Options avancées de déploiement d'applications AMS

Guide du développeur d'applications AMS Advanced

Version September 13, 2024

Guide du développeur d'applications AMS Advanced: Options avancées de déploiement d'applications AMS

Table of Contents

Intégration des applications

Bienvenue dans le plan d'opérations AMS d'AWS Managed Services (AMS). L'objectif de ce document est de décrire les différentes méthodes que vous pouvez utiliser lors de l'intégration de vos applications dans AMS une fois la mise en réseau initiale et la gestion des accès configurées, ainsi que les problèmes à prendre en compte lors du choix de ces méthodes.

Ce document est destiné aux intégrateurs de systèmes et aux développeurs d'applications afin de les aider à déterminer et à élaborer des processus d'application pour les nouveaux clients d'AMS.

Qu'est-ce que l'intégration des applications ?

L'intégration des applications AMS fait référence au déploiement de ressources et d'applications, selon les besoins, dans votre infrastructure AMS. L'architecture des applications et de l'infrastructure sur la plate-forme AMS est très similaire à celle sur la plate-forme native AWS. Le respect des meilleures pratiques en matière de conception d' AWS applications et d'infrastructures, tout en tenant compte des fonctionnalités fournies par AMS, permettra d'obtenir des applications performantes et opérationnelles hébergées dans l'environnement AMS.

> **Note**
>
> - USA Est (Virginie)
> - USA Ouest (Californie du Nord)
> - US West (Oregon)
> - USA Est (Ohio)
> - Canada (Centre)
> - Amérique du Sud (São Paulo)
> - UE (Irlande)
> - UE (Francfort)
> - UE (Londres)
> - UE Ouest (Paris)
> - Asie-Pacifique (Mumbai)
> - Asie-Pacifique (Séoul)
> - Asie-Pacifique (Singapour)

- Asie-Pacifique (Sydney)
- Asia Pacific (Tokyo)

De nouvelles régions sont ajoutées fréquemment. Pour en savoir plus, consultez la section Régions AWS et les zones de disponibilité.

Ce que nous faisons, ce que nous ne faisons pas

AMS vous propose une approche standardisée pour déployer l'infrastructure AWS et fournit la gestion opérationnelle continue nécessaire. Pour une description complète des rôles, des responsabilités et des services pris en charge, consultez la section Description du service.

> **Note**
>
> Pour demander à AMS de fournir un service AWS supplémentaire, déposez une demande de service. Pour plus d'informations, consultez la section Faire des demandes de service.

- Ce que nous faisons :

  Une fois l'intégration terminée, l'environnement AMS est disponible pour recevoir les demandes de modification (RFCs), les incidents et les demandes de service. Votre interaction avec le service AMS tourne autour du cycle de vie d'une pile d'applications. Les nouvelles piles sont commandées à partir d'une liste préconfigurée de modèles, lancées dans des sous-réseaux de cloud privé virtuel (VPC) spécifiques, modifiées au cours de leur durée de vie opérationnelle par le biais de demandes de modification (RFCs) et surveillées pour détecter les événements et les incidents 24 heures sur 24, 7 jours sur 7.

  Les piles d'applications actives sont surveillées et maintenues par AMS, y compris en appliquant des correctifs, et ne nécessitent aucune autre action pendant toute la durée de vie de la pile, sauf si une modification est requise ou si la pile est mise hors service. Les incidents détectés par AMS qui affectent l'état et le fonctionnement de la pile génèrent une notification et peuvent nécessiter ou non une action de votre part pour les résoudre ou les vérifier. Les questions pratiques et autres demandes de renseignements peuvent être posées en soumettant une demande de service.

En outre, AMS vous permet d'activer des services AWS compatibles qui ne sont pas gérés par AMS. Pour plus d'informations sur les services compatibles avec AWS-AMS, consultez la section Mode de provisionnement en libre-service.

- CE QUE NOUS NE FAISONS PAS :

  Bien qu'AMS simplifie le déploiement des applications en fournissant un certain nombre d'options manuelles et automatisées, vous êtes responsable du développement, des tests, de la mise à jour et de la gestion de votre application. AMS fournit une assistance pour résoudre les problèmes d'infrastructure qui ont un impact sur les applications, mais AMS ne peut pas accéder aux configurations de vos applications ni les valider.

Images de machines AMS Amazon (AMIs)

AMS produit des Amazon Machine Images (AMIs) mises à jour chaque mois pour les systèmes d'exploitation pris en charge par AMS. En outre, AMS produit également des images renforcées en matière de sécurité (AMIs) basées sur le benchmark CIS de niveau 1 pour un sous-ensemble des systèmes d'exploitation pris en charge par AMS. Pour savoir quels systèmes d'exploitation disposent d'une image de sécurité améliorée, consultez le guide de l'utilisateur de sécurité AMS, disponible sur la page AWS Artifact -> Reports (trouvez l'option Reports dans le volet de navigation de gauche) filtré pour AWS Managed Services. Pour accéder à AWS Artifact, vous pouvez contacter votre CSDM pour obtenir des instructions ou consulter Getting Started with AWS Artifact.

Pour recevoir des alertes lorsque de nouveaux AMS AMIs sont publiés, vous pouvez vous abonner à une rubrique de notification Amazon Simple Notification Service (Amazon SNS) appelée « AMI AMS ». Pour plus de détails, consultez la section Notifications AMI AMS avec SNS.

La convention de dénomination de l'AMI AMS est :`customer-ams-<operating system>-<release date> - <version>`. (par exemple,`customer-ams-rhel6-2018.11-3`)

Utilisez uniquement les AMS AMIs qui commencent par`customer`.

AMS recommande de toujours utiliser l'AMI la plus récente. Vous pouvez trouver la version la plus récente de l'une AMIs des manières suivantes :

- Je regarde dans la console AMS, sur la AMIspage.

- Affichage du dernier fichier CSV AMI AMS, disponible depuis votre CSDM ou via ce fichier ZIP : contenu de l'AMI AMS 11.2024 et fichier CSV dans un ZIP.

  Pour les anciens fichiers ZIP AMI, consultez l'historique des documents.
- Exécution de cette SKMS commande AMS (SDK AMS SKMS requis) :

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

Contenu AMI AMS ajouté à la base AWS AMIs, par système d'exploitation (OS)

- Linux AMIs :
  - AWS Outils CLI
  - NTP
  - Agent du service de protection des terminaux de Trend Micro
  - Déploiement de code
  - PBIS Enterprise/Beyond Trust AD Bridge

    > **Note**
    >
    > Depuis juin 2022, PBIS Open BeyondTrust n'est plus compatible. Vous ne pourrez pas utiliser PBIS Open sur AMIs un support pris en charge par AMS après juin 2022. Si AMS a pris en charge votre AMI avant juin 2022, vous pouvez continuer à utiliser PBIS Open à votre entière discrétion.
  - Agent SSM
  - Yum Upgrade pour les correctifs critiques
  - Scripts personnalisés/logiciels de gestion AMS (contrôle du démarrage, de la jointure AD, de la surveillance, de la sécurité et de la journalisation)
- Serveur Windows AMIs :
  - Microsoft .NET Framework 4.5
  - PowerShell 5,1
  - AWS Outils pour Windows PowerShell
  - PowerShell Modules AMS contrôlant le démarrage, la jointure AD, la surveillance, la sécurité et la journalisation

- Agent du service de protection des terminaux de Trend Micro
- Agent SSM
- CloudWatch Agent
- EC2Service de configuration (via Windows Server 2012 R2)
- EC2Lancement (Windows Server 2016 et Windows Server 2019)
- EC2LaunchV2 (Windows Server 2022 et versions ultérieures)

Basé sur Linux AMIs :

- Amazon Linux 2023 (dernière version mineure) (AMI minimale non prise en charge)
- Amazon Linux 2 (dernière version mineure)
- Amazon Linux (2ARM64)
- Red Hat Enterprise 8 (dernière version mineure)
- Red Hat Enterprise 9 (dernière version mineure)
- Serveur SUSE Linux Enterprise 15 SP6
- Ubuntu Linux 20.04
- Ubuntu Linux 22.04
- Ubuntu Linux 24.04
- Amazon Linux : pour une présentation du produit, des informations sur les prix, des informations d'utilisation et des informations de support, consultez Amazon Linux 2.

  Pour plus d'informations, consultez Amazon Linux 2 FAQs.
- SUSE Linux Enterprise Server pour applications SAP 15 SP6 :
  - Exécutez les étapes suivantes une fois par compte :
    1. Accédez à AWS Marketplace.
    2. Recherchez le produit SAP SUSE 15.
    3. Choisissez Continuer pour vous abonner.
    4. Choisissez Accepter les conditions.
  - Procédez comme suit chaque fois que vous devez lancer une nouvelle SP6 instance de SUSE Linux Enterprise Server pour SAP Applications 15 :
    1. Notez l'ID d'AMI de l'AMI SUSE Linux Enterprise Server for SAP Applications 15 abonnée.

2. Créer un déploiement | Composants de pile avancés | Pile EC2 | Créer un type de modification ct-14027q0sjyt1h RFC. *InstanceAmiId*Remplacez-le par l'ID AWS Marketplace AMI auquel vous vous êtes abonné.

Basé sur Windows AMIs :

Microsoft Windows Server (2016, 2019, 2022 et 2025), basé sur la dernière version de Windows AMIs.

Pour des exemples de création AMIs, consultez la section Créer une AMI.

Débarquement d'AMS AMIs :

AMS ne vous en communique aucune lors AMIs de l'offboarding afin d'éviter toute répercussion sur vos dépendances. Si vous souhaitez supprimer AMS AMIs de votre compte, vous pouvez utiliser l'`cancel-image-launch-permission`API pour masquer des informations spécifiques AMIs. Par exemple, vous pouvez utiliser le script ci-dessous pour masquer tous AMIs les AMS précédemment partagés avec votre compte :

```
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
```

L'AWS CLI v2 doit être installée pour que le script puisse s'exécuter sans erreur. Pour connaître les étapes d'installation de l'interface de ligne de commande AWS, consultez la section Installation ou mise à jour de la dernière version de l'interface de ligne de commande AWS. Pour plus de détails sur la `cancel-image-launch-permission` commande, consultez `cancel-image-launch-permission`.

Sécurité améliorée AMIs

AMS fournit des images de sécurité améliorées (AMIs) basées sur le benchmark CIS de niveau 1 pour un sous-ensemble des systèmes d'exploitation pris en charge par AMS. Pour savoir quels systèmes d'exploitation disposent d'une image de sécurité améliorée, consultez le guide de sécurité client AWS Managed Services (AMS). Pour accéder à ce guide, ouvrez AWS Artifact, sélectionnez Reports dans le volet de navigation de gauche, puis filtrez pour AWS Managed Services. Pour obtenir

des instructions sur la façon d'y accéder AWS Artifact, contactez votre CSDM ou consultez Getting Started with AWS Artifact pour plus d'informations.

Termes clés d'AMS

- AMS Advanced : les services décrits dans la section « Description du service » de la documentation avancée d'AMS. Voir la description du service.
- Comptes AMS Advanced : AWS comptes qui répondent à tout moment à toutes les exigences des exigences d'intégration avancées d'AMS. Pour plus d'informations sur les avantages d'AMS Advanced, pour des études de cas et pour contacter un commercial, consultez AWS Managed Services.
- Comptes AMS Accelerate : AWS comptes qui répondent à tout moment à toutes les exigences des exigences d'intégration d'AMS Accelerate. Consultez la section Mise en route avec AMS Accelerate.
- AWS Managed Services : AMS et/ou AMS Accelerate.
- Comptes AWS Managed Services : les comptes AMS et/ou les comptes AMS Accelerate.
- Recommandation critique : Recommandation émise par le AWS biais d'une demande de service vous informant que votre action est nécessaire pour vous protéger contre les risques potentiels ou les perturbations affectant vos ressources ou le Services AWS. Si vous décidez de ne pas suivre une recommandation critique à la date spécifiée, vous êtes seul responsable de tout préjudice résultant de votre décision.
- Configuration demandée par le client : tout logiciel, service ou autre configuration qui n'est pas identifié dans :
  - Accélérer : configurations prises en charge ou AMS Accelerate ; description du service.
  - AMS Advanced : configurations prises en charge ou AMS Advanced ; description du service.
- Communication en cas d'incident : AMS vous communique un incident ou vous demandez un incident avec AMS via un incident créé dans le Centre de support pour AMS Accelerate et dans la console AMS pour AMS. La console AMS Accelerate fournit un résumé des incidents et des demandes de service sur le tableau de bord et des liens vers le centre de support pour plus de détails.
- Environnement géré : les comptes AMS Advanced et/ou les comptes AMS Accelerate gérés par AMS.

  Pour AMS Advanced, cela inclut les comptes de zone d'atterrissage à comptes multiples (MALZ) et de zone d'atterrissage à compte unique (SALZ).

- Date de début de facturation : le jour ouvrable suivant la AWS réception des informations demandées dans l'e-mail d'accueil d'AWS Managed Services. L'e-mail d'intégration d'AWS Managed Services fait référence à l'e-mail que vous avez envoyé pour collecter les informations nécessaires à l'activation d'AWS Managed Services sur vos comptes. AWS

  Pour les comptes que vous avez inscrits ultérieurement, la date de début de facturation est le jour suivant l'envoi par AWS Managed Services d'une notification d'activation d'AWS Managed Services pour le compte inscrit. Une notification d'activation d'AWS Managed Services se produit lorsque :

  1. Vous accordez l'accès à un AWS compte compatible et vous le transmettez à AWS Managed Services.
  2. AWS Managed Services conçoit et crée le compte AWS Managed Services.
- Résiliation du service : vous pouvez résilier les services AWS Managed Services pour tous les comptes AWS Managed Services, ou pour un compte AWS Managed Services spécifique pour quelque raison que ce soit, en fournissant un préavis d' AWS au moins 30 jours par le biais d'une demande de service. À la date de fin du service, soit :

  1. AWS vous confie le contrôle de tous les comptes AWS Managed Services ou des comptes AWS Managed Services spécifiés, selon le cas, ou
  2. Les parties suppriment les Gestion des identités et des accès AWS rôles donnant AWS accès à tous les comptes AWS Managed Services ou aux comptes AWS Managed Services spécifiés, selon le cas.
- Date de fin du service : La date de fin du service est le dernier jour du mois civil suivant la fin de la période de préavis de résiliation requise de 30 jours. Si la fin de la période de préavis de résiliation requise tombe après le 20e jour du mois civil, la date de fin du service est le dernier jour du mois civil suivant. Voici des exemples de scénarios relatifs aux dates de résiliation.

  - Si l'avis de résiliation est fourni le 12 avril, le préavis de 30 jours prend fin le 12 mai. La date de fin du service est le 31 mai.
  - Si un préavis de résiliation est fourni le 29 avril, le préavis de 30 jours prend fin le 29 mai. La date de fin du service est le 30 juin.
- Fourniture d'AWS Managed Services : AWS met à votre disposition et vous pouvez accéder à AWS Managed Services et les utiliser pour chaque compte AWS Managed Services à compter de la date de début du service.
- Résiliation de comptes AWS Managed Services spécifiques : vous pouvez résilier les services AWS Managed Services pour un compte AWS Managed Services spécifique pour quelque raison que ce soit en fournissant un AWS préavis par le biais d'une demande de service (« Demande de résiliation de compte AMS »).

Termes de gestion des incidents :

- Événement : modification de votre environnement AMS.
- Alerte : chaque fois qu'un événement provenant d'un Service AWS service pris en charge dépasse un seuil et déclenche une alarme, une alerte est créée et une notification est envoyée à votre liste de contacts. En outre, un incident est créé dans votre liste d'incidents.
- Incident : interruption imprévue ou dégradation des performances de votre environnement AMS ou d'AWS Managed Services ayant un impact tel que signalé par AWS Managed Services ou par vous-même.
- Problème : cause sous-jacente partagée d'un ou de plusieurs incidents.
- Résolution d'un incident ou résolution d'un incident :
  - AMS a rétabli tous les services ou ressources AMS indisponibles relatifs à cet incident à un état disponible, ou
  - AMS a déterminé que les piles ou les ressources indisponibles ne peuvent pas être restaurées à un état disponible, ou
  - AMS a lancé une restauration de l'infrastructure que vous avez autorisée.
- Temps de réponse aux incidents : différence de temps entre le moment où vous créez un incident et le moment où AMS fournit une réponse initiale par le biais de la console, du courrier électronique, du centre de service ou du téléphone.
- Temps de résolution de l'incident : différence de temps entre le moment où AMS ou vous-même créez un incident et le moment où l'incident est résolu.
- Priorité des incidents : comment les incidents sont classés par ordre de priorité par AMS, ou par vous, comme étant faible, moyen ou élevé.
  - Faible : problème non critique lié à votre service AMS.
  - Moyen : au sein de votre environnement géré, un service AWS est disponible mais ne fonctionne pas comme prévu (conformément à la description du service applicable).
  - Élevé : soit (1) la console AMS, soit un ou plusieurs AMS de APIs votre environnement géré ne sont pas disponibles ; soit (2) une ou plusieurs piles ou ressources AMS de votre environnement géré ne sont pas disponibles et cette indisponibilité empêche votre application de fonctionner correctement.

  AMS peut reclasser les incidents conformément aux directives ci-dessus.
- Restauration de l'infrastructure : redéploiement des piles existantes, sur la base de modèles de piles touchées, et lancement d'une restauration des données en fonction du dernier point de

restauration connu, sauf indication contraire de votre part, lorsque la résolution de l'incident n'est pas possible.

Termes relatifs à l'infrastructure :

- Environnement de production géré : compte client sur lequel résident les applications de production du client.
- Environnement géré hors production : compte client contenant uniquement des applications hors production, telles que des applications de développement et de test.
- Pile AMS : groupe d'une ou plusieurs AWS ressources gérées par AMS comme une seule unité.
- Infrastructure immuable : modèle de maintenance d'infrastructure typique des groupes Amazon EC2 Auto Scaling (ASGs) dans lequel les composants d'infrastructure mis à jour (dans AWS l'AMI) sont remplacés à chaque déploiement, au lieu d'être mis à jour sur place. L'avantage d'une infrastructure immuable est que tous les composants restent dans un état synchrone puisqu'ils sont toujours générés à partir de la même base. L'immuabilité est indépendante de tout outil ou flux de travail utilisé pour créer l'AMI.
- Infrastructure mutable : modèle de maintenance d'infrastructure typique pour les stacks qui ne sont pas des groupes Amazon EC2 Auto Scaling et qui contiennent une seule instance ou seulement quelques instances. Ce modèle représente le plus fidèlement le déploiement de système traditionnel basé sur le matériel, dans lequel un système est déployé au début de son cycle de vie, puis des mises à jour sont ajoutées à ce système au fil du temps. Toute mise à jour du système est appliquée aux instances individuellement et peut entraîner une interruption du système (selon la configuration de la pile) en raison du redémarrage de l'application ou du système.
- Groupes de sécurité : pare-feux virtuels permettant à votre instance de contrôler le trafic entrant et sortant. Les groupes de sécurité agissent au niveau instance, et non au niveau sous-réseau. Par conséquent, un ensemble différent de groupes de sécurité peut être attribué à chaque instance d'un sous-réseau de votre VPC.
- Contrats de niveau de service (SLAs) : partie des contrats que vous concluez avec AMS et qui définissent le niveau de service attendu.
- SLA non disponible et indisponibilité :
  - Une demande d'API que vous avez soumise qui entraîne une erreur.
  - Une demande de console que vous avez soumise qui entraîne une réponse HTTP 5xx (le serveur est incapable d'exécuter la demande).

  - Toutes les Service AWS offres constituant des piles ou des ressources au sein de votre infrastructure gérée par AMS sont en état de « rupture de service », comme indiqué dans le Service Health Dashboard.
  - L'indisponibilité résultant directement ou indirectement d'une exclusion d'AMS n'est pas prise en compte pour déterminer l'éligibilité aux crédits de service. Les services sont considérés comme disponibles sauf s'ils répondent aux critères d'indisponibilité.
- Objectifs de niveau de service (SLOs) : partie des contrats que vous avez conclus avec AMS qui définissent des objectifs de service spécifiques pour les services AMS.

Conditions d'application des correctifs :

- Correctifs obligatoires : mises à jour de sécurité critiques pour résoudre les problèmes susceptibles de compromettre l'état de sécurité de votre environnement ou de votre compte. Une « mise à jour de sécurité critique » est une mise à jour de sécurité considérée comme « critique » par le fournisseur d'un système d'exploitation compatible avec AMS.
- Correctifs annoncés ou publiés : les correctifs sont généralement annoncés et publiés selon un calendrier. Les correctifs émergents sont annoncés lorsque le besoin du correctif a été découvert et, généralement, peu de temps après, le correctif est publié.
- Module complémentaire de correctif : application de correctifs basée sur des balises pour les instances AMS qui exploite les fonctionnalités AWS Systems Manager (SSM) afin que vous puissiez étiqueter des instances et leur faire appliquer des correctifs à l'aide d'une ligne de base et d'une fenêtre que vous configurez.
- Méthodes de correction :
  - Application de correctifs sur place : application de correctifs effectuée en modifiant des instances existantes.
  - Correctif de remplacement de l'AMI : application de correctifs effectuée en modifiant le paramètre de référence de l'AMI d'une configuration de lancement de groupe EC2 Auto Scaling existante.
- Fournisseur de correctifs (fournisseurs de systèmes d'exploitation, tiers) : les correctifs sont fournis par le fournisseur ou l'organe directeur de l'application.
- Types de correctifs :
  - Mise à jour de sécurité critique (CSU) : mise à jour de sécurité considérée comme « critique » par le fournisseur du système d'exploitation pris en charge.

  - Mise à jour importante (IU) : mise à jour de sécurité considérée comme « importante » ou mise à jour non liée à la sécurité considérée comme « critique » par le fournisseur du système d'exploitation pris en charge.
  - Autre mise à jour (OU) : mise à jour par le fournisseur d'un système d'exploitation pris en charge qui n'est ni un CSU ni une interface utilisateur.
- Correctifs pris en charge : AMS prend en charge les correctifs au niveau du système d'exploitation. Les mises à niveau sont publiées par le fournisseur pour corriger des failles de sécurité ou d'autres bogues ou pour améliorer les performances. Pour obtenir la liste des configurations actuellement prises en charge OSs, consultez la section Configurations de support.

Termes de sécurité :

- Detective Controls : bibliothèque de moniteurs créés ou activés par AMS qui fournissent une surveillance continue des environnements gérés par les clients et des charges de travail pour les configurations qui ne sont pas conformes aux contrôles de sécurité, opérationnels ou clients, et qui prennent des mesures en informant les propriétaires, en modifiant ou en mettant fin aux ressources de manière proactive.

Modalités de la demande de service :

- Demande de service : demande de votre part concernant une action que vous souhaitez qu'AMS entreprenne en votre nom.
- Notification d'alerte : notification publiée par AMS sur votre page de liste de demandes de service lorsqu'une alerte AMS est déclenchée. Le contact configuré pour votre compte est également averti par la méthode configurée (par exemple, e-mail). Si vous avez des balises de contact sur vos instances/ressources et que vous avez donné votre accord à votre responsable de prestation de services cloud (CSDM) pour les notifications basées sur des balises, les informations de contact (valeur clé) figurant dans la balise sont également notifiées pour les alertes AMS automatisées.
- Notification de service : notification d'AMS publiée sur votre page de liste de demandes de service.

Termes divers :

- Interface AWS Managed Services : pour AMS : la console AWS Managed Services Advanced, l'API AMS CM et Support l'API. Pour AMS Accelerate : la Support console et Support l'API.

- Satisfaction client (CSAT) : AMS CSAT s'appuie sur des analyses approfondies, notamment des évaluations de correspondance pour chaque cas ou correspondance lorsqu'elle est envoyée, des enquêtes trimestrielles, etc.
- DevOps: DevOps est une méthodologie de développement qui préconise fortement l'automatisation et le suivi à toutes les étapes. DevOps vise à raccourcir les cycles de développement, à augmenter la fréquence des déploiements et à rendre les versions plus fiables en réunissant les fonctions traditionnellement distinctes du développement et des opérations sur une base d'automatisation. Lorsque les développeurs peuvent gérer les opérations et que les opérations informent le développement, les problèmes sont découverts et résolus plus rapidement, et les objectifs commerciaux sont plus facilement atteints.
- ITIL : Information Technology Infrastructure Library (appelée ITIL) est un framework ITSM conçu pour standardiser le cycle de vie des services informatiques. L'ITIL est organisé en cinq étapes qui couvrent le cycle de vie des services informatiques : stratégie des services, conception des services, transition des services, fonctionnement des services et amélioration des services.
- Gestion des services informatiques (ITSM) : ensemble de pratiques qui alignent les services informatiques sur les besoins de votre entreprise.
- Services de surveillance gérés (MMS) : AMS exploite son propre système de surveillance, le Managed Monitoring Service (MMS), qui prend en compte les événements de AWS santé et agrège les données d' CloudWatch Amazon, ainsi que les données provenant Services AWS d'autres fournisseurs, en informant les opérateurs AMS (en ligne 24 heures sur 24, 7 jours sur 7) de toute alarme créée par le biais d'une rubrique Amazon Simple Notification Service (Amazon SNS).
- Espace de noms : lorsque vous créez des politiques IAM ou que vous travaillez avec Amazon Resource Names (ARNs), vous les identifiez Service AWS en utilisant un espace de noms. Vous utilisez les espaces de noms lors de l'identification des actions et des ressources.

Quel est mon modèle de fonctionnement ?

En tant que client AMS, votre entreprise a décidé de séparer les opérations d'application et d'infrastructure et d'utiliser AMS pour les opérations d'infrastructure. AMS travaillera avec votre équipe de conception et de développement d'applications ainsi qu'avec votre équipe de conception d'infrastructure pour garantir le bon fonctionnement des opérations de votre infrastructure. Le graphique suivant illustre ce concept :

AMS prend la responsabilité des opérations de votre AWS infrastructure tandis que vos équipes sont responsables des opérations de vos applications. En tant qu'équipes de conception d'applications et d'infrastructures, vous devez comprendre qui exploitera l'application une fois qu'elle aura été déployée en production dans l'infrastructure AMS. Ce guide couvre les approches courantes en matière de conception d'infrastructure en ce qui concerne le déploiement et la maintenance des applications.

Gestion des services dans AWS Managed Services

Rubriques

Comment le service AMS fonctionne pour vous.

Gouvernance des comptes dans AWS Managed Services

Cette section traite de la gouvernance des comptes AMS.

Vous êtes désigné comme responsable de la prestation de services cloud (CSDM) qui fournit une assistance consultative au sein d'AMS et possède une compréhension détaillée de votre cas d'utilisation et de votre architecture technologique pour l'environnement géré. CSDMs travailler avec les responsables de comptes, les responsables de comptes techniques, les architectes cloud AWS Managed Services (CAs) et les architectes de solutions AWS (SAs), le cas échéant, pour aider à lancer de nouveaux projets et formuler des recommandations sur les meilleures pratiques tout au long des processus de développement et d'exploitation des logiciels. Le CSDM est le principal point de contact pour AMS. Les principales responsabilités de votre CSDM sont les suivantes :

- Organisez et animez des réunions mensuelles d'évaluation des services avec les clients.
- Fournissez des détails sur la sécurité, les mises à jour logicielles pour l'environnement et les opportunités d'optimisation.
- Répondez à vos exigences, y compris les demandes de fonctionnalités pour AMS.
- Répondez aux demandes de facturation et de rapports de service et résolvez-les.
- Fournissez des informations pour les recommandations d'optimisation financière et de capacité.

Début du service dans AWS Managed Services

Début du service : La date de début du service pour un compte AWS Managed Services est le premier jour du premier mois civil après lequel AWS vous informe que les activités définies dans les exigences d'intégration pour ce compte AWS Managed Services sont terminées ; à condition que si AWS envoie une telle notification après le 20e jour d'un mois civil, la date de début du service soit le premier jour du deuxième mois civil suivant la date de cette notification.

Début du service

- R représente la partie responsable qui fait le travail pour accomplir la tâche.
- Je signifie informé ; une partie qui est informée des progrès, souvent uniquement une fois la tâche ou le résultat livrable terminé.

Début du service

Étape #	Titre de l'étape	Description	Client	AMS
1.	Transfert du compte AWS du client	Le client crée un nouveau compte AWS et le transmet à AWS Managed Services	R	I
2.	Compte AWS Managed Services : conception	Finalisation de la conception du compte AWS Managed Services	I	R
3.	Compte AWS Managed Services - création	Un compte AWS Managed Services est créé conformément à la conception de l'étape 2	I	R

Gestion de la relation client (CRM)

AWS Managed Services (AMS) fournit un processus de gestion de la relation client (CRM) pour garantir l'établissement et le maintien d'une relation bien définie avec vous. Le fondement de cette

relation repose sur la connaissance qu'AMS a de vos besoins commerciaux. Le processus CRM facilite une compréhension précise et complète de :

- Les besoins de votre entreprise et comment y répondre
- Vos capacités et vos contraintes
- AMS et vos différentes responsabilités et obligations

Le processus CRM permet à AMS d'utiliser des méthodes cohérentes pour vous fournir des services et assurer la gouvernance de votre relation avec AMS. Le processus CRM inclut :

- Identifier vos principales parties prenantes
- Création d'une équipe de gouvernance
- Conduite et documentation des réunions d'évaluation des services avec vous
- Mise en place d'une procédure officielle de traitement des plaintes relatives au service assortie d'une procédure d'escalade
- Mise en œuvre et suivi de votre processus de satisfaction et de feedback
- Gérer votre contrat

Processus CRM

Le processus CRM inclut les activités suivantes :

- Identifier et comprendre les processus et les besoins de votre entreprise. Votre accord avec AMS identifie vos parties prenantes.
- Définition des services à fournir pour répondre à vos besoins et exigences.
- Rencontre avec vous lors des réunions de révision des services pour discuter de toute modification de l'étendue du service AMS, du SLA, du contrat et des besoins de votre entreprise. Des réunions intérimaires peuvent avoir lieu avec vous pour discuter des performances, des réalisations, des problèmes et des plans d'action.
- Surveillez votre satisfaction en utilisant notre enquête de satisfaction client et les commentaires fournis lors des réunions.
- Rendre compte des performances sur des rapports de performance mensuels mesurés en interne.
- Passez en revue le service avec vous afin de déterminer les possibilités d'amélioration. Cela inclut des communications fréquentes avec vous concernant le niveau et la qualité du service AMS fourni.

Réunions CRM

Les responsables de la prestation de services cloud d'AMS (CSDMs) vous rencontrent régulièrement pour discuter des pistes de service (opérations, sécurité et innovations en matière de produits) et des pistes exécutives (rapports de niveau de service, mesures de satisfaction et évolution des besoins de votre entreprise).

Réunion	Objectif	Mode	Les participants
Révision hebdomadaire du statut (facultatif)	Problèmes ou incidents en suspens, correctifs, événements de sécurité, dossiers de problèmes Tendance opérationnelle sur 12 semaines (+/- 6) Préoccupations des opérateurs d'applications Horaire du week-end	Client sur site location/ Telecom/Chime	AMS : CSDM et architecte cloud (CA) Membres de l'équipe assignés par le client (ex : équipes Cloud/ Infrastructure, Support des applications, équipes d'architecture, etc.)
Bilan mensuel de l'activité	Examiner les performances des niveaux de service (rapports, analyses et tendances) Analyse financière Feuille de route du produit CSAT	Client sur site location/ Telecom/Chime	AMS : CSDM, architecte cloud (CA), équipe de compte AMS, chef de produit technique AMS (TPM) (facultatif), responsable AMS OPS (facultatif) Vous : représentant de l'opérate

Réunion	Objectif	Mode	Les participants
			ur de l'applica tion
Revue d'activité trimestrielle	Performances et tendances du tableau de bord et des accords de niveau de service (SLA) (6 mois) Plans/migrations des prochains mois, du 3 au 9 septembre 2012 Risque et atténuation des risques Initiatives d'amélioration clés Eléments de la feuille de route Opportunités alignées sur l'orienta tion future Financiers Initiatives de réduction des coûts Optimisation commerciale	Localisation du client sur site	AMS : CSDM, architecte cloud, équipe chargée des comptes AMS, directeur du service AMS, responsable des opérations AMS Vous : représent ant de l'opérate ur de l'applica tion, représent ant du service, directeur du service

Arrangements de réunions CRM

L'AMS CSDM est chargé de documenter la réunion, notamment :

- Création de l'ordre du jour, y compris les mesures à prendre, les problèmes et la liste des participants.
- Création de la liste des mesures examinées lors de chaque réunion pour s'assurer que les mesures sont achevées et résolues dans les délais.
- Distribution du compte rendu de la réunion et de la liste des mesures à prendre aux participants par e-mail dans un délai d'un jour ouvrable après la réunion.
- Stockage des comptes rendus de réunion dans le référentiel de documents approprié.

En l'absence du CSDM, le représentant de l'AMS qui dirige la réunion crée et distribue les procès-verbaux.

> **Note**
>
> Votre CSDM travaille avec vous pour établir la gouvernance de votre compte.

Rapports mensuels du CRM

Votre AMS CSDM prépare et envoie des présentations mensuelles sur les performances des services. Les présentations incluent des informations sur les points suivants :

- Date du rapport
- Résumé et aperçus :
  - Principaux appels : nombre total et actif, état de l'application des correctifs, statut d'intégration du compte (uniquement pendant l'intégration), résumés des problèmes spécifiques aux clients
  - Performances : statistiques sur la résolution des incidents, les alertes, les correctifs, les demandes de modification (RFCs), les demandes de service et la disponibilité des consoles et des API
  - Problèmes, défis, préoccupations et risques : état des problèmes spécifiques au client
  - Éléments à venir : plans d'intégration ou de résolution des incidents spécifiques au client
- Ressources gérées : graphiques et diagrammes à secteurs des piles
- Mesures AMS : mesures de surveillance et d'événements, mesures d'incidents, mesures de conformité aux accords de niveau de service AMS, mesures de demande de service, mesures de gestion du changement, mesures de stockage, mesures de continuité, mesures de Trusted Advisor et résumés des coûts (présentés de plusieurs manières). Demandes de fonctionnalités. Informations de contact.

> **Note**
>
> Outre les informations décrites, votre CSDM vous informe également de tout changement important dans le champ d'application ou les termes, y compris le recours à des sous-traitants par AMS pour les activités opérationnelles.

AMS génère des rapports sur les correctifs et les sauvegardes que votre CSDM inclut dans votre rapport mensuel. Dans le cadre du système de génération de rapports, AMS ajoute à votre compte une infrastructure à laquelle vous n'avez pas accès :

- Un compartiment S3, avec les données brutes rapportées
- Une instance Athena, avec des définitions de requêtes pour interroger les données
- Un Glue Crawler pour lire les données brutes du compartiment S3

Optimisation des coûts dans AWS Managed Services

AWS Managed Services vous fournit des rapports détaillés sur l'utilisation des coûts et les économies chaque mois lors de vos évaluations commerciales mensuelles (MBRs).

AMS suit un ensemble standard de processus et de mécanismes pour identifier les moyens de réduire les coûts dans vos comptes gérés et vous aider à planifier et à mettre en œuvre les modifications afin d'optimiser vos dépenses AWS.

Note

AMS développe une vidéo pour aider à optimiser les coûts. La première étape consiste à vous fournir un PDF et une feuille de calcul Excel présentant les meilleures pratiques en matière d'optimisation des coûts. Pour accéder à ces ressources, ouvrez le fichier ZIP du guide rapide d'optimisation des coûts.

Cadre d'optimisation des coûts

AMS suit une approche en trois étapes pour optimiser vos coûts AWS :

1. Identifiez les pistes d'optimisation des coûts dans votre environnement géré
2. Vous présenter un plan d'optimisation des coûts
3. Aider à optimiser les coûts de manière mesurable

Identifier les pistes d'optimisation des coûts dans l'environnement géré

AMS utilise des outils AWS natifs tels que Cost Explorer et Trusted Advisor tout en tirant parti de plus de 20 modèles de réduction des coûts grâce à l'optimisation de l'architecture, aux optimisations AWS

axées sur les EC2 instances et aux comptes pour élaborer des recommandations de réduction de coûts personnalisées pour vous.

Certaines des recommandations d'optimisation sont les suivantes.

Recommandations d'optimisation architecturale :

- Utilisation optimale des classes de stockage S3 : Amazon S3 propose une gamme de classes de stockage pour répondre aux différentes exigences de charge de travail en fonction de l'accès aux données, de la résilience et du coût. La hiérarchisation intelligente S3 et l'analyse des classes de stockage S3 basées sur les besoins de charge de travail vous permettent de gérer efficacement les coûts de S3.
- Utilisation d'architectures de mise en cache : l'utilisation des instances de cache, le cas échéant, peut vous aider à remplacer certaines instances de base de données, tout en répondant à vos exigences en matière d'IOPS.
- Économies liées à la mise à niveau d'EBS : la migration de vos volumes EBS de gp2 à gp3 permet de réaliser des économies allant jusqu'à 20 % et vous pouvez tirer parti de performances de base prévisibles de 3 000 IOPS et de 125 Mbits/s, quelle que soit la taille du volume.
- Utilisation de l'élasticité : les fonctionnalités d'auto-scaling qu'elle AWS fournit permettent une utilisation efficace des ressources et des pistes d'optimisation des coûts. La révision et la mise à jour régulières des politiques de dimensionnement des instances en fonction des besoins permettent de réaliser des économies supplémentaires.

EC2 recommandations axées sur les instances

- Redimensionnement des instances : recommandations axées sur le dimensionnement des instances et sur les configurations optimales en fonction de l'utilisation. Les recommandations incluent également l'utilisation de la fonctionnalité Amazon EC2 Auto Scaling et le remplacement des EC2 AWS Lambda instances, le cas échéant, par du contenu Web statique sur Amazon S3, etc.
- Planification des instances : l'utilisation d'AMS Resource Scheduler pour démarrer et arrêter automatiquement les instances en fonction d'un calendrier permet de limiter les coûts, en particulier pour les instances hors production qui ne sont pas utilisées en dehors des heures ouvrables.
- Abonnement à des plans d'épargne : le plan d'épargne est le moyen le plus simple d'économiser sur AWS l'utilisation. Les EC2 Instance Savings Plans offrent jusqu'à 72 % d'économies par rapport à la tarification à la demande sur l'utilisation de vos EC2 instances Amazon. Les Amazon SageMaker AI Savings Plans offrent jusqu'à 64 % d'économies sur votre utilisation des services

Amazon SageMaker AI. AMS fournit des recommandations appropriées sur les plans d'épargne en fonction de votre utilisation AWS des ressources.

- Conseils d'utilisation et de consommation des instances réservées (RI) : les instances EC2 réservées Amazon (RI) offrent une réduction significative (jusqu'à 75 %) par rapport à la tarification à la demande et fournissent une réservation de capacité lorsqu'elles sont utilisées dans une zone de disponibilité spécifique.
- Utilisation d'instances ponctuelles : les charges de travail tolérantes aux pannes peuvent utiliser des instances ponctuelles et réduire les prix jusqu'à 90 %.
- Résiliation des instances inactives : identification et signalement des instances inactives ou peu utilisées qui peuvent être résiliées.

Recommandations axées sur les comptes

- Nettoyage du compte : Au niveau du compte, AMS identifie également les volumes EBS inutilisés, les CloudTrail traces dupliquées, les comptes vides contenant des ressources inutilisées, etc., et fournit des recommandations pour le nettoyage.
- Recommandations relatives aux accords de niveau de service : AMS examine régulièrement vos comptes Plus et Premium et recommande de choisir le niveau de SLA approprié pour les comptes.
- Optimisation de l'automatisation AMS : AMS optimise en permanence l'automatisation AMS et l'infrastructure utilisée pour fournir les services AMS.

Présenter aux clients et aider à planifier

AMS réalise des revues commerciales mensuelles (MBRs) avec les principales parties prenantes du client et présente les pistes, mécanismes et recommandations de réduction des coûts identifiés, ainsi que les économies potentielles. Nous travaillons également avec vous pour planifier les changements nécessaires.

Aider à la mise en œuvre des recommandations et mesurer l'impact sur les coûts

AMS aide à atteindre et à mesurer les impacts sur les coûts et les changements d'optimisation.

Vous évaluez l'impact sur l'application, les critères de risque et de réussite des modifications recommandées, et vous soumettez les demandes de modification appropriées (RFCs) via la console AMS. AMS collabore avec vous et met en œuvre les modifications liées à l'optimisation des coûts dans vos comptes gérés. AMS mesure l'impact sur les coûts et inclut les économies réalisées dans les bilans d'activité mensuels (MBRs).

Matrice de responsabilité pour l'optimisation des coûts

Responsabilités en matière d'optimisation des coûts AMS.

Optimisation des coûts RACI

Activité	Client	AMS
Compilation des recommandations de réduction des coûts et préparation du rapport	I	R
Présentation d'un rapport sur les économies de coûts	C	R
Planification des changements associés à la réduction des coûts	R	C
Évaluation de l'impact du	R	C

Activité	Client	AMS
changement et des risques		
Récolter RFCs des fonds pour mettre en œuvre les changements	R	C
Révision RFCs et mise en œuvre des modifications	C	R
Tester l'application et valider la mise en œuvre des modifications	R	C

Activité	Client	AMS
Mesurer l'impact sur les coûts après le changement et le présenter au client	I	R

Heures de service dans AWS Managed Services

Fonctionnalité	AMS avancé
	Niveau Premium
Demande de service	24 h/24,
Gestion des incidents (P2-P3)	24 h/24,
Sauvegarde et restauration	24 h/24,
Gestion des correctifs	24 h/24,
Surveillance et alertes	24 h/24,
Demande de modification automatisée (RFC)	24 h/24,
Demande de modification non automatisée (RFC)	24 h/24,
Responsable de la prestation de services cloud (CSDM)	Du lundi au vendredi : de 8h00 à 17h00, heures d'ouverture locales

Obtenir de l'aide sur AWS Managed Services

AMS vous soutient en matière de gestion des incidents, de gestion des demandes de service et de gestion des modifications 24 heures sur 24, 7 jours sur 7, 365 jours par an (conformément au contrat de niveau de service AMS appliqué au compte).

Pour signaler un problème de performance des services AWS ou AMS ayant une incidence sur votre environnement géré, utilisez la console AMS et soumettez un rapport d'incident. Pour plus de détails, voir Signaler un incident. Pour des informations générales sur la gestion des incidents AMS, consultez la section Réponse aux incidents.

Pour demander des informations ou des conseils, ou pour demander des services supplémentaires à AMS, utilisez la console AMS et soumettez une demande de service. Pour plus de détails, voir Création d'une demande de service. Pour des informations générales sur les demandes de service AMS, consultez la section Gestion des demandes de service.

