 Instance (lokale Workstation) an und öffnen Sie MS SQL Management Studio

   b. Connect zu der in Schritt #1 als Voraussetzung erstellten RDS-Zielinstanz her. Gehen Sie wie folgt vor, um eine Verbindung herzustellen: Verbindung zu einer DB-Instance herstellen, auf der die Microsoft SQL Server Database Engine ausgeführt wird

   c. Starten Sie den Import- (Wiederherstellungs-) Job mit einer neuen SQL-Abfrage (Structured Query Language) (Einzelheiten zu SQL-Abfragen finden Sie unter Einführung in SQL). Der Name der Zieldatenbank muss neu sein (verwenden Sie nicht denselben Namen wie die Datenbank, die Sie zuvor erstellt haben). Beispiel ohne Verschlüsselung:

   ```
   exec msdb.dbo.rds_restore_database
           @restore_db_name=TARGET_DB_NAME,

           @s3_arn_to_restore_from='arn:aws:s3:::BUCKET_NAME/FILENAME.bak';
   ```

   d. Überprüfen Sie regelmäßig den Status des Importauftrags, indem Sie diese Abfrage in einem separaten Fenster ausführen:

   ```
   exec msdb.dbo.rds_task_status;
   ```

   Wenn sich der Status in Fehlgeschlagen ändert, suchen Sie in der Meldung nach den Fehlerdetails.

Bereinigen

Nachdem Sie die Datenbank importiert haben, möchten Sie möglicherweise nicht benötigte Ressourcen entfernen. Gehen Sie dazu wie folgt vor.

1. Löschen Sie die Sicherungsdatei (.bak) aus dem S3-Bucket. Sie können dazu die S3-Konsole verwenden. Informationen zum CLI-Befehl zum Löschen eines Objekts aus einem S3-Bucket finden Sie unter rm in der AWS CLI Command Reference.
2. Löschen Sie den S3-Bucket, wenn Sie ihn nicht verwenden möchten. Eine Anleitung dazu finden Sie unter Stapel löschen.
3. Wenn Sie nicht vorhaben, MS SQL-Importe durchzuführen, reichen Sie einen RFC für Management | Other | Other | Update (ct-0xdawir96cy7k) ein und fordern Sie AMS auf, die IAM-Rolle zu löschen. `customer_rds_s3_role`

Tier-and-Tie-App-Bereitstellungen in AMS

Bei einer Tier-and-Tie-Bereitstellung erstellen, konfigurieren und implementieren Sie die Ressourcen eines Stacks unabhängig voneinander RFCs, wobei Sie separate Ressourcen verwenden und die IDs einzelnen Stack-Komponenten im weiteren Verlauf verwenden, um sie miteinander zu verknüpfen.

Um beispielsweise eine (redundante) Website mit hoher Verfügbarkeit hinter einem Load Balancer und einer Datenbank mithilfe eines Tier-and-Tie-Ansatzes bereitzustellen, reichen Sie RFCs eine Datenbank und einen Load Balancer und zwei EC2 Instances oder eine Auto Scaling Scaling-Gruppe ein und konfigurieren Sie die EC2 Instances oder Auto Scaling Scaling-Gruppe mit der ID des ELB, den Sie erstellt haben.

Nach der Bereitstellung der Ressourcen können Sie eine Änderung bei der Erstellung der Sicherheitsgruppe einreichen, damit die Ressourcen mit der Datenbank kommunizieren können. Einzelheiten zum Erstellen von Sicherheitsgruppen finden Sie unter Sicherheitsgruppe erstellen.

Full-Stack-App-Bereitstellungen in AMS

Bei einer Full-Stack-Bereitstellung reichen Sie einen RFC mit einem CT ein, der alles, was Sie benötigen, auf einmal erstellt und konfiguriert. Um beispielsweise die eben beschriebene Hochverfügbarkeitswebsite (EC2 Instanzen, Load Balancer und Datenbank) bereitzustellen, würden Sie einen CT verwenden, der zusammen eine Auto Scaling Scaling-Gruppe, einen Load Balancer, eine Datenbank und die Sicherheitsgruppeneinstellungen erstellt und konfiguriert, die erforderlich sind, damit alle Instances als Stack funktionieren. Im Folgenden werden Beispiele für zwei AMS beschrieben CTs , die dies tun.

- High Availability Two-Tier Stack (ct-06mjngx5flwto): Mit diesem Änderungstyp können Sie einen Stack erstellen und eine Auto Scaling Scaling-Gruppe, eine RDS-gestützte Datenbank, einen

Load Balancer sowie eine Anwendung und Konfiguration konfigurieren. CodeDeploy Beachten Sie, dass der Load Balancer nicht als Stufe betrachtet wird, da er als Netzwerk-Appliance von mehreren Anwendungen gemeinsam genutzt wird und die Funktionen ebenfalls als Appliance betrachtet werden. CodeDeploy Darüber hinaus erstellt er eine CodeDeploy Bereitstellungsgruppe (mit dem Namen, den Sie der CodeDeploy Anwendung geben), die zur Bereitstellung Ihrer Anwendungen verwendet werden kann. Sicherheitsgruppeneinstellungen, damit die Ressourcen zusammenarbeiten können, werden automatisch erstellt.

- Hochverfügbarer One-Tier-Stack (ct-09t6q7j9v5hrn): Mit diesem Änderungstyp können Sie einen Stack erstellen und eine Auto Scaling Scaling-Gruppe sowie einen Application Load Balancer konfigurieren. Sicherheitsgruppeneinstellungen, die es ermöglichen, dass die Ressourcen zusammen funktionieren, werden automatisch erstellt.

Arbeiten mit Provisioning-Änderungstypen () CTs

AMS ist für Ihre verwaltete Infrastruktur verantwortlich. Um Änderungen vornehmen zu können, müssen Sie einen RFC mit der richtigen CT-Klassifizierung (Kategorie, Unterkategorie, Artikel und Vorgang) einreichen. In diesem Abschnitt wird beschrieben, wie Sie herausfinden CTs, ob ein CT Ihren Anforderungen entspricht, und wie Sie ein neues CT anfordern können, falls es keine gibt.

Finden Sie heraus, ob ein vorhandenes CT Ihren Anforderungen entspricht

Sobald Sie festgelegt haben, was Sie mit AMS implementieren möchten, besteht der nächste Schritt darin, die vorhandenen CloudFormation Vorlagen CTs und Vorlagen zu untersuchen, um festzustellen, ob bereits eine Lösung existiert.

Wenn Sie einen RFC erstellen, müssen Sie den CT angeben. Sie können die AWS Management Console oder die AMS API/CLI verwenden. Beispiele für die Verwendung von beiden werden im Folgenden beschrieben.

Sie können die Konsole oder die verwenden API/CLI , um nach einer Änderungstyp-ID (CT) oder Version zu suchen. Es gibt zwei Methoden: entweder eine Suche oder die Auswahl der Klassifizierung. Für beide Auswahltypen können Sie die Suche sortieren, indem Sie entweder „Am häufigsten verwendet“, „Zuletzt verwendet“ oder „Alphabetisch“ auswählen.

YouTube Video: Wie erstelle ich einen RFC mit der AWS Managed Services CLI und wo finde ich das CT-Schema?

Gehen Sie in der AMS-Konsole auf der Seite RFCs-> RFC erstellen wie folgt vor:

- Wenn „Nach Änderungstyp suchen“ ausgewählt ist (Standardeinstellung), können Sie entweder:
  - Verwenden Sie den Bereich Schnellerstellung, um aus den beliebtesten AMS-Programmen auszuwählen CTs. Klicken Sie auf ein Label und die Seite RFC ausführen wird geöffnet, auf der die Option Betreff automatisch für Sie ausgefüllt wird. Füllen Sie die verbleibenden Optionen nach Bedarf aus und klicken Sie auf Ausführen, um den RFC einzureichen.
  - Oder scrollen Sie nach unten zum Bereich Alle Änderungstypen und beginnen Sie, einen CT-Namen in das Optionsfeld einzugeben. Sie müssen nicht den genauen oder vollständigen Namen des Änderungstyps haben. Sie können auch anhand der ID des Änderungstyps, der Klassifizierung oder des Ausführungsmodus (automatisiert oder manuell) nach einem CT suchen, indem Sie die entsprechenden Wörter eingeben.

    Wenn die standardmäßige Kartenansicht ausgewählt ist, werden während der Eingabe passende CT-Karten angezeigt. Wählen Sie eine Karte aus und klicken Sie auf RFC erstellen. Wählen Sie bei ausgewählter Tabellenansicht das entsprechende CT aus und klicken Sie auf RFC erstellen. Bei beiden Methoden wird die Seite RFC ausführen geöffnet.
- Klicken Sie alternativ oben auf der Seite auf Nach Kategorie auswählen, um eine Reihe von Drop-down-Optionsfeldern zu öffnen, um die Auswahlmöglichkeiten für den Änderungstyp zu erkunden.
- Wählen Sie eine Kategorie, eine Unterkategorie, einen Artikel und einen Vorgang aus. Das Informationsfeld für diesen Änderungstyp wird angezeigt. Unten auf der Seite wird ein Bereich angezeigt.
- Wenn Sie bereit sind, drücken Sie die EINGABETASTE. Daraufhin wird eine Liste der passenden Änderungstypen angezeigt.
- Wählen Sie einen Änderungstyp aus der Liste aus. Das Informationsfeld für diesen Änderungstyp wird unten auf der Seite angezeigt.
- Wenn Sie den richtigen Änderungstyp gefunden haben, wählen Sie Create RFC.

> **Note**
>
> Die AMS-CLI muss installiert sein, damit diese Befehle funktionieren. Um die AMS-API oder CLI zu installieren, rufen Sie die Seite Entwicklerressourcen der AMS-Konsole auf. Referenzmaterial zur AMS CM API oder AMS SKMS API finden Sie im Abschnitt AMS-Informationsressourcen im Benutzerhandbuch. Möglicherweise müssen Sie eine `--profile` Option für die Authentifizierung hinzufügen, `aws amsskms` *`ams-cli-command`* `--profile SAML` z. B.. Möglicherweise müssen Sie die `--region` Option auch hinzufügen,

da allen AMS-Befehlen beispielsweise us-east-1 ausgeht. `aws amscm ams-cli-command --region=us-east-1`

> **Note**
>
> Die AMS-Endpunkte API/CLI (amscm und amsskms) befinden sich in der AWS-Region Nord-Virginia,. `us-east-1` Je nachdem, wie Ihre Authentifizierung eingerichtet ist und in welcher AWS-Region sich Ihr Konto und Ihre Ressourcen befinden, müssen Sie `--region us-east-1` bei der Ausgabe von Befehlen möglicherweise zusätzliche Informationen hinzufügen. Möglicherweise müssen Sie auch angeben`--profile saml`, ob dies Ihre Authentifizierungsmethode ist.

So suchen Sie mit der AMS CM API (siehe ListChangeTypeClassificationSummaries) oder CLI nach einem Änderungstyp:

Sie können einen Filter oder eine Abfrage für die Suche verwenden. Der ListChangeTypeClassificationSummaries Vorgang verfügt über Filteroptionen für `CategorySubcategory`,`Item`, und`Operation`, aber die Werte müssen exakt mit den vorhandenen Werten übereinstimmen. Für flexiblere Ergebnisse bei der Verwendung der CLI können Sie die `--query` Option verwenden.

Ändern Sie die Typfilterung mit der AMS CM API/CLI

Attribut	Zulässige Werte	Gültige/Standardbedingung	Hinweise
ChangeTypeId	Jede Zeichenfolge, die a darstellt ChangeTypeId (Zum Beispiel: ct-abc123xyz7890)	Gleichheitszeichen	Informationen zum Änderungstyp finden Sie in der Referenz zum Änderungstyp. IDs Informationen zum Änderungstyp IDs finden Sie unter Suche nach einem

Attribut	Zulässige Werte	Gültige/Standardbedingung	Hinweise
			Änderungstyp oder CSIO.
Kategorie	Beliebiger Text in freier Form	Enthält	Reguläre Ausdrücke in jedem einzelnen Feld werden nicht unterstützt. Suche ohne Berücksichtigung von Groß- und Kleinschreibung
Unterkategorie			
Item			
Operation			

1. Hier sind einige Beispiele für Klassifizierungen nach der Art der Angebotsänderung:

   Der folgende Befehl listet alle Kategorien von Änderungstypen auf.

   ```
   aws amscm list-change-type-categories
   ```

   Der folgende Befehl listet die Unterkategorien auf, die zu einer bestimmten Kategorie gehören.

   ```
   aws amscm list-change-type-subcategories --category CATEGORY
   ```

   Der folgende Befehl listet die Elemente auf, die zu einer bestimmten Kategorie und Unterkategorie gehören.

   ```
   aws amscm list-change-type-items --category CATEGORY --subcategory SUBCATEGORY
   ```

2. Hier sind einige Beispiele für die Suche nach Änderungstypen mit CLI-Abfragen:

   Der folgende Befehl durchsucht CT-Klassifikationszusammenfassungen nach solchen, die „S3" im Elementnamen enthalten, und erstellt eine Ausgabe der Kategorie-, Unterkategorie-, Element-, Vorgangs- und Änderungstyp-ID in Tabellenform.

   ```
   aws amscm list-change-type-classification-summaries --query
    "ChangeTypeClassificationSummaries [?contains(Item, 'S3')].
   [Category,Subcategory,Item,Operation,ChangeTypeId]" --output table
   ```

```
+-------------------------------------------------------------+
|               ListChangeTypeClassificationSummaries         |
+----------+-------------------------+--+------+--------------+
|Deployment|Advanced Stack Components|S3|Create|ct-1a68ck03fn98r|
+----------+-------------------------+--+------+--------------+
```

3. Anschließend können Sie die Änderungstyp-ID verwenden, um das CT-Schema abzurufen und die Parameter zu untersuchen. Mit dem folgenden Befehl wird das Schema in eine JSON-Datei mit dem Namen CreateS3Params.schema.json ausgegeben.

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateS3Params.schema.json
```

   Informationen zur Verwendung von CLI-Abfragen finden Sie unter So filtern Sie die Ausgabe mit der Option --query und in der Referenz zur Abfragesprache, JMESPath Spezifikation.

4. Nachdem Sie die Änderungstyp-ID erhalten haben, empfehlen wir, die Version für den Änderungstyp zu überprüfen, um sicherzustellen, dass es sich um die neueste Version handelt. Verwenden Sie diesen Befehl, um die Version für einen bestimmten Änderungstyp zu finden:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CHANGE_TYPE_ID
```

   Um die `AutomationStatus` für einen bestimmten Änderungstyp zu finden, führen Sie diesen Befehl aus:

```
aws amscm --profile saml get-change-type-version --change-type-id CHANGE_TYPE_ID --
query "ChangeTypeVersion.{AutomationStatus:AutomationStatus.Name}"
```

   Führen Sie diesen Befehl aus, um `ExpectedExecutionDurationInMinutes` nach dem für einen bestimmten Änderungstyp zu suchen:

```
aws amscm --profile saml get-change-type-version --change-type-id ct-14027q0sjyt1h
 --query "ChangeTypeVersion.{ExpectedDuration:ExpectedExecutionDurationInMinutes}"
```

Wenn Sie ein CT gefunden haben, das Sie für geeignet halten, schauen Sie sich das zugehörige JSON-Schema mit den Ausführungsparametern an, um zu erfahren, ob es für Ihren Anwendungsfall geeignet ist.

Verwenden Sie diesen Befehl, um ein CT-Schema in eine nach dem CT benannte JSON-Datei auszugeben. In diesem Beispiel wird das Create S3-Speicherschema ausgegeben:

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
--query "ChangeTypeVersion.ExecutionInputSchema" --output text >
CreateBucketParams.json
```

Schauen wir uns genauer an, was dieses Schema bietet.

S3-Bucket: Schema erstellen

```
{
  "$schema": "http://json-schema.org/draft-04/sch
ema#",
"name": "Create S3 Storage
"description": "Use to create an Amazon Simple
 Storage Service stack.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "The description of the
 stack.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to create the S3
 Bucket in, in the form vpc-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Required value: stm-s2b72
beb000000000.",
      "type": "string",
      "enum": ["stm-s2b72beb000000000"]
    },
    "Name":{
```

Das Schema beginnt mit der CT („Beschreibung“), die Ihnen sagt, wofür das Schema gedacht ist. In diesem Fall, um einen S3-Speicher-Stack zu erstellen.

Als Nächstes haben Sie die erforderlichen und optionale n Eigenschaften, die Sie angeben können. Standardw erte für Eigenschaften sind angegeben. Die erforderlichen Eigenschaften sind am Ende des Schemas aufgeführt.

In diesem StackTemplateId Bereich sehen Sie, dass es für diesen CT und dieses Schema eine spezifische Stack-Vorlage gibt, deren ID ein erforderl icher Eigenschaftswert ist.

```
      "description": "The name of the stack to
create.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value
pairs) for the stack.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 7
    },
    "TimeoutInMinutes": {
      "description": "The amount of time, in minutes,
to allow for creation of the stack.",
      "type": "number",
      "minimum": 0,
      "maximum": 60
    },
    "Parameters": {
```

Das Schema ermöglicht es Ihnen, den Stapel, den Sie gerade erstellen, für interne Buchhaltungszwecke zu taggen. Darüber hinaus erfordern einige Optionen, wie Backup, die Tags key:Backup und value:True. Ausführliche Informationen finden Sie unter Tagging Your Amazon EC2 Resources.

```
      "description": "Specifications for the
 stack.",
      "type": "object",
      "properties": {
        "AccessControl": {
          "description": "The canned (predefined)
 access control list (ACL) to assign to the bucket.",
          "type": "string",
          "enum": [
            "Private",
            "PublicRead",
            "AuthenticatedRead",
            "BucketOwnerRead"
          ]
        },
        "BucketName": {
          "description": "A name for the bucket.
 The bucket name must contain only lowercase letters,
 numbers, periods (.), and hyphens (-).",
          "type": "string",
          "pattern": "^[a-z0-9]([-.a-z0-9]+)[a-z
0-9]$",
          "minLength": 3,
          "maxLength": 63
        }
      },
      "additionalProperties": false,
      "required": [
        "AccessControl",
        "BucketName"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Im Abschnitt „Parameter“ des CT-JSON-Schemas geben Sie die Ausführungsparameter an.

Für dieses Schema BucketName sind nur die ACL und die Ausführungsparamet er erforderlich.

Fordern Sie ein neues CT an

Nachdem Sie das Schema geprüft haben, stellen Sie möglicherweise fest, dass es nicht genügend Parameter bietet, um die gewünschte Bereitstellung zu erstellen. Wenn das der Fall ist, überprüfen Sie die vorhandenen CloudFormation Vorlagen, um eine zu finden, die Ihren Vorstellungen am nächsten kommt. Sobald Sie wissen, welche zusätzlichen Parameter Sie benötigen, reichen Sie die Felder Management | Other | Other | Create CT ein.

> **Note**
>
> Alle anderen | Sonstige Erstellung und Aktualisierung CTs werden von einem AMS-Mitarbeiter bearbeitet, der sich mit Ihnen in Verbindung setzen wird, um das neue CT zu besprechen.

Um eine Anfrage für ein neues CT einzureichen, greifen Sie wie gewohnt auf die AMS-Konsole zu AWS Management Consoleund folgen Sie dann diesen Schritten.

1. Klicken Sie in der linken Navigationsleiste auf RFCs.

   Die RFCs Dashboard-Seite wird geöffnet.
2. Klicken Sie auf Create.

   Die Seite Änderungsantrag erstellen wird geöffnet.
3. Wählen Sie in der Dropdownliste Kategorie die Option Verwaltung und für die Unterkategorie und den Artikel die Option Andere aus. Wählen Sie für den Vorgang die Option Erstellen aus. Der RFC muss genehmigt werden, bevor er implementiert werden kann.
4. Geben Sie Informationen ein, warum Sie das CT benötigen, zum Beispiel: Fordern Sie ein modifiziertes Create S3-Storage-CT an ACLs, das auf dem vorhandenen Create S3-Storage-CT basiert und ein benutzerdefiniertes ermöglicht. Dies sollte zu einem neuen CT führen: Bereitstellung | Erweiterte Stack-Komponenten | S3-Speicher | Benutzerdefinierte S3-ACL erstellen. Dieses neue CT könnte öffentlich sein.
5. Klicken Sie auf Submit.

   Ihr RFC wird im RFC-Dashboard angezeigt.

Testen Sie das neue CT

Sobald AWS Managed Services diesen neuen CT erstellt hat, testen Sie ihn, indem Sie einen RFC zusammen mit ihm einreichen. Wenn Sie mit AMS zusammengearbeitet haben, um das neue CT vorab zu genehmigen, können Sie einfach einer standardmäßigen RFC-Einreichung folgen und auf das Ergebnis achten (Einzelheiten zur Einreichung finden Sie unter Erstellen und RFCs Einreichen eines RFC). Wenn das neue CT nicht vorab genehmigt wurde (Sie möchten sichergehen, dass es nie ohne ausdrückliche Genehmigung ausgeführt wird), müssen Sie seine Implementierung jedes Mal, wenn Sie es ausführen möchten, mit AMS besprechen.

Schnelle Starts

Themen

Mit einer Kombination von AMS-Änderungstypen können Sie komplexe Aufgaben erledigen.

Sie können das AMS Change Management System verwenden, um AMS Resource Scheduler für ein Multi-Account-Landingzone-Konto (MALZ) oder für ein Single-Account-Landingzone-Konto (SALZ) einzurichten. Der Prozess ist unterschiedlich. Auch für Dateiübertragungen und kontoübergreifende Schnappschüsse.

AMS Resource Scheduler Schnellstart

Verwenden Sie diese Schnellstartanleitung, um AMS Resource Scheduler, einen tagbasierten Instance-Scheduler, zu implementieren, um Kosten in AMS Advanced zu sparen.

Der AMS Resource Scheduler basiert auf dem AWS Instance Scheduler.

Terminologie von AMS Resource Scheduler

Bevor Sie beginnen, sollten Sie mit der Terminologie von AMS Resource Scheduler vertraut sein:

- Zeitraum: Jeder Zeitplan muss mindestens einen Zeitraum enthalten, der die Zeit (en) definiert, zu der die Instance ausgeführt werden soll. Ein Zeitplan kann mehr als einen Zeitraum enthalten. Wenn in einem Zeitplan mehr als ein Zeitraum verwendet wird, wendet der Resource Scheduler die entsprechende Startaktion an, wenn mindestens eine der Periodenregeln zutrifft.
- Zeitzone: Eine Liste der akzeptablen Zeitzonenwerte, die in dem DefaultTimezoneParameter verwendet werden können, auf den später verwiesen wird, finden Sie in der Spalte TZ der Liste der Zeitzonen der TZ-Datenbank.
- Ruhezustand: Wenn dieser Wert auf „true“ gesetzt ist, werden EC2 Instances, die für den Ruhezustand aktiviert sind und die Anforderungen für den Ruhezustand erfüllen, in den Ruhezustand versetzt (). suspend-to-disk Prüfen Sie in der EC2 Konsole, ob Ihre Instances für den Ruhezustand aktiviert sind. Verwenden Sie den Ruhezustand für gestoppte EC2 Amazon-Instances, auf denen Amazon Linux ausgeführt wird.

- erzwungen: Wenn dieser Wert auf true gesetzt ist, stoppt der Resource Scheduler basierend auf dem definierten Zeitplan eine laufende Ressource, wenn sie außerhalb der Laufzeit manuell gestartet wird, und startet eine Ressource, wenn sie während der Laufzeit manuell gestoppt wird.
- retain_running: Wenn auf true gesetzt, wird verhindert, dass der Resource Scheduler eine Instanz am Ende einer Laufzeit stoppt, wenn die Instanz vor Beginn der Laufzeit manuell gestartet wurde. Wenn beispielsweise eine Instanz mit einem konfigurierten Zeitraum, der zwischen 9 Uhr und 17 Uhr läuft, manuell vor 9 Uhr gestartet wird, stoppt der Resource Scheduler die Instanz nicht um 17 Uhr.
- ssm-maintenance-window: Fügt einem Zeitplan ein AWS Systems Manager Wartungsfenster als Laufzeit hinzu. Wenn Sie den Namen eines Wartungsfensters angeben, das in demselben Konto und derselben AWS-Region wie Ihr bereitgestellter Stack existiert, um Ihre EC2 Amazon-Instances zu planen, startet der Resource Scheduler die Instance vor dem Start des Wartungsfensters und stoppt die Instance am Ende des Wartungsfensters, sofern keine andere Laufzeit vorgibt, dass die Instance ausgeführt werden soll, und wenn das Wartungsereignis abgeschlossen ist.

  Der Resource Scheduler bestimmt anhand der AWS Lambda Häufigkeit, die Sie bei der Erstkonfiguration angegeben haben, wie lange es dauert, bis zum Wartungsfenster Ihre Instanz gestartet werden muss. Wenn Sie den AWS CloudFormation Frequenzparameter auf 10 Minuten oder weniger setzen, startet der Resource Scheduler die Instanz 10 Minuten vor dem Wartungsfenster. Wenn Sie die Häufigkeit auf mehr als 10 Minuten festlegen, startet der Resource Scheduler die Instanz in derselben Anzahl von Minuten wie in der von Ihnen angegebenen Häufigkeit. Wenn Sie beispielsweise die Häufigkeit des Systems Manager Manager-Wartungsfensters auf 30 Minuten festlegen, startet der Resource Scheduler die Instanz 30 Minuten vor dem Wartungsfenster.

  Weitere Informationen finden Sie unter AWS Systems Manager Wartungsfenster.
- override-status: Überschreibt vorübergehend die vom Resource Scheduler konfigurierten Start- und Stoppaktionen. Wenn Sie das Feld auf Running setzen, startet der Resource Scheduler die entsprechende Instanz, stoppt sie jedoch nicht. Die Instanz wird ausgeführt, bis Sie sie manuell beenden. Wenn Sie den Override-Status auf Beendet setzen, stoppt der Resource Scheduler die entsprechende Instanz, startet sie aber nicht. Die Instanz wird erst ausgeführt, wenn Sie sie manuell starten.

Implementierung von AMS Resource Scheduler

Gehen Sie folgendermaßen vor, um eine AMS Resource Scheduler-Lösung bereitzustellen.

