Opciones avanzadas de despliegue de aplicaciones de AMS

Guía para desarrolladores de aplicaciones avanzadas de AMS

Version September 13, 2024

Guía para desarrolladores de aplicaciones avanzadas de AMS: Opciones avanzadas de despliegue de aplicaciones de AMS

Table of Contents

Incorporación de aplicaciones

Bienvenido al plan de operaciones de AMS de AWS Managed Services (AMS). El objetivo de este documento es describir los distintos métodos que puede utilizar al incorporar sus aplicaciones a AMS una vez que se haya configurado la gestión inicial de redes y accesos, y los aspectos que debe tener en cuenta al elegir esos métodos.

Este documento está destinado a los integradores de sistemas y desarrolladores de aplicaciones para ayudarlos a determinar y elaborar los procesos de aplicación para los nuevos clientes de AMS.

¿Qué es la incorporación de aplicaciones?

La incorporación de aplicaciones de AMS se refiere al despliegue de recursos y aplicaciones, según sea necesario, en su infraestructura de AMS. Diseñar la arquitectura de las aplicaciones y la infraestructura en la plataforma AMS es muy similar a hacerlo en una plataforma nativa. AWS Si se siguen las mejores prácticas de diseño de AWS aplicaciones e infraestructuras y se tienen en cuenta las capacidades que ofrece AMS, se obtendrán aplicaciones capaces y operables alojadas en el entorno AMS.

> **Note**
>
> - EE.UU. Este (Virginia)
> - Oeste de EE. UU. (Norte de California)
> - Oeste de EE. UU. (Oregón)
> - Este de EE. UU. (Ohio)
> - Canadá (centro)
> - América del Sur (São Paulo)
> - UE (Irlanda)
> - UE (Fráncfort)
> - UE (Londres)
> - EU West (París)
> - Asia-Pacífico (Mumbai)
> - Asia-Pacífico (Seúl)
> - Asia-Pacífico (Singapur)

- Asia-Pacífico (Sídney)
- Asia-Pacífico (Tokio)

Se añaden nuevas regiones con frecuencia. Para obtener más información, consulta Regiones de AWS Zonas de disponibilidad.

Lo que hacemos, lo que no hacemos

AMS le ofrece un enfoque estandarizado para implementar la infraestructura de AWS y proporciona la administración operativa continua necesaria. Para obtener una descripción completa de las funciones, responsabilidades y servicios compatibles, consulte la descripción del servicio.

> **Note**
>
> Para solicitar que AMS proporcione un servicio de AWS adicional, presente una solicitud de servicio. Para obtener más información, consulte Realización de solicitudes de servicio.

- Qué hacemos:

  Una vez completada la incorporación, el entorno AMS estará disponible para recibir solicitudes de cambio (RFCs), incidentes y solicitudes de servicio. Su interacción con el servicio AMS gira en torno al ciclo de vida de una pila de aplicaciones. Las nuevas pilas se ordenan a partir de una lista preconfigurada de plantillas, se lanzan a subredes de nube privada virtual (VPC) específicas, se modifican durante su vida operativa mediante solicitudes de cambio (RFCs) y se supervisan las 24 horas del día, los 7 días de la semana para detectar eventos e incidentes.

  AMS supervisa y mantiene las pilas de aplicaciones activas, incluida la aplicación de parches, y no requieren ninguna otra acción durante la vida útil de la pila, a menos que sea necesario cambiarlas o retirarlas del servicio. Los incidentes detectados por AMS que afecten al estado y al funcionamiento de la pila generan una notificación y es posible que necesites o no que tomes medidas para resolverlos o verificarlos. Puede realizar preguntas prácticas y de otro tipo enviando una solicitud de servicio.

Además, AMS le permite habilitar servicios de AWS compatibles que no están gestionados por AMS. Para obtener información sobre los servicios compatibles con AWS-AMS, consulte Modo de aprovisionamiento de autoservicio.

- Lo que NO hacemos:

  Si bien AMS simplifica la implementación de aplicaciones al proporcionar una serie de opciones manuales y automatizadas, usted es responsable del desarrollo, las pruebas, la actualización y la administración de su aplicación. AMS proporciona asistencia para la resolución de problemas de infraestructura que afectan a las aplicaciones, pero AMS no puede acceder a las configuraciones de las aplicaciones ni validarlas.

Imágenes de máquinas AMS Amazon (AMIs)

AMS produce Amazon Machine Images (AMIs) actualizadas todos los meses para los sistemas operativos compatibles con AMS. Además, AMS también produce imágenes de seguridad mejorada (AMIs) basadas en el estándar de nivel 1 del CIS para un subconjunto de los sistemas operativos compatibles con AMS. Para saber qué sistemas operativos tienen disponible una imagen de seguridad mejorada, consulte la Guía del usuario de AMS Security, que está disponible en la página AWS Artifact -> Reports (busque la opción Informes en el panel de navegación izquierdo) filtrada para AWS Managed Services. Para acceder a AWS Artifact, puede ponerse en contacto con su CSDM para obtener instrucciones o ir a Introducción a AWS Artifact.

Para recibir alertas cuando se AMIs publiquen nuevos AMS, puedes suscribirte a un tema de notificaciones del Amazon Simple Notification Service (Amazon SNS) denominado «AMI de AMS». Para obtener más información, consulte Notificaciones AMI de AMS con SNS.

La convención de nomenclatura AMI de AMS es:`customer-ams-<operating system>-<release date> - <version>`. (por ejemplo,`customer-ams-rhel6-2018.11-3`)

Utilice únicamente los AMS AMIs que comiencen por`customer`.

AMS recomienda utilizar siempre la AMI más reciente. Puede encontrar la más reciente de AMIs las siguientes maneras:

- Busca en la consola AMS, en la AMIspágina.

- Visualización del archivo CSV AMI de AMS más reciente, disponible en su CSDM o a través de este archivo ZIP: contenido de la AMI AMS 11.2024 y archivo CSV en formato ZIP.

  Para ver los archivos ZIP AMI anteriores, consulte el historial del documento.

- Al ejecutar este SKMS comando AMS (se requiere el SDK de AMS SKMS):

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

Contenido AMI de AMS agregado a la base AWS AMIs, por sistema operativo (SO)

- Linux AMIs:
  - AWS Herramientas CLI
  - NTP
  - Agente de Trend Micro Endpoint Protection Service
  - Implementación de código
  - PBIS/Beyond Trust AD Bridge
  - SSM Agent
  - Actualización de Yum para parches críticos
  - Software de administración y scripts personalizados de AMS (que controlan el arranque, la unión a AD, la supervisión, la seguridad y el registro)
- Servidor Windows AMIs:
  - Microsoft.NET Framework 4.5
  - PowerShell 5.1
  - AWS Herramientas para Windows PowerShell
  - PowerShell Módulos AMS que controlan el arranque, la unión a AD, la supervisión, la seguridad y el registro
  - Agente de Trend Micro Endpoint Protection Service
  - SSM Agent
  - CloudWatch Agente
  - EC2Servicio de configuración (a través de Windows Server 2012 R2)
  - EC2Lanzamiento (Windows Server 2016 y Windows Server 2019)
  - EC2LaunchV2 (Windows Server 2022 y versiones posteriores)

Basado en Linux: AMIs

- Amazon Linux 2023 (última versión secundaria) (no se admite una AMI mínima)
- Amazon Linux 2 (última versión secundaria)
- Amazon Linux (2ARM64)
- Red Hat Enterprise 7 (última versión secundaria)
- Red Hat Enterprise 8 (última versión secundaria)
- Red Hat Enterprise 9 (última versión secundaria)
- SUSE Linux Enterprise Server 15 SP6
- Ubuntu Linux 18.04
- Ubuntu Linux 20.04
- Ubuntu Linux 22.04
- Ubuntu Linux 24.04
- Amazon Linux: para obtener información general del producto, información sobre precios, información de uso e información de soporte, consulte la AMI de Amazon Linux (HVM/64 bits) y Amazon Linux 2.

  Para obtener más información, consulte Amazon Linux 2 FAQs.
- RedHat Enterprise Linux (RHEL): para obtener una descripción general del producto, información sobre precios, información de uso e información de soporte, consulte Red Hat Enterprise Linux (RHEL) 7 (HVM).
- Ubuntu Linux 18.04: Para obtener una descripción general del producto, información sobre precios, información de uso e información de soporte, consulte Ubuntu 18.04 LTS - Bionic.
- SUSE Linux Enterprise Server para aplicaciones SAP 15: SP6
  - Ejecute los siguientes pasos una vez por cuenta:
    1. Vaya a AWS Marketplace.
    2. Busque el producto SAP SUSE 15.
    3. Elija Continuar para suscribirse.
    4. Elija Aceptar condiciones.
  - Complete los siguientes pasos cada vez que necesite lanzar una nueva instancia de SUSE Linux Enterprise Server for SAP Applications 15 SP6:
    1. Anote el ID de AMI de la AMI de SUSE Linux Enterprise Server for SAP Applications 15 suscrita.

2. Crear una implementación | Componentes de pila avanzados | pila | Crear tipo de cambio EC2 ct-14027q0sjyt1h RFC. *InstanceAmiId*Sustitúyala por la ID de AWS Marketplace AMI a la que te has suscrito.

Basado en Windows AMIs:

Microsoft Windows Server (2016, 2019 y 2022), basado en la versión más reciente de Windows AMIs.

Para ver ejemplos de creación AMIs, consulte Crear AMI.

Desembarcar AMS AMIs:

AMS no deja AMIs de compartir nada tuyo durante la baja para evitar que tus dependencias se vean afectadas. Si quieres eliminar AMS AMIs de tu cuenta, puedes usar la `cancel-image-launch-permission` API para ocultar información específica. AMIs Por ejemplo, puedes usar el siguiente script para ocultar todos los AMS AMIs que se compartieron anteriormente con tu cuenta:

```
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
```

Debe tener la versión 2 de AWS CLI instalada para que el script se ejecute sin errores. Para ver los pasos de instalación de la CLI de AWS, consulte Instalación o actualización de la última versión de la CLI de AWS. Para obtener más información sobre el `cancel-image-launch-permission` comando, consulte `cancel-image-launch-permission`.

Seguridad mejorada AMIs

AMS proporciona imágenes de seguridad mejorada (AMIs) basadas en la referencia de nivel 1 del CIS para un subconjunto de los sistemas operativos compatibles con AMS. Para saber qué sistemas operativos tienen disponible una imagen de seguridad mejorada, consulte la Guía de seguridad para clientes de AWS Managed Services (AMS). Para acceder a esta guía, abra AWS Artifact, seleccione Informes en el panel de navegación izquierdo y, a continuación, filtre para ver AWS Managed Services. Para obtener instrucciones sobre cómo acceder AWS Artifact, póngase en contacto con su CSDM o consulte Primeros pasos AWS Artifact para obtener más información.

Términos clave de AMS

- AMS Advanced: los servicios descritos en la sección «Descripción del servicio» de la documentación avanzada de AMS. Consulte la descripción del servicio.
- Cuentas avanzadas de AMS: AWS cuentas que cumplen en todo momento todos los requisitos de incorporación avanzada de AMS. Para obtener información sobre las ventajas de AMS Advanced, los casos prácticos y ponerse en contacto con un representante de ventas, consulte AWS Managed Services.
- Cuentas AMS Accelerate: AWS cuentas que cumplen en todo momento todos los requisitos de incorporación de AMS Accelerate. Consulte Cómo empezar a utilizar AMS Accelerate.
- AWS Managed Services: AMS o AMS Accelerate.
- Cuentas de AWS Managed Services: las cuentas AMS o las cuentas AMS Accelerate.
- Recomendación fundamental: recomendación emitida AWS mediante una solicitud de servicio en la que se le informa de que es necesario tomar medidas para protegerse de posibles riesgos o interrupciones en sus recursos o en los Servicios de AWS mismos. Si decide no seguir una recomendación crítica antes de la fecha especificada, usted es el único responsable de cualquier daño que se derive de su decisión.
- Configuración solicitada por el cliente: cualquier software, servicio u otra configuración que no esté identificada en:
  - Accelerate: configuraciones compatibles o AMS Accelerate; descripción del servicio.
  - AMS Advanced: configuraciones compatibles o AMS Advanced; descripción del servicio.
- Comunicación de incidentes: AMS le comunica un incidente o usted solicita un incidente a AMS a través de un incidente creado en el Support Center de AMS Accelerate y en la consola de AMS para AMS. La consola AMS Accelerate proporciona un resumen de los incidentes y las solicitudes de servicio en el panel de control y enlaces al Support Center para obtener más información.
- Entorno gestionado: las cuentas AMS Advanced o las cuentas AMS Accelerate gestionadas por AMS.

  En el caso de AMS Advanced, se incluyen las cuentas de zona de aterrizaje multicuenta (MALZ) y de zona de aterrizaje de cuenta única (SALZ).
- Fecha de inicio de la facturación: al día hábil siguiente, AWS recibirá la información solicitada en el correo electrónico de incorporación de AWS Managed Services. El correo electrónico de incorporación de AWS Managed Services hace referencia al correo electrónico que usted envía

para recopilar la información necesaria para activar los servicios gestionados de AWS en sus cuentas. AWS

En el caso de las cuentas que usted inscriba posteriormente, la fecha de inicio de la facturación es el día siguiente al envío por parte de AWS Managed Services de una notificación de activación de AWS Managed Services para la cuenta inscrita. Se produce una notificación de activación de AWS Managed Services cuando:

1. Concede acceso a una AWS cuenta compatible y se la entrega a AWS Managed Services.
2. AWS Managed Services diseña y crea la cuenta de AWS Managed Services.

- Finalización del servicio: puede cancelar los servicios gestionados de AWS para todas las cuentas de AWS Managed Services o para una cuenta específica de AWS Managed Services por cualquier motivo avisando con al AWS menos 30 días de antelación mediante una solicitud de servicio. En la fecha de finalización del servicio, puede:
  1. AWS le entrega el control de todas las cuentas de AWS Managed Services o de las cuentas de AWS Managed Services especificadas, según corresponda, o
  2. Las partes eliminan las AWS Identity and Access Management funciones que dan AWS acceso a todas las cuentas de AWS Managed Services o a las cuentas de AWS Managed Services especificadas, según proceda.
- Fecha de finalización del servicio: la fecha de finalización del servicio es el último día del mes natural siguiente al final del período de preaviso de rescisión obligatorio de 30 días. Si el final del período de notificación de rescisión requerido cae después del día 20 del mes calendario, la fecha de finalización del servicio es el último día del mes calendario siguiente. Los siguientes son ejemplos de escenarios para las fechas de terminación.
  - Si el aviso de rescisión se proporciona el 12 de abril, el aviso de 30 días finaliza el 12 de mayo. La fecha de finalización del servicio es el 31 de mayo.
  - Si se proporciona un aviso de rescisión el 29 de abril, el aviso de 30 días finaliza el 29 de mayo. La fecha de finalización del servicio es el 30 de junio.
- Prestación de AWS Managed Services: AWS pone a su disposición los servicios gestionados de AWS Managed Services y los puede utilizar para cada cuenta de AWS Managed Services a partir de la fecha de inicio del servicio.
- Cancelación de cuentas de AWS Managed Services específicas: puede cancelar los servicios de AWS Managed Services de una cuenta específica de AWS Managed Services por cualquier motivo mediante AWS notificación mediante una solicitud de servicio («Solicitud de cancelación de la cuenta AMS»).