Développement d'applications

Processus et pratiques de développement d'applications qui permettent une conception et un déploiement efficaces des applications dans un environnement AWS Managed Services (AMS). AMS vous guide tout au long du processus de haut niveau suivant :

1. Imaginez et concevez une application à développer ou à intégrer à votre environnement géré par AMS. Quelques considérations :
    a. Comment allez-vous déployer votre application ? Avec l'automatisation à l'aide d'un outil de déploiement tel qu'Ansible, ou manuellement en téléchargeant directement les fichiers nécessaires ?
    b. Comment allez-vous mettre à jour votre candidature ? Avec une approche mutable mettant à jour chaque instance séparément, ou avec une approche immuable mettant à jour chaque instance avec une seule AMI mise à jour dans un groupe Auto Scaling ?
2. Planifiez et concevez l'infrastructure qui sera utilisée pour héberger l'application à l'aide des bibliothèques d' AWS architecture, du guide AWS « Well-Architected », d'AMS et d'autres experts en architecture cloud. Les sections suivantes de ce guide fournissent des informations qui peuvent vous aider à cet égard.
3. Sélectionnez une approche de déploiement de l'infrastructure :
    a. Suite complète : tous les composants de l'infrastructure sont déployés simultanément, ensemble.
    b. Niveau et égalité : les déploiements d'infrastructure sont déployés séparément, puis liés aux modifications des groupes de sécurité. Ce type de déploiement est également possible grâce à une configuration en série de composants de pile qui s'appuient les uns sur les autres ; par exemple, en spécifiant l'équilibreur de charge que vous avez créé précédemment lorsque vous créez un groupe Auto Scaling.
    c. Quels environnements, tels que Dev, Staging et Prod, utiliserez-vous ?
4. Choisissez les types de modification AMS (CTs) qui fourniront les piles, ou niveaux, nécessaires et prépareront les demandes de modification nécessaires (RFCs).
5. Soumettez le RFCs pour déclencher le déploiement de l'infrastructure dans l'environnement approprié.
6. Déployez l'application en utilisant l'approche de déploiement d'applications sélectionnée.
7. Retravaillez l'infrastructure et les applications selon les besoins.

8. Déployez l'infrastructure et les applications dans les environnements suivants appropriés, en supposant que votre premier déploiement concerne un environnement hors production.
9. La maintenance continue est assurée par AMS qui gère l'infrastructure sous-jacente et par vos équipes opérationnelles qui exploitent les infrastructures des applications.
10. Pour mettre hors service une application, mettez fin à l'infrastructure AMS correspondante.

Être bien architecturé

Chez AWS nous, les systèmes bien conçus augmentent considérablement les chances de réussite d'une entreprise. Le Centre AWS d'architecture fournit des conseils d'experts sur l'architecture dans le AWS Cloud.

Nous vous recommandons les articles et livres blancs suivants pour vous aider à comprendre les avantages et les inconvénients des décisions que vous devez prendre lors de la création de systèmes. AWS

Êtes-vous Well-Architected ? : présente le cadre AWS Well-Architected, basé sur six piliers :

- Excellence opérationnelle : le pilier de l'excellence opérationnelle se concentre sur le fonctionnement et le suivi des systèmes afin de créer de la valeur commerciale, ainsi que sur l'amélioration continue des processus et des procédures. Les sujets clés incluent la gestion et l'automatisation des changements, la réponse aux événements et la définition de normes pour gérer avec succès les opérations quotidiennes.
- Sécurité : le pilier de sécurité se concentre sur la protection des informations et des systèmes. Les sujets clés incluent la confidentialité et l'intégrité des données, l'identification et la gestion des personnes habilitées à faire quoi en matière de gestion des autorisations, la protection des systèmes et la mise en place de contrôles pour détecter les événements de sécurité.
- Fiabilité : le pilier de la fiabilité met l'accent sur la capacité à prévenir les défaillances et à y remédier rapidement afin de répondre à la demande des entreprises et des clients. Les sujets clés incluent les éléments fondamentaux relatifs à la configuration, aux exigences relatives à l'ensemble des projets, à la planification de la reprise et à la manière dont nous gérons le changement.
- Efficacité des performances : le pilier de l'efficacité des performances met l'accent sur l'utilisation efficace des ressources informatiques et informatiques. Les sujets clés incluent la sélection des types et des tailles de ressources appropriés en fonction des exigences de charge de travail, la surveillance des performances et la prise de décisions éclairées pour maintenir l'efficacité à mesure que les besoins de l'entreprise évoluent.

- Optimisation des coûts : le pilier de l'optimisation des coûts vise à éviter les coûts inutiles. Les sujets clés incluent la compréhension et le contrôle de l'utilisation de l'argent, la sélection du nombre de types de ressources le plus approprié et le plus approprié, l'analyse des dépenses au fil du temps et la mise à l'échelle pour répondre aux besoins de l'entreprise sans trop dépenser.
- Durabilité : le pilier de la durabilité met l'accent sur la capacité à améliorer continuellement les impacts sur le développement durable en réduisant la consommation d'énergie et en augmentant l'efficacité de tous les composants d'une charge de travail en maximisant les avantages des ressources allouées et en minimisant le total des ressources nécessaires.

AWS Well-Architected Framework : décrit AWS comment permet aux clients d'évaluer et d'améliorer leurs architectures basées sur le cloud et de mieux comprendre l'impact commercial de leurs décisions de conception. Il aborde les principes généraux de conception ainsi que les meilleures pratiques et directives spécifiques dans six domaines conceptuels qui constituent AWS les piliers du Well-Architected Framework.

Responsabilités de la couche application et responsabilités de la couche infrastructure dans AMS

En utilisant AMS, votre infrastructure et tout ce dont elle a besoin pour la maintenance et la croissance sont maintenues par AMS. Cependant, tout ce dont vous avez besoin pour line-of-business les applications ou les applications de produits est développé, déployé et maintenu par vos soins.

À l'aide d'outils de déploiement d'applications, tels que CodeDeploy and CloudFormation, ou Chef, Puppet, Ansible ou Saltstack, le déploiement de vos applications sur votre infrastructure gérée par AMS peut être entièrement automatisé.

Pour plus de détails sur ce que fait et ne fait pas AMS, consultezCe que nous faisons, ce que nous ne faisons pas.

Mutabilité des EC2 instances Amazon dans AMS

AMS et vous-même pouvez gérer les instances Amazon Elastic Compute Cloud (Amazon EC2) dans votre infrastructure de deux manières :

- Immuable : ce modèle utilise Amazon Machine Images (AMIs) cuit (créé) avec les fonctionnalités nécessaires. Lors du déploiement d'une mise à jour, les instances existantes sont détruites et

complètement remplacées par de nouvelles instances créées à partir d'une AMI mise à jour. Pour minimiser les temps d'arrêt, ce processus continu rend certaines instances non mises à jour et accessibles tandis que d'autres sont mises à jour jusqu'à ce que la nouvelle modification soit finalement complètement déployée.

- Mutable : Dans ce modèle, l'infrastructure est mise à jour avec le déploiement du nouveau code sur les systèmes existants dans le cloud. Ce modèle combine l'envoi manuel de mises à jour et leur utilisation infrastructure-as-code pour déployer des mises à jour et ne repose pas sur de nouvelles mises à jour AMIs.

Ces modèles de maintenance sont décrits plus en détail dans les sections suivantes de ce guide.

Utilisation de AWS Secrets Manager avec les ressources AMS

Dans de nombreux cas, vous devrez peut-être partager des secrets avec AMS, par exemple :

- Réinitialisation du mot de passe principal pour l'instance RDS
- Certificats pour équilibreurs de charge
- Obtention d'informations d'identification à long terme pour les utilisateurs IAM auprès d'AMS

Le moyen le plus sûr de partager des informations confidentielles avec AMS est d'utiliser le AWS Secrets Manager. Procédez comme suit :

1. Connectez-vous à la AWS console en utilisant votre accès fédéré et le CustomerReadOnly rôle pour la zone d'atterrissage à compte unique (SALZ) ; utilisez l'un de ces rôles, AWSManaged ServicesSecurityOpsRole AWSManagedServicesAdminRole, et AWSManaged ServicesChangeManagementRole pour la zone d'atterrissage multicompte (MALZ).
2. Accédez à la console AWS Secrets Manager et cliquez sur Enregistrer un nouveau secret.
3. Sélectionnez « Autre type de secret ».
4. Entrez la valeur secrète sous forme de texte brut et cliquez sur Suivant.
5. Entrez le nom et la description du secret. Le nom doit toujours commencer par « customer-shared/* ». Par exemple « customer-shared/license-2018 ». Une fois que vous avez terminé, continuez en cliquant sur Suivant.
6. Utilisez le chiffrement KMS par défaut.
7. Laissez la rotation automatique désactivée et cliquez sur Suivant.
8. Vérifiez et cliquez sur Store pour enregistrer le secret.

9. Répondez-nous dans une demande de service AMS avec le nom du secret et l'ARN, afin que nous puissions identifier et récupérer le secret. Pour plus d'informations sur la création de demandes de service, consultez la section Exemples de demandes de service.

Déploiement d'applications dans AMS

Lors de l'intégration, AWS Managed Services (AMS) travaille avec vous pour déterminer l'infrastructure dont vous avez besoin.

L'infrastructure de base inclut un cloud privé AWS virtuel (VPC), la sécurité des communications via un trust forestier ADFS, les sous-réseaux de base (DMZ, Shared Services et Private) mis en miroir sur deux zones de disponibilité et configurés avec un NAT géré, des bastions, des équilibreurs de charge publics (DX) et la sécurité requise. Direct Connect Les ressources de vos applications seront déployées dans votre sous-réseau privé ou celui des applications clients. Pour en savoir plus sur une architecture AMS typique, consultez le guide de l'utilisateur d'AWS Managed Services.

L'infrastructure que vous déployez une fois les bases terminées doit inclure tous les composants nécessaires à vos applications et au développement d'applications.

Capacités de déploiement d'applications dans AMS

Voici quelques-unes des manières dont vous pouvez déployer des applications dans AMS. Les détails de chaque méthode sont présentés ci-dessous.

Exemples de fonctionnalités de déploiement d'applications

Nom de méthode	Déploiement d'infrastructures	AMI ou élément (s) clé (s)	Installation de l'application
Applications mutables, AMI AMS			
Déploiement manuel des applications	Full stack CT ou Tier and Tie CTs	AMI fournie par AMS	Soumettez Access Management CT, installez l'application manuellement.
UserData déploiement d'applications avec un agent d'application (par exemple Chef, Puppet, etc.)			Utilisez Provisioning CT avec des UserData scripts qui installent un agent d'application et

Nom de méthode	Déploiement d'infrastructures	AMI ou élément (s) clé (s)	Installation de l'application
			script/agent installent l'application.
UserData déploiement d'applications sans agent (par exemple, Ansible, Salt SSH, etc.)			Soumettez Access Management CT, installez l'agent d'application. Déployez l'application à l'aide des outils de déploiement d'applications.
Applications mutables, AMI personnalisée			
Déploiement d'applications AMI personnalisées (non ASG)	Full stack CT ou Tier and Tie CTs	AMI personnalisée. AMI AMS -> personnaliser avec l'agent d'outillage de déploiement d'applications -> créer une EC2 instance (CT) -> créer une AMI (CT).	L'outillage de déploiement d'applications (c'est-à-dire Chef), qui tire parti des agents, déploie l'application.
Déploiement de l'application AWS Database Migration Service (DMS)	Synchronisation AWS DMS avec la pile de base de données relationnelle AMS existante.	AMI personnalisée	Le client ou le partenaire utilise AWS Database Migration Service ; AMS vérifie les composants AMS au lancement

Nom de méthode	Déploiement d'infrastructures	AMI ou élément (s) clé (s)	Installation de l'application
Déploiement de l'application Workload Ingest	Workload Ingest CT, migré par les partenaires instance/ AMI et initié par le client.		Le partenaire migre l'instance, crée une AMI dans le VPC géré par AMS du client ; le client utilise Workload Ingest CT pour lancer une pile dans AMS. Pour en savoir plus, consultez Ingestion de la charge de travail AMS (WIGS).
Applications immuables			
Déploiement d'applications AMI personnalisées (ASG)	Full stack CT ou Tier and Tie CTs	AMI AMS -> personnaliser -> créer une EC2 instance (CT) -> créer une AMI (CT) -> créer un groupe Auto Scaling.	Auto Scaling déploie l'application avec l'AMI personnalisée Pour en savoir plus, consultez Déploiements d'applications Tier and Tie dans AMS.
Applications mutables ou immuables			

Nom de méthode	Déploiement d'infrastructures	AMI ou élément (s) clé (s)	Installation de l'application
Déploiement CloudFormation d'applications de modèles personnalisés	CloudFormation modèle	CloudFormation Modèle AWS -> customize/ prepare pour AMS -> Déploiement | Ingestion | Stack from CloudFormation Template | Create (ct-36cn2avfrrj9v).	AMS déploie votre application sur votre compte à l'aide de votre CloudFormation modèle personnalisé et valide le déploiement de l'application. Pour en savoir plus, consultez CloudFormation Ingestion d'AMS.
Importation de base de données SQL	Opérations AMS (Autres | Autres CT)	Base de données SQL sur site -> fichier .bak -> Base de données SQL AMS RDS -> Gestion | Autre | Autre | Création (ct-1e1xtak34nx76) pour l'importation.	AMS importe votre base de données locale dans votre base de données RDS gérée par AMS. Pour en savoir plus, consultez Importation de base de données (DB) vers AMS RDS pour Microsoft SQL Server.

Nom de méthode	Déploiement d'infrastructures	AMI ou élément (s) clé (s)	Installation de l'application
Service de migration de base de données (DMS)	Opérations AMS (multiples CTs)	Base de données sur site -> Instance de réplication DMS -> Groupe de sous-réseaux de réplication DMS -> Point de terminaison cible DMS -> Point de terminaison source DMS -> Tâche de réplication DMS.	AMS importe votre base de données locale dans votre base de données S3 ou RDS cible gérée par AMS. Pour en savoir plus, consultez AWS Database Migration Serv (AWS DMS).
CodeDeploy déploiement d'applications	CodeDeploy	Application -> CodeDeploy application -> groupe CodeDeploy de déploiement -> CodeDeploy déploiement.	En fonction de l'utilisation, du déploiement sur place ou de Blue/Green l'application. Pour en savoir plus, consultez CodeDeploy demandes.

Planification du déploiement de votre application dans AMS

Pour un ensemble de questions recommandées auxquelles il faut répondre pour permettre les déploiements d'applications, voirAnnexe : Questionnaire d'accueil des candidatures. Les questions couvrent la description de votre :

- Récapitulatif du déploiement
- Composants de déploiement de l'infrastructure
- Plateforme d'hébergement d'applications
- Modèle de déploiement d'applications
- Dépendances des applications
- Certificats SSL pour les applications de produits

Ingestion de la charge de travail AMS (WIGS)

Rubriques

Utilisez le type de modification d'ingestion de charge de travail (CT) AMS avec un partenaire de migration vers le cloud AMS, pour déplacer vos charges de travail existantes vers un VPC géré par AMS. À l'aide de l'ingestion de charge de travail AMS, vous pouvez créer une AMI AMS personnalisée après avoir déplacé des instances migrées vers AMS. Cette section décrit le processus, les prérequis et les étapes que votre partenaire de migration et vous-même devez suivre pour l'ingestion de la charge de travail AMS.

> **Important**
>
> Le système d'exploitation doit être compatible avec l'ingestion de charge de travail AMS. Pour les systèmes d'exploitation pris en charge, voirMigration des charges de travail : conditions préalables pour Linux et Windows.
> Chaque charge de travail et chaque compte sont différents. AMS travaillera avec vous pour vous préparer à un résultat positif.

Le schéma suivant décrit le processus d'ingestion de la charge de travail AMS.

Migration des charges de travail : conditions préalables pour Linux et Windows

Avant d'ingérer une copie d'une instance sur site dans AWS Managed Services (AMS), certaines conditions doivent être remplies. Ce sont les prérequis, y compris ceux qui diffèrent entre les systèmes d'exploitation Windows et Linux.

> **Note**
>
> Pour simplifier le processus permettant de déterminer si les instances sont prêtes pour l'ingestion, des outils de validation pour Windows et Linux ont été créés. Ces outils peuvent être téléchargés et exécutés directement sur vos serveurs locaux ainsi que sur les EC2 instances d'AWS. Linux Pre-Wigs Validation.zip, Windows Pre-Wigs Validation.zip.

AVANT DE COMMENCER, pour Linux et Windows :

- Effectuez une analyse antivirus complète.
- L'instance doit avoir le profil d'`customer-mc-ec2-instance-profile`instance.
- Installez l'agent Amazon EC2 Systems Manager (SSM) et assurez-vous qu'il est opérationnel.
- Un minimum de 10 Go d'espace disque libre sur le volume racine est recommandé pour exécuter AMS workload ingest (WIGS). Sur le plan opérationnel, AMS recommande une utilisation du disque inférieure à 75 % et émet des alertes lorsque le taux d'utilisation du disque atteint 85 %.
- Déterminez un délai pour l'ingestion avec votre partenaire de migration.
- L'AMI personnalisée existe sous forme d' EC2 instance dans le compte AMS de production cible (c'est la responsabilité du partenaire de migration).

> **Important**
>
> Le système d'exploitation doit être compatible avec l'ingestion de charge de travail AMS.
>
> - Les systèmes d'exploitation suivants sont pris en charge :
>   - Microsoft Windows Server : 2008 R2, 2012, 2012 R2, 2016, 2019 et 2022
>   - Linux : Amazon Linux 2023, Amazon Linux 2 et Amazon Linux, CentOS 7.x, CentOS 6.5-6.10, Oracle Linux 7 : versions mineures 7.5 et supérieures, Oracle Linux 8 : versions

mineures jusqu'à 8.3, RHEL 8.x, RHEL 7.x, RHEL 6.5-6.10, SUSE Linux Enterprise Server 15 et versions spécifiques à SAP, SUSE Linux Enterprise Server 12, Ubuntu 18.04 SP3 SP4 SP5

- Les éléments suivants ne AMIs sont pas pris en charge :
  - AMI minimale d'Amazon Linux 2023.

> **Note**
>
> Les points de terminaison AMS API/CLI (amscm et amsskms) se trouvent dans la région AWS de Virginie du Nord,. `us-east-1` En fonction de la configuration de votre authentification et de la région AWS dans laquelle se trouvent votre compte et vos ressources, vous devrez peut-être en ajouter `--region us-east-1` lors de l'émission de commandes. Vous devrez peut-être également ajouter`--profile saml`, s'il s'agit de votre méthode d'authentification.

Prérequis pour LINUX

Respectez les exigences répertoriées dans Migration des charges de travail : conditions préalables pour Linux et Windows et assurez-vous de ce qui suit avant de soumettre une RFC WIGS :

- Les derniers pilotes réseau améliorés sont installés ; voir Mise en réseau améliorée sous Linux.
- Les composants logiciels tiers susceptibles d'entrer en conflit avec les composants AMS ont été supprimés :
  - Clients antivirus
  - Clients de sauvegarde
  - Logiciels de virtualisation (tels que VM Tools ou les services d'intégration Hyper-V)
  - Logiciel de gestion des accès (tel que SSSD, Centrify ou PBIS)
- Assurez-vous que le protocole SSH est correctement configuré : cela active temporairement l'authentification par clé privée pour le protocole SSH. AMS l'utilise avec notre outil de gestion de configuration. Utilisez les commandes suivantes :

```
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
```

- Assurez-vous que Yum est correctement configuré : RedHat nécessite une licence pour utiliser leurs référentiels Yum. L'instance doit être licenciée via un serveur satellite ou un serveur RedHat cloud. Utilisez l'un de ces liens si vous avez besoin d'une licence :
  - Red Hat Satellite
  - Accès au cloud Red Hat
- Si vous utilisez Red Hat Satellite, WIGS nécessite l'ajout de Red Hat Software Collections (RHSCL). Le système WIGS utilise RHSCL pour ajouter un interpréteur Python3.6 à côté de tout ce qui est configuré sur le système. Pour prendre en charge cette solution, les référentiels suivants doivent être disponibles :
  - rhel-server-rhscl
  - rhel-server-releases-optional

Prérequis Windows

Respectez les exigences répertoriées dans Migration des charges de travail : conditions préalables pour Linux et Windows et assurez-vous de ce qui suit avant de soumettre une RFC WIGS :

- La version 3 ou supérieure de Powershell est installée.
- AWS EC2 Config est installé sur l'instance avec la charge de travail que vous allez migrer.
- Installez les pilotes AWS qui prennent en charge les types d'instances de dernière génération : PV, ENA et NVMe. Vous pouvez utiliser les informations contenues dans ces liens :
  - Mise à niveau des pilotes PV sur vos instances Windows
  - Mise en réseau améliorée sous Windows
  - NVMe Pilotes AWS pour instances Windows
  - Partie 3 : Mise à niveau des NVMe pilotes AWS
  - Partie 5 : Installation du pilote de port série pour les instances Bare Metal
  - Partie 6 : Mise à jour des paramètres de gestion de l'alimentation
- (Facultatif mais recommandé) Désactiver les services critiques : définissez les services d'application critiques, tels que les bases de données, sur désactivés, mais assurez-vous que

toutes les modifications sont documentées afin qu'ils puissent revenir à leur mode de démarrage d'origine lors de la phase de vérification des applications.

- (Facultatif mais recommandé) Créez une AMI Failsafe à partir de l'instance préparée :
  - Utiliser le Déploiement | Composants de pile avancés | AMI | Créer
  - Lors de la création, ajoutez une balise Key=Name, Value=Application-ID_ IngestReady
  - Attendez que l'AMI soit créée avant de continuer
- Les composants logiciels tiers susceptibles d'entrer en conflit avec les composants AMS ont été supprimés :
  - Clients antivirus
  - Clients de sauvegarde
  - Logiciels de virtualisation (tels que VM Tools ou les services d'intégration Hyper-V)

> **Note**
>
> Le programme de End-of-Support migration pour Windows Server (EMP) inclut des outils permettant de migrer vos anciennes applications de Windows Server 2003, 2008 et 2008 R2 vers des versions plus récentes prises en charge sur AWS, sans aucune refactorisation.

Comment la migration modifie vos ressources

La RFC d'ingestion décrite dans cette section passe à l'étape suivante qui consiste à ajouter des configurations à l'instance, une fois celle-ci migrée vers votre compte AMS, afin qu'AMS puisse la gérer.

Les configurations ajoutées sont spécifiques à AMS comme suit.

Modifications apportées aux instances Linux ingérées :

- Logiciel installé :
  - Cloud Init : utilisé pour configurer les clés privées pour Jarvis Access.
  - Python 3 (langage de script) pour tous les systèmes d'exploitation pris en charge (à l'exception de CentOS 6, RHEL 8 OracleLinux , 7).
  - Scripts d'assistance AWS CloudFormation Python : AWS CloudFormation fournit des scripts utilisés pour installer des logiciels et démarrer des services sur des EC2 instances Amazon.

  - CLI AWS : L'interface de ligne de commande AWS est un outil open source basé sur le kit SDK AWS pour Python (Boto) qui fournit des commandes pour interagir avec les services AWS.
  - Agent AWS SSM : L'agent SSM traite les demandes du service Systems Manager et configure la machine comme indiqué dans la demande.
  - AWS CloudWatch Logs Agent : envoie les journaux à CloudWatch.
  - AWS CodeDeploy : service de déploiement qui automatise les déploiements d'applications sur des EC2 instances Amazon, des instances sur site ou des fonctions Lambda sans serveur.
  - Ruby : obligatoire pour CodeDeploy
  - Outils de performance du système (sysstat) : Sysstat contient divers utilitaires permettant de surveiller les performances du système et l'activité d'utilisation.
  - AD Bridge (anciennement PowerBroker Identity Services) : relie des hôtes autres que Microsoft à des domaines Active Directory.
  - Trend Micro Deep Security Agent : logiciel antivirus.
- Logiciel modifié :
  - Les instances sont configurées pour utiliser le fuseau horaire UTC.

Modifications apportées aux instances Windows ingérées :

- Logiciel installé :
  - Outils AWS pour Windows PowerShell : les outils AWS PowerShell permettent aux développeurs et aux administrateurs de gérer leurs services et ressources AWS dans l'environnement PowerShell de script.
  - Agent de sécurité Trend Micro Deep : protection antivirus
  - PowerShell Modules AMS contenant PowerShell du code permettant de contrôler le démarrage, la connexion à Active Directory, la surveillance, la sécurité et la journalisation.
- Logiciel modifié :
  - La version 1 du Server Message Block (SMB) est désactivée.
  - La gestion à distance de Windows (WinRM) est activée et configurée pour écouter sur le port 5986. Une règle de pare-feu autorisant ce port entrant est également créée.
- Logiciels susceptibles d'être installés ou modifiés :
  - Microsoft .Net Framework 4.5 (plate-forme de développement), si une version inférieure à .Net Framework 4.5 est détectée.

- Pour Windows 2012 et Windows 2012R2, nous effectuons une mise à niveau vers PowerShell la version 5.1.

Migration des charges de travail : processus standard

> **Note**
>
> Étant donné que deux parties sont nécessaires pour ce processus, cette section décrit les tâches de chacune : un partenaire de migration vers le cloud AMS (partenaire de migration) et un propriétaire de l'application (vous).

1. Partenaire de migration, configuration :
   a. Le partenaire de migration soumet une demande de service à AMS pour un rôle IAM dans le but de migrer votre instance. Pour plus de détails sur la soumission de demandes de service, consultez les exemples de demandes de service.
   b. Le partenaire de migration soumet une demande d'accès administrateur. L'équipe des opérations AMS fournit au partenaire de migration l'accès à votre compte via le rôle IAM demandé.
2. Partenaire de migration, Migrate Individual Workloads :
   a. Le partenaire de migration migre votre instance non AWS instance vers un sous-réseau de votre compte AMS via Amazon EC2 natif ou un autre outil de migration, avec `customer-mc-ec2-instance-profile` le profil d'instance IAM (doit figurer dans le compte).
   b. Le partenaire de migration soumet une RFC avec le formulaire Deployment | Ingestion | Stack from migration partner migrated instance | Create CT (ct-257p9zjk14ija) ; pour plus de détails sur la création et la soumission de cette RFC, consultez. Workload Ingest Stack : création

      La sortie d'exécution de la RFC renvoie un ID d'instance, une adresse IP et un ID d'AMI.

      Le partenaire de migration vous fournit l'ID d'instance de la charge de travail créée dans votre compte.
3. Vous, accédez à la migration et validez :

a. À l'aide du résultat d'exécution qui vous a été fourni (ID AMI, ID d'instance et adresse IP) par le partenaire de migration, soumettez une RFC d'accès, connectez-vous à la pile AMS nouvellement créée et vérifiez que votre application fonctionne correctement. Pour plus de détails, consultez la section Demande d'accès à une instance.

b. Si vous êtes satisfait, vous pouvez continuer à utiliser l'instance lancée comme une pile à un niveau et and/or utiliser l'AMI pour créer des piles supplémentaires, y compris des groupes Auto Scaling.

c. Si vous n'êtes pas satisfait de la migration, déposez une demande de service et référencez la pile et le RFC IDs ; AMS travaillera avec vous pour répondre à vos préoccupations.

CloudEndure le processus d'ingestion de la charge de travail dans la zone d'atterrissage est décrit ci-dessous.

Migration des charges de travail : zone CloudEndure d'atterrissage (SALZ)

Cette section fournit des informations sur la configuration d'une zone d'atterrissage à compte unique (SALZ) de migration intermédiaire pour les instances de transition CloudEndure (CE) devant être disponibles pour une RFC d'ingestion de charge de travail (WIGS).

Pour en savoir plus CloudEndure, consultez la section CloudEndure Migration.

> **Note**
>
> Il s'agit d'une LZ et d'un modèle de migration prédéfinis et renforcés en termes de sécurité.

Prérequis :

- Un compte client AMS
- Intégration du réseau et des accès entre le compte AMS et le client sur site
- Un CloudEndure compte
- Un flux de travail de pré-approbation pour une révision et une approbation de la sécurité AMS, exécuté avec votre CA and/or CSDM (par exemple, une utilisation abusive des informations d'identification permanentes de l'utilisateur IAM permet d'accéder à des instances et à des groupes de sécurité) create/delete

> **Note**
>
> Les processus spécifiques de préparation et de migration sont décrits dans cette section.

Préparation : Vous et l'opérateur AMS :

1. Préparez une demande de modification (RFC) avec le type de modification Management | Other | Other | Update vers AMS pour les ressources et mises à jour suivantes. Vous pouvez soumettre une mise à jour Autre | Autre mise à jour séparée RFCs, ou une seule. Pour plus de détails sur ce RFC/CT, voir Autre | Autre mise à jour avec les demandes suivantes :

   a. Attribuez un bloc d'adresse CIDR secondaire dans votre VPC AMS ; un bloc d'adresse CIDR temporaire qui sera supprimé une fois la migration terminée. Assurez-vous que le blocage n'entrera pas en conflit avec les itinéraires existants vers votre réseau sur site. Par exemple, si votre adresse CIDR VPC AMS est 10.0.0.0/16 et qu'il existe une route de retour vers votre réseau local de 10.1.0.0/16, le CIDR secondaire temporaire peut être 10.255.255.0/24. Pour plus d'informations sur les blocs d'adresse CIDR AWS, consultez la section Dimensionnement des VPC et des sous-réseaux.

   b. Créez un nouveau sous-réseau privé dans le VPC AMS du jardin initial. Exemple de nom :`migration-temp-subnet`.

   c. Créez une nouvelle table de routage pour le sous-réseau avec uniquement des routes VPC et NAT (Internet) locales, afin d'éviter les conflits avec le serveur source lors du transfert d'instance et d'éventuelles pannes. Assurez-vous que le trafic sortant vers Internet est autorisé pour le téléchargement des correctifs et que les prérequis d'AMS WIGS peuvent être téléchargés et installés.

   d. Mettez à jour votre groupe de sécurité Managed AD pour autoriser le trafic entrant et sortant. to/from `migration-temp-subnet` Demandez également que le groupe de sécurité de votre équilibreur de charge EPS (ELB) (ex :`mc-eps-McEpsElbPrivateSecurityGroup-M790XBZEEX74`) soit mis à jour pour autoriser le nouveau sous-réseau privé (c'est-à-dire). `migration-temp-subnet` Si le trafic provenant du sous-réseau dédié CloudEndure (CE) n'est pas autorisé sur les trois ports TCP, l'ingestion du WIGS échouera.

   e. Enfin, demandez une nouvelle politique CloudEndure IAM et un nouvel utilisateur IAM. <Customer Application Subnet (s) + Temp Migration Subnet>La politique nécessite votre

numéro de compte correct, et le sous-réseau indiqué IDs dans le `RunInstances` relevé doit être : votre.

Pour consulter une CloudEndure politique IAM préapprouvée par AMS : décompressez le fichier d'exemple de zone d'atterrissage WIGS Cloud Endure et ouvrez le. `customer_cloud_endure_policy.json`

> **Note**
>
> Si vous souhaitez une politique plus permissive, discutez-en avec vous CloudArchitect/CSDM et obtenez, si nécessaire, un examen de sécurité AMS et une approbation avant de soumettre une RFC mettant en œuvre la politique.

2. Les étapes de préparation à utiliser CloudEndure pour l'ingestion de la charge de travail AMS sont terminées et, si votre partenaire de migration a terminé ses étapes de préparation, la migration est prête à être effectuée. Le WIGS RFC est soumis par votre partenaire de migration.

> **Note**
>
> Les clés utilisateur IAM ne seront pas partagées directement, mais doivent être saisies dans la console de CloudEndure gestion par l'opérateur AMS lors d'une session de partage d'écran.

Préparation : partenaire de migration et opérateur AMS :

1. Créez un projet de CloudEndure migration.
   a. Lors de la création du projet, demandez à AMS de saisir les informations d'identification utilisateur IAM lors des sessions de partage d'écran.
   b. Dans Paramètres de réplication -> Choisissez le sous-réseau dans lequel les serveurs de réplication seront lancés, sélectionnez le customer-application-xsous-réseau.
   c. Dans Paramètres de réplication -> Choisissez les groupes de sécurité à appliquer aux serveurs de réplication, sélectionnez les deux groupes de sécurité Sentinel (privé uniquement et EgressAll).
2. Définissez les options de transfert pour les machines (instances).

a. Sous-réseau : migration-temp-subnet.

b. Groupe de sécurité : les deux groupes de sécurité « Sentinel » (privé uniquement et EgressAll).

   Les instances Cutover doivent être en mesure de communiquer avec l'AMS Managed AD et avec les points de terminaison publics AWS.

c. IP élastique : aucune

d. IP publique : non

e. Rôle IAM : customer-mc-ec profil à 2 instances

   Le rôle IAM doit permettre la communication SSM. Mieux vaut utiliser AMS par défaut.

f. Définissez les balises conformément à la convention.

Migration : Partenaire de migration :

1. Créez une pile factice sur AMS. Vous utilisez l'identifiant de pile pour accéder aux bastions.
2. Installez l'agent CloudEndure (CE) sur le serveur source. Pour plus de détails, consultez la section Installation des agents.
3. Créez des informations d'identification d'administrateur local sur le serveur source.
4. Planifiez une courte fenêtre de découpage et cliquez sur Découper lorsque vous êtes prêt. Cela finalise la migration et redirige les utilisateurs vers la région AWS cible.
5. Demandez un accès administrateur à la pile fictive, voir Demande d'accès administrateur.
6. Connectez-vous au bastion, puis à l'instance de transition à l'aide des informations d'identification d'administrateur local que vous avez créées.
7. Créez une AMI à sécurité intégrée. Pour plus de détails sur la création AMIs, consultez AMI Create.
8. Préparez l'instance pour l'ingestion, voirMigration des charges de travail : conditions préalables pour Linux et Windows.
9. Exécutez WIGS RFC sur l'instance, voir. Workload Ingest Stack : création

Compte AMS Tools (migration des charges de travail)

Votre compte Multi-Account Landing Zone Tools (avec VPC) permet d'accélérer les efforts de migration, d'améliorer votre position en matière de sécurité, de réduire les coûts et la complexité et de normaliser votre modèle d'utilisation.

Un compte d'outils fournit les éléments suivants :

- Une limite bien définie pour l'accès aux instances de réplication pour les intégrateurs de systèmes en dehors de vos charges de travail de production.
- Vous permet de créer une chambre isolée pour vérifier la présence de logiciels malveillants ou de routes réseau inconnues dans une charge de travail avant de la placer dans un compte associé à d'autres charges de travail.
- En tant que configuration de compte définie, elle permet d'accélérer l'intégration et la configuration pour la migration des charges de travail.
- Routes réseau isolées pour sécuriser le trafic depuis un compte sur site CloudEndure -> -> Outils -> Image ingérée AMS. Une fois qu'une image a été ingérée, vous pouvez la partager sur le compte de destination via une RFC AMS Management | Advanced stack components | AMI | Share (ct-1eiczxw8ihc18).

Schéma d'architecture de haut niveau :

Utilisez le type de changement de type Déploiement | Zone d'atterrissage gérée | Compte de gestion | Créer un compte d'outils (avec VPC) (ct-2j7q1hgf26x5c) pour déployer rapidement un compte d'outils et instancier un processus d'ingestion de charge de travail dans un environnement de zone d'atterrissage multicompte. Voir Compte de gestion, compte Outils : création (avec VPC).

> **Note**
>
> Nous recommandons d'avoir deux zones de disponibilité (AZs), car il s'agit d'un hub de migration.
> Par défaut, AMS crée les deux groupes de sécurité suivants (SGs) dans chaque compte. Confirmez que ces deux SGs éléments sont présents. S'ils ne sont pas présents, veuillez ouvrir une nouvelle demande de service auprès de l'équipe AMS pour en faire la demande.
>
> - SentinelDefaultSecurityGroupPrivateOnlyEgressAll

- InitialGarden-SentinelDefaultSecurityGroupPrivateOnly

Assurez-vous que les instances de CloudEndure réplication sont créées dans le sous-réseau privé où se trouvent des itinéraires permettant de revenir sur site. Vous pouvez le confirmer en vous assurant que les tables de routage du sous-réseau privé disposent d'une route par défaut vers TGW. Cependant, l'exécution d'une coupure de CloudEndure machine doit se faire dans le sous-réseau privé « isolé » où il n'y a pas de route de retour vers le réseau local, seul le trafic sortant d'Internet est autorisé. Il est essentiel de s'assurer que le transfert a lieu dans le sous-réseau isolé afin d'éviter d'éventuels problèmes avec les ressources sur site.

Prérequis :

1. Niveau de support Plus ou Premium.
2. Le compte d'application IDs pour la clé KMS sur laquelle AMIs ils sont déployés.
3. Le compte d'outils, créé comme décrit précédemment.

AWS Service de migration d'applications (AWS MGN)

AWS Le service de migration d'applications (AWS MGN) peut être utilisé dans votre compte MALZ Tools via le rôle `AWSManagedServicesMigrationRole` IAM créé automatiquement lors du provisionnement du compte Tools. Vous pouvez utiliser AWS MGN pour migrer des applications et des bases de données qui s'exécutent sur des versions prises en charge des systèmes d'exploitation Windows et Linux.

Pour plus d' up-to-dateinformations sur le Région AWS support, consultez la liste des services AWS régionaux.

Si votre système préféré n' Région AWS est pas actuellement pris en charge par AWS MGN, ou si le système d'exploitation sur lequel vos applications s'exécutent n'est pas actuellement pris en charge par AWS MGN, envisagez plutôt d'utiliser la CloudEndure migration dans votre compte Tools.

Demande d' AWS initialisation de MGN

AWS MGN doit être initialisé par AMS avant la première utilisation. Pour en faire la demande pour un nouveau compte Tools, soumettez une RFC Management | Other | Other depuis le compte Tools avec les informations suivantes :

```
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.
```

Une fois qu'AMS a terminé avec succès le RFC et initialisé AWS MGN dans votre compte Tools, vous pouvez l'utiliser `AWSManagedServicesMigrationRole` pour modifier le modèle par défaut en fonction de vos besoins.

Activer l'accès au nouveau compte AMS Tools

Une fois le compte Tools créé, AMS vous fournit un identifiant de compte. L'étape suivante consiste à configurer l'accès au nouveau compte. Procédez comme suit :

1. Mettez à jour les groupes Active Directory appropriés vers le compte approprié IDs.

   Les nouveaux comptes créés par AMS sont dotés de la politique de ReadOnly rôle ainsi que d'un rôle permettant aux utilisateurs de déposer des dossiers. RFCs

   Le compte Tools dispose également d'un rôle et d'un utilisateur IAM supplémentaires :

   - Rôle IAM : `AWSManagedServicesMigrationRole`
   - Utilisateur IAM : `customer_cloud_endure_user`
2. Demandez des politiques et des rôles pour permettre aux membres de l'équipe d'intégration des services de configurer le niveau d'outils suivant.

   Accédez à la console AMS et enregistrez les fichiers suivants RFCs :

   a. Créez une clé KMS. Utilisez Create KMS Key (auto) ou Create KMS Key (automatisation gérée).

      Lorsque vous utilisez KMS pour chiffrer les ressources ingérées, l'utilisation d'une clé KMS unique partagée avec les autres comptes de l'application Multi-Account Landing Zone permet de sécuriser les images ingérées afin qu'elles puissent être déchiffrées dans le compte de destination.

   b. Partagez la clé KMS.

Utilisez le type de modification Management | Advanced stack components | KMS key | Share (automatisation gérée) (ct-05yb337abq3x5) pour demander que la nouvelle clé KMS soit partagée avec les comptes de votre application où résidera la clé ingérée. AMIs

Exemple graphique de la configuration finale d'un compte :

Exemple de politique CloudEndure IAM pré-approuvée par AMS

Pour consulter une CloudEndure politique IAM préapprouvée par AMS : décompressez le fichier d'exemple de zone d'atterrissage WIGS Cloud Endure et ouvrez le. `customer_cloud_endure_policy.json`

Test de la connectivité et de la end-to-end configuration du compte AMS Tools

1. Commencez par configurer CloudEndure et installer l' CloudEndure agent sur un serveur qui sera répliqué sur AMS.
2. Créez un projet dans CloudEndure.
3. Entrez les AWS informations d'identification partagées lorsque vous avez effectué les prérequis, via le gestionnaire de secrets.
4. Dans les paramètres de réplication :
   a. Sélectionnez les deux groupes de sécurité AMS « Sentinel » (privé uniquement EgressAll) pour l'option Choisissez les groupes de sécurité à appliquer aux serveurs de réplication.
   b. Définissez les options de transfert pour les machines (instances). Pour plus d'informations, reportez-vous à l'étape 5. Découper
   c. Sous-réseau : sous-réseau privé.
5. Groupe de sécurité :
   a. Sélectionnez les deux groupes de sécurité AMS « Sentinel » (privé uniquement et EgressAll).
   b. Les instances de transition doivent communiquer avec l'Active Directory (MAD) géré par AMS et avec les points de terminaison publics : AWS
      i. IP élastique : aucune
      ii. IP publique : non
      iii. Rôle IAM : customer-mc-ec profil à 2 instances
   c. Définissez les balises conformément à votre convention de balisage interne.

6. Installez l' CloudEndure agent sur la machine et recherchez l'instance de réplication qui apparaîtra dans votre compte AMS dans la console EC2.

Le processus d'ingestion d'AMS :

Hygiène des comptes AMS Tools

Vous devrez procéder au nettoyage une fois que vous aurez terminé de partager l'AMI dans le compte et que vous n'aurez plus besoin des instances répliquées :

- Après l' WIGs ingestion de l'instance :
  - Instance de transition : au minimum, arrêtez ou mettez fin à cette instance, une fois le travail terminé, via la console AWS
  - Sauvegardes d'AMI avant ingestion : supprimez une fois que l'instance a été ingérée et que l'instance sur site a été arrêtée
  - Instances ingérées par AMS : désactivez la pile ou mettez-la hors service une fois que l'AMI a été partagée
  - AMS-ingested AMIs : Supprimer une fois le partage avec le compte de destination terminé
- Nettoyage de fin de migration : documentez les ressources déployées via le mode développeur pour vous assurer que le nettoyage a lieu régulièrement, par exemple :
  - Groupes de sécurité
  - Ressources créées via Cloud-formation
  - Réseau ACK
  - Sous-réseau
  - VPC
  - Table de routage
  - Rôles
  - Utilisateurs et comptes

Migration à grande échelle - Migration Factory

Voir Présentation de la solution AWS CloudEndure Migration Factory.