1. Reichen Sie ein Deployment ein | AMS Resource Scheduler | Solution | Deploy (ct-0ywnhc8e5k9z5) RFC und geben Sie die folgenden Parameter an:

   - SchedulingActive: Ja, um die Ressourcenplanung zu aktivieren, Nein, um sie zu deaktivieren. Die Standardeinstellung ist Ja.
   - ScheduledServices: Geben Sie eine durch Kommas getrennte Liste von Diensten ein, für die Ressourcen geplant werden sollen. Zu den gültigen Werten gehört eine Kombination aus Autoscaling, ec2 und rds. Die Standardeinstellung ist Autoscaling, ec2, rds.
   - TagName: Der Name des Tag-Schlüssels, der Ressourcenplanschemas mit Serviceressourcen verknüpft. Die Standardeinstellung ist Schedule.

     > **Note**
     >
     > Ihre Resource Scheduler-Bereitstellung funktioniert nur mit Ressourcen, die über dieses Tag verfügen.

   - DefaultTimezone: Der Name der Zeitzone im Format US/Pacific, die als Standardzeitzone verwendet werden soll. Die Standardeinstellung ist UTC.
2. Nachdem Sie eine Bestätigung erhalten haben, dass der RFC in Schritt 1 erfolgreich ausgeführt wurde, können Sie den Änderungstyp Zeitraum | Hinzufügen einreichen.
3. Reichen Sie abschließend einen RFC ein, um dem Zeitraum, der in Schritt zwei erstellt wurde, einen Zeitplan hinzuzufügen. Verwenden Sie den Änderungstyp Zeitplan | Hinzufügen.

Implementierung und Verwendung von AMS Resource Scheduler FAQs

Häufig gestellte Fragen zu AMS Resource Scheduler.

F: Was passiert, wenn ich den Ruhezustand aktiviere, die EC2 Instance ihn aber nicht unterstützt?

A: Hibernation speichert den Inhalt aus dem Instance-Speicher (RAM) auf Ihrem Amazon Elastic Block Store (Amazon EBS) -Root-Volume. Wenn dieses Feld auf true gesetzt ist, werden Instances in den Ruhezustand versetzt, wenn Resource Scheduler sie stoppt.

Wenn Sie Resource Scheduler so einstellen, dass er den Ruhezustand verwendet, Ihre Instances jedoch nicht für den Ruhezustand aktiviert sind oder die Voraussetzungen für den Ruhezustand nicht erfüllen, protokolliert Resource Scheduler eine Warnung und die Instanzen werden ohne Ruhezustand gestoppt. Weitere Informationen finden Sie unter Instance Hibernate Your Instance.

F: Was passiert, wenn ich sowohl override_status als auch enforced festlege?

A: Wenn Sie override_status auf running und enforced auf true setzen (verhindert, dass eine Instanz außerhalb einer Laufzeit manuell gestartet wird), stoppt Resource Scheduler die Instanz.

Wenn Sie override_status auf stopped und enforced auf true setzen (verhindert, dass eine Instanz während einer Laufzeit manuell gestoppt wird), startet der Resource Scheduler die Instanz neu.

> **Note**
>
> Wenn enforced auf False gesetzt ist, wird das konfigurierte Override-Verhalten angewendet.

F: Wie deaktiviere oder aktiviere ich nach der Bereitstellung des AMS Resource Scheduler den Resource Scheduler in meinem Konto?

A: Um AMS Resource Scheduler zu deaktivieren oder zu aktivieren:

- Zur Deaktivierung: Erstellen Sie mit State | Disable einen RFC. Stellen Sie sicher, dass das auf DISABLE SchedulerStategesetzt ist
- Um zu aktivieren: Erstellen Sie mit State | Enable einen RFC. Stellen Sie sicher, dass Sie das auf ENABLE SchedulerStatesetzen

F: Was passiert, wenn der AMS Resource Scheduler-Zeitraum in mein Wartungsfenster für Patches fällt?

A: Resource Scheduler arbeitet auf der Grundlage der konfigurierten Zeitpläne. Wenn es so konfiguriert ist, dass eine Instanz angehalten wird, während das Patchen läuft, dann stoppt es die Instanz, es sei denn, das Patch-Fenster wird dem Zeitplan als Zeitraum hinzugefügt, bevor das Patchen beginnt. Mit anderen Worten, Resource Scheduler startet keine gestoppten Instanzen automatisch zum Patchen, es sei denn, ein bestimmter Zeitraum ist konfiguriert. Um Konflikte mit Ihrem Wartungsfenster für das Patchen zu vermeiden, fügen Sie dem Resource Scheduler-Zeitplan das für das Patchen zugewiesene Zeitfenster als Zeitraum hinzu. Um einen Zeitraum zu einem bestehenden Zeitplan hinzuzufügen, erstellen Sie mit Period | Add einen RFC.

F: Kann ich in meinem Konto mehr als einen Zeitplan einrichten, wenn ich einen anderen Zeitplan für verschiedene EC2 Instanzen benötige?

A: Ja, Sie können mehrere Zeitpläne erstellen. Jeder Zeitplan kann je nach Anforderung mehrere Perioden haben. Wenn AMS Resource Scheduler im Konto aktiviert ist, wird ein Tag-Schlüssel

konfiguriert. Wenn der Tag-Schlüssel beispielsweise „Schedule“ lautet, kann der Tag-Wert auf der Grundlage verschiedener Zeitpläne unterschiedlich sein, was dem Zeitplannamen von AMS Resource Scheduler entspricht. Um einen neuen Zeitplan hinzuzufügen, können Sie mit dem Änderungstyp Verwaltung | AMS Resource Scheduler | Zeitplan | Hinzufügen (ct-2bxelbn765ive) einen RFC erstellen, siehe Zeitplan | Hinzufügen.

F: Wo finde ich all die verschiedenen Änderungstypen, die für AMS Resource Scheduler unterstützt werden?

A: AMS verfügt über Resource Scheduler-Änderungstypen, um den AMS Resource Scheduler für Ihr Konto bereitzustellen, ihn zu aktivieren oder zu deaktivieren, Zeitpläne und Zeiträume zu definieren, hinzuzufügen, zu aktualisieren und zu löschen und die Zeitpläne und Perioden zu beschreiben (um eine detaillierte Beschreibung zu erhalten).

Einrichtung kontenübergreifender Backups (innerhalb der Region)

AWS Backup unterstützt die Möglichkeit, Snapshots von einem Konto auf ein anderes innerhalb derselben AWS-Region zu kopieren, sofern sich die beiden Konten innerhalb derselben AWS-Organisation befinden. In der AMS Advanced Multi-Account-Landing landing zone (MALZ) können Sie beispielsweise mit diesem Schnellstart eine kontoübergreifende Snapshot-Kopie innerhalb derselben AWS-Region einrichten.

Weitere Informationen finden Sie unter AWS Backup und AWS Organizations bieten kontoübergreifende Backup-Funktion

Sie kopieren Snapshots kontoübergreifend für die Notfallwiederherstellung (DR). Aus Datenschutzgründen müssen Sie möglicherweise Snapshots innerhalb derselben AWS-Region, aber außerhalb der Kontogrenzen aufbewahren.

Überblick:

Im Allgemeinen sind dies die Schritte für kontenübergreifende Backups innerhalb von AMS:

- Erstellen Sie ein Zielkonto für das Hosten von Backups in der AWS-Region, in der Ihre AMS-Landingzone gehostet wird (Schritt 1)
- Erstellen Sie einen KMS-Schlüssel zum Verschlüsseln von Backups im Zielkonto (Schritt 3)
- Erstellen Sie einen Backup-Tresor im Zielkonto derselben Region wie Ihre AMS Advanced-Landezone (Schritt 4)

- Aktivieren Sie die kontoübergreifende Einstellung in Ihrem Verwaltungskonto (Schritt 5)
- Erstellen oder ändern Sie den Backup-Plan und die Regel (n) für das Quellkonto (Schritt 6)

> **Note**
>
> Stellen Sie sicher, dass sich sowohl das Quell- als auch das Zielkonto in derselben Region befinden. Wenn Sie Ihre Backups regionsübergreifend kopieren möchten, wenden Sie sich an Ihre CA oder CSDM.

So aktivieren und richten Sie kontoübergreifende Backups ein:

1. Erstellen Sie ein Zielkonto zum Hosten von Backups. Wenn Sie bereits über ein solches Konto verfügen, können Sie diesen Schritt überspringen. Um das Konto zu erstellen, reichen Sie einen RFC von Ihrem Management Payer-Konto aus ein. Verwenden Sie dazu den Änderungstyp Deployment | Managed landing zone | Management account | Create application account (with VPC) (ct-1zdasmc2ewzrs).
2. [Optional] Wenn Ressourcen oder Snapshots im Quellkonto (z. B. Prod) verschlüsselt sind, teilen Sie den für die Verschlüsselung verwendeten KMS-Schlüssel mit dem Zielkonto. Senden Sie dazu einen RFC mit dem Änderungstyp Management | Advanced Stack Components | KMS-Schlüssel | Update (ct-3ovo7px2vsa6n).
3. Erstellen Sie im Zielkonto einen KMS-Schlüssel, der für die Backup Vault-Verschlüsselung verwendet werden soll. Senden Sie dazu einen RFC mit dem Typ Deployment | Advanced Stack Components | KMS-Schlüssel | Create (auto) change type (ct-1d84keiri1jhg).
4. Erstellen Sie im Zielkonto einen Backup-Tresor mit dem zuvor erstellten Schlüssel. AWS-Backup-Tresore können mithilfe des automatischen Änderungstyps CFN-Ingest, Deployment | Ingestion | Stack from CloudFormation Template | Create (ct-36cn2avfrrj9v), erstellt werden. In derselben Anfrage muss die Tresorzugriffsrichtlinie geändert werden, um den Quellkonten den Zugriff auf den Tresor zu ermöglichen. Hier ist ein Beispiel für eine Richtlinie:

   CloudFormation Beispielvorlage für einen Backup-Tresor:

```
{
  "Description": "Test infrastructure",
  "Resources": {
  "BackupVaultForTesting": {
    "Type": "AWS::Backup::BackupVault",
```

```
    "Properties": {
      "BackupVaultName": "backup-vault-for-test",
      "EncryptionKeyArn" : "arn:aws:kms:us-east-2:123456789012:key/227d8xxx-
aefx-44ex-a09x-b90c487b4xxx",
        "AccessPolicy" : {
          "Version": "2012-10-17",
          "Statement": [
            {
              "Sid": "AllowSrcAccountPermissionsToCopy",
              "Effect": "Allow",
              "Action": "backup:CopyIntoBackupVault",
              "Resource": "*",
              "Principal": {
                "AWS": ["arn:aws:iam::987654321098:root"]
              }
            }
          ]
        }
      }
    }
  }
}
```

5. Aktivieren Sie von Ihrem Management Payer-Konto aus die kontoübergreifende Sicherung. Senden Sie dazu einen RFC mit dem Änderungstyp Management | AWS Backup | Backup-Plan | Kontenübergreifendes Kopieren aktivieren (Verwaltungskonto) (ct-2yja7ihh30ply).
6. Erstellen Sie abschließend von dem Quellkonto aus, aus dem die Backups stammen, die Regel oder Regeln des Backup-Plans, die die Backups für das kontoübergreifende Kopieren von Snapshots regeln. Reichen Sie dazu einen RFC mit dem Typ Deployment | AWS Backup | Backup plan | Create change type (ct-2hyozbpa0sx0m) ein. Wenn Sie einen vorhandenen Backup-Plan aktualisieren müssen, reichen Sie über den Änderungstyp Management | Other | Other | Update (ct-0xdawir96cy7k) einen RFC mit diesen Informationen ein:

   1. Der Name des Sicherungsplans sowie der Name der Regel, die aktualisiert werden sollen.
   2. Der ARN für den destination/ICE Konto-Backup-Tresor.
   3. Die Aufbewahrung, für die days/months Sie die Snapshots im ICE-Zieltresor aufbewahren möchten.

Tutorials

Themen

In den folgenden Tutorials werden die Schritte zur Erstellung eines zweistufigen Stacks mit Hochverfügbarkeit (ct-06mjngx5flwto), zur Verwendung der CLI und zur Verwendung der Konsole sowie zur Bereitstellung einer Linux- oder RHEL Amazon Auto Scaling Scaling-Gruppe (ASG) beschrieben. EC2 Es folgt jeweils ein ähnliches tier-and-tie Tutorial (eines für die Konsole und eines für die CLI), das separate CTs, in einer solchen Reihenfolge erstellte Ressourcen verwendet, dass Sie Ressourcen bei ihrer Erstellung miteinander verknüpfen können.

Beschreibungen aller CT-Optionen, einschließlich, ChangeTypeId finden Sie in der Referenz zumanagedservices/latest/ctref/Change Type.

Konsolen-Tutorial: Zweistufiger Stack mit hoher Verfügbarkeit (Linux/RHEL)

In diesem Abschnitt wird beschrieben, wie Sie mithilfe der AMS-Konsole eine WordPress Hochverfügbarkeits-Site (HA) in einer AMS-Umgebung bereitstellen.

> **Note**
>
> Diese exemplarische Vorgehensweise für die Bereitstellung wurde in AMZN Linux- und RHEL-Umgebungen getestet.

Zusammenfassung der Aufgaben und der erforderlichen Aufgaben: RFCs

1. Infrastruktur erstellen (zweistufiger HA-Stack)
2. Erstellen Sie einen S3-Bucket für Anwendungen CodeDeploy
3. Erstellen Sie das WordPress Anwendungspaket und laden Sie es in den S3-Bucket hoch

4. Stellen Sie die Anwendung bereit mit CodeDeploy
5. Greifen Sie auf die WordPress Site zu und melden Sie sich an, um die Bereitstellung zu validieren
6. Reißen Sie die Bereitstellung ab

Beschreibungen aller CT-Optionen, einschließlich ChangeTypeId, finden Sie in der AMS Change Type Reference.

Bevor Sie beginnen

Deployment | Advanced Stack Components | High Availability Two Tier Stack | Create CT erstellt eine Auto Scaling Scaling-Gruppe, einen Load Balancer, eine Datenbank sowie einen CodeDeploy Anwendungsnamen und eine Bereitstellungsgruppe (mit demselben Namen, den Sie der Anwendung geben). Weitere Informationen finden CodeDeploy Sie unter Was ist CodeDeploy?

In dieser exemplarischen Vorgehensweise wird ein Two-Tier-Stack-RFC für hohe Verfügbarkeit verwendet, in dem auch beschrieben wird, wie ein WordPress Paket erstellt wird, das CodeDeploy bereitgestellt werden kann. `UserData`

Die im Beispiel `UserData` gezeigte Methode ruft Instanz-Metadaten wie Instanz-ID, Region usw. aus einer laufenden Instanz ab, indem der Instanz-Metadatendienst abgefragt wird, der unter EC2 http://169.254.169.254/latest/meta-data/ verfügbar ist. Diese Zeile im Benutzerdatenskript:`REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed 's/[a-z]$//')`, ruft den Namen der Verfügbarkeitszone aus dem Metadatendienst in die $REGION-Variable für unsere unterstützten Regionen ab und vervollständigt damit die URL für den S3-Bucket, in den der Agent heruntergeladen wird. CodeDeploy Die 169.254.169.254 IP ist nur innerhalb der VPC routingfähig (alle können den Service abfragen). VPCs Informationen zum Service finden Sie unter Instanz-Metadaten und Benutzerdaten. Beachten Sie auch, dass Skripte, UserData die als eingegeben wurden, als „root“ -Benutzer ausgeführt werden und den Befehl „sudo“ nicht verwenden müssen.

In dieser exemplarischen Vorgehensweise werden die folgenden Parameter auf dem Standardwert belassen (siehe Abbildung):

- Auto Scaling Scaling-Gruppe:`Cooldown=300, DesiredCapacity=2, EBSOptimized=false, HealthCheckGracePeriod=600, IAMInstanceProfile=customer-mc-ec2-instance-profile, InstanceDetailedMonitoring=true, InstanceRootVolumeIops=0, InstanceRootVolumeType=standard, InstanceType=m3.medium,`

`MaxInstances=2, MinInstances=2, ScaleDownPolicyCooldown=300, ScaleDownPolicyEvaluationPeriods=4, ScaleDownPolicyPeriod=60, ScaleDownPolicyScalingAdjustment=-1, ScaleDownPolicyStatistic=Average, ScaleDownPolicyThreshold=35, ScaleMetricName=CPUUtilization, ScaleUpPolicyCooldown=60, ScaleUpPolicyEvaluationPeriods=2, ScaleUpPolicyPeriod=60, ScaleUpPolicyScalingAdjustment=2, ScaleUpPolicyStatistic=Average, ScaleUpPolicyThreshold=75`.

- Load Balancer:`HealthCheckInterval=30, HealthCheckTimeout=5`.
- Datenbank:`BackupRetentionPeriod=7, Backups=true, InstanceType=db.m3.medium, IOPS=0, MultiAZ=true, PreferredBackupWindow=22:00-23:00, PreferredMaintenanceWindow=wed:03:32-wed:04:02, StorageEncrypted=false, StorageEncryptionKey="", StorageType=gp2`.
- Anwendung:`DeploymentConfigName=CodeDeployDefault.OneAtATime`.

Variable Parameter:

Die Konsole bietet eine ASAP-Option für die Startzeit, und in dieser exemplarischen Vorgehensweise wird empfohlen, diese zu verwenden. ASAP veranlasst, dass der RFC ausgeführt wird, sobald die Genehmigungen bestanden wurden.

> **Note**
>
> Es gibt viele Parameter, die Sie möglicherweise anders als in der Abbildung festlegen möchten. Die im Beispiel gezeigten Werte für diese Parameter wurden getestet, sind aber möglicherweise nicht für Sie geeignet. In den Beispielen werden nur die erforderlichen Werte angezeigt. Die *`replaceable`* Schriftwerte sollten geändert werden, da sie für Ihr Konto spezifisch sind.

Erstellen Sie die Infrastruktur

Bei diesem Verfahren wird der zweistufige Hochverfügbarkeits-Stack-CT verwendet, gefolgt vom Create S3-Speicher-CT.

Wenn Sie die folgenden Daten sammeln, bevor Sie beginnen, kann die Bereitstellung schneller vonstattengehen.

DIE ERFORDERLICHEN DATEN SIND GESTAPELT:

- AutoScalingGroup:
  - UserData: Dieser Wert wird in diesem Tutorial bereitgestellt. Er enthält Befehle zum Einrichten der Ressource für den Agenten CodeDeploy und zum Starten des CodeDeploy Agenten.
  - AMI-ID: Dieser Wert bestimmt das Betriebssystem der EC2 Instances, die Ihre Auto Scaling Scaling-Gruppe (ASG) hochfahren wird. Wählen Sie in Ihrem Konto ein AMI aus, das mit „Kunde-“ beginnt und das von Ihnen gewünschte Betriebssystem hat. Suchen Sie AMI IDs in der AMS-Konsole VPCs -> VPCs Detailseite. Diese exemplarische Vorgehensweise bezieht sich auf die ASGs Konfiguration für die Verwendung eines Amazon Linux- oder RHEL-AMI.
- Datenbank:
  - Diese Parameter DBEngine, EngineVersion, und LicenseModelsollten entsprechend Ihrer Situation eingestellt werden, obwohl die im Beispiel gezeigten Werte getestet wurden. Das Tutorial verwendet jeweils diese Werte:*MySQL*,*8.0.16*,*general-public-license*.
  - Diese Parameter, DBNameMasterUserPassword, und MasterUsernamesind für die Bereitstellung des Anwendungspakets erforderlich. Das Tutorial verwendet jeweils diese Werte:*wordpressDB*,*p4ssw0rd*,*admin*. Beachten DBName Sie, dass dieser nur alphanumerische Zeichen enthalten darf.
  - Wenn Sie das MasterUsernamefür die RDS-Datenbank eingeben, wird es im Klartext angezeigt. Melden Sie sich daher so schnell wie möglich bei der Datenbank an und ändern Sie das Passwort, um Ihre Sicherheit zu gewährleisten.
  - Verwenden Sie für RDSSubnetIDs zwei private Subnetze. Geben Sie sie nacheinander ein und drücken Sie jeweils die Eingabetaste. Suchen Sie das Subnetz IDs mit der Referenz For the AMS SKMS API auf der Registerkarte Berichte in der AWS Artifact Console. Operation (CLI: list-subnet-summaries) oder auf der Seite AMS-Konsole VPCs -> VPC-Details.
- LoadBalancer:
  - Setzen Sie diesen Parameter Public auf true, da das Tutorial öffentliche ELB-Subnetze verwendet.
  - ELBSubnetIds: Verwenden Sie zwei öffentliche Subnetze. Geben Sie sie nacheinander ein und drücken Sie jeweils die Eingabetaste. Suchen Sie das Subnetz IDs mit der Referenz For the AMS SKMS API auf der Registerkarte Berichte in der AWS Artifact Console. Operation (CLI: list-subnet-summaries) oder auf der Seite AMS-Konsole VPCs -> VPC-Details.
- Anwendung: Der ApplicationNameWert legt den Anwendungsnamen und den Namen der CodeDeploy Bereitstellungsgruppe fest. CodeDeploy Sie verwenden ihn, um Ihre Anwendung bereitzustellen. Es muss für das Konto eindeutig sein. In der CodeDeploy Konsole können Sie in

Ihrem Konto nach CodeDeploy Namen suchen. Das Beispiel verwendet, *WordPress* aber wenn Sie diesen Wert verwenden, stellen Sie sicher, dass er nicht bereits verwendet wird.

1. Starten Sie den Hochverfügbarkeits-Stack.

   a. Wählen Sie auf der Seite RFC erstellen aus der Liste die Kategorie Deployment, die Unterkategorie Standard-Stacks, die Elemente High Availability Two-Tier-Stack und Operation Create aus.

   b. WICHTIG: Wählen Sie „Erweitert“ und legen Sie die Werte wie abgebildet fest.

      Sie müssen nur Werte für Optionen mit Sternchen (*) eingeben. Die getesteten Werte werden im Beispiel gezeigt. Sie können die nicht erforderlichen leeren Optionen leer lassen.

   c. Für den Abschnitt mit der RFC-Beschreibung:

      ```
      Subject: WP-HA-2-Tier-RFC
      ```

   d. Legen Sie für den Abschnitt Ressourceninformationen die Parameter für AutoScalingGroup, Datenbank LoadBalancer, Anwendung und Tags fest.

      Der Tag-Schlüssel "AppName" dient außerdem dazu, dass Sie in der EC2 Konsole einfach nach den ASG-Instanzen suchen können. Sie können diesen Tag-Schlüssel „Name“ oder einen beliebigen anderen Schlüsselnamen nennen. Beachten Sie, dass Sie bis zu 50 Tags hinzufügen können.

      ```
      UserData:
          #!/bin/bash
          REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/
       | sed 's/[a-z]$//')
          yum -y install ruby httpd
          chkconfig httpd on
          service httpd start
          touch /var/www/html/status
          cd /tmp
          curl -O https://aws-codedeploy-$REGION.s3.amazonaws.com/latest/install
          chmod +x ./install
          ./install auto
          chkconfig codedeploy-agent on
          service codedeploy-agent start
      AmiId:              AMI-ID
      Description:        WP-HA-2-Tier-Stack
      ```

```
Database:
    LicenseModel:     general-public-license (USE RADIO BUTTON)
    EngineVersion:    8.0.16
    DBEngine:         MySQL
    RDSSubnetIds:     PRIVATE_AZ1 PRIVATE_AZ2 (ENTER ONE AT A TIME PRESSING
 "ENTER" AFTER EACH)
    MasterUserPassword:  p4ssw0rd
    MasterUsername:      admin
    DBName:              wordpressDB

LoadBalancer:
    Public:              true (USE RADIO BUTTON)
    ELBSubnetIds:        PUBLIC_AZ1 PUBLIC_AZ2

Application:
    ApplicationName:     WordPress

Tags:
    Name:                WP-Rhel-Stack
```

e. Wenn Sie fertig sind, klicken Sie auf Senden.

2. Melden Sie sich bei der Datenbank an, die Sie erstellt haben, und ändern Sie das Passwort.
3. Starten Sie einen S3-Bucket-Stack.

   Wenn Sie die folgenden Daten sammeln, bevor Sie beginnen, kann die Bereitstellung schneller vonstattengehen.

   ERFORDERLICHER DATEN-S3-BUCKET:

   - VPC-ID: Dieser Wert bestimmt, wo sich Ihr S3-Bucket befinden wird. Suchen Sie VPC IDs mit der Referenz For the AMS SKMS API auf der Registerkarte Berichte in der AWS Artifact Console. Operation (CLI: list-vpc-summaries) oder auf der AMS-Konsolenseite. VPCs
   - BucketName: Dieser Wert legt den Namen des S3-Buckets fest. Sie verwenden ihn, um Ihr Anwendungspaket hochzuladen. Er muss in der gesamten Region des Kontos eindeutig sein und darf keine Großbuchstaben enthalten. Die Angabe Ihrer Konto-ID als Teil von BucketName ist keine Voraussetzung, erleichtert jedoch die spätere Identifizierung des Buckets. Um zu sehen, welche S3-Bucket-Namen in dem Konto vorhanden sind, rufen Sie die Amazon S3 S3-Konsole für Ihr Konto auf.

a. Wählen Sie auf der Seite Create RFC die Kategorie Deployment, die Unterkategorie Advanced Stack Components, den Artikel S3 storage und Operation Create aus der RFC CT-Auswahlliste aus.

b. Behalten Sie die Standardoption Basic bei und legen Sie die Werte wie gezeigt fest.

```
Subject:              S3-Bucket-WP-HA-RFC
Description:          S3BucketForWordPressBundles
BucketName:           ACCOUNT_ID-BUCKET_NAME
AccessControl:        Private
VpcId:                VPC_ID
Name:                 S3-Bucket-WP-HA-Stack
TimeoutInMinutes:     60
```

c. Wenn Sie fertig sind, klicken Sie auf Senden. Der mit diesem Änderungstyp bereitgestellte Bucket ermöglicht vollen read/write Zugriff auf das gesamte Konto.

Anwendung erstellen, hochladen und bereitstellen

Erstellen Sie zunächst ein WordPress Anwendungspaket und verwenden Sie dann das, CodeDeploy CTs um die Anwendung zu erstellen und bereitzustellen.