Condiciones de gestión de incidentes:

- Evento: un cambio en su entorno de AMS.
- Alerta: cada vez que un evento de un Servicio de AWS dispositivo compatible supera un umbral y activa una alarma, se crea una alerta y se envía un aviso a tu lista de contactos. Además, se crea un incidente en tu lista de incidentes.
- Incidente: una interrupción no planificada o una degradación del rendimiento de su entorno de AMS o AWS Managed Services que se traduce en un impacto según lo informado por AWS Managed Services o por usted.
- Problema: una causa raíz subyacente compartida de uno o más incidentes.
- Resolución de incidentes o resolución de un incidente:
  - AMS ha restablecido todos los servicios o recursos de AMS no disponibles relacionados con ese incidente a un estado disponible, o
  - AMS ha determinado que las pilas o los recursos no disponibles no se pueden restaurar a un estado disponible, o
  - AMS ha iniciado una restauración de la infraestructura autorizada por usted.
- Tiempo de respuesta ante un incidente: diferencia de tiempo entre el momento en que se crea un incidente y el momento en que AMS proporciona una respuesta inicial a través de la consola, el correo electrónico, el centro de servicio o el teléfono.
- Tiempo de resolución de incidentes: la diferencia de tiempo entre el momento en que AMS o usted crean un incidente y el momento en que se resuelve el incidente.
- Prioridad de incidentes: cómo AMS o usted priorizan los incidentes, ya sea baja, media o alta.
  - Baja: un problema no crítico con su servicio AMS.
  - Medio: hay un servicio de AWS disponible en su entorno gestionado, pero no funciona según lo previsto (según la descripción del servicio correspondiente).
  - Alto: (1) la consola AMS o uno o más AMS de su APIs entorno gestionado no están disponibles; o (2) una o más pilas o recursos de AMS de su entorno gestionado no están disponibles y la falta de disponibilidad impide que la aplicación desempeñe su función.

  AMS puede reclasificar los incidentes de acuerdo con las directrices anteriores.
- Restauración de la infraestructura: volver a implementar las pilas existentes, en función de las plantillas de las pilas afectadas, e iniciar una restauración de datos en función del último punto de restauración conocido, a menos que se especifique lo contrario, cuando no sea posible resolver los incidentes.

Términos de infraestructura:

- Entorno de producción gestionado: una cuenta de cliente en la que residen las aplicaciones de producción del cliente.
- Entorno no productivo gestionado: cuenta de cliente que solo contiene aplicaciones que no son de producción, como aplicaciones de desarrollo y pruebas.
- Pila de AMS: grupo de uno o más AWS recursos que AMS administra como una sola unidad.
- Infraestructura inmutable: modelo de mantenimiento de infraestructura típico de los grupos de Amazon EC2 Auto Scaling (ASGs) en el que los componentes de infraestructura actualizados (en AWS la AMI) se sustituyen en cada implementación, en lugar de actualizarse in situ. La ventaja de una infraestructura inmutable es que todos los componentes permanecen en un estado sincrónico, ya que siempre se generan a partir de la misma base. La inmutabilidad es independiente de cualquier herramienta o flujo de trabajo para crear la AMI.
- Infraestructura mutable: modelo de mantenimiento de infraestructura típico de las pilas que no son grupos de Amazon EC2 Auto Scaling y que contienen una sola instancia o solo unas pocas instancias. Este modelo es el que más se parece a la implementación de sistemas tradicional, basada en hardware, en la que un sistema se implementa al principio de su ciclo de vida y, posteriormente, las actualizaciones se van añadiendo capas a ese sistema a lo largo del tiempo. Todas las actualizaciones del sistema se aplican a las instancias de forma individual y pueden provocar un tiempo de inactividad del sistema (según la configuración de la pila) debido al reinicio de la aplicación o del sistema.
- Grupos de seguridad: firewalls virtuales para que la instancia controle el tráfico entrante y saliente. Los grupos de seguridad actúan en el ámbito de la instancia, no en el de la subred. Por lo tanto, cada instancia de una subred de la VPC podría tener asignado un conjunto diferente de grupos de seguridad.
- Acuerdos de nivel de servicio (SLAs): parte de los contratos de AMS con usted que definen el nivel de servicio esperado.
- El SLA no está disponible y no está disponible:
  - Una solicitud de API enviada por usted que produce un error.
  - Una solicitud de consola enviada por ti que da como resultado una respuesta HTTP de cinco veces mayor (el servidor no puede ejecutar la solicitud).
  - Cualquiera de las Servicio de AWS ofertas que constituyen pilas o recursos en su infraestructura gestionada por AMS se encuentra en un estado de «interrupción del servicio», como se muestra en el Service Health Dashboard.

  - La falta de disponibilidad que resulte directa o indirectamente de una exclusión de AMS no se tiene en cuenta al determinar la elegibilidad para los créditos de servicio. Los servicios se consideran disponibles a menos que cumplan con los criterios para no estar disponibles.
- Objetivos de nivel de servicio (SLOs): parte de los contratos de AMS con usted que definen objetivos de servicio específicos para los servicios de AMS.

Términos de aplicación de parches:

- Parches obligatorios: actualizaciones de seguridad críticas para abordar problemas que podrían comprometer el estado de seguridad de su entorno o cuenta. Una «actualización de seguridad crítica» es una actualización de seguridad calificada como «crítica» por el proveedor de un sistema operativo compatible con AMS.
- Parches anunciados frente a parches publicados: los parches se anuncian y publican por lo general según un calendario. Los parches emergentes se anuncian cuando se descubre la necesidad de aplicarlos y, por lo general, poco después de su publicación.
- Complemento de parche: parche basado en etiquetas para instancias de AMS que aprovecha la funcionalidad AWS Systems Manager (SSM) para que pueda etiquetar instancias y parchearlas utilizando una línea base y una ventana que usted configure.
- Métodos de parches:
  - Parcheo in situ: parcheo que se realiza cambiando las instancias existentes.
  - Parcheo de reemplazo de la AMI: aplicación de parches que se realiza cambiando el parámetro de referencia de la AMI de una configuración de lanzamiento de grupo de EC2 Auto Scaling existente.
- Proveedor de parches (proveedores de sistemas operativos, terceros): los parches los proporciona el proveedor o el órgano rector de la aplicación.
- Tipos de parches:
  - Actualización de seguridad crítica (CSU): actualización de seguridad calificada como «crítica» por el proveedor de un sistema operativo compatible.
  - Actualización importante (IU): una actualización de seguridad calificada como «importante» o una actualización no relacionada con la seguridad calificada como «crítica» por el proveedor de un sistema operativo compatible.
  - Otra actualización (OU): actualización realizada por el proveedor de un sistema operativo compatible que no es una CSU ni una IU.

- Parches compatibles: AMS admite parches a nivel de sistema operativo. El proveedor publica actualizaciones para corregir vulnerabilidades de seguridad u otros errores o para mejorar el rendimiento. Para obtener una lista de las configuraciones compatibles actualmente OSs, consulte Support Configurations.

Términos de seguridad:

- Controles de Detective: una biblioteca de monitores creados o habilitados por AMS que proporcionan una supervisión continua de los entornos y las cargas de trabajo gestionados por los clientes para detectar configuraciones que no se ajustan a los controles de seguridad, operativos o del cliente, y toman medidas al notificar a los propietarios, modificar de forma proactiva o cancelar los recursos.

Condiciones de la solicitud de servicio:

- Solicitud de servicio: una solicitud suya para que AMS tome una medida en su nombre.
- Notificación de alerta: aviso que AMS publica en su página de lista de solicitudes de servicio cuando se activa una alerta de AMS. El contacto configurado para su cuenta también recibe una notificación mediante el método configurado (por ejemplo, el correo electrónico). Si tiene etiquetas de contacto en sus instancias o recursos y ha dado su consentimiento a su administrador de prestación de servicios en la nube (CSDM) para las notificaciones basadas en etiquetas, la información de contacto (valor clave) de la etiqueta también se notificará para recibir alertas de AMS automatizadas.
- Notificación de servicio: una notificación de AMS que se publica en la página de la lista de solicitudes de servicios.

Términos varios:

- Interfaz de AWS Managed Services: para AMS: la consola avanzada de AWS Managed Services, la API CM de AMS y la Soporte API. Para AMS Accelerate: la Soporte consola y la Soporte API.
- Satisfacción del cliente (CSAT): El CSAT de AMS se basa en análisis exhaustivos, que incluyen valoraciones de casos por correspondencia en cada caso o correspondencia, cuando se presentan, encuestas trimestrales, etc.
- DevOps: DevOps es una metodología de desarrollo que aboga firmemente por la automatización y la supervisión en todos los pasos. DevOps tiene como objetivo acortar los ciclos de desarrollo, aumentar la frecuencia de despliegue y ofrecer versiones más fiables, ya que reúne las funciones

de desarrollo y operaciones, que tradicionalmente estaban separadas, sobre la base de la automatización. Cuando los desarrolladores pueden gestionar las operaciones y las operaciones sirven de base para el desarrollo, los problemas se descubren y resuelven más rápidamente y los objetivos empresariales se alcanzan más fácilmente.

- ITIL: La biblioteca de infraestructura de tecnología de la información (denominada ITIL) es un marco de ITSM diseñado para estandarizar el ciclo de vida de los servicios de TI. ITIL se organiza en cinco etapas que cubren el ciclo de vida de los servicios de TI: estrategia del servicio, diseño del servicio, transición del servicio, operación del servicio y mejora del servicio.
- Administración de servicios de TI (ITSM): conjunto de prácticas que alinean los servicios de TI con las necesidades de su empresa.
- Servicios de monitorización gestionados (MMS): AMS opera su propio sistema de monitorización, Managed Monitoring Service (MMS), que consume eventos de AWS salud y agrega datos de CloudWatch Amazon y datos de Servicios de AWS otros, y notifica a los operadores de AMS (en línea las 24 horas del día, los 7 días de la semana) cualquier alarma creada a través de un tema del Amazon Simple Notification Service (Amazon SNS).
- Espacio de nombres: cuando crea políticas de IAM o trabaja con Amazon Resource Names (ARNs), identifica y utiliza un espacio Servicio de AWS de nombres. Los espacios de nombres se utilizan para identificar acciones y recursos.

¿Cuál es mi modelo operativo?

Como cliente de AMS, su organización ha decidido separar las operaciones de aplicaciones de las de infraestructura y utilizar AMS para las operaciones de infraestructura. AMS trabajará con su equipo de diseño y desarrollo de aplicaciones y con su equipo de diseño de infraestructuras para garantizar que sus operaciones de infraestructura funcionen sin problemas. El siguiente gráfico ilustra este concepto:

AMS asume la responsabilidad de las operaciones de su AWS infraestructura, mientras que sus equipos son responsables de las operaciones de sus aplicaciones. Como equipos de diseño de aplicaciones e infraestructuras, deben saber quién operará la aplicación una vez que se haya implementado en producción en la infraestructura de AMS. Esta guía describe los enfoques más comunes del diseño de infraestructuras en lo que respecta a la implementación y el mantenimiento de las aplicaciones.

Administración de servicios en AWS Managed Services

Temas

Cómo funciona el servicio AMS para usted.

Gestión de cuentas en AWS Managed Services

En esta sección se describe la gobernanza de las cuentas de AMS.

Se le designa como administrador de prestación de servicios en la nube (CSDM), que proporciona asesoramiento en todo AMS y tiene un conocimiento detallado de su caso de uso y de la arquitectura tecnológica para el entorno gestionado. CSDMs colabore con los administradores de cuentas, los administradores técnicos de cuentas, los arquitectos de nube de AWS Managed Services (CAs) y los arquitectos de soluciones de AWS (SAs), según proceda, para ayudar a lanzar nuevos proyectos y ofrecer recomendaciones sobre las mejores prácticas a lo largo de los procesos de desarrollo y operaciones del software. El CSDM es el principal punto de contacto de AMS. Las principales responsabilidades de su CSDM son:

- Organice y dirija reuniones mensuales de revisión del servicio con los clientes.
- Proporcione detalles sobre la seguridad, las actualizaciones de software para el entorno y las oportunidades de optimización.
- Defienda sus requisitos, incluidas las solicitudes de funciones para AMS.
- Responda y resuelva las solicitudes de informes de facturación y servicio.
- Proporcione información para recomendaciones de optimización financiera y de capacidad.

Inicio del servicio en AWS Managed Services

Inicio del servicio: la fecha de inicio del servicio de una cuenta de AWS Managed Services es el primer día del primer mes natural después del cual AWS le notifica que se han completado las actividades establecidas en los requisitos de incorporación de esa cuenta de AWS Managed Services; siempre que si AWS realiza dicha notificación después del día 20 de un mes natural, la fecha de inicio del servicio será el primer día del segundo mes natural siguiente a la fecha de dicha notificación.

Inicio del servicio

- R significa parte responsable que hace el trabajo para lograr la tarea.
- La letra I significa «informado»: una parte que recibe información sobre el progreso, a menudo solo una vez finalizada la tarea o el producto final.

Inicio del servicio

N.° de paso	Título del paso	Descripción	Cliente	AMS
1.	Entrega de la cuenta de AWS al cliente	El cliente crea una nueva cuenta de AWS y se la entrega a AWS Managed Services	R	I
2.	Cuenta AWS Managed Services: diseño	Finalice el diseño de la cuenta de AWS Managed Services	I	R
3.	Cuenta AWS Managed Services: creación	Se crea una cuenta de AWS Managed Services según el diseño del paso 2	I	R

Gestión de relaciones con los clientes (CRM)

AWS Managed Services (AMS) proporciona un proceso de administración de las relaciones con los clientes (CRM) para garantizar que se establezca y mantenga una relación bien definida con

usted. La base de esta relación se basa en el conocimiento que AMS tiene de las necesidades de su empresa. El proceso de CRM facilita una comprensión precisa y completa de:

- Las necesidades de su empresa y cómo satisfacerlas
- Sus capacidades y limitaciones
- AMS y sus diferentes responsabilidades y obligaciones

El proceso de CRM permite a AMS utilizar métodos coherentes para prestarle servicios y gestionar su relación con AMS. El proceso de CRM incluye:

- Identificar a las partes interesadas clave
- Establecer un equipo de gobierno
- Realización y documentación de reuniones de revisión del servicio con usted
- Proporcionar un procedimiento formal de quejas de servicio con un procedimiento de escalamiento
- Implementar y monitorear su proceso de satisfacción y comentarios
- Gestionar su contrato

Proceso de CRM

El proceso de CRM incluye las siguientes actividades:

- Identificar y comprender los procesos y necesidades de su negocio. Su acuerdo con AMS identifica a sus partes interesadas.
- Definir los servicios que se prestarán para satisfacer sus necesidades y requisitos.
- Reunirnos con usted durante las reuniones de revisión del servicio para analizar cualquier cambio en el alcance del servicio de AMS, el SLA, el contrato y las necesidades de su empresa. Es posible que se celebren reuniones provisionales con usted para analizar el desempeño, los logros, los problemas y los planes de acción.
- Controle su satisfacción mediante nuestra encuesta de satisfacción del cliente y los comentarios proporcionados en las reuniones.
- Informar sobre el rendimiento mediante informes mensuales de rendimiento medidos internamente.
- Revisar el servicio con usted para determinar las oportunidades de mejora. Esto incluye una comunicación frecuente con usted en relación con el nivel y la calidad del servicio de AMS prestado.