Migration des charges de travail : validation préalable à l'ingestion de Linux

Vous pouvez vérifier que votre instance est prête à être incorporée dans votre compte AMS. La validation préalable à l'ingestion de la charge de travail (WIGS) permet de vérifier le type de système d'exploitation, l'espace disque disponible, l'existence de logiciels tiers conflictuels, etc. Lorsqu'elle est exécutée, la validation préalable à l'ingestion du WIGS produit un tableau à l'écran, avec un fichier journal facultatif. Les résultats fournissent un pass/fail statut pour chaque contrôle de validation ainsi que la raison de tout échec. En outre, vous pouvez personnaliser les tests de validation en fonction de vos besoins.

Questions fréquemment posées :

- Comment utiliser la validation préalable à l'ingestion de Linux WIGS ?

  Procédez comme suit pour télécharger et utiliser les scripts de validation préalable à l'ingestion d'AMS Linux WIGS :

  1. Téléchargez un fichier ZIP contenant les scripts de validation

     Fichier zip de validation préalable à l'ingestion de Linux WIGS.
  2. Décompressez les règles jointes dans le répertoire de votre choix.
  3. Suivez les instructions du fichier readme.md.
- Quelles sont les validations effectuées par la validation préalable à l'ingestion de Linux WIGS ?

  La solution de validation préalable à l'ingestion AMS Linux WIGS valide les points suivants :

  1. Il y a au moins 5 gigaoctets libres sur le volume de démarrage.
  2. Le système d'exploitation est supporté par AMS.
  3. L'instance possède un profil d'instance spécifique.
  4. L'instance ne contient pas de logiciel antivirus ni de logiciel de virtualisation.
  5. SSH est correctement configuré.
  6. L'instance a accès aux référentiels Yum.
  7. Des pilotes réseau améliorés sont installés.
  8. L'instance possède l'agent SSM et il est en cours d'exécution.
- Pourquoi un fichier de configuration personnalisé est-il pris en charge ?

  Les scripts sont conçus pour s'exécuter à la fois sur des serveurs physiques sur site et sur des EC2 instances AWS. Toutefois, comme indiqué dans la liste ci-dessus, certains tests échoueront

s'ils sont exécutés sur site. Par exemple, un serveur physique dans un centre de données n'aurait pas de profil d'instance. Dans de tels cas, vous pouvez modifier le fichier de configuration pour ignorer le test du profil d'instance afin d'éviter toute confusion.

- Comment puis-je m'assurer que je dispose de la dernière version du script ?

  Une up-to-date version de la solution de validation préalable à l'ingestion de Linux WIGS sera disponible dans la section AMS Helper Files de la page de documentation principale.

- Le script est-il en lecture seule ?

  Le script est conçu pour être en lecture seule, à l'exception des fichiers journaux qu'il produit, mais les meilleures pratiques doivent être suivies pour exécuter le script dans un environnement hors production.

- La validation préalable à l'ingestion de WIGS est-elle disponible pour Windows ?

  Oui. Il est disponible dans la section AMS Helper Files de la page de documentation principale.

Migration des charges de travail : validation préalable à l'ingestion de Windows

Vous pouvez utiliser le script de WIGs pré-validation pour vérifier que votre instance est prête à être incorporée dans votre compte AMS. La validation préalable à l'ingestion de la charge de travail (WIGS) effectue des vérifications telles que le type de système d'exploitation, l'espace disque disponible, l'existence de logiciels tiers conflictuels, etc. Lorsqu'elle est exécutée, la validation préalable à l'ingestion du WIGS produit un tableau affiché à l'écran et un fichier journal facultatif. Les résultats fournissent un pass/fail statut pour chaque contrôle de validation ainsi que la raison de l'échec. En outre, vous pouvez personnaliser les tests de validation.

Questions fréquemment posées :

- Comment utiliser la validation préalable à l'ingestion de Windows WIGS ?

  Vous pouvez exécuter la validation à partir d'une interface graphique et d'un navigateur Web, ou vous pouvez utiliser Windows PowerShell, SSM Run Command ou SSM Session Manager.

  Option 1 : Exécuter à partir d'une interface graphique et d'un navigateur Web

  Pour exécuter la WIGs prévalidation de Windows à partir d'une interface graphique et d'un navigateur Web, procédez comme suit :

1. Téléchargez un fichier ZIP contenant les scripts de validation :

   Fichier ZIP de validation préalable à l'ingestion de Windows WIGS.

2. Décompressez les règles jointes dans le répertoire de votre choix.
3. Suivez les instructions du fichier README.md.

Option 2 : Exécuter depuis Windows PowerShell, SSM Run Command ou SSM Session Manager

Windows 2016 et versions ultérieures

1. Téléchargez le fichier ZIP contenant les scripts de validation.

```
$DestinationFile = "$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. Supprimez les fichiers existants de`C:\Users\AppData\Local\Temp \AWSManagedServices.PreWigs.Validation`.

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

3. Invoquez le script.

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

4. Décompressez les fichiers joints dans le répertoire de votre choix.

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

5. Exécutez le script de validation de manière interactive et visualisez les résultats.

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

6. (Facultatif) Pour capturer les codes d'erreur répertoriés dans la section Codes de sortie, exécutez le script sans l'`RunWithoutExitCodes`option. Notez que cette commande met fin à la PowerShell session active.

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

Windows 2012 R2 et versions antérieures

Si vous utilisez Windows Server 2012R2 ou une version antérieure, vous devez définir le protocole TLS avant de télécharger le fichier zip. Pour configurer le protocole TLS, procédez comme suit :

1. Téléchargez le fichier ZIP contenant les scripts de validation.

```
$DestinationFile = "$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. S'il existe des fichiers de validation, supprimez-les.

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

3. Définissez la version TLS.

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

4. Téléchargez la validation WIG.

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

5. Décompressez les règles jointes dans le répertoire de votre choix.

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

6. Exécutez le script de validation de manière interactive et visualisez les résultats.

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

7. (Facultatif) Pour capturer les codes d'erreur répertoriés dans la section Codes de sortie, exécutez le script sans l' RunWithoutExitCodes option. Notez que cette commande met fin à la PowerShell session active.

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

> **Note**
>
> Vous pouvez télécharger et exécuter les PowerShell scripts. Pour ce faire, téléchargez le pre-wigs-validation-powershellfichier -scripts.zip.

- Quelles sont les validations effectuées par la validation préalable à l'ingestion de Windows WIGS ?

  La solution de validation préalable à l'ingestion AMS Windows WIGS valide les points suivants :

  1. Il y a au moins 10 gigaoctets libres sur le volume de démarrage.
  2. Le système d'exploitation est pris en charge par AMS.
  3. L'instance possède un profil d'instance spécifique.
  4. L'instance ne contient pas de logiciel antivirus ni de logiciel de virtualisation.
  5. Le protocole DHCP est activé sur au moins un adaptateur réseau.
  6. L'instance est prête pour Sysprep.
     - Pour 2008 R2, 2012 Base et R2, Sysprep vérifie que :
       - Il existe un fichier unattend.xml
       - Le fichier sppnp.dll (s'il existe) n'est pas endommagé
       - Le système d'exploitation n'a pas été mis à niveau
       - Sysprep n'a pas été exécuté plus de fois que le nombre maximum de fois conformément aux directives de Microsoft
     - Pour 2016 et les années ultérieures, toutes les vérifications ci-dessus sont ignorées car aucune ne pose de problème pour ce système d'exploitation
  7. Le sous-système WMI (Windows Management Instrumentation) est en bon état.
  8. Les pilotes requis sont installés.
  9. L'agent SSM est installé et en cours d'exécution.
  10. Un avertissement est émis pour vérifier si la machine est en période de grâce en raison de la configuration de la licence RDS.

11 Les clés de registre requises sont définies correctement. Pour plus de détails, consultez le fichier README dans le fichier zip de validation préalable à l'ingestion.

- Pourquoi un fichier de configuration personnalisé est-il pris en charge ?

  Les scripts sont conçus pour s'exécuter à la fois sur des serveurs physiques sur site et sur des EC2 instances AWS. Toutefois, comme indiqué dans la liste ci-dessus, certains tests échoueront s'ils sont exécutés sur site. Par exemple, un serveur physique dans un centre de données n'aurait pas de profil d'instance. Dans de tels cas, vous pouvez modifier le fichier de configuration pour ignorer le test du profil d'instance afin d'éviter toute confusion.

- Comment puis-je m'assurer que je dispose de la dernière version du script ?

  Une up-to-date version de la solution de validation préalable à l'ingestion de Windows WIGS sera disponible dans la section AMS Helper Files de la page de documentation principale.

- Le script est-il en lecture seule ?

  Le script est conçu pour être en lecture seule, à l'exception des fichiers journaux qu'il produit, mais les meilleures pratiques doivent être suivies pour exécuter le script dans un environnement hors production.

- La validation préalable à l'ingestion de WIGS est-elle disponible pour Linux ?

  Oui. La version Linux a été lancée le 31 octobre 2019. Il est disponible dans la section AMS Helper Files de la page de documentation principale.

Workload Ingest Stack : création

Migration d'une instance vers une pile AMS à l'aide de la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.

- Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

  Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
- Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.

3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

> **Note**
>
> Si la RFC est rejetée, le résultat d'exécution inclut un lien vers les CloudWatch journaux Amazon. AMS Workload Ingest (WIGS) est rejeté lorsque RFCs les exigences ne sont pas satisfaites, par exemple, si un logiciel antivirus est détecté sur l'instance. Les CloudWatch journaux incluront des informations sur l'exigence non satisfaite et les mesures à prendre pour y remédier.

Migration d'une instance vers une pile AMS à l'aide de la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

Vous pouvez utiliser la CLI AMS pour créer une instance AMS à partir d'une instance non-AMS migrée vers un compte AMS.

> **Note**
>
> Assurez-vous d'avoir respecté les conditions préalables ; voir Migration des charges de travail : conditions préalables pour Linux et Windows.

Pour vérifier la version du type de modification, utilisez cette commande :

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

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}"
```

CRÉATION D'UN MODÈLE :

1. 0utput le schéma JSON des paramètres d'exécution pour ce type de modification d'un fichier ; l'exemple le nomme .json : MigrateStackParams

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

2. Modifiez et enregistrez le fichier JSON des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

3. Sortez le fichier JSON du modèle RFC ; l'exemple le nomme MigrateStackRfc .json :

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

4. Modifiez et enregistrez le fichier MigrateStackRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez la RFC en spécifiant le MigrateStackRfc fichier et le MigrateStackParams fichier :

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

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

   La nouvelle instance apparaît dans la liste des instances du compte du propriétaire de l'application pour le VPC concerné.

6. Une fois le RFC terminé avec succès, informez le propriétaire de l'application afin qu'il puisse se connecter à la nouvelle instance et vérifier que la charge de travail est opérationnelle.

> **Note**
>
> Si la RFC est rejetée, le résultat d'exécution inclut un lien vers les CloudWatch journaux Amazon. AMS Workload Ingest (WIGS) est rejeté lorsque RFCs les exigences ne sont pas satisfaites, par exemple, si un logiciel antivirus est détecté sur l'instance. Les CloudWatch journaux incluront des informations sur l'exigence non satisfaite et les mesures à prendre pour y remédier.

Conseils

> **Note**
>
> Assurez-vous d'avoir respecté les conditions préalables ; voir Migration des charges de travail : conditions préalables pour Linux et Windows.

> **Note**
>
> Si une balise de l'instance en cours de migration possède la même clé qu'une balise fournie dans la RFC, la RFC échoue.

> **Note**
>
> Vous pouvez spécifier jusqu'à quatre cibles IDs, ports et zones de disponibilité.

> **Note**
>
> Si la RFC est rejetée, le résultat d'exécution inclut un lien vers les CloudWatch journaux Amazon. AMS Workload Ingest (WIGS) est rejeté lorsque RFCs les exigences ne sont pas satisfaites, par exemple, si un logiciel antivirus est détecté sur l'instance. Les CloudWatch journaux incluront des informations sur l'exigence non satisfaite et les mesures à prendre pour y remédier.

> **Note**
>
> Si la RFC est rejetée, le résultat d'exécution inclut un lien vers les CloudWatch journaux Amazon. AMS Workload Ingest (WIGS) est rejeté lorsque RFCs les exigences ne sont pas satisfaites, par exemple, si un logiciel antivirus est détecté sur l'instance. Les CloudWatch journaux incluront des informations sur l'exigence non satisfaite et les mesures à prendre pour y remédier.

Si nécessaire, consultez la section Défaillance de l'ingestion de charge de travail (WIGS).

CloudFormation Ingestion d'AMS

Le type de modification d' CloudFormation ingestion (CT) AMS AWS vous permet d'utiliser vos CloudFormation modèles existants, avec quelques modifications, pour déployer des piles personnalisées dans un VPC géré par AMS.

Rubriques

- CloudFormation Directives, meilleures pratiques et limites relatives à l'ingestion
- CloudFormation Ingestion : exemples
- Création d'une CloudFormation pile d'ingestion
- Mettre à jour CloudFormation la pile d'ingestion
- Approuver un ensemble CloudFormation de modifications à la pile d'ingestion
- Protection contre les mises à jour, CloudFormation piles et terminaisons
- Déploiements IAM automatisés à l'aide de CFN ingest ou de stack update dans AMS CTs

Le processus CloudFormation d'ingestion d'AMS implique les étapes suivantes :

- Préparez et téléchargez votre CloudFormation modèle personnalisé dans un compartiment S3, ou fournissez le modèle en ligne lors de la création de la RFC. Si vous utilisez un compartiment S3 avec une URL présignée, consultez presign pour plus d'informations.
- Soumettez le type CloudFormation de modification d'ingestion à AMS dans une RFC. Pour la procédure pas à pas du type de modification du type d'ingestion CFN, consultez. Création d'une CloudFormation pile d'ingestion Pour des exemples d'ingestion de CFN, voir. CloudFormation Ingestion : exemples
- Une fois votre stack créé, vous pouvez le mettre à jour et remédier à la dérive ; en outre, en cas d'échec de la mise à jour, vous pouvez explicitement approuver et implémenter la mise à jour. Toutes ces procédures sont décrites dans cette section.

  Pour plus d'informations sur la détection de la dérive CFN, voir Nouveau — Détection de la CloudFormation dérive.

> **Note**
>
> - Ce type de modification possède désormais une version 2.0. La version 2.0 est automatisée ; elle n'est pas exécutée manuellement. Cela permet d'accélérer l'exécution de la tomodensitométrie. Deux nouveaux paramètres sont introduits dans cette version : CloudFormationTemplate, qui vous permet de coller un CloudFormation modèle personnalisé dans le RFC VpcId, et qui permet d'utiliser l' CloudFormation ingestion avec la zone de landing multi-comptes AMS.

- La version 1.0 est un type de modification manuelle. Cela signifie qu'un opérateur AMS doit prendre certaines mesures avant que le type de modification puisse être conclu avec succès. Au minimum, un examen est requis. Cette version nécessite également que la valeur du paramètre CloudFormationTemplateS3Endpoint soit une URL pré-signée.

CloudFormation Directives, meilleures pratiques et limites relatives à l'ingestion

Pour qu'AMS puisse traiter votre CloudFormation modèle, certaines directives et restrictions s'appliquent.

Consignes

Pour réduire les CloudFormation erreurs lors de CloudFormation l'ingestion, suivez les instructions suivantes :

- N'intégrez pas d'informations d'identification ou d'autres informations sensibles dans le modèle : le CloudFormation modèle est visible dans la CloudFormation console. Vous ne souhaitez donc pas intégrer d'informations d'identification ou de données sensibles dans le modèle. Le modèle ne peut pas contenir d'informations sensibles. Les ressources suivantes ne sont autorisées que si vous utilisez AWS Secrets Manager pour la valeur :
  - `AWS::RDS::DBInstance` - [MasterUserPassword,TdeCredentialPassword]
  - `AWS::RDS::DBCluster` - [MasterUserPassword]
  - `AWS::ElastiCache::ReplicationGroup` - [AuthToken]

  > **Note**
  >
  > Pour plus d'informations sur l'utilisation d'un secret AWS Secrets Manager dans une propriété de ressource, consultez Comment créer et récupérer des secrets gérés dans AWS Secrets Manager à l'aide de CloudFormation modèles AWS et Utiliser des références dynamiques pour spécifier les valeurs des modèles.

- Utilisez les instantanés Amazon RDS pour créer des instances de base de données RDS. Vous évitez ainsi d'avoir à fournir un. MasterUserPassword
- Si le modèle que vous soumettez contient un profil d'instance IAM, celui-ci doit être préfixé par « client ». Par exemple, l'utilisation d'un profil d'instance nommé « example-instance-profile »

entraîne un échec. Utilisez plutôt un profil d'instance nommé « customer-example-instance-profile ».

- N'incluez aucune donnée sensible dans **`AWS::EC2::Instance`** - [UserData]. UserData ne doit pas contenir de mots de passe, de clés d'API ou d'autres données sensibles. Ce type de données peut être chiffré et stocké dans un compartiment S3 et téléchargé sur l'instance à l'aide de UserData.
- La création de politiques IAM à l'aide de CloudFormation modèles est soumise à des contraintes : les politiques IAM doivent être examinées et approuvées par AMS. SecOps À l'heure actuelle, nous prenons uniquement en charge le déploiement de rôles IAM avec des politiques en ligne contenant des autorisations préapprouvées. Dans d'autres cas, les politiques IAM ne peuvent pas être créées à l'aide de CloudFormation modèles, car cela remplacerait le processus AMS SecOps .
- Le protocole SSH KeyPairs n'est pas pris en charge : EC2 les instances Amazon doivent être accessibles via le système de gestion des accès AMS. Le processus AMS RFC vous authentifie. Vous ne pouvez pas inclure de paires de clés SSH dans les CloudFormation modèles car vous n'êtes pas autorisé à créer des paires de clés SSH et à remplacer le modèle de gestion des accès AMS.
- Les règles d'entrée des groupes de sécurité sont restreintes : vous ne pouvez pas avoir une plage d'adresses CIDR source comprise entre 0.0.0.0/0, ou un espace d'adressage routable publiquement, avec un port TCP autre que 80 ou 443.
- Suivez les CloudFormation directives lors de la rédaction de modèles de CloudFormation ressources : assurez-vous d'utiliser le bon type/property nom de données pour la ressource en vous référant au guide de AWS CloudFormation l'utilisateur de cette ressource. Par exemple, le type de données d'une SecurityGroupIds propriété dans une AWS::EC2::Instance ressource est « Liste de valeurs de chaîne », donc ["sg-aaaaaaaa"] est correct (avec crochets), mais pas « sg-aaaaaaaaaa » (sans crochets).

  Pour plus d'informations, consultez le document de référence des types de ressources et de propriétés AWS.
- Configurez vos CloudFormation modèles personnalisés pour utiliser les paramètres définis dans l'AMS CloudFormation ingest CT — Lorsque vous configurez votre CloudFormation modèle pour utiliser les paramètres définis dans l'AMS CloudFormation ingest CT, vous pouvez réutiliser le CloudFormation modèle pour créer des piles similaires en le soumettant avec les valeurs de paramètres modifiées dans l'entrée CT avec Management | Custom stack | Stack from CloudFormation template | Update CT (ct-361tlo1k7339x). Pour obtenir un exemple, consultez CloudFormation Exemples d'ingestion : définition des ressources.

- Les points de terminaison de compartiment Amazon S3 dotés d'une URL présignée ne peuvent pas être expirés — Si vous utilisez un point de terminaison de compartiment Amazon S3 avec une URL présignée, vérifiez que l'URL Amazon S3 présignée n'est pas expirée. Une RFC d' CloudFormation ingestion soumise avec une URL de compartiment Amazon S3 présignée expirée est rejetée.
- La condition d'attente nécessite une logique de signal : la condition d'attente est utilisée pour coordonner la création de ressources de pile avec des actions de configuration externes à la création de la pile. Si vous utilisez la ressource CloudFormation Wait Condition dans le modèle, attendez un signal de réussite et celle-ci marque la création de la pile comme un échec si le nombre de signaux de réussite n'est pas émis. Vous devez disposer d'une logique pour le signal si vous utilisez la ressource Wait Condition. Pour plus d'informations, voir Création de conditions d'attente dans un modèle.

Bonnes pratiques

Voici quelques bonnes pratiques que vous pouvez utiliser pour migrer des ressources à l'aide du processus CloudFormation d'ingestion AMS :

- Soumettez des ressources IAM et d'autres ressources liées aux politiques en un seul CT — Si vous pouvez utiliser des outils automatisés CTs tels que CloudFormation Ingest pour déployer des rôles IAM, nous vous recommandons de le faire. Dans d'autres cas, AMS vous recommande de rassembler toutes les ressources IAM ou autres ressources liées aux politiques et de les soumettre dans un seul type Management | Other | Other | Create change (ct-1e1xtak34nx76). Par exemple, combinez tous les rôles IAM nécessaires, les profils d' EC2 instance IAM Amazon, les mises à jour des politiques IAM pour les rôles IAM existants, les politiques de compartiment Amazon S3, les politiques Amazon SNS/Amazon SQS, etc., et soumettez une RFC ct-1e1xtak34nx76 afin que ces ressources préexistantes puissent simplement être référencées dans les futurs modèles d'ingestion. CloudFormation
- EC2 les instances sont démarrées et jointes avec succès au domaine. Cela se fait automatiquement, conformément à la meilleure pratique. Pour s'assurer que les EC2 instances Amazon lancées via une pile d' CloudFormation ingestion sont démarrées et rejoignent le domaine avec succès, AMS inclut une CreationPolicy et une UpdatePolicy pour une ressource de groupe Auto Scaling (c'est-à-dire, si ces politiques n'existent pas déjà).
- Le paramètre de l'instance de base de données Amazon RDS doit être spécifié — Lorsque vous créez une base de données Amazon RDS par CloudFormation ingestion, vous devez spécifier le `DBSnapshotIdentifier` paramètre afin de procéder à une restauration à partir d'un instantané

de base de données précédent. Cela est nécessaire car l' CloudFormation ingestion ne gère actuellement pas les données sensibles.

Pour un exemple d'utilisation d'un CloudFormation modèle pour l'ingestion de CloudFormation modèles AMS, consultezCloudFormation Ingestion : exemples.

Validation du modèle

Vous pouvez valider vous-même votre CloudFormation modèle avant de le soumettre à AMS.

Les modèles soumis à AMS CloudFormation ingest sont validés pour garantir qu'ils peuvent être déployés en toute sécurité au sein d'un compte AMS. Le processus de validation vérifie les points suivants :

- Ressources prises en charge : seules les ressources prises en CloudFormation charge par AMS ingest sont utilisées. Pour de plus amples informations, veuillez consulter Ressources prises en charge.
- Pris en charge AMIs : l'AMI figurant dans le modèle est une AMI compatible avec AMS. Pour plus d'informations sur AMS AMIs, consultezImages de machines AMS Amazon (AMIs).
- Sous-réseau AMS Shared Services : le modèle ne tente pas de lancer des ressources dans le sous-réseau AMS Shared Services.
- Politiques relatives aux ressources : il n'existe aucune politique de ressources trop permissive, telle qu'une politique de compartiment S3 lisible ou inscriptible par le public. AMS n'autorise pas l'entrée de compartiments S3 lisibles ou inscriptibles par le public. Comptes AWS

Valider avec CloudFormation Linter

Vous pouvez valider vous-même votre CloudFormation modèle avant de le soumettre à AMS à l'aide de l'outil CloudFormation Linter.

L'outil CloudFormation Linter est le meilleur moyen de valider votre CloudFormation modèle car il permet de valider les resource/property noms, les types de données et les fonctions. Pour plus d'informations, consultez cfn-python-lintaws-cloudformation/.

La sortie CloudFormation Linter du modèle présenté précédemment est la suivante :

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

```
./testtmpl.json:6:9
```

Pour faciliter la validation hors ligne des CloudFormation modèles, AMS a développé un ensemble de règles de validation personnalisées enfichables pour l'outil CloudFormation Linter. Ils se trouvent sur la page Ressources pour développeurs de la console AMS.

Pour utiliser les scripts de validation CloudFormation avant ingestion, procédez comme suit :

1. Installez l'outil CloudFormation Linter. Pour les instructions d'installation, consultez aws-cloudformation/cfn-lint.
2. Téléchargez un fichier .zip contenant des scripts de validation :

   Règles personnalisées de CFN Lint.
3. Décompressez les règles jointes dans le répertoire de votre choix.
4. Validez votre CloudFormation modèle en exécutant la commande suivante :

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

CloudFormation ingest stack : exemples de validateurs CFN

Ces exemples peuvent vous aider à préparer votre modèle en vue d'une ingestion réussie.

Validation du format

Vérifiez que le modèle contient une section « Ressources » et que toutes les ressources définies sous celui-ci ont une valeur « Type ».

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

Vérifiez que les clés racines du modèle sont autorisées. Les clés root autorisées sont les suivantes :

```
[
```

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

Validation manuelle de l'automatisation gérée

Si le modèle contient les ressources suivantes, la validation automatique échoue et vous devez procéder à une révision manuelle.

Les politiques présentées sont des zones à haut risque du point de vue de la sécurité. Par exemple, une politique de compartiment S3 permettant à quiconque, à l'exception d'utilisateurs ou de groupes spécifiques, de créer des objets ou d'écrire des autorisations est extrêmement dangereuse. Nous validons donc les politiques et les approuvons ou les refusons en fonction du contenu, et ces politiques ne peuvent pas être créées automatiquement. Nous étudions les approches possibles pour résoudre ce problème.

Nous n'avons actuellement pas de validation automatique pour les ressources suivantes.

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

Validation de paramètres

Vérifiez que si aucune valeur n'est fournie pour un paramètre de modèle, il doit avoir une valeur par défaut.

Validation des attributs de ressources

Vérification des attributs requise : certains attributs doivent exister pour certains types de ressources.

- « VPCOptions » doit exister dans `AWS::OpenSearch::Domain`

- « CludsterSubnetGroupName » doit exister dans `AWS::Redshift::Cluster`

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

Vérification des attributs non autorisés : certains attributs ne doivent *pas* exister pour certains types de ressources.

- «SecretString" ne doit pas exister dans "AWS::SecretsManager::Secret»
- «MongoDbSettings" ne doit pas exister dans "AWS::DMS::Endpoint»

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

Vérification des paramètres SSM : pour les attributs de la liste suivante, les valeurs doivent être spécifiées via Secrets Manager ou Systems Manager Parameter Store (Secure String Parameter) :

```
{
  "RDS::DBInstance": [
    "MasterUserPassword",
    "TdeCredentialPassword"
  ],
  "RDS::DBCluster": [
    "MasterUserPassword"
  ],
  "ElastiCache::ReplicationGroup": [
    "AuthToken"
  ],
```

```
  "DMS::Certificate": [
    "CertificatePem",
    "CertificateWallet"
  ],
  "DMS::Endpoint": [
    "Password"
  ],
  "CodePipeline::Webhook": {
    "AuthenticationConfiguration": [
        "SecretToken"
    ]
  },
  "DocDB::DBCluster": [
    "MasterUserPassword"
  ]
},
```

Certains attributs doivent respecter certains modèles ; par exemple, les noms de profil d'instance IAM ne doivent pas commencer par des préfixes réservés AMS, et la valeur de l'attribut doit correspondre à l'expression régulière spécifique, comme indiqué :

```
{
    "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).+"
      ]
    }
}
```

Validation des ressources

Seules les ressources autorisées peuvent être spécifiées dans le modèle ; ces ressources sont décrites dansRessources prises en charge.

Les piles EC2 et les groupes Auto Scaling (ASGs) ne sont pas autorisés dans la même pile en raison des limites liées aux correctifs.

Validation des règles d'entrée des groupes de sécurité

- Pour les demandes provenant des types de modification CFN Ingest Create ou Stack Update CT :
  - Si (`IpProtocol`est TCP ou 6) ET (le port est 80 ou 443), il n'y a aucune restriction quant à la valeur `CidrIP`
  - Sinon, le `CidrIP` ne peut pas être 0.0.0.0/0
- Pour les demandes provenant de Service Catalog (produits Service Catalog) :
  - Outre la validation du type de changement de type CFN Ingest Create ou Stack Update CT, le port `management_ports` contenant le protocole est uniquement `ip_protocols` accessible via : `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"]
  }
```

Limites

Les caractéristiques et fonctionnalités suivantes ne sont actuellement pas prises en charge par le processus CloudFormation d'ingestion AMS.

- YAML — Non pris en charge. Seuls les CloudFormation modèles JSON sont pris en charge.
- Stacks imbriqués : concevez plutôt votre infrastructure d'applications de manière à utiliser un modèle unique. Vous pouvez également utiliser le référencement entre piles pour séparer les ressources entre plusieurs piles lorsqu'une ressource dépend d'une autre. Pour plus d'informations, consultez Procédure pas à pas : reportez-vous aux sorties de ressources dans une autre CloudFormation pile AWS.
- CloudFormation ensembles de piles : non pris en charge pour des raisons de sécurité.
- Création de ressources IAM à l'aide CloudFormation de modèles : seuls les rôles IAM sont pris en charge, pour des raisons de sécurité.
- Données sensibles : non prises en charge. N'incluez pas de données sensibles dans le modèle ou dans les valeurs des paramètres. Si vous devez référencer des données sensibles, utilisez Secrets Manager pour stocker et récupérer ces valeurs. Pour plus d'informations sur l'utilisation des secrets d'AWS Secrets Manager dans une propriété de ressource, consultez Comment créer et récupérer des secrets gérés dans AWS Secrets Manager à l'aide de CloudFormation modèles AWS et Utilisation de références dynamiques pour spécifier les valeurs des modèles.

Ressources prises en charge

Les ressources AWS suivantes sont prises en charge dans le cadre du processus CloudFormation d'ingestion d'AMS.

CloudFormation Ingest Stack : ressources prises en charge

Le système d'exploitation de l'instance doit être compatible avec l'ingestion de charge de travail AMS. Seules les ressources AWS répertoriées ici sont prises en charge.

- Amazon API Gateway
  - AWS::ApiGateway::Account
  - AWS::ApiGateway::ApiKey
  - AWS::ApiGateway::Authorizer
  - AWS::ApiGateway::BasePathCartographie
  - 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::UsagePlanClé
- 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
- Amazon CloudWatch Logs
  - 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::UserPoolClient
  - AWS::Cognito::UserPoolDomaine

  - AWS::Cognito::UserPoolGroupe

  - AWS::Cognito::UserPoolIdentityProvider
  - AWS::Cognito::UserPoolResourceServer
  - AWS::Cognito::UserPoolRiskConfigurationAttachment
  - AWS::Cognito::UserPoolUICustomizationPièce jointe
  - AWS::Cognito::UserPoolUser
  - AWS::Cognito::UserPoolUserToGroupAttachment
- Amazon DocumentDB
  - AWS::DocBase de données : DBCluster
  - AWS::DocBase de données : DBCluster ParameterGroup
  - AWS::DocBase de données : DBInstance
  - AWS::DocDB : : DBSubnet Groupe
- 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::NetworkInterfacePièce jointe
  - AWS::EC2::SecurityGroup
  - AWS::EC2::SecurityGroupEntrée
  - AWS::EC2::SecurityGroupSortie
  - 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::PrimaryTaskSet
  - 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::SecurityGroupEntrée
  - AWS::ElastiCache::SubnetGroup
- Amazon EventBridge
  - AWS::Events::EventBus
  - AWS::Events::EventBusPolitique
  - 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
  - Groupe AWS : :RDS : : DBParameter
  - Groupe AWS : :RDS : : DBSubnet
  - AWS::RDS::EventSubscription
  - AWS::RDS::OptionGroup
- Amazon Route 53
  - AWS::Route53::HealthCheck
  - AWS::Route53::HostedZone
  - AWS::Route53::RecordSet
  - AWS::Route53::RecordSetGroupe
  - AWS::Route53Resolver::ResolverRule
  - AWS::Route53Resolver::ResolverRuleAssociation
- 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::ReceiptRuleSet
  - 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
- Application AutoScaling
  - AWS::ApplicationAutoScaling::ScalableTarget
  - AWS::ApplicationAutoScaling::ScalingPolicy
- Amazon EC2 AutoScaling
  - AWS::AutoScaling::AutoScalingGroupe

  - 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::WaitConditionPoignée
- 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::CustomActionType
  - AWS::CodePipeline::Pipeline
  - AWS::CodePipeline::Webhook
- Service de migration de base de données AWS (DMS)
  - AWS::DMS::Certificate
  - AWS::DMS::Endpoint
  - AWS::DMS::EventSubscription
  - AWS::DMS::ReplicationInstance

  - AWS::DMS::ReplicationSubnetGroupe

  - AWS::DMS::ReplicationTask

    La MongoDbSettings propriété dans la AWS::DMS::Endpoint ressource n'est pas autorisée.

    Les propriétés suivantes ne sont autorisées que si elles sont résolues par AWS Secrets Manager : CertificatePem les CertificateWallet propriétés de la AWS::DMS::Certificate ressource et la propriété Password de la AWS::DMS::Endpoint ressource.
- AWS Elastic Load Balancing - Application 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
- Gestion des identités et des accès AWS (JE SUIS)
  - AWS::IAM::Role
- Streaming géré par AWS pour 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::DataLakeRéglages
  - AWS::LakeFormation::Permissions
  - AWS::LakeFormation::Resource
- AWS Lambda
  - AWS::Lambda::Alias
  - AWS::Lambda::EventInvokeConfig
  - AWS::Lambda::EventSourceCartographie
  - AWS::Lambda::Function
  - AWS::Lambda::LayerVersion
  - AWS::Lambda::LayerVersionAutorisation
  - AWS::Lambda::Permission
  - AWS::Lambda::Version
- Amazon Redshift
  - AWS::Redshift::Cluster
  - AWS::Redshift::ClusterParameterGroupe
  - AWS::Redshift::ClusterSubnetGroupe
- AWS Secrets Manager
  - AWS::SecretsManager::ResourcePolicy

  - AWS::SecretsManager::RotationSchedule

- AWS::SecretsManager::Secret
- AWS::SecretsManager::SecretTargetPièce jointe

- 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::SizeConstraintSet
  - AWS::WAF::SqlInjectionMatchSet
  - AWS::WAF::WebACL
  - AWS::WAF::XssMatchSet
- AWS WAF Régional
  - AWS::WAFRegional::ByteMatchSet
  - AWS::WAFRegional::GeoMatchSet
  - AWS : WAFRegional : : IPSet
  - AWS::WAFRegional::RateBasedRègle
  - AWS::WAFRegional::RegexPatternSet
  - AWS::WAFRegional::Rule

- AWS::WAFRegional::SizeConstraintSet

  - AWS::WAFRegional::SqlInjectionMatchSet
  - AWS::WAFRegional::WebACL
  - AWS::WAFRegional::WebACLAssociation
  - AWS::WAFRegional::XssMatchSet
- AWS WAFv2
  - AWS : WAFv2 : : IPSet
  - AWS::WAFv2::RegexPatternSet
  - AWS::WAFv2::RuleGroup
  - AWS::WAFv2::WebACL
  - AWS::WAFv2::WebACLAssociation

CloudFormation Ingestion : exemples

Vous trouverez ici des exemples détaillés d'utilisation du type Create stack with CloudFormation template change.

Pour télécharger un ensemble d'exemples de CloudFormation modèles par modèle Région AWS, consultez la section Exemples de modèles.

Pour obtenir des informations de référence sur les CloudFormation ressources, consultez le document de référence des types de ressources et de propriétés AWS. Cependant, AMS prend en charge un ensemble de ressources plus restreint, qui est décrit dansCloudFormation Ingestion d'AMS.

> **Note**
>
> AMS vous conseille de rassembler toutes les ressources IAM ou autres ressources liées aux politiques et de les soumettre dans un seul type de gestion | Autre | Autre | Créer un changement (ct-1e1xtak34nx76). Par exemple, combinez tous les rôles IAM, les profils d'instance IAM, les mises à jour des politiques IAM pour les rôles IAM existants, les politiques de compartiment S3, les SNS/SQS politiques, etc., puis soumettez une RFC ct-1e1xtak34nx76 afin que ces ressources préexistantes puissent être référencées dans les futurs modèles CFN Ingest.

Rubriques

CloudFormation Exemples d'ingestion : définition des ressources

Lorsque vous utilisez AMS CloudFormation ingest, vous personnalisez un CloudFormation modèle et vous le soumettez à AMS dans une RFC avec le type de modification d'ingestion (CloudFormation ct-36cn2avfrrj9v). Pour créer un CloudFormation modèle qui peut être réutilisé plusieurs fois, vous ajoutez les paramètres de configuration de la pile à l'entrée d'exécution du type de modification d' CloudFormation ingestion plutôt que de les coder en dur dans le CloudFormation modèle. Le principal avantage est que vous pouvez réutiliser le modèle.

Le schéma de saisie du type de modification AMS CloudFormation ingest vous permet de choisir jusqu'à soixante paramètres dans un CloudFormation modèle et de fournir des valeurs personnalisées.

Cet exemple montre comment définir une propriété de ressource, qui peut être utilisée dans divers CloudFormation modèles, en tant que paramètre dans le CT d' CloudFormation ingestion AMS. Les exemples de cette section montrent spécifiquement l'utilisation des rubriques SNS.

Rubriques

Exemple 1 : coder en dur la **`TopicName`** propriété de la CloudFormation SNSTopic ressource

Dans cet exemple, vous devez coder en dur la `TopicName` propriété de la CloudFormation SNSTopic ressource dans le CloudFormation modèle. Notez que la `Parameters` section est vide.

Pour disposer d'un CloudFormation modèle qui vous permet de modifier la valeur du SNSTopic nom d'une nouvelle pile sans avoir à créer un nouveau CloudFormation modèle, vous pouvez utiliser la

`Parameters` section AMS du type de modification d' CloudFormation ingestion pour effectuer cette configuration. Ce faisant, vous utiliserez le même CloudFormation modèle ultérieurement pour créer une nouvelle pile portant un `SNSTopic` nom différent.

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

Exemple 2 : Utiliser une SNSTopic ressource pour référencer un paramètre dans le type de modification AMS

Dans cet exemple, vous utilisez une `TopicName` propriété de `SNSTopic` ressource définie dans le CloudFormation modèle pour référencer a `Parameter` dans le type de modification AMS.

```
{
  "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"}
      }
```

```
    }
  }
}
```

Exemple 3 : créer une rubrique SNS en soumettant un fichier de paramètres d'exécution JSON avec le type de modification AMS ingest

Dans cet exemple, vous soumettez un fichier de paramètres d'exécution JSON avec le CT d'ingestion AMS qui crée le sujet SNS. `TopicName` Le sujet SNS doit être défini dans le CloudFormation modèle de la manière modifiable illustrée dans cet exemple.

```
{
  "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
}
```

Exemple 4 : Soumettre un nouveau type de modification qui fait référence au même CloudFormation modèle

Cet exemple JSON modifie la `TopicName` valeur SNS sans modifier le CloudFormation modèle. Au lieu de cela, vous soumettez un nouveau type de changement Deployment | Ingestion | Stack from CloudFormation Template | Create qui fait référence au même modèle CFN.

```
{
  "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
}
```

Exemple 5 : utilisation des valeurs de paramètres par défaut dans le CloudFormation modèle

Dans cet exemple, le SNS `TopicName` = « MyTopicName » est créé car aucune `TopicName` valeur n'a été fournie dans le paramètre `Parameters` d'exécution. Si vous ne fournissez pas de `Parameters` définitions, les valeurs des paramètres par défaut du CloudFormation modèle sont utilisées.

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

CloudFormation Exemples d'ingestion : application Web à 3 niveaux

Ingérez un CloudFormation modèle pour une application Web standard à 3 niveaux.

Cela inclut un Application Load Balancer, un groupe cible Application Load Balancer, un groupe Auto Scaling, un modèle de lancement de groupe Auto Scaling, Amazon Relational Database Service (RDS pour SQL Server) avec une base de données MySQL, un magasin de paramètres SSM et AWS Secrets Manager. AWS Prévoyez 30 à 60 minutes pour suivre cet exemple.

Conditions préalables

- Créez un secret contenant un nom d'utilisateur et un mot de passe avec les valeurs correspondantes à l'aide du AWS Secrets Manager. Vous pouvez vous référer à cet exemple de modèle JSON (fichier zip) qui contient le nom du `ams-shared/myapp/dev/dbsecrets` secret et le remplacer par votre nom secret. Pour plus d'informations sur l'utilisation de AWS Secrets Manager avec AMS, consultezUtilisation de AWS Secrets Manager avec les ressources AMS.

- Configurez les paramètres requis dans le magasin de paramètres AWS SSM (PS). Dans cet exemple, les `VPCId` sous-réseaux privé et public sont stockés dans le SSM PS dans des chemins tels que`/app/DemoApp/PublicSubnet1a`, `PublicSubnet1cPrivateSubnet1a`, `PrivateSubnet1c` et. `Subnet-Id` `VPCCidr` Mettez à jour les chemins, les noms et les valeurs des paramètres en fonction de vos besoins.
- Créez un rôle d'instance IAM Amazon EC2 avec des autorisations de lecture sur les chemins Secrets AWS Manager et SSM Parameter Store (le rôle IAM créé et utilisé dans ces exemples est). `customer-ec2_secrets_manager_instance_profile` Si vous créez des politiques conformes aux normes IAM, telles que le rôle de profil d'instance, le nom du rôle doit commencer par. `customer-` Pour créer un nouveau rôle IAM (vous pouvez le nommer ou autre chose)`customer-ec2_secrets_manager_instance_profile`, utilisez le code AMS Change Type Management | Applications | IAM instance profile | Create (ct-0ixp4ch2tiu04) CT et attachez les politiques requises. Vous pouvez consulter les politiques standard d'AMS IAM `customer_secrets_manager_policy` et`customer_systemsmanager_parameterstore_policy`, dans la console AWS IAM, pour les utiliser telles quelles ou comme référence.

Ingérer un CloudFormation modèle pour une application Web standard à 3 niveaux

1. Téléchargez l'exemple de modèle CloudFormation JSON ci-joint sous forme de fichier zip, 3- tier-cfn-ingest .zip dans un compartiment S3 et générez une URL S3 signée à utiliser dans la RFC CFN Ingest. Pour plus d'informations, consultez presign. Le modèle CFN peut également être copy/pasted intégré à la RFC CFN Ingest lorsque vous soumettez la RFC via la console AMS.
2. Créez une RFC d'ingestion (Deployment | CloudFormation Ingestion | Stack from CloudFormation template | Create (ct-36cn2avfrrj9v)), via la console AMS ou la CLI AMS. Le processus d'automatisation de l' CloudFormation ingestion valide le CloudFormation modèle pour s'assurer qu'il dispose de ressources valides prises en charge par AMS et qu'il est conforme aux normes de sécurité.
   - À l'aide de la console : pour le type de modification, sélectionnez Deployment -> Ingestion -> Stack from CloudFormation Template -> Create, puis ajoutez les paramètres suivants à titre d'exemple (notez que la valeur par défaut pour Multi AZDatabase est false) :

```
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"
```

- Utilisation du AWS CLI - Pour plus de détails sur la création RFCs à l'aide du AWS CLI, voir Création RFCs. Par exemple, exécutez la commande suivante :

```
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
```

Trouvez l'URL de l'Application Load Balancer dans la sortie d'exécution de la CloudFormation RFC pour accéder au site Web. Pour plus d'informations sur l'accès aux ressources, consultez la section Accès aux instances.