1. Laden Sie die Dateien herunter WordPress, extrahieren Sie sie und erstellen Sie eine. Verzeichnis /scripts.

   Linux-Befehl:

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows: `https://github.com/WordPress/WordPress/archive/master.zip` In ein Browserfenster einfügen und die Zip-Datei herunterladen.

   Erstellen Sie ein temporäres Verzeichnis, in dem das Paket zusammengestellt werden soll.

   Linux:

   ```
   mkdir /tmp/WordPress
   ```

Windows: Erstellen Sie ein "WordPress" Verzeichnis. Sie werden den Verzeichnispfad später verwenden.

2. Extrahieren Sie die WordPress Quelle in das Verzeichnis WordPress "" und erstellen Sie ein. Verzeichnis /scripts.

   Linux:

   ```
   unzip master.zip -d /tmp/WordPress_Temp
   cp -paf /tmp/WordPress_Temp/WordPress-master/* /tmp/WordPress
   rm -rf /tmp/WordPress_Temp
   rm -f master
   cd /tmp/WordPress
   mkdir scripts
   ```

   Windows: Gehen Sie in das Verzeichnis "WordPress", das Sie erstellt haben, und erstellen Sie dort ein Verzeichnis „scripts“.

   Wenn Sie sich in einer Windows-Umgebung befinden, stellen Sie sicher, dass Sie den Unterbrechungstyp für die Skriptdateien auf Unix (LF) setzen. In Notepad ++ ist dies eine Option unten rechts im Fenster.

3. Erstellen Sie die Datei CodeDeploy appspec.yml im WordPress Verzeichnis (wenn Sie das Beispiel kopieren, überprüfen Sie den Einzug, jedes Leerzeichen zählt). WICHTIG: Stellen Sie sicher, dass der Quellpfad für das Kopieren der WordPress Dateien (in diesem Fall in Ihrem WordPress Verzeichnis) an das erwartete Ziel (/) korrekt ist. var/www/html/WordPress Im Beispiel befindet sich die Datei appspec.yml im Verzeichnis mit den WordPress Dateien, sodass nur „/“ benötigt wird. Auch wenn Sie ein RHEL-AMI für Ihre Auto Scaling Scaling-Gruppe verwendet haben, sollten Sie die Zeile „os: linux“ unverändert lassen. Beispiel für eine appspec.yml-Datei:

   ```
   version: 0.0
   os: linux
   files:
     - source: /
       destination: /var/www/html/WordPress
   hooks:
     BeforeInstall:
       - location: scripts/install_dependencies.sh
         timeout: 300
         runas: root
   ```

```
  AfterInstall:
    - location: scripts/config_wordpress.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

4. Erstellen Sie Bash-Dateiskripts in der. WordPress Verzeichnis /scripts.

   Erstellen Sie zunächst `config_wordpress.sh` mit dem folgenden Inhalt (wenn Sie möchten, können Sie die Datei wp-config.php direkt bearbeiten).

   > **Note**
   >
   > *DBName*Ersetzen Sie durch den Wert, der im HA Stack-RFC angegeben ist (z. B.`wordpress`).
   > *DB_MasterUsername*Ersetzen Sie durch den `MasterUsername` Wert, der im HA Stack-RFC angegeben ist (z. B.`admin`).
   > *DB_MasterUserPassword*Ersetzen Sie durch den `MasterUserPassword` Wert, der im HA Stack-RFC angegeben ist (z. B.`p4ssw0rd`).
   > *DB_ENDPOINT*Ersetzen Sie es in den Ausführungsausgaben des HA Stack-RFC durch den DNS-Namen des Endpunkts (z. B.`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Sie finden dies mit der GetRfcOperation (CLI: get-rfc --rfc-id RFC_ID) oder auf der RFC-Detailseite der AMS-Konsole für den HA Stack-RFC, den Sie zuvor eingereicht haben.

```
#!/bin/bash
chmod -R 755 /var/www/html/WordPress
cp /var/www/html/WordPress/wp-config-sample.php /var/www/html/WordPress/wp-
config.php
cd /var/www/html/WordPress
sed -i "s/database_name_here/DBName/g" wp-config.php
sed -i "s/username_here/DB_MasterUsername/g" wp-config.php
```

```
sed -i "s/password_here/DB_MasterUserPassword/g" wp-config.php
sed -i "s/localhost/DB_ENDPOINT/g" wp-config.php
```

5. Erstellen Sie im selben Verzeichnis mit dem folgenden Inhalt: `install_dependencies.sh`

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS wird beim Start als Teil der Benutzerdaten installiert, damit Integritätsprüfungen von Anfang an funktionieren.

6. Erstellen Sie im selben Verzeichnis `start_server.sh` mit dem folgenden Inhalt:

   - Verwenden Sie für Amazon Linux-Instances Folgendes:

   ```
   #!/bin/bash
   service httpd start
   ```

   - Verwenden Sie für RHEL-Instances Folgendes (die zusätzlichen Befehle sind Richtlinien, die es SELINUX ermöglichen, sie zu akzeptieren): WordPress

   ```
   #!/bin/bash
   setsebool -P  httpd_can_network_connect_db 1
   setsebool -P  httpd_can_network_connect 1
   chcon -t httpd_sys_rw_content_t /var/www/html/WordPress/wp-content -R
   restorecon -Rv /var/www/html
   service httpd start
   ```

7. Erstellen Sie im selben Verzeichnis `stop_server.sh` mit dem folgenden Inhalt:

```
#!/bin/bash
service httpd stop
```

8. Erstellen Sie das Zip-Bundle.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

Windows: Gehen Sie in Ihr "WordPress" -Verzeichnis, wählen Sie alle Dateien aus und erstellen Sie eine ZIP-Datei. Achten Sie darauf, sie wordpress.zip zu nennen.

1. Laden Sie das Anwendungspaket in den S3-Bucket hoch

   Das Paket muss vorhanden sein, um den Stack weiter bereitstellen zu können.

   Sie haben automatisch Zugriff auf jede S3-Bucket-Instanz, die Sie erstellen. Sie können über Ihre Bastions (siehe Zugreifen auf Instances) oder über die S3-Konsole darauf zugreifen und das CodeDeploy Paket mit drag-and-drop der Datei hochladen oder indem Sie die Datei aufrufen und auswählen.

   Sie können auch den folgenden Befehl in einem Shell-Fenster verwenden. Vergewissern Sie sich, dass Sie den richtigen Pfad zur ZIP-Datei haben:

   ```
   aws s3 cp wordpress/wordpress.zip s3://BUCKET_NAME/
   ```

2. Stellen Sie das WordPress CodeDeploy Anwendungspaket bereit

   ERFORDERLICHER DATENCODE/ANWENDUNGSBEREITSTELLUNG BEREITSTELLEN:

   - CodeDeployApplicationName: Der Name, den Sie der CodeDeploy Anwendung gegeben haben.
   - CodeDeployGroupName: Da sowohl die CodeDeploy Anwendung als auch die Gruppe anhand des Namens erstellt wurden, den Sie der CodeDeploy Anwendung im HA-Stack-RFC gegeben haben, ist dies derselbe Name wie der CodeDeployApplicationName.
   - S3Bucket: Der Name, den Sie dem S3-Bucket gegeben haben.
   - S3 BundleType und S3Key: Diese sind Teil des von Ihnen bereitgestellten WordPress Anwendungspakets.
   - VpcId: Die entsprechende VPC.

a. Wählen Sie auf der Seite RFC erstellen die Kategorie Deployment, die Unterkategorie Applications, Item CodeDeploy Application und Operation Deploy aus der RFC CT-Auswahlliste aus.

b. Behalten Sie die Standardoption Basic bei und legen Sie die Werte wie gezeigt fest.

> **Note**
>
> Verweisen Sie auf die CodeDeploy Anwendung, die CodeDeploy Bereitstellungsgruppe, den S3-Bucket und das Paket, die zuvor erstellt wurden.

```
Subject:                                WP-CD-Deploy-RFC
Description:                            DeployWordPress
S3Bucket:                               BUCKET_NAME
S3Key:                                  wordpress.zip
S3BundleType:                           zip
CodeDeployApplicationName:              WordPress
CodeDeployDeploymentGroupName:          WordPress
CodeDeployIgnoreApplicationStopFailures: false
RevisionType:                           S3



VpcId:                                  VPC_ID
Name:                                   WP-CD-Deploy-Op
TimeoutInMinutes:                       60
```

c. Wenn Sie fertig sind, klicken Sie auf Senden.

Validieren Sie die Anwendungsbereitstellung

Navigieren Sie zum Endpunkt (LoadBalancerCName) des zuvor erstellten Load Balancers mit dem bereitgestellten Pfad:/. WordPress WordPress Beispiel:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Sie sollten eine Seite wie diese sehen:

Zerreißen Sie die Hochverfügbarkeitsbereitstellung

Um die Bereitstellung rückgängig zu machen, reichen Sie den Delete Stack CT für den HA Two-Tier Stack und den S3-Bucket ein und Sie können beantragen, dass RDS-Snapshots gelöscht werden (sie werden nach zehn Tagen automatisch gelöscht, kosten aber dort einen geringen Betrag). Stellen Sie den Stack IDs für den HA-Stack und den S3-Bucket zusammen und folgen Sie dann diesen Schritten. Siehe Stapel | Löschen.

Konsolen-Tutorial: Bereitstellen einer WordPress Tier-and-Tie-Website

In diesem Abschnitt wird beschrieben, wie Sie mithilfe der AMS-Konsole eine WordPress Hochverfügbarkeits-Site (HA) in einer AMS-Umgebung bereitstellen. Diese Anweisungen enthalten ein Beispiel für die Erstellung der erforderlichen WordPress CodeDeploy -kompatiblen Paketdatei (z. B. Zip). Die Bereitstellung der Ressourcen folgt einer Reihenfolge, die es Ihnen ermöglicht, sie zu „Stufen“ zusammenzufassen.

> **Note**
>
> Diese exemplarische Vorgehensweise für die Bereitstellung ist für die Verwendung mit einem AMZN-Linux-Betriebssystem konzipiert.
> Die wesentlichen Variablenparameter sind wie folgt *`replaceable`* notiert. Möglicherweise möchten Sie jedoch andere Parameter an Ihre Situation anpassen.

Zusammenfassung der Aufgaben und der erforderlichen RFCs Aufgaben:

1. Erstellen Sie die Infrastruktur:
   a. Erstellen Sie einen MySQL RDS-Datenbankcluster
   b. Erstellen eines Load Balancers
   c. Erstellen Sie eine Auto Scaling-Gruppe und verknüpfen Sie sie mit dem Load Balancer
   d. Erstellen Sie einen S3-Bucket für Anwendungen CodeDeploy
2. Erstellen Sie ein WordPress Anwendungspaket (erfordert keinen RFC)
3. Stellen Sie das WordPress Anwendungspaket bereit mit CodeDeploy:
   a. Erstellen Sie eine CodeDeploy Anwendung

   b. Erstellen Sie eine CodeDeploy Bereitstellungsgruppe
   c. Laden Sie Ihr WordPress Anwendungspaket in den S3-Bucket hoch (erfordert keinen RFC)
   d. Stellen Sie die Anwendung CodeDeploy bereit
4. Validieren Sie die Bereitstellung
5. Reißen Sie die Bereitstellung ab

Beschreibungen aller CT-Optionen, einschließlich, ChangeTypeId finden Sie in der AMS Change Type Reference.

Einen RFC mit der Konsole erstellen (Grundlagen)

Dies sind einige Schritte, die Sie jedes Mal ausführen müssen, wenn Sie einen RFC mit der Konsole erstellen.

1. Klicken Sie RFCsin den linken Navigationsbereich, um die RFCs Listenseite zu öffnen, und klicken Sie dann auf RFC erstellen.

   Die Seite RFC erstellen wird geöffnet.
2. Wählen Sie entweder „Änderungstypen durchsuchen“ (Standardeinstellung) oder „Nach Kategorie auswählen“.
3. Änderungstypen durchsuchen:

   a. Klicken Sie auf eine Schnellerstellungsoption, um einen RFC mit einem der am häufigsten verwendeten Änderungstypen zu beginnen.

      Der Bereich Allgemeine Konfiguration für diesen Änderungstyp wird geöffnet, und die Betreffzeile ist ausgefüllt. Um die Details zum Änderungstyp zu sehen, öffnen Sie den Bereich oben auf der Seite.
   b. Verwenden Sie den Bereich Alle Änderungstypen.

      Filtern Sie, wechseln Sie zwischen einer Karten- oder Tabellenansicht oder sortieren Sie die Änderungstypen. Wenn Sie den gewünschten gefunden haben, wählen Sie ihn aus und klicken Sie oben auf der Seite auf RFC erstellen.

      Der Bereich Allgemeine Konfiguration für diesen Änderungstyp wird geöffnet, und die Betreffzeile ist ausgefüllt. Um die Details zum Änderungstyp zu sehen, öffnen Sie den Bereich oben auf der Seite.

4. Wählen Sie nach Kategorie:

   a. Wählen Sie die entsprechende Kategorie, Unterkategorie, Artikel und Operation aus.

      Das Feld mit den Details zum Änderungstyp wird unten auf der Seite angezeigt.

   b. Klicken Sie unten auf der Seite auf RFC erstellen.

   c. Der Bereich Allgemeine Konfiguration für diesen Änderungstyp wird geöffnet, und die Betreffzeile ist ausgefüllt. Um die Details zum Änderungstyp zu sehen, öffnen Sie den Bereich oben auf der Seite.

5. Um sicherzustellen, dass bestimmte Personen über den RFC-Fortschritt informiert werden, geben Sie die E-Mail-Adressen ein. Um Details zum Änderungstyp hinzuzufügen, geben Sie die Beschreibung ein. Öffnen Sie den Bereich Zusätzliche Konfiguration, um weitere Einzelheiten zum RFC hinzuzufügen.
6. Wählen Sie unter Planung entweder Diese Änderung so schnell wie möglich ausführen oder Diese Änderung planen aus. Wenn Sie „Diese Änderung so schnell wie möglich ausführen“ auswählen, wird Ihr RFC ausgeführt, sobald die Genehmigungen bestanden sind. Wenn Sie „Diese Änderung planen“ auswählen, werden ein Auswahlkalender, eine Uhrzeit und eine Zeitzone angezeigt, und Ihr RFC startet nach der Übermittlung wie geplant.
7. Konfigurieren Sie im Bereich Ausführungskonfiguration die Parameter für den Änderungstyp. Um optionale Parameter zu sehen, öffnen Sie den Bereich Zusätzliche Konfiguration.
8. Wenn Sie bereit sind, klicken Sie auf Ausführen.

Schaffung der Infrastruktur

Melden Sie sich bei der AWS-Konsole für das AMS-Zielkonto und dann bei der AMS-Konsole für das Konto an.

In den folgenden Verfahren wird beschrieben, wie Sie eine RDS-Datenbank, einen Load Balancer und eine Auto Scaling Scaling-Gruppe so erstellen, dass Sie die Ressource IDs zum Aufbau der Infrastruktur verwenden.

Einen RDS-Stack erstellen

Siehe RDS-Stack | Erstellen.

Einen ELB-Stack erstellen

Starten Sie ein öffentliches ELB.

ERFORDERLICHE DATEN:

- `VpcId`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.
- `ELBSubnetIds`: Eine Reihe von Subnetzen, über die der Load Balancer den Datenverkehr verteilt. Wählen Sie entweder öffentliche oder private Subnetze aus. Suchen Sie das Subnetz IDs mit der Referenz For the AMS SKMS API auf der Registerkarte Berichte in der AWS Artifact Console. Operation (CLI: list-subnet-summaries) oder auf der Seite AMS-Konsole VPCs -> VPC-Details.
- `VpcId`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.

1. Wählen Sie auf der Seite RFC erstellen die Kategorie Deployment, die Unterkategorie Advanced Stack Components, den Eintrag Load Balancer (ELB) -Stack aus und klicken Sie auf Create. Wählen Sie „Erweitert“ und akzeptieren Sie alle Standardwerte (auch solche ohne Wert) mit Ausnahme der als Nächstes angezeigten.

```
Subject:                        WP-ELB-RFC
ELBSubnetIds:                   PUBLIC_AZ1
                              PUBLIC_AZ2
ELBScheme                       true
ELBCookieExpirationPeriod       600
VpcId:                          VPC_ID
Name:                           WP-Public-ELB
```

2. Wenn Sie fertig sind, klicken Sie auf Senden.

Erstellen Sie einen Auto Scaling Scaling-Gruppenstapel

Starten Sie eine Auto Scaling-Gruppe.

ERFORDERLICHE DATEN:

- `VpcId`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.
- `AMI-ID`: Dieser Wert bestimmt, welche Art von EC2 Instances Ihre Auto Scaling Scaling-Gruppe (ASG) hochfahren wird. Stellen Sie sicher, dass Sie in Ihrem Konto ein AMI auswählen, das mit „Kunde-“ beginnt und das von Ihnen gewünschte Betriebssystem hat. Finden Sie AMI IDs mit der Referenz zur AMS SKMS API auf der Registerkarte Berichte in der AWS Artifact Console. Operation (CLI: list-amis) oder auf der Seite AMS-Konsole -> Details. VPCs VPCs Diese exemplarische Vorgehensweise ist für die ASGs Konfiguration zur Verwendung eines Linux-AMI AMI.

- `ASGLoadBalancerNames`: Der Load Balancer, den Sie zuvor erstellt haben. Suchen Sie nach dem Namen, indem Sie unter EC2 Console -> Load Balancers (im linken Navigationsbereich) nach dem Namen suchen. Beachten Sie, dass dies nicht der „Name“ ist, den Sie bei der vorherigen Erstellung des ELB angegeben haben.

1. Wählen Sie auf der Seite RFC erstellen die Kategorie Deployment, die Unterkategorie Advanced Stack Components und den Eintrag Auto Scaling-Gruppe aus, und klicken Sie auf Erstellen. Wählen Sie „Erweitert“ und akzeptieren Sie alle Standardwerte (auch solche ohne Wert) mit Ausnahme der als Nächstes angezeigten.

> **Note**
>
> Geben Sie das neueste AMS-AMI an. Geben Sie das zuvor erstellte ELB an.

```
Subject:                        WP-ASG-RFC
ASGSubnetIds:                   PRIVATE_AZ1
                                                    PRIVATE_AZ2
ASGAmiId:                       AMI_ID
VpcId:                          VPC_ID
Name:                           WP_ASG
ASGLoadBalancerNames:           ELB_NAME
ASGUserData:
#!/bin/bash
REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed
 's/[a-z]$//')
yum -y install ruby httpd
chkconfig httpd on
service httpd start
touch /var/www/html/status
cd /tmp
curl -O https://aws-codedeploy-$REGION.s3.amazonaws.com/latest/install
chmod +x ./install
./install auto
chkconfig codedeploy-agent on
service codedeploy-agent start
```

2. Wenn Sie fertig sind, klicken Sie auf Senden.

Einen S3-Stack erstellen

Starten Sie einen S3-Bucket. In den S3-Bucket laden Sie das von Ihnen erstellte Anwendungspaket hoch.

ERFORDERLICHE DATEN:

- `VPC-ID`: Dieser Wert bestimmt, wo sich Ihr S3-Bucket befinden wird. Dies sollte derselbe sein wie bei der zuvor verwendeten VPC.
- `AccessControl`: Die voreingestellten AccessControl Listenoptionen (ACL) sind`Private`, und. `PublicRead` Weitere Informationen finden Sie unter Amazon Simple Storage Service Canned ACL.
- `BucketName`: Dieser Wert legt den Namen des S3-Buckets fest. Sie verwenden ihn, um Ihr Anwendungspaket hochzuladen. Er muss in der gesamten Region des Kontos eindeutig sein und darf keine Großbuchstaben enthalten. Die Angabe Ihrer Konto-ID als Teil von BucketName ist keine Voraussetzung, erleichtert jedoch die spätere Identifizierung des Buckets. Um zu sehen, welche S3-Bucket-Namen in dem Konto vorhanden sind, rufen Sie die Amazon S3 S3-Konsole für Ihr Konto auf.

1. Wählen Sie auf der Seite Create RFC die Kategorie Deployment, die Unterkategorie Advanced Stack Components, den Artikel S3-Speicher aus und klicken Sie auf Create.

   Sie können die Standardparameteroption auf Basic belassen, um die Standardwerte wie beschrieben zu akzeptieren. Um andere Werte festzulegen, wählen Sie Erweitert.

   > **Note**
   >
   > Der mit diesem Änderungstyp bereitgestellte Bucket ermöglicht vollen read/write Zugriff auf das gesamte Konto. Möglicherweise sind neue Änderungstypen erforderlich, um eingeschränktere Zugriffsberechtigungen zu ermöglichen.

   ```
   Subject:            S3-Bucket-RFC
   BucketName:         ACCOUNT_ID-codedeploy-bundles
   AccessControl:      Private

   VpcId:              VPC_ID
   Name:               S3BucketForWP
   ```

2. Wenn Sie fertig sind, klicken Sie auf Senden.

Ein WordPress CodeDeploy Bundle erstellen

Der Abschnitt enthält ein Beispiel für die Erstellung eines Anwendungsbereitstellungspakets.

1. Laden Sie die Dateien herunter WordPress, extrahieren Sie sie und erstellen Sie eine. Verzeichnis /scripts.

   Linux-Befehl:

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows: `https://github.com/WordPress/WordPress/archive/master.zip` In ein Browserfenster einfügen und die Zip-Datei herunterladen.

   Erstellen Sie ein temporäres Verzeichnis, in dem das Paket zusammengestellt werden soll.

   Linux:

   ```
   mkdir /tmp/WordPress
   ```

   Windows: Erstellen Sie ein "WordPress" Verzeichnis. Sie werden den Verzeichnispfad später verwenden.

2. Extrahieren Sie die WordPress Quelle in das Verzeichnis WordPress "" und erstellen Sie ein. Verzeichnis /scripts.

   Linux:

   ```
   unzip master.zip -d /tmp/WordPress_Temp
   cp -paf /tmp/WordPress_Temp/WordPress-master/* /tmp/WordPress
   rm -rf /tmp/WordPress_Temp
   rm -f master
   cd /tmp/WordPress
   mkdir scripts
   ```

   Windows: Gehen Sie in das Verzeichnis "WordPress", das Sie erstellt haben, und erstellen Sie dort ein Verzeichnis „scripts“.

Wenn Sie sich in einer Windows-Umgebung befinden, stellen Sie sicher, dass Sie den Unterbrechungstyp für die Skriptdateien auf Unix (LF) setzen. In Notepad ++ ist dies eine Option unten rechts im Fenster.

3. Erstellen Sie die Datei CodeDeploy appspec.yml im WordPress Verzeichnis (wenn Sie das Beispiel kopieren, überprüfen Sie den Einzug, jedes Leerzeichen zählt). WICHTIG: Stellen Sie sicher, dass der Quellpfad für das Kopieren der WordPress Dateien (in diesem Fall in Ihrem WordPress Verzeichnis) an das erwartete Ziel (/) korrekt ist. var/www/html/WordPress Im Beispiel befindet sich die Datei appspec.yml im Verzeichnis mit den WordPress Dateien, sodass nur „/“ benötigt wird. Auch wenn Sie ein RHEL-AMI für Ihre Auto Scaling Scaling-Gruppe verwendet haben, sollten Sie die Zeile „os: linux“ unverändert lassen. Beispiel für eine appspec.yml-Datei:

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/config_wordpress.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

4. Erstellen Sie Bash-Dateiskripts in der. WordPress Verzeichnis /scripts.

   Erstellen Sie zunächst `config_wordpress.sh` mit dem folgenden Inhalt (wenn Sie möchten, können Sie die Datei wp-config.php direkt bearbeiten).

> **Note**
>
> *DBName*Ersetzen Sie durch den Wert, der im HA Stack-RFC angegeben ist (z. B.`wordpress`).
> *DB_MasterUsername*Ersetzen Sie durch den `MasterUsername` Wert, der im HA Stack-RFC angegeben ist (z. B.`admin`).
> *DB_MasterUserPassword*Ersetzen Sie durch den `MasterUserPassword` Wert, der im HA Stack-RFC angegeben ist (z. B.`p4ssw0rd`).
> *DB_ENDPOINT*Ersetzen Sie es in den Ausführungsausgaben des HA Stack-RFC durch den DNS-Namen des Endpunkts (z. B.`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Sie finden dies mit der GetRfcOperation (CLI: get-rfc --rfc-id RFC_ID) oder auf der RFC-Detailseite der AMS-Konsole für den HA Stack-RFC, den Sie zuvor eingereicht haben.