Reuniones de CRM

Los gerentes de prestación de servicios en la nube de AMS (CSDMs) se reúnen con usted periódicamente para hablar sobre las áreas de servicio (operaciones, seguridad e innovaciones de productos) y ejecutivas (informes de acuerdos de nivel de servicio, medidas de satisfacción y cambios en las necesidades de su empresa).

Reunión	Finalidad	Mode	Participantes
Revisión de estado semanal (opcional)	Problemas o incidentes pendientes, parches, eventos de seguridad, registros de problemas Tendencia operativa de 12 semanas (+/- 6) Dudas del operador de la aplicación Horario de fin de semana	Cliente presencial location/ Telecom/Chime	AMS: CSDM y arquitecto de nube (CA) Miembros del equipo asignados por el cliente (p. ej.: equipos de Cloud/ Infrastructure, Application Support, Architecture, etc.)
Revisión empresarial mensual	Revise el rendimiento del nivel de servicio (informes, análisis y tendencias) Análisis financiero Hoja de ruta del producto ELENCO	Cliente presencial location/ Telecom/Chime	AMS: CSDM, arquitecto de nube (CA), equipo de cuentas de AMS, gerente de productos técnicos de AMS (TPM) (opcional), gerente de OPS de AMS (opcional)

Reunión	Finalidad	Mode	Participantes
			Usted: representante del operador de aplicaciones
Revisión comercial trimestral	Rendimiento y tendencias del cuadro de mando y del acuerdo de nivel de servicio (SLA) (6 meses) Planes o migraciones para los próximos 3, 6 y 9 meses Riesgos y mitigaciones de riesgos Iniciativas clave de mejora Elementos de la hoja de ruta del producto Oportunidades alineadas con la dirección futura Finanzas Iniciativas de ahorro de costos Optimización empresarial	Ubicación in situ del cliente	AMS: CSDM, arquitecto de la nube, equipo de cuentas de AMS, director de servicios de AMS, gerente de operaciones de AMS Usted: representante del operador de aplicaciones, representante de servicio, director de servicio

Organización de reuniones de CRM

El CSDM de la AMS es responsable de documentar la reunión, lo que incluye:

- Crear la agenda, incluidos los puntos de acción, los temas y la lista de asistentes.
- Crear la lista de puntos de acción revisados en cada reunión para garantizar que los puntos se completen y se resuelvan según lo programado.
- Distribuir las actas de la reunión y la lista de elementos de acción a los asistentes a la reunión por correo electrónico en el plazo de un día hábil después de la reunión.

- Almacenar las actas de las reuniones en el repositorio de documentos correspondiente.

En ausencia del CSDM, el representante de la AMS que dirige la reunión crea y distribuye las actas.

> **Note**
>
> Su CSDM trabaja con usted para establecer el gobierno de su cuenta.

Informes mensuales de CRM

Su AMS CSDM prepara y envía presentaciones mensuales sobre el rendimiento del servicio. Las presentaciones incluyen información sobre lo siguiente:

- Fecha del informe
- Resumen e información:
  - Información clave: recuento total y activo de las pilas, estado de las actualizaciones en las pilas, estado de incorporación de la cuenta (solo durante la incorporación), resúmenes de los problemas específicos de los clientes
  - Rendimiento: estadísticas sobre la resolución de incidentes, las alertas, la aplicación de parches, las solicitudes de cambio (RFCs), las solicitudes de servicio y la disponibilidad de la consola y la API
  - Problemas, desafíos, inquietudes y riesgos: estado de los problemas específicos del cliente
  - Próximos temas: planes de incorporación o resolución de incidentes específicos para cada cliente
- Recursos gestionados: gráficos y diagramas circulares de pilas
- Métricas de AMS: métricas de monitoreo y eventos, métricas de incidentes, métricas de cumplimiento de los SLA de AMS, métricas de solicitudes de servicio, métricas de administración de cambios, métricas de almacenamiento, métricas de continuidad, métricas de Trusted Advisor y resúmenes de costos (presentados de varias maneras). Solicitudes de funciones. Información de contacto.

> **Note**
>
> Además de la información descrita, su CSDM también le informa de cualquier cambio sustancial en el alcance o las condiciones, incluido el uso de subcontratistas por parte de AMS para actividades operativas.
> AMS genera informes sobre la aplicación de parches y las copias de seguridad que su CSDM incluye en su informe mensual. Como parte del sistema de generación de informes, AMS añade cierta infraestructura a tu cuenta a la que no puedes acceder:
>
> - Un bucket de S3, con los datos sin procesar informados
> - Una instancia de Athena, con definiciones de consulta para consultar los datos
> - Un Glue Crawler para leer los datos sin procesar del depósito S3

Optimización de costos en AWS Managed Services

AWS Managed Services le proporciona informes detallados de costos, utilización y ahorros todos los meses durante las revisiones empresariales mensuales (MBRs).

AMS sigue un conjunto estándar de procesos y mecanismos para identificar vías de ahorro de costes en sus cuentas gestionadas y ayudarle a planificar e implementar los cambios para optimizar su gasto en AWS.

> **Note**
>
> AMS está desarrollando un vídeo para ayudar a optimizar los costes. El primer paso es proporcionarle una hoja de cálculo en PDF y Excel con las mejores prácticas de optimización de costes. Para acceder a estos recursos, abra el archivo ZIP de la guía rápida para la optimización de costos.

Marco de optimización de costos

AMS sigue un enfoque de tres etapas con usted para optimizar sus costos de AWS:

1. Identifique las vías de optimización de costes en su entorno gestionado
2. Preséntele un plan de optimización de costos
3. Ayude a lograr la optimización de costos de una manera mensurable

Identifique las vías de optimización de costos en el entorno gestionado

AMS utiliza herramientas AWS nativas, como Cost Explorer y Trusted Advisor, a la vez que aprovecha más de 20 patrones de ahorro de costos en la optimización de la arquitectura, las optimizaciones AWS centradas en las EC2 instancias y las cuentas para crear recomendaciones de ahorro de costos personalizadas para usted.

Algunas de las recomendaciones de optimización incluyen las siguientes.

Recomendaciones de optimización arquitectónica:

- Uso óptimo de la clase de almacenamiento S3: Amazon S3 ofrece una variedad de clases de almacenamiento para cumplir con varios requisitos de carga de trabajo en función del acceso a los datos, la resiliencia y el costo. Los análisis de las clases de almacenamiento S3 Intelligent-Tiering y S3 basados en las necesidades de carga de trabajo le permiten administrar los costos de S3 de manera eficiente.
- Uso de arquitecturas de almacenamiento en caché: aprovechar las instancias de caché, cuando proceda, puede ayudarlo a reemplazar algunas instancias de bases de datos y, al mismo tiempo, cumplir con sus requisitos de IOPS.
- Ahorros en la actualización de EBS: la migración de sus volúmenes de EBS de gp2 a gp3 proporciona un ahorro de costos de hasta un 20% y puede aprovechar un rendimiento básico predecible de 3000 IOPS y 125 MiB/s, independientemente del tamaño del volumen.
- Uso de la elasticidad: las capacidades de autoescalado que AWS proporciona permiten una utilización eficaz de los recursos y vías para la optimización de los costos. Revisar y actualizar las políticas de escalado de instancias con regularidad en función de las necesidades también permite ahorrar costos.

EC2 recomendaciones centradas en las instancias

- Ajuste del tamaño de las instancias: las recomendaciones se centraron en el tamaño de las instancias y en las configuraciones óptimas en función del uso. Las recomendaciones también incluyen el uso de la función Amazon EC2 Auto Scaling y la sustitución de las EC2 instancias, cuando proceda, por contenido web estático en Amazon S3, etc. AWS Lambda
- Programación de instancias: el uso del programador de recursos de AMS para iniciar y detener automáticamente las instancias en función de un cronograma ayuda a reducir los costos, especialmente en el caso de las instancias que no son de producción y que no se utilizan fuera del horario laboral.

- Suscribirse a planes de ahorro: el plan de ahorro es la forma más fácil de ahorrar en el consumo. AWS Los EC2 Instance Savings Plans ofrecen hasta un 72% de ahorro en comparación con los precios bajo demanda en el uso de EC2 instancias de Amazon. Los Amazon SageMaker AI Savings Plans ofrecen hasta un 64% de ahorro en el uso de los servicios de Amazon SageMaker AI. AMS ofrece las recomendaciones adecuadas sobre los planes de ahorro en función del uso AWS de los recursos.
- Guía de uso y consumo de instancias reservadas (RI): las instancias EC2 reservadas (RI) de Amazon ofrecen un descuento significativo (hasta un 75%) en comparación con los precios bajo demanda y ofrecen una reserva de capacidad cuando se utilizan en una zona de disponibilidad específica.
- Uso de instancias puntuales: las cargas de trabajo tolerantes a errores pueden utilizar instancias puntuales y reducir los precios hasta un 90%.
- Terminación de instancias inactivas: identificar e informar sobre las instancias que están inactivas o de baja utilización y que se pueden cerrar.

Recomendaciones centradas en las cuentas

- Limpieza de cuentas: a nivel de cuentas, AMS también identifica los volúmenes de EBS no utilizados, los registros duplicados CloudTrail , las cuentas vacías con recursos no utilizados, etc., y ofrece recomendaciones para la limpieza.
- Recomendaciones de SLA: Además, AMS revisa periódicamente sus cuentas Plus y Premium y recomienda elegir el nivel de SLA adecuado para las cuentas.
- Optimización de la automatización de AMS: AMS optimiza continuamente la automatización y la infraestructura de AMS utilizadas para proporcionar los servicios de AMS.

Preséntelo a los clientes y ayúdeles en la planificación

AMS realiza revisiones comerciales mensuales (MBRs) con las principales partes interesadas de los clientes y presenta las vías, los mecanismos y las recomendaciones de ahorro de costos identificadas, junto con los posibles ahorros de costos. Seguiremos trabajando con usted para planificar los cambios necesarios.

Ayudamos en la implementación de las recomendaciones y medimos el impacto en los costos

El AMS ayuda a lograr y medir los impactos en los costos y los cambios de optimización.

Usted evalúa el impacto en la aplicación, el riesgo y los criterios de éxito de los cambios recomendados y formula las solicitudes de cambio adecuadas (RFCs) a través de la consola AMS. AMS colabora con usted e implementa los cambios relacionados con la optimización de costes en sus cuentas gestionadas. AMS mide el impacto en los costes e incluye los ahorros obtenidos en las revisiones empresariales mensuales (MBRs).

Matriz de responsabilidad de optimización de costes

Responsabilidades en la optimización de costes de AMS.

Optimización de costes (RACI)

Actividad	Cliente	AMS
Compilación de recomendaciones de ahorro de costes y preparación del informe	I	R
Presentación del informe de ahorro de costos	C	R
Planificar los cambios asociados al ahorro de costes	R	C
Evaluar el	R	C

Actividad	Cliente	AMS
impacto y el riesgo del cambio		
Recaudar RFCs para implementar los cambios	R	C
Revisar RFCs e implementar los cambios	C	R
Probar la aplicación y validar la implementación del cambio	R	C
Medir el impacto en los costos después del cambio y presentarlo al cliente	I	R

Horas de servicio en AWS Managed Services

Característica	AMS Advanced
	Nivel premium
Solicitud de servicio	24/7
Gestión de incidentes (P2-P3)	24/7
Copia de seguridad y recuperación	24/7
Administración de parches	24/7
Monitorización y alertas	24/7
Solicitud de cambio automatizada (RFC)	24/7
Solicitud de cambio no automatizada (RFC)	24/7
Gestor de prestación de servicios en la nube (CSDM)	De lunes a viernes: de 08:00 a 17:00, horario comercial local

Obtener ayuda en AWS Managed Services

AMS lo apoya en la gestión de incidentes, solicitudes de servicio y cambios las 24 horas del día, los 7 días de la semana y los 365 días del año (de conformidad con el acuerdo de nivel de servicio de AMS que se aplique a la cuenta).

Para informar de un problema de rendimiento de los servicios de AWS o AMS que afecte a su entorno gestionado, utilice la consola AMS y envíe un informe de incidente. Para obtener más información, consulte Notificación de un incidente. Para obtener información general sobre la gestión de incidentes de AMS, consulte Respuesta a incidentes.

Para solicitar información o consejos, o para solicitar servicios adicionales a AMS, utilice la consola de AMS y envíe una solicitud de servicio. Para obtener más información, consulte Creación de una solicitud de servicio. Para obtener información general sobre las solicitudes de servicio de AMS, consulte Administración de solicitudes de servicio.

Desarrollo de aplicaciones

Procesos y prácticas de desarrollo de aplicaciones que permiten el diseño y la implementación efectivos de aplicaciones en un entorno de AWS Managed Services (AMS). AMS lo guía a través del siguiente proceso de alto nivel:

1. Imagine y diseñe una aplicación para desarrollarla o integrarla en su entorno gestionado por AMS. Algunas consideraciones:

    a. ¿Cómo va a implementar su aplicación? ¿Con la automatización mediante una herramienta de despliegue como Ansible o manualmente cargando directamente los archivos necesarios?

    b. ¿Cómo actualizará su aplicación? ¿Con un enfoque mutable que actualiza cada instancia por separado o con un enfoque inmutable que actualiza cada instancia con una sola AMI actualizada en un grupo de Auto Scaling?

2. Planifique y diseñe la infraestructura que se utilizará para alojar la aplicación utilizando bibliotecas de AWS arquitectura, orientación sobre AWS «Well-Architected» y expertos en la materia de AMS y otros temas de arquitectura de nube. En las siguientes secciones de esta guía se proporciona información que puede ayudar en este sentido.
3. Seleccione un enfoque de despliegue de infraestructura:

    a. Paquete completo: todos los componentes de la infraestructura se implementan a la vez, juntos.

    b. Nivel y vínculo: las implementaciones de infraestructura se implementan por separado y, después, se combinan con las modificaciones de los grupos de seguridad. Este tipo de despliegue también se logra mediante una configuración en serie de los componentes de la pila que se construyen unos sobre otros; por ejemplo, especificando el balanceador de carga que creó anteriormente al crear un grupo de Auto Scaling.

    c. ¿Qué entornos, como Dev, Staging y Prod, va a emplear?

4. Elija los tipos de cambio de AMS (CTs) que aprovisionarán las pilas o niveles necesarios y prepararán las solicitudes de cambio necesarias (). RFCs
5. Envíe el RFCs para iniciar el despliegue de la infraestructura en el entorno adecuado.
6. Implemente la aplicación utilizando el enfoque de implementación de aplicaciones seleccionado.
7. Rediseñe la infraestructura y las aplicaciones según sea necesario.