Création d'une CloudFormation pile d'ingestion

Création d'une pile d' CloudFormation ingestion à l'aide de la console

Pour créer une pile d' CloudFormation ingestion à l'aide de la console

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.

- Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.

3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Création d'une pile d' CloudFormation ingestion à l'aide de la CLI

Pour créer une pile d' CloudFormation ingestion à l'aide de la CLI

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

1. Préparez le CloudFormation modèle que vous utiliserez pour créer la pile et téléchargez-le dans votre compartiment S3. Pour obtenir des informations importantes, consultez les directives, CloudFormation les meilleures pratiques et les limites d'AWS Ingest.
2. Créez et soumettez le RFC à AMS :

   - Créez et enregistrez le fichier JSON des paramètres d'exécution, incluez les paramètres du CloudFormation modèle que vous souhaitez. L'exemple suivant le nomme CreateCfnParams .json.

     Exemple de fichier CreateCfnParams .json de pile d'applications Web :

     ```
     {
       "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"
   }
  ],
}
```

Exemple de CreateCfnParams fichier .json de rubrique SNS :

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

3. Créez et enregistrez le fichier JSON des paramètres RFC avec le contenu suivant. L'exemple suivant le nomme fichier CreateCfnRfc .json :

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

4. Créez la RFC en spécifiant le CreateCfnRfc fichier et le CreateCfnParams fichier :

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

Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

> **Note**
>
> Ce type de modification est en version 2.0 et est automatisé (il n'est pas exécuté manuellement). Cela permet à l'exécution du CT d'aller plus rapidement et, un nouveau paramètre CloudFormationTemplate, vous permet de coller un CloudFormation modèle personnalisé dans le RFC. De plus, dans cette version, nous n'associons pas les groupes de sécurité AMS par défaut si vous spécifiez vos propres groupes de sécurité. Si vous ne spécifiez pas vos propres groupes de sécurité dans la demande, AMS associera les groupes de sécurité AMS par défaut. Dans CFN Ingest v1.0, nous avons toujours ajouté les groupes de sécurité AMS par défaut, que vous ayez fourni ou non vos propres groupes de sécurité. AMS a activé 17 services AMS auto-provisionnés à utiliser dans ce type de modification. Pour plus d'informations sur les ressources prises en charge, voir CloudFormation Ingest Stack : Supported Resources.

> **Note**
>
> La version 2.0 accepte un point de terminaison S3 qui n'est pas une URL présignée.
> Si vous utilisez la version précédente de ce CT, la valeur du paramètre CloudFormationTemplateS3Endpoint doit être une URL présignée.
> Exemple de commande pour générer une URL de compartiment S3 présignée (Mac/Linux) :

```
export S3_PRESIGNED_URL=$(aws s3 presign DASHDASHexpires-in 86400
 s3://BUCKET_NAME/CFN_TEMPLATE.json)
```

> Exemple de commande pour générer une URL de compartiment S3 présignée (Windows) :

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

Consultez également la section Création de compartiments pré-signés URLs pour Amazon S3.

> **Note**
>
> Si le compartiment S3 existe dans un compte AMS, vous devez utiliser vos informations d'identification AMS pour cette commande. Par exemple, il se peut que vous deviez ajouter `--profile saml` après avoir obtenu vos informations d'identification AMS AWS Security Token Service (AWS STS).

Types de modifications connexes :Approuver un ensemble CloudFormation de modifications à la pile d'ingestion, Mettre à jour CloudFormation la pile d'ingestion

Pour en savoir plus sur AWS CloudFormation, consultez AWS CloudFormation. Pour voir les CloudFormation modèles, ouvrez le manuel AWS CloudFormation Template Reference.

Validation d'une ingestion CloudFormation

Le modèle est validé pour garantir qu'il peut être créé dans un compte AMS. S'il passe la validation, il est mis à jour pour inclure toutes les ressources ou configurations requises pour être conforme à AMS. Cela inclut l'ajout de ressources telles que les CloudWatch alarmes Amazon afin de permettre à AMS Operations de surveiller la pile.

La RFC est rejetée si l'une des conditions suivantes est vraie :

- La syntaxe RFC JSON est incorrecte ou ne suit pas le format indiqué.
- L'URL présignée du compartiment S3 fournie n'est pas valide.
- La CloudFormation syntaxe du modèle n'est pas valide.
- Le modèle n'a pas de valeurs par défaut définies pour toutes les valeurs de paramètres.
- Le modèle échoue à la validation AMS. Pour les étapes de validation AMS, consultez les informations plus loin dans cette rubrique.

La RFC échoue si la CloudFormation pile ne parvient pas à se créer en raison d'un problème de création de ressources.

Pour en savoir plus sur la validation et le validateur CFN, voir Validation de modèles et pile d'CloudFormation ingestion : exemples de validateurs CFN.

Mettre à jour CloudFormation la pile d'ingestion

Mettre à jour une pile CloudFormation d'ingestion à l'aide de la console

Pour mettre à jour un CloudFormation Ingest Stack à l'aide de la console

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.

5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Mise à jour d'une pile CloudFormation d'ingestion à l'aide de la CLI

Pour mettre à jour une pile d' CloudFormation ingestion à l'aide de la CLI

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

1. Préparez le CloudFormation modèle que vous souhaitez utiliser pour mettre à jour la pile, puis chargez-le dans votre compartiment S3. Pour obtenir des informations importantes, consultez les directives, CloudFormation les meilleures pratiques et les limites d'AWS Ingest.
2. Créez et soumettez le RFC à AMS :

- Créez et enregistrez le fichier JSON des paramètres d'exécution, incluez les paramètres du CloudFormation modèle que vous souhaitez. Cet exemple le nomme UpdateCfnParams .json.

  Exemple de fichier UpdateCfnParams .json avec mises à jour des paramètres en ligne :

```
{
  "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
}
```

  Exemple de fichier UpdateCfnParams .json avec point de terminaison du compartiment S3 contenant un CloudFormation modèle mis à jour :

```
{
  "StackId": "stack-yjjoo9aicjyqw4ro2",
  "VpcId": "VPC_ID",
  "CloudFormationTemplateS3Endpoint": "s3_url",
  "TemplateParameters": [
    {
      "Key": "TopicName",
      "Value": "TopicNameCLI"
    },
    {
      "Key": "DisplayName",
```

```
      "Value": "DisplayNameCLI"
    }
  ],
  "TimeoutInMinutes": 1080
}
```

3. Créez et enregistrez le fichier JSON des paramètres RFC avec le contenu suivant. Cet exemple le nomme fichier UpdateCfnRfc .json.

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

4. Créez la RFC en spécifiant le UpdateCfnRfc fichier et le UpdateCfnParams fichier :

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

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

- Ce type de modification est désormais disponible en version 2.0. Les modifications incluent la suppression du AutoApproveUpdateForResourcesparamètre, qui était utilisé dans la version 1.0 de ce CT, et l'ajout de deux nouveaux paramètres : AutoApproveRiskyUpdateset BypassDriftCheck.
- Si le compartiment S3 existe dans un compte AMS, vous devez utiliser vos informations d'identification AMS pour cette commande. Par exemple, il se peut que vous deviez ajouter `--profile saml` après avoir obtenu vos informations d'identification AMS AWS Security Token Service (AWS STS).
- Toutes les `Parameter` valeurs des ressources du CloudFormation modèle doivent avoir une valeur, soit via une valeur par défaut, soit une valeur personnalisée via la section des paramètres du CT. Vous pouvez remplacer la valeur du paramètre en structurant les ressources du CloudFormation modèle pour faire référence à une clé de paramètres. Pour des exemples montrant comment procéder, voir CloudFormation ingest stack : CFN validator examples.

IMPORTANT : Les paramètres manquants ne sont pas fournis explicitement dans le formulaire. Les valeurs par défaut sont celles actuellement définies sur la pile ou le modèle existant.

- Pour obtenir la liste des services auto-approvisionnés que vous pouvez ajouter à l'aide d'Ingest, voir CloudFormation CloudFormation Ingest Stack : Supported Resources.

  Pour en savoir plus CloudFormation, consultez AWS CloudFormation.

Validation d'une ingestion CloudFormation

Le modèle est validé pour garantir qu'il peut être créé dans un compte AMS. S'il passe la validation, il est mis à jour pour inclure toutes les ressources ou configurations requises pour être conforme à AMS. Cela inclut l'ajout de ressources telles que les CloudWatch alarmes Amazon afin de permettre à AMS Operations de surveiller la pile.

La RFC est rejetée si l'une des conditions suivantes est vraie :

- La syntaxe RFC JSON est incorrecte ou ne suit pas le format indiqué.
- L'URL présignée du compartiment S3 fournie n'est pas valide.
- La CloudFormation syntaxe du modèle n'est pas valide.
- Le modèle n'a pas de valeurs par défaut définies pour toutes les valeurs de paramètres.
- Le modèle échoue à la validation AMS. Pour les étapes de validation AMS, consultez les informations plus loin dans cette rubrique.

La RFC échoue si la CloudFormation pile ne parvient pas à se créer en raison d'un problème de création de ressources.

Pour en savoir plus sur la validation et le validateur CFN, voir Validation de modèles et pile d'CloudFormation ingestion : exemples de validateurs CFN.

Approuver un ensemble CloudFormation de modifications à la pile d'ingestion

Approuver et mettre à jour une pile d' CloudFormation ingestion à l'aide de la console

Pour approuver et mettre à jour une pile d' CloudFormation ingestion à l'aide de la console

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Approbation et mise à jour d'une pile d' CloudFormation ingestion à l'aide de la CLI

Pour approuver et mettre à jour une pile d' CloudFormation ingestion à l'aide de la CLI

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les

paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.

2. Soumettez la `aws amscm submit-rfc --rfc-id` *ID* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *ID* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

1. Exportez le schéma JSON des paramètres d'exécution pour ce type de modification dans un fichier de votre dossier actuel. Cet exemple le nomme CreateAsgParams .json :

   ```
   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. Modifiez et enregistrez le schéma comme suit :

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

Conseils

> **Note**
>
> Si une pile contient plusieurs ressources et que vous souhaitez supprimer uniquement un sous-ensemble des ressources de la pile, utilisez le CT de CloudFormation mise à jour ; voir CloudFormation Ingestion de la pile : mise à jour. Vous pouvez également soumettre un dossier de demande de service et les ingénieurs d'AMS peuvent vous aider à élaborer l'ensemble des modifications, si nécessaire.

Pour en savoir plus AWS CloudFormation, consultez AWS CloudFormation.

Protection contre les mises à jour, CloudFormation piles et terminaisons

Mise à jour d'une CloudFormation pile de protection contre les interruptions de service avec la console

Ce qui suit montre ce type de modification dans la console AMS.

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.

3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Mise à jour d'une protection contre la terminaison d'une CloudFormation pile avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification`

`"{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

Spécifiez uniquement les paramètres que vous souhaitez modifier. Les paramètres absents conservent les valeurs existantes.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"]}}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution de ce type de modification dans un fichier JSON ; cet exemple le nomme EnableTermPro CFNParams .json :

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

2. Modifiez et enregistrez le EnableTermPro CFNParams fichier en ne conservant que les paramètres que vous souhaitez modifier. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
     "DocumentName": "AWSManagedServices-ManageResourceTerminationProtection",
     "Region": "us-east-1",
   ```

```
  "Parameters": {
    "ResourceId": ["stack-psvnq6cupymio3enl"],
    "TerminationProtectionDesiredState": ["enabled"]
  }
}
```

3. Exportez le modèle RFC dans un fichier de votre dossier actuel ; cet exemple le nomme EnableTermPro CFNRfc .json :

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

4. Modifiez et enregistrez le fichier EnableTermPro CFNRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez la RFC en spécifiant le EnableTermPro CFNRfc fichier et le EnableTermPro CFNParams fichier :

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

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

> **Note**
>
> Il existe un CT connexe pour Amazon EC2, EC2 stack : Updating termination protection.

Pour en savoir plus sur la protection contre le licenciement, consultez la section Protection d'une pile contre la suppression.

Déploiements IAM automatisés à l'aide de CFN ingest ou de stack update dans AMS CTs

Vous pouvez utiliser ces types de modification AMS pour déployer des rôles IAM (la `AWS::IAM::Role` ressource) à la fois dans une zone d'atterrissage multi-comptes (MALZ) et une zone d'atterrissage à compte unique (SALZ) :

- Déploiement | Ingestion | Stack à partir d' CloudFormation un modèle | Créer (ct-36cn2avfrrj9v)
- Gestion | Stack personnalisé | Stack à partir d'un CloudFormation modèle | Mise à jour (ct-361tlo1k7339x)
- Gestion | Stack personnalisé | Stack à partir d'un CloudFormation modèle | Approuver et mettre à jour (ct-1404e21baa2ox)

Validations effectuées sur les rôles IAM dans votre modèle CFN :

- ManagedPolicyArns: L'attribut ne ManagedPolicyArnsdoit pas exister dans`AWS::IAM::Role`. La validation interdit d'associer des politiques gérées au rôle en cours de provisionnement. Au lieu de cela, les autorisations associées au rôle peuvent être gérées à l'aide de la politique intégrée via la propriété Policies.
- PermissionsBoundary: La politique utilisée pour définir la limite des autorisations pour le rôle ne peut être que la politique gérée par AMS vended :`AWSManagedServices_IAM_PermissionsBoundary`. Cette politique agit comme un garde-fou qui protège les ressources de l'infrastructure AMS contre toute modification à l'aide du rôle fourni. Avec cette limite d'autorisation par défaut, les avantages de sécurité fournis par AMS sont préservés.

  Le `AWSManagedServices_IAM_PermissionsBoundary` (par défaut) est obligatoire, sans lui, la demande est rejetée.
- MaxSessionDuration: La durée maximale de session pouvant être définie pour le rôle IAM est de 1 à 4 heures. La norme technique AMS exige une acceptation des risques par le client pour une durée de session supérieure à 4 heures.
- RoleName: Les espaces de noms suivants sont préservés par AMS et ne peuvent pas être utilisés comme préfixes de nom de rôle IAM :

```
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
```

- Politiques : La politique intégrée au rôle IAM ne peut inclure qu'un ensemble d'actions IAM préapprouvées par AMS. Il s'agit de la limite supérieure de toutes les actions IAM autorisées pour créer un rôle IAM (politique de contrôle). La politique de contrôle consiste à :
  - Toutes les actions de la politique AWS gérée ReadOnlyAccess qui fournit un accès en lecture seule à toutes les ressources Services AWS
  - Les actions suivantes, avec la restriction des actions S3 entre comptes, c'est-à-dire les actions S3 autorisées, ne peuvent être effectuées que sur les ressources présentes dans le même compte que le rôle créé :

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

```
sns:Publish
```

Tout rôle IAM créé ou mis à jour via CFN ingest peut autoriser les actions répertoriées dans cette politique de contrôle, ou les actions dont la portée est limitée (moins permissive que) aux actions répertoriées dans la politique de contrôle. Actuellement, nous autorisons ces actions IAM sécurisées qui peuvent être classées dans la catégorie des actions en lecture seule, ainsi que les actions non en lecture seule mentionnées ci-dessus qui ne peuvent pas être effectuées CTs et qui sont préapprouvées conformément à la norme technique AMS.

- AssumeRolePolicyDocument: Les entités suivantes sont préapprouvées et peuvent être incluses dans la politique de confiance pour assumer le rôle créé :
  - Toute entité IAM (rôle, utilisateur, utilisateur root, session à rôle assumé par STS) du même compte peut assumer le rôle.
  - Les personnes suivantes Services AWS peuvent assumer le rôle :

```
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
```

- Le fournisseur SAML du même compte peut assumer le rôle. Actuellement, le seul nom de fournisseur SAML pris en charge est`customer-saml`.

Si une ou plusieurs validations échouent, le RFC est rejeté. Voici un exemple de raison de rejet d'une RFC :

```
{"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"}
```

Si vous avez besoin d'aide en cas d'échec de la validation ou de l'exécution d'une RFC, utilisez la correspondance RFC pour contacter AMS. Pour obtenir des instructions, voir Correspondance RFC et pièce jointe (console). Pour toute autre question, envoyez une demande de service. Pour savoir comment faire, consultez la section Création d'une demande de service.

> **Note**
>
> Nous n'appliquons actuellement aucune bonne pratique IAM dans le cadre de nos validations IAM. Pour connaître les meilleures pratiques en matière d'IAM, consultez la section Meilleures pratiques de sécurité dans IAM.

Création de rôles IAM avec des actions plus permissives ou application des meilleures pratiques IAM

Créez vos entités IAM avec les types de modifications manuelles suivants :

- Déploiement | Composants de pile avancés | Identity and Access Management (IAM) | Création d'une entité ou d'une politique (ct-3dpd8mdd9jn1r)
- Gestion | Composants de pile avancés | Identity and Access Management (IAM) | Mettre à jour l'entité ou la politique (ct-27tuth19k52b4)

Nous vous recommandons de lire et de comprendre nos normes techniques avant de déposer ce manuel RFCs. Pour y accéder, voir Comment accéder aux normes techniques.

> **Note**
>
> Chaque rôle IAM créé directement avec ces types de modifications manuelles appartient à sa propre pile individuelle et ne réside pas dans la même pile où les autres ressources d'infrastructure sont créées via CFN Ingest CT.

Mise à jour des rôles IAM créés avec CFN ingest via des types de modification manuels lorsque les mises à jour ne peuvent pas être effectuées via des types de modification automatisés

Utilisez le type de modification Management | Advanced stack components | Identity and Access Management (IAM) | Update entity or policy (ct-27tuth19k52b4).

> **Important**
>
> Les mises à jour des rôles IAM par le biais du CT manuel ne sont pas reflétées dans les modèles de pile CFN et provoquent une dérive de la pile. Une fois que le rôle a été mis à jour par le biais d'une demande manuelle dans un état qui ne répond pas à nos validations, le rôle ne peut plus être mis à jour à l'aide du Stack Update CT (ct-361tlo1k7339x) tant qu'il n'est toujours pas conforme à nos validations. La mise à jour CT ne peut être utilisée que si le modèle de pile CFN est conforme à nos validations. Cependant, la pile peut toujours être mise à jour via le Stack Update CT (ct-361tlo1k7339x), tant que la ressource IAM non conforme à nos validations n'est pas mise à jour et que le modèle CFN passe nos validations.

Suppression de vos rôles IAM créés par ingestion AWS CloudFormation

Si vous souhaitez supprimer l'intégralité de la pile, utilisez le type de modification automatique Delete Stack suivant. Pour obtenir des instructions, voir Delete Stack :

- Changer l'identifiant du type : ct-0q0bic0ywqk6c
- Classification : Gestion | Piles standard | Empiler | Supprimer et gérer | Composants de pile avancés | Empiler | Supprimer

Si vous souhaitez supprimer un rôle IAM sans supprimer l'intégralité de la pile, vous pouvez supprimer le rôle IAM du CloudFormation modèle et utiliser le modèle mis à jour comme entrée pour le type de modification automatique de Stack Update :

- Changer l'ID du type : ct-361tlo1k7339x
- Classification : Gestion | Pile personnalisée | Pile à partir d' CloudFormation un modèle | Mise à jour

Pour obtenir des instructions, voir Mettre à jour AWS CloudFormation la pile d'ingestion.

CodeDeploy demandes

Vous pouvez utiliser AWS CodeDeploy pour créer des conteneurs d'applications que vous pouvez ensuite déployer via un groupe d' CodeDeploy applications. Pour plus d'informations CodeDeploy, consultez CodeDeploy la documentation AWS.

Travailler avec AWS CodeDeploy implique le processus suivant :

1. Créez une CodeDeploy application. L' CodeDeploy application est un nom ou un conteneur utilisé CodeDeploy pour garantir que la révision, la configuration de déploiement et le groupe de déploiement corrects sont référencés lors d'un déploiement.
2. Créez un groupe CodeDeploy de déploiement. Un groupe de CodeDeploy déploiement définit un ensemble d'instances individuelles ciblées pour un déploiement. AMS dispose d'un type de modification distinct pour les groupes de CodeDeploy déploiement pour EC2.
3. Déployez l' CodeDeploy application via le groupe CodeDeploy de déploiement.

CodeDeploy candidature

Créez ou déployez CodeDeploy des applications.

Création d'une CodeDeploy application

Création d'une CodeDeploy application avec la console

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler le RFC ou en créer une copie avec les options en haut de la page.

Création d'une CodeDeploy application avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les

paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.

2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}}"
```

CRÉATION DU MODÈLE :

1. Exportez le schéma JSON des paramètres d'exécution de l' CodeDeploy application CT dans un fichier de votre dossier actuel ; cet exemple le nomme Create CDApp Params.json :

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

2. Modifiez et enregistrez le fichier JSON comme suit. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

3. Exportez le modèle JSON CreateRfc pour un fichier de votre dossier actuel ; cet exemple le nomme Create CDApp RFC.json :

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

4. Modifiez et enregistrez le fichier JSON comme suit. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez le RFC en spécifiant le fichier Create CDApp Rfc et le fichier de paramètres d'exécution :

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

Vous recevez l'identifiant de la nouvelle RFC dans la réponse et pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

Pour plus d'informations sur AWS CodeDeploy, consultez Créer une application avec AWS CodeDeploy.

Déployer CodeDeploy l'application

Déploiement d'une CodeDeploy application avec la console

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.

5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler le RFC ou en créer une copie avec les options en haut de la page.

Déploiement d'une CodeDeploy application avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}}"
```

CRÉATION DU MODÈLE :

1. Produisez le schéma JSON des paramètres d'exécution pour le déploiement de CodeDeploy l'application CT ; cet exemple le nomme Deploy CDApp Params.json :

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

2. Modifiez le fichier JSON comme suit. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
   "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. Exportez le modèle JSON CreateRfc pour un fichier de votre dossier actuel ; cet exemple le nomme Deploy CDApp RFC.json :

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

4. Modifiez et enregistrez le fichier Deploy CDApp RFC.json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le fichier Deploy CDApp Rfc :

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

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

Pour plus d'informations, consultez la section Créer un déploiement avec CodeDeploy.

CodeDeploy groupes de déploiement

Créez des groupes CodeDeploy d'applications.

Création d'un groupe CodeDeploy de déploiement

Création d'un groupe de CodeDeploy déploiement avec la console

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.

- Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

  Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
- Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.

3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Création d'un groupe CodeDeploy de déploiement avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

Surveillez la `aws amscm get-rfc --rfc-id ID` commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}}"
```

CRÉATION D'UN MODÈLE :

1. Exportez le schéma JSON des paramètres d'exécution dans un fichier de votre dossier actuel ; cet exemple le nomme Create CDDep GroupParams .json :

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

2. Modifiez et enregistrez le fichier JSON. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"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. Exportez le modèle JSON CreateRfc pour un fichier de votre dossier actuel ; cet exemple le nomme Create CDDep GroupRfc .json :

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

4. Modifiez et enregistrez le fichier JSON. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez la RFC en spécifiant le fichier de création et le CDDep GroupRfc fichier de paramètres d'exécution :

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

Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

Pour plus d'informations sur les groupes de CodeDeploy déploiement AWS, consultez Créer un groupe de déploiement avec AWS CodeDeploy.

Création d'un groupe de CodeDeploy déploiement pour EC2

Création d'un groupe de CodeDeploy déploiement pour EC2 avec la console

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types

de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.

4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Création d'un groupe CodeDeploy de déploiement pour EC2 avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste

de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\":\"\"}}"
```

CRÉATION D'UN MODÈLE :

1. Exportez le schéma JSON des paramètres d'exécution dans un fichier ; cet exemple le nomme Create CDDep GroupEc 2Params.json :

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

2. Modifiez et enregistrez le fichier JSON. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
   "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. Exportez le modèle JSON CreateRfc pour un fichier de votre dossier actuel ; cet exemple le nomme Create CDDep GroupEc 2RFC.json :

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

4. Modifiez et enregistrez le fichier JSON. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez le RFC en spécifiant le fichier Create CDDep GroupEc 2Rfc et le fichier de paramètres d'exécution :

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

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

Pour plus d'informations sur les groupes de CodeDeploy déploiement AWS, consultez Créer un groupe de déploiement avec AWS CodeDeploy.

AWS Database Migration Service (AWS DMS)

AWS Database Migration Service (AWS DMS) vous aide à migrer des bases de données vers AMS facilement et en toute sécurité. Vous pouvez migrer vos données vers et depuis les bases de données commerciales et Open Source les plus répandues, telles qu'Oracle, MySQL et PostgreSQL.

Le service prend en charge les migrations homogènes telles qu'Oracle vers Oracle, ainsi que les migrations hétérogènes entre différentes plateformes de base de données, telles qu'Oracle vers PostgreSQL ou MySQL vers Oracle. AWS DMS est un AWS service ; l'AMS vous CTs aide à créer des AWS DMS ressources dans votre compte géré par AMS

Le graphique suivant décrit le flux de travail d'une migration de base de données.

Rubriques

AWS Database Migration Service (AWS DMS), avant de commencer

Lorsque vous planifiez une migration de base de données à l'aide de l'AMS AWS DMS, tenez compte des points suivants :

- Points de terminaison source et cible : vous devez savoir quelles informations et quelles tables de la base de données source doivent être migrées vers la base de données cible. AMS AWS DMS prend en charge la migration de schéma de base, y compris la création de tables et de clés primaires. Toutefois, AMS AWS DMS ne crée pas automatiquement d'index secondaires, de clés étrangères, de comptes, etc. dans la base de données cible. Voir Sources pour la migration des données et cibles pour la migration des données pour plus d'informations.
- Migration de schéma/code : AMS AWS DMS n'effectue pas de conversion de schéma ou de code. Vous pouvez utiliser des outils tels que Oracle SQL Developer, MySQL Workbench ou pgAdmin III pour convertir votre schéma. Si vous souhaitez convertir un schéma existant vers un autre moteur de base de données, vous pouvez utiliser l'outil AWS Schema Conversion Tool. Il peut créer un schéma cible et générer et créer un schéma entier : tables, index, vues etc. Vous pouvez également utiliser l'outil pour convertir PL/SQL TSQL en pgSQL et dans d'autres formats.
- Types de données non pris en charge : certains types de données sources doivent être convertis en types de données équivalents pour la base de données cible.

AWS DMS scénarios à envisager

Les scénarios suivants, documentés, peuvent vous aider à élaborer votre propre chemin de migration de base de données.

- Migrer les données d'un serveur MySQL sur site vers Amazon RDS MySQL : voir le billet de blog AWS Migrer des données MySQL sur site vers Amazon RDS (et vice versa)
- Migrer des données d'une base de données Oracle vers une base de données Amazon RDS Aurora PostgreSQL : voir le billet de blog AWS Présentation rapide de la migration d'une base de données Oracle vers une base de données Amazon Aurora PostgreSQL
- Migrer des données de RDS MySQL vers S3 : voir le billet de blog AWS Comment archiver des données depuis des bases de données relationnelles vers Amazon Glacier à l'aide d'AWS DMS

Pour migrer une base de données, vous devez effectuer les opérations suivantes :

- Planifiez la migration de votre base de données, notamment en configurant un groupe de sous-réseaux de réplication.
- Allouez une instance de réplication qui exécute tous les processus de migration.
- Spécifiez un point de terminaison de base de données source et cible.
- Créez une tâche ou un ensemble de tâches pour définir les tables et processus de réplication à utiliser.
- Créez l' AWS DMS IAM `dms-cloudwatch-logs-role` et les `dms-vpc-role` rôles. Si vous utilisez Amazon Redshift comme base de données cible, vous devez également créer et ajouter le rôle IAM à `dms-access-for-endpoint` votre compte AWS. Pour plus d'informations, consultez Création des rôles IAM à utiliser avec l'AWS CLI et l'API AWS DMS.

Ces procédures pas à pas fournissent un exemple d'utilisation de la console AMS ou de la CLI AMS pour créer un AWS Database Migration Service ()AWS DMS. Des commandes CLI permettant de créer l'instance de AWS DMS réplication, le groupe de sous-réseaux et la tâche, ainsi qu'un point de terminaison AWS DMS source et un point de terminaison cible sont fournies.

Pour en savoir plus sur AMS AWS DMS, consultez la page AWS Database Migration Servicepour obtenir des informations générales et AWS Database Migration Service FAQsdes réponses aux questions les plus fréquemment posées.

AWS DMS, données requises pour la configuration

Pour chacune des AWS DMS procédures pas à pas suivantes, certaines données communes sont nécessaires.

- `Description`: informations pertinentes sur la ressource, distinctes des autres `Description` options de paramètres.
- `VpcId`: Le VPC à utiliser. Vous pouvez le découvrir en exécutant le ListVpcSummaries fonctionnement de l'API SKMS (`list-vpc-summaries`dans la CLI) ou en consultant la VPCspage de la console AMS. Pour la référence de l'API AMS SKMS, consultez l'onglet Rapports de la console AWS Artifact.
- `Name`: nom de la pile ou du composant de pile ; il devient le nom de la pile.
- `TimeoutInMinutes`: Combien de minutes sont autorisées pour la création de la pile avant l'échec de la RFC. Ce paramètre ne retardera pas l'exécution de la RFC, mais vous devez prévoir suffisamment de temps (par exemple, ne pas spécifier`"5"`).
- `ChangeTypeId`,`ChangeTypeVersion`, et `StackTemplateId` : Ils sont obligatoires mais varient selon le scanner et leurs valeurs sont fournies dans chaque section pertinente, ci-dessous.

AWS DMS tâches de configuration

Configurez à l' AWS DMS aide des procédures pas à pas suivantes.

1 : groupe AWS DMS de sous-réseaux de réplication : Créer

Vous pouvez utiliser la console AMS ou API/CLI créer un groupe de sous-réseaux de AWS DMS réplication AMS.

Création d'un AWS DMS groupe de sous-réseaux de réplication

Création d'un groupe AWS DMS de sous-réseaux de réplication avec la console

> **Note**
>
> Ce CT échoue si le rôle `dms-vpc-role` IAM n'existe pas dans le compte.

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Création d'un groupe AWS DMS de sous-réseaux de réplication à l'aide de la CLI

> **Note**
>
> Ce CT échoue si le rôle `dms-vpc-role` IAM n'existe pas dans le compte.

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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
\"}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution de ce type de modification dans un fichier JSON ; cet exemple le nomme CreateDmsRsgParams .json :

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

2. Modifiez et enregistrez le fichier CreateDmsRsgParams .json des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

3. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme CreateDmsRsgRfc .json :

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

4. Modifiez et enregistrez le fichier CreateDmsRsgRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le CreateDmsRsgRfc fichier :

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

Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

- Ce CT échoue si le rôle `dms-vpc-role` IAM n'existe pas dans le compte.
- Vous pouvez ajouter jusqu'à 50 balises, mais pour cela, vous devez activer la vue Configuration supplémentaire.

  Pour plus d'informations sur les instances de réplication DMS et les groupes de sous-réseaux, consultez Configuration d'un réseau pour une instance de réplication.

2 : instance AWS DMS de réplication : créer

Vous pouvez utiliser la console AMS ou API/CLI créer une instance de AWS DMS réplication AMS.

Création d'une instance AWS DMS de réplication

Création d'une instance de AWS DMS réplication avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.

- Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.

3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Création d'une instance AWS DMS de réplication à l'aide de la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}"
```

Lors de la création de votre instance de réplication, vous pouvez spécifier les magasins de données source et cible. Les magasins de données source et cible peuvent se trouver sur une instance Amazon Elastic Compute Cloud (Amazon EC2), un compartiment AWS S3, une instance de base de données Amazon Relational Database Service (Amazon RDS) ou une base de données sur site.

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution de ce type de modification dans un fichier JSON ; cet exemple le nomme CreateDmsRiParams .json :

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

2. Modifiez et enregistrez le fichier CreateDmsRiParams .json des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"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. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme CreateDmsRiRfc .json :

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

4. Modifiez et enregistrez le fichier CreateDmsRiRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le CreateDmsRiRfc fichier :

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

Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

- Vous pouvez ajouter jusqu'à 50 balises, mais pour cela, vous devez activer la vue Configuration supplémentaire.
- Vous devez créer une instance de réplication sur une instance EC2 de votre VPC AMS dotée d'un stockage et d'une puissance de traitement suffisants pour effectuer les tâches que vous attribuez et migrez les données de votre base de données source vers la base de données cible. La taille de cette instance varie en fonction de la quantité de données que vous devez migrer et des tâches que vous souhaitez que l'instance effectue. L'instance de réplication fournit une haute disponibilité et une prise en charge du basculement à l'aide d'un déploiement multi-AZ lorsque vous sélectionnez l'`MultiAZ`option. Pour plus d'informations sur les instances de réplication, consultez la section Utilisation d'une instance de réplication AWS DMS.

3 : point de terminaison AWS DMS source : créer, créer pour Mongo DB, créer pour S3

Vous pouvez utiliser la console AMS ou API/CLI créer un point de terminaison source AMS DMS pour différentes bases de données. Nous fournissons trois exemples.

Point de terminaison source DMS : création

Création d'un point de terminaison source DMS avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.

- Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.

3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler le RFC ou en créer une copie avec les options en haut de la page.

Création d'un point de terminaison source DMS avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution pour ce type de modification dans un fichier JSON nommé CreateDmsSeParams .json.

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

2. Modifiez et enregistrez le fichier JSON des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
   "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. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme CreateDmsSeRfc .json :

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

4. Modifiez et enregistrez le fichier CreateDmsSeRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le CreateDmsSeRfc fichier :

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

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

Avant de créer le point de terminaison DMS, assurez-vous que votre mot de passe ne contient pas de caractères non pris en charge. Pour plus d'informations, consultez la section Création de points de terminaison source et cible dans le guide de l'AWS Database Migration Service utilisateur.

Pour en savoir plus, consultez la section Sources pour la migration des données.

Pour un point de terminaison source S3, consultezPoint de terminaison source DMS pour S3 : création.

Pour un point de terminaison source de base de données Mongo, voirPoint de terminaison source DMS pour MongoDB : création.

Point de terminaison source DMS pour MongoDB : création

Création d'un point de terminaison source de base de données DMS Mongo avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.

4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler le RFC ou en créer une copie avec les options en haut de la page.

Création d'un point de terminaison source de base de données DMS Mongo avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste

de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution pour ce type de modification dans un fichier JSON nommé CreateDmsSeMongoParams .json.

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

2. Modifiez et enregistrez le fichier JSON des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
   "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. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme CreateDmsSeMongoRfc .json :

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

4. Modifiez et enregistrez le fichier CreateDmsSeMongoRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

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

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le CreateDmsSeMongoRfc fichier :

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

Vous recevez l'identifiant de la nouvelle RFC dans la réponse et pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

> **Note**
>
> Vous pouvez ajouter jusqu'à 50 balises, mais pour cela, vous devez activer la vue Configuration supplémentaire.

AMS DMS peut utiliser Mongo ou tout autre service de base de données relationnelle (RDS) comme point de terminaison source. Pour un point de terminaison source S3, consultezPoint de terminaison source DMS pour S3 : création.

Point de terminaison source DMS pour S3 : création

Création d'un point de terminaison source DMS S3 avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler le RFC ou en créer une copie avec les options en haut de la page.

Création d'un point de terminaison source DMS S3 avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}"

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution pour ce type de modification dans un fichier JSON nommé CreateDmsSe S3Params.json.

```
aws amscm get-change-type-version --change-type-id "ct-2oxl37nphsrjz" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsSeS3Params.json
```

2. Modifiez et enregistrez le fichier JSON des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"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. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme CreateDmsSe S3RFC.json :

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeS3Rfc.json
```

4. Modifiez et enregistrez le fichier CreateDmsSe S3RFC.json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2oxl37nphsrjz",
"Title":                "DMS_Source_S3"
}
```

5. Créez le RFC en spécifiant le fichier de paramètres d'exécution et le fichier CreateDmsSe S3Rfc :

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeS3Rfc.json --execution-
parameters file://CreateDmsSeS3Params.json
```

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

> **Note**
>
> Vous pouvez ajouter jusqu'à 50 balises, mais pour cela, vous devez activer la vue Configuration supplémentaire.

AMS DMS peut utiliser S3 ou n'importe quel point de terminaison source du Relational Database Service (RDS). Pour un point de terminaison source de base de données Mongo, voir[Point de terminaison source DMS pour MongoDB : création](#).

4 : point de terminaison AWS DMS cible : créer, créer pour S3

Vous pouvez utiliser la console AMS ou API/CLI créer un point de terminaison cible AMS DMS pour différentes bases de données. Nous fournissons deux exemples.

Point de terminaison cible DMS : création

AMS DMS peut utiliser S3 ou tout autre service de base de données relationnelle (RDS) avec MySQL, MariaDB, Oracle, Postgresql ou Microsoft SQL comme point de terminaison cible.

Création d'un point de terminaison cible DMS avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.

4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Création d'un point de terminaison cible DMS avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste

de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution pour ce type de modification dans un fichier JSON nommé CreateDmsTeParams .json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3gf8dolbo8x9p" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeParams.json
   ```

2. Modifiez et enregistrez le fichier JSON des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
   "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. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme CreateDmsTeRfc .json :

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

4. Modifiez et enregistrez le fichier CreateDmsTeRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3gf8dolbo8x9p",
"Title":                "DB-DMS-Target-Endpoint"
}
```

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le CreateDmsTeRfc fichier :

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeRfc.json --execution-
parameters file://CreateDmsTeParams.json
```

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

- Ce type de modification est désormais disponible en version 2.0.
- AMS DMS peut utiliser S3 ou tout autre service de base de données relationnelle (RDS) avec MySQL, MariaDB, Oracle, Postgresql ou Microsoft SQL comme point de terminaison cible. Pour un point de terminaison cible S3, consultezPoint de terminaison cible DMS pour S3 : création.
- Pour plus d'informations, consultez la section Cibles pour la migration des données.
- Vous pouvez ajouter jusqu'à 50 balises, mais pour cela, vous devez activer la vue Configuration supplémentaire.

Point de terminaison cible DMS pour S3 : création

Création d'un point de terminaison cible DMS S3 avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Création d'un point de terminaison cible DMS S3 avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"}}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution de ce type de modification dans un fichier JSON ; cet exemple le nomme CreateDmsTe S3Params.json :

```
aws amscm get-change-type-version --change-type-id "ct-05muqzievnxk5" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeS3Params.json
```

2. Modifiez et enregistrez les paramètres d'exécution dans le fichier CreateDmsTe S3Params.json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"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. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme CreateDmsTe S3RFC.json :

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsTeS3Rfc.json
```

4. Modifiez et enregistrez le fichier CreateDmsTe S3RFC.json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-05muqzievnxk5",
"Title":                "DMS_Target_S3"
}
```

5. Créez le RFC en spécifiant le fichier de paramètres d'exécution et le fichier CreateDmsTe S3Rfc :

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeS3Rfc.json --execution-
parameters file://CreateDmsTeS3Params.json
```

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

> **Note**
>
> Vous pouvez ajouter jusqu'à 50 balises, mais pour cela, vous devez activer la vue Configuration supplémentaire.

AMS fournit un type de modification distinct pour créer un point de terminaison cible pour S3. Pour plus d'informations, consultez Utilisation d'Amazon S3 comme cible pour AWS Database Migration Service et Attributs de connexion supplémentaires lors de l'utilisation d'Amazon S3 comme cible pour AWS DMS.

5 : tâche AWS DMS de réplication : créer

Vous pouvez utiliser la console AMS ou API/CLI créer une tâche de AWS DMS réplication AMS.

Création d'une tâche AWS DMS de réplication

Création d'une tâche de AWS DMS réplication avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.

- Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

  Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
- Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.

3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Création d'une tâche AWS DMS de réplication avec la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

Surveillez la `aws amscm get-rfc --rfc-id` *ID* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution de ce type de modification dans un fichier JSON ; cet exemple le nomme CreateDmsRtParams .json :

```
aws amscm get-change-type-version --change-type-id "ct-1d2fml15b9eth" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRtParams.json
```

2. Modifiez et enregistrez le fichier JSON des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"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. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme CreateDmsRtRfc .json :

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

4. Modifiez et enregistrez le fichier CreateDmsRtRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1d2fml15b9eth",
"Title":                "DMS-RI-Create-RFC"
}
```

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le CreateDmsRtRfc fichier :

```
aws amscm create-rfc --cli-input-json file://CreateDmsRtRfc.json --execution-
parameters file://CreateDmsRtParams.json
```

Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

Vous pouvez créer une AWS DMS tâche qui capture trois types différents de modifications ou de données. Pour plus d'informations, consultez les sections Utilisation des tâches AWS DMS, Création d'une tâche et Création de tâches pour une réplication continue à l'aide d'AWS DMS.

AWS DMS gestion

AWS DMS exemples de gestion.

Lancer AWS DMS la tâche de réplication

Démarrage d'une tâche de AWS DMS réplication avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.

- Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.

3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Démarrage d'une tâche AWS DMS de réplication à l'aide de la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\":[\"\"]}}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution de ce type de modification dans un fichier JSON ; cet exemple le nomme StartDmsRtParams .json :

   ```
   aws amscm get-change-type-version --change-type-id "ct-1yq7hhqse71yg" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StartDmsRtParams.json
   ```

2. Modifiez et enregistrez le fichier JSON des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
     "DocumentName": "AWSManagedServices-StartDmsTask",
     "Region": "us-east-1",
     "Parameters": {
         "ReplicationTaskArn": [
           "TASK_ARN"
   ```

```
      ],
      "StartReplicationTaskType": [
        "start-replication"
      ],
      "CdcStartPosition": [
        ""
      ],
      "CdcStopPosition": [
        ""
      ]
  }
}
```

3. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme StartDmsRtRfc .json :

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

4. Modifiez et enregistrez le fichier StartDmsRtRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
{
  "ChangeTypeId": "ct-1yq7hhqse71yg",
  "ChangeTypeVersion": "1.0",
  "Title": "Start DMS Replication Task"
}
```

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le StartDmsRtRfc fichier :

```
aws amscm create-rfc --cli-input-json file://StartDmsRtRfc.json --execution-
parameters file://StartDmsRtParams.json
```

   Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

Vous pouvez démarrer une tâche de AWS DMS réplication à l'aide de la console AMS ou de l'API/CLI AMS. Pour plus d'informations, consultez la section Utilisation des tâches AWS DMS.