```
#!/bin/bash
chmod -R 755 /var/www/html/WordPress
cp /var/www/html/WordPress/wp-config-sample.php /var/www/html/WordPress/wp-
config.php
cd /var/www/html/WordPress
sed -i "s/database_name_here/DBName/g" wp-config.php
sed -i "s/username_here/DB_MasterUsername/g" wp-config.php
sed -i "s/password_here/DB_MasterUserPassword/g" wp-config.php
sed -i "s/localhost/DB_ENDPOINT/g" wp-config.php
```

5. Erstellen Sie im selben Verzeichnis mit dem folgenden Inhalt: `install_dependencies.sh`

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS wird beim Start als Teil der Benutzerdaten installiert, damit Integritätsprüfungen von Anfang an funktionieren.

6. Erstellen Sie im selben Verzeichnis `start_server.sh` mit dem folgenden Inhalt:

   - Verwenden Sie für Amazon Linux-Instances Folgendes:

     ```
     #!/bin/bash
     service httpd start
     ```

   - Verwenden Sie für RHEL-Instances Folgendes (die zusätzlichen Befehle sind Richtlinien, die es SELINUX ermöglichen, sie zu akzeptieren): WordPress

     ```
     #!/bin/bash
     setsebool -P  httpd_can_network_connect_db 1
     setsebool -P  httpd_can_network_connect 1
     chcon -t httpd_sys_rw_content_t /var/www/html/WordPress/wp-content -R
     restorecon -Rv /var/www/html
     service httpd start
     ```

7. Erstellen Sie im selben Verzeichnis `stop_server.sh` mit dem folgenden Inhalt:

   ```
   #!/bin/bash
   service httpd stop
   ```

8. Erstellen Sie das Zip-Bundle.

   Linux:

   ```
   $ cd /tmp/WordPress
   $ zip -r wordpress.zip .
   ```

   Windows: Gehen Sie in Ihr "WordPress" -Verzeichnis, wählen Sie alle Dateien aus und erstellen Sie eine ZIP-Datei. Achten Sie darauf, sie wordpress.zip zu nennen.

Stellen Sie das WordPress Anwendungspaket bereit mit CodeDeploy

Das CodeDeploy ist ein AWS-Bereitstellungsservice, der Anwendungsbereitstellungen auf EC2 Amazon-Instances automatisiert. Dieser Teil des Prozesses umfasst das Erstellen einer CodeDeploy Anwendung, das Erstellen einer CodeDeploy Bereitstellungsgruppe und das anschließende Bereitstellen der Anwendung mithilfe von. CodeDeploy

Eine CodeDeploy Anwendung erstellen

Die CodeDeploy Anwendung ist einfach ein Name oder ein Container, der von AWS verwendet wird, CodeDeploy um sicherzustellen, dass während einer Bereitstellung auf die richtige Version, Bereitstellungskonfiguration und Bereitstellungsgruppe verwiesen wird. Die Bereitstellungskonfiguration ist in diesem Fall das WordPress Paket, das Sie zuvor erstellt haben.

ERFORDERLICHE DATEN:

- `VpcId`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.
- `CodeDeployApplicationName`: Muss im Konto eindeutig sein. Suchen Sie in der CodeDeploy Konsole nach vorhandenen Anwendungsnamen.

1. Erstellen Sie die CodeDeploy Anwendung für WordPress

   Wählen Sie auf der Seite RFC erstellen die Kategorie Deployment, die Unterkategorie Applications, den Artikel CodeDeploy Application and Operation Create aus der RFC CT-Auswahlliste aus. Wählen Sie Basic und legen Sie die Werte wie in der Abbildung gezeigt fest. Wenn Sie fertig sind, klicken Sie auf Senden.

   ```
   Subject:                        CD-WP-App-RFC
   CodeDeployApplicationName:      WordPress
   VpcId:                          VPC_ID
   Name:                           WP-CD-App
   ```

2. Wenn Sie fertig sind, klicken Sie auf Senden.

Eine CodeDeploy Bereitstellungsgruppe erstellen

Erstellen Sie die CodeDeploy Bereitstellungsgruppe.

Eine CodeDeploy Bereitstellungsgruppe definiert eine Reihe von einzelnen Instanzen, die für eine Bereitstellung vorgesehen sind.

ERFORDERLICHE DATEN:

- `VpcId`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.
- `CodeDeployApplicationName`: Verwenden Sie den Wert, den Sie zuvor erstellt haben.

- `CodeDeployAutoScalingGroups`: Verwenden Sie den Namen der Auto Scaling Scaling-Gruppe, die Sie zuvor erstellt haben.
- `CodeDeployDeploymentGroupName`: Ein Name für die Bereitstellungsgruppe. Dieser Name muss für jede dieser Bereitstellungsgruppe zugeordnete Anwendung eindeutig sein.
- `CodeDeployServiceRoleArn`: Verwenden Sie die im Beispiel angegebene Formel.

1. Wählen Sie auf der Seite „RFC erstellen“ die Kategorie „Bereitstellung“, die Unterkategorie „Anwendungen“, die CodeDeploy Artikelbereitstellungsgruppe und die Operation „Erstellen“ aus der Auswahlliste von RFC CT aus. Wählen Sie „Erweitert“ und legen Sie die Werte wie gezeigt fest (für den RFC ist nur ein Betreff erforderlich). Wenn Sie fertig sind, klicken Sie auf Senden.

   > **Note**
   >
   > Verweisen Sie in diesem Format auf die CodeDeploy Dienstrolle ARN `"arn:aws:iam::085398962942:role/aws-codedeploy-role"` und verwenden Sie den zuvor erstellten Auto Scaling-Gruppennamen für „ASG_NAME“.

   ```
   Description:                        Create CodeDeploy Deployment Group for WP
   CodeDeployApplicationName:          WordPress
   CodeDeployAutoScalingGroups:        ASG_NAME
   CodeDeployDeploymentConfigName:     CodeDeployDefault.HalfAtATime
   CodeDeployDeploymentGroupName:      WP CD Group
   CodeDeployServiceRoleArn:           arn:aws:iam::ACCOUNT_ID:role/aws-codedeploy-role

   VpcId:                              VPC_ID
   Name:                               WP Deployment Group
   ```

2. Wenn Sie fertig sind, klicken Sie auf Senden.

Laden Sie die WordPress Bewerbung hoch

Sie haben automatisch Zugriff auf jede S3-Bucket-Instanz, die Sie erstellen. Sie können über Ihre Bastions (siehe Zugreifen auf Instances) oder über die S3-Konsole darauf zugreifen und das CodeDeploy Paket hochladen. Das Bundle muss vorhanden sein, um den Stack weiter bereitstellen zu können. Das Beispiel verwendet den zuvor erstellten Bucket-Namen.

Sie können diesen AWS-Befehl verwenden, um das Paket zu komprimieren:

```
aws s3 cp wordpress/wordpress.zip s3://ACCOUNT_ID-codedeploy-bundles/
```

Stellen Sie die WordPress Anwendung bereit mit CodeDeploy

Stellen Sie die CodeDeploy Anwendung bereit.

ERFORDERLICHE DATEN:

- `VPC-ID`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.
- `CodeDeployApplicationName`: Verwenden Sie den Namen für die CodeDeploy Anwendung, die Sie zuvor erstellt haben.
- `CodeDeployDeploymentGroupName`: Verwenden Sie den Namen der CodeDeploy Bereitstellungsgruppe, die Sie zuvor erstellt haben.
- `S3Location`(wo Sie das Anwendungspaket hochgeladen haben)`S3Bucket`:: BucketName Das, das Sie zuvor erstellt haben, `S3BundleType` und`S3Key`: Der Typ und der Name des Bundles, das Sie in Ihren S3-Store gestellt haben.

1. Stellen Sie das WordPress CodeDeploy Anwendungspaket bereit

   Wählen Sie auf der Seite RFC erstellen die Kategorie Bereitstellung, die Unterkategorie Anwendungen, den Artikel CodeDeploy Anwendung und den Vorgang Deploy aus der RFC CT-Auswahlliste aus. Wählen Sie Basic und legen Sie die Werte wie in der Abbildung gezeigt fest. Wenn Sie fertig sind, klicken Sie auf Senden.

   > **Note**
   >
   > Verweisen Sie auf die CodeDeploy Anwendung, die CodeDeploy Bereitstellungsgruppe, den S3-Bucket und das Paket, die zuvor erstellt wurden.

   ```
   Subject:                          WP-CD-Deploy-RFC
   CodeDeployApplicationName:        WordPress
   CodeDeployDeploymentGroupName:    WPCDGroup
   RevisionType:                     S3
   S3Bucket:                         ACCOUNT_ID-codedeploy-bundles
   S3BundleType:                     zip
   S3Key:                            wordpress.zip
   VpcId:                            VPC_ID
   ```

```
Name:                                    WordPress
```

2. Wenn Sie fertig sind, klicken Sie auf Senden.

Validieren Sie die Anwendungsbereitstellung

Navigieren Sie zum Endpunkt (ELB CName) des zuvor erstellten Load Balancers mit dem bereitgestellten Pfad:/. WordPress WordPress Beispiel:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Machen Sie die Anwendungsbereitstellung rückgängig

Um die Bereitstellung zu beenden, reichen Sie den Delete Stack CT gegen den RDS-Datenbank-Stack, den Application Load Balancer, die Auto Scaling Scaling-Gruppe, den S3-Bucket und die Code Deploy-Anwendung und -Gruppe ein — insgesamt sechs RFCs . Darüber hinaus können Sie eine Serviceanfrage für das Löschen der RDS-Snapshots stellen (sie werden nach zehn Tagen automatisch gelöscht, kosten aber nur einen geringen Betrag). Sammeln Sie den Stapel IDs für alle und folgen Sie dann diesen Schritten. Siehe Stapel | Löschen.

CLI-Tutorial: Zweistufiger Stack mit hoher Verfügbarkeit (Linux/RHEL)

In diesem Abschnitt wird beschrieben, wie Sie mithilfe der AMS-CLI einen zweistufigen Hochverfügbarkeits-Stack (HA) in einer AMS-Umgebung bereitstellen.

> **Note**
>
> Diese exemplarische Vorgehensweise für die Bereitstellung wurde in AMZN Linux- und RHEL-Umgebungen getestet.

Zusammenfassung der Aufgaben und der erforderlichen Aufgaben: RFCs

1. Infrastruktur erstellen (zweistufiger HA-Stack)
2. Erstellen Sie einen S3-Bucket für Anwendungen CodeDeploy

3. Erstellen Sie das WordPress Anwendungspaket und laden Sie es in den S3-Bucket hoch
4. Stellen Sie die Anwendung bereit mit CodeDeploy
5. Greifen Sie auf die WordPress Site zu und melden Sie sich an, um die Bereitstellung zu validieren

Bevor Sie beginnen

Deployment | Advanced Stack Components | High Availability Two Tier Stack Advanced | Create CT erstellt eine Auto Scaling Scaling-Gruppe, einen Load Balancer, eine Datenbank sowie einen CodeDeploy Anwendungsnamen und eine Bereitstellungsgruppe (mit demselben Namen, den Sie der Anwendung geben). Weitere Informationen finden CodeDeploy Sie unter Was ist CodeDeploy?

In dieser exemplarischen Vorgehensweise wird ein High Availability Two-Tier Stack (Advanced) -RFC verwendet, der beinhaltet UserData und auch beschreibt, wie ein WordPress Paket erstellt wird, das CodeDeploy bereitgestellt werden kann.

Im Beispiel werden Instanzmetadaten wie Instanz-ID, Region usw. aus einer laufenden Instanz abgerufen, indem der unter http://169.254.169.254/latest/meta-data/ verfügbare Instanz-Metadatendienst abgefragt wird. `UserData` EC2 Diese Zeile im Benutzerdatenskript:`REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed 's/[a-z]$//')`, ruft den Namen der Verfügbarkeitszone aus dem Metadatendienst in die $REGION-Variable für unsere unterstützten Regionen ab und vervollständigt damit die URL für den S3-Bucket, in den der Agent heruntergeladen wird. CodeDeploy Die 169.254.169.254 IP ist nur innerhalb der VPC routingfähig (alle können den Service abfragen). VPCs Informationen zum Service finden Sie unter Instanz-Metadaten und Benutzerdaten. Beachten Sie auch, dass Skripte, UserData die als eingegeben wurden, als „root“ -Benutzer ausgeführt werden und den Befehl „sudo“ nicht verwenden müssen.

In dieser exemplarischen Vorgehensweise werden die folgenden Parameter auf dem Standardwert belassen (siehe Abbildung):

- Auto Scaling Scaling-Gruppe:`Cooldown=300, DesiredCapacity=2, EBSOptimized=false, HealthCheckGracePeriod=600, IAMInstanceProfile=customer-mc-ec2-instance-profile, InstanceDetailedMonitoring=true, InstanceRootVolumeIops=0, InstanceRootVolumeType=standard, InstanceType=m3.medium, MaxInstances=2, MinInstances=2, ScaleDownPolicyCooldown=300, ScaleDownPolicyEvaluationPeriods=4, ScaleDownPolicyPeriod=60,`

`ScaleDownPolicyScalingAdjustment=-1, ScaleDownPolicyStatistic=Average, ScaleDownPolicyThreshold=35, ScaleMetricName=CPUUtilization, ScaleUpPolicyCooldown=60, ScaleUpPolicyEvaluationPeriods=2, ScaleUpPolicyPeriod=60, ScaleUpPolicyScalingAdjustment=2, ScaleUpPolicyStatistic=Average, ScaleUpPolicyThreshold=75`.

- Load Balancer:`HealthCheckInterval=30, HealthCheckTimeout=5`.
- Datenbank:`BackupRetentionPeriod=7, Backups=true, InstanceType=db.m3.medium, IOPS=0, MultiAZ=true, PreferredBackupWindow=22:00-23:00, PreferredMaintenanceWindow=wed:03:32-wed:04:02, StorageEncrypted=false, StorageEncryptionKey="", StorageType=gp2`.
- Anwendung:`DeploymentConfigName=CodeDeployDefault.OneAtATime`.
- S3-Eimer:`AccessControl=Private`.

ZUSÄTZLICHE EINSTELLUNGEN:

`RequestedStartTime`und `RequestedEndTime` wenn Sie Ihren RFC planen möchten: Sie können Time.is verwenden, um die richtige UTC-Zeit zu ermitteln. Die bereitgestellten Beispiele müssen entsprechend angepasst werden. Ein RFC kann nicht fortgesetzt werden, wenn die Startzeit abgelaufen ist. Alternativ können Sie diese Werte auch weglassen, um einen ASAP-RFC zu erstellen, der ausgeführt wird, sobald die Genehmigungen bestanden wurden.

> **Note**
>
> Es gibt viele Parameter, die Sie möglicherweise anders als in der Abbildung festlegen möchten. Die im Beispiel gezeigten Werte für diese Parameter wurden getestet, sind aber möglicherweise nicht für Sie geeignet.

Erstellen Sie die Infrastruktur

Wenn Sie die folgenden Daten sammeln, bevor Sie beginnen, kann die Bereitstellung schneller vonstattengehen.

DIE ERFORDERLICHEN DATEN SIND GESTAPELT:

- AutoScalingGroup:

  - `UserData`: Dieser Wert wird in diesem Tutorial bereitgestellt. Er enthält Befehle zum Einrichten der Ressource für den Agenten CodeDeploy und zum Starten des CodeDeploy Agenten.
  - `AMI-ID`: Dieser Wert bestimmt, welche Art von EC2 Instances Ihre Auto Scaling Scaling-Gruppe (ASG) hochfahren wird. Stellen Sie sicher, dass Sie in Ihrem Konto ein AMI auswählen, das mit „Kunde-“ beginnt und das von Ihnen gewünschte Betriebssystem hat. Finden Sie AMI IDs mit der Referenz zur AMS SKMS API auf der Registerkarte Berichte in der AWS Artifact Console. Operation (CLI: list-amis) oder auf der Seite AMS-Konsole -> Details. VPCs VPCs Diese exemplarische Vorgehensweise ist für die ASGs Konfiguration zur Verwendung eines Linux-AMI AMI.
- Datenbank:
  - Diese Parameter, `DBEngineEngineVersion`, und `LicenseModel` sollten entsprechend Ihrer Situation eingestellt werden, obwohl die im Beispiel gezeigten Werte getestet wurden.
  - Diese Parameter, `RDSSubnetIds` `DBNameMasterUsername`, und `MasterUserPassword` sind bei der Bereitstellung des Anwendungspakets erforderlich. Verwenden Sie für RDSSubnet IDs zwei private Subnetze.
- LoadBalancer:
  - Diese Parameter, `DBEngineEngineVersion`, und `LicenseModel` sollten entsprechend Ihrer Situation eingestellt werden, obwohl die im Beispiel gezeigten Werte getestet wurden.
  - `ELBSubnetIds`: Verwenden Sie zwei öffentliche Subnetze.
- Anwendung: Der `ApplicationName` Wert legt den CodeDeploy Anwendungsnamen und den Namen der CodeDeploy Bereitstellungsgruppe fest. Sie verwenden ihn, um Ihre Anwendung bereitzustellen. Es muss für das Konto eindeutig sein. Um in Ihrem Konto nach CodeDeploy Namen zu suchen, schauen Sie in der CodeDeploy Konsole nach. Im Beispiel wird "WordPress" verwendet, aber wenn Sie diesen Wert verwenden, stellen Sie sicher, dass er nicht bereits verwendet wird.

Bei diesem Verfahren werden der zweistufige Hochverfügbarkeits-Stack (Advanced) CT (ct-06mjngx5flwto) und der Create S3-Speicher-CT (ct-1a68ck03fn98r) verwendet. Gehen Sie von Ihrem authentifizierten Konto aus in der Befehlszeile wie folgt vor.

1. Starten Sie den Infrastruktur-Stack.

    a. Geben Sie das JSON-Schema der Ausführungsparameter für den HA Two-Tier-Stack-CT in eine Datei in Ihrem aktuellen Ordner namens CreateStackParams .json aus.

```
aws amscm get-change-type-version --change-type-id "ct-06mjngx5flwto"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateStackParams.json
```

b. Ändern Sie das Schema. Ersetzen Sie das nach *variables* Bedarf. Verwenden Sie beispielsweise das gewünschte Betriebssystem für die EC2 Instanzen, die die ASG erstellen wird. Notieren Sie das, `ApplicationName` wie Sie es später für die Bereitstellung der Anwendung verwenden werden. Beachten Sie, dass Sie bis zu 50 Tags hinzufügen können.

```
{
"Description":      "HA two tier stack for WordPress",
"Name":             "WordPressStack",
"TimeoutInMinutes":  360,
"Tags": [
        {
            "Key": "ApplicationName",
            "Value": "WordPress"
        }
    ],
"AutoScalingGroup": {
            "AmiId":    "AMI-ID",
            "UserData": "#!/bin/bash \n
            REGION=$(curl 169.254.169.254/latest/meta-data/placement/
availability-zone/ | sed 's/[a-z]$//') \n
            yum -y install ruby httpd \n
            chkconfig httpd on \n
            service httpd start \n
            touch /var/www/html/status \n
            cd /tmp \n
            curl -O https://aws-codedeploy-$REGION.s3.amazonaws.com/latest/
install \n
            chmod +x ./install \n
            ./install auto \n
            chkconfig codedeploy-agent on \n
            service codedeploy-agent start"
    },
    "LoadBalancer": {
        "Public":               true,
        "HealthCheckTarget":    "HTTP:80/status"
    },
    "Database":     {
        "DBEngine":             "MySQL",
```

```
        "DBName":               "wordpress",
        "EngineVersion":        "8.0.16 ",
        "LicenseModel":         "general-public-license",
        "MasterUsername":       "admin",
        "MasterUserPassword":   "p4ssw0rd"
    },
    "Application":  {
    "ApplicationName":  "WordPress"
        }
}
```

c. Geben Sie die CreateRfc JSON-Vorlage in eine Datei in Ihrem aktuellen Ordner namens CreateStackRfc .json aus:

```
aws amscm create-rfc --generate-cli-skeleton > CreateStackRfc.json
```

d. Ändern Sie die RFC-Vorlage wie folgt und speichern Sie sie. Sie können den Inhalt löschen und ersetzen. Beachten Sie, dass `RequestedStartTime` sie jetzt optional `RequestedEndTime` sind. Wenn Sie sie ausschließen, wird ein ASAP-RFC erstellt, der ausgeführt wird, sobald er genehmigt wurde (was normalerweise automatisch geschieht). Um einen geplanten RFC einzureichen, fügen Sie diese Werte hinzu.

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-06mjngx5flwto",
"Title":                "HA-Stack-For-WP-RFC"
}
```

e. Erstellen Sie den RFC, indem Sie die CreateStackRfc JSON-Datei und die Ausführungsparameterdatei mit den CreateStackParams JSON-Dateien angeben:

```
aws amscm create-rfc --cli-input-json file://CreateStackRfc.json --execution-
parameters file://CreateStackParams.json
```

Sie erhalten die RFC-ID in der Antwort. Speichern Sie die ID für nachfolgende Schritte.

f. Reichen Sie den RFC ein:

```
aws amscm submit-rfc --rfc-id  RFC_ID
```

Wenn der RFC erfolgreich ist, erhalten Sie keine Ausgabe.

g. Führen Sie folgenden Befehl aus, um den RFC-Status zu überprüfen

```
aws amscm get-rfc --rfc-id RFC_ID
```

Notieren Sie sich die RFC-ID.

2. Starten Sie einen S3-Bucket

Wenn Sie die folgenden Daten sammeln, bevor Sie beginnen, kann die Bereitstellung schneller vonstattengehen.

ERFORDERLICHER DATEN-S3-BUCKET:

- `VPC-ID`: Dieser Wert bestimmt, wo sich Ihr S3-Bucket befinden wird. Verwenden Sie dieselbe VPC-ID, die Sie zuvor verwendet haben.
- `BucketName`: Dieser Wert legt den Namen des S3-Buckets fest. Sie verwenden ihn, um Ihr Anwendungspaket hochzuladen. Er muss in der gesamten Region des Kontos eindeutig sein und darf keine Großbuchstaben enthalten. Die Angabe Ihrer Konto-ID als Teil von BucketName ist keine Voraussetzung, erleichtert jedoch die spätere Identifizierung des Buckets. Um zu sehen, welche S3-Bucket-Namen in dem Konto vorhanden sind, rufen Sie die Amazon S3 S3-Konsole für Ihr Konto auf.

a. Geben Sie das JSON-Schema der Ausführungsparameter für den S3-Speicher Create CT in eine JSON-Datei mit dem Namen CreateS3 StoreParams .json aus.

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateS3StoreParams.json
```

b. Ändern Sie das Schema wie folgt. Sie können den Inhalt löschen und ersetzen. Ersetze es *VPC_ID* entsprechend. Die Werte im Beispiel wurden getestet, sind aber möglicherweise nicht für Sie geeignet.

> **Tip**
>
> Die `BucketName` müssen in der gesamten Region des Kontos eindeutig sein und dürfen keine Großbuchstaben enthalten. Die Angabe Ihrer Konto-ID als Teil von BucketName ist keine Voraussetzung, erleichtert jedoch die spätere Identifizierung

des Buckets. Um zu sehen, welche S3-Bucket-Namen in dem Konto vorhanden sind, rufen Sie die Amazon S3 S3-Konsole für Ihr Konto auf.

```
{
"Description":      "S3BucketForWordPressBundle",
"VpcId":            "VPC_ID",
"StackTemplateId":  "stm-s2b72beb000000000",
"Name":             "S3BucketForWP",
"TimeoutInMinutes":  60,
"Parameters":   {
    "AccessControl":    "Private",
    "BucketName":       "ACCOUNT_ID-BUCKET_NAME"
    }
}
```

c. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner mit dem Namen CreateS3 StoreRfc .json aus:

```
aws amscm create-rfc --generate-cli-skeleton > CreateS3StoreRfc.json
```

d. Ändern und speichern Sie die Datei CreateS3 StoreRfc .json. Sie können den Inhalt löschen und ersetzen. Beachten Sie, dass `RequestedStartTime` sie jetzt optional `RequestedEndTime` sind. Wenn Sie sie ausschließen, wird ein ASAP-RFC erstellt, der ausgeführt wird, sobald er genehmigt wurde (was normalerweise automatisch geschieht). Um einen geplanten RFC einzureichen, fügen Sie diese Werte hinzu.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1a68ck03fn98r",
"Title":                "S3-Stack-For-WP-RFC"
}
```

e. Erstellen Sie den RFC, indem Sie die Datei CreateS3 StoreRfc .json und die Ausführungsparameterdatei StoreParams CreateS3 .json angeben:

```
aws amscm create-rfc --cli-input-json file://CreateS3StoreRfc.json  --
execution-parameters file://CreateS3StoreParams.json
```

Sie erhalten den Wert des neuen RFC in RfcId der Antwort. Speichern Sie die ID für nachfolgende Schritte.

f. Reichen Sie den RFC ein:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Wenn der RFC erfolgreich ist, erhalten Sie keine Ausgabe.

g. Führen Sie folgenden Befehl aus, um den RFC-Status zu überprüfen

```
aws amscm get-rfc --rfc-id RFC_ID
```

Anwendung erstellen, hochladen und bereitstellen

Erstellen Sie zunächst ein WordPress Anwendungspaket und verwenden Sie dann das, CodeDeploy CTs um die Anwendung zu erstellen und bereitzustellen.

1. Laden Sie die Dateien herunter WordPress, extrahieren Sie sie und erstellen Sie eine. Verzeichnis /scripts.

   Linux-Befehl:

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows: `https://github.com/WordPress/WordPress/archive/master.zip` In ein Browserfenster einfügen und die Zip-Datei herunterladen.

   Erstellen Sie ein temporäres Verzeichnis, in dem das Paket zusammengestellt werden soll.

   Linux:

   ```
   mkdir /tmp/WordPress
   ```

   Windows: Erstellen Sie ein "WordPress" Verzeichnis. Sie werden den Verzeichnispfad später verwenden.

2. Extrahieren Sie die WordPress Quelle in das Verzeichnis WordPress "" und erstellen Sie ein. Verzeichnis /scripts.

Linux:

```
unzip master.zip -d /tmp/WordPress_Temp
cp -paf /tmp/WordPress_Temp/WordPress-master/* /tmp/WordPress
rm -rf /tmp/WordPress_Temp
rm -f master
cd /tmp/WordPress
mkdir scripts
```

Windows: Gehen Sie in das Verzeichnis "WordPress", das Sie erstellt haben, und erstellen Sie dort ein Verzeichnis „scripts“.

Wenn Sie sich in einer Windows-Umgebung befinden, stellen Sie sicher, dass Sie den Unterbrechungstyp für die Skriptdateien auf Unix (LF) setzen. In Notepad ++ ist dies eine Option unten rechts im Fenster.

3. Erstellen Sie die Datei CodeDeploy appspec.yml im WordPress Verzeichnis (wenn Sie das Beispiel kopieren, überprüfen Sie den Einzug, jedes Leerzeichen zählt). WICHTIG: Stellen Sie sicher, dass der Quellpfad für das Kopieren der WordPress Dateien (in diesem Fall in Ihrem WordPress Verzeichnis) an das erwartete Ziel (/) korrekt ist. var/www/html/WordPress Im Beispiel befindet sich die Datei appspec.yml im Verzeichnis mit den WordPress Dateien, sodass nur „/“ benötigt wird. Auch wenn Sie ein RHEL-AMI für Ihre Auto Scaling Scaling-Gruppe verwendet haben, sollten Sie die Zeile „os: linux“ unverändert lassen. Beispiel für eine appspec.yml-Datei:

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/config_wordpress.sh
      timeout: 300
      runas: root
  ApplicationStart:
```

```
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

4. Erstellen Sie Bash-Dateiskripts in der. WordPress Verzeichnis /scripts.

   Erstellen Sie zunächst `config_wordpress.sh` mit dem folgenden Inhalt (wenn Sie möchten, können Sie die Datei wp-config.php direkt bearbeiten).