8. Implemente la infraestructura y las aplicaciones en los entornos de seguimiento adecuados, suponiendo que la primera implementación se realice en un entorno que no sea de producción.
9. El mantenimiento continuo está a cargo de AMS, que opera la infraestructura subyacente, y de sus equipos de operaciones, de las infraestructuras de aplicaciones.
10. Para dar de baja una aplicación, cancele la infraestructura de AMS correspondiente.

Tener una buena arquitectura

En AWS creemos que los sistemas bien diseñados aumentan en gran medida las probabilidades de éxito empresarial. El Centro de AWS Arquitectura ofrece orientación experta sobre la arquitectura en el. Nube de AWS

Recomendamos los siguientes artículos y documentos técnicos para ayudarlo a comprender las ventajas y desventajas de las decisiones que debe tomar al construir sistemas. AWS

¿Eres Well-Architected? : Presenta el marco AWS Well-Architected, que se basa en seis pilares:

- Excelencia operativa: el pilar de la excelencia operativa se centra en el funcionamiento y la supervisión de los sistemas para ofrecer valor empresarial y en la mejora continua de los procesos y procedimientos. Los temas clave incluyen la gestión y la automatización de los cambios, la respuesta a los eventos y la definición de estándares para gestionar con éxito las operaciones diarias.
- Seguridad: el pilar de seguridad se centra en la protección de la información y los sistemas. Los temas clave incluyen la confidencialidad e integridad de los datos, la identificación y la administración de quién puede hacer qué con la administración de permisos, la protección de los sistemas y el establecimiento de controles para detectar eventos de seguridad.
- Fiabilidad: el pilar de la confiabilidad se centra en la capacidad de prevenir las fallas y recuperarse rápidamente de ellas para satisfacer la demanda empresarial y de los clientes. Los temas clave incluyen los elementos fundamentales relacionados con la configuración, los requisitos de todos los proyectos, la planificación de la recuperación y la forma en que gestionamos los cambios.
- Eficiencia del rendimiento: el pilar de la eficiencia del rendimiento se centra en el uso eficiente de los recursos informáticos y de TI. Los temas clave incluyen la selección de los tipos y tamaños de recursos correctos en función de los requisitos de la carga de trabajo, la supervisión del rendimiento y la toma de decisiones informadas para mantener la eficiencia a medida que evolucionan las necesidades empresariales.

- Optimización de costes: el pilar de optimización de costes se centra en evitar costes innecesarios. Los temas clave incluyen comprender y controlar dónde se gasta el dinero, seleccionar los tipos de recursos más adecuados y adecuados, analizar el gasto a lo largo del tiempo y ampliarlo para satisfacer las necesidades empresariales sin gastar de más.
- Sostenibilidad: el pilar de la sostenibilidad se centra en la capacidad de mejorar continuamente los impactos en la sostenibilidad mediante la reducción del consumo de energía y el aumento de la eficiencia en todos los componentes de una carga de trabajo, maximizando los beneficios de los recursos aprovisionados y minimizando los recursos totales necesarios.

AWS Well-Architected Framework: describe AWS cómo permite a los clientes evaluar y mejorar sus arquitecturas basadas en la nube y comprender mejor el impacto empresarial de sus decisiones de diseño. Aborda los principios generales de diseño, así como las mejores prácticas y directrices específicas en seis áreas conceptuales que se AWS definen como los pilares del Marco de Trabajo de Buena Arquitectura.

Responsabilidades de la capa de aplicación frente a responsabilidades de la capa de infraestructura en AMS

Al usar AMS, AMS mantiene su infraestructura y todo lo que necesita para su mantenimiento y crecimiento. Sin embargo, usted desarrolla, despliega y mantiene todo lo que necesite para line-of-business aplicaciones o aplicaciones de productos.

Con la ayuda de herramientas de implementación de aplicaciones, como CodeDeploy and, o Chef CloudFormation, Puppet, Ansible o Saltstack, la implementación de las aplicaciones en la infraestructura gestionada por AMS puede automatizarse por completo.

Para obtener más información sobre lo que hace y lo que no hace AMS, consulte. Lo que hacemos, lo que no hacemos

Mutabilidad de EC2 instancias de Amazon en AMS

Usted y AMS pueden mantener las instancias de Amazon Elastic Compute Cloud (Amazon EC2) en su infraestructura de dos maneras:

- Inmutable: este modelo utiliza Amazon Machine Images (AMIs) horneadas (creadas) con las funciones necesarias. Al implementar una actualización, las instancias existentes se desmontan y se sustituyen por completo por otras nuevas creadas a partir de una AMI actualizada. Para

minimizar el tiempo de inactividad, este proceso continuo hace que algunas instancias no estén actualizadas ni accesibles, mientras que otras se actualizan hasta que, finalmente, el nuevo cambio se implemente por completo.

- Mutable: en este modelo, la infraestructura se actualiza con el nuevo código que se implementa en los sistemas existentes en la nube. Este modelo es una combinación de enviar actualizaciones manualmente y utilizarlas infrastructure-as-code para implementarlas, y no se basa en nada nuevo AMIs.

Estos modelos de mantenimiento se analizan con más detalle en secciones posteriores de esta guía.

Uso de AWS Secrets Manager con los recursos de AMS

Hay muchos casos en los que es posible que necesite compartir secretos con AMS, por ejemplo:

- Restablecimiento de la contraseña maestra para la instancia de RDS
- Certificados para balanceadores de carga
- Obtención de credenciales de larga duración para los usuarios de IAM de AMS

La forma más segura de compartir información confidencial con AMS es a través del AWS Secrets Manager; siga estos pasos:

1. Inicie sesión en la AWS consola con su acceso federado y el CustomerReadOnly rol de zona de aterrizaje con una sola cuenta (SALZ); utilice cualquiera de estos roles y AWSManaged ServicesSecurityOpsRole AWSManagedServicesAdminRole, AWSManaged ServicesChangeManagementRole para la zona de aterrizaje con varias cuentas (MALZ).
2. Diríjase a la consola del administrador de AWS Secrets y haga clic en Guardar un secreto nuevo.
3. Seleccione «Otro tipo de secretos».
4. Introduzca el valor secreto como texto plano y haga clic en Siguiente.
5. Introduzca el nombre y la descripción del secreto. El nombre siempre debe empezar por "customer-shared/*». Por ejemplo, "customer-shared/license-2018». Una vez que haya terminado, continúe haciendo clic en Siguiente.
6. Utilice el cifrado KMS predeterminado.
7. Deje la rotación automática desactivada y haga clic en Siguiente.
8. Revisa y haz clic en Guardar para guardar el secreto.

9. Respóndanos en una solicitud de servicio AMS con el nombre secreto y el ARN para que podamos identificarlo y recuperarlo. Para obtener información sobre cómo crear solicitudes de servicio, consulte los ejemplos de solicitudes de servicio.

Despliegue de aplicaciones en AMS

Durante la incorporación, AWS Managed Services (AMS) trabaja con usted para determinar la infraestructura que necesita.

La infraestructura básica incluye una nube privada AWS virtual (VPC), seguridad de las comunicaciones a través de un fideicomiso forestal ADFS, las subredes básicas (DMZ, servicios compartidos y privada) reflejadas en dos zonas de disponibilidad y configuradas con una NAT gestionada, bastiones, balanceadores de carga públicos (DX) y la seguridad requerida. Direct Connect Los recursos de sus aplicaciones se desplegarán en su subred privada o subred de aplicaciones de clientes. Puede obtener más información sobre una arquitectura AMS típica en la Guía del usuario de AWS Managed Services.

La infraestructura que implemente, una vez que haya completado los aspectos básicos, debe incluir todos los componentes de sus aplicaciones y de su desarrollo.

Capacidades de despliegue de aplicaciones en AMS

Algunas de las formas de implementar aplicaciones en AMS. A continuación se ofrecen detalles sobre cada método.

Ejemplos de capacidades de despliegue de aplicaciones

Método del método	Despliegue de infraestructura	AMI o elemento (s) clave	Instalación de la aplicación
Aplicaciones mutables, AMS AMI			
Despliegue manual de aplicaciones	Full Stack CT o Tier and Tie CTs	AMI proporcionada por AMS	Envíe el CT de administración de acceso e instale la aplicación manualmente.
UserData despliegue de aplicaciones con un agente de aplicaciones (por			Utilice el aprovisionamiento de CT con UserData secuencias de comandos que

Método del método	Despliegue de infraestructura	AMI o elemento (s) clave	Instalación de la aplicación
ejemplo, Chef, Puppet, etc.)			instalen un agente de aplicación y que script/agent instalen la aplicación.
UserData Despliegue de aplicaciones sin agentes (por ejemplo, Ansible, Salt, SSH, etc.)			Envíe el CT de administración de acceso e instale el agente de aplicaciones. Implemente la aplicación con las herramientas de implementación de aplicaciones.
Aplicaciones mutables, AMI personalizada			
Implementación personalizada de aplicaciones AMI (no ASG)	Compilación completa: CT o Tier and Tie CTs	AMI personalizada. AMI AMS -> personalizar con el agente de herramientas de implementación de aplicaciones -> crear EC2 instancia (CT) -> crear AMI (CT).	Las herramientas de despliegue de aplicaciones (por ejemplo, Chef), que aprovechan los agentes, despliegan la aplicación.
Implementación de la aplicación AWS Database Migration Service (DMS)	Sincronización de AWS DMS con la pila de bases de datos relacionales AMS existente.	AMI personalizada	El cliente o socio emplea AWS Database Migration Service; AMS verifica los componentes de AMS en el momento del lanzamiento

Método del método	Despliegue de infraestructura	AMI o elemento (s) clave	Instalación de la aplicación
Implementación de la aplicación Workload Ingest	Workload Ingest CT, migrada por socios instance/AMI e iniciada por el cliente.		El socio migra la instancia y crea una AMI en la VPC administrada por AMS del cliente; el cliente usa Workload Ingest CT para lanzar la pila en AMS. Para obtener más información, consulte Ingesta de carga de trabajo de AMS (WIGS).
Aplicaciones inmutables			
Despliegue personalizado de aplicaciones AMI (ASG)	Compilación completa: CT o Tier and Tie CTs	AMI de AMS -> personalizar -> crear EC2 instancia (CT) -> crear AMI (CT) -> crear un grupo de Auto Scaling.	Auto Scaling implementa la aplicación con la AMI personalizada Para obtener más información, consulte Implementaciones de aplicaciones Tier y Tie en AMS.
Aplicaciones mutables o inmutables			

Método del método	Despliegue de infraestructura	AMI o elemento (s) clave	Instalación de la aplicación
Despliegue de aplicaciones de plantilla personalizada CloudFormation	CloudFormation plantilla	CloudFormation Plantilla AWS -> customize/ prepare para AMS -> Implementación | Ingestión | Apilación a partir de CloudFormation plantilla | Crear (ct-36cn2avfrrj9v).	AMS implementa la aplicación en su cuenta mediante su plantilla personalizada y valida la implementación de la aplicación. CloudFormation Para obtener más información, consulte CloudFormation Ingesta de AMS.
Importación de bases de datos SQL	Operaciones AMS (Otras | Otras CT)	Base de datos SQL local -> archivo.bak -> Base de datos SQL de AMS RDS -> Administración | Otros | Otros | Otros | Crear (ct-1e1xtak34nx76) para la importación.	AMS importa la base de datos local a la base de datos RDS gestionada por AMS. Para obtener más información, consulte Importación de bases de datos (DB) a AMS RDS para Microsoft SQL Server.
Database Migration Service (DMS)	Operaciones de AMS (múltiples CTs)	Base de datos local -> Instancia de replicación del DMS -> grupo de subredes de replicación del DMS -> punto final de destino del DMS -> punto final de origen del DMS -> tarea de replicación del DMS.	AMS importa la base de datos local a la base de datos S3 gestionada por AMS o a la base de datos de RDS de destino. Para obtener más información, consulte AWS Database Migration Service (AWS DMS).

Método del método	Despliegue de infraestructura	AMI o elemento (s) clave	Instalación de la aplicación
CodeDeploy despliegue de aplicaciones	CodeDeploy	Aplicación -> CodeDeploy aplicación -> grupo CodeDeploy de despliegue -> CodeDeploy despliegue.	Según el uso, despliegue in situ o por Blue/Green aplicación. Para obtener información, consulte CodeDeploy solicitudes.

Planificación del despliegue de aplicaciones en AMS

Para obtener una serie de preguntas recomendadas que deben responderse para permitir la implementación de aplicaciones, consulteApéndice: Cuestionario de incorporación de solicitudes. Las preguntas abarcan la descripción de:

- Resumen de despliegue
- Componentes de despliegue de infraestructura
- Plataforma de alojamiento de aplicaciones
- Modelo de despliegue de aplicaciones
- Dependencias de aplicaciones
- Certificados SSL para aplicaciones de productos

Ingesta de carga de trabajo de AMS (WIGS)

Temas

- Migración de cargas de trabajo: requisitos previos para Linux y Windows
- How Migration Changes Your Resource (Cómo la migración cambia el recurso)
- Migrating Workloads: Standard Process (Migración de cargas de trabajo: proceso estándar)
- Migración de cargas de trabajo: CloudEndure landing zone (SALZ)
- Cuenta AMS Tools (migración de cargas de trabajo)
- Migración de cargas de trabajo: validación previa a la ingesta de Linux

- Migración de cargas de trabajo: validación previa a la ingestión de Windows
- Pila de ingesta de carga de trabajo: creación

Utilice el tipo de cambio de ingesta de carga de trabajo (CT) de AMS con un socio de migración a la nube de AMS para trasladar sus cargas de trabajo existentes a una VPC gestionada por AMS. Con la ingesta de carga de trabajo de AMS, puede crear una AMI de AMS personalizada después de mover las instancias migradas a AMS. En esta sección se describen el proceso, los requisitos previos y los pasos que usted y su socio de migración deben seguir para incorporar la carga de trabajo de AMS.

> **Important**
>
> El sistema operativo debe ser compatible con la ingesta de cargas de trabajo de AMS. Para obtener información sobre los sistemas operativos compatibles, consulteMigración de cargas de trabajo: requisitos previos para Linux y Windows.
> Cada carga de trabajo y cuenta es diferente. AMS trabajará con usted para prepararse para obtener un resultado exitoso.

El siguiente diagrama muestra el proceso de incorporación de la carga de trabajo de AMS.

Migración de cargas de trabajo: requisitos previos para Linux y Windows

Antes de incorporar una copia de una instancia local a AWS Managed Services (AMS), se deben cumplir ciertos requisitos previos. Estos son los requisitos previos, incluidos los que difieren entre los sistemas operativos Windows y Linux.

> **Note**
>
> Para simplificar el proceso de determinar si las instancias están listas para su ingestión, se han creado herramientas de validación tanto para Windows como para Linux. Estas herramientas se pueden descargar y ejecutar directamente en los servidores locales, así como en las EC2 instancias de AWS. Windows pre-WIGS Validation.zip para Linux y WIGS Validation.zip para Windows.