Arrêter AWS DMS la tâche de réplication

Arrêt d'une tâche de AWS DMS réplication avec la console

Capture d'écran de ce type de modification dans la console AMS :

Fonctionnement :

1. Accédez à la page Créer une RFC : Dans le volet de navigation de gauche de la console AMS, cliquez RFCspour ouvrir la page de RFCs liste, puis cliquez sur Créer une RFC.
2. Choisissez un type de modification (CT) populaire dans la vue Parcourir les types de modification par défaut, ou sélectionnez un CT dans la vue Choisir par catégorie.
   - Parcourir par type de modification : vous pouvez cliquer sur un CT populaire dans la zone de création rapide pour ouvrir immédiatement la page Run RFC. Notez que vous ne pouvez pas choisir une ancienne version CT avec création rapide.

     Pour trier CTs, utilisez la zone Tous les types de modifications dans l'affichage Carte ou Tableau. Dans l'une ou l'autre vue, sélectionnez un CT, puis cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC. Le cas échéant, une option Créer avec une ancienne version apparaît à côté du bouton Créer une RFC.
   - Choisissez par catégorie : sélectionnez une catégorie, une sous-catégorie, un article et une opération et la zone de détails du CT s'ouvre avec une option permettant de créer avec une ancienne version, le cas échéant. Cliquez sur Créer une RFC pour ouvrir la page Exécuter une RFC.
3. Sur la page Run RFC, ouvrez la zone de nom du CT pour voir la boîte de détails du CT. Un sujet est requis (il est renseigné pour vous si vous choisissez votre CT dans la vue Parcourir les types de modification). Ouvrez la zone de configuration supplémentaire pour ajouter des informations sur le RFC.

   Dans la zone Configuration de l'exécution, utilisez les listes déroulantes disponibles ou entrez des valeurs pour les paramètres requis. Pour configurer les paramètres d'exécution facultatifs, ouvrez la zone de configuration supplémentaire.
4. Lorsque vous avez terminé, cliquez sur Exécuter. S'il n'y a aucune erreur, la page RFC créée avec succès s'affiche avec les détails de la RFC soumise et le résultat d'exécution initial.
5. Ouvrez la zone Paramètres d'exécution pour voir les configurations que vous avez soumises. Actualisez la page pour mettre à jour l'état d'exécution de la RFC. Vous pouvez éventuellement annuler la RFC ou en créer une copie à l'aide des options en haut de la page.

Arrêt d'une tâche AWS DMS de réplication à l'aide de la CLI

Fonctionnement :

1. Utilisez soit le Inline Create (vous émettez une `create-rfc` commande avec tous les paramètres RFC et d'exécution inclus), soit le Template Create (vous créez deux fichiers JSON, un pour les paramètres RFC et un pour les paramètres d'exécution) et émettez la `create-rfc` commande avec les deux fichiers en entrée. Les deux méthodes sont décrites ici.
2. Soumettez la `aws amscm submit-rfc --rfc-id` *`ID`* commande RFC : avec l'ID RFC renvoyé.

   Surveillez la `aws amscm get-rfc --rfc-id` *`ID`* commande RFC :.

Pour vérifier la version du type de modification, utilisez cette commande :

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

> **Note**
>
> Vous pouvez utiliser n'importe quel `CreateRfc` paramètre avec n'importe quelle RFC, qu'ils fassent ou non partie du schéma du type de modification. Par exemple, pour recevoir des notifications lorsque le statut de la RFC change, ajoutez cette ligne `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` aux paramètres RFC de la demande (et non aux paramètres d'exécution). Pour obtenir la liste de tous les CreateRfc paramètres, consultez le manuel AMS Change Management API Reference.

CRÉATION EN LIGNE :

Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne), puis soumettez l'ID RFC renvoyé. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

```
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\"]}}"
```

CRÉATION D'UN MODÈLE :

1. Exportez les paramètres d'exécution de ce type de modification dans un fichier JSON ; cet exemple le nomme StopDmsRtParams .json :

   ```
   aws amscm get-change-type-version --change-type-id "ct-1vd3y4ygbqmfk" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StopDmsRtParams.json
   ```

2. Modifiez et enregistrez le fichier JSON des paramètres d'exécution. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
     "DocumentName": "AWSManagedServices-StopDmsTask",
     "Region": "us-east-1",
     "Parameters": {
         "ReplicationTaskArn": [
           "TASK_ARN"
         ]
     }
   }
   ```

3. Exportez le modèle JSON dans un fichier de votre dossier actuel ; cet exemple le nomme StopDmsRtRfc .json :

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

4. Modifiez et enregistrez le fichier StopDmsRtRfc .json. Par exemple, vous pouvez remplacer le contenu par quelque chose comme ceci :

   ```
   {
     "ChangeTypeId": "ct-1vd3y4ygbqmfk",
     "ChangeTypeVersion": "1.0",
     "Title": "Stop DMS Replication Task"
   }
   ```

5. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le StopDmsRtRfc fichier :

```
aws amscm create-rfc --cli-input-json file://StopDmsRtRfc.json --execution-
parameters file://StopDmsRtParams.json
```

Vous recevez l'identifiant de la nouvelle RFC dans la réponse et vous pouvez l'utiliser pour soumettre et surveiller la RFC. Tant que vous ne l'avez pas soumise, la RFC reste en cours d'édition et ne démarre pas.

Conseils

Vous pouvez arrêter une tâche de réplication DMS à l'aide de la console AMS ou de l'API/CLI AMS. Pour plus d'informations, consultez la section Utilisation des tâches AWS DMS.

Importation de base de données (DB) vers AMS RDS pour Microsoft SQL Server

> **Note**
>
> Les points de terminaison AMS API/CLI (amscm et amsskms) se trouvent dans la région AWS de Virginie du Nord,. `us-east-1` En fonction de la configuration de votre authentification et de la région AWS dans laquelle se trouvent votre compte et vos ressources, vous devrez peut-être en ajouter `--region us-east-1` lors de l'émission de commandes. Vous devrez peut-être également ajouter`--profile saml`, s'il s'agit de votre méthode d'authentification.

Le processus d'importation de bases de données vers AMS RDS pour SQL Server repose sur les types de modification AMS (CTs) soumis sous forme de demandes de modification (RFCs) et utilise les paramètres de l'API Amazon RDS en entrée. MicroSoft SQL Server est un système de gestion de base de données relationnelle (RDBMS). Pour en savoir plus, consultez également : Amazon Relational Database Service (Amazon RDS) et la référence de l'API RDS ou Amazon RDS.

> **Note**
>
> Assurez-vous que chaque RFC se termine correctement avant de passer à l'étape suivante.

Étapes d'importation de haut niveau :

1. Sauvegardez votre base de données MS SQL locale source dans un fichier .bak (sauvegarde)
2. Copiez le fichier .bak dans le compartiment de transit (chiffré) Amazon Simple Storage Service (S3)
3. Importez le fichier .bak dans une nouvelle base de données sur votre instance MS SQL Amazon RDS cible

Prérequis:

- Pile MS SQL RDS dans AMS
- Stack RDS avec option de restauration () `SQLSERVER_BACKUP_RESTORE`
- Seau Transit S3
- Rôle IAM avec accès au compartiment permettant à Amazon RDS d'assumer le rôle
- Une EC2 instance sur laquelle MS SQL Management Studio est installé pour gérer le RDS (il peut s'agir d'un poste de travail sur site)

Configuration

Effectuez ces tâches pour démarrer le processus d'importation.

1. Soumettez une RFC pour créer une pile RDS à l'aide de Deployment | Advanced stack components | Stack RDS database | Create (ct-2z60dyvto9g6c). N'utilisez pas le nom de la base de données cible (`RDSDBName`paramètre) dans la demande de création, la base de données cible sera créée lors de l'importation. Assurez-vous de laisser suffisamment d'espace (`RDSAllocatedStorage`paramètre). Pour plus de détails sur cette procédure, consultez le guide de gestion des modifications AMS RDS DB Stack | Create.
2. Soumettez une RFC pour créer le compartiment S3 de transit (s'il n'existe pas déjà) à l'aide de Deployment | Advanced stack components | S3 storage | Create (ct-1a68ck03fn98r). Pour plus de détails sur cette procédure, consultez le guide de gestion des modifications AMS S3 Storage | Create.
3. Soumettez une RFC de gestion | Autre | Autre | Mise à jour (ct-1e1xtak34nx76) pour implémenter la avec les détails suivants : `customer_rds_s3_role`

   Dans la console :

- Objet : « Pour prendre en charge l'importation de bases de données MS SQL Server, implémentez `customer_rds_s3_role` sur ce compte.
- Nom du compartiment Transit S3 :*BUCKET_NAME*.
- Informations de contact :*EMAIL*.

Avec un fichier ImportDbParams .json pour la CLI :

```
{
       "Comment": "{"Transit S3 bucket name":"BUCKET_NAME"}",
       "Priority": "High"
     }
}
```

4. Soumettez une RFC Management | Other | Other | Update demandant à AMS de définir l'`SQLSERVER_BACKUP_RESTORE`option sur le RDS créé à l'étape 1 (utilisez l'ID de pile indiqué dans la sortie de l'étape 1, et le rôle `customer_rds_s3_role` IAM dans cette demande, dans cette demande).
5. Soumettez une RFC pour créer une EC2 instance (vous pouvez utiliser n'importe quel poste de travail/instance existant EC2 ou sur site) et installez Microsoft SQL Management Studio sur l'instance.

Importation de la base de données

Pour importer la base de données (DB), procédez comme suit.

1. Sauvegardez votre base de données source sur site à l'aide de la sauvegarde et de la restauration natives de MS SQL (voir Support pour la sauvegarde et la restauration natives dans SQL Server). À la suite de l'exécution de cette opération, vous devriez disposer d'un fichier .bak (sauvegarde).
2. Téléchargez le fichier .bak dans un compartiment de transit S3 existant à l'aide de la CLI AWS S3 ou de la console AWS S3. Pour plus d'informations sur les compartiments S3 de transit, consultez la section Protection des données à l'aide du chiffrement.
3. Importez le fichier .bak dans une nouvelle base de données sur votre instance MS SQL RDS for SQL Server cible (pour plus de détails sur les types, consultez les types d'instances Amazon RDS for MySQL) :

a. Connectez-vous à l' EC2 instance (station de travail sur site) et ouvrez MS SQL Management Studio

b. Connectez-vous à l'instance RDS cible créée comme condition préalable à l'étape #1. Suivez cette procédure pour vous connecter : Connexion à une instance de base de données exécutant le moteur de base de données Microsoft SQL Server

c. Démarrez la tâche d'importation (restauration) avec une nouvelle requête SQL (Structured Query Language) (pour plus de détails sur les requêtes SQL, voir Introduction au SQL). Le nom de la base de données cible doit être nouveau (n'utilisez pas le même nom que celui de la base de données que vous avez créée précédemment). Exemple sans chiffrement :

```
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. Vérifiez régulièrement l'état de la tâche d'importation en exécutant cette requête dans une fenêtre séparée :

```
exec msdb.dbo.rds_task_status;
```

Si le statut passe à Échec, recherchez les détails de l'échec dans le message.

Nettoyage

Une fois que vous avez importé la base de données, vous souhaiterez peut-être supprimer des ressources inutiles. Procédez comme suit.

1. Supprimez le fichier de sauvegarde (.bak) du compartiment S3. Pour ce faire, vous pouvez utiliser la console S3. Pour la commande CLI permettant de supprimer un objet d'un compartiment S3, consultez rm dans le manuel de référence des commandes de l'AWS CLI.
2. Supprimez le compartiment S3 si vous ne comptez pas l'utiliser. Pour savoir comment procéder, voir Delete Stack.
3. Si vous ne prévoyez pas d'importer MS SQL, soumettez une RFC Management | Other | Other | Update (ct-0xdawir96cy7k) et demandez à AMS de supprimer le rôle IAM. `customer_rds_s3_role`

Déploiements d'applications Tier and Tie dans AMS

Dans le cadre d'un déploiement Tier and Tie, vous créez, configurez et déployez les ressources d'une pile indépendamment en utilisant des composants distincts RFCs, et vous utilisez les composants IDs de la pile au fur et à mesure de votre progression pour les associer les uns aux autres.

Par exemple, pour déployer un site Web de haute disponibilité (redondant) derrière un équilibreur de charge et une base de données, en utilisant une approche Tier and Tie, soumettez RFCs une base de données, un équilibreur de charge et deux EC2 instances ou un groupe Auto Scaling, et configurez les EC2 instances ou le groupe Auto Scaling avec l'ID de l'ELB que vous avez créé.

Une fois les ressources déployées, vous pouvez soumettre une modification à un groupe de sécurité pour permettre aux ressources de communiquer avec la base de données. Pour plus de détails sur la création de groupes de sécurité, consultez la section Créer un groupe de sécurité.

Déploiements complets d'applications dans AMS

Dans le cadre d'un déploiement Full Stack, vous soumettez une RFC avec un CT qui crée et configure tout ce dont vous avez besoin en même temps. Par exemple, pour déployer le site Web de haute disponibilité qui vient d'être décrit (EC2 instances, équilibreur de charge et base de données), vous devez utiliser un CT qui, ensemble, a créé et configuré un groupe Auto Scaling, un équilibreur de charge, une base de données et les paramètres de groupe de sécurité requis pour que toutes les instances fonctionnent comme une pile. Des exemples de deux AMS effectuant CTs cette opération sont décrits ci-dessous.

- Stack à deux niveaux à haute disponibilité (ct-06mjngx5flwto) : ce type de modification vous permet de créer une pile et de configurer un groupe Auto Scaling, une base de données basée sur RDS, un Load Balancer, une application et une configuration. CodeDeploy Notez que l'équilibreur de charge n'est pas considéré comme un niveau car il est partagé entre plusieurs applications en tant qu'appliance réseau et les CodeDeploy fonctions sont également considérées comme une appliance. En outre, il crée un groupe de CodeDeploy déploiement (avec le nom que vous donnez à l' CodeDeploy application) qui peut être utilisé pour déployer vos applications. Les paramètres du groupe de sécurité permettant aux ressources de fonctionner ensemble sont automatiquement créés.
- High Availability One-Tier Stack (ct-09t6q7j9v5hrn) : ce type de modification vous permet de créer une pile et de configurer un groupe Auto Scaling et un Application Load Balancer. Les paramètres du groupe de sécurité qui permettent aux ressources de fonctionner ensemble sont automatiquement créés.

Utilisation des types de modification du provisionnement () CTs

AMS est responsable de votre infrastructure gérée. Pour apporter des modifications, vous devez soumettre un RFC avec la classification CT correcte (catégorie, sous-catégorie, article et opération). Cette section décrit comment trouver CTs, déterminer s'ils répondent à vos besoins et demander un nouveau scanner s'il n'y en a pas.

Vérifiez si un scanner existant répond à vos exigences

Une fois que vous avez déterminé ce que vous souhaitez déployer avec AMS, l'étape suivante consiste à étudier CTs les CloudFormation modèles existants pour voir si une solution existe déjà.

Lorsque vous créez une RFC, vous devez spécifier le CT. Vous pouvez utiliser l'API/CLI AWS Management Console ou AMS. Des exemples d'utilisation des deux sont décrits ci-dessous.

Vous pouvez utiliser la console ou le API/CLI pour rechercher un ID de type de modification (CT) ou une version. Il existe deux méthodes, soit la recherche, soit le choix de la classification. Pour les deux types de sélection, vous pouvez trier la recherche en choisissant le plus fréquemment utilisé, le plus récemment utilisé ou Alphabétique.

YouTube Vidéo : Comment créer une RFC à l'aide de l'AWS Managed Services CLI et où puis-je trouver le schéma CT ?

Dans la console AMS, sur la page RFCs-> Créer une RFC :

- Lorsque l'option Parcourir par type de modification est sélectionnée (valeur par défaut), vous pouvez soit :
  - Utilisez la zone de création rapide pour sélectionner l'une des solutions les plus populaires d'AMS CTs. Cliquez sur une étiquette et la page Run RFC s'ouvre avec l'option Objet remplie automatiquement pour vous. Complétez les options restantes selon vos besoins et cliquez sur Exécuter pour soumettre la RFC.
  - Vous pouvez également faire défiler l'écran jusqu'à la zone Tous les types de modification et commencer à taper un nom CT dans la case d'option. Vous n'avez pas besoin du nom exact ou complet du type de modification. Vous pouvez également rechercher un CT par identifiant de type de modification, classification ou mode d'exécution (automatique ou manuel) en saisissant les mots pertinents.

    Lorsque la vue Cartes par défaut est sélectionnée, les cartes CT correspondantes apparaissent au fur et à mesure que vous tapez, sélectionnez une carte et cliquez sur Créer une RFC. Une

fois la vue sous forme de tableau sélectionnée, choisissez le CT approprié et cliquez sur Créer une RFC. Les deux méthodes ouvrent la page Run RFC.

- Sinon, pour explorer les choix de type de modification, cliquez sur Choisir par catégorie en haut de la page pour ouvrir une série de boîtes d'options déroulantes.
- Choisissez une catégorie, une sous-catégorie, un article et une opération. La zone d'information correspondant à ce type de modification apparaît et un panneau apparaît en bas de page.
- Lorsque vous êtes prêt, appuyez sur Entrée pour afficher la liste des types de modifications correspondants.
- Choisissez un type de modification dans la liste. La zone d'information correspondant à ce type de modification apparaît au bas de la page.
- Une fois que vous avez sélectionné le type de modification correct, choisissez Create RFC.

> **Note**
>
> L'AMS CLI doit être installée pour que ces commandes fonctionnent. Pour installer l'API ou la CLI AMS, rendez-vous sur la page Ressources pour développeurs de la console AMS. Pour des informations de référence sur l'API AMS CM ou l'API AMS SKMS, consultez la section Ressources d'information AMS du guide de l'utilisateur. Vous devrez peut-être ajouter une `--profile` option d'authentification ; par exemple,`aws amsskms` *`ams-cli-command`* `--profile SAML`. Vous devrez peut-être également ajouter `--region` cette option car toutes les commandes AMS sont exécutées à partir de us-east-1, par exemple. `aws amscm` *`ams-cli-command`* `--region=us-east-1`

> **Note**
>
> Les points de terminaison AMS API/CLI (amscm et amsskms) se trouvent dans la région AWS de Virginie du Nord,. `us-east-1` En fonction de la configuration de votre authentification et de la région AWS dans laquelle se trouvent votre compte et vos ressources, vous devrez peut-être en ajouter `--region us-east-1` lors de l'émission de commandes. Vous devrez peut-être également ajouter`--profile saml`, s'il s'agit de votre méthode d'authentification.

Pour rechercher un type de modification à l'aide de l'API AMS CM (voir ListChangeTypeClassificationSummaries) ou de la CLI :

Vous pouvez utiliser un filtre ou une requête pour effectuer une recherche. L'ListChangeTypeClassificationSummaries opération comporte des options de filtres pour `Category` `SubcategoryItem`,, et`Operation`, mais les valeurs doivent correspondre exactement aux valeurs existantes. Pour des résultats plus flexibles lors de l'utilisation de la CLI, vous pouvez utiliser l'`--query`option.

Changer le type de filtrage avec l'API/CLI AMS CM

Attribut	Valeurs valides	Condition valide/par défaut	Remarques
ChangeTypeId	Toute chaîne représentant un ChangeTypeId (par exemple : ct-abc123 xyz7890)	Égal à	Pour le type de modification IDs, consultez la référence du type de modification. Pour le type de modification IDs, voir Trouver un type de modification ou CSIO.
Catégorie	Tout texte de forme libre	Contains	Les expressions régulières dans chaque champ individuel ne sont pas prises en charge. Recherche insensible aux majuscules et minuscules
Sous-catégorie			
Élément			
Opération			

1. Voici quelques exemples de classification des types de modification de liste :

   La commande suivante répertorie toutes les catégories de types de modifications.

```
aws amscm list-change-type-categories
```

La commande suivante répertorie les sous-catégories appartenant à une catégorie spécifiée.

```
aws amscm list-change-type-subcategories --category CATEGORY
```

La commande suivante répertorie les éléments appartenant à une catégorie et à une sous-catégorie spécifiées.

```
aws amscm list-change-type-items --category CATEGORY --subcategory SUBCATEGORY
```

2. Voici quelques exemples de recherche de types de modifications à l'aide de requêtes CLI :

   La commande suivante recherche dans les résumés de classification CT ceux qui contiennent « S3 » dans le nom de l'élément et crée une sortie de la catégorie, de la sous-catégorie, de l'élément, de l'opération et de l'ID de type de modification sous forme de tableau.

```
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. Vous pouvez ensuite utiliser l'ID du type de modification pour obtenir le schéma CT et examiner les paramètres. La commande suivante génère le schéma dans un fichier JSON nommé Creates3Params.Schema.json.

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateS3Params.schema.json
```

Pour plus d'informations sur l'utilisation des requêtes CLI, consultez Comment filtrer la sortie avec l'option --query et la référence du langage de requête, JMESPath Spécification.

4. Une fois que vous avez obtenu l'identifiant du type de modification, nous vous recommandons de vérifier la version du type de modification afin de vous assurer qu'il s'agit de la dernière version. Utilisez cette commande pour trouver la version correspondant à un type de modification spécifié :

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

   Pour trouver le `AutomationStatus` type de modification correspondant à un type de modification spécifique, exécutez cette commande :

   ```
   aws amscm --profile saml get-change-type-version --change-type-id CHANGE_TYPE_ID --
   query "ChangeTypeVersion.{AutomationStatus:AutomationStatus.Name}"
   ```

   Pour trouver le `ExpectedExecutionDurationInMinutes` type de modification correspondant à un type de modification spécifique, exécutez cette commande :

   ```
   aws amscm --profile saml get-change-type-version --change-type-id ct-14027q0sjyt1h
    --query "ChangeTypeVersion.{ExpectedDuration:ExpectedExecutionDurationInMinutes}"
   ```

Une fois que vous avez trouvé un CT qui vous semble approprié, examinez le schéma JSON des paramètres d'exécution qui lui est associé pour savoir s'il répond à votre cas d'utilisation.

Utilisez cette commande pour générer un schéma CT dans un fichier JSON nommé d'après le CT ; cet exemple génère le schéma de stockage Create S3 :

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
--query "ChangeTypeVersion.ExecutionInputSchema" --output text >
CreateBucketParams.json
```

Examinons de près ce que propose ce schéma.

Schéma de création du compartiment S3

```
{
  "$schema": "http://json-schema.org/draft-04/sch
ema#",
"name": "Create S3 Storage
```

Le schéma commence par le CT (« description »), qui indique à quoi sert le schéma. Dans ce cas, pour créer une pile de stockage S3.

```
"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":{
      "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",
```

Ensuite, vous avez les propriétés obligatoires et facultatives que vous pouvez spécifier. Les valeurs de propriété par défaut sont indiquées. Les propriétés requises sont répertoriées à la fin du schéma.

Dans la StackTemplateId zone, vous voyez qu'il existe un modèle de pile spécifique pour ce CT et ce schéma, et que son ID est une valeur de propriété obligatoire.

Le schéma vous permet de baliser la pile que vous créez, à des fins de comptabilité interne. En outre, certaines options, comme la sauvegard e, nécessitent une balise key:backup et value:Tru e. Pour des informations détaillées, consultez Tagging Your Amazon EC2 Resources.

```
            "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": {
      "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,
```

La section Paramètres du schéma CT JSON permet de fournir les paramètres d'exécution.

Pour ce schéma, seuls l'ACL et BucketName les paramètre s d'exécution sont obligatoires.

```
          "maxLength": 63
        }
      },
      "additionalProperties": false,
      "required": [
        "AccessControl",
        "BucketName"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Demandez un nouveau CT

Après avoir examiné le schéma, vous pouvez décider qu'il ne fournit pas suffisamment de paramètres pour créer le déploiement souhaité. Si tel est le cas, examinez les CloudFormation modèles existants pour trouver celui qui correspond le mieux à ce que vous recherchez. Une fois que vous connaissez les paramètres supplémentaires dont vous avez besoin, soumettez un document Management | Other | Other | Create CT.

> **Note**
>
> Toutes les autres | Autres créations et mises à jour CTs reçoivent l'attention d'un opérateur AMS, qui vous contactera pour discuter du nouveau CT.

Pour soumettre une demande de nouveau CT, accédez à la console AMS par le biais de la procédure normale, AWS Management Consolepuis suivez ces étapes.

1. Dans le menu de navigation de gauche, cliquez sur RFCs.

La page du RFCs tableau de bord s'ouvre.

2. Cliquez sur Create.

   La page Créer une demande de modification s'ouvre.

3. Sélectionnez Gestion dans la liste déroulante des catégories, puis Autre pour la sous-catégorie et l'article. Pour l'opération, choisissez Create. Le RFC devra être approuvé avant de pouvoir être mis en œuvre.
4. Entrez les informations expliquant pourquoi vous souhaitez le CT, par exemple : demande d'un CT de stockage Create S3 modifié qui permet la personnalisation ACLs, sur la base du CT de stockage Create S3 existant. Cela devrait donner lieu à un nouveau CT : Deployment | Advanced Stack Components | Stockage S3 | Création d'une ACL personnalisée S3. Ce nouveau scanner pourrait être public.
5. Cliquez sur Soumettre.

   Votre RFC s'affiche sur le tableau de bord RFC.

Testez le nouveau CT

Une fois qu'AWS Managed Services a créé ce nouveau CT, vous pouvez le tester en soumettant une RFC avec celui-ci. Si vous avez travaillé avec AMS pour que le nouveau CT soit préapprouvé, vous pouvez simplement suivre une soumission RFC standard et surveiller le résultat (pour plus de détails sur la soumission RFCs, voir Création et soumission d'une RFC). Si le nouveau CT n'est pas pré-approuvé (vous voulez être sûr qu'il ne sera jamais exécuté sans approbation explicite), vous devrez discuter de sa mise en œuvre avec AMS chaque fois que vous souhaiterez l'exécuter.

Démarrages rapides

Rubriques

En combinant plusieurs types de modifications AMS, vous pouvez accomplir des tâches complexes.

Vous pouvez utiliser le système de gestion des modifications AMS pour configurer le planificateur de ressources AMS, pour une zone d'atterrissage multi-comptes (MALZ) ou pour un compte de zone d'atterrissage à compte unique (SALZ). Le processus varie. Également, pour effectuer des transferts de fichiers et des instantanés entre comptes.

Démarrage rapide du planificateur de ressources AMS

Utilisez ce guide de démarrage rapide pour implémenter AMS Resource Scheduler, un planificateur d'instance basé sur des balises afin de réduire les coûts dans AMS Advanced.

Le planificateur de ressources AMS est basé sur le planificateur d'instances AWS.

Terminologie du planificateur de ressources AMS

Avant de commencer, il est bon de connaître la terminologie du planificateur de ressources AMS :

- période : chaque calendrier doit contenir au moins une période qui définit le ou les moments pendant lesquels l'instance doit s'exécuter. Un calendrier peut contenir plusieurs périodes. Lorsque plusieurs périodes sont utilisées dans un calendrier, le planificateur de ressources applique l'action de démarrage appropriée lorsqu'au moins une des règles de période est vraie.
- fuseau horaire : pour obtenir la liste des valeurs de fuseau horaire acceptables à utiliser dans le DefaultTimezoneparamètre référencé ultérieurement, consultez la colonne TZ de la liste des fuseaux horaires de la base de données TZ.
- hibernation : lorsqu'elles sont définies sur true, les EC2 instances activées pour l'hibernation et répondant aux exigences d'hibernation sont mises en veille prolongée (). suspend-to-disk Consultez la EC2 console pour savoir si vos instances sont activées pour l'hibernation. Utilisez l'hibernation pour les EC2 instances Amazon arrêtées exécutant Amazon Linux.

- appliqué : lorsqu'il est défini sur true, en fonction du calendrier défini, le planificateur de ressources arrête une ressource en cours d'exécution si elle est démarrée manuellement en dehors de la période d'exécution, et il démarre une ressource si elle est arrêtée manuellement pendant la période d'exécution.
- retain_running : lorsqu'il est défini sur true, il empêche le planificateur de ressources d'arrêter une instance à la fin d'une période d'exécution si l'instance a été démarrée manuellement avant le début de la période. Par exemple, si une instance dont la période configurée s'étend de 9 h à 17 h est démarrée manuellement avant 9 h, le planificateur de ressources n'arrête pas l'instance à 17 heures.
- ssm-maintenance-window: Ajoutez une fenêtre de AWS Systems Manager maintenance sous forme de période de fonctionnement à un calendrier. Lorsque vous spécifiez le nom d'une fenêtre de maintenance qui existe dans le même compte et dans la même région AWS que votre stack déployée pour planifier vos EC2 instances Amazon, le planificateur de ressources démarre l'instance avant le début de la fenêtre de maintenance et arrête l'instance à la fin de la fenêtre de maintenance, si aucune autre période d'exécution n'indique que l'instance doit être exécutée et si l'événement de maintenance est terminé.

  Le planificateur de ressources utilise la AWS Lambda fréquence que vous avez spécifiée lors de la configuration initiale pour déterminer le délai de démarrage de votre instance avant le créneau de maintenance. Si vous définissez le AWS CloudFormation paramètre Fréquence sur 10 minutes ou moins, le planificateur de ressources démarre l'instance 10 minutes avant le créneau de maintenance. Si vous définissez une fréquence supérieure à 10 minutes, le planificateur de ressources démarre l'instance pendant le même nombre de minutes que la fréquence que vous avez spécifiée. Par exemple, si vous définissez la fréquence de la fenêtre de maintenance de Systems Manager sur 30 minutes, les Resource Schedulers démarrent l'instance 30 minutes avant la fenêtre de maintenance.

  Pour plus d'informations, consultez la section AWS Systems Manager Maintenance Windows.
- override-status : remplace temporairement les actions de démarrage et d'arrêt du calendrier configurées par le planificateur de ressources. Si vous configurez le champ pour qu'il s'exécute, le planificateur de ressources démarre, mais ne l'arrête pas, l'instance applicable. L'instance s'exécute jusqu'à ce que vous l'arrêtiez manuellement. Si vous définissez le statut de remplacement sur Arrêté, le planificateur de ressources arrête mais ne démarre pas l'instance applicable. L'instance ne s'exécute pas tant que vous ne l'avez pas démarrée manuellement.

Implémentation du planificateur de ressources AMS

Pour déployer une solution de planificateur de ressources AMS, procédez comme suit.

1. Soumettez une RFC Deployment | AMS Resource Scheduler | Solution | Deploy (ct-0ywnhc8e5k9z5) et fournissez les paramètres suivants :

   - SchedulingActive: Oui pour activer la planification des ressources, Non pour désactiver. La valeur par défaut est Oui.
   - ScheduledServices: Entrez une liste de services séparés par des virgules pour lesquels vous souhaitez planifier des ressources. Les valeurs valides incluent une combinaison d'autoscaling, ec2 et rds. La valeur par défaut est autoscaling, ec2, rds.
   - TagName: nom de la clé de balise qui associe les schémas de planification des ressources aux ressources de service. La valeur par défaut est Schedule.

     > **Note**
     >
     > Le déploiement de votre planificateur de ressources ne fonctionnera que sur les ressources dotées de cette balise.

   - DefaultTimezone: nom du fuseau horaire, sous la forme US/Pacific, à utiliser comme fuseau horaire par défaut. La valeur par défaut est UTC.
2. Après avoir reçu la confirmation que la RFC de la première étape a été exécutée avec succès, vous pouvez soumettre le type Période | Ajouter une modification.
3. Enfin, soumettez une RFC pour ajouter un calendrier à la période créée à la deuxième étape. Utilisez le type Calendrier | Ajouter une modification.

Implémentation et utilisation du planificateur de ressources AMS FAQs

Questions fréquemment posées sur AMS Resource Scheduler.

Q : Que se passe-t-il si j'active l'hibernation mais que l' EC2 instance ne le prend pas en charge ?

R : Hibernation enregistre le contenu de la mémoire d'instance (RAM) sur votre volume racine Amazon Elastic Block Store (Amazon EBS). Si ce champ est défini sur true, les instances sont mises en veille prolongée lorsque le Resource Scheduler les arrête.

Si vous configurez le planificateur de ressources pour utiliser l'hibernation mais que vos instances ne sont pas activées pour l'hibernation ou qu'elles ne répondent pas aux exigences d'hibernation, le planificateur de ressources enregistre un avertissement et les instances sont arrêtées sans mise en veille prolongée. Pour plus d'informations, consultez Hibernate Your Instance.

Q : Que se passe-t-il si je définis à la fois override_status et forced ?

R : Si vous définissez override_status sur running et que vous définissez forced sur true (empêche le démarrage manuel d'une instance en dehors d'une période d'exécution), le Resource Scheduler arrête l'instance.

Si vous définissez override_status sur stoppé et que vous définissez forced sur true (empêche l'arrêt manuel d'une instance pendant une période d'exécution), le planificateur de ressources redémarre l'instance.

> **Note**
>
> Si la valeur false est appliquée, le comportement de remplacement configuré est appliqué.

Q : Une fois le planificateur de ressources AMS déployé, comment puis-je le désactiver ou l'activer dans mon compte ?

R : Pour désactiver ou activer le planificateur de ressources AMS, procédez comme suit :

- Pour désactiver : créez une RFC à l'aide de State | Disable. Assurez-vous de régler le paramètre sur SchedulerStateDISABLE
- Pour activer : créez une RFC à l'aide de State | Enable. Assurez-vous de régler le paramètre sur SchedulerStateENABLE

Q Que se passe-t-il si la période du planificateur de ressources AMS tombe pendant la période de maintenance de mon application de correctifs ?

R : Le planificateur de ressources fonctionne en fonction de ses plannings configurés. S'il est configuré pour arrêter une instance pendant que l'application des correctifs est en cours, il arrête l'instance à moins que la fenêtre d'application des correctifs ne soit ajoutée en tant que période au calendrier avant le début de l'application des correctifs. En d'autres termes, le planificateur de ressources ne démarre automatiquement aucune instance arrêtée pour l'application de correctifs, sauf si une période définie est configurée. Pour éviter tout conflit avec votre fenêtre de maintenance

des correctifs, ajoutez la fenêtre de temps allouée à l'application des correctifs au calendrier du planificateur de ressources sous forme de période. Pour ajouter une période au calendrier existant, créez une RFC à l'aide de Période | Ajouter.

Q Si j'ai besoin d'un calendrier différent pour différentes EC2 instances, puis-je configurer plusieurs plannings dans mon compte ?

R : Oui, vous pouvez créer plusieurs plannings. Chaque programme peut comporter plusieurs périodes en fonction des besoins. Lorsque le planificateur de ressources AMS est activé dans le compte, une clé de balise est configurée. Par exemple, si la clé de balise est « Schedule », la valeur de tag peut varier en fonction des plannings, ce qui correspond au nom du planning d'AMS Resource Scheduler. Pour ajouter un nouveau calendrier, vous pouvez créer une RFC à l'aide du type de modification Management | AMS Resource Scheduler | Schedule | Add (ct-2bxelbn765ive), voir Planification | Ajouter.

Q : Où puis-je trouver les différents types de modifications pris en charge par AMS Resource Scheduler ?

R : AMS propose des types de modifications au planificateur de ressources pour déployer le planificateur de ressources AMS sur votre compte, l'activer ou le désactiver, définir, ajouter, mettre à jour et supprimer les calendriers et les périodes à utiliser avec celui-ci, et décrire (obtenir une description détaillée) les calendriers et les périodes.

Configuration de sauvegardes entre comptes (intra-région)

AWS Backup permet de copier des instantanés d'un compte à un autre au sein de la même région AWS, à condition que les deux comptes appartiennent à la même organisation AWS. Par exemple, dans la zone d'atterrissage multi-comptes (MALZ) AMS Advanced, vous pouvez configurer une copie instantanée entre comptes au sein de la même région AWS à l'aide de ce démarrage rapide.

Pour plus d'informations, consultez AWS Backup et AWS Organizations bring cross-account backup feature

Vous copiez des instantanés entre comptes à des fins de reprise après sinistre (DR). Pour protéger les données, vous devez peut-être conserver les instantanés au sein de la même région AWS, mais en dehors des limites du compte.

Présentation :

De manière générale, voici les étapes à suivre pour les sauvegardes entre comptes dans AMS :

- Créez un compte de destination pour héberger les sauvegardes dans la région AWS où est hébergée votre zone de landing zone AMS (étape 1)
- Création d'une clé KMS pour chiffrer les sauvegardes dans le compte de destination (étape 3)
- Créez un coffre-fort de sauvegarde dans le compte de destination de la même région que votre zone de landing zone AMS Advanced (étape 4)
- Activez le paramètre multi-comptes dans votre compte de gestion (étape 5)
- Création ou modification du plan et des règles de sauvegarde du compte source (étape 6)

> **Note**
>
> Assurez-vous que les comptes source et de destination se trouvent dans la même région. Si vous souhaitez copier vos sauvegardes d'une région à l'autre, contactez votre autorité de certification ou votre CSDM.

Pour activer et configurer les sauvegardes entre comptes, procédez comme suit :

1. Créez un compte de destination pour héberger les sauvegardes ; si vous possédez déjà un tel compte, vous pouvez ignorer cette étape. Pour créer le compte, soumettez un RFC depuis votre compte Management Payer en utilisant le type de changement de type (ct-1zdasmc2ewzrs) Déploiement | Zone d'atterrissage gérée | Compte de gestion | Créer un compte d'application (avec VPC).
2. [Facultatif] Si les ressources ou les instantanés sont chiffrés dans le compte source (par exemple, Prod), partagez la clé KMS utilisée pour le chiffrement avec le compte de destination. Pour ce faire, soumettez une RFC en utilisant le type de modification Management | Advanced stack components | KMS key | Update (ct-3ovo7px2vsa6n).
3. Dans le compte de destination, créez une clé KMS à utiliser pour le chiffrement de Backup Vault. Pour ce faire, soumettez une RFC à l'aide du type Deployment | Advanced stack components | KMS key | Create (auto) change (ct-1d84keiri1jhg).
4. Dans le compte de destination, créez un Backup Vault à l'aide de la clé créée précédemment. Les coffres-forts de sauvegarde AWS peuvent être créés à l'aide du type de modification automatique CFN, Deployment | Ingestion | Stack from CloudFormation Template | Create (ct-36cn2avfrrj9v). Dans la même demande, la politique d'accès au coffre-fort doit être modifiée

pour permettre au ou aux comptes source d'accéder au coffre-fort. Voici un exemple de politique :

Exemple CloudFormation de modèle pour un Backup Vault :

```
{
  "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. À partir de votre compte Management Payer, activez la sauvegarde entre comptes. Pour ce faire, soumettez une RFC en utilisant le type de modification Management | AWS Backup | Backup plan | Enable cross-account copy (Management account) (ct-2yja7ihh30ply).
6. Enfin, à partir du compte source d'où proviennent les sauvegardes, créez la ou les règles du plan de sauvegarde qui régissent les sauvegardes afin de copier les instantanés entre comptes. Pour ce faire, soumettez une RFC à l'aide du type Deployment | AWS Backup | Backup plan | Create change (ct-2hyozbpa0sx0m). Si vous devez mettre à jour un plan de sauvegarde existant,

soumettez une RFC en utilisant le type de modification Management | Other | Other | Update (ct-0xdawir96cy7k) avec les informations suivantes :

1. Le nom du plan de sauvegarde ainsi que le nom de la règle à mettre à jour.
2. L'ARN du coffre de sauvegarde du destination/ICE compte.
3. La durée de conservation pour laquelle days/months vous souhaitez conserver les instantanés dans le coffre ICE cible.

Didacticiels

Rubriques

Les didacticiels suivants décrivent les étapes à suivre pour créer une pile à deux niveaux avec le High Availability (ct-06mjngx5flwto), en utilisant la CLI, en utilisant la console et en déployant un groupe Linux ou RHEL Amazon Auto Scaling (ASG). EC2 Un tier-and-tie didacticiel similaire suit chacun (un pour la console et un pour la CLI), qui utilise des ressources séparées CTs, créées dans un ordre tel qu'elles vous permettent de relier les ressources au fur et à mesure de leur création.

Les descriptions de toutes les options de tomodensitométrie, y compris, se ChangeTypeId trouvent dans la managedservices/latest/ctref section/Change Type Reference.

Tutoriel sur console : pile à deux niveaux de haute disponibilité (Linux/RHEL)

Cette section décrit comment déployer un WordPress site haute disponibilité (HA) dans un environnement AMS à l'aide de la console AMS.

> **Note**
>
> Cette procédure de déploiement a été testée dans les environnements AMZN Linux et RHEL.

Résumé des tâches et des exigences RFCs :

1. Création d'une infrastructure (pile HA à deux niveaux)
2. Création d'un compartiment S3 pour les CodeDeploy applications
3. Créez le bundle WordPress d'applications et chargez-le dans le compartiment S3
4. Déployez l'application avec CodeDeploy
5. Accédez au WordPress site et connectez-vous pour valider le déploiement

6. Démanteler le déploiement

Les descriptions de toutes les options de tomodensitométrie ChangeTypeId, y compris, se trouvent dans AMS Change Type Reference.

Avant de commencer

Le Deployment | Advanced Stack Components | High Availability Two Tier Stack | Create CT crée un groupe Auto Scaling, un équilibreur de charge, une base de données, ainsi qu'un nom d' CodeDeploy application et un groupe de déploiement (avec le même nom que celui que vous avez donné à l'application). Pour plus d'informations sur la CodeDeploy section Qu'est-ce que c'est CodeDeploy ?

Cette procédure pas à pas utilise une RFC à deux niveaux de haute disponibilité qui inclut `UserData` et décrit également comment créer un WordPress bundle pouvant CodeDeploy être déployé.

L'exemple `UserData` illustré permet d'obtenir les métadonnées d'instance telles que l'ID d'instance, la région, etc., à partir d'une instance en cours d'exécution en interrogeant le service de métadonnées d' EC2 instance disponible sur http://169.254.169.254/latest/meta-data/. Cette ligne du script de données utilisateur :`REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed 's/[a-z]$//')`, extrait le nom de la zone de disponibilité du service de métadonnées dans la variable $REGION pour nos régions prises en charge, et l'utilise pour compléter l'URL du compartiment S3 dans lequel l' CodeDeploy agent est téléchargé. L'adresse IP 169.254.169.254 est routable uniquement au sein du VPC (tout le monde peut interroger le service). VPCs Pour plus d'informations sur le service, consultez la section Métadonnées d'instance et données utilisateur. Notez également que les scripts saisis en tant que UserData sont exécutés en tant qu'utilisateur « root » et n'ont pas besoin d'utiliser la commande « sudo ».

Cette procédure pas à pas laisse les paramètres suivants à la valeur par défaut (illustrée) :

- Groupe Auto Scaling :`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`
- Base de données :`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`.
- Candidature :`DeploymentConfigName=CodeDeployDefault.OneAtATime`.

Paramètres variables :

La console fournit une option ASAP pour l'heure de début et cette procédure pas à pas recommande de l'utiliser. ASAP entraîne l'exécution de la RFC dès que les approbations sont passées.

> **Note**
>
> Il existe de nombreux paramètres que vous pouvez choisir de définir différemment de ceux illustrés. Les valeurs des paramètres présentés dans l'exemple ont été testées, mais elles ne vous conviennent peut-être pas. Seules les valeurs obligatoires sont indiquées dans les exemples. Les valeurs de *`replaceable`* police doivent être modifiées car elles sont propres à votre compte.

Création de l'infrastructure

Cette procédure utilise le Stack CT à deux niveaux à haute disponibilité suivi du CT de stockage Create S3.

La collecte des données suivantes avant de commencer accélérera le déploiement.

LES DONNÉES REQUISES ONT UNE PILE :

- AutoScalingGroup:
  - UserData: Cette valeur est fournie dans ce didacticiel. Il inclut des commandes permettant de configurer la ressource pour l' CodeDeploy agent CodeDeploy et de le démarrer.
  - AMI-ID : cette valeur détermine le système d'exploitation des EC2 instances que votre groupe Auto Scaling (ASG) créera. Sélectionnez une AMI dans votre compte qui commence par

« customer- » et qui utilise le système d'exploitation que vous souhaitez. Trouvez l'AMI IDs dans la console AMS VPCs -> page de VPCs détails. Cette procédure pas à pas est destinée aux ASGs personnes configurées pour utiliser une AMI Amazon Linux ou RHEL.

- Base de données :
  - Ces paramètres, DBEngineEngineVersion, et LicenseModeldoivent être définis en fonction de votre situation, bien que les valeurs indiquées dans l'exemple aient été testées. Le didacticiel utilise les valeurs suivantes, respectivement :*MySQL*,*8.0.16*,*general-public-license*.
  - Ces paramètres, DBNameMasterUserPassword, et MasterUsernamesont obligatoires lors du déploiement du bundle d'applications. Le didacticiel utilise les valeurs suivantes, respectivement :*wordpressDB*,*p4ssw0rd*,*admin*. Notez qu'il ne DBName peut contenir que des caractères alphanumériques.
  - Lorsque vous entrez le code MasterUsernamepour la base de données RDS, il apparaît en texte clair. Connectez-vous à la base de données dès que possible et modifiez le mot de passe pour garantir votre sécurité.
  - Pour RDSSubnetles identifiants, utilisez deux sous-réseaux privés. Entrez-les un par un en appuyant sur « Entrée » après chaque. Trouvez un sous-réseau IDs avec la référence For the AMS SKMS API, consultez l'onglet Rapports dans l'opération AWS Artifact Console (CLI list-subnet-summaries :) ou sur la page de détails de la console AMS VPCs -> VPC.
- LoadBalancer:
  - Définissez ce paramètre Public sur true car le didacticiel utilise des sous-réseaux ELB publics.
  - ELBSubnetIdentifiants : utilisez deux sous-réseaux publics. Entrez-les un par un en appuyant sur « Entrée » après chaque. Trouvez un sous-réseau IDs avec la référence For the AMS SKMS API, consultez l'onglet Rapports dans l'opération AWS Artifact Console (CLI list-subnet-summaries :) ou sur la page de détails de la console AMS VPCs -> VPC.
- Application : la ApplicationNamevaleur définit le nom de l' CodeDeploy application et le nom du groupe de CodeDeploy déploiement. Vous l'utilisez pour déployer votre application. Il doit être unique dans le compte. Pour vérifier les CodeDeploy noms de votre compte, consultez la CodeDeploy console. L'exemple utilise *WordPress* mais, si vous voulez utiliser cette valeur, assurez-vous qu'elle n'est pas déjà utilisée.

1. Lancez la pile de haute disponibilité.

   a. Sur la page Create RFC, sélectionnez la catégorie Deployment, la sous-catégorie Standard Stacks, l'article High Availability two-tier stack et operation Create dans la liste.

b. IMPORTANT : Choisissez Avancé et définissez les valeurs comme indiqué.

Il vous suffit de saisir des valeurs pour les options marquées d'un astérisque (*). Les valeurs testées sont indiquées dans l'exemple ; vous pouvez laisser les options vides non obligatoires vides vides.

c. Pour la section Description de la RFC :

```
Subject: WP-HA-2-Tier-RFC
```

d. Dans la section Informations sur les ressources, définissez les paramètres de la base de données AutoScalingGroupLoadBalancer, de l'application et des balises.

De plus, le but de la touche de balise AppName « » est de vous permettre de rechercher facilement les instances ASG dans la EC2 console ; vous pouvez appeler cette clé de balise « Nom » ou tout autre nom de clé de votre choix. Notez que vous pouvez ajouter jusqu'à 50 balises.