   > **Note**
   >
   > *DBName*Ersetzen Sie durch den Wert, der im HA Stack-RFC angegeben ist (z. B.`wordpress`).
   > *DB_MasterUsername*Ersetzen Sie durch den `MasterUsername` Wert, der im HA Stack-RFC angegeben ist (z. B.`admin`).
   > *DB_MasterUserPassword*Ersetzen Sie durch den `MasterUserPassword` Wert, der im HA Stack-RFC angegeben ist (z. B.`p4ssw0rd`).
   > *DB_ENDPOINT*Ersetzen Sie es in den Ausführungsausgaben des HA Stack-RFC durch den DNS-Namen des Endpunkts (z. B.`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Sie finden dies mit der GetRfcOperation (CLI: get-rfc --rfc-id RFC_ID) oder auf der RFC-Detailseite der AMS-Konsole für den HA Stack-RFC, den Sie zuvor eingereicht haben.

```
#!/bin/bash
chmod -R 755 /var/www/html/WordPress
cp /var/www/html/WordPress/wp-config-sample.php /var/www/html/WordPress/wp-
config.php
cd /var/www/html/WordPress
sed -i "s/database_name_here/DBName/g" wp-config.php
sed -i "s/username_here/DB_MasterUsername/g" wp-config.php
sed -i "s/password_here/DB_MasterUserPassword/g" wp-config.php
sed -i "s/localhost/DB_ENDPOINT/g" wp-config.php
```

5. Erstellen Sie im selben Verzeichnis mit dem folgenden Inhalt: `install_dependencies.sh`

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS wird beim Start als Teil der Benutzerdaten installiert, damit Integritätsprüfungen von Anfang an funktionieren.

6. Erstellen Sie im selben Verzeichnis `start_server.sh` mit dem folgenden Inhalt:

   - Verwenden Sie für Amazon Linux-Instances Folgendes:

     ```
     #!/bin/bash
     service httpd start
     ```

   - Verwenden Sie für RHEL-Instances Folgendes (die zusätzlichen Befehle sind Richtlinien, die es SELINUX ermöglichen, sie zu akzeptieren): WordPress

     ```
     #!/bin/bash
     setsebool -P  httpd_can_network_connect_db 1
     setsebool -P  httpd_can_network_connect 1
     chcon -t httpd_sys_rw_content_t /var/www/html/WordPress/wp-content -R
     restorecon -Rv /var/www/html
     service httpd start
     ```

7. Erstellen Sie im selben Verzeichnis `stop_server.sh` mit dem folgenden Inhalt:

   ```
   #!/bin/bash
   service httpd stop
   ```

8. Erstellen Sie das Zip-Bundle.

   Linux:

   ```
   $ cd /tmp/WordPress
   $ zip -r wordpress.zip .
   ```

Windows: Gehen Sie in Ihr "WordPress" -Verzeichnis, wählen Sie alle Dateien aus und erstellen Sie eine ZIP-Datei. Achten Sie darauf, sie wordpress.zip zu nennen.

1. Laden Sie das Anwendungspaket in den S3-Bucket hoch.

   Das Bundle muss vorhanden sein, um den Stack weiter bereitstellen zu können.

   Sie haben automatisch Zugriff auf jede S3-Bucket-Instanz, die Sie erstellen. Sie können über Ihre Bastions oder über die S3-Konsole darauf zugreifen und das WordPress Paket mit hochladen drag-and-drop oder die ZIP-Datei durchsuchen und auswählen.

   Sie können auch den folgenden Befehl in einem Shell-Fenster verwenden. Vergewissern Sie sich, dass Sie den richtigen Pfad zur ZIP-Datei haben:

   ```
   aws s3 cp wordpress.zip s3://BUCKET_NAME/
   ```

2. Stellen Sie das WordPress Anwendungspaket bereit.

   Wenn Sie die folgenden Daten sammeln, bevor Sie beginnen, kann die Bereitstellung schneller vonstattengehen.

   ERFORDERLICHE DATEN:

   - `VPC-ID`: Dieser Wert bestimmt, wo sich Ihr S3-Bucket befinden wird. Verwenden Sie dieselbe VPC-ID, die Sie zuvor verwendet haben.
   - `CodeDeployApplicationNameundCodeDeployApplicationName`: Der `ApplicationName` Wert, den Sie im HA 2-Tier Stack-RFC verwendet haben, legt den und den CodeDeployApplicationName fest. CodeDeployDeploymentGroupName Das Beispiel verwendet "WordPress", aber Sie haben möglicherweise einen anderen Wert verwendet.
   - `S3Location`: Verwenden Sie für den `S3BucketBucketName`, den Sie zuvor erstellt haben. Die `S3BundleType` und `S3Key` stammen aus dem Paket, das Sie in Ihren S3-Shop gestellt haben.

   a. Geben Sie das JSON-Schema der Ausführungsparameter für die CodeDeploy Anwendung Deploy CT in eine JSON-Datei mit dem Namen Deploy CDApp Params.json aus.

```
aws amscm get-change-type-version --change-type-id "ct-2edc3sd1sqmrb"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 DeployCDAppParams.json
```

b. Ändern Sie das Schema wie folgt und speichern Sie es unter. Sie können den Inhalt löschen und ersetzen.

```
{
"Description":                          "DeployWPCDApp",
"VpcId":                                "VPC_ID",
"Name":                                 "WordPressCDAppDeploy",
"TimeoutInMinutes":                     60,
"Parameters":   {
    "CodeDeployApplicationName":                "WordPress",
    "CodeDeployDeploymentGroupName":            "WordPress",
    "CodeDeployIgnoreApplicationStopFailures":   false,
    "CodeDeployRevision": {
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket":     "BUCKET_NAME",
        "S3BundleType": "zip",
        "S3Key":        "wordpress.zip" }
        }
    }
}
```

c. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner mit dem Namen Deploy CDApp rfc.json aus:

```
aws amscm create-rfc --generate-cli-skeleton > DeployCDAppRfc.json
```

d. Ändern und speichern Sie die Datei Deploy CDApp RFC.json. Sie können den Inhalt löschen und ersetzen. Beachten Sie, dass `RequestedStartTime` sie jetzt optional `RequestedEndTime` sind. Wenn Sie sie ausschließen, wird ein ASAP-RFC erstellt, der ausgeführt wird, sobald er genehmigt wurde (was normalerweise automatisch geschieht). Um einen geplanten RFC einzureichen, fügen Sie diese Werte hinzu.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-WP-RFC"
```

```
}
```

e. Erstellen Sie den RFC, indem Sie die Datei Deploy CDApp Rfc und die Ausführungsparameterdatei Deploy CDApp Params angeben:

```
aws amscm create-rfc --cli-input-json file://DeployCDAppRfc.json  --execution-
parameters file://DeployCDAppParams.json
```

Sie erhalten den Wert RfcId des neuen RFC in der Antwort. Speichern Sie die ID für nachfolgende Schritte.

f. Reichen Sie den RFC ein:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Wenn der RFC erfolgreich ist, erhalten Sie keine Ausgabe.

g. Führen Sie folgenden Befehl aus, um den RFC-Status zu überprüfen

```
aws amscm get-rfc --rfc-id RFC_ID
```

Validieren Sie die Anwendungsbereitstellung

Navigieren Sie zum Endpunkt (ELB CName) des zuvor erstellten Load Balancers mit dem bereitgestellten Pfad:/. WordPress WordPress Beispiel:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Machen Sie die Anwendungsbereitstellung rückgängig

Sobald Sie mit dem Tutorial fertig sind, sollten Sie die Bereitstellung beenden, damit Ihnen die Ressourcen nicht in Rechnung gestellt werden.

Im Folgenden wird ein generischer Vorgang zum Löschen von Stacks beschrieben. Sie sollten ihn zweimal einreichen, einmal für den HA 2-Tier-Stack und einmal für den S3-Bucket-Stack. Als letzten Schritt reichen Sie eine Serviceanfrage ein, dass alle Snapshots für den S3-Bucket (einschließlich der S3-Bucket-Stack-ID in der Serviceanfrage) gelöscht werden. Sie werden nach 10 Tagen automatisch gelöscht, aber wenn Sie sie vorzeitig löschen, sparen Sie ein wenig Kosten.

Diese exemplarische Vorgehensweise bietet ein Beispiel für die Verwendung der AMS-Konsole zum Löschen eines S3-Stacks. Dieses Verfahren gilt für das Löschen eines beliebigen Stacks mithilfe der AMS-Konsole.

> **Note**
>
> Wenn Sie einen S3-Bucket löschen, müssen Sie ihn zuerst von Objekten leeren.

ERFORDERLICHE DATEN:

- `StackId`: Der zu verwendende Stapel. Sie finden ihn auf der Seite AMS Console Stacks, die Sie über einen Link im linken Navigationsbereich aufrufen können. Führen Sie mithilfe der AMS SKMS API/CLI den Vorgang For the AMS SKMS API reference, see the Reports in der AWS Artifact Console. (in der CLI) aus. `list-stack-summaries`
- Die Änderungstyp-ID für diese exemplarische Vorgehensweise lautet `ct-0q0bic0ywqk6c` „1.0“. Führen Sie den folgenden Befehl aus, um die neueste Version herauszufinden:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0q0bic0ywqk6c
```

INLINE-ERSTELLUNG:

- Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben). E

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0"
 --title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

- Senden Sie den RFC mit der RFC-ID, die bei der RFC-Erstellung zurückgegeben wurde. Bis zur Übermittlung verbleibt der RFC im `Editing` Status und es wird nicht darauf reagiert.

```
aws amscm submit-rfc --rfc-id RFC_ID
```

- Überwachen Sie den RFC-Status und sehen Sie sich die Ausführungsausgabe an:

```
aws amscm get-rfc --rfc-id RFC_ID
```

VORLAGE ERSTELLEN:

1. Gibt die RFC-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. Beispiel nennt sie DeleteStackRfc .json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteStackRfc.json
   ```

2. Ändern und speichern Sie die DeleteStackRfc JSON-Datei. Da das Löschen eines Stacks nur einen Ausführungsparameter hat, können sich die Ausführungsparameter in der DeleteStackRfc JSON-Datei selbst befinden (es ist nicht erforderlich, eine separate JSON-Datei mit Ausführungsparametern zu erstellen).

   Die internen Anführungszeichen in der ExecutionParameters JSON-Erweiterung müssen mit einem Backslash (\) maskiert werden. Beispiel ohne Start- und Endzeit:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0q0bic0ywqk6c",
   "Title":                "Delete-My-Stack-RFC"
   "ExecutionParameters":  "{
           \"StackId\":\"STACK_ID\"}"
   }
   ```

3. Erstellen Sie den RFC:

   ```
   aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
   ```

   In der Antwort erhalten Sie den RfcId des neuen RFC. Beispiel:

   ```
   {
   "RfcId": "daaa1867-ffc5-1473-192a-842f6b326102"
   }
   ```

   Speichern Sie die ID für nachfolgende Schritte.

4. Reichen Sie den RFC ein:

   ```
   aws amscm submit-rfc --rfc-id RFC_ID
   ```

   Wenn der RFC erfolgreich ist, erhalten Sie keine Bestätigung in der Befehlszeile.

5. Um den Status der Anfrage zu überwachen und die Ausführungsausgabe anzuzeigen:

```
aws amscm get-rfc --rfc-id RFC_ID --query "Rfc.
{Status:Status.Name,Exec:ExecutionOutput}" --output table
```

CLI-Tutorial: Bereitstellen einer WordPress Tier-and-Tie-Website

In diesem Abschnitt wird beschrieben, wie Sie mithilfe der AMS-CLI eine WordPress Hochverfügbarkeits-Site (HA) in einer AMS-Umgebung bereitstellen. Diese Anweisungen enthalten ein Beispiel für die Erstellung der erforderlichen WordPress CodeDeploy -kompatiblen Paketdatei (z. B. Zip).

> **Note**
>
> Diese exemplarische Vorgehensweise für die Bereitstellung ist für die Verwendung in einer AMZN-Linux-Umgebung konzipiert.
> Die wesentlichen Variablenparameter sind wie folgt *replaceable* notiert. Möglicherweise möchten Sie jedoch andere Parameter an Ihre Situation anpassen.

Zusammenfassung der Aufgaben und der erforderlichen RFCs Aufgaben:

1. Erstellen Sie die Infrastruktur:
   a. Erstellen Sie einen RDS-Stack (CLI)
   b. Erstellen eines Load Balancers
   c. Erstellen Sie eine Auto Scaling-Gruppe und verknüpfen Sie sie mit dem Load Balancer
   d. Erstellen Sie einen S3-Bucket für Anwendungen CodeDeploy
2. Erstellen Sie ein WordPress Anwendungspaket (erfordert keinen RFC)
3. Stellen Sie das WordPress Anwendungspaket bereit mit CodeDeploy:
   a. Erstellen Sie eine CodeDeploy Anwendung
   b. Erstellen Sie eine CodeDeploy Bereitstellungsgruppe
   c. Laden Sie Ihr WordPress Anwendungspaket in den S3-Bucket hoch (erfordert keinen RFC)
   d. Stellen Sie die Anwendung CodeDeploy bereit
4. Validieren Sie die Bereitstellung

5. Reißen Sie die Bereitstellung ab

Folgen Sie allen Schritten in der Befehlszeile von Ihrem authentifizierten Konto aus.

Einen RFC mit der CLI erstellen

Ausführliche Informationen zum Erstellen RFCs finden Sie unter Erstellen RFCs; eine Erläuterung gängiger RFC-Parameter finden Sie unter Allgemeine RFC-Parameter.

Erstellen Sie die Infrastruktur

In den folgenden Verfahren wird beschrieben, wie Sie eine RDS-Datenbank, einen Load Balancer und eine Auto Scaling Scaling-Gruppe so erstellen, dass Sie die Ressource IDs zum Aufbau der Infrastruktur verwenden.

Erstellen Sie einen RDS-Stack (CLI)

Siehe RDS-Stack | Erstellen.

Erstellen Sie einen ELB-Stack

Starten Sie einen öffentlichen Load Balancer (ELB). Siehe Load Balancer (ELB) Stack | Erstellen.

Erstellen Sie einen Auto Scaling Scaling-Gruppenstapel

Starten Sie eine Auto Scaling-Gruppe.

Weitere Informationen finden Sie unter Auto Scaling Scaling-Gruppe | Erstellen.

Erstellen Sie einen S3-Store

Starten Sie einen S3-Bucket. In den S3-Bucket laden Sie das von Ihnen erstellte Anwendungspaket hoch. Weitere Informationen finden Sie unter S3-Speicher | Erstellen.

Erstellen Sie ein WordPress Anwendungspaket für CodeDeploy

Dieser Abschnitt enthält ein Beispiel für die Erstellung eines Anwendungsbereitstellungspakets.

1. Laden Sie die Dateien herunter WordPress, extrahieren Sie sie und erstellen Sie eine. Verzeichnis /scripts.

Linux-Befehl:

```
wget https://github.com/WordPress/WordPress/archive/master.zip
```

Windows: `https://github.com/WordPress/WordPress/archive/master.zip` In ein Browserfenster einfügen und die Zip-Datei herunterladen.

Erstellen Sie ein temporäres Verzeichnis, in dem das Paket zusammengestellt werden soll.

Linux:

```
mkdir /tmp/WordPress
```

Windows: Erstellen Sie ein "WordPress" Verzeichnis. Sie werden den Verzeichnispfad später verwenden.

2. Extrahieren Sie die WordPress Quelle in das Verzeichnis WordPress "" und erstellen Sie ein. Verzeichnis /scripts.

   Linux:

   ```
   unzip master.zip -d /tmp/WordPress_Temp
   cp -paf /tmp/WordPress_Temp/WordPress-master/* /tmp/WordPress
   rm -rf /tmp/WordPress_Temp
   rm -f master
   cd /tmp/WordPress
   mkdir scripts
   ```

   Windows: Gehen Sie in das Verzeichnis "WordPress", das Sie erstellt haben, und erstellen Sie dort ein Verzeichnis „scripts“.

   Wenn Sie sich in einer Windows-Umgebung befinden, stellen Sie sicher, dass Sie den Unterbrechungstyp für die Skriptdateien auf Unix (LF) setzen. In Notepad ++ ist dies eine Option unten rechts im Fenster.

3. Erstellen Sie die Datei CodeDeploy appspec.yml im WordPress Verzeichnis (wenn Sie das Beispiel kopieren, überprüfen Sie den Einzug, jedes Leerzeichen zählt). WICHTIG: Stellen Sie sicher, dass der Quellpfad für das Kopieren der WordPress Dateien (in diesem Fall in Ihrem WordPress Verzeichnis) an das erwartete Ziel (/) korrekt ist. var/www/html/WordPress Im Beispiel befindet sich die Datei appspec.yml im Verzeichnis mit den WordPress Dateien,

sodass nur „/“ benötigt wird. Auch wenn Sie ein RHEL-AMI für Ihre Auto Scaling Scaling-Gruppe verwendet haben, sollten Sie die Zeile „os: linux“ unverändert lassen. Beispiel für eine appspec.yml-Datei:

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/config_wordpress.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server.sh
      timeout: 300
      runas: root
```

4. Erstellen Sie Bash-Dateiskripts in der. WordPress Verzeichnis /scripts.

   Erstellen Sie zunächst `config_wordpress.sh` mit dem folgenden Inhalt (wenn Sie möchten, können Sie die Datei wp-config.php direkt bearbeiten).

   > **Note**
   >
   > *DBName* Ersetzen Sie durch den Wert, der im HA Stack-RFC angegeben ist (z. B. `wordpress`).
   > *DB_MasterUsername* Ersetzen Sie durch den `MasterUsername` Wert, der im HA Stack-RFC angegeben ist (z. B. `admin`).
   > *DB_MasterUserPassword* Ersetzen Sie durch den `MasterUserPassword` Wert, der im HA Stack-RFC angegeben ist (z. B. `p4ssw0rd`).

*DB_ENDPOINT*Ersetzen Sie es in den Ausführungsausgaben des HA Stack-RFC durch den DNS-Namen des Endpunkts (z. B.`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Sie finden dies mit der GetRfcOperation (CLI: get-rfc --rfc-id RFC_ID) oder auf der RFC-Detailseite der AMS-Konsole für den HA Stack-RFC, den Sie zuvor eingereicht haben.

```
#!/bin/bash
chmod -R 755 /var/www/html/WordPress
cp /var/www/html/WordPress/wp-config-sample.php /var/www/html/WordPress/wp-
config.php
cd /var/www/html/WordPress
sed -i "s/database_name_here/DBName/g" wp-config.php
sed -i "s/username_here/DB_MasterUsername/g" wp-config.php
sed -i "s/password_here/DB_MasterUserPassword/g" wp-config.php
sed -i "s/localhost/DB_ENDPOINT/g" wp-config.php
```

5. Erstellen Sie im selben Verzeichnis mit dem folgenden Inhalt: `install_dependencies.sh`

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS wird beim Start als Teil der Benutzerdaten installiert, damit Integritätsprüfungen von Anfang an funktionieren.

6. Erstellen Sie im selben Verzeichnis `start_server.sh` mit dem folgenden Inhalt:

   - Verwenden Sie für Amazon Linux-Instances Folgendes:

     ```
     #!/bin/bash
     service httpd start
     ```

   - Verwenden Sie für RHEL-Instances Folgendes (die zusätzlichen Befehle sind Richtlinien, die es SELINUX ermöglichen, sie zu akzeptieren): WordPress

```
#!/bin/bash
setsebool -P  httpd_can_network_connect_db 1
setsebool -P  httpd_can_network_connect 1
chcon -t httpd_sys_rw_content_t /var/www/html/WordPress/wp-content -R
restorecon -Rv /var/www/html
service httpd start
```

7. Erstellen Sie im selben Verzeichnis `stop_server.sh` mit dem folgenden Inhalt:

```
#!/bin/bash
service httpd stop
```

8. Erstellen Sie das Zip-Bundle.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows: Gehen Sie in Ihr "WordPress" -Verzeichnis, wählen Sie alle Dateien aus und erstellen Sie eine ZIP-Datei. Achten Sie darauf, sie wordpress.zip zu nennen.

Stellen Sie das WordPress Anwendungspaket bereit mit CodeDeploy

Das CodeDeploy ist ein AWS-Bereitstellungsservice, der Anwendungsbereitstellungen auf EC2 Amazon-Instances automatisiert. Dieser Teil des Prozesses umfasst das Erstellen einer CodeDeploy Anwendung, das Erstellen einer CodeDeploy Bereitstellungsgruppe und das anschließende Bereitstellen der Anwendung mithilfe von. CodeDeploy

Erstellen Sie eine CodeDeploy Anwendung

Die CodeDeploy Anwendung ist einfach ein Name oder ein Container, der von AWS verwendet wird, CodeDeploy um sicherzustellen, dass während einer Bereitstellung auf die richtige Version, Bereitstellungskonfiguration und Bereitstellungsgruppe verwiesen wird. Die Bereitstellungskonfiguration ist in diesem Fall das WordPress Paket, das Sie zuvor erstellt haben.

ERFORDERLICHE DATEN:

- `VpcId`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.

- `CodeDeployApplicationName`: Muss im Konto eindeutig sein. Suchen Sie in der CodeDeploy Konsole nach vorhandenen Anwendungsnamen.
- `ChangeTypeIdundChangeTypeVersion`: Die Änderungstyp-ID für diese exemplarische Vorgehensweise lautet`ct-0ah3gwb9seqk2`: Um die neueste Version herauszufinden, führen Sie diesen Befehl aus:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0ah3gwb9seqk2
```

1. Geben Sie das JSON-Schema der Ausführungsparameter für die CodeDeploy Anwendung CT in eine Datei in Ihrem aktuellen Ordner aus. Das Beispiel nennt es Create CDApp Params.json.

```
aws amscm get-change-type-version --change-type-id "ct-0ah3gwb9seqk2" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateCDAppParams.json
```

2. Ändern und speichern Sie die JSON-Datei wie folgt. Sie können den Inhalt löschen und ersetzen.

```
{
"Description":                      "Create WordPress CodeDeploy App",
"VpcId":                            "VPC_ID",
"StackTemplateId":                  "stm-sft6rv00000000000",
"Name":                             "WordPressCDApp",
"TimeoutInMinutes":                 60,
"Parameters":   {
    "CodeDeployApplicationName":    "WordPressCDApp"
    }
}
```

3. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner aus. Das Beispiel nennt sie Create CDApp Rfc.json.

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDAppRfc.json
```

4. Ändern und speichern Sie die JSON-Datei wie folgt. Sie können den Inhalt löschen und ersetzen. Beachten Sie, dass `RequestedStartTime` und jetzt optional `RequestedEndTime` sind. Wenn Sie sie ausschließen, wird der RFC ausgeführt, sobald er genehmigt wurde (was normalerweise automatisch geschieht). Um einen „geplanten“ RFC einzureichen, fügen Sie diese Werte hinzu.

```
{
```

```
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0ah3gwb9seqk2",
"Title":                "CD-App-For-WP-Stack-RFC"
}
```

5. Erstellen Sie den RFC und geben Sie die Datei Create CDApp Rfc und die Datei mit den Ausführungsparametern an:

```
aws amscm create-rfc --cli-input-json file://CreateCDAppRfc.json --execution-
parameters file://CreateCDAppParams.json
```

   In der Antwort erhalten Sie die RFC-ID des neuen RFC. Speichern Sie die ID für nachfolgende Schritte.

6. Reichen Sie den RFC ein:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Wenn der RFC erfolgreich ist, erhalten Sie keine Ausgabe.

7. Reichen Sie den RFC ein:

```
aws amscm get-rfc --rfc-id RFC_ID
```

Erstellen Sie eine CodeDeploy Bereitstellungsgruppe

Erstellen Sie die CodeDeploy Bereitstellungsgruppe.

Eine CodeDeploy Bereitstellungsgruppe definiert eine Reihe von einzelnen Instanzen, die für eine Bereitstellung vorgesehen sind.

ERFORDERLICHE DATEN:

- `VpcId`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.
- `CodeDeployApplicationName`: Verwenden Sie den Wert, den Sie zuvor erstellt haben.
- `CodeDeployAutoScalingGroups`: Verwenden Sie den Namen der Auto Scaling Scaling-Gruppe, die Sie zuvor erstellt haben.
- `CodeDeployDeploymentGroupName`: Ein Name für die Bereitstellungsgruppe. Dieser Name muss für jede dieser Bereitstellungsgruppe zugeordnete Anwendung eindeutig sein.

- `CodeDeployServiceRoleArn`: Verwenden Sie die im Beispiel angegebene Formel.
- `ChangeTypeIdundChangeTypeVersion`: Die Änderungstyp-ID für diese exemplarische Vorgehensweise lautet`ct-2gd0u847qd9d2`: Um die neueste Version herauszufinden, führen Sie diesen Befehl aus:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-2gd0u847qd9d2
```

1. Gibt das JSON-Schema der Ausführungsparameter in eine Datei in Ihrem aktuellen Ordner aus. Das Beispiel nennt es Create CDDep GroupParams .json.

```
aws amscm get-change-type-version --change-type-id "ct-2gd0u847qd9d2"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateCDDepGroupParams.json
```

2. Ändern und speichern Sie die JSON-Datei wie folgt. Sie können den Inhalt löschen und ersetzen.

```
{
"Description":                      "CreateWPCDDeploymentGroup",
"VpcId":                            "VPC_ID",
"StackTemplateId":                  "stm-sp9lrk00000000000",
"Name":                             "WordPressCDAppGroup",
"TimeoutInMinutes":                 60,
"Parameters":   {
    "CodeDeployApplicationName":        "WordPressCDApp",
    "CodeDeployAutoScalingGroups":      ["ASG_NAME"],
    "CodeDeployDeploymentConfigName":   "CodeDeployDefault.HalfAtATime",
    "CodeDeployDeploymentGroupName":    "UNIQUE_CDDepGroupNAME",
    "CodeDeployServiceRoleArn":         "arn:aws:iam::ACCOUNT_ID:role/aws-
codedeploy-role"
    }
}
```

3. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner aus. Das Beispiel nennt sie Create CDDep GroupRfc .json.