ANTES DE EMPEZAR, para Linux y Windows:

- Realice un análisis completo de virus.
- La instancia debe tener el perfil de la `customer-mc-ec2-instance-profile` instancia.
- Instale el agente Amazon EC2 Systems Manager (SSM) y asegúrese de que el agente SSM esté en funcionamiento.
- Se recomienda disponer de un mínimo de 10 GB de espacio libre en disco en el volumen raíz para ejecutar AMS Workload Ingest (WIGS). Desde el punto de vista operativo, AMS recomienda una utilización del disco inferior al 75% y envía alertas cuando la utilización del disco alcanza el 85%.
- Determine un plazo para la ingestión con su socio de migración.
- La AMI personalizada existe como una EC2 instancia en la cuenta AMS de producción de destino (es responsabilidad del socio de migración).

Important

El sistema operativo debe ser compatible con la ingesta de cargas de trabajo de AMS.

- Los sistemas operativos admitidos son los siguientes:
  - Microsoft Windows Server: 2008 R2, 2012, 2012 R2, 2016, 2019 y 2022
  - Linux: Amazon Linux 2023, Amazon Linux 2 y Amazon Linux, CentOS 7.x, CentOS 6.5-6.10, Oracle Linux 7: versiones secundarias 7.5 y superiores, Oracle Linux 8: versiones secundarias hasta 8.3, RHEL 8.x, RHEL 7.x, RHEL 6.5-6.10, SUSE Linux Enterprise Server 15 y versiones específicas de SAP, SUSE Linux Enterprise Server 12, Ubuntu 18.04 SP3 SP4 SP5
- No se admiten las siguientes funciones: AMIs
  - AMI mínima de Amazon Linux 2023.

Note

Los puntos de enlace AMS API/CLI (amscm y amsskms) se encuentran en la región de AWS del Norte de Virginia,. `us-east-1` En función de cómo esté configurada su autenticación y de la región de AWS en la que se encuentren su cuenta y sus recursos, es posible que tenga que `--region us-east-1` añadirlos al emitir comandos. Es posible que también necesite añadirlo`--profile saml`, si ese es su método de autenticación.

Requisitos previos de LINUX

Cumpla los requisitos enumerados en Migración de cargas de trabajo: requisitos previos para Linux y Windows y asegúrese de lo siguiente antes de enviar un RFC del WIGS:

- Están instalados los controladores de red mejorados más recientes; consulte Redes mejoradas en Linux.
- Se han eliminado los componentes de software de terceros que podían entrar en conflicto con los componentes de AMS:
  - Clientes antivirus
  - Clientes de Backup
  - Software de virtualización (como los servicios de integración de VM Tools o Hyper-V)
  - Software de administración de acceso (como SSSD, Centrify o PBIS)
- Asegúrese de que SSH esté configurado correctamente: esto habilita temporalmente la autenticación con clave privada para SSH. AMS lo usa con nuestra herramienta de administración de configuración. Utilice estos comandos:

```
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
```

- Asegúrese de que Yum esté correctamente configurado; RedHat requiere una licencia para usar sus repositorios de Yum. La licencia de la instancia debe realizarse a través de un servidor satélite o RedHat un servidor en la nube. Utilice uno de estos enlaces si necesita obtener una licencia:
  - Red Hat Satellite
  - Acceso a la nube de Red Hat
- Si utiliza Red Hat Satellite, WIGS requiere la adición de las colecciones de software de Red Hat (RHSCL). El sistema WIGS utiliza RHSCL para añadir un intérprete de Python 3.6 junto con todo lo que esté configurado en el sistema. Para admitir esta solución, deben estar disponibles los siguientes repositorios:
  - rhel-server-rhscl

- rhel-server-releases-optional

Requisitos previos de para Windows

Cumpla los requisitos enumerados en Migración de cargas de trabajo: requisitos previos para Linux y Windows y asegúrese de lo siguiente antes de enviar un RFC del WIGS:

- Está instalada la versión 3 o superior de Powershell.
- AWS EC2 Config se instala en la instancia con la carga de trabajo que va a migrar.
- Instale los controladores de AWS compatibles con los tipos de instancias de última generación: PV, ENA y NVMe. Puede utilizar la información de estos enlaces:
  - Actualización de los controladores fotovoltaicos en sus instancias de Windows
  - Redes mejoradas en Windows
  - NVMe Controladores de AWS para instancias de Windows
  - Parte 3: Actualización de NVMe los controladores de AWS
  - Parte 5: Instalación del controlador de puerto serie para instancias completas
  - Parte 6: Actualización de la configuración de administración de energía
- (Opcional pero recomendado) Desactivar los servicios críticos: desactive los servicios de aplicaciones críticos, como las bases de datos, pero asegúrese de documentar cualquier cambio para que pueda volver a su modo de inicio original durante la etapa de verificación de la aplicación.
- (Opcional pero recomendado) Cree una AMI a prueba de errores a partir de la instancia preparada:
  - Utilice la implementación | Componentes de pila avanzados | AMI | Crear
  - Durante la creación, añada una etiqueta Key=Name, value=Application-ID_ IngestReady
  - Espere a que se cree la AMI antes de continuar
- Se han eliminado los componentes de software de terceros que entrarían en conflicto con los componentes de AMS:
  - Clientes antivirus
  - Clientes de Backup
  - Software de virtualización (como los servicios de integración de VM Tools o Hyper-V)

> **Note**
>
> El programa de End-of-Support migración para Windows Server (EMP) incluye herramientas para migrar sus aplicaciones heredadas de Windows Server 2003, 2008 y 2008 R2 a versiones más recientes compatibles en AWS, sin necesidad de refactorización.

How Migration Changes Your Resource (Cómo la migración cambia el recurso)

El RFC de ingesta que se describe en esta sección es el siguiente paso: añadir configuraciones a la instancia, una vez que se haya migrado a tu cuenta de AMS, para que AMS pueda gestionarla.

Las configuraciones agregadas son específicas de AMS, tal como se indica a continuación.

Cambios realizados en las instancias de Linux ingeridas:

- Software que está instalado:
  - Cloud Init: se utiliza para configurar las claves privadas de Jarvis Access.
  - Python 3 (lenguaje de secuencias de comandos) para todos los sistemas operativos compatibles (excepto CentOS 6, RHEL 8 OracleLinux y 7).
  - Secuencias de comandos de AWS CloudFormation Python Helper: AWS CloudFormation proporciona secuencias de comandos que se utilizan para instalar software e iniciar servicios en EC2 instancias de Amazon.
  - AWS CLI: la CLI de AWS es una herramienta de código abierto creada sobre el AWS SDK para Python (Boto) que proporciona comandos para interactuar con los servicios de AWS.
  - Agente SSM de AWS: el agente de SSM procesa las solicitudes del servicio Systems Manager y configura la máquina según se especifica en la solicitud.
  - Agente CloudWatch de registros de AWS: envía los registros a CloudWatch.
  - AWS CodeDeploy: servicio de implementación que automatiza las implementaciones de aplicaciones en instancias de Amazon, EC2 instancias locales o funciones Lambda sin servidor.
  - Ruby: Necesario para CodeDeploy
  - Herramientas de rendimiento del sistema (sysstat): Sysstat contiene varias utilidades para monitorear el rendimiento del sistema y la actividad de uso.
  - AD Bridge (anteriormente PowerBroker Identity Services): une hosts que no son de Microsoft a dominios de Active Directory.

  - Trend Micro Deep Security Agent: software antivirus.
- Software que se ha modificado:
  - Las instancias están configuradas para usar la zona horaria UTC.

Cambios realizados en las instancias de Windows ingeridas:

- Software que está instalado:
  - Herramientas de AWS para Windows PowerShell: las herramientas de AWS PowerShell permiten a los desarrolladores y administradores gestionar sus servicios y recursos de AWS en el entorno de PowerShell secuencias de comandos.
  - Trend Micro Deep Security Agent: protección antivirus
  - PowerShell Los módulos AMS contienen PowerShell código para controlar el arranque, la unión a Active Directory, la supervisión, la seguridad y el registro.
- Software que se ha modificado:
  - La versión 1 del bloque de mensajes del servidor (SMB) está deshabilitada.
  - La administración remota de Windows (WinRM) está habilitada y configurada para escuchar en el puerto 5986. También se crea una regla de firewall que permite este puerto de entrada.
- Software que puede instalarse o modificarse:
  - Microsoft.Net Framework 4.5 (plataforma para desarrolladores), si se detecta una versión anterior a la 4.5 de Microsoft.Net Framework.
  - Para Windows 2012 y Windows 2012R2, actualizamos a PowerShell 5.1.

Migrating Workloads: Standard Process (Migración de cargas de trabajo: proceso estándar)

> **Note**
>
> Como se requieren dos partes para este proceso, en esta sección se describen las tareas de cada una de ellas: un socio de migración a la nube de AMS (socio de migración) y un propietario de la aplicación (usted).

1. Socio de migración, configuración:

   a. El socio de migración envía una solicitud de servicio a AMS para un puesto de IAM con el fin de migrar la instancia. Para obtener más información sobre cómo enviar solicitudes de servicio, consulta los ejemplos de solicitudes de servicio.

   b. El socio de migración envía una solicitud de acceso de administrador. El equipo de operaciones de AMS proporciona al socio de migración acceso a su cuenta a través del rol de IAM solicitado.

2. Socio de migración, Migrate Individual Workloads:

   a. El socio de migración migra la instancia que no es una AWS instancia a una subred de su cuenta de AMS mediante Amazon nativo EC2 u otras herramientas de migración, con el perfil de instancia de `customer-mc-ec2-instance-profile` IAM (debe estar en la cuenta).

   b. El socio de migración envía una RFC junto con la instancia migrada | Create CT (ct-257p9zjk14ija); para obtener más información sobre cómo crear y enviar esta RFC, consulte. Pila de ingesta de carga de trabajo: creación

      El resultado de ejecución de la RFC devuelve un ID de instancia, una dirección IP y un ID de AMI.

      El socio de migración le proporciona el ID de instancia de la carga de trabajo creada en su cuenta.

3. Tú, accede a la migración y valida la misma:

   a. Con el resultado de ejecución que le proporcionó el socio de migración (ID de AMI, ID de instancia y dirección IP), envíe una RFC de acceso e inicie sesión en la pila AMS recién creada y compruebe que la aplicación funciona correctamente. Para obtener más información, consulte Solicitar el acceso a la instancia.

   b. Si está satisfecho, puede seguir utilizando la instancia lanzada como una pila de 1 nivel y and/or utilizar la AMI para crear pilas adicionales, incluidos los grupos de Auto Scaling.

   c. Si no está satisfecho con la migración, presente una solicitud de servicio y consulte la pila y el RFC IDs; AMS trabajará con usted para resolver sus dudas.

CloudEndure El proceso de ingesta de carga de trabajo en la zona de landing zone se describe a continuación.

Migración de cargas de trabajo: CloudEndure landing zone (SALZ)

En esta sección, se proporciona información sobre cómo configurar una zona de aterrizaje de cuenta única (SALZ) de migración intermedia para que las instancias de CloudEndure transición (CE) estén disponibles para una RFC de ingesta de carga de trabajo (WIGS).

Para obtener más información, consulte Migración. CloudEndure CloudEndure

> **Note**
>
> Se trata de un patrón y una zona de migración predefinidos y reforzados por la seguridad.

Requisitos previos:

- Una cuenta AMS de un cliente
- Integración de red y acceso entre la cuenta AMS y el cliente en las instalaciones
- ¿Una cuenta CloudEndure
- Un flujo de trabajo de aprobación previa para una revisión y aprobación de AMS Security, que se ejecuta con su CA and/or CSDM (por ejemplo, el uso indebido de las credenciales permanentes de los usuarios de IAM permite que las instancias y los grupos de seguridad usen instancias y grupos de seguridad) create/delete

> **Note**
>
> En esta sección se describen los procesos específicos de preparación y migración.

Preparación: Usted y el operador de AMS:

1. Prepare una solicitud de cambio (RFC) con la gerencia | Otros | Otros | Actualice el tipo de cambio a AMS para obtener los siguientes recursos y actualizaciones. Puede enviar Otro | Otro

tipo de actualización RFCs por separado o solo uno. Para obtener más información sobre ese RFC/CT, consulte Otras | Otras actualizaciones con estas solicitudes:

a. Asigne un bloque CIDR secundario a su VPC de AMS; un bloque CIDR temporal que se eliminará una vez finalizada la migración. Asegúrese de que el bloqueo no entre en conflicto con ninguna ruta existente de regreso a su red local. Por ejemplo, si el CIDR de la VPC de AMS es 10.0.0.0/16 y hay una ruta de regreso a la red local de 10.1.0.0/16, el CIDR secundario temporal podría ser 10.255.255.0/24. Para obtener información sobre los bloques CIDR de AWS, consulte VPC y tamaño de subred.

b. Cree una nueva subred privada dentro de la VPC AMS de jardín inicial. Nombre de ejemplo:. `migration-temp-subnet`

c. Cree una nueva tabla de enrutamiento para la subred con solo rutas locales de VPC y NAT (Internet), a fin de evitar conflictos con el servidor de origen durante la transición de la instancia y posibles interrupciones. Asegúrese de que el tráfico saliente a Internet esté permitido para la descarga de parches y de que se puedan descargar e instalar los requisitos previos del AMS WIGS.

d. Actualice su grupo de seguridad de AD gestionado para permitir el tráfico entrante y saliente. to/from `migration-temp-subnet` Solicita también que se actualice tu grupo de seguridad del balanceador de cargas EPS (ELB) (por ejemplo,`mc-eps-McEpsElbPrivateSecurityGroup-M790XBZEEX74`) para permitir la nueva subred privada (es decir,). `migration-temp-subnet` Si el tráfico de la subred dedicada CloudEndure (CE) no está permitido en los tres puertos TCP, se producirá un error en la ingestión de WIGS.

e. Por último, solicite una nueva política de IAM y un nuevo CloudEndure usuario de IAM. <Customer Application Subnet (s) + Temp Migration Subnet>La política necesita su número de cuenta correcto y la subred IDs de la `RunInstances` declaración debe ser: la suya.

   Para ver una CloudEndure política de IAM aprobada previamente por AMS: descomprima el archivo de ejemplo de WIGS Cloud Endure Landing Zone y abra el. `customer_cloud_endure_policy.json`

   > **Note**
   >
   > Si desea una política más permisiva, hable con usted sobre lo que necesita CloudArchitect/CSDM y obtenga, si es necesario, una revisión de seguridad de AMS y su aprobación antes de enviar una RFC para implementar la política.

2. Los pasos de preparación que debe seguir CloudEndure para la incorporación de la carga de trabajo de AMS están listos y, si su socio de migración ha completado los pasos de preparación, la migración está lista para llevarse a cabo. El RFC del WIGS lo envía su socio de migración.

> **Note**
>
> Las claves de usuario de IAM no se compartirán directamente, sino que el operador del AMS debe escribirlas en la consola CloudEndure de administración en una sesión de pantalla compartida.