```
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. Cliquez sur Soumettre lorsque vous avez terminé.

2. Connectez-vous à la base de données que vous avez créée et modifiez le mot de passe.
3. Lancez une pile de compartiments S3.

   La collecte des données suivantes avant de commencer accélérera le déploiement.

   BUCKET S3 DE DONNÉES REQUIS :

   - VPC-ID : cette valeur détermine l'emplacement de votre compartiment S3. Trouvez un VPC à l' IDs aide de la référence pour l'API For the AMS SKMS, consultez l'onglet Rapports dans l'opération AWS Artifact Console (CLI : list-vpc-summaries) ou sur la page de la console AMS. VPCs
   - BucketName: Cette valeur définit le nom du compartiment S3, vous l'utilisez pour télécharger votre bundle d'applications. Il doit être unique dans la région du compte et ne peut pas contenir de majuscules. Il n' BucketName est pas obligatoire d'inclure votre identifiant de compte, mais cela permet d'identifier plus facilement le compartiment ultérieurement. Pour voir quels noms de compartiment S3 existent dans le compte, accédez à la console Amazon S3 de votre compte.

   a. Sur la page Create RFC, sélectionnez la catégorie Deployment, la sous-catégorie Advanced Stack Components, l'élément S3 storage et l'opération Create dans la liste de sélection RFC CT.

   b. Conservez l'option Basic par défaut et définissez les valeurs comme indiqué.

```
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. Cliquez sur Soumettre lorsque vous avez terminé. Le bucket déployé avec ce type de modification permet un read/write accès complet à l'ensemble du compte.

Création, téléchargement et déploiement de l'application

Créez d'abord un bundle d' WordPress applications, puis utilisez-le CodeDeploy CTs pour créer et déployer l'application.

1. Téléchargez WordPress, extrayez les fichiers et créez un. répertoire /scripts.

   Commande Linux :

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows : collez `https://github.com/WordPress/WordPress/archive/master.zip` le fichier dans une fenêtre de navigateur et téléchargez le fichier zip.

   Créez un répertoire temporaire dans lequel assembler le package.

   Linux :

   ```
   mkdir /tmp/WordPress
   ```

   Windows : Créez un répertoire WordPress « », vous utiliserez le chemin du répertoire ultérieurement.

2. Extrayez la WordPress source dans le répertoire WordPress « » et créez un fichier. répertoire / 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 : Accédez au répertoire « WordPress » que vous avez créé et créez-y un répertoire « scripts ».

Si vous êtes dans un environnement Windows, veillez à définir le type de rupture des fichiers de script sur Unix (LF). Dans Notepad ++, il s'agit d'une option en bas à droite de la fenêtre.

3. Créez le fichier CodeDeploy appspec.yml dans le WordPress répertoire (si vous copiez l'exemple, vérifiez l'indentation, chaque espace compte). IMPORTANT : Assurez-vous que le chemin « source » est correct pour copier les WordPress fichiers (dans ce cas, dans votre WordPress répertoire) vers la destination prévue (/var/www/html/WordPress). Dans l'exemple, le fichier appspec.yml se trouve dans le répertoire contenant les WordPress fichiers, donc seul «/» est nécessaire. De plus, même si vous avez utilisé une AMI RHEL pour votre groupe Auto Scaling, laissez la ligne « os : linux » telle quelle. Exemple de fichier appspec.yml :

```
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. Créez des scripts de fichiers bash dans le WordPress . répertoire /scripts.

Commencez `config_wordpress.sh` par créer avec le contenu suivant (si vous préférez, vous pouvez modifier directement le fichier wp-config.php).

> **Note**
>
> Remplacez *DBName* par la valeur indiquée dans la RFC HA Stack (par exemple,`wordpress`).
> Remplacez *DB_MasterUsername* par la `MasterUsername` valeur indiquée dans la RFC HA Stack (par exemple,`admin`).
> Remplacez *DB_MasterUserPassword* par la `MasterUserPassword` valeur indiquée dans la RFC HA Stack (par exemple,`p4ssw0rd`).
> *DB_ENDPOINT*Remplacez-le par le nom DNS du point de terminaison dans les sorties d'exécution de la HA Stack RFC (par exemple,`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Vous pouvez le trouver dans l'GetRfcopération (CLI : get-rfc --rfc-id RFC_ID) ou dans la page de détails de la RFC de la console AMS pour la RFC HA Stack que vous avez précédemment soumise.

```
#!/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. Dans le même répertoire, créez `install_dependencies.sh` avec le contenu suivant :

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> Le protocole HTTPS est installé dans les données utilisateur au lancement afin de permettre aux contrôles de santé de fonctionner dès le départ.

6. Dans le même répertoire, créez `start_server.sh` avec le contenu suivant :

   - Pour les instances Amazon Linux, utilisez ceci :

   ```
   #!/bin/bash
   service httpd start
   ```

   - Pour les instances RHEL, utilisez ceci (les commandes supplémentaires sont des politiques qui autorisent SELINUX à accepter) : 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. Dans le même répertoire, créez `stop_server.sh` avec le contenu suivant :

   ```
   #!/bin/bash
   service httpd stop
   ```

8. Créez le bundle zip.

   Linux :

   ```
   $ cd /tmp/WordPress
   $ zip -r wordpress.zip .
   ```

   Windows : Accédez à votre répertoire WordPress « », sélectionnez tous les fichiers et créez un fichier zip. N'oubliez pas de le nommer wordpress.zip.

1. Téléchargez le bundle d'applications dans le compartiment S3

Le package doit être en place pour continuer à déployer la pile.

Vous avez automatiquement accès à toutes les instances de compartiment S3 que vous créez. Vous pouvez y accéder via vos Bastions (voir Accès aux instances) ou via la console S3, et télécharger le CodeDeploy package avec drag-and-drop, ou en naviguant vers le fichier et en le sélectionnant.

Vous pouvez également utiliser la commande suivante dans une fenêtre shell ; assurez-vous que le chemin d'accès au fichier zip est correct :

```
aws s3 cp wordpress/wordpress.zip s3://BUCKET_NAME/
```

2. Déployer le bundle WordPress CodeDeploy d'applications

DÉPLOIEMENT DE L'APPLICATION DE DÉPLOIEMENT DU CODE DE DONNÉES REQUIS :

- CodeDeployApplicationName: le nom que vous avez donné à l' CodeDeploy application.
- CodeDeployGroupName: Étant donné que l' CodeDeploy application et le groupe ont tous deux été créés à partir du nom que vous avez donné à l' CodeDeploy application dans la RFC de la pile HA, il s'agit du même nom que le CodeDeployApplicationName.
- S3Bucket : nom que vous avez donné au compartiment S3.
- S3 BundleType et S3Key : ils font partie du bundle d' WordPress applications que vous avez déployé.
- VpcId: Le VPC concerné.

a. Sur la page Créer une RFC, sélectionnez la catégorie Déploiement, la sous-catégorie Applications, l'article CodeDeploy application et l'opération Deploy dans la liste de sélection RFC CT.

b. Conservez l'option Basic par défaut et définissez les valeurs comme indiqué.

> **Note**
>
> Référencez l' CodeDeploy application, CodeDeploy le groupe de déploiement, le compartiment S3 et le bundle créés précédemment.

```
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. Cliquez sur Soumettre lorsque vous avez terminé.

Valider le déploiement de l'application

Accédez au point de terminaison (LoadBalancerCName) de l'équilibreur de charge créé précédemment, avec le chemin WordPress déployé :/. WordPress Par exemple :

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Vous devriez voir une page comme celle-ci :

Arrêtez le déploiement de la haute disponibilité

Pour réduire le déploiement, vous soumettez le Delete Stack CT à la pile HA à deux niveaux et au compartiment S3, et vous pouvez demander la suppression des instantanés RDS (ils sont supprimés automatiquement au bout de dix jours, mais ils coûtent peu cher pendant leur séjour). Rassemblez la pile IDs pour la pile HA et le compartiment S3, puis procédez comme suit. Voir Stack | Delete.

Tutoriel sur console : déploiement d'un WordPress site Web Tier and Tie

Cette section décrit comment déployer un WordPress site haute disponibilité (HA) dans un environnement AMS à l'aide de la console AMS. Cet ensemble d'instructions inclut un exemple de création du fichier de package WordPress CodeDeploy compatible nécessaire (par exemple, zip). L'approvisionnement des ressources suit un ordre qui vous permet de les lier entre elles pour former des « niveaux ».

> **Note**
>
> Cette procédure de déploiement est conçue pour être utilisée avec un système d'exploitation Linux AMZN.
> Les paramètres variables essentiels sont notés comme suit *`replaceable`* ; toutefois, vous souhaiterez peut-être modifier d'autres paramètres en fonction de votre situation.

Résumé des tâches et des exigences RFCs :

1. Créez l'infrastructure :
   a. Création d'un cluster de bases de données MySQL RDS
   b. Créer un équilibreur de charge
   c. Créez un groupe Auto Scaling et associez-le à l'équilibreur de charge
   d. Création d'un compartiment S3 pour les CodeDeploy applications
2. Création d'un bundle d' WordPress applications (ne nécessite pas de RFC)
3. Déployez le bundle WordPress d'applications avec CodeDeploy :
   a. Création d'une CodeDeploy application
   b. Création d'un groupe CodeDeploy de déploiement
   c. Téléchargez votre bundle WordPress d'applications dans le compartiment S3 (aucune RFC n'est requise)
   d. Déployez l' CodeDeploy application
4. Valider le déploiement
5. Détruire le déploiement

Les descriptions de toutes les options de tomodensitométrie, y compris, se ChangeTypeId trouvent dans le manuel AMS Change Type Reference.

Création d'une RFC à l'aide de la console (Notions de base)

Voici quelques étapes que vous devez suivre chaque fois que vous créez une RFC à l'aide de la console.

1. Choisissez RFCsdans le volet de navigation de gauche pour ouvrir la page de RFCs liste, puis choisissez Create RFC.

   La page Créer une RFC s'ouvre.
2. Choisissez Parcourir les types de modifications (par défaut) ou Choisir par catégorie.
3. Parcourez les types de modifications :

   a. Choisissez une option de création rapide pour commencer une RFC avec l'un des types de modification les plus utilisés.

      La zone de configuration générale pour ce type de modification s'ouvre, la ligne d'objet est renseignée. Pour voir les détails du type de modification, ouvrez la zone en haut de la page.

   b. Utilisez la zone Tous les types de modifications.

      Filtrez, passez d'une carte à une vue sous forme de tableau, ou triez les types de modifications. Lorsque vous avez trouvé celui que vous recherchez, sélectionnez-le et choisissez Create RFC en haut de la page.

      La zone de configuration générale pour ce type de modification s'ouvre, la ligne d'objet est renseignée. Pour voir les détails du type de modification, ouvrez la zone en haut de la page.
4. Choisissez par catégorie :

   a. Sélectionnez la catégorie, la sous-catégorie, l'article et l'opération appropriés.

      La zone de détails du type de modification apparaît au bas de la page.

   b. Choisissez Create RFC au bas de la page.

   c. La zone de configuration générale pour ce type de modification s'ouvre, la ligne d'objet est renseignée. Pour voir les détails du type de modification, ouvrez la zone en haut de la page.

5. Pour que certaines personnes soient informées de la progression de la RFC, saisissez les adresses e-mail. Pour ajouter des détails sur le type de modification, renseignez la description. Ouvrez la zone de configuration supplémentaire pour ajouter plus de détails sur le RFC.
6. Pour la planification, sélectionnez Exécuter cette modification dès que possible ou Planifier cette modification. Si vous sélectionnez Exécuter cette modification dès que possible, votre RFC s'exécute dès que les approbations sont passées. Si vous sélectionnez Planifier ce type de modification, un calendrier, une heure et un fuseau horaire de sélection s'affichent et votre RFC démarre, après la soumission, comme prévu.
7. Dans la zone de configuration de l'exécution, configurez les paramètres du type de modification. Pour voir les paramètres facultatifs, ouvrez la zone Configuration supplémentaire.
8. Lorsque vous êtes prêt, choisissez Exécuter.

Création de l'infrastructure

Connectez-vous à la console AWS pour le compte AMS cible, puis à la console AMS pour le compte.

Les procédures suivantes décrivent la création d'une base de données RDS, d'un équilibreur de charge et d'un groupe Auto Scaling de manière à ce que vous utilisiez la ressource IDs pour créer l'infrastructure.

Création d'une pile RDS

Voir RDS stack | Create.

Création d'une pile ELB

Lancez un ELB public.

DONNÉES REQUISES :

- `VpcId`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.
- `ELBSubnetIds`: ensemble de sous-réseaux sur lesquels l'équilibreur de charge distribuera le trafic. Choisissez des sous-réseaux publics ou privés. Trouvez un sous-réseau IDs avec la référence For the AMS SKMS API, consultez l'onglet Rapports dans l'opération AWS Artifact Console (CLI list-subnet-summaries :) ou sur la page de détails de la console AMS VPCs -> VPC.
- `VpcId`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.

1. Sur la page Create RFC, sélectionnez la catégorie Deployment, la sous-catégorie Advanced Stack Components, l'élément Load Balancer (ELB) stack, puis cliquez sur Create. Choisissez Avancé et acceptez toutes les valeurs par défaut (y compris celles sans valeur) à l'exception de celles indiquées ci-dessous.

```
Subject:                        WP-ELB-RFC
ELBSubnetIds:                   PUBLIC_AZ1
                              PUBLIC_AZ2
ELBScheme                       true
ELBCookieExpirationPeriod       600
VpcId:                          VPC_ID
Name:                           WP-Public-ELB
```

2. Cliquez sur Soumettre lorsque vous avez terminé.

Création d'une pile de groupes Auto Scaling

Lancez un groupe Auto Scaling.

DONNÉES REQUISES :

- `VpcId`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.
- `AMI-ID`: Cette valeur détermine le type d' EC2 instances que votre groupe Auto Scaling (ASG) va créer. Assurez-vous de sélectionner une AMI dans votre compte qui commence par « customer- » et qui utilise le système d'exploitation que vous souhaitez. Trouvez l'AMI à l' IDs aide de la référence For the AMS SKMS API, consultez l'onglet Rapports de l'opération AWS Artifact Console (CLI : list-amis) ou sur la page de détails de la console AMS ->. VPCs VPCs Cette procédure pas à pas est destinée aux ASGs personnes configurées pour utiliser une AMI Linux.
- `ASGLoadBalancerNames`: L'équilibreur de charge que vous avez créé précédemment. Trouvez le nom en consultant la EC2 console -> équilibreurs de charge (dans le menu de navigation de gauche). Notez qu'il ne s'agit pas du « nom » que vous avez spécifié lors de la création de l'ELB précédemment.

1. Sur la page Create RFC, sélectionnez la catégorie Deployment, la sous-catégorie Advanced Stack Components, l'élément Auto Scaling group, puis cliquez sur Créer. Choisissez Avancé et acceptez toutes les valeurs par défaut (y compris celles sans valeur) à l'exception de celles indiquées ci-dessous.

> **Note**
>
> Spécifiez l'AMI AMS la plus récente. Spécifiez l'ELB créé précédemment.

```
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. Cliquez sur Soumettre lorsque vous avez terminé.

Création d'une pile S3

Lancez un compartiment S3. Le compartiment S3 est l'endroit où vous téléchargez le bundle d'applications que vous avez créé.

DONNÉES REQUISES :

- `VPC-ID`: Cette valeur détermine l'emplacement de votre compartiment S3. Il doit être identique à celui du VPC utilisé précédemment.

- `AccessControl`: les options de AccessControl liste prédéfinie (ACL) sont`Private`, et`PublicRead`. Pour plus d'informations, consultez Amazon Simple Storage Service Canned ACL.
- `BucketName`: Cette valeur définit le nom du compartiment S3, vous l'utilisez pour télécharger votre bundle d'applications. Il doit être unique dans la région du compte et ne peut pas contenir de majuscules. Il n' BucketName est pas obligatoire d'inclure votre identifiant de compte, mais cela permet d'identifier plus facilement le compartiment ultérieurement. Pour voir quels noms de compartiment S3 existent dans le compte, accédez à la console Amazon S3 de votre compte.

1. Sur la page Create RFC, sélectionnez la catégorie Deployment, la sous-catégorie Advanced Stack Components, l'élément S3 storage, puis cliquez sur Create.

   Vous pouvez laisser l'option de paramètre par défaut sur Basic pour accepter les valeurs par défaut comme décrit. Pour définir différentes valeurs, choisissez Avancé.

   > **Note**
   >
   > Le bucket déployé avec ce type de modification permet un read/write accès complet à l'ensemble du compte. De nouveaux types de modifications peuvent être nécessaires pour autoriser des autorisations d'accès plus restreintes.

   ```
   Subject:            S3-Bucket-RFC
   BucketName:         ACCOUNT_ID-codedeploy-bundles
   AccessControl:      Private

   VpcId:              VPC_ID
   Name:               S3BucketForWP
   ```

2. Cliquez sur Soumettre lorsque vous avez terminé.

Création d'un WordPress CodeDeploy bundle

La section fournit un exemple de création d'un bundle de déploiement d'applications.

1. Téléchargez WordPress, extrayez les fichiers et créez un. répertoire /scripts.

   Commande Linux :

```
wget https://github.com/WordPress/WordPress/archive/master.zip
```

Windows : collez `https://github.com/WordPress/WordPress/archive/master.zip` le fichier dans une fenêtre de navigateur et téléchargez le fichier zip.

Créez un répertoire temporaire dans lequel assembler le package.

Linux :

```
mkdir /tmp/WordPress
```

Windows : Créez un répertoire WordPress « », vous utiliserez le chemin du répertoire ultérieurement.

2. Extrayez la WordPress source dans le répertoire WordPress « » et créez un fichier. répertoire / 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 : Accédez au répertoire « WordPress » que vous avez créé et créez-y un répertoire « scripts ».

   Si vous êtes dans un environnement Windows, veillez à définir le type de rupture des fichiers de script sur Unix (LF). Dans Notepad ++, il s'agit d'une option en bas à droite de la fenêtre.

3. Créez le fichier CodeDeploy appspec.yml dans le WordPress répertoire (si vous copiez l'exemple, vérifiez l'indentation, chaque espace compte). IMPORTANT : Assurez-vous que le chemin « source » est correct pour copier les WordPress fichiers (dans ce cas, dans votre WordPress répertoire) vers la destination prévue (/var/www/html/WordPress). Dans l'exemple, le fichier appspec.yml se trouve dans le répertoire contenant les WordPress fichiers, donc seul «/» est nécessaire. De plus, même si vous avez utilisé une AMI RHEL pour votre groupe Auto Scaling, laissez la ligne « os : linux » telle quelle. Exemple de fichier appspec.yml :

```
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. Créez des scripts de fichiers bash dans le WordPress . répertoire /scripts.

   Commencez `config_wordpress.sh` par créer avec le contenu suivant (si vous préférez, vous pouvez modifier directement le fichier wp-config.php).

   > **Note**
   >
   > Remplacez *DBName* par la valeur indiquée dans la RFC HA Stack (par exemple,`wordpress`).
   > Remplacez *DB_MasterUsername* par la `MasterUsername` valeur indiquée dans la RFC HA Stack (par exemple,`admin`).
   > Remplacez *DB_MasterUserPassword* par la `MasterUserPassword` valeur indiquée dans la RFC HA Stack (par exemple,`p4ssw0rd`).
   > *DB_ENDPOINT*Remplacez-le par le nom DNS du point de terminaison dans les sorties d'exécution de la HA Stack RFC (par exemple,`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Vous pouvez le trouver dans l'GetRfcopération (CLI :

get-rfc --rfc-id RFC_ID) ou dans la page de détails de la RFC de la console AMS pour la RFC HA Stack que vous avez précédemment soumise.

```
#!/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. Dans le même répertoire, créez `install_dependencies.sh` avec le contenu suivant :

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> Le protocole HTTPS est installé dans les données utilisateur au lancement afin de permettre aux bilans de santé de fonctionner dès le départ.

6. Dans le même répertoire, créez `start_server.sh` avec le contenu suivant :

   - Pour les instances Amazon Linux, utilisez ceci :

     ```
     #!/bin/bash
     service httpd start
     ```

   - Pour les instances RHEL, utilisez ceci (les commandes supplémentaires sont des politiques qui autorisent SELINUX à accepter) : 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. Dans le même répertoire, créez `stop_server.sh` avec le contenu suivant :

```
#!/bin/bash
service httpd stop
```

8. Créez le bundle zip.

   Linux :

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows : Accédez à votre répertoire WordPress « », sélectionnez tous les fichiers et créez un fichier zip. N'oubliez pas de le nommer wordpress.zip.

Déployez le bundle WordPress d'applications avec CodeDeploy

CodeDeploy Il s'agit d'un service de déploiement AWS qui automatise les déploiements d'applications sur les instances Amazon EC2 . Cette partie du processus implique la création d'une CodeDeploy application, la création d'un groupe de CodeDeploy déploiement, puis le déploiement de l'application à l'aide de CodeDeploy.

Création d'une CodeDeploy application

L' CodeDeploy application est simplement un nom ou un conteneur utilisé par AWS CodeDeploy pour garantir que la révision, la configuration de déploiement et le groupe de déploiement appropriés sont référencés lors d'un déploiement. La configuration de déploiement, dans ce cas, est le WordPress bundle que vous avez créé précédemment.

DONNÉES REQUISES :

- `VpcId`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.
- `CodeDeployApplicationName`: Doit être unique dans le compte. Consultez la CodeDeploy console pour vérifier les noms d'applications existants.

1. Créez l' CodeDeploy application pour WordPress

   Sur la page Créer une RFC, sélectionnez la catégorie Déploiement, la sous-catégorie Applications, l'élément CodeDeploy application et l'opération Créer dans la liste de sélection RFC CT. Choisissez Basic et définissez les valeurs comme indiqué. Cliquez sur Soumettre lorsque vous avez terminé.

   ```
   Subject:                    CD-WP-App-RFC
   CodeDeployApplicationName:  WordPress
   VpcId:                      VPC_ID
   Name:                       WP-CD-App
   ```

2. Cliquez sur Soumettre lorsque vous avez terminé.

Création d'un groupe CodeDeploy de déploiement

Créez le groupe CodeDeploy de déploiement.

Un groupe de CodeDeploy déploiement définit un ensemble d'instances individuelles ciblées pour un déploiement.

DONNÉES REQUISES :

- `VpcId`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.
- `CodeDeployApplicationName`: utilisez la valeur que vous avez créée précédemment.
- `CodeDeployAutoScalingGroups`: utilisez le nom du groupe Auto Scaling que vous avez créé précédemment.
- `CodeDeployDeploymentGroupName`: nom du groupe de déploiement. Ce nom doit être unique pour chaque application associée au groupe de déploiement.
- `CodeDeployServiceRoleArn`: Utilisez la formule donnée dans l'exemple.

1. Sur la page Créer une RFC, sélectionnez la catégorie Déploiement, la sous-catégorie Applications, le groupe de CodeDeploy déploiement d'éléments et l'opération Créer dans la liste de sélection RFC CT. Choisissez Avancé et définissez les valeurs comme indiqué (seul un objet est nécessaire pour la RFC). Cliquez sur Soumettre lorsque vous avez terminé.

> **Note**
>
> Référencez l'ARN du rôle de CodeDeploy service dans ce format `"arn:aws:iam::085398962942:role/aws-codedeploy-role"` et utilisez le nom du groupe Auto Scaling créé précédemment pour « 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. Cliquez sur Soumettre lorsque vous avez terminé.

Téléchargez l' WordPress application

Vous avez automatiquement accès à toutes les instances de compartiment S3 que vous créez. Vous pouvez y accéder via vos Bastions (voir Accès aux instances) ou via la console S3, et télécharger le CodeDeploy bundle. Le bundle doit être en place pour continuer à déployer la pile. L'exemple utilise le nom du bucket créé précédemment.

Vous pouvez utiliser cette commande AWS pour compresser le bundle :

```
aws s3 cp wordpress/wordpress.zip s3://ACCOUNT_ID-codedeploy-bundles/
```

Déployez WordPress l'application avec CodeDeploy

Déployez CodeDeploy l'application.

DONNÉES REQUISES :

- `VPC-ID`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.
- `CodeDeployApplicationName`: utilisez le nom de l' CodeDeploy application que vous avez créée précédemment.

- `CodeDeployDeploymentGroupName`: utilisez le nom du groupe de CodeDeploy déploiement que vous avez créé précédemment.
- `S3Location`(où vous avez téléchargé le bundle d'applications) `S3Bucket` : le bundle BucketName que vous avez créé précédemment, `S3BundleType` et `S3Key` : le type et le nom du bundle que vous avez mis sur votre boutique S3.

1. Déployer le bundle WordPress CodeDeploy d'applications

   Sur la page Créer une RFC, sélectionnez la catégorie Déploiement, la sous-catégorie Applications, l'article CodeDeploy application et l'opération Deploy dans la liste de sélection RFC CT. Choisissez Basic et définissez les valeurs comme indiqué. Cliquez sur Soumettre lorsque vous avez terminé.

   > **Note**
   >
   > Référencez l' CodeDeploy application, CodeDeploy le groupe de déploiement, le compartiment S3 et le bundle créés précédemment.

   ```
   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. Cliquez sur Soumettre lorsque vous avez terminé.

Valider le déploiement de l'application

Accédez au point de terminaison (ELB CName) de l'équilibreur de charge créé précédemment, avec le WordPress chemin déployé :/. WordPress Exemples :

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Démanteler le déploiement des applications

Pour réduire le déploiement, vous soumettez le Delete Stack CT à la pile de base de données RDS, à l'équilibreur de charge de l'application, au groupe Auto Scaling, au compartiment S3, ainsi qu'à l'application et au groupe Code Deploy, soit six en tout. RFCs En outre, vous pouvez soumettre une demande de service pour que les instantanés RDS soient supprimés (ils sont supprimés automatiquement au bout de dix jours, mais ils coûtent peu cher pendant leur séjour). Rassemblez la pile IDs pour tous, puis suivez ces étapes. Voir Stack | Delete.

Tutoriel CLI : Stack à deux niveaux de haute disponibilité (Linux/RHEL)

Cette section décrit comment déployer une pile à deux niveaux de haute disponibilité (HA) dans un environnement AMS à l'aide de la CLI AMS.

> **Note**
>
> Cette procédure de déploiement a été testée dans les environnements AMZN Linux et RHEL.

Résumé des tâches et des exigences RFCs :

1. Création d'une infrastructure (pile HA à deux niveaux)
2. Création d'un compartiment S3 pour les CodeDeploy applications
3. Créez le bundle WordPress d'applications et chargez-le dans le compartiment S3
4. Déployez l'application avec CodeDeploy
5. Accédez au WordPress site et connectez-vous pour valider le déploiement

Avant de commencer

Deployment | Advanced Stack Components | High Availability Two Tier Stack Advanced | Create CT crée un groupe Auto Scaling, un équilibreur de charge, une base de données, ainsi qu'un nom d' CodeDeploy application et un groupe de déploiement (avec le même nom que celui que vous donnez à l'application). Pour plus d'informations, CodeDeploy voir Qu'est-ce que c'est CodeDeploy ?

Cette procédure pas à pas utilise une RFC de haute disponibilité à deux niveaux (avancée) qui inclut UserData et décrit également comment créer un WordPress bundle pouvant CodeDeploy être déployé.

L'exemple `UserData` illustré permet d'obtenir les métadonnées d'instance telles que l'ID d'instance, la région, etc., à partir d'une instance en cours d'exécution en interrogeant le service de métadonnées d' EC2 instance disponible sur http://169.254.169.254/latest/meta-data/. Cette ligne du script de données utilisateur :`REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed 's/[a-z]$//')`, extrait le nom de la zone de disponibilité du service de métadonnées dans la variable $REGION pour nos régions prises en charge, et l'utilise pour compléter l'URL du compartiment S3 dans lequel l' CodeDeploy agent est téléchargé. L'IP 169.254.169.254 est routable uniquement au sein du VPC (tout le monde peut interroger le service). VPCs Pour plus d'informations sur le service, consultez la section Métadonnées d'instance et données utilisateur. Notez également que les scripts saisis UserData sont exécutés en tant qu'utilisateur « root » et n'ont pas besoin d'utiliser la commande « sudo ».

Cette procédure pas à pas laisse les paramètres suivants à la valeur par défaut (illustrée) :

- Groupe Auto Scaling :`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`
- Base de données :`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`.
- Application :`DeploymentConfigName=CodeDeployDefault.OneAtATime`.
- Seau S3 :`AccessControl=Private`.

PARAMÈTRES SUPPLÉMENTAIRES :

`RequestedStartTime`et `RequestedEndTime` si vous souhaitez planifier votre RFC : vous pouvez utiliser Time.is pour déterminer l'heure UTC correcte. Les exemples fournis doivent être adaptés de manière appropriée. Une RFC ne peut pas continuer si l'heure de début est dépassée. Vous pouvez également omettre ces valeurs pour créer une RFC ASAP qui s'exécute dès que les approbations sont passées.

> **Note**
>
> Il existe de nombreux paramètres que vous pouvez choisir de définir différemment de ceux illustrés. Les valeurs des paramètres présentés dans l'exemple ont été testées mais ne vous conviennent peut-être pas.

Création de l'infrastructure

La collecte des données suivantes avant de commencer accélérera le déploiement.

LES DONNÉES REQUISES ONT UNE PILE :

- AutoScalingGroup:
  - `UserData`: Cette valeur est fournie dans ce didacticiel. Il inclut des commandes permettant de configurer la ressource pour l' CodeDeploy agent CodeDeploy et de le démarrer.
  - `AMI-ID`: Cette valeur détermine le type d' EC2 instances que votre groupe Auto Scaling (ASG) va créer. Assurez-vous de sélectionner une AMI dans votre compte qui commence par « customer- » et qui utilise le système d'exploitation que vous souhaitez. Trouvez l'AMI à l' IDs aide de la référence For the AMS SKMS API, consultez l'onglet Rapports de l'opération AWS Artifact Console (CLI : list-amis) ou sur la page de détails de la console AMS ->. VPCs VPCs Cette procédure pas à pas est destinée aux ASGs personnes configurées pour utiliser une AMI Linux.
- Base de données :
  - Ces paramètres, `DBEngineEngineVersion`, et `LicenseModel` doivent être définis en fonction de votre situation, bien que les valeurs indiquées dans l'exemple aient été testées.
  - Ces paramètres,`RDSSubnetIds`, `DBNameMasterUsername`, et `MasterUserPassword` sont obligatoires lors du déploiement du bundle d'applications. Pour RDSSubnet les identifiants, utilisez deux sous-réseaux privés.

- LoadBalancer:
  - Ces paramètres, `DBEngineEngineVersion`, et `LicenseModel` doivent être définis en fonction de votre situation, bien que les valeurs indiquées dans l'exemple aient été testées.
  - `ELBSubnetIds`: utilisez deux sous-réseaux publics.
- Application : la `ApplicationName` valeur définit le nom de l' CodeDeploy application et le nom du groupe de CodeDeploy déploiement. Vous l'utilisez pour déployer votre application. Il doit être unique dans le compte. Pour vérifier les CodeDeploy noms de votre compte, consultez la CodeDeploy console. L'exemple utilise « WordPress » mais, si vous voulez utiliser cette valeur, assurez-vous qu'elle n'est pas déjà utilisée.

Cette procédure utilise le CT à deux niveaux (avancé) à haute disponibilité (ct-06mjngx5flwto) et le CT Create S3 storage (ct-1a68ck03fn98r). À partir de votre compte authentifié, suivez ces étapes sur la ligne de commande.

1. Lancez la pile d'infrastructure.

   a. Exportez le schéma JSON des paramètres d'exécution pour la pile HA à deux niveaux CT dans un fichier de votre dossier actuel nommé CreateStackParams .json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-06mjngx5flwto"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateStackParams.json
   ```

   b. Modifiez le schéma. Remplacez-le *`variables`* comme il convient. Par exemple, utilisez le système d'exploitation que vous souhaitez pour les EC2 instances que l'ASG créera. Enregistrez-le `ApplicationName` tel que vous l'utiliserez ultérieurement pour déployer l'application. Notez que vous pouvez ajouter jusqu'à 50 balises.

   ```
   {
   "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. Exportez le modèle CreateRfc JSON dans un fichier de votre dossier actuel nommé CreateStackRfc .json :

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

d. Modifiez le modèle RFC comme suit et enregistrez-le, vous pouvez supprimer et remplacer le contenu. Notez que `RequestedStartTime` et `RequestedEndTime` sont désormais facultatifs ; leur exclusion crée une RFC ASAP qui s'exécute dès qu'elle est approuvée (ce qui se produit généralement automatiquement). Pour soumettre une RFC planifiée, ajoutez ces valeurs.

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-06mjngx5flwto",
"Title":                "HA-Stack-For-WP-RFC"
}
```

e. Créez la RFC en spécifiant le fichier CreateStackRfc .json et le fichier de paramètres d'exécution CreateStackParams .json :

```
aws amscm create-rfc --cli-input-json file://CreateStackRfc.json --execution-
parameters file://CreateStackParams.json
```

Vous recevez l'identifiant RFC dans la réponse. Enregistrez l'identifiant pour les étapes suivantes.

f. Soumettez le RFC :

```
aws amscm submit-rfc --rfc-id  RFC_ID
```

Si la RFC réussit, vous ne recevrez aucune sortie.

g. Pour vérifier l'état de la RFC, exécutez

```
aws amscm get-rfc --rfc-id RFC_ID
```

Notez l'identifiant RFC.

2. Lancer un compartiment S3

La collecte des données suivantes avant de commencer accélérera le déploiement.

BUCKET S3 DE DONNÉES REQUIS :

- `VPC-ID`: Cette valeur détermine l'emplacement de votre compartiment S3. Utilisez le même ID VPC que celui que vous avez utilisé précédemment.
- `BucketName`: Cette valeur définit le nom du compartiment S3, vous l'utilisez pour télécharger votre bundle d'applications. Il doit être unique dans la région du compte et ne peut pas contenir de majuscules. Il n' BucketName est pas obligatoire d'inclure votre identifiant de compte, mais cela permet d'identifier plus facilement le compartiment ultérieurement. Pour voir quels

noms de compartiment S3 existent dans le compte, accédez à la console Amazon S3 de votre compte.

a. Exportez le schéma JSON des paramètres d'exécution pour le stockage S3 create CT dans un fichier JSON nommé CreateS3 StoreParams .json.

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateS3StoreParams.json
```

b. Modifiez le schéma comme suit, vous pouvez supprimer et remplacer le contenu. Remplacez *VPC_ID* de manière appropriée. Les valeurs de l'exemple ont été testées, mais elles ne vous conviennent peut-être pas.

> **Tip**
>
> Ils `BucketName` doivent être uniques dans la région du compte et ne peuvent pas inclure de majuscules. Il n' BucketName est pas obligatoire d'inclure votre identifiant de compte, mais cela permet d'identifier plus facilement le compartiment ultérieurement. Pour voir quels noms de compartiment S3 existent dans le compte, accédez à la console Amazon S3 de votre compte.

```
{
"Description":      "S3BucketForWordPressBundle",
"VpcId":            "VPC_ID",
"StackTemplateId":  "stm-s2b72beb000000000",
"Name":             "S3BucketForWP",
"TimeoutInMinutes":  60,
"Parameters":   {
    "AccessControl":    "Private",
    "BucketName":       "ACCOUNT_ID-BUCKET_NAME"
    }
}
```

c. Exportez le modèle JSON CreateRfc pour dans un fichier, dans votre dossier actuel, nommé CreateS3 StoreRfc .json :

```
aws amscm create-rfc --generate-cli-skeleton > CreateS3StoreRfc.json
```

d. Modifiez et enregistrez le fichier CreateS3 StoreRfc .json, vous pouvez supprimer et remplacer le contenu. Notez que `RequestedStartTime` et `RequestedEndTime` sont désormais facultatifs ; leur exclusion crée une RFC ASAP qui s'exécute dès qu'elle est approuvée (ce qui se produit généralement automatiquement). Pour soumettre une RFC planifiée, ajoutez ces valeurs.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1a68ck03fn98r",
"Title":                "S3-Stack-For-WP-RFC"
}
```

e. Créez la RFC en spécifiant le fichier CreateS3 StoreRfc .json et le fichier de paramètres d'exécution StoreParams CreateS3 .json :

```
aws amscm create-rfc --cli-input-json file://CreateS3StoreRfc.json  --
execution-parameters file://CreateS3StoreParams.json
```

Vous recevez RfcId le nouveau RFC dans la réponse. Enregistrez l'identifiant pour les étapes suivantes.

f. Soumettez le RFC :

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Si la RFC réussit, vous ne recevrez aucune sortie.

g. Pour vérifier l'état de la RFC, exécutez

```
aws amscm get-rfc --rfc-id RFC_ID
```

Création, téléchargement et déploiement de l'application

Créez d'abord un bundle d' WordPress applications, puis utilisez-le CodeDeploy CTs pour créer et déployer l'application.

1. Téléchargez WordPress, extrayez les fichiers et créez un. répertoire /scripts.

   Commande Linux :

```
wget https://github.com/WordPress/WordPress/archive/master.zip
```

Windows : collez `https://github.com/WordPress/WordPress/archive/master.zip` le fichier dans une fenêtre de navigateur et téléchargez le fichier zip.

Créez un répertoire temporaire dans lequel assembler le package.

Linux :

```
mkdir /tmp/WordPress
```

Windows : Créez un répertoire WordPress « », vous utiliserez le chemin du répertoire ultérieurement.

2. Extrayez la WordPress source dans le répertoire WordPress « » et créez un fichier. répertoire / 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 : Accédez au répertoire « WordPress » que vous avez créé et créez-y un répertoire « scripts ».

   Si vous êtes dans un environnement Windows, veillez à définir le type de rupture des fichiers de script sur Unix (LF). Dans Notepad ++, il s'agit d'une option en bas à droite de la fenêtre.

3. Créez le fichier CodeDeploy appspec.yml dans le WordPress répertoire (si vous copiez l'exemple, vérifiez l'indentation, chaque espace compte). IMPORTANT : Assurez-vous que le chemin « source » est correct pour copier les WordPress fichiers (dans ce cas, dans votre WordPress répertoire) vers la destination prévue (/var/www/html/WordPress). Dans l'exemple, le fichier appspec.yml se trouve dans le répertoire contenant les WordPress fichiers, donc seul «/» est nécessaire. De plus, même si vous avez utilisé une AMI RHEL pour votre groupe Auto Scaling, laissez la ligne « os : linux » telle quelle. Exemple de fichier appspec.yml :

```
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. Créez des scripts de fichiers bash dans le WordPress . répertoire /scripts.