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDDepGroupRfc.json
```

4. Ändern und speichern Sie die JSON-Datei wie folgt. Sie können den Inhalt löschen und ersetzen. Beachten Sie, dass `RequestedStartTime` und jetzt optional `RequestedEndTime` sind. Wenn

Sie sie ausschließen, wird der RFC ausgeführt, sobald er genehmigt wurde (was normalerweise automatisch geschieht). Um einen „geplanten“ RFC einzureichen, fügen Sie diese Werte hinzu.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2gd0u847qd9d2",
"Title":                "CD-Dep-Group-For-WP-Stack-RFC"
}
```

5. Erstellen Sie den RFC, indem Sie die CDDep GroupRfc Datei Create und die Datei mit den Ausführungsparametern angeben:

```
aws amscm create-rfc --cli-input-json file://CreateCDDepGroupRfc.json --execution-
parameters file://CreateCDDepGroupParams.json
```

In der Antwort erhalten Sie die RFC-ID des neuen RFC. Speichern Sie die ID für nachfolgende Schritte.

6. Reichen Sie den RFC ein:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Wenn der RFC erfolgreich ist, erhalten Sie keine Ausgabe.

7. Überprüfen Sie den RFC-Status:

```
aws amscm get-rfc --rfc-id RFC_ID
```

Laden Sie die Anwendung WordPress hoch

Sie haben automatisch Zugriff auf jede S3-Bucket-Instanz, die Sie erstellen. Sie können über Ihre Bastions (siehe Zugreifen auf Instances) oder über die S3-Konsole darauf zugreifen und das CodeDeploy Paket hochladen. Das Bundle muss vorhanden sein, um den Stack weiter bereitstellen zu können. Das Beispiel verwendet den zuvor erstellten Bucket-Namen.

```
aws s3 cp wordpress/wordpress.zip s3://ACCOUNT_ID-codedeploy-bundles/
```

Stellen Sie die WordPress Anwendung bereit mit CodeDeploy

Stellen Sie die CodeDeploy Anwendung bereit.

Sobald Sie Ihr CodeDeploy Anwendungspaket und Ihre Bereitstellungsgruppe haben, verwenden Sie diesen RFC, um die Anwendung bereitzustellen.

ERFORDERLICHE DATEN:

- `VPC-ID`: Die VPC, die Sie verwenden, sollte mit der zuvor verwendeten VPC identisch sein.
- `CodeDeployApplicationName`: Verwenden Sie den Namen für die CodeDeploy Anwendung, die Sie zuvor erstellt haben.
- `CodeDeployDeploymentGroupName`: Verwenden Sie den Namen der CodeDeploy Bereitstellungsgruppe, die Sie zuvor erstellt haben.
- `S3Location`(wo Sie das Anwendungspaket hochgeladen haben)`S3Bucket`:: BucketName Das, das Sie zuvor erstellt haben, `S3BundleType` und`S3Key`: Der Typ und der Name des Bundles, das Sie in Ihren S3-Store gestellt haben.
- `ChangeTypeIdUndChangeTypeVersion`: Die Änderungstyp-ID für diese exemplarische Vorgehensweise lautet`ct-2edc3sd1sqmrb`: Um die neueste Version herauszufinden, führen Sie diesen Befehl aus:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-2edc3sd1sqmrb
```

1. Geben Sie das JSON-Schema der Ausführungsparameter für das CodeDeploy Anwendungs-Deployment CT in eine Datei in Ihrem aktuellen Ordner aus. Das Beispiel nennt es Deploy CDApp Params.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2edc3sd1sqmrb" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeployCDAppParams.json
   ```

2. Ändern Sie die JSON-Datei wie folgt: Sie können den Inhalt löschen und ersetzen. Verwenden Sie für `S3Bucket` die`BucketName`, die Sie zuvor erstellt haben.

   ```
   {
   "Description":                      "Deploy WordPress CodeDeploy Application",
   "VpcId":                            "VPC_ID",
   "Name":                             "WP CodeDeploy Deployment Group",
   "TimeoutInMinutes":                 60,
   "Parameters":   {
       "CodeDeployApplicationName":        "WordPressCDApp",
       "CodeDeployDeploymentGroupName":    "WordPressCDDepGroup",
   ```

```
    "CodeDeployIgnoreApplicationStopFailures": false,
    "CodeDeployRevision": {
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket": "ACCOUNT_ID.BUCKET_NAME",
        "S3BundleType": "zip",
        "S3Key": "wordpress.zip" }
        }
    }
}
```

3. Geben Sie die JSON-Vorlage für CreateRfc in eine Datei in Ihrem aktuellen Ordner aus. Das Beispiel nennt sie Deploy CDApp rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeployCDAppRfc.json
```

4. Ändern und speichern Sie die Datei Deploy CDApp RFC.json. Sie können den Inhalt löschen und ersetzen.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-WP-Stack-RFC",
"RequestedStartTime":   "2017-04-28T22:45:00Z",
"RequestedEndTime":     "2017-04-28T22:45:00Z"
}
```

5. Erstellen Sie den RFC und geben Sie dabei die Ausführungsparameterdatei und die CDApp Deploy-RFC-Datei an:

```
aws amscm create-rfc --cli-input-json file://DeployCDAppRfc.json  --execution-
parameters file://DeployCDAppParams.json
```

In der Antwort erhalten Sie den RfcId des neuen RFC. Speichern Sie die ID für nachfolgende Schritte.

6. Reichen Sie den RFC ein:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Wenn der RFC erfolgreich ist, erhalten Sie keine Ausgabe.

Überprüfen Sie die Anwendungsbereitstellung

Navigieren Sie zum Endpunkt (ELB CName) des zuvor erstellten Load Balancers mit dem WordPress bereitgestellten Pfad:/. WordPress Beispiel:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Machen Sie die Anwendungsbereitstellung rückgängig

Um die Bereitstellung zu beenden, reichen Sie den Delete Stack CT gegen den RDS-Datenbank-Stack, den Application Load Balancer, die Auto Scaling Scaling-Gruppe, den S3-Bucket und die Code Deploy-Anwendung und -Gruppe ein — insgesamt sechs RFCs . Darüber hinaus können Sie eine Serviceanfrage für das Löschen der RDS-Snapshots stellen (sie werden nach zehn Tagen automatisch gelöscht, kosten aber nur einen geringen Betrag). Sammeln Sie den Stapel IDs für alle und folgen Sie dann diesen Schritten.

Diese exemplarische Vorgehensweise bietet ein Beispiel für die Verwendung der AMS-Konsole zum Löschen eines S3-Stacks. Dieses Verfahren gilt für das Löschen eines beliebigen Stacks mithilfe der AMS-Konsole.

> **Note**
>
> Wenn Sie einen S3-Bucket löschen, müssen Sie ihn zuerst von Objekten leeren.

ERFORDERLICHE DATEN:

- `StackId`: Der zu verwendende Stapel. Sie finden ihn auf der Seite AMS Console Stacks, die Sie über einen Link im linken Navigationsbereich aufrufen können. Führen Sie mithilfe der AMS SKMS API/CLI den Vorgang For the AMS SKMS API reference, see the Reports in der AWS Artifact Console. (in der CLI) aus. `list-stack-summaries`
- Die Änderungstyp-ID für diese exemplarische Vorgehensweise lautet `ct-0q0bic0ywqk6c` „1.0“. Führen Sie den folgenden Befehl aus, um die neueste Version herauszufinden:

  ```
  aws amscm list-change-type-version-summaries --filter
   Attribute=ChangeTypeId,Value=ct-0q0bic0ywqk6c
  ```

INLINE-ERSTELLUNG:

- Geben Sie den Befehl create RFC mit den direkt angegebenen Ausführungsparametern aus (vermeiden Sie Anführungszeichen, wenn Sie die Ausführungsparameter inline angeben). E

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0"
 --title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

- Senden Sie den RFC mit der RFC-ID, die bei der RFC-Erstellung zurückgegeben wurde. Bis zur Übermittlung verbleibt der RFC im `Editing` Status und es wird nicht darauf reagiert.

```
aws amscm submit-rfc --rfc-id RFC_ID
```

- Überwachen Sie den RFC-Status und sehen Sie sich die Ausführungsausgabe an:

```
aws amscm get-rfc --rfc-id RFC_ID
```

VORLAGE ERSTELLEN:

1. Gibt die RFC-Vorlage in eine Datei in Ihrem aktuellen Ordner aus. Beispiel nennt sie DeleteStackRfc .json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteStackRfc.json
   ```

2. Ändern und speichern Sie die DeleteStackRfc JSON-Datei. Da das Löschen eines Stacks nur einen Ausführungsparameter hat, können sich die Ausführungsparameter in der DeleteStackRfc JSON-Datei selbst befinden (es ist nicht erforderlich, eine separate JSON-Datei mit Ausführungsparametern zu erstellen).

   Die internen Anführungszeichen in der ExecutionParameters JSON-Erweiterung müssen mit einem Backslash (\) maskiert werden. Beispiel ohne Start- und Endzeit:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0q0bic0ywqk6c",
   "Title":                "Delete-My-Stack-RFC"
   "ExecutionParameters":  "{
           \"StackId\":\"STACK_ID\"}"
   }
   ```

3. Erstellen Sie den RFC:

```
aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
```

In der Antwort erhalten Sie den RfcId des neuen RFC. Beispiel:

```
{
"RfcId": "daaa1867-ffc5-1473-192a-842f6b326102"
}
```

Speichern Sie die ID für nachfolgende Schritte.

4. Reichen Sie den RFC ein:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Wenn der RFC erfolgreich ist, erhalten Sie keine Bestätigung in der Befehlszeile.

5. Um den Status der Anfrage zu überwachen und die Ausführungsausgabe anzuzeigen:

```
aws amscm get-rfc --rfc-id RFC_ID --query "Rfc.
{Status:Status.Name,Exec:ExecutionOutput}" --output table
```

Wartung der Anwendung

Sobald die Infrastruktur bereitgestellt ist, besteht die Herausforderung darin, sie in all Ihren AMS-Umgebungen, von der Qualitätssicherung über die Bereitstellung bis hin zur Produktion, einheitlich zu aktualisieren.

Dieser Abschnitt bietet einen Überblick über den AMS-Workload-Erfassungsprozess und einige Beispiele für verschiedene Methoden, mit denen Sie Ihre Cloud-Infrastrukturebene auf dem neuesten Stand halten können.

Strategien zur Anwendungswartung

Die Art und Weise, wie Sie Ihre Anwendungen bereitstellen, wirkt sich darauf aus, wie Sie sie verwalten. Dieser Abschnitt enthält einige Strategien für die Anwendungswartung.

Umgebungsupdates können jede der folgenden Änderungen beinhalten:

- Sicherheits-Updates
- Neue Versionen Ihrer Anwendungen
- Änderungen der Anwendungskonfiguration
- Aktualisierungen der Abhängigkeiten

> **Note**
>
> Reichen Sie bei jeder Anwendungsbereitstellung, unabhängig von der Methode, immer im Voraus eine Serviceanfrage ein, um AMS darüber zu informieren, dass Sie eine Anwendung bereitstellen werden.

Beispiele für die Installation unveränderlicher und veränderbarer Anwendungen

Veränderlichkeit von Instanzen berechnen	Methode zur Installation der App	AMI
Mutable	Mit CodeDeploy	Von AMS bereitgestellt
	manuell	

Veränderlichkeit von Instanzen berechnen	Methode zur Installation der App	AMI
	Mit einem Koch oder einer Marionette, Pull-Based	
	Mit Ansible oder Salt, Push-basiert	
Immutable (Unveränderlich)	Mit einem goldenen AMI	Benutzerdefiniert (basierend auf der von AMS bereitgestellten Version)

Veränderbare Bereitstellung mit einem CodeDeploy -fähigen AMI

AWS CodeDeploy ist ein Service, der Codebereitstellungen für jede Instance automatisiert, einschließlich EC2 Amazon-Instances und Instances, die lokal ausgeführt werden. Sie können es CodeDeploy zusammen mit AMS verwenden, um eine Anwendung zu erstellen und bereitzustellen. CodeDeploy Beachten Sie, dass AMS ein Standard-Instanzprofil für CodeDeploy Anwendungen bereitstellt.

- Amazon Linux (Version 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

Vor der ersten Verwendung CodeDeploy müssen Sie eine Reihe von Einrichtungsschritten ausführen:

1. Installieren oder aktualisieren Sie die AWS-CLI
2. Erstellen Sie eine Service Role für AWS CodeDeploy, Sie verwenden die Service Role ARN in der Bereitstellung

IDs Informationen zu allen CT-Optionen finden Sie in der Change Type Reference.

> **Note**
>
> Derzeit müssen Sie Amazon S3 S3-Speicher mit dieser Lösung verwenden.

Die grundlegenden Schritte werden hier beschrieben und das Verfahren ist im AMS-Benutzerhandbuch detailliert beschrieben.

1. Erstellen Sie einen Amazon S3 S3-Speicher-Bucket. CT: ct-1a68ck03fn98r. Für den S3-Bucket muss die Versionierung aktiviert sein (Informationen dazu finden Sie unter Bucket Versioning aktivieren).
2. Legen Sie Ihre CodeDeploy gebündelten Artefakte darauf. Sie können dies mit der Amazon S3 S3-Konsole tun, ohne den Zugriff über AMS anfordern zu müssen. Oder mit einer Variante dieses Befehls:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Finden Sie ein `customer-` AMS-AMI; verwenden Sie entweder:

   - AMS-Konsole: Die VPC-Detailseite für die entsprechende VPC
   - AMS-API Die AMS SKMS-API-Referenz finden Sie auf der Registerkarte Berichte in der AWS Artifact Console. oder CLI: `aws amsskms list-amis`

4. Erstellen Sie eine Autoscaling-Gruppe (ASG). CT: ct-2tylseo8rxfsc. Geben Sie das AMS-AMI an, stellen Sie den Load Balancer so ein, dass er offene Ports hat, spezifizieren Sie `customer-mc-ec2-instance-profile` für die`ASGIAMInstanceProfile`.
5. Erstellen Sie Ihre CodeDeploy Anwendung. CT: ct-0ah3gwb9seqk2. Zu den Parametern gehört beispielsweise ein Anwendungsname. `WordpressProd`
6. Erstellen Sie Ihre CodeDeploy Bereitstellungsgruppe. CT: ct-2gd0u847qd9d2. Zu den Parametern gehören Ihr CodeDeploy Anwendungsname, der ASG-Name, der Name des Konfigurationstyps und der ARN der Servicerolle.
7. Stellen Sie die CodeDeploy Anwendung bereit. CT: ct-2edc3sd1sqmrb. Zu den Parametern gehören Ihr CodeDeploy Anwendungsname, der Name des Konfigurationstyps, der Name der Bereitstellungsgruppe, der Revisionstyp und der S3-Bucket-Speicherort, an dem sich die Artefakte befinden. CodeDeploy

Veränderbare Bereitstellung, manuell konfigurierte und aktualisierte Anwendungsinstanzen

Bei dieser Strategie zur Anwendungsbereitstellung handelt es sich um eine einfache und manuelle Aktualisierung von Anwendungsinstanzen. Dies sind die grundlegenden Schritte.

IDs Informationen zu allen CT-Optionen finden Sie in der Change Type Reference.

> **Note**
>
> Derzeit müssen Sie Amazon S3 S3-Speicher mit dieser Lösung verwenden.

Die grundlegenden Schritte werden hier beschrieben. Die verschiedenen Verfahren sind im AMS-Benutzerhandbuch detailliert beschrieben.

1. Erstellen Sie einen Amazon S3 S3-Speicher-Bucket. CT: ct-1a68ck03fn98r. Für den S3-Bucket muss die Versionierung aktiviert sein (Informationen dazu finden Sie unter Bucket Versioning aktivieren).
2. Platzieren Sie Ihre gebündelten Anwendungsartefakte darauf (alles, was Ihre Anwendung benötigt, um beim Booten zu starten und zu funktionieren). Sie können dies mit der Amazon S3 S3-Konsole tun, ohne den Zugriff über AMS anfordern zu müssen. Oder mit einer Variante dieses Befehls:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Finden Sie ein AMS-AMI, alles wird es CodeDeploy bei sich haben. Um ein „Kunden-“ -AMI zu finden, verwenden Sie entweder:

   - AMS-Konsole: Die VPC-Detailseite für die entsprechende VPC
   - AMS-API Die AMS SKMS-API-Referenz finden Sie auf der Registerkarte Berichte in der AWS Artifact Console. oder CLI: `aws amsskms list-amis`

4. Erstellen Sie eine EC2 Instanz mit diesem AMI. CT: ct-14027q0sjyt1h. Geben Sie das AMS-AMI an, legen Sie ein Tag fest `Key=backup, Value=true` und geben Sie das `customer-mc-ec2-instance-profile` für den `InstanceProfile` Parameter an. Notieren Sie sich die zurückgegebene Instanz-ID.

5. Fordern Sie Administratorzugriff auf die Instanz an. CT: ct-1dmlg9g1l91h6. Sie benötigen den FQDN für Ihr Konto. Wenn Sie sich nicht sicher sind, wie Ihr FQDN lautet, können Sie ihn wie folgt finden:

   - Verwenden der AWS-Managementkonsole für Verzeichnisdienste (unter Sicherheit und Identität), Registerkarte Verzeichnisname.
   - Ausführen eines dieser Befehle (Rückgabe von Verzeichnisklassen; DC+DC+DC=FQDN): Windows: oder Linux:. `whoami /fqdn` `hostname --fqdn`

6. Melden Sie sich bei der Instance an. Weitere Informationen finden Sie unter Zugreifen auf Instances via Bastions im AMS-Benutzerhandbuch.
7. Laden Sie Ihre gebündelten Anwendungsdateien von Ihrem S3-Bucket auf die Instance herunter.
8. Fordern Sie ein sofortiges Backup mit einer Serviceanfrage an AMS an. Sie müssen die Instanz-ID kennen.
9. Wenn Sie Ihre Anwendung aktualisieren müssen, laden Sie neue Dateien in Ihren S3-Bucket und folgen Sie dann den Schritten 3 bis 8.

Veränderbare Bereitstellung mit einem mit einem Pull-basierten Bereitstellungstool konfigurierten AMI

Diese Strategie basiert auf dem `InstanceUserData` Parameter im Managed Services Create EC2 CT. Weitere Informationen zur Verwendung dieses Parameters finden Sie unter Instanzen mit Benutzerdaten konfigurieren. In diesem Beispiel wird von einem Pull-basierten Anwendungsbereitstellungstool wie Chef oder Puppet ausgegangen.

Der CodeDeploy Agent wird auf allen AMS unterstützt. AMIs Hier ist die Liste der unterstützten AMIs:

- Amazon Linux (Version 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

IDs Informationen zu allen CT-Optionen finden Sie in der Change Types-Referenz.

> **Note**
>
> Derzeit müssen Sie Amazon S3 S3-Speicher mit dieser Lösung verwenden.

Die grundlegenden Schritte werden hier beschrieben und das Verfahren ist im AMS-Benutzerhandbuch detailliert beschrieben.

1. Erstellen Sie einen Amazon S3 S3-Speicher-Bucket. CT: ct-1a68ck03fn98r. Für den S3-Bucket muss die Versionierung aktiviert sein (Informationen dazu finden Sie unter Bucket Versioning aktivieren).
2. Platziere deine CodeDeploy gebündelten Artefakte darauf. Sie können dies mit der Amazon S3 S3-Konsole tun, ohne den Zugriff über AMS anfordern zu müssen. Oder mit einer Variante dieses Befehls:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Suchen Sie ein `customer-` AMS-AMI; verwenden Sie entweder:

   - AMS-Konsole: Die VPC-Detailseite für die entsprechende VPC
   - AMS-API Die AMS SKMS-API-Referenz finden Sie auf der Registerkarte Berichte in der AWS Artifact Console. oder CLI: `aws amsskms list-amis`

4. Erstellen Sie eine Instanz. EC2 CT: ct-14027q0sjyt1h; setzen Sie ein Tag und verwenden Sie den `InstanceUserData` Parameter`Key=backup, Value=true`, um einen Bootstrap und andere Skripte (Chef/Puppet Download-Agent usw.) anzugeben und die erforderlichen Autorisierungsschlüssel hinzuzufügen. Ein Beispiel hierfür finden Sie im AMS-Benutzerhandbuch im Abschnitt Change Management — Beispiele für die Erstellung einer zweistufigen HA-Bereitstellung. Alternativ können Sie Zugriff auf die Instance anfordern, sich bei ihr anmelden und sie mit den erforderlichen Bereitstellungsartefakten konfigurieren. Denken Sie daran, dass Pull-basierte Bereitstellungsbefehle von den Agenten auf Ihren Instances an Ihren Unternehmens-Masterserver weitergeleitet werden und möglicherweise eine Autorisierung benötigen, um Bastionen zu durchlaufen. Möglicherweise benötigen Sie eine Serviceanfrage an AMS, um den Zugriff auf group/AD Sicherheitsgruppen ohne Bastionen zu beantragen.
5. Wiederholen Sie Schritt 4, um eine weitere EC2 Instanz zu erstellen und sie mit dem Master-Server des Deployment Tools zu konfigurieren.
6. Wenn Sie Ihre Anwendung aktualisieren müssen, verwenden Sie das Bereitstellungstool, um die Updates für Ihre Instanzen bereitzustellen.

Veränderbare Bereitstellung mit einem mit einem Push-basierten Bereitstellungstool konfigurierten AMI

Diese Strategie basiert auf dem `InstanceUserData` Parameter im Managed Services Create EC2 CT. Weitere Informationen zur Verwendung dieses Parameters finden Sie unter Instanzen mit Benutzerdaten konfigurieren. In diesem Beispiel wird von einem Pull-basierten Anwendungsbereitstellungstool wie Chef oder Puppet ausgegangen.

IDs Informationen zu allen CT-Optionen finden Sie in der Change Type Reference.

> **Note**
>
> Derzeit müssen Sie Amazon S3 S3-Speicher mit dieser Lösung verwenden.

Die grundlegenden Schritte werden hier beschrieben und das Verfahren ist im AMS-Benutzerhandbuch detailliert beschrieben.

1. Erstellen Sie einen Amazon S3 S3-Speicher-Bucket. CT: ct-1a68ck03fn98r. Für den S3-Bucket muss die Versionierung aktiviert sein (Informationen dazu finden Sie unter Bucket Versioning aktivieren).
2. Legen Sie Ihre CodeDeploy gebündelten Artefakte darauf. Sie können dies mit der Amazon S3 S3-Konsole tun, ohne den Zugriff über AMS anfordern zu müssen. Oder mit einer Variante dieses Befehls:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Finden Sie ein AMS-AMI, alles wird es CodeDeploy bei sich haben. Um ein „Kunden-“ -AMI zu finden, verwenden Sie entweder:

   - AMS-Konsole: Die VPC-Detailseite für die entsprechende VPC
   - AMS-API Die AMS SKMS-API-Referenz finden Sie auf der Registerkarte Berichte in der AWS Artifact Console. oder CLI: `aws amsskms list-amis`

4. Erstellen Sie eine Instanz. EC2 CT: ct-14027q0sjyt1h; setzen Sie ein Tag und verwenden Sie den `InstanceUserData` ParameterKey=backup, Value=true, um einen Bootstrap und andere Skripts auszuführen, darunter Autorisierungsschlüssel, SALT-Stack (Bootstrap einen Minion — weitere Informationen finden Sie unter Bootstrapping Salt auf Linux EC2 mit Cloud-

Init) oder Ansible (Installation eines Schlüsselpaars — weitere Informationen finden Sie unter Erste Schritte mit Ansible und Dynamic Amazon Inventory Management). EC2 Fordern Sie alternativ Zugriff auf die Instanz an, melden Sie sich bei ihr an und konfigurieren Sie sie mit den erforderlichen Bereitstellungsartefakten. Denken Sie daran, dass Push-Befehle aus Ihrem Unternehmenssubnetz an Ihre Instances gesendet werden und dass Sie möglicherweise die Autorisierung konfigurieren müssen, damit sie Bastionen passieren können. Möglicherweise benötigen Sie eine Serviceanfrage an AMS, um den Zugriff auf group/AD Sicherheitsgruppen ohne Bastionen zu beantragen.

5. Wiederholen Sie Schritt 4, um eine weitere EC2 Instanz zu erstellen und sie mit dem Master-Server des Deployment Tools zu konfigurieren.
6. Wenn Sie Ihre Anwendung aktualisieren müssen, verwenden Sie das Bereitstellungstool, um die Updates für Ihre Instanzen bereitzustellen.

Unveränderlicher Einsatz mit einem goldenen AMI

Diese Strategie verwendet ein „goldenes“ AMI, das Sie so konfiguriert haben, dass es sich so verhält, wie Sie es sich für alle Ihre Anwendungsinstanzen wünschen. Die mit diesem Golden AMI erstellten Instances würden sich beispielsweise selbst mit der richtigen Domain und dem richtigen DNS verbinden, alle erforderlichen Systeme selbst konfigurieren, neu starten und starten. Wenn Sie Ihre Anwendungsinstanzen aktualisieren möchten, erstellen Sie das Goldene AMI neu und führen damit ganz neue Anwendungsinstanzen ein.

Der CodeDeploy Agent wird auf allen AMS unterstützt. AMIs Hier ist die Liste der unterstützten AMIs:

- Amazon Linux (Version 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

IDs Informationen zu allen CT-Optionen finden Sie in der Change Type Reference.

> **Note**
>
> Derzeit müssen Sie Amazon S3 S3-Speicher mit dieser Lösung verwenden.