Preparación: Socio de migración y operador de AMS:

1. Cree un proyecto de CloudEndure migración.
   a. Durante la creación del proyecto, pida a AMS que escriba las credenciales de usuario de IAM en las sesiones de pantalla compartida.
   b. En Configuración de replicación -> Elija la subred en la que se lanzarán los servidores de replicación y seleccione subred. customer-application-x
   c. En Configuración de replicación -> Elija los grupos de seguridad que desee aplicar a los servidores de replicación, seleccione ambos grupos de seguridad Sentinel (solo privados y). EgressAll
2. Defina las opciones de transición para las máquinas (instancias).
   a. Subred:. migration-temp-subnet
   b. Grupo de seguridad: ambos grupos de seguridad «Sentinel» (solo privados y). EgressAll

      Las instancias de transición deben poder comunicarse con el AD administrado por AMS y con los puntos de enlace públicos de AWS.
   c. IP elástica: ninguna
   d. IP pública: no
   e. Función de IAM: customer-mc-ec perfil de 2 instancias

      La función de IAM debe permitir la comunicación por SSM. Es mejor usar AMS por defecto.
   f. Establezca las etiquetas según la convención.

Migración: Socio de migración:

1. Cree una pila ficticia en AMS. Usas el ID de la pila para acceder a los bastiones.
2. Instale el agente CloudEndure (CE) en el servidor de origen. Para obtener más información, consulte Instalación de los agentes.
3. Cree credenciales de administrador local en el servidor de origen.
4. Programe una breve ventana de transición y haga clic en Cambiar cuando esté listo. Esto finaliza la migración y redirige a los usuarios a la región de AWS de destino.
5. Solicite acceso de administrador de pila a la pila ficticia; consulte Solicitud de acceso de administrador.
6. Inicia sesión en el bastión y, después, en la instancia de transición con las credenciales de administrador local que creaste.
7. Cree una AMI a prueba de fallos. Para obtener más información sobre la creación AMIs, consulte AMI Create.
8. Prepare la instancia para su ingestión, consulteMigración de cargas de trabajo: requisitos previos para Linux y Windows.
9. Ejecute el RFC de WIGS en la instancia, consulte. Pila de ingesta de carga de trabajo: creación

Cuenta AMS Tools (migración de cargas de trabajo)

Su cuenta de herramientas de zona de destino multicuenta (con VPC) ayuda a acelerar los esfuerzos de migración, aumenta su posición de seguridad, reduce los costes y la complejidad y estandariza su patrón de uso.

Una cuenta de herramientas proporciona lo siguiente:

- Un límite bien definido para el acceso a las instancias de replicación para los integradores de sistemas ajenos a sus cargas de trabajo de producción.
- Le permite crear una cámara aislada para comprobar si una carga de trabajo contiene malware o rutas de red desconocidas antes de colocarla en una cuenta con otras cargas de trabajo.
- Al tratarse de una configuración de cuenta definida, agiliza la incorporación y la preparación para la migración de las cargas de trabajo.
- Rutas de red aisladas para proteger el tráfico desde las instalaciones -> -> Cuenta de herramientas CloudEndure -> imagen ingerida por AMS. Una vez ingerida una imagen, puede compartirla con la

cuenta de destino mediante un RFC de AMS Management | Advanced stack components | AMI | Share (ct-1eiczxw8ihc18).

Diagrama de arquitectura de alto nivel:

Utilice el tipo de cambio Deployment | Managed landing zone | Management Account | Create tools account (con VPC) (ct-2j7q1hgf26x5c) para implementar rápidamente una cuenta de herramientas e instanciar un proceso de ingestión de carga de trabajo en un entorno de zona de aterrizaje multicuenta. Consulte Cuenta de administración, Cuenta de herramientas: creación (con VPC).

> **Note**
>
> Recomendamos tener dos zonas de disponibilidad (AZs), ya que se trata de un centro de migración.
> De forma predeterminada, AMS crea los siguientes dos grupos de seguridad (SGs) en cada cuenta. Confirme que estos dos SGs estén presentes. Si no están presentes, abra una nueva solicitud de servicio con el equipo de AMS para solicitarlos.
>
> - SentinelDefaultSecurityGroupPrivateOnlyEgressAll
> - InitialGarden-SentinelDefaultSecurityGroupPrivateOnly
>
> Asegúrese de que las instancias de CloudEndure replicación se creen en la subred privada donde hay rutas de regreso a la red local. Puede confirmarlo asegurándose de que las tablas de enrutamiento de la subred privada tengan una ruta predeterminada de regreso a TGW. Sin embargo, si se realiza un corte de CloudEndure máquina, se debe utilizar una subred privada «aislada», donde no hay ninguna ruta de regreso a la red local y solo se permite el tráfico saliente de Internet. Es fundamental garantizar que la transición se produzca en la subred aislada para evitar posibles problemas con los recursos locales.

Requisitos previos:

1. Nivel de soporte Plus o Premium.
2. La cuenta de aplicación IDs de la clave KMS en la que AMIs se despliegan.
3. La cuenta de herramientas, creada como se describió anteriormente.

AWS Servicio de migración de aplicaciones (AWS MGN)

AWS El Servicio de migración de aplicaciones (AWS MGN) se puede utilizar en su cuenta de MALZ Tools mediante la función de `AWSManagedServicesMigrationRole` IAM que se crea automáticamente durante el aprovisionamiento de la cuenta de Tools. Puede usar AWS MGN para migrar aplicaciones y bases de datos que se ejecutan en versiones compatibles de los sistemas operativos Windows y Linux.

Para up-to-date obtener más información sobre el Región de AWS soporte, consulte la Lista de servicios AWS regionales.

Si MGN no admite actualmente su preferido Región de AWS o si AWS MGN no admite actualmente el sistema operativo en el que se ejecutan sus aplicaciones, considere la posibilidad de utilizar la CloudEndure migración en su cuenta de herramientas. AWS

Solicitar AWS la inicialización de MGN

AMS debe inicializar AWS MGN antes del primer uso. Para solicitarlo para una nueva cuenta de Herramientas, envíe un RFC de administración | Otros | Otros desde la cuenta de Herramientas con estos detalles:

```
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.
```

Una vez que AMS complete correctamente la RFC e inicialice AWS MGN en su cuenta de Tools, podrá utilizarla `AWSManagedServicesMigrationRole` para editar la plantilla predeterminada según sus necesidades.

Habilitar el acceso a la nueva cuenta de AMS Tools

Una vez creada la cuenta de herramientas, AMS le proporciona un identificador de cuenta. El siguiente paso es configurar el acceso a la nueva cuenta. Siga estos pasos.

1. Actualice los grupos de Active Directory correspondientes a la cuenta correspondiente IDs.

Las nuevas cuentas creadas por AMS se aprovisionan con la política de funciones, así como con una ReadOnly función que permite a los usuarios archivar archivos. RFCs

La cuenta Tools también tiene un rol de IAM y un usuario adicionales disponibles:

- Rol de IAM: `AWSManagedServicesMigrationRole`
- Usuario de IAM: `customer_cloud_endure_user`

2. Solicite políticas y funciones que permitan a los miembros del equipo de integración de servicios configurar el siguiente nivel de herramientas.

   Navegue hasta la consola AMS y archive lo siguiente RFCs:

   a. Cree una clave KMS. Utilice Crear clave KMS (auto) o Crear clave KMS (se requiere revisión).

      Al usar KMS para cifrar los recursos ingeridos, el uso de una única clave KMS que se comparte con el resto de las cuentas de la aplicación MultiAccount Landing Zone proporciona seguridad a las imágenes ingeridas, ya que se pueden descifrar en la cuenta de destino.

   b. Comparta la clave KMS.

      Utilice el tipo de cambio Management | Advanced stack components | KMS key | Share (es necesario revisar) (ct-05yb337abq3x5) para solicitar que la nueva clave KMS se comparta con las cuentas de la aplicación en las que residirá la ingestión. AMIs

Ejemplo gráfico de la configuración final de una cuenta:

Ejemplo de política de CloudEndure IAM preaprobada por AMS

Para ver una CloudEndure política de IAM preaprobada por AMS: desempaquete el archivo de ejemplo de zona de aterrizaje de WIGS Cloud Endure y abra el. `customer_cloud_endure_policy.json`

Prueba de la conectividad y la end-to-end configuración de la cuenta de AMS Tools

1. Comience por configurar CloudEndure e instalar el CloudEndure agente en un servidor que se replicará en AMS.
2. Cree un proyecto en CloudEndure.
3. Introduzca las AWS credenciales compartidas cuando realizó los requisitos previos, a través del administrador de secretos.

4. En la configuración de replicación:
   a. Seleccione los dos grupos de seguridad «Sentinel» de AMS (solo privados y EgressAll) en la opción Elija los grupos de seguridad que desee aplicar a los servidores de replicación.
   b. Defina las opciones de transición para las máquinas (instancias). Para obtener información, consulte el paso 5. Corte
   c. Subred: subred privada.
5. Grupo de seguridad:
   a. Seleccione los dos grupos de seguridad AMS «Sentinel» (solo privados y EgressAll).
   b. Las instancias de transición deben comunicarse con el Active Directory (MAD) administrado por AMS y con los puntos finales públicos: AWS
      i. IP elástica: ninguna
      ii. IP pública: no
      iii. Función de IAM: customer-mc-ec perfil de 2 instancias
   c. Configure las etiquetas según su convención de etiquetado interna.
6. Instale el CloudEndure agente en la máquina y busque la instancia de replicación que aparecerá en su cuenta de AMS en la EC2 consola.

El proceso de ingestión de AMS:

Higiene de cuentas de AMS Tools

Querrá limpiar una vez que haya terminado en la cuenta, haya compartido la AMI y ya no necesite las instancias replicadas:

- Tras la WIGs ingestión de instancias:
  - Instancia de transición: como mínimo, detenga o finalice esta instancia, una vez que se haya completado el trabajo, a través de la consola de AWS
  - Respaldos de AMI previos a la ingesta: elimínelos una vez que la instancia se haya ingerido y la instancia local haya terminado
  - Instancias ingeridas por AMS: apague la pila o finalice una vez que se haya compartido la AMI
  - Ingeridas por AMS AMIs: se eliminan una vez que se haya completado el uso compartido con la cuenta de destino

- Limpieza final de la migración: documenta los recursos desplegados a través del modo Desarrollador para garantizar que la limpieza se realice de forma periódica, por ejemplo:
  - Grupos de seguridad
  - Recursos creados mediante la formación de nubes
  - ACK de red
  - Subred
  - VPC
  - Tabla de enrutamiento
  - Roles
  - Usuarios y cuentas

Migración a gran escala - Migration Factory

Consulte Presentación de la solución AWS CloudEndure Migration Factory.

Migración de cargas de trabajo: validación previa a la ingesta de Linux

Puede validar que la instancia esté lista para su incorporación a su cuenta de AMS. La validación previa a la ingesta de cargas de trabajo (WIGS) realiza comprobaciones como el tipo de sistema operativo, el espacio disponible en disco, la existencia de software de terceros conflictivo, etc. Cuando se ejecuta, la validación previa a la ingesta del WIGS genera una tabla en pantalla con un archivo de registro opcional. Los resultados proporcionan el pass/fail estado de cada comprobación de validación junto con el motivo de los errores. Además, puede personalizar las pruebas de validación para adaptarlas a sus necesidades.

Preguntas frecuentes:

- ¿Cómo utilizo la validación previa a la ingestión de WIGS de Linux?

  Siga estos pasos para descargar y utilizar los scripts de validación previa a la ingestión de AMS Linux WIGS:

  1. Descargue un archivo ZIP con los scripts de validación

     Archivo zip de validación previa a la ingestión del WIGS de Linux.
  2. Descomprima las reglas adjuntas en el directorio que prefiera.
  3. Siga las instrucciones del archivo readme.md.

- ¿Qué validaciones realiza la validación previa a la ingestión del WIGS de Linux?

  La solución de validación previa a la ingestión WIGS de AMS Linux valida lo siguiente:

  1. Hay al menos 5 gigabytes libres en el volumen de arranque.
  2. El sistema operativo es compatible con AMS.
  3. La instancia tiene un perfil de instancia específico.
  4. La instancia no contiene software antivirus ni software de virtualización.
  5. SSH está configurado correctamente.
  6. La instancia tiene acceso a los repositorios de Yum.
  7. Están instalados controladores de red mejorados.
  8. La instancia tiene el agente SSM y se está ejecutando.
- ¿Por qué se admite un archivo de configuración personalizado?

  Los scripts están diseñados para ejecutarse tanto en servidores físicos locales como en EC2 instancias de AWS. Sin embargo, como se muestra en la lista anterior, algunas pruebas fallarán cuando se ejecuten en las instalaciones. Por ejemplo, un servidor físico de un centro de datos no tendría un perfil de instancia. En estos casos, puede editar el archivo de configuración para omitir la prueba del perfil de la instancia y evitar confusiones.
- ¿Cómo me aseguro de tener la última versión del script?

  Encontrará una up-to-date versión de la solución de validación previa a la ingestión WIGS de Linux en la sección Archivos auxiliares de AMS de la página principal de documentación.
- ¿El script es de solo lectura?

  El script está diseñado para ser de solo lectura, excepto los archivos de registro que produce, pero se deben seguir las mejores prácticas para ejecutar el script en un entorno que no sea de producción.
- ¿La validación previa a la ingesta de WIGS está disponible para Windows?

  Sí. Está disponible en la sección Archivos auxiliares de AMS, en la página principal de documentación.

Migración de cargas de trabajo: validación previa a la ingestión de Windows

Puede usar el script de WIGs prevalidación para validar que la instancia esté lista para ser incorporada a su cuenta de AMS. La validación previa a la ingesta de carga de trabajo (WIGS)

realiza comprobaciones como el tipo de sistema operativo, el espacio disponible en disco, la existencia de software de terceros en conflicto, etc. Cuando se ejecuta, la validación previa a la ingesta del WIGS genera una tabla en pantalla y un archivo de registro opcional. Los resultados proporcionan el pass/fail estado de cada comprobación de validación junto con el motivo del error. Además, puede personalizar las pruebas de validación.

Preguntas frecuentes:

- ¿Cómo utilizo la validación previa a la ingestión de Windows WIGS?

  Puede ejecutar la validación desde una interfaz gráfica de usuario y un navegador web, o puede utilizar Windows PowerShell, SSM Run Command o SSM Session Manager.

  Opción 1: ejecutar desde una interfaz gráfica de usuario y un navegador web

  Para ejecutar la WIGs validación previa de Windows desde una GUI y un navegador web, haga lo siguiente:

  1. Descargue un archivo ZIP con los scripts de validación:

     Archivo ZIP de validación previa a la ingestión del WIGS de Windows.
  2. Descomprima las reglas adjuntas en el directorio que prefiera.
  3. Siga las instrucciones del archivo README.md.

  Opción 2: ejecutar desde Windows PowerShell, SSM Run Command o SSM Session Manager

  Windows 2016 y versiones posteriores

  1. Descargue el archivo ZIP con los scripts de validación.

     ```
     $DestinationFile = "$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. Elimine los archivos existentes de`C:\Users\AppData\Local\Temp\AWSManagedServices.PreWigs.Validation`.