   Commencez `config_wordpress.sh` par créer avec le contenu suivant (si vous préférez, vous pouvez modifier directement le fichier wp-config.php).

   > **Note**
   >
   > Remplacez *`DBName`* par la valeur indiquée dans la RFC HA Stack (par exemple,`wordpress`).
   > Remplacez *`DB_MasterUsername`* par la `MasterUsername` valeur indiquée dans la RFC HA Stack (par exemple,`admin`).
   > Remplacez *`DB_MasterUserPassword`* par la `MasterUserPassword` valeur indiquée dans la RFC HA Stack (par exemple,`p4ssw0rd`).
   > *`DB_ENDPOINT`*Remplacez-le par le nom DNS du point de terminaison dans les sorties d'exécution de la HA Stack RFC (par exemple,`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Vous pouvez le trouver dans l'GetRfcopération (CLI :

get-rfc --rfc-id RFC_ID) ou dans la page de détails de la RFC de la console AMS pour la RFC HA Stack que vous avez précédemment soumise.

```
#!/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. Dans le même répertoire, créez `install_dependencies.sh` avec le contenu suivant :

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> Le protocole HTTPS est installé dans les données utilisateur au lancement afin de permettre aux contrôles de santé de fonctionner dès le départ.

6. Dans le même répertoire, créez `start_server.sh` avec le contenu suivant :

   - Pour les instances Amazon Linux, utilisez ceci :

     ```
     #!/bin/bash
     service httpd start
     ```

   - Pour les instances RHEL, utilisez ceci (les commandes supplémentaires sont des politiques qui autorisent SELINUX à accepter) : 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. Dans le même répertoire, créez `stop_server.sh` avec le contenu suivant :

```
#!/bin/bash
service httpd stop
```

8. Créez le bundle zip.

   Linux :

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows : Accédez à votre répertoire WordPress « », sélectionnez tous les fichiers et créez un fichier zip. N'oubliez pas de le nommer wordpress.zip.

1. Téléchargez le bundle d'applications dans le compartiment S3.

   Le bundle doit être en place pour continuer à déployer la pile.

   Vous avez automatiquement accès à toutes les instances de compartiment S3 que vous créez. Vous pouvez y accéder via vos bastions ou via la console S3, et télécharger le WordPress bundle avec le fichier zip drag-and-drop ou en le recherchant et en le sélectionnant.

   Vous pouvez également utiliser la commande suivante dans une fenêtre shell ; assurez-vous que le chemin d'accès au fichier zip est correct :

```
aws s3 cp wordpress.zip s3://BUCKET_NAME/
```

2. Déployez le bundle WordPress d'applications.

   La collecte des données suivantes avant de commencer accélérera le déploiement.

   DONNÉES REQUISES :

   - `VPC-ID`: Cette valeur détermine l'emplacement de votre compartiment S3. Utilisez le même ID VPC que celui que vous avez utilisé précédemment.

- `CodeDeployApplicationNameet CodeDeployApplicationName` : La `ApplicationName` valeur que vous avez utilisée dans la RFC HA 2-Tier Stack définit le CodeDeployApplicationName et le. CodeDeployDeploymentGroupName L'exemple utilise « WordPress », mais vous avez peut-être utilisé une valeur différente.
- `S3Location`: Pour`S3Bucket`, utilisez celui `BucketName` que vous avez créé précédemment. Les `S3BundleType` et `S3Key` proviennent du bundle que vous avez ajouté à votre boutique S3.

a. Exportez le schéma JSON des paramètres d'exécution pour le déploiement de CodeDeploy l'application CT dans un fichier JSON nommé Deploy CDApp Params.json.

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

b. Modifiez le schéma comme suit et enregistrez-le sous le nom, vous pouvez supprimer et remplacer le contenu.

```
{
"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. Exportez le modèle JSON CreateRfc pour dans un fichier, dans votre dossier actuel, nommé Deploy CDApp RFC.json :

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

d. Modifiez et enregistrez le fichier Deploy CDApp RFC.json, vous pouvez supprimer et remplacer le contenu. Notez que `RequestedStartTime` et `RequestedEndTime` sont désormais facultatifs ; leur exclusion crée une RFC ASAP qui s'exécute dès qu'elle est approuvée (ce qui se produit généralement automatiquement). Pour soumettre une RFC planifiée, ajoutez ces valeurs.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-WP-RFC"
}
```

e. Créez le RFC en spécifiant le fichier Deploy CDApp Rfc et le fichier de paramètres d'exécution de Deploy CDApp Params :

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

Vous recevez RfcId le nouveau RFC dans la réponse. Enregistrez l'identifiant pour les étapes suivantes.

f. Soumettez le RFC :

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Si la RFC réussit, vous ne recevrez aucune sortie.

g. Pour vérifier l'état de la RFC, exécutez

```
aws amscm get-rfc --rfc-id RFC_ID
```

Valider le déploiement de l'application

Accédez au point de terminaison (ELB CName) de l'équilibreur de charge créé précédemment, avec le WordPress chemin déployé :/. WordPress Exemples :

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Démanteler le déploiement de l'application

Une fois que vous aurez terminé le didacticiel, vous souhaiterez réduire le déploiement afin de ne pas avoir à vous facturer les ressources.

Voici une opération générique de suppression de pile. Vous devez le soumettre deux fois, une fois pour la pile HA à 2 niveaux et une fois pour la pile de compartiments S3. Enfin, soumettez une demande de service demandant que tous les instantanés du compartiment S3 (incluez l'ID de pile du compartiment S3 dans la demande de service) soient supprimés. Ils sont automatiquement supprimés au bout de 10 jours, mais leur suppression anticipée permet d'économiser un peu d'argent.

Cette procédure pas à pas fournit un exemple d'utilisation de la console AMS pour supprimer une pile S3 ; cette procédure s'applique à la suppression de toute pile à l'aide de la console AMS.

> **Note**
>
> Si vous supprimez un compartiment S3, il doit d'abord être vidé de ses objets.

DONNÉES REQUISES :

- `StackId`: La pile à utiliser. Vous pouvez le trouver en consultant la page AMS Console Stacks, disponible via un lien dans le menu de navigation de gauche. À l'aide de l'API/CLI AMS SKMS, exécutez la référence For the AMS SKMS API, voir l'onglet Rapports dans l'opération AWS Artifact Console (dans l'interface de ligne de commande). `list-stack-summaries`
- L'identifiant du type de modification pour cette procédure pas à pas est « 1.0 ». Pour connaître la dernière version, exécutez cette commande : `ct-0q0bic0ywqk6c`

  ```
  aws amscm list-change-type-version-summaries --filter
   Attribute=ChangeTypeId,Value=ct-0q0bic0ywqk6c
  ```

CRÉATION EN LIGNE :

- Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne). E

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0"
 --title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

- Soumettez le RFC à l'aide de l'ID RFC renvoyé lors de l'opération de création du RFC. Jusqu'à ce qu'elle soit soumise, la RFC reste en l'`Editing`état et ne fait l'objet d'aucune action.

```
aws amscm submit-rfc --rfc-id RFC_ID
```

- Surveillez l'état de la RFC et visualisez le résultat de l'exécution :

```
aws amscm get-rfc --rfc-id RFC_ID
```

CRÉATION D'UN MODÈLE :

1. Exportez le modèle RFC dans un fichier de votre dossier actuel ; l'exemple le nomme DeleteStackRfc .json :

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

2. Modifiez et enregistrez le fichier DeleteStackRfc .json. Comme la suppression d'une pile ne comporte qu'un seul paramètre d'exécution, les paramètres d'exécution peuvent se trouver dans le fichier DeleteStackRfc .json lui-même (il n'est pas nécessaire de créer un fichier JSON distinct avec les paramètres d'exécution).

   Les guillemets internes de l'extension ExecutionParameters JSON doivent être masqués par une barre oblique inverse (\). Exemple sans heure de début et de fin :

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0q0bic0ywqk6c",
   "Title":                "Delete-My-Stack-RFC"
   "ExecutionParameters":  "{
           \"StackId\":\"STACK_ID\"}"
   }
   ```

3. Créez le RFC :

   ```
   aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
   ```

Vous recevez le code RfcId de la nouvelle RFC dans la réponse. Exemples :

```
{
"RfcId": "daaa1867-ffc5-1473-192a-842f6b326102"
}
```

Enregistrez l'identifiant pour les étapes suivantes.

4. Soumettez le RFC :

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Si le RFC réussit, vous ne recevez aucune confirmation sur la ligne de commande.

5. Pour surveiller l'état de la demande et consulter le résultat de l'exécution, procédez comme suit :

```
aws amscm get-rfc --rfc-id RFC_ID --query "Rfc.
{Status:Status.Name,Exec:ExecutionOutput}" --output table
```

Tutoriel CLI : Déploiement d'un WordPress site Web Tier and Tie

Cette section décrit comment déployer un WordPress site haute disponibilité (HA) dans un environnement AMS à l'aide de la CLI AMS. Cet ensemble d'instructions inclut un exemple de création du fichier de package WordPress CodeDeploy compatible nécessaire (par exemple, zip).

> **Note**
>
> Cette procédure de déploiement est conçue pour être utilisée avec un environnement Linux AMZN.
> Les paramètres variables essentiels sont notés comme suit *replaceable* ; toutefois, vous souhaiterez peut-être modifier d'autres paramètres en fonction de votre situation.

Résumé des tâches et des exigences RFCs :

1. Créez l'infrastructure :
   a. Création d'une pile RDS (CLI)
   b. Créer un équilibreur de charge

   c. Créez un groupe Auto Scaling et associez-le à l'équilibreur de charge
   d. Création d'un compartiment S3 pour les CodeDeploy applications
2. Création d'un bundle d' WordPress applications (ne nécessite pas de RFC)
3. Déployez le bundle WordPress d'applications avec CodeDeploy :
   a. Création d'une CodeDeploy application
   b. Création d'un groupe CodeDeploy de déploiement
   c. Téléchargez votre bundle WordPress d'applications dans le compartiment S3 (aucune RFC n'est requise)
   d. Déployez l' CodeDeploy application
4. Valider le déploiement
5. Démanteler le déploiement

Suivez toutes les étapes de la ligne de commande depuis votre compte authentifié.

Création d'une RFC à l'aide de la CLI

Pour des informations détaillées sur la création RFCs, voir Création RFCs ; pour une explication des paramètres RFC courants, voir Paramètres communs RFC.

Création de l'infrastructure

Les procédures suivantes décrivent la création d'une base de données RDS, d'un équilibreur de charge et d'un groupe Auto Scaling de manière à ce que vous utilisiez la ressource IDs pour créer l'infrastructure.

Création d'une pile RDS (CLI)

Voir RDS stack | Create.

Création d'une pile ELB

Lancez un équilibreur de charge public (ELB). Voir Load Balancer (ELB) Stack | Create.

Création d'une pile de groupes Auto Scaling

Lancez un groupe Auto Scaling.

Voir Auto Scaling Group | Create.

Création d'un magasin S3

Lancez un compartiment S3. Le compartiment S3 est l'endroit où vous chargez le bundle d'applications que vous avez créé. Voir S3 Storage | Create.

Créez un bundle WordPress d'applications pour CodeDeploy

Cette section fournit un exemple de création d'un bundle de déploiement d'applications.

1. Téléchargez WordPress, extrayez les fichiers et créez un. répertoire /scripts.

   Commande Linux :

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows : collez `https://github.com/WordPress/WordPress/archive/master.zip` le fichier dans une fenêtre de navigateur et téléchargez le fichier zip.

   Créez un répertoire temporaire dans lequel assembler le package.

   Linux :

   ```
   mkdir /tmp/WordPress
   ```

   Windows : Créez un répertoire WordPress « », vous utiliserez le chemin du répertoire ultérieurement.

2. Extrayez la WordPress source dans le répertoire WordPress « » et créez un fichier. répertoire / 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 : Accédez au répertoire « WordPress » que vous avez créé et créez-y un répertoire « scripts ».

Si vous êtes dans un environnement Windows, veillez à définir le type de rupture des fichiers de script sur Unix (LF). Dans Notepad ++, il s'agit d'une option en bas à droite de la fenêtre.

3. Créez le fichier CodeDeploy appspec.yml dans le WordPress répertoire (si vous copiez l'exemple, vérifiez l'indentation, chaque espace compte). IMPORTANT : Assurez-vous que le chemin « source » est correct pour copier les WordPress fichiers (dans ce cas, dans votre WordPress répertoire) vers la destination prévue (/var/www/html/WordPress). Dans l'exemple, le fichier appspec.yml se trouve dans le répertoire contenant les WordPress fichiers, donc seul «/» est nécessaire. De plus, même si vous avez utilisé une AMI RHEL pour votre groupe Auto Scaling, laissez la ligne « os : linux » telle quelle. Exemple de fichier appspec.yml :

```
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. Créez des scripts de fichiers bash dans le WordPress . répertoire /scripts.

   Commencez `config_wordpress.sh` par créer avec le contenu suivant (si vous préférez, vous pouvez modifier directement le fichier wp-config.php).

> **Note**
>
> Remplacez *DBName* par la valeur indiquée dans la RFC HA Stack (par exemple,`wordpress`).
> Remplacez *DB_MasterUsername* par la `MasterUsername` valeur indiquée dans la RFC HA Stack (par exemple,`admin`).
> Remplacez *DB_MasterUserPassword* par la `MasterUserPassword` valeur indiquée dans la RFC HA Stack (par exemple,`p4ssw0rd`).
> *DB_ENDPOINT*Remplacez-le par le nom DNS du point de terminaison dans les sorties d'exécution de la HA Stack RFC (par exemple,`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Vous pouvez le trouver dans l'GetRfcopération (CLI : get-rfc --rfc-id RFC_ID) ou dans la page de détails de la RFC de la console AMS pour la RFC HA Stack que vous avez précédemment soumise.

```
#!/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. Dans le même répertoire, créez `install_dependencies.sh` avec le contenu suivant :

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> Le protocole HTTPS est installé dans les données utilisateur au lancement afin de permettre aux bilans de santé de fonctionner dès le départ.

6. Dans le même répertoire, créez `start_server.sh` avec le contenu suivant :

   - Pour les instances Amazon Linux, utilisez ceci :

   ```
   #!/bin/bash
   service httpd start
   ```

   - Pour les instances RHEL, utilisez ceci (les commandes supplémentaires sont des politiques qui autorisent SELINUX à accepter) : 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. Dans le même répertoire, créez `stop_server.sh` avec le contenu suivant :

   ```
   #!/bin/bash
   service httpd stop
   ```

8. Créez le bundle zip.

   Linux :

   ```
   $ cd /tmp/WordPress
   $ zip -r wordpress.zip .
   ```

   Windows : Accédez à votre répertoire WordPress « », sélectionnez tous les fichiers et créez un fichier zip. N'oubliez pas de le nommer wordpress.zip.

Déployez le bundle WordPress d'applications avec CodeDeploy

CodeDeploy Il s'agit d'un service de déploiement AWS qui automatise les déploiements d'applications sur les instances Amazon EC2 . Cette partie du processus implique la création d'une CodeDeploy application, la création d'un groupe de CodeDeploy déploiement, puis le déploiement de l'application à l'aide de CodeDeploy.

Création d'une CodeDeploy application

L' CodeDeploy application est simplement un nom ou un conteneur utilisé par AWS CodeDeploy pour garantir que la révision, la configuration de déploiement et le groupe de déploiement appropriés sont référencés lors d'un déploiement. La configuration de déploiement, dans ce cas, est le WordPress bundle que vous avez créé précédemment.

DONNÉES REQUISES :

- `VpcId`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.
- `CodeDeployApplicationName`: Doit être unique dans le compte. Consultez la CodeDeploy console pour vérifier les noms d'applications existants.
- `ChangeTypeId`et `ChangeTypeVersion` : L'ID de type de modification pour cette procédure pas à pas est`ct-0ah3gwb9seqk2`, pour connaître la dernière version, exécutez cette commande :

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0ah3gwb9seqk2
```

1. Exportez le schéma JSON des paramètres d'exécution pour l' CodeDeploy application CT dans un fichier de votre dossier actuel ; l'exemple le nomme Create CDApp Params.json.

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

2. Modifiez et enregistrez le fichier JSON comme suit ; vous pouvez supprimer et remplacer le contenu.

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

3. Exportez le modèle JSON CreateRfc pour un fichier de votre dossier actuel ; l'exemple le nomme Create CDApp RFC.json.

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

4. Modifiez et enregistrez le fichier JSON comme suit ; vous pouvez supprimer et remplacer le contenu. Notez que `RequestedStartTime` et `RequestedEndTime` sont désormais facultatifs ; leur exclusion entraîne l'exécution de la RFC dès qu'elle est approuvée (ce qui se produit généralement automatiquement). Pour soumettre une RFC « planifiée », ajoutez ces valeurs.

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

5. Créez le RFC en spécifiant le fichier Create CDApp Rfc et le fichier de paramètres d'exécution :

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

   Vous recevez l'ID RFC de la nouvelle RFC dans la réponse. Enregistrez l'identifiant pour les étapes suivantes.

6. Soumettez le RFC :

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Si la RFC réussit, vous ne recevrez aucune sortie.

7. Soumettez le RFC :

```
aws amscm get-rfc --rfc-id RFC_ID
```

Création d'un groupe CodeDeploy de déploiement

Créez le groupe CodeDeploy de déploiement.

Un groupe de CodeDeploy déploiement définit un ensemble d'instances individuelles ciblées pour un déploiement.

DONNÉES REQUISES :

- `VpcId`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.
- `CodeDeployApplicationName`: utilisez la valeur que vous avez créée précédemment.
- `CodeDeployAutoScalingGroups`: utilisez le nom du groupe Auto Scaling que vous avez créé précédemment.
- `CodeDeployDeploymentGroupName`: nom du groupe de déploiement. Ce nom doit être unique pour chaque application associée au groupe de déploiement.
- `CodeDeployServiceRoleArn`: Utilisez la formule donnée dans l'exemple.
- `ChangeTypeId`et `ChangeTypeVersion` : L'ID de type de modification pour cette procédure pas à pas est`ct-2gd0u847qd9d2`, pour connaître la dernière version, exécutez cette commande :

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-2gd0u847qd9d2
```

1. Exportez le schéma JSON des paramètres d'exécution dans un fichier de votre dossier actuel ; l'exemple le nomme Create CDDep GroupParams .json.

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

2. Modifiez et enregistrez le fichier JSON comme suit ; vous pouvez supprimer et remplacer le contenu.

```
{
"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. Exportez le modèle JSON CreateRfc pour un fichier de votre dossier actuel ; l'exemple le nomme Create CDDep GroupRfc .json.

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

4. Modifiez et enregistrez le fichier JSON comme suit ; vous pouvez supprimer et remplacer le contenu. Notez que `RequestedStartTime` et `RequestedEndTime` sont désormais facultatifs ; leur exclusion entraîne l'exécution de la RFC dès qu'elle est approuvée (ce qui se produit généralement automatiquement). Pour soumettre une RFC « planifiée », ajoutez ces valeurs.

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

5. Créez la RFC en spécifiant le fichier de création et le CDDep GroupRfc fichier de paramètres d'exécution :

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

   Vous recevez l'ID RFC de la nouvelle RFC dans la réponse. Enregistrez l'identifiant pour les étapes suivantes.

6. Soumettez le RFC :

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Si la RFC réussit, vous ne recevrez aucune sortie.

7. Vérifiez l'état de la RFC :

```
aws amscm get-rfc --rfc-id RFC_ID
```

Téléchargez l' WordPress application

Vous avez automatiquement accès à toutes les instances de compartiment S3 que vous créez. Vous pouvez y accéder via vos Bastions (voir Accès aux instances) ou via la console S3, et télécharger le CodeDeploy bundle. Le bundle doit être en place pour continuer à déployer la pile. L'exemple utilise le nom du bucket créé précédemment.

```
aws s3 cp wordpress/wordpress.zip s3://ACCOUNT_ID-codedeploy-bundles/
```

Déployez WordPress l'application avec CodeDeploy

Déployez CodeDeploy l'application.

Une fois que vous avez votre bundle CodeDeploy d'applications et votre groupe de déploiement, utilisez cette RFC pour déployer l'application.

DONNÉES REQUISES :

- `VPC-ID`: Le VPC que vous utilisez doit être le même que le VPC utilisé précédemment.
- `CodeDeployApplicationName`: utilisez le nom de l' CodeDeploy application que vous avez créée précédemment.
- `CodeDeployDeploymentGroupName`: utilisez le nom du groupe de CodeDeploy déploiement que vous avez créé précédemment.
- `S3Location`(où vous avez téléchargé le bundle d'applications) `S3Bucket` : le bundle BucketName que vous avez créé précédemment, `S3BundleType` et `S3Key` : le type et le nom du bundle que vous avez mis sur votre boutique S3.
- `ChangeTypeId`et `ChangeTypeVersion` : L'ID de type de modification pour cette procédure pas à pas est`ct-2edc3sd1sqmrb`, pour connaître la dernière version, exécutez cette commande :

  ```
  aws amscm list-change-type-version-summaries --filter
   Attribute=ChangeTypeId,Value=ct-2edc3sd1sqmrb
  ```

1. Exportez le schéma JSON des paramètres d'exécution pour le déploiement de l' CodeDeploy application CT dans un fichier de votre dossier actuel ; l'exemple le nomme Deploy CDApp Params.json.

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

2. Modifiez le fichier JSON comme suit ; vous pouvez supprimer et remplacer le contenu. Pour`S3Bucket`, utilisez celui `BucketName` que vous avez créé précédemment.

```
{
"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. Exportez le modèle JSON CreateRfc pour un fichier de votre dossier actuel ; l'exemple le nomme Deploy CDApp RFC.json :

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

4. Modifiez et enregistrez le fichier Deploy CDApp RFC.json ; vous pouvez supprimer et remplacer le contenu.

```
{
"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. Créez la RFC en spécifiant le fichier de paramètres d'exécution et le fichier Deploy CDApp Rfc :

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

   Vous recevez le code RfcId de la nouvelle RFC dans la réponse. Enregistrez l'identifiant pour les étapes suivantes.

6. Soumettez le RFC :

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Si la RFC réussit, vous ne recevrez aucune sortie.

Valider le déploiement de l'application

Accédez au point de terminaison (ELB CName) de l'équilibreur de charge créé précédemment, avec le chemin WordPress déployé :/. WordPress Exemples :

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Démanteler le déploiement de l'application

Pour réduire le déploiement, vous soumettez le Delete Stack CT à la pile de base de données RDS, à l'équilibreur de charge de l'application, au groupe Auto Scaling, au compartiment S3, ainsi qu'à l'application et au groupe Code Deploy, soit six en tout. RFCs En outre, vous pouvez soumettre une demande de service pour que les instantanés RDS soient supprimés (ils sont supprimés automatiquement au bout de dix jours, mais ils coûtent peu cher pendant leur séjour). Rassemblez la pile IDs pour tous, puis suivez ces étapes.

Cette procédure pas à pas fournit un exemple d'utilisation de la console AMS pour supprimer une pile S3 ; cette procédure s'applique à la suppression de toute pile à l'aide de la console AMS.

> **Note**
>
> Si vous supprimez un compartiment S3, il doit d'abord être vidé de ses objets.

DONNÉES REQUISES :

- `StackId`: La pile à utiliser. Vous pouvez le trouver en consultant la page AMS Console Stacks, disponible via un lien dans le menu de navigation de gauche. À l'aide de l'API/CLI AMS SKMS, exécutez la référence For the AMS SKMS API, voir l'onglet Rapports dans l'opération AWS Artifact Console (dans l'interface de ligne de commande). `list-stack-summaries`
- L'identifiant du type de modification pour cette procédure pas à pas est « 1.0 ». Pour connaître la dernière version, exécutez cette commande : `ct-0q0bic0ywqk6c`

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0q0bic0ywqk6c
```

CRÉATION EN LIGNE :

- Émettez la commande create RFC avec les paramètres d'exécution fournis en ligne (évitez les guillemets lorsque vous fournissez des paramètres d'exécution en ligne). E

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0"
 --title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

- Soumettez le RFC à l'aide de l'ID RFC renvoyé lors de l'opération de création du RFC. Jusqu'à ce qu'elle soit soumise, la RFC reste en l'`Editing`état et ne fait l'objet d'aucune action.

```
aws amscm submit-rfc --rfc-id RFC_ID
```

- Surveillez l'état de la RFC et visualisez le résultat de l'exécution :

```
aws amscm get-rfc --rfc-id RFC_ID
```

CRÉATION D'UN MODÈLE :

1. Exportez le modèle RFC dans un fichier de votre dossier actuel ; l'exemple le nomme DeleteStackRfc .json :

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

2. Modifiez et enregistrez le fichier DeleteStackRfc .json. Comme la suppression d'une pile ne comporte qu'un seul paramètre d'exécution, les paramètres d'exécution peuvent se trouver dans

le fichier DeleteStackRfc .json lui-même (il n'est pas nécessaire de créer un fichier JSON distinct avec les paramètres d'exécution).

Les guillemets internes de l'extension ExecutionParameters JSON doivent être masqués par une barre oblique inverse (\). Exemple sans heure de début et de fin :

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0q0bic0ywqk6c",
"Title":                "Delete-My-Stack-RFC"
"ExecutionParameters":  "{
        \"StackId\":\"STACK_ID\"}"
}
```

3. Créez le RFC :

```
aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
```

Vous recevez le code RfcId de la nouvelle RFC dans la réponse. Exemples :

```
{
"RfcId": "daaa1867-ffc5-1473-192a-842f6b326102"
}
```

Enregistrez l'identifiant pour les étapes suivantes.

4. Soumettez le RFC :

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Si le RFC réussit, vous ne recevez aucune confirmation sur la ligne de commande.

5. Pour surveiller l'état de la demande et consulter le résultat de l'exécution, procédez comme suit :