1. Erstellen Sie einen Amazon S3 S3-Speicher-Bucket. CT: ct-1a68ck03fn98r. Für den S3-Bucket muss die Versionierung aktiviert sein (Informationen dazu finden Sie unter Bucket Versioning aktivieren).
2. Platzieren Sie Ihre gebündelten Anwendungsartefakte darauf (alles, was Ihre Anwendung benötigt, um beim Booten zu starten und zu funktionieren). Sie können dies mit der Amazon S3 S3-Konsole tun, ohne den Zugriff über AMS anfordern zu müssen. Oder mit einer Variante dieses Befehls:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Finden Sie ein `customer-` AMS-AMI; verwenden Sie entweder:

   - AMS-Konsole: Die VPC-Detailseite für die entsprechende VPC
   - AMS-API Die AMS SKMS-API-Referenz finden Sie auf der Registerkarte Berichte in der AWS Artifact Console. oder CLI: `aws amsskms list-amis`

4. Erstellen Sie eine EC2 Instanz mit diesem AMI. CT: ct-14027q0sjyt1h. Geben Sie das AMS-AMI an, legen Sie ein Tag fest `Key=backup, Value=true` und spezifizieren Sie `customer-mc-ec2-instance-profile` für`InstanceProfile`. Notieren Sie sich die zurückgegebene Instanz-ID.
5. Fordern Sie Administratorzugriff auf die Instanz an. CT: ct-1dmlg9g1l91h6. Sie benötigen den FQDN für Ihr Konto. Wenn Sie sich nicht sicher sind, wie Ihr FQDN lautet, können Sie ihn wie folgt finden:

   - Verwenden der AWS-Managementkonsole für Verzeichnisdienste (unter Sicherheit und Identität), Registerkarte Verzeichnisname.
   - Ausführen eines dieser Befehle (Rückgabe von Verzeichnisklassen; DC+DC+DC=FQDN): Windows: oder Linux:. `whoami /fqdn` `hostname --fqdn`

6. Melden Sie sich bei der Instance an, siehe Accessing Instances im AMS-Benutzerhandbuch.
7. Laden Sie Ihre gebündelten Anwendungsdateien aus Ihrem S3-Bucket auf die Instance herunter. Konfigurieren Sie die Instance so, dass sie die voll funktionsfähige Anwendung beim Booten selbst bereitstellt.
8. Erstellen Sie das goldene AMI auf der Instance. CT: ct-3rqqu43krekby. Einzelheiten finden Sie unter AMI | Create.
9. Konfigurieren Sie eine Auto Scaling Scaling-Gruppe, um mithilfe dieses AMI neue Instances zu erstellen. CT: ct-2tylseo8rxfsc. Wenn Sie Ihre Anwendung aktualisieren müssen, gehen Sie wie

folgt vor und fordern Sie AMS auf, die ASG so zu aktualisieren, dass sie das neue goldene AMI verwendet. Verwenden Sie hierfür ein Management | Other | Other | Update CT.

Strategien aktualisieren

Es gibt verschiedene Strategien, mit denen Sie Ihre Anwendungen oder Instanzen in Ihrer AMS-verwalteten Umgebung aktualisieren können.

- Geplante Ausfallzeit: Diese einfache Strategie beinhaltet die Planung einer Zeit, in der Ihre Anwendung offline ist und manuell aktualisiert wird. Senden Sie dazu eine Management | Other | Other | Update CT (ct-0xdawir96cy7k) -Anforderung, um die erforderlichen Instanzen zu beenden. Nehmen Sie die erforderlichen Aktualisierungen vor und reichen Sie dann eine weitere Management | Other | Other | Update CT (ct-0xdawir96cy7k) -Anforderung ein, um die Instances zu starten.

- Blau/Grün: Diese Strategie setzt voraus, dass Sie über eine redundante Umgebung (zwei vollständig funktionsfähige Umgebungen) verfügen und eine Umgebung offline schalten, indem Sie DNS- (Domain Name System) - oder Web-Firewall (WAF) -Updates verwenden, um den Datenverkehr umzuleiten. Aktualisieren Sie eine Umgebung und leiten Sie sie dann erneut um, um die andere Umgebung zu aktualisieren.

  Weitere Informationen finden Sie unter AWS CodeDeploy stellt Blue/Green Bereitstellungen vor.
- Laufendes Update mit neuem AMI: Hier haben Sie ein neues AMI, das Sie anpassen (siehe AMI erstellen) und dann beantragen, dass AMS es für Ihre Auto Scaling Scaling-Gruppe bereitstellt. Verwenden Sie dazu ein Management | Other | Other | Update CT (ct-0xdawir96cy7k).

Ressourcenplaner für AWS Managed Services

Verwenden Sie den AWS Managed Services (AMS) Resource Scheduler, um den automatischen Start und Stopp von AutoScaling Gruppen, EC2 Amazon-Instances und RDS-Instances in Ihrem Konto zu planen. Dies trägt dazu bei, die Infrastrukturkosten dort zu senken, wo die Ressourcen nicht rund um die Uhr laufen sollen. Die Lösung baut auf Instance Scheduler auf AWS, enthält jedoch zusätzliche Funktionen und Anpassungen, die speziell auf die AMS-Bedürfnisse zugeschnitten sind.

> **Note**
>
> Standardmäßig interagiert AMS Resource Scheduler nicht mit Ressourcen, die nicht Teil eines Stacks sind. AWS CloudFormation Die Ressource muss Teil eines Stacks sein, der mit „stack-“, „sc-“ oder „SC-“ beginnt. Um die Ressourcen zu planen, die nicht Teil eines CloudFormation Stacks sind, können Sie den Stackparameter Resource Scheduler `ScheduleNonStackResources` auf `Yes` aktualisieren.

AMS Resource Scheduler verwendet Zeiträume und Zeitpläne:

- Perioden definieren die Zeiten, zu denen Resource Scheduler ausgeführt wird, z. B. Startzeit, Endzeit und Tage des Monats.
- Zeitpläne enthalten Ihre definierten Zeiträume sowie zusätzliche Konfigurationen wie SSM-Wartungsfenster, Zeitzone, Ruhezustand usw. und geben anhand der konfigurierten Periodenregeln an, wann Ressourcen ausgeführt werden sollen.

Sie können diese Zeiträume und Zeitpläne mithilfe der automatisierten Änderungstypen () von AMS Resource Scheduler konfigurieren. CTs

Vollständige Informationen zu den für AMS Resource Scheduler verfügbaren Einstellungen finden Sie in der entsprechenden AWS Instance Scheduler-Dokumentation unter Lösungskomponenten. Eine architektonische Ansicht der Lösung finden Sie in der entsprechenden AWS Instance Scheduler-Dokumentation unter Architecture overview.html.

Bereitstellung von AMS Resource Scheduler

Um AMS Resource Scheduler bereitzustellen, verwenden Sie den automatisierten Änderungstyp (CT): Deployment | AMS Resource Scheduler | Solution | Deploy (ct-0ywnhc8e5k9z5), um einen RFC auszulösen, der dann die Lösung in Ihrem Konto bereitstellt. Sobald der RFC ausgeführt wurde, wird Ihrem Konto automatisch ein CloudFormation Stack bereitgestellt, der AMS Resource Scheduler-Ressourcen mit Standardkonfiguration enthält. Weitere Informationen zu den Änderungstypen von Resource Scheduler finden Sie unter AMS Resource Scheduler.

> **Note**
>
> Um herauszufinden, ob AMS Resource Scheduler bereits in Ihrem Konto bereitgestellt ist, überprüfen Sie die AWS Lambda-Konsole für dieses Konto und suchen Sie nach der AMSResourceScheduler-Funktion.

Nachdem der AMS Resource Scheduler in Ihrem Konto bereitgestellt wurde, empfehlen wir Ihnen, die Standardkonfiguration zu überprüfen und bei Bedarf Konfigurationen wie Tag-Schlüssel, Zeitzone, geplante Dienste usw. an Ihre Präferenzen anzupassen. Einzelheiten zu den empfohlenen Anpassungen finden Sie unter Weiter. AMS Resource Scheduler anpassen

Um die benutzerdefinierten Konfigurationen vorzunehmen oder einfach die Resource Scheduler-Konfiguration zu bestätigen,

AMS Resource Scheduler anpassen

Wir empfehlen Ihnen, die folgenden Eigenschaften von AMS Resource Scheduler mithilfe der aktualisierten AMS Resource Scheduler-Änderungstypen anzupassen, siehe AMS Resource Scheduler.

- Tagname: Der Name des Tags, das Resource Scheduler verwendet, um Instanzpläne mit Ressourcen zu verknüpfen. Der Standardwert ist Schedule.
- Geplante Dienste: Eine durch Kommas getrennte Liste von Diensten, die Resource Scheduler verwalten kann. Der Standardwert ist „ec2, rds, autoscaling“. Gültige Werte sind „ec2", „rds“ und „autoscaling“
- Standardzeitzone: Geben Sie die Standardzeitzone an, die der Resource Scheduler verwenden soll. Der Standardwert ist UTC.
- Verwenden Sie CMK: Eine durch Kommas getrennte Liste von Amazon KMS Customer Managed Key (CMK) ARNs , für die Resource Scheduler Berechtigungen erteilt werden können.
- Verwendung LicenseManager: Eine durch Kommas getrennte Liste der AWS Lizenzmanager, für die Resource Scheduler ARNs Berechtigungen erteilt werden können.

> **Note**
>
> AMS kann von Zeit zu Zeit Funktionen und Korrekturen veröffentlichen, um AMS Resource Scheduler in Ihrem Konto auf dem neuesten Stand zu halten. In diesem Fall bleiben alle Anpassungen, die Sie am AMS Resource Scheduler vornehmen, erhalten.

AMS Resource Scheduler verwenden

Um AMS Resource Scheduler nach der Bereitstellung der Lösung zu konfigurieren, verwenden Sie den automatisierten Resource Scheduler, CTs um AMS Resource Scheduler-Perioden (die Zeiten, zu denen Resource Scheduler ausgeführt wird) und Zeitpläne (die konfigurierten Zeiträume und andere Optionen) zu erstellen, zu löschen, zu aktualisieren und zu beschreiben (Details zu erhalten). Ein Beispiel für die Verwendung der Änderungstypen des AMS Resource Scheduler finden Sie unter AMS Resource Scheduler.

Um Ressourcen auszuwählen, die von AMS Resource Scheduler verwaltet werden sollen, verwenden Sie nach der Bereitstellung und Erstellung des Zeitplans das AMS-Tag Create, CTs um Auto Scaling Scaling-Gruppen, Amazon RDS-Stacks und EC2 Amazon-Ressourcen mit dem Tag-Schlüssel, den Sie bei der Bereitstellung angegeben haben, und dem definierten Zeitplan als Tag-Wert zu taggen. Nachdem die Ressourcen markiert wurden, werden die Ressourcen gemäß Ihrem definierten Resource Scheduler-Zeitplan für den Start oder Stopp geplant.

Für die Nutzung von AMS Resource Scheduler fallen keine zusätzlichen Kosten an. Die Lösung verwendet jedoch mehrere, AWS-Services und diese Ressourcen werden Ihnen je nach Nutzung in Rechnung gestellt. Weitere Informationen finden Sie unter Überblick über die Architektur.

Um AMS Resource Scheduler zu deaktivieren:

- Zur vorübergehenden Deaktivierung oder Deaktivierung: Senden Sie einen RFC über den automatisierten Änderungstyp Management | AMS Resource Scheduler | State | Disable (ct-14v49adibs4db)
- Zur dauerhaften Entfernung: Reichen Sie einen RFC für Management | Sonstige | Andere | Aktualisierung (Überprüfung erforderlich) (ct-0xdawir96cy7k) ein, der die Entfernung aus dem Release-Automatisierungssystem von Resource Scheduler anfordert

Kostenschätzer für AMS Resource Scheduler

Um Kosteneinsparungen nachzuverfolgen, bietet AMS Resource Scheduler eine Komponente, die stündlich die geschätzten Kosteneinsparungen für Amazon EC2 - und RDS-Ressourcen berechnet, die vom Scheduler verwaltet werden. Diese Daten zu den Kosteneinsparungen werden dann als CloudWatch Metrik (`AMS/ResourceScheduler`) veröffentlicht, um Ihnen zu helfen, sie nachzuverfolgen. Der Kosteneinsparungsschätzer schätzt nur die Einsparungen bei den Betriebsstunden der Instance. Andere Kosten, wie z. B. die mit einer Ressource verbundenen Datenübertragungskosten, werden nicht berücksichtigt.

Der Kosteneinsparungsrechner ist mit Resource Scheduler aktiviert. Er läuft stündlich und ruft Kosten- und Nutzungsdaten von ab. AWS Cost Explorer Aus diesen Daten berechnet es die durchschnittlichen Kosten pro Stunde für jeden Instanztyp und prognostiziert dann die Kosten für einen ganzen Tag, wenn die Instanz ohne Zeitplan ausgeführt wurde. Die Kosteneinsparungen sind die Differenz zwischen den prognostizierten Kosten und den tatsächlich gemeldeten Kosten aus dem Cost Explorer für einen bestimmten Tag.

Wenn Instanz A beispielsweise so konfiguriert ist, dass Resource Scheduler von 9.00 Uhr bis 17.00 Uhr ausgeführt wird, sind das acht Stunden an einem bestimmten Tag. Cost Explorer meldet die Kosten mit 1$ und die Nutzung mit 8. Die durchschnittlichen Kosten pro Stunde betragen daher 0,125$. Wenn die Instanz nicht mit Resource Scheduler geplant wurde, würde die Instanz an diesem Tag rund um die Uhr laufen. In diesem Fall hätten die Kosten 24x0,125 = 3$ betragen. Resource Scheduler hat Ihnen geholfen, Kosteneinsparungen von 2$ zu erzielen.

Damit der Kosteneinsparungsschätzer Kosten und Nutzung nur für Ressourcen abrufen kann, die von Resource Scheduler aus dem Cost Explorer verwaltet werden, muss der Tag-Schlüssel, den Resource Scheduler für die Ausrichtung von Ressourcen verwendet, als Kostenzuordnungs-Tag im Fakturierungs-Dashboard aktiviert werden. Wenn das Konto zu einer Organisation gehört, muss der Tag-Schlüssel im Verwaltungskonto der Organisation aktiviert werden. Informationen dazu finden Sie unter Benutzerdefinierte Kostenverrechnungs-Tags und Benutzerdefinierte Kostenverrechnungs-Tags aktivieren

Nachdem der Tag-Schlüssel als Cost Allocation Tag aktiviert wurde, beginnt die AWS Fakturierung mit der Erfassung der Kosten und der Nutzung der von Resource Scheduler verwalteten Ressourcen. Sobald diese Daten verfügbar sind, beginnt der Kosteneinsparungsrechner mit der Berechnung der Kosteneinsparungen und veröffentlicht die Daten unter dem `AMS/ResourceScheduler` Metrik-Namespace in. CloudWatch

Tipps für Kostenschätzer

Der Cost Savings Estimator akzeptiert keine Rabatte wie Reserved Instances, Sparpläne usw., die bei der Berechnung berücksichtigt werden. Der Estimator verwendet die Nutzungskosten aus dem Cost Explorer und berechnet die durchschnittlichen Kosten pro Stunde für die Ressourcen. Weitere Informationen finden Sie unter Grundlegendes zu Ihren AWS Kostendatensätzen: Ein Spickzettel

Damit der Kosteneinsparungsschätzer Kosten und Nutzung nur für Ressourcen abrufen kann, die von Resource Scheduler aus dem Cost Explorer verwaltet werden, muss der Tag-Schlüssel, den Resource Scheduler für die Ausrichtung von Ressourcen verwendet, im Fakturierungs-Dashboard als Cost Allocation Tag aktiviert werden. Wenn das Konto zu einer Organisation gehört, muss der Tag-Schlüssel im Verwaltungskonto der Organisation aktiviert werden. Informationen dazu finden Sie unter Benutzerdefinierte Tags für die Kostenzuweisung. Wenn das Kostenzuweisungs-Tag nicht aktiviert ist, kann der Kalkulator die Einsparungen nicht berechnen und die Metrik nicht veröffentlichen, auch wenn er aktiviert ist.

Bewährte Methoden für AMS Resource Scheduler

Planung von EC2 Amazon-Instances

- Das Verhalten beim Herunterfahren der Instance muss auf `stop` und nicht auf eingestellt seinterminate. Dies ist `stop` für Instances voreingestellt, die mit dem automatisierten Änderungstyp AMS Amazon EC2 Create (ct-14027q0sjyt1h) erstellt wurden, und kann für EC2 Amazon-Instances, die mit Ingestion erstellt wurden, festgelegt werden, indem die Eigenschaft auf gesetzt wird. AWS CloudFormation `InstanceInitiatedShutdownBehavior` `stop` Wenn für Instances das Verhalten beim Herunterfahren auf eingestellt ist, werden die Instances beendetterminate, wenn der Resource Scheduler sie stoppt und der Scheduler sie nicht wieder starten kann.
- EC2 Amazon-Instances, die Teil einer Auto Scaling Scaling-Gruppe sind, werden vom AMS Resource Scheduler nicht einzeln verarbeitet, auch wenn sie markiert sind.
- Wenn das Root-Volume der Ziel-Instance mit einem KMS-Kundenhauptschlüssel (CMK) verschlüsselt ist, muss Ihrer Resource Scheduler IAM-Rolle eine zusätzliche `kms:CreateGrant` Berechtigung hinzugefügt werden, damit der Scheduler solche Instances starten kann. Diese Berechtigung wird der Rolle standardmäßig nicht hinzugefügt, um die Sicherheit zu erhöhen. Wenn Sie diese Berechtigung benötigen, reichen Sie einen RFC mit dem Änderungstyp Management | AMS Resource Scheduler | Solution | Update ein und geben Sie eine kommagetrennte ARNs KMS-Liste an. CMKs

Auto Scaling Scaling-Gruppen planen

- AMS Resource Scheduler startet oder stoppt die Auto Scaling von Auto Scaling-Gruppen, nicht von einzelnen Instances in der Gruppe. Das heißt, der Scheduler stellt die Größe der Auto Scaling Scaling-Gruppe wieder her (Start) oder setzt die Größe auf 0 (Stopp).
- Kennzeichnen Sie die AutoScaling Gruppe mit dem angegebenen Tag und nicht die Instances innerhalb der Gruppe.
- Während des Stopps speichert AMS Resource Scheduler die minimalen, gewünschten und maximalen Kapazitätswerte der Auto Scaling Scaling-Gruppe und setzt die Minimal- und Gewünschte Kapazität auf 0. Während des Starts stellt der Scheduler die Auto Scaling Scaling-Gruppengröße wieder her, wie sie beim Stopp war. Daher müssen Auto Scaling Scaling-Gruppeninstanzen eine geeignete Kapazitätskonfiguration verwenden, damit die Beendigung und der Neustart der Instances keine Auswirkungen auf Anwendungen haben, die in der Auto Scaling Scaling-Gruppe ausgeführt werden.
- Wenn die Auto Scaling Scaling-Gruppe während einer Laufzeit geändert wird (die minimale oder maximale Kapazität), speichert der Scheduler die neue Auto Scaling Scaling-Gruppengröße und verwendet sie, wenn die Gruppe am Ende eines Stopp-Zeitplans wiederhergestellt wird.

Planung von Amazon RDS-Instances

- Der Scheduler kann vor dem Stoppen der RDS-Instances einen Snapshot erstellen (gilt nicht für den Aurora-DB-Cluster). Diese Funktion ist standardmäßig aktiviert, wobei der AWS CloudFormation Vorlagenparameter Create RDS Instance Snapshot auf true gesetzt ist. Der Snapshot wird aufbewahrt, bis die Amazon RDS-Instance das nächste Mal gestoppt und ein neuer Snapshot erstellt wird.

  Scheduler kann start/stop Amazon RDS-Instances verwenden, die Teil eines Clusters oder einer Amazon RDS Aurora-Datenbank sind oder sich in einer Multi-AZone-Konfiguration (Multi-AZ) befinden. Überprüfen Sie jedoch die Amazon RDS-Beschränkung, wenn der Scheduler die Amazon RDS-Instance, insbesondere Multi-AZ-Instances, nicht stoppen kann. Verwenden Sie den Vorlagenparameter Schedule Aurora Clusters, um Aurora Cluster für den Start oder Stopp zu planen (der Standardwert ist true). Der Aurora-Cluster (nicht die einzelnen Instances innerhalb des Clusters) muss mit dem Tag-Schlüssel, der bei der Erstkonfiguration definiert wurde, und dem Zeitplannamen als Tag-Wert für die Planung dieses Clusters gekennzeichnet werden.

  Jede Amazon RDS-Instance hat ein wöchentliches Wartungsfenster, in dem alle Systemänderungen vorgenommen werden. Während des Wartungsfensters startet Amazon RDS

automatisch Instances, die länger als sieben Tage angehalten wurden, um Wartungsarbeiten durchzuführen. Beachten Sie, dass Amazon RDS die Instance nicht stoppt, sobald das Wartungsereignis abgeschlossen ist.

Der Scheduler ermöglicht die Angabe, ob das bevorzugte Wartungsfenster einer Amazon RDS-Instance als Laufzeit zu ihrem Zeitplan hinzugefügt werden soll. Die Lösung startet die Instance zu Beginn des Wartungsfensters und stoppt die Instance am Ende des Wartungsfensters, wenn keine andere Laufzeit vorgibt, dass die Instance ausgeführt werden soll, und wenn das Wartungsereignis abgeschlossen ist.

Wenn das Wartungsereignis bis zum Ende des Wartungsfensters nicht abgeschlossen ist, wird die Instanz bis zum Planungsintervall nach Abschluss des Wartungsereignisses ausgeführt.

> **Note**
>
> Der Scheduler überprüft nicht, ob eine Ressource gestartet oder gestoppt wurde. Er ruft die API auf und fährt fort. Wenn der API-Aufruf fehlschlägt, wird der Fehler zur Untersuchung protokolliert.

Überlegungen zur Anwendungssicherheit

Zur Anwendungssicherheit gehört die Überlegung, welche Berechtigungen die Anwendung für die Ausführung benötigt, welche Firewallregeln und welche IAM-Rollen für den Zugriff auf die Anwendung aktiviert werden sollten.

Weitere Informationen zur allgemeinen AWS Sicherheit finden Sie unter Bewährte Methoden für Sicherheit, Identität und Compliance.

Zugriff für die Konfigurationsverwaltung

AWS Managed Services (AMS) möchte Ihnen eine Infrastruktur bieten, die keine Probleme bereitet, sodass Sie sich keine Gedanken über Sicherheitsprobleme, Patching-Probleme, Backup-Probleme usw. machen müssen. Zu diesem Zweck empfiehlt AMS minimale IAM-Rollen, die nur einer bestimmten Gruppe oder einem Masterserver, falls Sie ein Tool zur Anwendungsbereitstellung verwenden, Zugriff auf die Instances gewähren, auf denen Ihre Anwendung ausgeführt wird.

Firewall-Regeln für den Anwendungszugriff

Genau wie das Betriebssystem (OS) sollte der gesamte Anwendungszugriff mithilfe von Active Directory-Gruppen (AD) gesteuert werden. Wenn Sie Amazon Relational Database Service (Amazon RDS) als Beispiel verwenden, müssen Sie die Spiegelung (Replikation) unterbrechen, um einen neuen Benutzer hinzuzufügen. Der beste Ansatz besteht darin, eine Gruppe in AD zu erstellen und sie bei der Erstellung der Datenbank hinzuzufügen. Wenn Sie die Gruppen in Ihrem AMS AD haben, können Sie sie CTs für den Anwendungszugriff erstellen. Informationen zur offiziellen Gruppierungsstrategie für AD finden Sie unter Using Group Nesting Strategy — AD Best Practices for Group Strategy.

Weitere Informationen zu Domänenstrukturen und parent/child Domänen finden Sie unter Funktionsweise von Domänen und Gesamtstrukturen.

Die folgenden Regeln veranschaulichen eine Lösung, die sich für eine Gesamtstrukturvertrauensstellung mit mehreren Domänen eignet, bei der sich Benutzer in untergeordneten Domänen befinden.

Windows-Instanzen

Dies sind die Regeln, die Sie für Ihre übergeordneten und untergeordneten Windows-Domänencontroller konfigurieren müssen.

Übergeordneter Domänencontroller, Windows

VON: Übergeordnete Domänencontroller BIS: Windows-Stack- und Shared Services-Subnetze

Quell-Port	Ziel-Port	Protokoll
88	49152–65535	TCP
389	49152–65535	UDP

VON: Stack-Subnetze, einschließlich Shared Services, ZU: Stammdomänencontrollern der Windows-Gesamtstruktur

Quell-Port	Ziel-Port	Protokoll
49152–65535	88	TCP
49152–65535	389	UDP

Untergeordneter Domänencontroller, Windows

VON: Untergeordnete Domänencontroller BIS: Windows AWS-Domänencontroller

Quell-Port	Ziel-Port	Protokoll
49152–65535	53	TCP
49152–65535	88	TCP
49152–65535	389	UDP

VON: Untergeordnete Domänencontroller BIS: Windows-Stack- und Shared Services-Subnetze

Quell-Port	Ziel-Port	Protokoll
88	49152–65535	TCP
135	49152–65535	TCP
389	49152–65535	TCP
389	49152–65535	UDP
445	49152–65535	TCP
49152–65535	49152–65535	TCP

VON: Stack-Subnetze, einschließlich Shared Services, ZU: Untergeordnete Windows-Domänencontroller

Quell-Port	Ziel-Port	Protokoll
49152–65535	88	TCP
49152–65535	135	TCP
49152–65535	389	TCP
49152–65535	389	UDP
49152–65535	445	TCP
49152–65535	49152–65535	TCP

Linux-Instanzen

Dies sind die Regeln, die Sie für Ihre übergeordneten und untergeordneten Linux-Domänencontroller konfigurieren müssen.

Alle Tests wurden mit Amazon Linux durchgeführt. Während der dynamische Portbereich für Windows zwischen 49152 und 65535 liegt, verwenden viele Linux-Kernel den Portbereich 32768 bis 61000. Führen Sie den folgenden Befehl aus, um den IP-Portbereich anzuzeigen.

```
cat /proc/sys/net/ipv4/ip_local_port_range
```

Übergeordneter Domänencontroller, Linux

VON: Übergeordnete Domänencontroller BIS: Linux-Stack und Shared Services-Subnetze

Quell-Port	Ziel-Port	Protokoll
389	32768 - 61000	UDP
88	32768 - 61000	TCP

VON: Stack-Subnetze, einschließlich Shared Services, ZU: Stammdomänencontrollern der Linux-Gesamtstruktur

Quell-Port	Ziel-Port	Protokoll
32768 — 61000	88	TCP
32768 - 61000	389	UDP

Untergeordneter Domänencontroller, Linux

VON: Untergeordnete Domänencontroller BIS: Linux-AWS-Domänencontroller

Quell-Port	Ziel-Port	Protokoll
49152–65535	53	TCP
49152–65535	88	TCP
389	49152–65535	UDP
49152–65535	389	UDP

VON: Untergeordnete Domänencontroller BIS: Linux-Stack- und Shared-Services-Subnetze

Quell-Port	Ziel-Port	Protokoll
88	32768 - 61000	TCP
389	32768 - 61000	UDP

VON: Stack-Subnetze, einschließlich Shared Services, BIS: Child-Domaincontroller unter Linux

Quell-Port	Ziel-Port	Protokoll
32768 — 61000	88	TCP
32768 - 61000	389	UDP

Verwaltung des Ausgangsverkehrs mit AMS

Standardmäßig hat die Route mit einem Ziel-CIDR von 0.0.0.0/0 für private AMS-Subnetze und Subnetze für Kundenanwendungen ein NAT-Gateway (Network Address Translation) als Ziel. AMS-Dienste TrendMicro und Patching sind Komponenten, die ausgehenden Zugriff auf das Internet haben müssen, damit AMS seinen Dienst bereitstellen kann und Betriebssysteme Updates abrufen können. TrendMicro

AMS unterstützt die Umleitung des ausgehenden Datenverkehrs über ein vom Kunden verwaltetes Ausgangsgerät in das Internet, sofern:

- Es fungiert als impliziter (z. B. transparenter) Proxy.

  and

- Es ermöglicht AMS-HTTP- und HTTPS-Abhängigkeiten (in diesem Abschnitt aufgeführt), um das kontinuierliche Patchen und die Wartung der von AMS verwalteten Infrastruktur zu ermöglichen.