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

3. Invoca el script.

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

4. Descomprima los archivos adjuntos en un directorio de su elección.

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

5. Ejecute el script de validación de forma interactiva y vea los resultados.

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

6. (Opcional) Para capturar los códigos de error que aparecen en la sección de códigos de salida, ejecute el script sin `RunWithoutExitCodes` esta opción. Tenga en cuenta que este comando finaliza la PowerShell sesión activa.

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

Windows 2012 R2 y versiones anteriores

Si utilizas Windows Server 2012R2 o una versión anterior, debes configurar TLS antes de descargar el archivo zip. Para configurar TLS, complete los siguientes pasos:

1. Descargue el archivo ZIP con los scripts de validación.

```
$DestinationFile = "$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. Si hay archivos de validación existentes, elimínelos.

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

3. Configure la versión de TLS.

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

4. Descarga la validación de WIG.

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

5. Descomprima las reglas adjuntas en el directorio que prefiera.

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

6. Ejecute el script de validación de forma interactiva y vea los resultados.

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

7. (Opcional) Para capturar los códigos de error que aparecen en la sección de códigos de salida, ejecute el script sin RunWithoutExitCodes esta opción. Tenga en cuenta que este comando finaliza la PowerShell sesión activa.

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

> **Note**
>
> Puede descargar y ejecutar los PowerShell scripts. Para ello, descargue el pre-wigs-validation-powershellarchivo -scripts.zip.

- ¿Qué validaciones realiza la validación previa a la ingestión del WIGS de Windows?

  La solución de validación previa a la ingestión de Windows WIGS de AMS valida lo siguiente:

  1. Hay al menos 10 gigabytes libres en el volumen de arranque.
  2. El sistema operativo es compatible con AMS.
  3. La instancia tiene un perfil de instancia específico.
  4. La instancia no contiene software antivirus ni software de virtualización.
  5. El DHCP está habilitado en al menos un adaptador de red.
  6. La instancia está lista para Sysprep.

- Para 2008 R2 y 2012 Base y R2, Sysprep comprueba que:
  - Hay un archivo unattend.xml
  - El archivo sppnp.dll (si está presente) no está dañado
  - El sistema operativo no se ha actualizado
  - Sysprep no se ha ejecutado más de la cantidad máxima de veces según las directrices de Microsoft
- A partir de 2016, se omiten todas las comprobaciones anteriores, ya que ninguna de las dos causa problemas a ese sistema operativo

7. El subsistema del instrumento de administración de Windows (WMI) funciona correctamente.
8. Están instalados los controladores necesarios.
9. El agente SSM ya está instalado y en ejecución.
10. Se emite una advertencia para comprobar si la máquina se encuentra en período de gracia debido a la configuración de la licencia de RDS.
11. Las claves de registro requeridas están configuradas correctamente. Para obtener más información, consulte el archivo README del archivo zip de validación previa a la ingestión.

- ¿Por qué se admite un archivo de configuración personalizado?

  Los scripts están diseñados para ejecutarse tanto en servidores físicos locales como en EC2 instancias de AWS. Sin embargo, como se muestra en la lista anterior, algunas pruebas fallarán cuando se ejecuten en las instalaciones. Por ejemplo, un servidor físico de un centro de datos no tendría un perfil de instancia. En estos casos, puede editar el archivo de configuración para omitir la prueba del perfil de la instancia y evitar confusiones.

- ¿Cómo me aseguro de tener la última versión del script?

  Encontrará una up-to-date versión de la solución de validación previa a la ingesta WIGS de Windows en la sección Archivos auxiliares de AMS de la página principal de documentación.

- ¿El script es de solo lectura?

  El script está diseñado para ser de solo lectura, excepto los archivos de registro que produce, pero se deben seguir las mejores prácticas para ejecutar el script en un entorno que no sea de producción.

- ¿La validación previa a la ingestión de WIGS está disponible para Linux?

  Sí. La versión para Linux se lanzó el 31 de octubre de 2019. Está disponible en la sección Archivos auxiliares de AMS, en la página principal de documentación.

Pila de ingesta de carga de trabajo: creación

Migración de una instancia a una pila de AMS con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada Buscar tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

> **Note**
>
> Si se rechaza la RFC, el resultado de la ejecución incluye un enlace a Amazon CloudWatch Logs. Los AMS Workload Ingest (WIGS) RFCs se rechazan cuando no se cumplen los requisitos; por ejemplo, si se detecta un software antivirus en la instancia. Los CloudWatch registros incluirán información sobre el requisito no cumplido y las medidas que se deben tomar para solucionarlo.

Migración de una instancia a una pila AMS con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

Puede usar la CLI de AMS para crear una instancia de AMS a partir de una instancia que no sea de AMS migrada a una cuenta de AMS.

> **Note**
>
> Asegúrese de cumplir los requisitos previos; consulte Migración de cargas de trabajo: requisitos previos para Linux y Windows.

Para comprobar la versión del tipo de cambio, utilice este comando:

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

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución del esquema JSON para este tipo de cambio a un archivo; el ejemplo lo MigrateStackParams nombra .json:

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

2. Modifique y guarde el archivo JSON de los parámetros de ejecución. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "InstanceId":          "MIGRATED_INSTANCE_ID",
   "TargetVpcId":         "VPC_ID",
   ```

```
"TargetSubnetId":      "SUBNET_ID",
"Name":                "Migrated-Stack",
"Description":         "Create-Migrated-Stack",
"EnforceIMDSV2":       "false"
}
```

3. Genera el archivo JSON de la plantilla RFC; el ejemplo lo nombra MigrateStackRfc .json:

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

4. Modifique y guarde el MigrateStackRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

5. Cree el RFC, especificando el MigrateStackRfc archivo y el MigrateStackParams archivo:

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

   Recibirá el ID de la nueva RFC en la respuesta y podrá utilizarla para enviar y supervisar la RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

   La nueva instancia aparece en la lista de instancias de la cuenta del propietario de la aplicación para la VPC correspondiente.

6. Cuando la RFC se complete correctamente, notifique al propietario de la aplicación para que pueda iniciar sesión en la nueva instancia y comprobar que la carga de trabajo está operativa.

> **Note**
>
> Si se rechaza la RFC, el resultado de la ejecución incluye un enlace a Amazon CloudWatch Logs. Los AMS Workload Ingest (WIGS) RFCs se rechazan cuando no se cumplen los requisitos; por ejemplo, si se detecta un software antivirus en la instancia. Los CloudWatch registros incluirán información sobre el requisito no cumplido y las medidas que se deben tomar para solucionarlo.

Consejos

> **Note**
>
> Asegúrese de cumplir los requisitos previos; consulte Migración de cargas de trabajo: requisitos previos para Linux y Windows.

> **Note**
>
> Si una etiqueta de la instancia que se está migrando tiene la misma clave que la etiqueta proporcionada en la RFC, la RFC fallará.

> **Note**
>
> Puede especificar hasta cuatro zonas de destino IDs, puertos y disponibilidad.

> **Note**
>
> Si se rechaza la RFC, el resultado de la ejecución incluye un enlace a Amazon CloudWatch Logs. Los AMS Workload Ingest (WIGS) RFCs se rechazan cuando no se cumplen los requisitos; por ejemplo, si se detecta un software antivirus en la instancia. Los CloudWatch registros incluirán información sobre el requisito no cumplido y las medidas que se deben tomar para solucionarlo.

> **Note**
>
> Si se rechaza la RFC, el resultado de la ejecución incluye un enlace a Amazon CloudWatch Logs. Los AMS Workload Ingest (WIGS) RFCs se rechazan cuando no se cumplen los requisitos; por ejemplo, si se detecta un software antivirus en la instancia. Los CloudWatch registros incluirán información sobre el requisito no cumplido y las medidas que se deben tomar para solucionarlo.

Si es necesario, consulte Fallo en la ingestión de carga de trabajo (WIGS).

CloudFormation Ingesta de AMS

El tipo de cambio de CloudFormation ingesta (CT) de AMS AWS le permite utilizar las CloudFormation plantillas existentes, con algunas modificaciones, para implementar pilas personalizadas en una VPC administrada por AMS.

Temas

- CloudFormation Pautas, mejores prácticas y limitaciones de ingesta
- CloudFormation Ingerir: ejemplos
- Cree una pila de ingesta CloudFormation
- Actualice la pila de ingesta CloudFormation
- CloudFormation Aprueba un conjunto de cambios en la pila de ingesta
- Protección de terminación CloudFormation de Update Stacks
- Implementaciones de IAM automatizadas mediante la ingesta de CFN o la actualización de pilas en AMS CTs

El proceso de ingesta de AMS implica CloudFormation lo siguiente:

- Prepare y cargue su CloudFormation plantilla personalizada en un depósito de S3 o proporcione la plantilla en línea al crear la RFC. Si utiliza un bucket de S3 con una URL prefirmada, consulte presign para obtener más información.
- Envíe el tipo de cambio CloudFormation de ingesta a AMS en una RFC. Para ver el tutorial del tipo de cambio de ingesta de CFN, consulte. Cree una pila de ingesta CloudFormation Para ver ejemplos de ingesta de CFN, consulte. CloudFormation Ingerir: ejemplos
- Una vez creada la pila, puedes actualizarla y corregir sus errores. Además, si la actualización no se realiza correctamente, puedes aprobarla e implementarla de forma explícita. Todos estos procedimientos se describen en esta sección.

  Para obtener información sobre la detección de la deriva del CFN, consulte Nuevo: detección de CloudFormation deriva.

> **Note**

- Este tipo de cambio ahora tiene una versión 2.0. La versión 2.0 es automática; no se ejecuta manualmente. Esto permite que la ejecución de la tomografía computarizada sea más rápida. En esta versión se introducen dos parámetros nuevos: CloudFormationTemplate, que permite pegar una CloudFormation plantilla personalizada en el RFC, y VpcId, que permite utilizar CloudFormation ingest con la zona de aterrizaje multicuenta de AMS.
- La versión 1.0 es un tipo de cambio manual. Esto significa que un operador de AMS debe realizar alguna acción antes de que el tipo de cambio pueda concluir satisfactoriamente. Como mínimo, se requiere una revisión. Esta versión también requiere que el valor del parámetro CloudFormationTemplateS3Endpoint sea una URL prefirmada.

CloudFormation Pautas, mejores prácticas y limitaciones de ingesta

Para que AMS procese tu CloudFormation plantilla, existen algunas pautas y restricciones.

Directrices

Para reducir CloudFormation los errores al realizar CloudFormation la ingesta, sigue estas pautas:

- No insertes credenciales ni otra información confidencial en la plantilla: la CloudFormation plantilla está visible en la CloudFormation consola, por lo que no querrás incrustar credenciales ni datos confidenciales en la plantilla. La plantilla no puede contener información confidencial. Los siguientes recursos solo están permitidos si utiliza AWS Secrets Manager como valor:
  - `AWS::RDS::DBInstance` - [MasterUserPassword,TdeCredentialPassword]
  - `AWS::RDS::DBCluster` - [MasterUserPassword]
  - `AWS::ElastiCache::ReplicationGroup` - [AuthToken]

  > **Note**
  >
  > Para obtener información sobre el uso de un secreto de AWS Secrets Manager en una propiedad de recurso, consulte Cómo crear y recuperar secretos gestionados en AWS Secrets Manager mediante CloudFormation plantillas de AWS y Uso de referencias dinámicas para especificar valores de plantillas.

- Utilice instantáneas de Amazon RDS para crear instancias de base de datos de RDS: de este modo, evitará tener que proporcionar un. MasterUserPassword

- Si la plantilla que envía contiene un perfil de instancia de IAM, debe llevar el prefijo «cliente». Por ejemplo, si se utiliza un perfil de instancia con el nombre "example-instance-profile, se produce un error. En su lugar, usa un perfil de instancia con el nombre 'customer-example-instance-profile'.
- No incluyas ningún dato confidencial en **`AWS::EC2::Instance`** - [UserData]. UserData no debe contener contraseñas, claves de API ni ningún otro dato confidencial. Este tipo de datos se pueden cifrar y almacenar en un bucket de S3 y descargar a la instancia mediante UserData.
- La creación de políticas de IAM mediante CloudFormation plantillas se admite con restricciones: las políticas de IAM deben ser revisadas y aprobadas por AMS. SecOps Actualmente, solo admitimos el despliegue de funciones de IAM con políticas integradas que contengan permisos previamente aprobados. En otros casos, las políticas de IAM no se pueden crear mediante CloudFormation plantillas porque eso anularía el proceso de AMS. SecOps
- KeyPairs No se admiten SSH: se debe acceder a EC2 las instancias de Amazon a través del sistema de gestión de acceso AMS. El proceso RFC de AMS lo autentica. No puede incluir pares de claves SSH en CloudFormation las plantillas porque no tiene los permisos para crear pares de claves SSH y anular el modelo de administración de acceso de AMS.
- Las reglas de entrada de los grupos de seguridad están restringidas: no puede tener un CIDR de origen comprendido entre 0.0.0.0/0 ni un espacio de direcciones enrutable públicamente con un puerto TCP que no sea 80 o 443.
- Siga CloudFormation las pautas al escribir plantillas de CloudFormation recursos: asegúrese de usar el type/property nombre de datos correcto para el recurso consultando la Guía del usuario de ese recurso.AWS CloudFormation Por ejemplo, el tipo de datos de la SecurityGroupIds propiedad de un AWS::EC2::Instance recurso es «Lista de valores de cadena», por lo que ["sg-aaaaaaaa"] está bien (entre corchetes), pero «sg-aaaaaaaa» no lo es (sin corchetes).

  Para obtener más información, consulte la referencia de tipos de recursos y propiedades de AWS.
- Configure sus CloudFormation plantillas personalizadas para que usen los parámetros definidos en el CT de CloudFormation ingesta del AMS: si configura su CloudFormation plantilla para que utilice los parámetros definidos en el CT de CloudFormation ingesta del AMS, puede reutilizar la CloudFormation plantilla para crear pilas similares enviándola con los valores de los parámetros modificados en la entrada del CT con el CT Management | Custom stack | Stack from CloudFormation template | Update CT (ct-361tlo1k7339x). Para ver un ejemplo, consulta CloudFormation Ejemplos de ingesta: definición de recursos.
- Los puntos de enlace de bucket de Amazon S3 con una URL prefirmada no pueden caducar: si utiliza un punto de enlace de bucket de Amazon S3 con una URL prefirmada, compruebe que la

URL prefirmada de Amazon S3 no esté caducada. Se rechaza un CloudFormation RFC de ingesta enviado con una URL de bucket de Amazon S3 prefirmada y caducada.

- La condición de espera requiere una lógica de señal: la condición de espera se utiliza para coordinar la creación de recursos de la pila con las acciones de configuración que son externas a la creación de la pila. Si utilizas el recurso Wait Condition de la plantilla, CloudFormation espera una señal de éxito y marca la creación de la pila como un error si no se detecta el número de señales de éxito. Debe tener una lógica para la señal si utiliza el recurso Wait Condition. Para obtener más información, consulte Creación de condiciones de espera en una plantilla.