```
aws amscm get-rfc --rfc-id RFC_ID --query "Rfc.
{Status:Status.Name,Exec:ExecutionOutput}" --output table
```

Maintenance des applications

Une fois l'infrastructure déployée, le défi consiste à la mettre à jour de manière cohérente dans tous vos environnements AMS, de l'assurance qualité à la phase de préparation en passant par la production.

Cette section fournit une vue d'ensemble du processus d'ingestion de la charge de travail AMS et quelques exemples des différentes méthodes que vous pouvez utiliser pour maintenir votre couche d'infrastructure cloud à jour.

Stratégies de maintenance des applications

La façon dont vous déployez vos applications a un impact sur la façon dont vous les gérez. Cette section fournit des stratégies pour la maintenance des applications.

Les mises à jour de l'environnement peuvent impliquer l'une des modifications suivantes :

- Mises à jour de sécurité
- Nouvelles versions de vos applications
- Modifications de configuration de l'application
- Mises à jour des dépendances

> **Note**
>
> Pour tout déploiement d'application, quelle que soit la méthode, déposez toujours une demande de service au préalable pour informer AMS que vous allez déployer une application.

Exemples d'installation d'applications immuables ou mutables

Mutabilité des instances de calcul	Méthode d'installation de l'application	AMI
Mutable	Avec CodeDeploy	Fourni par AMS

Mutabilité des instances de calcul	Méthode d'installation de l'application	AMI
	Manuellement	
	Avec un chef ou une marionnette, à Pull-Based	
	Avec Ansible ou Salt, en mode push	
Immuable	Avec un AMI doré	Personnalisé (basé sur l'AMS fourni)

Déploiement mutable avec une AMI CodeDeploy activée

AWS CodeDeploy est un service qui automatise les déploiements de code sur n'importe quelle instance, y compris les EC2 instances Amazon et les instances exécutées sur site. Vous pouvez utiliser CodeDeploy AMS pour créer et déployer une CodeDeploy application. Notez qu'AMS fournit un profil d'instance par défaut pour les CodeDeploy applications.

- Amazon Linux (version 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

Avant de l'utiliser CodeDeploy pour la première fois, vous devez effectuer un certain nombre d'étapes de configuration :

1. Installation ou mise à niveau de l'AWS CLI
2. Créez un rôle de service pour AWS CodeDeploy, vous utilisez l'ARN du rôle de service dans le déploiement

IDs pour toutes les options de tomodensitométrie, reportez-vous à la référence du type de modification.

> **Note**
>
> À l'heure actuelle, vous devez utiliser le stockage Amazon S3 avec cette solution.

Les étapes de base sont décrites ici et la procédure est détaillée dans le guide de l'utilisateur d'AMS.

1. Créez un compartiment de stockage Amazon S3. Numéro d'identification : ct-1a68ck03fn98r. La gestion des versions du compartiment S3 doit être activée (pour plus d'informations à ce sujet, voir Activation de la gestion des versions des compartiments).
2. Mettez-y vos CodeDeploy artefacts groupés. Vous pouvez le faire avec la console Amazon S3 sans demander d'accès via AMS. Ou en utilisant une variante de cette commande :

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trouvez une `customer-` AMI AMS ; utilisez l'une des options suivantes :

   - Console AMS : page de détails du VPC correspondant au VPC concerné
   - API AMS Pour obtenir des informations sur l'API AMS SKMS, consultez l'onglet Rapports de la console AWS Artifact ou de la CLI : `aws amsskms list-amis`

4. Créez un groupe Autoscaling (ASG). CT : ct-2tylseo8rxfsc. Spécifiez l'AMI AMS, configurez l'équilibreur de charge pour qu'il ait des ports ouverts, spécifiez `customer-mc-ec2-instance-profile` pour le`ASGIAMInstanceProfile`.
5. Créez votre CodeDeploy application. CT : ct-0ah3gwb9seqk2. Les paramètres incluent le nom de l'application, par exemple`WordpressProd`.
6. Créez votre groupe CodeDeploy de déploiement. CT : ct-2gd0u847qd9d2. Les paramètres incluent le nom de votre CodeDeploy application, le nom ASG, le nom du type de configuration et l'ARN du rôle de service.
7. Déployez CodeDeploy l'application. CT : ct-2edc3sd1sqmrb. Les paramètres incluent le nom de votre CodeDeploy application, le nom du type de configuration, le nom du groupe de déploiement, le type de révision et l'emplacement du compartiment S3 où se trouvent les CodeDeploy artefacts.

Déploiement mutable, instances d'application configurées et mises à jour manuellement

Cette stratégie de déploiement d'applications consiste en une mise à jour simple et manuelle des instances d'applications. Ce sont les étapes de base.

IDs pour toutes les options de tomodensitométrie, reportez-vous à la référence du type de modification.

> **Note**
>
> Actuellement, vous devez utiliser le stockage Amazon S3 avec cette solution.

Les étapes de base sont décrites ici ; les différentes procédures sont détaillées dans le guide de l'utilisateur AMS.

1. Créez un compartiment de stockage Amazon S3. Numéro d'identification : ct-1a68ck03fn98r. La gestion des versions du compartiment S3 doit être activée (pour plus d'informations à ce sujet, voir Activation de la gestion des versions des compartiments).
2. Mettez-y les artefacts de votre application groupée (tout ce dont votre application a besoin pour démarrer et fonctionner). Vous pouvez le faire avec la console Amazon S3 sans demander d'accès via AMS. Ou en utilisant une variante de cette commande :

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trouvez un AMI AMS, tout y CodeDeploy figurera. Pour trouver une AMI « client », utilisez l'une des méthodes suivantes :

   - Console AMS : page de détails du VPC correspondant au VPC concerné
   - API AMS Pour obtenir des informations sur l'API AMS SKMS, consultez l'onglet Rapports de la console AWS Artifact ou de la CLI : `aws amsskms list-amis`

4. Créez une EC2 instance avec cette AMI. Numéro d'identification : ct-14027q0sjyt1h. Spécifiez l'AMI AMS, définissez une balise `Key=backup, Value=true` et spécifiez le `customer-mc-ec2-instance-profile` `InstanceProfile` paramètre. Notez l'ID d'instance renvoyé.

5. Demandez un accès administrateur à l'instance. CT : ct-1dmlg9g1l91h6. Vous aurez besoin du FQDN pour votre compte. Si vous ne savez pas quel est votre FQDN, vous pouvez le trouver en :

   - Utilisation de la console de gestion AWS pour les services d'annuaire (sous l'onglet « Sécurité et identité »).
   - Exécution de l'une des commandes suivantes (classes de répertoire de retour ; DC+DC +DC=FQDN) : Windows : ou Linux :. `whoami /fqdn` `hostname --fqdn`

6. Connectez-vous à l'instance, consultez la section Accès aux instances via des bastions dans le guide de l'utilisateur AMS.
7. Téléchargez vos fichiers d'application groupés depuis votre compartiment S3 vers l'instance.
8. Demandez une sauvegarde immédiate en adressant une demande de service à AMS, vous devez connaître l'ID de l'instance.
9. Lorsque vous devez mettre à jour votre application, chargez de nouveaux fichiers dans votre compartiment S3, puis suivez les étapes 3 à 8.

Déploiement mutable avec une AMI configurée par un outil de déploiement basé sur le pull

Cette stratégie repose sur le `InstanceUserData` paramètre du Managed Services Create EC2 CT. Pour plus d'informations sur l'utilisation de ce paramètre, consultez la section Configuration des instances avec des données utilisateur. Cet exemple suppose un outil de déploiement d'applications basé sur le pull tel que Chef ou Puppet.

L' CodeDeploy agent est pris en charge sur tous les AMS AMIs. Voici la liste des produits pris en charge AMIs :

- Amazon Linux (version 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

IDs pour toutes les options de tomodensitométrie, reportez-vous à la référence des types de modifications.

> **Note**
>
> À l'heure actuelle, vous devez utiliser le stockage Amazon S3 avec cette solution.

Les étapes de base sont décrites ici et la procédure est détaillée dans le guide de l'utilisateur d'AMS.

1. Créez un compartiment de stockage Amazon S3. Numéro d'identification : ct-1a68ck03fn98r. La gestion des versions du compartiment S3 doit être activée (pour plus d'informations à ce sujet, voir Activation de la gestion des versions des compartiments).
2. Mettez-y vos CodeDeploy artefacts groupés. Vous pouvez le faire avec la console Amazon S3 sans demander d'accès via AMS. Ou en utilisant une variante de cette commande :

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trouvez une `customer-` AMI AMS ; utilisez l'une des options suivantes :

   - Console AMS : page de détails du VPC correspondant au VPC concerné
   - API AMS Pour obtenir des informations sur l'API AMS SKMS, consultez l'onglet Rapports de la console AWS Artifact ou de la CLI : `aws amsskms list-amis`

4. Créez une EC2 instance. CT : ct-14027q0sjyt1h ; définissez une balise `Key=backup, Value=true` et utilisez le `InstanceUserData` paramètre pour spécifier un bootstrap et d'autres scripts (Chef/Puppet agent de téléchargement, etc.), et incluez les clés d'autorisation nécessaires. Vous trouverez un exemple de cette méthode dans le guide de l'utilisateur d'AMS, section Gestion des modifications. Exemples de création d'un déploiement HA à deux niveaux. Vous pouvez également demander l'accès à l'instance, vous y connecter et la configurer avec les artefacts de déploiement nécessaires. N'oubliez pas que les commandes de déploiement basées sur le pull sont transmises par les agents de vos instances au serveur principal de votre entreprise et peuvent nécessiter une autorisation pour passer par les bastions. Vous pouvez avoir besoin d'une demande de service auprès d'AMS pour demander l'accès à un group/AD groupe de sécurité sans bastions.
5. Répétez l'étape 4 pour créer une autre EC2 instance et la configurer avec le serveur principal de l'outil de déploiement.
6. Lorsque vous devez mettre à jour votre application, utilisez l'outil de déploiement pour déployer les mises à jour sur vos instances.

Déploiement mutable avec une AMI configurée par un outil de déploiement basé sur le push

Cette stratégie repose sur le `InstanceUserData` paramètre du Managed Services Create EC2 CT. Pour plus d'informations sur l'utilisation de ce paramètre, consultez la section Configuration des instances avec des données utilisateur. Cet exemple suppose un outil de déploiement d'applications basé sur le pull tel que Chef ou Puppet.

IDs pour toutes les options de tomodensitométrie, reportez-vous à la référence du type de modification.

> **Note**
>
> Actuellement, vous devez utiliser le stockage Amazon S3 avec cette solution.

Les étapes de base sont décrites ici et la procédure est détaillée dans le guide de l'utilisateur d'AMS.

1. Créez un compartiment de stockage Amazon S3. Numéro d'identification : ct-1a68ck03fn98r. La gestion des versions du compartiment S3 doit être activée (pour plus d'informations à ce sujet, voir Activation de la gestion des versions des compartiments).
2. Mettez-y vos CodeDeploy artefacts groupés. Vous pouvez le faire avec la console Amazon S3 sans demander d'accès via AMS. Ou en utilisant une variante de cette commande :

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trouvez un AMI AMS, tout y CodeDeploy figurera. Pour trouver une AMI « client », utilisez l'une des méthodes suivantes :

   - Console AMS : page de détails du VPC correspondant au VPC concerné
   - API AMS Pour obtenir des informations sur l'API AMS SKMS, consultez l'onglet Rapports de la console AWS Artifact ou de la CLI : `aws amsskms list-amis`

4. Créez une EC2 instance. CT : ct-14027q0sjyt1h ; définissez une balise et utilisez le `InstanceUserData` paramètre pour exécuter un bootstrap et d'autres scriptsKey=backup, Value=true, notamment des clés d'autorisation, SALT stack (bootstrap un minion, pour plus d'informations, voir Bootstrapping Salt sous Linux EC2 avec Cloud-Init) ou Ansible (installez une paire de clés. Pour plus d'informations, consultez Getting Started with Ansible et Dynamic

Amazon Inventory Management). EC2 Vous pouvez également demander l'accès à l'instance, vous y connecter et la configurer avec les artefacts de déploiement nécessaires. N'oubliez pas que les commandes push sont transmises de votre sous-réseau d'entreprise à vos instances et que vous devrez peut-être configurer des autorisations pour qu'elles puissent passer par des bastions. Vous pouvez avoir besoin d'une demande de service auprès d'AMS pour demander l'accès à un group/AD groupe de sécurité sans bastions.

5. Répétez l'étape 4 pour créer une autre EC2 instance et la configurer avec le serveur principal de l'outil de déploiement.
6. Lorsque vous devez mettre à jour votre application, utilisez l'outil de déploiement pour déployer les mises à jour sur vos instances.

Déploiement immuable avec une AMI dorée

Cette stratégie utilise une AMI « dorée » que vous avez configurée pour qu'elle se comporte comme vous le souhaitez pour toutes vos instances d'application. Par exemple, les instances créées avec cette AMI dorée se joindraient automatiquement au domaine et au DNS appropriés, configureraient, redémarreraient et lanceraient automatiquement tous les systèmes nécessaires. Lorsque vous souhaitez mettre à jour vos instances d'application, vous devez recréer l'AMI dorée et déployer de toutes nouvelles instances d'application avec elle.

L' CodeDeploy agent est pris en charge sur tous les AMS AMIs. Voici la liste des produits pris en charge AMIs :

- Amazon Linux (version 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

IDs pour toutes les options de tomodensitométrie, reportez-vous à la référence du type de modification.

> **Note**
>
> Actuellement, vous devez utiliser le stockage Amazon S3 avec cette solution.

1. Créez un compartiment de stockage Amazon S3. Numéro d'identification : ct-1a68ck03fn98r. La gestion des versions du compartiment S3 doit être activée (pour plus d'informations à ce sujet, voir Activation de la gestion des versions des compartiments).
2. Mettez-y les artefacts de votre application groupée (tout ce dont votre application a besoin pour démarrer et fonctionner). Vous pouvez le faire avec la console Amazon S3 sans demander d'accès via AMS. Ou en utilisant une variante de cette commande :

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Trouvez une `customer-` AMI AMS ; utilisez soit :

   - Console AMS : page de détails du VPC correspondant au VPC concerné
   - API AMS Pour obtenir des informations sur l'API AMS SKMS, consultez l'onglet Rapports de la console AWS Artifact ou de la CLI : `aws amsskms list-amis`

4. Créez une EC2 instance avec cette AMI. Numéro d'identification : ct-14027q0sjyt1h. Spécifiez l'AMI AMS, définissez une balise `Key=backup, Value=true` et spécifiez `customer-mc-ec2-instance-profile` pour`InstanceProfile`. Notez l'ID d'instance renvoyé.
5. Demandez un accès administrateur à l'instance. CT : ct-1dmlg9g1l91h6. Vous aurez besoin du FQDN pour votre compte. Si vous ne savez pas quel est votre FQDN, vous pouvez le trouver en :

   - Utilisation de la console de gestion AWS pour les services d'annuaire (sous l'onglet « Sécurité et identité »).
   - Exécution de l'une des commandes suivantes (classes de répertoire de retour ; DC+DC +DC=FQDN) : Windows : ou Linux :. `whoami /fqdn` `hostname --fqdn`

6. Connectez-vous à l'instance, voir Accès aux instances dans le guide de l'utilisateur AMS.
7. Téléchargez sur l'instance vos fichiers d'application groupés depuis votre compartiment S3. Configurez l'instance afin qu'elle déploie automatiquement l'application entièrement fonctionnelle au démarrage.
8. Créez l'AMI dorée sur l'instance. CT : ct-3rqqu43krekby. Pour plus de détails, voir AMI | Create.
9. Configurez un groupe Auto Scaling pour créer de nouvelles instances à l'aide de cette AMI. CT : ct-2tylseo8rxfsc. Lorsque vous devez mettre à jour votre application, suivez cette procédure et demandez à AMS de mettre à jour l'ASG afin d'utiliser la nouvelle AMI dorée ; utilisez un scanner Management | Other | Other | Update pour cela.

Mettre à jour les stratégies

Il existe différentes stratégies que vous pouvez utiliser pour mettre à jour vos applications ou instances dans votre environnement géré par AMS.

- Interruption planifiée : cette stratégie simple consiste à planifier le temps pendant lequel votre application sera hors ligne et mise à jour manuellement. Pour ce faire, soumettez une demande Management | Other | Other | Update CT (ct-0xdawir96cy7k) pour arrêter les instances requises. Effectuez les mises à jour nécessaires, puis soumettez une autre demande Management | Other | Update CT (ct-0xdawir96cy7k) pour démarrer les instances.

- Bleu/Vert : cette stratégie nécessite que vous disposiez d'un environnement redondant (deux environnements entièrement fonctionnels) et que vous mettiez un environnement hors ligne à l'aide des mises à jour du système de noms de domaine (DNS) ou du pare-feu Web (WAF) pour rediriger le trafic. Mettez à jour un environnement, puis redirigez à nouveau pour mettre à jour l'autre environnement.

  Pour en savoir plus, consultez AWS CodeDeploy présente les Blue/Green déploiements.
- Mise à jour continue avec une nouvelle AMI : c'est ici que vous avez une nouvelle AMI que vous personnalisez (voir Create AMI), puis que vous demandez à AMS de la déployer dans votre groupe Auto Scaling. Utilisez un outil de gestion | Autre | Autre | Mise à jour CT (ct-0xdawir96cy7k) pour ce faire.

Planificateur de ressources AWS Managed Services

Utilisez le planificateur de ressources AWS Managed Services (AMS) pour planifier le démarrage et l'arrêt automatiques des AutoScaling groupes, des EC2 instances Amazon et des instances RDS dans votre compte. Cela permet de réduire les coûts d'infrastructure lorsque les ressources ne sont pas censées fonctionner 24 heures sur 24, 7 jours sur 7. La solution repose sur Instance Scheduler AWS, mais contient des fonctionnalités supplémentaires et des personnalisations spécifiques aux besoins d'AMS.

> **Note**
>
> Par défaut, AMS Resource Scheduler n'interagit pas avec les ressources qui ne font pas partie d'une AWS CloudFormation pile. La ressource doit faire partie d'une pile commençant par « stack- », « sc- » ou « SC- ». Pour planifier les ressources qui ne

font pas partie d'une CloudFormation pile, vous pouvez mettre à jour le paramètre `ScheduleNonStackResources` de pile Resource Scheduler sur. `Yes`

Le planificateur de ressources AMS utilise des périodes et des plannings :

- Les périodes définissent les heures d'exécution du planificateur de ressources, telles que l'heure de début, l'heure de fin et les jours du mois.
- Les plannings contiennent les périodes que vous avez définies, ainsi que des configurations supplémentaires, telles que la fenêtre de maintenance SSM, le fuseau horaire, le paramètre de mise en veille prolongée, etc. Ils spécifient le moment où les ressources doivent être exécutées, conformément aux règles de période configurées.

Vous pouvez configurer ces périodes et ces plannings à l'aide des types de modifications automatisés d'AMS Resource Scheduler ()CTs.

Pour plus de détails sur les paramètres disponibles pour AMS Resource Scheduler, consultez la documentation correspondante du planificateur d' AWS instance sur Composants de la solution. Pour une vue architecturale de la solution, consultez la documentation correspondante du planificateur d' AWS instance sur Architecture overview.html.

Déploiement du planificateur de ressources AMS

Pour déployer le planificateur de ressources AMS, utilisez le type de modification automatique (CT) : Déploiement | Planificateur de ressources AMS | Solution | Déploiement (ct-0ywnhc8e5k9z5) pour créer un RFC qui déploie ensuite la solution dans votre compte. Une fois le RFC exécuté, une CloudFormation pile contenant les ressources du planificateur de ressources AMS avec configuration par défaut est automatiquement provisionnée dans votre compte. Pour en savoir plus sur les types de modification du planificateur de ressources, consultez le planificateur de ressources AMS.

> **Note**
>
> Pour savoir si le planificateur de ressources AMS est déjà déployé dans votre compte, consultez la console AWS Lambda de ce compte et recherchez la fonction Scheduler. AMSResource

Une fois le planificateur de ressources AMS configuré dans votre compte, nous vous recommandons de revoir la configuration par défaut et, si nécessaire, de personnaliser les configurations telles que la clé de balise, le fuseau horaire, les services planifiés, etc., en fonction de vos préférences. Pour plus de détails sur les personnalisations recommandéesPersonnalisation du planificateur de ressources AMS, reportez-vous à la section suivante.

Pour effectuer les configurations personnalisées, ou simplement confirmer la configuration du planificateur de ressources,

Personnalisation du planificateur de ressources AMS

Nous vous recommandons de personnaliser les propriétés suivantes du planificateur de ressources AMS à l'aide des types de modification mis à jour du planificateur de ressources AMS, voir Planificateur de ressources AMS.

- Nom de balise : nom de la balise que le planificateur de ressources utilisera pour associer les plannings d'instance aux ressources. La valeur par défaut est Schedule.
- Services planifiés : liste séparée par des virgules de services que le planificateur de ressources peut gérer. La valeur par défaut est « ec2, rds, autoscaling ». Les valeurs valides sont « ec2 », « rds » et « autoscaling »
- Fuseau horaire par défaut : Spécifiez le fuseau horaire par défaut que le planificateur de ressources doit utiliser. La valeur par défaut est UTC.
- Utiliser CMK : liste séparée par des virgules des clés gérées par le client (CMK) Amazon KMS pour lesquelles le Resource Scheduler ARNs peut être autorisé à obtenir des autorisations.
- Utilisation LicenseManager : une liste séparée par des virgules des gestionnaires de AWS licences ARNs pour lesquels le planificateur de ressources peut être autorisé.

> **Note**
>
> AMS peut, de temps à autre, publier des fonctionnalités et des correctifs pour maintenir AMS Resource Scheduler à jour sur votre compte. Dans ce cas, toutes les personnalisations que vous apportez au planificateur de ressources AMS sont conservées.

Utilisation du planificateur de ressources AMS

Pour configurer le planificateur de ressources AMS après le déploiement de la solution, utilisez le planificateur de ressources automatisé CTs pour créer, supprimer, mettre à jour et décrire (obtenir des détails sur) les périodes du planificateur de ressources AMS (les heures d'exécution du planificateur de ressources) et les calendriers (les périodes configurées et autres options). Pour un exemple d'utilisation des types de modification du planificateur de ressources AMS, consultez le planificateur de ressources AMS.

Pour sélectionner les ressources à gérer par AMS Resource Scheduler, après le déploiement et la création du calendrier, vous utilisez l'AMS Tag Create CTs pour étiqueter les groupes Auto Scaling, les piles Amazon RDS et les EC2 ressources Amazon avec la clé de balise que vous avez fournie lors du déploiement, et le calendrier défini comme valeur de balise. Une fois les ressources balisées, le démarrage ou l'arrêt des ressources sont planifiés selon le calendrier que vous avez défini dans le planificateur de ressources.

L'utilisation d'AMS Resource Scheduler est gratuite. Cependant, la solution en utilise plusieurs Services AWS et ces ressources vous sont facturées au fur et à mesure de leur utilisation. Pour plus de détails, consultez la section Présentation de l'architecture.

Pour désactiver le planificateur de ressources AMS, procédez comme suit :

- Pour une désinscription ou une désactivation temporaire : soumettez une RFC à l'aide du gestionnaire automatisé | AMS Resource Scheduler | State | Disable change type (ct-14v49adibs4db)
- Pour une suppression définitive : soumettez une RFC de gestion | Autre | Autre | Mise à jour (révision requise) (ct-0xdawir96cy7k) demandant la suppression du système d'automatisation des versions du planificateur de ressources

Estimateur de coûts AMS Resource Scheduler

Afin de suivre les économies de coûts, AMS Resource Scheduler comporte un composant qui calcule toutes les heures les économies estimées pour les ressources Amazon EC2 et RDS gérées par le planificateur. Ces données de réduction des coûts sont ensuite publiées sous forme de CloudWatch métrique (`AMS/ResourceScheduler`) pour vous aider à les suivre. L'estimateur d'économies de coûts estime uniquement les économies réalisées sur les heures de fonctionnement des instances. Il ne prend pas en compte les autres coûts, tels que les coûts de transfert de données associés à une ressource.

L'estimateur d'économies de coûts est activé avec le planificateur de ressources. Il fonctionne toutes les heures et extrait les données sur les coûts et l'utilisation à partir de AWS Cost Explorer. À partir de ces données, il calcule le coût horaire moyen pour chaque type d'instance, puis projette le coût pour une journée complète si l'instance était exécutée sans être planifiée. Les économies de coûts correspondent à la différence entre le coût prévu et le coût réel indiqué par Cost Explorer pour un jour donné.

Par exemple, si l'instance A est configurée avec le planificateur de ressources pour fonctionner de 9 h à 17 h, cela représente huit heures par jour. Cost Explorer indique que le coût est de 1$ et que l'utilisation est de 8$. Le coût horaire moyen est donc de 0,125$. Si l'instance n'était pas planifiée avec le planificateur de ressources, elle s'exécuterait 24 heures sur 24 ce jour-là. Dans ce cas, le coût aurait été de 24 x 0,125 = 3$. Le planificateur de ressources vous a permis de réaliser des économies de 2$.

Pour que l'estimateur d'économies puisse récupérer les coûts et l'utilisation uniquement pour les ressources gérées par Resource Scheduler à partir de Cost Explorer, la clé de balise que le planificateur de ressources utilise pour cibler les ressources doit être activée en tant que balise de répartition des coûts dans le tableau de bord de facturation. Si le compte appartient à une organisation, la clé du tag doit être activée dans le compte de gestion de l'organisation. Pour plus d'informations sur cette procédure, voir Activation des balises de répartition des coûts définies par l'utilisateur et des balises de répartition des coûts définies par l'utilisateur

Une fois que la clé de balise est activée en tant que balise de répartition des coûts, la AWS facturation commence à suivre les coûts et l'utilisation des ressources gérées par le planificateur de ressources. Une fois ces données disponibles, l'estimateur des économies de coûts commence à calculer les économies de coûts et à publier les données sous l'`AMS/ResourceScheduler`espace de noms des métriques dans. CloudWatch

Conseils aux estimateurs de coûts

Cost Savings Estimator n'accepte pas les remises telles que les instances réservées, les plans d'épargne, etc., dans son calcul. L'estimateur prend les coûts d'utilisation de Cost Explorer et calcule le coût horaire moyen des ressources. Pour plus de détails, voir Comprendre vos ensembles de données de AWS coûts : un aide-mémoire

Pour que l'estimateur d'économies puisse récupérer les coûts et l'utilisation uniquement pour les ressources gérées par Resource Scheduler à partir de Cost Explorer, la clé de balise que le planificateur de ressources utilise pour cibler les ressources doit être activée en tant que balise

de répartition des coûts dans le tableau de bord de facturation. Si le compte appartient à une organisation, la clé du tag doit être activée dans le compte de gestion de l'organisation. Pour plus d'informations à ce sujet, consultez la section Balises de répartition des coûts définies par l'utilisateur. Si l'étiquette de répartition des coûts n'est pas activée, l'estimateur n'est pas en mesure de calculer les économies et de publier la métrique, même si elle est activée.

Bonnes pratiques du planificateur de ressources AMS

Planification d' EC2 instances Amazon

- Le comportement d'arrêt de l'instance doit être défini sur `stop` et non sur`terminate`. Ceci est prédéfini `stop` pour les instances créées avec le type de modification automatique AMS Amazon EC2 Create (ct-14027q0sjyt1h) et peut être défini pour les instances EC2 Amazon créées AWS CloudFormation par ingestion, en définissant la propriété sur. `InstanceInitiatedShutdownBehavior` `stop` Si le comportement d'arrêt des instances est défini sur`terminate`, elles se termineront lorsque le planificateur de ressources les arrêtera et le planificateur ne pourra pas les redémarrer.
- Les EC2 instances Amazon qui font partie d'un groupe Auto Scaling ne sont pas traitées individuellement par AMS Resource Scheduler, même si elles sont étiquetées.
- Si le volume racine de l'instance cible est chiffré à l'aide d'une clé principale client KMS (CMK), une `kms:CreateGrant` autorisation supplémentaire doit être ajoutée à votre rôle IAM de planificateur de ressources pour que le planificateur puisse démarrer de telles instances. Cette autorisation n'est pas ajoutée au rôle par défaut pour améliorer la sécurité. Si vous avez besoin de cette autorisation, soumettez une RFC avec le type de modification Management | AMS Resource Scheduler | Solution | Update et spécifiez une liste des KMS séparée par des ARNs virgules. CMKs

Planification de groupes Auto Scaling

- AMS Resource Scheduler démarre ou arrête le dimensionnement automatique des groupes Auto Scaling, et non des instances individuelles du groupe. C'est-à-dire que le planificateur rétablit la taille du groupe Auto Scaling (start) ou définit la taille sur 0 (stop).
- Marquez le AutoScaling groupe avec le tag spécifié et non les instances du groupe.
- Pendant l'arrêt, AMS Resource Scheduler enregistre les valeurs de capacité minimale, souhaitée et maximale du groupe Auto Scaling et définit les capacités minimale et souhaitée sur 0. Au démarrage, le planificateur rétablit la taille du groupe Auto Scaling telle qu'elle était lors de l'arrêt. Par conséquent, les instances du groupe Auto Scaling doivent utiliser une configuration de capacité

appropriée afin que la fermeture et le redémarrage des instances n'affectent aucune application exécutée dans le groupe Auto Scaling.

- Si le groupe Auto Scaling est modifié (capacité minimale ou maximale) pendant une période d'exécution, le planificateur enregistre la nouvelle taille du groupe Auto Scaling et l'utilise lors de la restauration du groupe à la fin d'un calendrier d'arrêt.

Planification d'instances Amazon RDS

- Le planificateur peut prendre un instantané avant d'arrêter les instances RDS (cela ne s'applique pas au cluster de base de données Aurora). Cette fonctionnalité est activée par défaut lorsque le paramètre Create RDS Instance Snapshot CloudFormation template est défini sur true. L'instantané est conservé jusqu'au prochain arrêt de l'instance Amazon RDS et à la création d'un nouvel instantané.

  Le planificateur peut utiliser des instances start/stop Amazon RDS faisant partie d'un cluster ou d'une base de données Amazon RDS Aurora ou dans une configuration de zones de disponibilité multiples (multi-AZ). Vérifiez toutefois les limites d'Amazon RDS lorsque le planificateur ne peut pas arrêter l'instance Amazon RDS, en particulier les instances multi-AZ. Pour planifier le démarrage ou l'arrêt d'Aurora Cluster, utilisez le paramètre de modèle Schedule Aurora Clusters (la valeur par défaut est true). Le cluster Aurora (et non les instances individuelles du cluster) doit être étiqueté avec la clé de balise définie lors de la configuration initiale et le nom du planning comme valeur de balise pour planifier ce cluster.

  Chaque instance Amazon RDS dispose d'une fenêtre de maintenance hebdomadaire au cours de laquelle toutes les modifications du système sont appliquées. Pendant la période de maintenance, Amazon RDS démarrera automatiquement les instances qui ont été arrêtées pendant plus de sept jours pour appliquer la maintenance. Notez qu'Amazon RDS n'arrêtera pas l'instance une fois l'événement de maintenance terminé.

  Le planificateur permet de spécifier s'il faut ajouter la fenêtre de maintenance préférée d'une instance Amazon RDS comme période d'exécution à son calendrier. La solution démarrera l'instance au début de la fenêtre de maintenance et l'arrêtera à la fin de la fenêtre de maintenance si aucune autre période d'exécution n'indique que l'instance doit s'exécuter et si l'événement de maintenance est terminé.

  Si l'événement de maintenance n'est pas terminé à la fin de la fenêtre de maintenance, l'instance sera exécutée jusqu'à l'intervalle de planification suivant la fin de l'événement de maintenance.

> **Note**
>
> Le planificateur ne valide pas le démarrage ou l'arrêt d'une ressource. Il fait l'appel d'API et passe à autre chose. Si l'appel d'API échoue, il enregistre l'erreur à des fins d'investigation.

Considérations concernant la sécurité des applications

La sécurité des applications inclut la prise en compte des autorisations dont l'application aura besoin pour s'exécuter, des règles de pare-feu et des rôles IAM à activer pour accéder à l'application.

Pour mieux comprendre AWS la sécurité générale, consultez les meilleures pratiques en matière de sécurité, d'identité et de conformité.

Accès pour la gestion de la configuration

AWS Managed Services (AMS) vise à vous fournir une infrastructure exempte de maux de tête afin que vous n'ayez pas à vous soucier des problèmes de sécurité, des correctifs, des problèmes de sauvegarde, etc. Pour ce faire, AMS recommande des rôles IAM minimaux permettant uniquement à un groupe spécifique ou à un serveur principal, si vous utilisez un outil de déploiement d'applications, d'accéder aux instances exécutant votre application.

Règles de pare-feu d'accès aux applications

Tout comme le système d'exploitation (OS), tous les accès aux applications doivent être régis par des groupes Active Directory (AD). En utilisant Amazon Relational Database Service (Amazon RDS) comme exemple, vous devez casser le miroir (réplication) pour ajouter un nouvel utilisateur. La meilleure approche consiste à créer un groupe dans AD et à l'ajouter au moment de la création de la base de données. La présence des groupes dans votre AMS AD signifie que vous pouvez créer des groupes CTs pour l'accès aux applications. Pour plus d'informations sur la stratégie de regroupement officielle pour AD, voir Utilisation de la stratégie d'imbrication de groupe — Meilleures pratiques d'AD pour la stratégie de groupe.

Pour en savoir plus sur les arborescences de parent/child domaines et les domaines, consultez Comment fonctionnent les domaines et les forêts.

Les règles suivantes illustrent une solution adaptée à un trust forestier multidomaine avec des utilisateurs situés dans des domaines enfants.

Instances Windows

Voici les règles à configurer pour vos contrôleurs de domaine Windows parent et enfant.

Contrôleur de domaine parent, Windows

DEPUIS : contrôleurs de domaine parents VERS : sous-réseaux Windows Stack et Shared Services

Port source	Port de destination	Protocole
88	49152 - 65535	TCP
389	49152 - 65535	UDP

DE : empiler les sous-réseaux, y compris les services partagés VERS : contrôleurs de domaine racine de forêt Windows

Port source	Port de destination	Protocole
49152 - 65535	88	TCP
49152 - 65535	389	UDP

Contrôleur de domaine enfant, Windows

DE : contrôleurs de domaine enfants VERS : contrôleurs de domaine Windows AWS

Port source	Port de destination	Protocole
49152 - 65535	53	TCP
49152 - 65535	88	TCP
49152 - 65535	389	UDP

DE : contrôleurs de domaine enfants VERS : sous-réseaux Windows Stack et Shared Services

Port source	Port de destination	Protocole
88	49152 - 65535	TCP
135	49152 - 65535	TCP

Port source	Port de destination	Protocole
389	49152 - 65535	TCP
389	49152 - 65535	UDP
445	49152 - 65535	TCP
49152 - 65535	49152 - 65535	TCP

DE : empiler les sous-réseaux, y compris les services partagés VERS : contrôleurs de domaine enfants Windows

Port source	Port de destination	Protocole
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

Instances Linux

Voici les règles à configurer pour vos contrôleurs de domaine parent et enfant Linux.

Tous les tests ont été réalisés à l'aide d'Amazon Linux. Alors que la plage de ports dynamiques pour Windows est comprise entre 49152 et 65535, de nombreux noyaux Linux utilisent la plage de ports 32768 à 61000. Exécutez la commande ci-dessous pour afficher la plage de ports IP.

```
cat /proc/sys/net/ipv4/ip_local_port_range
```

Contrôleur de domaine parent, Linux

DE : contrôleurs de domaine parents VERS : stack Linux et sous-réseaux de services partagés

Port source	Port de destination	Protocole
389	32768 - 61000	UDP
88	32768 - 61000	TCP

DE : sous-réseaux empilés, y compris les services partagés VERS : contrôleurs de domaine racine de la forêt Linux

Port source	Port de destination	Protocole
32768 - 61000	88	TCP
32768 - 61000	389	UDP

Contrôleur de domaine enfant, Linux

DE : contrôleurs de domaine enfants VERS : contrôleurs de domaine Linux AWS

Port source	Port de destination	Protocole
49152 - 65535	53	TCP
49152 - 65535	88	TCP
389	49152 - 65535	UDP
49152 - 65535	389	UDP

DE : contrôleurs de domaine enfants VERS : stack Linux et sous-réseaux de services partagés

Port source	Port de destination	Protocole
88	32768 - 61000	TCP

Port source	Port de destination	Protocole
389	32768 - 61000	UDP

DE : empiler les sous-réseaux, y compris les services partagés VERS : contrôleur de domaine enfant Linux

Port source	Port de destination	Protocole
32768 - 61000	88	TCP
32768 - 61000	389	UDP

Gestion du trafic de sortie AMS

Par défaut, la route dont le CIDR de destination est 0.0.0.0/0 pour les sous-réseaux privés et d'applications client AMS a une passerelle de traduction d'adresses réseau (NAT) comme cible. Les services AMS TrendMicro et les correctifs sont des composants qui doivent disposer d'un accès de sortie à Internet afin qu'AMS puisse fournir ses services TrendMicro et que les systèmes d'exploitation puissent obtenir des mises à jour.

AMS prend en charge le transfert du trafic de sortie vers Internet via un dispositif de sortie géré par le client, à condition que :

- Il agit comme un proxy implicite (transparent, par exemple).

  and

- Il autorise les dépendances HTTP et HTTPS AMS (répertoriées dans cette section) afin de permettre l'application continue des correctifs et la maintenance de l'infrastructure gérée par AMS.

Voici quelques exemples :

- La passerelle de transit (TGW) possède un itinéraire par défaut pointant vers le pare-feu sur site géré par le client via la connexion AWS Direct Connect dans le compte Multi-Account Landing Zone Networking.

- Le TGW dispose d'un itinéraire par défaut pointant vers un point de terminaison AWS dans le VPC de sortie de la zone d'accueil multi-comptes utilisant PrivateLink AWS, pointant vers un proxy géré par le client dans un autre compte AWS.
- Le TGW dispose d'un itinéraire par défaut pointant vers un pare-feu géré par le client dans un autre compte AWS, avec une connexion site-to-site VPN en pièce jointe à la zone d'atterrissage multi-comptes TGW.

AMS a identifié les dépendances HTTP et HTTPS AMS correspondantes, et développe et affine ces dépendances de manière continue. Voir egressMgmt.zip. Outre le fichier JSON, le fichier ZIP contient un fichier README.

> **Note**
>
> - Ces informations ne sont pas exhaustives : certains sites externes obligatoires ne sont pas répertoriés ici.
> - N'utilisez pas cette liste dans le cadre d'une liste de refus ou d'une stratégie de blocage.
> - Cette liste est conçue comme le point de départ d'un ensemble de règles de filtrage des sorties, dans l'espoir que des outils de reporting seront utilisés pour déterminer précisément où le trafic réel diverge de la liste.

Pour demander des informations sur le filtrage du trafic sortant, envoyez un e-mail à votre CSDM : ams-csdm@amazon.com.

Groupes de sécurité

Dans AWS VPCs, les groupes de sécurité AWS agissent comme des pare-feux virtuels, contrôlant le trafic pour une ou plusieurs piles (une instance ou un ensemble d'instances). Lorsqu'une pile est lancée, elle est associée à un ou plusieurs groupes de sécurité, qui déterminent le trafic autorisé à l'atteindre :

- Pour les piles de vos sous-réseaux publics, les groupes de sécurité par défaut acceptent le trafic HTTP (80) et HTTPS (443) en provenance de tous les emplacements (Internet). Les piles acceptent également le trafic SSH et RDP interne en provenance de votre réseau d'entreprise et des bastions AWS. Ces piles peuvent ensuite sortir via n'importe quel port vers Internet. Ils peuvent également accéder à vos sous-réseaux privés et à d'autres piles de votre sous-réseau public.

- Les piles de vos sous-réseaux privés peuvent être transférées vers n'importe quelle autre pile de votre sous-réseau privé, et les instances d'une pile peuvent communiquer pleinement entre elles via n'importe quel protocole.

> **Important**
>
> Le groupe de sécurité par défaut pour les piles sur les sous-réseaux privés permet à toutes les piles de votre sous-réseau privé de communiquer avec les autres piles de ce sous-réseau privé. Si vous souhaitez restreindre les communications entre les piles d'un sous-réseau privé, vous devez créer de nouveaux groupes de sécurité décrivant cette restriction. Par exemple, si vous souhaitez restreindre les communications avec un serveur de base de données afin que les piles de ce sous-réseau privé ne puissent communiquer qu'à partir d'un serveur d'applications spécifique via un port spécifique, demandez un groupe de sécurité spécial. La procédure à suivre est décrite dans cette section.

Groupes de sécurité par défaut

MALZ

Le tableau suivant décrit les paramètres par défaut du groupe de sécurité entrant (SG) pour vos piles. Le SG est nommé « SentinelDefaultSecurityGroupPrivateOnly -VPC-ID ». Il s'agit *ID* d'un identifiant VPC dans votre compte de zone d'atterrissage multi-comptes AMS. Tout le trafic sortant vers « mc-initial-garden - SentinelDefaultSecurityGroupPrivateOnly » est autorisé via ce groupe de sécurité (tout le trafic local au sein des sous-réseaux de pile est autorisé).

Tout le trafic sortant vers 0.0.0.0/0 est autorisé par un deuxième groupe de sécurité « ». SentinelDefaultSecurityGroupPrivateOnly

> **Tip**
>
> Si vous choisissez un groupe de sécurité pour un type de modification AMS, tel que EC2 create ou OpenSearch create domain, vous devez utiliser l'un des groupes de sécurité par défaut décrits ici, ou un groupe de sécurité que vous avez créé. Vous trouverez la liste des groupes de sécurité, par VPC, dans la console AWS EC2 ou dans la console VPC.

D'autres groupes de sécurité par défaut sont utilisés à des fins AMS internes.

Groupes de sécurité AMS par défaut (trafic entrant)

Type	Protocole	Plage de ports	Source
Tout le trafic	Tous	Tous	SentinelDefaultSecurityGroupPrivateOnly (limite le trafic sortant aux membres du même groupe de sécurité)
Tout le trafic	Tous	Tous	SentinelDefaultSecurityGroupPrivateOnlyEgressAll (ne limite pas le trafic sortant)
HTTP, HTTPS, SSH, RDP	TCP	80/ 443 (Source : 0,0,0,0/0) L'accès SSH et RDP est autorisé depuis les bastions	SentinelDefaultSecurityGroupPublic (ne limite pas le trafic sortant)
Bastions du MALZ :			
SSH	TCP	22	SharedServices VPC CIDR et DMZ VPC CIDR, ainsi que sur site fournis par le client CIDRs
SSH	TCP	22	
RDP	TCP	3389	
RDP	TCP	3389	
Bastions SALZ :			
SSH	TCP	22	mc-initial-garden- LinuxBastion SG
SSH	TCP	22	mc-initial-garden- LinuxBastion DMZG
RDP	TCP	3389	mc-initial-garden- WindowsBastion SG
RDP	TCP	3389	mc-initial-garden- WindowsBastion DMZG

SALZ

Le tableau suivant décrit les paramètres par défaut du groupe de sécurité entrant (SG) pour vos piles. Le SG est nommé « mc-initial-garden - SentinelDefaultSecurityGroupPrivateOnly - *ID* » où se *ID* trouve un identifiant unique. Tout le trafic sortant vers « mc-initial-garden - SentinelDefaultSecurityGroupPrivateOnly » est autorisé via ce groupe de sécurité (tout le trafic local au sein des sous-réseaux de pile est autorisé).

Tout le trafic sortant vers 0.0.0.0/0 est autorisé par un deuxième groupe de sécurité « - - »mc-initial-garden. SentinelDefaultSecurityGroupPrivateOnlyEgressAll *ID*

> **Tip**
>
> Si vous choisissez un groupe de sécurité pour un type de modification AMS, tel que EC2 create ou OpenSearch create domain, vous devez utiliser l'un des groupes de sécurité par défaut décrits ici, ou un groupe de sécurité que vous avez créé. Vous trouverez la liste des groupes de sécurité, par VPC, dans la console AWS EC2 ou dans la console VPC.

D'autres groupes de sécurité par défaut sont utilisés à des fins AMS internes.

Groupes de sécurité AMS par défaut (trafic entrant)

Type	Protocole	Plage de ports	Source
Tout le trafic	Tous	Tous	SentinelDefaultSecurityGroupPrivateOnly (limite le trafic sortant aux membres du même groupe de sécurité)
Tout le trafic	Tous	Tous	SentinelDefaultSecurityGroupPrivateOnlyEgressAll (ne limite pas le trafic sortant)
HTTP, HTTPS, SSH, RDP	TCP	80/ 443 (Source : 0,0,0,0/0) L'accès SSH et RDP est autorisé depuis les bastions	SentinelDefaultSecurityGroupPublic (ne limite pas le trafic sortant)

Type	Protocole	Plage de ports	Source
Bastions du MALZ :			
SSH	TCP	22	SharedServices VPC CIDR et DMZ VPC CIDR, ainsi que sur site fournis par le client CIDRs
SSH	TCP	22	
RDP	TCP	3389	
RDP	TCP	3389	
Bastions SALZ :			
SSH	TCP	22	mc-initial-garden- LinuxBastion SG
SSH	TCP	22	mc-initial-garden- LinuxBastion DMZG
RDP	TCP	3389	mc-initial-garden- WindowsBastion SG
RDP	TCP	3389	mc-initial-garden- WindowsBastion DMZG

Création, modification ou suppression de groupes de sécurité

Vous pouvez demander des groupes de sécurité personnalisés. Dans les cas où les groupes de sécurité par défaut ne répondent pas aux besoins de vos applications ou de votre organisation, vous pouvez modifier ou créer de nouveaux groupes de sécurité. Une telle demande serait considérée comme nécessitant une approbation et serait examinée par l'équipe des opérations de l'AMS.

Pour créer un groupe de sécurité en dehors des piles VPCs, soumettez une RFC en utilisant le type de `Deployment | Advanced stack components | Security group | Create (managed automation)` modification (ct-1oxx2g2d7hc90).

Pour les modifications du groupe de sécurité Active Directory (AD), utilisez les types de modifications suivants :

- Pour ajouter un utilisateur : soumettez une RFC à l'aide de Management | Directory Service | Utilisateurs et groupes | Ajouter un utilisateur au groupe [ct-24pi85mjtza8k]
- Pour supprimer un utilisateur : soumettez une RFC à l'aide de Management | Directory Service | Utilisateurs et groupes | Supprimer un utilisateur du groupe [ct-2019s9y3nfml4]

> **Note**
>
> Lorsque vous utilisez le mode manuel CTs, AMS vous recommande d'utiliser l'option ASAP Scheduling (choisissez ASAP dans la console, laissez les heures de début et de fin vides dans l'API/CLI) car elles CTs nécessitent qu'un opérateur AMS examine la RFC et communique éventuellement avec vous avant qu'elle ne puisse être approuvée et exécutée. Si vous les planifiez RFCs, veillez à prévoir au moins 24 heures. Si l'approbation n'intervient pas avant l'heure de début prévue, le RFC est automatiquement rejeté.

Rechercher des groupes de sécurité

Pour rechercher les groupes de sécurité attachés à une pile ou à une instance, utilisez la console EC2. Après avoir trouvé la pile ou l'instance, vous pouvez voir tous les groupes de sécurité qui y sont attachés.

Pour savoir comment rechercher des groupes de sécurité sur la ligne de commande et filtrer la sortie, consultez `describe-security-groups`.

Annexe : Questionnaire d'accueil des candidatures

Utilisez ce questionnaire pour décrire les éléments et la structure de votre déploiement afin qu'AMS puisse déterminer les composants d'infrastructure nécessaires. Les exigences d'intégration pour les applications Line-of-Business (LoB) sont très différentes de celles des applications de produits. Ce questionnaire est donc conçu pour répondre à ces deux exigences.

Rubriques

Récapitulatif du déploiement

Description du déploiement. Exemples :

- Ce compte est destiné au déploiement d'une application Line-of-Business (LoB) (par opposition au déploiement d'une application de produit).
- Le déploiement implique un ARP (proxy inverse authentifié) à mise à l'échelle automatique au sein du sous-réseau du compte. public/DMZ
- Les serveurs Web et d'applications seront déployés dans le sous-réseau privé du compte.
- Une instance Amazon RDS (Amazon Relational Database Service) sera également déployée dans le sous-réseau privé du compte.
- Les serveurs (ARP, Web, application, base de données, équilibreur de charge, etc.) sont séparés en groupes de sécurité distincts.
- Le compte nécessite une conception HA (haute disponibilité) répartie sur les zones de disponibilité (AZs), c'est-à-dire multi-AZ.

Composants de déploiement de l'infrastructure

Quels sont les différents composants qui devront être configurés pour prendre en charge votre application ?

- Région : Quelles sont les régions Région AWS ou les régions nécessaires ?
- Haute disponibilité (HA) : quelles zones de disponibilité seront utilisées ?
- Virtual Private Cloud (VPC) : Qu'est-ce que le bloc CIDR pour le VPC ?
- Quelles sont les instances de serveur nécessaires ?
  - Proxy inverse authentifié (ARP) : système d'exploitation, AMI, type d'instance, ID de sous-réseau, groupe de sécurité, port d'entrée ?
  - Serveur d'outils de déploiement d'applications : système d'exploitation, AMI, type d'instance, ID de sous-réseau, groupe de sécurité, port d'entrée (Chef, Puppet) ou port de sortie (Ansible, Saltstack) ?
  - Amazon RDS with MySQL : version de base de données, type d'utilisation, classe d'instance, ID de sous-réseau, groupe de sécurité, ID d'instance de base de données, taille de stockage, multi-AZ, type d'authentification, chiffrement ?
  - Stockage : votre application est-elle apatride ? Avez-vous besoin de compartiments S3 ? Avez-vous besoin d'un stockage permanent ? Avez-vous besoin d'un chiffrement des données au repos sur vos volumes EBS ? Avez-vous besoin d'un chiffrement de base de données ?
  - Points de terminaison du serveur externes (au VPC Managed Services) : SMTP ? LDAP ?
  - Exigences du réseau : filtrage du réseau (basé sur les groupes de sécurité ?) ? Inspection du trafic Web (entrant ? sortant ?) ?
- Balisage : quelles balises doivent être utilisées pour regrouper les ressources dans des collections logiques ? Par exemple, toutes les ressources d'une pile d'applications. Sélectionnez des balises adaptées à votre cas d'utilisation, par exemple `backup=true` pour activer les sauvegardes. En outre, vous devez utiliser la balise `name=value` pour que les EC2 instances que vous créez puissent afficher un nom dans la console.
- Groupes de sécurité :
  - Quels sont les groupes de sécurité nécessaires ?
  - Règles d'accès aux groupes de sécurité ?
  - Règles de sortie des groupes de sécurité ?

Plateforme d'hébergement d'applications

Pour votre plateforme d'hébergement d'applications, tenez compte des exigences possibles suivantes :

- Bases de données cryptées ?
- Les clés de chiffrement sont gérées par qui ?
- Toutes les données en transit et au repos sont-elles cryptées ?
- Tous les utilisateurs accèdent-ils au système via HTTPS ?
- Toutes les system-to-system interactions ont-elles été approuvées par votre équipe des opérations de sécurité ?

Modèle de déploiement d'applications

Considérations relatives à la façon dont vous planifiez les déploiements d'applications. Consultez Quel est mon modèle de fonctionnement ?.

- Automatisé ou manuel ? L'absence d'automatisation du déploiement signifie l'absence de mise à l'échelle automatique. Si vous demandez l'accès, que vous vous connectez et que vous mettez à jour manuellement votre application, la mise à jour échoue. AMS s'attend à ce que vous annuliez votre mise à jour ou que vous nous alertiez par le biais d'une demande de service afin que nous puissions vous aider.
- En cas d'automatisation, quel est le cadre ? Des scripts ? Basé sur un agent () ? puppet/chef)? Agentless (SALT/Ansible CodeDeploy? Les outils de déploiement basés sur un agent et sans agent nécessitent la création et le déploiement d'une instance distincte en tant que serveur principal pour l'outillage. AMS attend de vous que vous connaissiez tous les éléments nécessaires à la réussite des outils de déploiement d'applications. Toutefois, nous sommes heureux de répondre aux questions d'infrastructure connexes.
- Vos Line-of-Business applications (les applications que vous utilisez pour créer et gérer vos applications) nécessitent-elles des correctifs ?

Dépendances des applications

Avez-vous besoin d'instances pour les applications Line-of-Business (LoB) ? Pour les applications de produits ?

De quoi ont besoin les applications de vos produits pour fonctionner correctement ?

- Dépendances au niveau du réseau : par exemple, Direct Connect
- Dépendances du package : par exemple, `pip`
- Applications dont dépend cette application : Par exemple, MySql
- Dépendances liées au pare-feu ?

De quoi ont besoin vos applications LoB pour fonctionner correctement ?

- Dépendances au niveau du réseau : par exemple, Direct Connect
- Dépendances du package : par exemple, Firefox Saucy
- Applications dont dépend cette application : Par exemple, MySql
- Dépendances liées au pare-feu ?

Certificats SSL pour les applications de produits

De quels certificats SSL vos serveurs auront-ils besoin pour que vos applications (LoB et produit) puissent accéder à tout ce dont elles ont besoin pour fonctionner et être accessibles ?

- Groupe Auto Scaling ?
- Base de données (Amazon RDS) ?
- Load Balancer ?
- Serveur d'outils de déploiement ?
- Pare-feu pour applications Web (AWS WAF) ?
- D'autres instances ?

Par exemple, pour chacune des instances répertoriées ci-dessus, vous pourriez avoir besoin des certificats suivants :

WAF (certificat 1) -> ELB-Ext (certificat 2) -> ARP (certificat 3) - > ELB-int (certificat 4) -> Site Web (certificat 5) - > ELB-int (certificat 6) -> Service Web (certificat 7).

Historique du document

Le tableau suivant décrit la documentation de cette version d'AMS.

- Version de l'API : 2019-05-21
- Dernière mise à jour de la documentation : 16 février 2023

Modification	Description	Lien
Lien vers la table des matières supprimé	Le lien vers le AWS glossaire de la table des matières a été supprimé.	8 août 2025
Contenu mis à jour : Migration des charges de travail : validation préalable à l'ingesti on de Windows	Section mise à jour pour inclure les étapes détaillées d'utilisation du script de WIGs pré-validation afin de valider que votre instance Windows est prête à être ingérée dans votre compte AMS ;	Migration des charges de travail : validation préalable à l'ingestion de Windows
Contenu mis à jour, configura tion DMS	une note importante concernant le rôle requis, dms-vpc-role	1 : groupe AWS DMS de sous-réseaux de réplicati on : Créer
Contenu mis à jour, ressource s prises en charge par CFN Ingest	Ajouté OpenSearch.	Ressource s prises en charge
Contenu mis à jour, migration des charges de travail	Instructions mises à jour pour la validation avant ingestion.	Migration des charges de travail : validation préalable à

Modification	Description	Lien
		l'ingestion de Windows
Contenu mis à jour, CFN Ingest.	Les « ressources prises en charge » restreintes ont été supprimées du contenu d'ingestion CFN.	CloudFormation Ingest Stack : ressources prises en charge
Versions Windows prises en charge mises à jour	Ajout du support pour Windows Server 2022.	Images de machines AMS Amazon (AMIs), Migration des charges de travail : conditions préalables pour Linux et Windows et Migration des charges de travail : validation préalable à l'ingestion de Windows
Contenu mis à jour, planificateur de ressources.	Instructions mises à jour pour utiliser le CT de déploiement dédié, ct-0ywnhc8e5k9z5, applicable à la fois à SALZ et à MALZ.	Démarrage rapide du planificateur de ressources AMS

Modification	Description	Lien
Contenu mis à jour, Workload Ingest.	Mise à jour des versions de SUSE Linux prises en charge.	Migration des charges de travail : conditions préalables pour Linux et Windows
Contenu mis à jour, Database Migration Service.	Ajouté aux prérequis et apporté plusieurs modifications pour des raisons d'utilité et de convivialité.	AWS Database Migrat (AWS DMS)
Contenu mis à jour, Workload Ingest.	Le zip de validation Linux pré-Wigs a été mis à jour.	Migration des charges de travail : conditions préalables pour Linux et Windows
Contenu mis à jour.	Mise à jour du fichier zip de validation pré-Wigs pour Linux. Windows Server 2008 R2 a également été ajouté en tant que système d'exploitation pris en charge.	Migration des charges de travail : conditions préalables pour Linux et Windows
Nouveau contenu	Les tutoriels et les guides de démarrage rapide ont été déplacés ici à partir de l'ancien guide AMS Advanced Change Management.	Démarrages rapides, Didacticiels.

Modification	Description	Lien
Contenu mis à jour	Déploiement | Composants de pile avancés | Service de migration de base de données (DMS) | Lancer la tâche de réplication (ct-1yq7hhqse71yg) Mis à jour pour indiquer que les paramètres DocumentNameet la région sont obligatoires ; auparavant, ils étaient répertoriés par erreur comme facultatifs.	Service de migration de base de données (DMS) | Démarrer la tâche de réplication
Contenu mis à jour	CloudFormation Ingérer Mis à jour pour indiquer deux nouvelles ressources prises en charge, AWS::Route53Resolver::ResolverRuleAssociation et AWS::Route53Resolver::ResolverRule.	Ressources prises en charge
Contenu mis à jour	Migration des charges de travail : validation préalable à l'ingestion de Windows	Informations Sysprep mises à jour avec des informations plus spécifiques. Migration des charges de travail : validation préalable à l'ingestion de Windows

Modification	Description	Lien
Contenu mis à jour	Gestion | Pile personnalisée | Stack à partir d'un CloudFormation modèle | Approuver l'ensemble de modifications et la mise à jour (ct-1404e21baa2ox) La description détaillée de la tomodensitométrie pour le ChangeSetNameparamètre a été mise à jour avec des informations supplémentaires.	Empiler à partir d' CloudForm ation un modèle | Approuver l'ensemble des modificat ions et les mettre à jour
	Ubuntu 18.04 et Oracle Linux 8.3 sont disponibl es	Migration des charges de travail : conditions préalables pour Linux et Windows
Nouveau contenu :	Déploiements IAM via CFN Ingest et Stack Update. CTs	10 février 2022
Tâches de réplication du Service de Migration de Base de Données (DMS)	Les types de modifications ont été mis à jour afin que les expressions régulières autorisent ARNs les tâches contenant des traits d'union. Lancer AWS DMS la tâche de réplicationet Database Migration Service (DMS) | Arrêter la tâche de réplication.	13 janvier 2022
Validation préalable à l'ingesti on de Linux WIGS	Le fichier zip a été mis à jour. Migration des charges de travail : validation préalable à l'ingestion de Linux.	13 janvier 2022

Modification	Description	Lien
Liens fixes	La Configuration section Importation de base de données (DB) vers AMS SQL RDS -> contenait des liens erronés.	13 janvier 2022

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.