Einige Beispiele sind:

- Das Transit Gateway (TGW) hat eine Standardroute, die über die AWS Direct Connect Connect-Verbindung im Multi-Account Landing Zone Networking-Konto auf die vom Kunden verwaltete, lokale Firewall verweist.

- Der TGW hat eine Standardroute, die auf einen AWS-Endpunkt in der ausgehenden Multi-Account-Landingzone-VPC verweist, der AWS nutzt PrivateLink, und auf einen vom Kunden verwalteten Proxy in einem anderen AWS-Konto verweist.
- Der TGW hat eine Standardroute, die auf eine vom Kunden verwaltete Firewall in einem anderen AWS-Konto verweist, mit einer site-to-site VPN-Verbindung als Anhang zur Multi-Account-Landing Zone TGW.

AMS hat die entsprechenden HTTP- und HTTPS-Abhängigkeiten von AMS identifiziert und entwickelt und verfeinert diese Abhängigkeiten kontinuierlich. Weitere Informationen finden Sie unter egressMgmt.zip. Zusammen mit der JSON-Datei enthält die ZIP-Datei eine README-Datei.

> **Note**
>
> - Diese Informationen sind nicht vollständig — einige erforderliche externe Websites sind hier nicht aufgeführt.
> - Verwenden Sie diese Liste nicht im Rahmen einer Ablehnungsliste oder einer Blockierungsstrategie.
> - Diese Liste dient als Ausgangspunkt für einen Regelsatz zur Filterung ausgehender Zugriffe, wobei davon ausgegangen wird, dass mithilfe von Berichtstools genau ermittelt werden kann, wo der tatsächliche Datenverkehr von der Liste abweicht.

Wenn Sie Informationen zur Filterung des ausgehenden Datenverkehrs benötigen, senden Sie eine E-Mail an Ihren CSDM: ams-csdm@amazon.com.

Sicherheitsgruppen

In AWS VPCs agieren AWS-Sicherheitsgruppen als virtuelle Firewalls und kontrollieren den Datenverkehr für einen oder mehrere Stacks (eine Instanz oder eine Reihe von Instances). Wenn ein Stack gestartet wird, wird er einer oder mehreren Sicherheitsgruppen zugeordnet, die festlegen, welcher Datenverkehr ihn erreichen darf:

- Für Stacks in Ihren öffentlichen Subnetzen akzeptieren die Standardsicherheitsgruppen Traffic von HTTP (80) und HTTPS (443) von allen Standorten (dem Internet). Die Stacks akzeptieren auch internen SSH- und RDP-Verkehr aus Ihrem Unternehmensnetzwerk und AWS-Bastionen. Diese

Stacks können dann über jeden beliebigen Port ins Internet gelangen. Sie können auch in Ihre privaten Subnetze und andere Stacks in Ihrem öffentlichen Subnetz gelangen.

- Stacks in Ihren privaten Subnetzen können zu jedem anderen Stack in Ihrem privaten Subnetz austreten, und Instances innerhalb eines Stacks können vollständig über jedes Protokoll miteinander kommunizieren.

> **Important**
>
> Die Standardsicherheitsgruppe für Stacks in privaten Subnetzen ermöglicht es allen Stacks in Ihrem privaten Subnetz, mit anderen Stacks in diesem privaten Subnetz zu kommunizieren. Wenn Sie die Kommunikation zwischen Stacks innerhalb eines privaten Subnetzes einschränken möchten, müssen Sie neue Sicherheitsgruppen erstellen, die die Einschränkung beschreiben. Wenn Sie beispielsweise die Kommunikation mit einem Datenbankserver einschränken möchten, sodass die Stacks in diesem privaten Subnetz nur von einem bestimmten Anwendungsserver über einen bestimmten Port kommunizieren können, fordern Sie eine spezielle Sicherheitsgruppe an. Wie das geht, wird in diesem Abschnitt beschrieben.

Standardsicherheitsgruppen

MALZ

In der folgenden Tabelle werden die Standardeinstellungen für eingehende Sicherheitsgruppen (SG) für Ihre Stacks beschrieben. Die SG trägt den Namen „SentinelDefaultSecurityGroupPrivateOnly-VPC-ID“, wobei *ID* es sich um eine VPC-ID in Ihrem AMS-Landingzone-Konto mit mehreren Konten handelt. Der gesamte Datenverkehr darf über diese Sicherheitsgruppe nach "mc-initial-garden-SentinelDefaultSecurityGroupPrivateOnly" ausgehen (der gesamte lokale Verkehr innerhalb von Stack-Subnetzen ist zulässig).

Jeglicher Datenverkehr darf von einer zweiten Sicherheitsgruppe "" nach 0.0.0.0/0 ausgehen. SentinelDefaultSecurityGroupPrivateOnly

> **Tip**
>
> Wenn Sie eine Sicherheitsgruppe für einen AMS-Änderungstyp wie EC2 Create oder OpenSearch Create Domain auswählen, würden Sie eine der hier beschriebenen

Standardsicherheitsgruppen oder eine von Ihnen erstellte Sicherheitsgruppe verwenden. Sie finden die Liste der Sicherheitsgruppen pro VPC entweder in der EC2 AWS-Konsole oder in der VPC-Konsole.

Es gibt zusätzliche Standardsicherheitsgruppen, die für interne AMS-Zwecke verwendet werden.

AMS-Standardsicherheitsgruppen (eingehender Verkehr)

Typ	Protocol (Protokoll)	Port-Bereich	Quelle
Gesamter Datenverkehr	Alle	Alle	SentinelDefaultSecurityGroupPrivateOnly (schränkt ausgehenden Datenverkehr auf Mitglieder derselben Sicherheitsgruppe ein)
Gesamter Datenverkehr	Alle	Alle	SentinelDefaultSecurityGroupPrivateOnlyEgressAll (schränkt ausgehenden Verkehr nicht ein)
HTTP, HTTPS, SSH, RDP	TCP	80/443 (Quelle 0.0.0.0/0) SSH- und RDP-Zugriff ist von Bastionen aus erlaubt	SentinelDefaultSecurityGroupPublic (schränkt den ausgehenden Verkehr nicht ein)
MALZ-Bastionen:			
SSH	TCP	22	SharedServices VPC CIDR und DMZ VPC CIDR sowie vom Kunden bereitgestelltes On-Premise-System CIDRs
SSH	TCP	22	
RDP	TCP	3389	
RDP	TCP	3389	
SALZ-Bastionen:			

Typ	Protocol (Protokoll)	Port-Bereich	Quelle
SSH	TCP	22	mc-initial-garden- SG LinuxBastion
SSH	TCP	22	mc-initial-garden- LinuxBastion DMZSG
RDP	TCP	3389	mc-initial-garden- SG WindowsBastion
RDP	TCP	3389	mc-initial-garden- WindowsBastion DMZSG

SALZ

In der folgenden Tabelle werden die Standardeinstellungen für eingehende Sicherheitsgruppen (Inbound Security Group, SG) für Ihre Stacks beschrieben. Die SG trägt den Namen "mc-initial-garden- SentinelDefaultSecurityGroupPrivateOnly -*ID*", wobei *ID* es sich um einen eindeutigen Bezeichner handelt. Der gesamte Datenverkehr darf über diese Sicherheitsgruppe nach "mc-initial-garden-SentinelDefaultSecurityGroupPrivateOnly" ausgehen (der gesamte lokale Verkehr innerhalb von Stack-Subnetzen ist zulässig).

Der gesamte Datenverkehr darf von einer zweiten Sicherheitsgruppe "- -“ nach 0.0.0.0/0 ausgesendet werden. mc-initial-garden SentinelDefaultSecurityGroupPrivateOnlyEgressAll *ID*

> **Tip**
>
> Wenn Sie eine Sicherheitsgruppe für einen AMS-Änderungstyp wie EC2 Create oder OpenSearch Create Domain auswählen, würden Sie eine der hier beschriebenen Standardsicherheitsgruppen oder eine von Ihnen erstellte Sicherheitsgruppe verwenden. Sie finden die Liste der Sicherheitsgruppen pro VPC entweder in der EC2 AWS-Konsole oder in der VPC-Konsole.

Es gibt zusätzliche Standardsicherheitsgruppen, die für interne AMS-Zwecke verwendet werden.

AMS-Standardsicherheitsgruppen (eingehender Verkehr)

Typ	Protocol (Protokoll)	Port-Bereich	Quelle
Gesamter Datenverkehr	Alle	Alle	SentinelDefaultSecurityGroupPrivateOnly (schränkt ausgehenden Datenverkehr auf Mitglieder derselben Sicherheitsgruppe ein)
Gesamter Datenverkehr	Alle	Alle	SentinelDefaultSecurityGroupPrivateOnlyEgressAll (schränkt ausgehenden Verkehr nicht ein)
HTTP, HTTPS, SSH, RDP	TCP	80/443 (Quelle 0.0.0.0/0) SSH- und RDP-Zugriff ist von Bastionen aus erlaubt	SentinelDefaultSecurityGroupPublic (schränkt den ausgehenden Verkehr nicht ein)
MALZ-Bastionen:			
SSH	TCP	22	SharedServices VPC CIDR und DMZ VPC CIDR sowie vom Kunden bereitgestelltes On-Premise-System CIDRs
SSH	TCP	22	
RDP	TCP	3389	
RDP	TCP	3389	
SALZ-Bastionen:			
SSH	TCP	22	mc-initial-garden- SG LinuxBastion
SSH	TCP	22	mc-initial-garden- LinuxBastion DMZSG
RDP	TCP	3389	mc-initial-garden- SG WindowsBastion
RDP	TCP	3389	mc-initial-garden- WindowsBastion DMZSG

Sicherheitsgruppen erstellen, ändern oder löschen

Sie können benutzerdefinierte Sicherheitsgruppen anfordern. In Fällen, in denen die Standardsicherheitsgruppen nicht den Anforderungen Ihrer Anwendungen oder Ihrer Organisation entsprechen, können Sie neue Sicherheitsgruppen ändern oder erstellen. Eine solche Anfrage würde als genehmigungspflichtig erachtet und vom AMS-Betriebsteam geprüft.

Um eine Sicherheitsgruppe außerhalb von Stacks zu erstellen und einen RFC mit dem `Deployment | Advanced stack components | Security group | Create (review required)` Änderungstyp (VPCsct-1oxx2g2d7hc90) einzureichen.

Verwenden Sie für Änderungen an Active Directory (AD) -Sicherheitsgruppen die folgenden Änderungstypen:

- Um einen Benutzer hinzuzufügen: Senden Sie einen RFC mit Management | Directory Service | Benutzer und Gruppen | Benutzer zur Gruppe hinzufügen [ct-24pi85mjtza8k]
- Um einen Benutzer zu entfernen: Senden Sie einen RFC mit Management | Directory Service | Benutzer und Gruppen | Benutzer aus Gruppe entfernen [ct-2019s9y3nfml4]

> **Note**
>
> Wenn Sie „Überprüfung erforderlich“ verwenden CTs, empfiehlt AMS, die ASAP-Scheduling-Option zu verwenden (wählen Sie ASAP in der Konsole, lassen Sie Start- und Endzeit leer in der API/CLI), da diese einen AMS-Operator CTs erfordern, der den RFC überprüft und möglicherweise mit Ihnen kommuniziert, bevor er genehmigt und ausgeführt werden kann. Wenn Sie diese einplanen, achten Sie darauf RFCs, dass Sie mindestens 24 Stunden einplanen. Wenn die Genehmigung nicht vor der geplanten Startzeit erfolgt, wird der RFC automatisch abgelehnt.

Suchen Sie nach Sicherheitsgruppen

Verwenden Sie die EC2 Konsole, um die Sicherheitsgruppen zu finden, die einem Stack oder einer Instance zugeordnet sind. Nachdem Sie den Stack oder die Instance gefunden haben, können Sie alle damit verbundenen Sicherheitsgruppen sehen.

Möglichkeiten, Sicherheitsgruppen in der Befehlszeile zu finden und die Ausgabe zu filtern, finden Sie unter `describe-security-groups`.

Anhang: Fragebogen zum Onboarding von Bewerbungen

Verwenden Sie diesen Fragebogen, um Ihre Implementierungselemente und Ihre Struktur zu beschreiben, sodass AMS ermitteln kann, welche Infrastrukturkomponenten benötigt werden. Die Onboarding-Anforderungen für Line-of-Business (LoB-) Anwendungen unterscheiden sich erheblich von denen für Produktanwendungen. Daher ist dieser Fragebogen darauf ausgelegt, beide zu berücksichtigen.

Themen

Zusammenfassung der Bereitstellung

Eine Beschreibung der Bereitstellung. Beispiel:

- Dieses Konto ist für eine Line-of-Business (LoB-) Anwendungsbereitstellung (im Gegensatz zur Bereitstellung einer Produktanwendung) vorgesehen.
- Die Bereitstellung beinhaltet einen automatisch skalierten ARP (authentifizierter Reverse-Proxy) innerhalb des Subnetzes des Kontos. public/DMZ
- Web- und Anwendungsserver werden im privaten Subnetz des Kontos bereitgestellt.
- Eine Amazon RDS-Instance (Amazon Relational Database Service) wird ebenfalls im privaten Subnetz des Kontos bereitgestellt.
- Die Server (ARP, Web, Anwendung, Datenbank, Load Balancer usw.) sind in verschiedene Sicherheitsgruppen unterteilt.
- Für das Konto ist ein HA-Design (Hochverfügbarkeit) erforderlich, das auf mehrere Availability Zones (AZs) verteilt ist, d. h. Multi-AZ.

Komponenten für die Infrastrukturbereitstellung

Was sind all die verschiedenen Komponenten, die konfiguriert werden müssen, um Ihre Anwendung zu unterstützen?

- Region: Welche AWS-Region Regionen werden benötigt?
- Hochverfügbarkeit (HA): Welche Availability Zones werden verwendet?
- Virtual Private Cloud (VPC): Was ist der CIDR-Block für die VPC?
- Welche Serverinstanzen werden benötigt?
  - Authentifizierter Reverse Proxy (ARP): Betriebssystem, AMI, Instance-Typ, Subnetz-ID, Sicherheitsgruppe, Eingangsport?
  - Application Deployment Tool-Server: Betriebssystem, AMI, Instanztyp, Subnetz-ID, Sicherheitsgruppe, Eingangsport (Chef, Puppet) oder Ausgangsport (Ansible, Saltstack)?
  - Amazon RDS mit MySQL: DB-Version, Nutzungstyp, Instance-Klasse, Subnetz-ID, Sicherheitsgruppe, DB-Instance-ID, Speichergröße, Multi-AZ, Authentifizierungstyp, Verschlüsselung?
  - Speicher: Ist Ihre App zustandslos? Benötigen Sie S3-Buckets? Benötigen Sie persistenten Speicher? Benötigen Sie Verschlüsselung von Daten im Ruhezustand auf Ihren EBS-Volumes? Benötigen Sie DB-Verschlüsselung?
  - Externe Serverendpunkte (zur Managed Services VPC): SMTP? LDAP?
  - Netzwerkanforderungen: Netzwerkfilterung (basierend auf Sicherheitsgruppen?)? Inspektion des Webverkehrs (eingehend? ausgehend?)?
- Tagging: Welche Tags sollten verwendet werden, um Ressourcen in logischen Sammlungen zu gruppieren? Zum Beispiel alle Ressourcen für einen Anwendungsstapel. Wählen Sie Tags für Ihren Anwendungsfall aus, z. B. `backup=true` um Backups zu aktivieren. Darüber hinaus müssen Sie das Tag `name=value` verwenden, damit für alle von Ihnen erstellten EC2 Instanzen ein Name in der Konsole angezeigt wird.
- Sicherheitsgruppen:
  - Welche Sicherheitsgruppen werden benötigt?
  - Regeln für den Zugriff auf Sicherheitsgruppen?
  - Regeln für ausgehenden Datenverkehr in Sicherheitsgruppen?

Plattform für das Hosten von Anwendungen

Beachten Sie für Ihre Anwendungshosting-Plattform die folgenden möglichen Anforderungen:

- Verschlüsselte Datenbanken?
- Von wem verwaltete Verschlüsselungsschlüssel?
- Sind alle Daten während der Übertragung und im Speicher verschlüsselt?
- Zugriff aller Benutzer auf das System über HTTPS?
- Alle system-to-system Interaktionen wurden von Ihrem Sicherheitsteam genehmigt?

Modell zur Anwendungsbereitstellung

Überlegungen zur Planung Ihrer Anwendungsbereitstellungen. Siehe Was ist mein Betriebsmodell?

- Automatisiert oder manuell? Keine Automatisierung der Bereitstellung bedeutet keine automatische Skalierung. Wenn Sie Zugriff anfordern und sich anmelden und Ihre Anwendung manuell aktualisieren und Ihr Update fehlschlägt. AMS erwartet, dass Sie Ihr Update rückgängig machen oder uns über eine Serviceanfrage benachrichtigen, damit wir Ihnen weiterhelfen können.
- Was ist das Framework, falls es automatisiert ist? Skripte? Agentenbasiert (puppet/chef)? Agentless (SALT/Ansible)? CodeDeploy? Für agentenbasierte und agentenlose Bereitstellungstools muss eine separate Instanz erstellt und als Master-Server für die Tools bereitgestellt werden. AMS erwartet von Ihnen, dass Sie sich aller Elemente bewusst sind, die für eine erfolgreiche Implementierung von Tools für die Anwendungsbereitstellung erforderlich sind. Wir helfen Ihnen jedoch gerne bei diesbezüglichen Infrastrukturfragen weiter.
- Müssen Ihre Line-of-Business Anwendungen (die Anwendungen, mit denen Sie Ihre Anwendungen erstellen und verwalten) gepatcht werden?

Abhängigkeiten von Anwendungen

Benötigen Sie Instances für Line-of-Business (LoB-) Anwendungen? Für Produktanwendungen?

Was benötigen Ihre Produktanwendungen, um ordnungsgemäß zu funktionieren?

- Abhängigkeiten auf Netzwerkebene: Zum Beispiel AWS Direct Connect
- Paketabhängigkeiten: Zum Beispiel `pip`

- Anwendungen, von denen diese Anwendung abhängt: Zum Beispiel MySql
- Firewall-Abhängigkeiten?

Was benötigen Ihre LoB-Anwendungen, um ordnungsgemäß zu funktionieren?

- Abhängigkeiten auf Netzwerkebene: Zum Beispiel AWS Direct Connect
- Paketabhängigkeiten: Zum Beispiel Firefox Saucy
- Anwendungen, von denen diese Anwendung abhängt: Zum Beispiel MySql
- Firewall-Abhängigkeiten?

SSL-Zertifikate für Produktanwendungen

Welche SSL-Zertifikate benötigen Ihre Server, damit Ihre Anwendungen (LoB und Produkt) alles erreichen können, was sie für die Ausführung und den Zugriff benötigen?

- Auto Scaling Scaling-Gruppe?
- Datenbank (Amazon RDS)?
- Load Balancer?
- Server mit Bereitstellungstools?
- Firewall für Webanwendungen (AWS WAF)?
- Andere Instanzen?

Beispielsweise benötigen Sie für jede der oben aufgeführten Instanzen möglicherweise die folgenden Zertifikate:

WAF (Zertifikat 1) -> ELB-Ext (Zertifikat 2) -> ARP (Zertifikat 3) -> ELB-Int (Zertifikat 4) -> Website (Zertifikat 5) -> ELB-Int (Zertifikat 6) -> Webservice (Zertifikat 7).

Dokumentverlauf

In der folgenden Tabelle wird die Dokumentation für diese Version von AMS beschrieben.

- API-Version: 2019-05-21
- Letzte Aktualisierung der Dokumentation: 16. Februar 2023

Änderungen	Beschreibung	Link
Der Link zum Inhaltsverzeichnis wurde entfernt	Der Link zum AWS Inhaltsverzeichnis-Glossar wurde entfernt.	08. August 2025
Aktualisierter Inhalt: Migration von Workloads: Windows-Validierung vor der Datenaufnahme	Der Abschnitt wurde aktualisiert und enthält nun detaillierte Schritte zur Verwendung des WIGs Pre-Validater-Skripts zur Überprüfung, ob Ihre Windows-Instance für die Aufnahme in Ihr AMS-Konto bereit ist;.	Migration von Workloads: Windows-Validierung vor der Erfassung
Aktualisierter Inhalt, DMS-Konfiguration	ein wichtiger Hinweis zur erforderlichen Rolle, dms-vpc-role.	1: AWS DMS Replikationssubnetzgruppe: Erstellen
Aktualisierter Inhalt, von CFN Ingest unterstützte Ressourcen	Hinzugefügt. OpenSearch	Unterstützte Ressourcen
Aktualisierter Inhalt, Migration von Workloads	Aktualisierte Anweisungen für die Validierung vor der Aufnahme.	Migration von Workloads: Windows-Validierung vor der Erfassung

Änderungen	Beschreibung	Link
Aktualisierter Inhalt, CFN Ingest.	Eingeschränkte „unterstützte Ressourcen“ wurden aus CFN-Ingest-Inhalten entfernt.	CloudFormation Ingest Stack: Unterstützte Ressourcen
Die unterstützten Windows-Versionen wurden aktualisiert	Unterstützung für Windows Server 2022 hinzugefügt.	AMS Amazon-Maschinenbilder (AMIs), Migration von Workloads: Voraussetzungen für Linux und Windows und Migration von Workloads: Windows-Validierung vor der Erfassung
Aktualisierter Inhalt, Resource Scheduler.	Die Anweisungen zur Verwendung des speziellen Bereitstellungs-CT, ct-0ywnhc8e5k9z5, wurden aktualisiert und gelten sowohl für SALZ als auch für MALZ.	AMS Resource Scheduler Schnellstart
Inhalt aktualisiert, Workload Ingest.	Die unterstützten SUSE Linux-Versionen wurden aktualisiert.	Migration von Workloads: Voraussetzungen für Linux und Windows

Änderungen	Beschreibung	Link
Aktualisierter Inhalt, Database Migration Service.	Zu den Voraussetzungen hinzugefügt und mehrere Änderungen im Hinblick auf Nützlichkeit und Benutzerfreundlichkeit vorgenommen.	AWS Database Migration Service (AWS DMS)
Inhalt aktualisiert, Workload Ingest.	Das Linux Pre-WIGS Validation Zip wurde aktualisiert.	Migration von Workloads: Voraussetzungen für Linux und Windows
Inhalt aktualisiert.	Das Pre-WIGS-Validierungs-ZIP-Dokument für Linux wurde aktualisiert. Außerdem wurde Windows Server 2008 R2 als unterstütztes Betriebssystem hinzugefügt.	Migration von Workloads: Voraussetzungen für Linux und Windows
Neuer Inhalt	Schnellstarts und Tutorials wurden aus dem älteren AMS Advanced Change Management Guide hierher verschoben.	Schnelle Starts, Tutorials.
Aktualisierter Inhalt	Bereitstellung | Erweiterte Stack-Komponenten | Database Migration Service (DMS) | Replikationsaufgabe starten (ct-1yq7hhqse71yg) Es wurde aktualisiert und gibt nun an, dass es sich bei den Parametern DocumentNameund Region um erforderliche Parameter handelt. Zuvor wurden sie fälschlicherweise als optional aufgeführt.	Database Migration Service (DMS) | Replikationsaufgabe starten

Änderungen	Beschreibung	Link
Aktualisierter Inhalt	CloudFormation Investieren Es wurde aktualisiert und weist nun auf zwei neue unterstützte Ressourcen hin, AWS::Route53Resolver::ResolverRuleAssociation und AWS::Route53Resolver::ResolverRule.	Unterstützte Ressourcen
Aktualisierter Inhalt	Migration von Workloads: Windows-Validierung vor der Datenaufnahme	Die Sysprep-Informationen wurden mit weiteren Einzelheiten aktualisiert. Migration von Workloads: Windows-Validierung vor der Erfassung
Aktualisierter Inhalt	Verwaltung | Benutzerdefinierter Stapel | Aus CloudFormation Vorlage stapeln | Changeset und aktualisieren (ct-1404e21baa2ox) Die CT-Walkthrough-Beschreibung für den Parameter wurde mit zusätzlichen Informationen aktualisiert. ChangeSetName	Aus CloudFormation Vorlage stapeln | Changeset genehmigen und aktualisieren
	Ubuntu 18.04 und Oracle Linux 8.3 verfügbar	Migration von Workloads: Voraussetzungen für Linux und Windows

Änderungen	Beschreibung	Link
Neuer Inhalt:	IAM-Bereitstellungen über CFN Ingest und Stack Update. CTs	10. Februar 2022
Replikationsaufgaben des Database Migration Service (DMS)	Die Typen wurden aktualisiert, sodass reguläre Ausdrücke Aufgaben zulassen ARNs , die Bindestriche enthalten. Starten Sie die AWS DMS Replikationsaufgabeund Database Migration Service (DMS) | Replikationsaufgabe beenden.	13. Januar 2022
Validierung von Linux WIGS vor der Datenaufnahme	Die Zip-Datei wurde aktualisiert. Migration von Workloads: Linux-Validierung vor der Datenaufnahme.	13. Januar 2022
Feste Links	Der Einrichten Abschnitt Datenbank (DB) Import nach AMS SQL RDS -> hatte einige fehlerhafte Links.	13. Januar 2022

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.