Prácticas recomendadas

Las siguientes son algunas de las prácticas recomendadas que puede utilizar para migrar recursos mediante el proceso de CloudFormation ingesta de AMS:

- Presente la IAM y otros recursos relacionados con las políticas en un solo lugar: si puede utilizar la automatización, CTs como CloudFormation Ingest, para implementar las funciones de IAM, le recomendamos que lo haga. En otros casos, AMS recomienda reunir todos los recursos de IAM u otros recursos relacionados con las políticas y enviarlos en un único tipo de gestión | Otros | Otros | Crear cambio (ct-1e1xtak34nx76). Por ejemplo, combine todas las funciones de IAM necesarias, los perfiles de EC2 instancias de Amazon de IAM, las actualizaciones de las políticas de IAM para las funciones de IAM existentes, las políticas de bucket de Amazon S3, las políticas de Amazon SNS/Amazon SQS, etc., y envíe una RFC ct-1e1xtak34nx76 para que se pueda hacer referencia a estos recursos preexistentes de forma sencilla en las plantillas de ingesta futuras. CloudFormation
- EC2 Las instancias se inician y se unen correctamente al dominio. Esto se hace automáticamente como práctica recomendada. Para garantizar que las EC2 instancias de Amazon lanzadas mediante una pila de CloudFormation ingesta se inicien y se unan al dominio correctamente, AMS incluye un recurso grupal CreationPolicy y un UpdatePolicy para un recurso grupal de Auto Scaling (es decir, si estas políticas aún no existen).
- Debe especificarse el parámetro de la instancia de base de datos de Amazon RDS: al crear una base de datos de Amazon RDS mediante CloudFormation ingesta, debe especificar el `DBSnapshotIdentifier` parámetro para poder restaurarla desde una instantánea de base de datos anterior. Esto es obligatorio porque, actualmente, CloudFormation ingest no gestiona datos confidenciales.

Para ver un ejemplo de cómo utilizar una CloudFormation plantilla para la ingesta de CloudFormation plantillas de AMS, consulte. CloudFormation Ingerir: ejemplos

Validación de plantillas

Puedes autovalidar tu CloudFormation plantilla antes de enviarla a AMS.

Las plantillas enviadas a AMS CloudFormation Ingest se validan para garantizar que se puedan implementar de forma segura en una cuenta de AMS. El proceso de validación comprueba lo siguiente:

- Recursos compatibles: solo se utilizan los recursos CloudFormation compatibles con AMS Ingest. Para obtener más información, consulte Recursos admitidos.
- Compatible AMIs : la AMI de la plantilla es una AMI compatible con AMS. Para obtener información acerca de AMS AMIs, consulte. Imágenes de máquinas AMS Amazon (AMIs)
- Subred de AMS Shared Services: la plantilla no intenta lanzar recursos en la subred de AMS Shared Services.
- Políticas de recursos: no hay políticas de recursos demasiado permisivas, como una política de bucket de S3 que se pueda leer o escribir públicamente. AMS no permite la entrada de buckets de S3 en los que se pueda escribir o leer públicamente. Cuentas de AWS

Valide con Linter CloudFormation

Puede autovalidar su CloudFormation plantilla antes de enviarla a AMS mediante la CloudFormation herramienta Linter.

La herramienta CloudFormation Linter es la mejor forma de validar la CloudFormation plantilla, ya que permite validar resource/property los nombres, los tipos de datos y las funciones. Para obtener más información, consulte cfn-python-lintaws-cloudformation/.

El resultado de CloudFormation Linter de la plantilla que se muestra anteriormente es el siguiente:

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

Para facilitar la validación de las CloudFormation plantillas sin conexión, AMS ha desarrollado un conjunto de reglas de validación personalizadas y conectables para la CloudFormation herramienta Linter. Se encuentran en la página de recursos para desarrolladores de la consola AMS.

Siga estos pasos para utilizar los scripts de validación CloudFormation previos a la ingesta:

1. Instale la herramienta CloudFormation Linter. Para obtener instrucciones de instalación, consulte aws-cloudformation/cfn-lint.
2. Descargue un archivo.zip con los scripts de validación:

   Reglas personalizadas de CFN Lint.
3. Descomprima las reglas adjuntas en un directorio de su elección.
4. Valide la CloudFormation plantilla ejecutando el siguiente comando:

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

CloudFormation pila de ingesta: ejemplos de validadores CFN

Estos ejemplos pueden ayudarte a preparar tu plantilla para una ingesta exitosa.

Validación de formato

Compruebe que la plantilla contenga una sección de «Recursos» y que todos los recursos definidos en ella tengan un valor de «Tipo».

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

Valide que las claves raíz de la plantilla estén permitidas. Las claves raíz permitidas son:

```
[
  "AWSTemplateFormatVersion",
  "Description",
  "Mappings",
  "Parameters",
  "Conditions",
  "Resources",
  "Rules",
```

```
  "Outputs",
  "Metadata"
]
```

La revisión manual requiere validación

Si la plantilla contiene los siguientes recursos, la validación automática fallará y necesitarás una revisión manual.

Las políticas que se muestran son áreas de alto riesgo desde el punto de vista de la seguridad. Por ejemplo, una política de compartimentos de S3 que permita a cualquier persona, excepto a usuarios o grupos específicos, crear objetos o escribir permisos es extremadamente peligrosa. Por lo tanto, validamos las políticas y las aprobamos o rechazamos en función del contenido, y esas políticas no se pueden crear automáticamente. Estamos investigando posibles enfoques para abordar este problema.

Actualmente, no contamos con una validación automática de los siguientes recursos.

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

Validación de parámetros

Valide que si un parámetro de plantilla no tiene un valor proporcionado, debe tener un valor predeterminado.

Validación de atributos de recursos

Verificación de atributos obligatoria: deben existir ciertos atributos para ciertos tipos de recursos.

- "VPCOptions" debe existir en `AWS::OpenSearch::Domain`
- "CludsterSubnetGroupName" debe existir en `AWS::Redshift::Cluster`

```
{
    "AWS::OpenSearch::Domain": [
      "VPCOptions"
```

```
    ],
    "AWS::Redshift::Cluster": [
      "ClusterSubnetGroupName"
    ]
}
```

Verificación de atributos no permitidos: ciertos atributos *no* deben existir para ciertos tipos de recursos.

- "SecretString" no debe existir en "» AWS::SecretsManager::Secret
- "MongoDbSettings" no debe existir en "AWS::DMS::Endpoint»

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

Comprobación de parámetros SSM: para los atributos de la siguiente lista, los valores deben especificarse mediante Secrets Manager o 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"
  ]
},
```

Algunos atributos deben cumplir ciertos patrones; por ejemplo, los nombres de los perfiles de las instancias de IAM no deben empezar con prefijos reservados de AMS y el valor del atributo debe coincidir con la expresión regular específica, como se muestra:

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

```
}
```

Validación de recursos

En la plantilla solo se pueden especificar los recursos permitidos; esos recursos se describen en. Recursos admitidos

EC2 Las pilas y los grupos de Auto Scaling (ASGs) no están permitidos en la misma pila debido a las limitaciones de aplicación de parches.

Validación de las reglas de ingreso de grupos de seguridad

- Para las solicitudes que provienen de los tipos de cambio CFN Ingest Create o Stack Update CT:
  - Si (`IpProtocoles` tcp o 6) AND (el puerto es 80 o 443), no hay restricciones en cuanto al valor `CidrIP`
  - De lo contrario, `CidrIP` no puede ser 0.0.0.0/0
- Para las solicitudes que provienen de Service Catalog (productos de Service Catalog):
  - Además de la validación del tipo de cambio CFN Ingest Create o Stack Update CT, solo se `ip_protocols` puede acceder al puerto de entrada `management_ports` con el protocolo introducido a través de: `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"]
  }
```

Limitaciones

Actualmente, el proceso de CloudFormation ingesta de AMS no admite las siguientes características y funciones.

- YAML: no se admite. Solo se admiten las CloudFormation plantillas basadas en JSON.
- Pilas anidadas: en su lugar, diseñe la infraestructura de aplicaciones para usar una sola plantilla. O bien, puede utilizar la referencia cruzada entre pilas para separar los recursos en varias pilas

en las que un recurso depende de otro. Para obtener más información, consulte Tutorial: consulte Resource Output in Another AWS CloudFormation Stack.

- CloudFormation conjuntos de pilas: no se admiten por motivos de seguridad.
- Creación de recursos de IAM mediante CloudFormation plantillas: por motivos de seguridad, solo se admiten las funciones de IAM.
- Datos confidenciales: no se admiten. No incluya datos confidenciales en la plantilla ni en los valores de los parámetros. Si necesita hacer referencia a datos confidenciales, utilice Secrets Manager para almacenar y recuperar estos valores. Para obtener información sobre el uso de los secretos de AWS Secrets Manager en una propiedad de recurso, consulte Cómo crear y recuperar los secretos gestionados en AWS Secrets Manager mediante CloudFormation plantillas de AWS y Uso de referencias dinámicas para especificar valores de plantillas.

Recursos admitidos

Los siguientes recursos de AWS se admiten en el proceso de CloudFormation ingesta de AMS.

CloudFormation Ingest Stack: recursos compatibles

El sistema operativo de la instancia debe ser compatible con la ingesta de carga de trabajo de AMS. Solo se admiten los recursos de AWS que se enumeran aquí.

- Amazon API Gateway
  - AWS::ApiGateway::Account
  - AWS::ApiGateway::ApiKey
  - AWS::ApiGateway::Authorizer
  - AWS::ApiGateway::BasePathMapeo
  - 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::UsagePlanClave
- 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::UserPoolCliente
  - AWS::Cognito::UserPoolDominio
  - AWS::Cognito::UserPoolGrupo
  - AWS::Cognito::UserPoolIdentityProvider
  - AWS::Cognito::UserPoolResourceServer
  - AWS::Cognito::UserPoolRiskConfigurationAttachment

- AWS::Cognito::UserPoolUICustomizationAdjunto

- AWS::Cognito::UserPoolUser
- AWS::Cognito::UserPoolUserToGroupAttachment

- Amazon DocumentDB
  - AWS::DocBase de datos: DBCluster
  - AWS::DocDB: DBCluster ParameterGroup
  - AWS::DocDB: DBInstance
  - AWS::DocDB: DBSubnet Grupo
- 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::NetworkInterfaceAdjunto
  - AWS::EC2::SecurityGroup
  - AWS::EC2::SecurityGroupEntrada
  - AWS::EC2::SecurityGroupSalida
  - 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::SecurityGroupEntrada
  - AWS::ElastiCache::SubnetGroup
- Amazon EventBridge
  - AWS::Events::EventBus
  - AWS::Events::EventBusPolítica
  - AWS::Events::Rule
- Amazon FSx
  - AWS::FSx::FileSystem
- Amazon Inspector
  - AWS::Inspector::AssessmentTarget
  - AWS::Inspector::AssessmentTemplate
  - AWS::Inspector::ResourceGroup
- Amazon Kinesis Data Analytics
  - AWS::KinesisAnalytics::Application
  - AWS::KinesisAnalytics::ApplicationOutput
  - AWS::KinesisAnalytics::ApplicationReferenceDataSource
- Amazon Kinesis Data Firehose
  - AWS::KinesisFirehose::DeliveryStream

- Amazon Kinesis Data Streams
  - AWS::Kinesis::Stream
  - AWS::Kinesis::StreamConsumer
- Amazon MQ
  - AWS::AmazonMQ::Broker
  - AWS::AmazonMQ::Configuration
  - AWS::AmazonMQ::ConfigurationAssociation
- Amazon OpenSearch
  - AWS::OpenSearchService::Domain
- Amazon Relational Database Service (RDS)
  - AWS: :RDS: DBCluster
  - AWS: :RDS: DBCluster ParameterGroup
  - AWS: :RDS: DBInstance
  - AWS: :RDS:: Grupo DBParameter
  - AWS: :RDS:: Grupo DBSubnet
  - AWS::RDS::EventSubscription
  - AWS::RDS::OptionGroup
- Amazon Route 53
  - AWS::Route53::HealthCheck
  - AWS::Route53::HostedZone
  - AWS::Route53::RecordSet
  - AWS::Route53::RecordSetGrupo
  - AWS::Route53Resolver::ResolverRule
  - AWS::Route53Resolver::ResolverRuleAsociación
- 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
- Aplicación AutoScaling
  - AWS::ApplicationAutoScaling::ScalableTarget
  - AWS::ApplicationAutoScaling::ScalingPolicy
- Amazon EC2 AutoScaling
  - AWS::AutoScaling::AutoScalingGrupo
  - 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::WaitConditionManejar
- 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::CustomActionTipo
  - AWS::CodePipeline::Pipeline
  - AWS::CodePipeline::Webhook
- AWS Database Migration Service (DMS)
  - AWS::DMS::Certificate
  - AWS::DMS::Endpoint
  - AWS::DMS::EventSubscription
  - AWS::DMS::ReplicationInstance
  - AWS::DMS::ReplicationSubnetGrupo
  - AWS::DMS::ReplicationTask

  No se permite la MongoDbSettings propiedad del AWS::DMS::Endpoint recurso.

Las siguientes propiedades solo están permitidas si las resuelve AWS Secrets Manager: CertificateWallet las propiedades CertificatePem and del AWS::DMS::Certificate recurso y la propiedad Password del AWS::DMS::Endpoint recurso.

- AWS Elastic Load Balancing: Application Load Balance/Network Load Balancer
  - AWS::ElasticLoadBalancingV2::Listener
  - AWS::ElasticLoadBalancingV2::ListenerCertificate
  - AWS::ElasticLoadBalancingV2::ListenerRule
  - AWS::ElasticLoadBalancingV2::LoadBalancer
  - AWS::ElasticLoadBalancingV2::TargetGroup
- AWS Elastic Load Balancing: balanceador de carga clásico
  - AWS::ElasticLoadBalancing::LoadBalancer
- AWS Elemental MediaConvert
  - AWS::MediaConvert::JobTemplate
  - AWS::MediaConvert::Preset
  - AWS::MediaConvert::Queue
- AWS Elemental MediaStore
  - AWS::MediaStore::Container
- AWS Identity and Access Management (IAM)
  - AWS::IAM::Role
- Transmisión gestionada por AWS para Apache Kafka 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

- Servicio de administración de claves de AWS (KMS)
  - AWS::KMS::Key
  - AWS::KMS::Alias
- AWS Lake Formation
  - AWS::LakeFormation::DataLakeConfiguración
  - AWS::LakeFormation::Permissions
  - AWS::LakeFormation::Resource
- AWS Lambda
  - AWS::Lambda::Alias
  - AWS::Lambda::EventInvokeConfig
  - AWS::Lambda::EventSourceMapeo
  - AWS::Lambda::Function
  - AWS::Lambda::LayerVersion
  - AWS::Lambda::LayerVersionPermiso
  - AWS::Lambda::Permission
  - AWS::Lambda::Version
- Amazon Redshift
  - AWS::Redshift::Cluster
  - AWS::Redshift::ClusterParameterGrupo
  - AWS::Redshift::ClusterSubnetGrupo
- AWS Secrets Manager
  - AWS::SecretsManager::ResourcePolicy
  - AWS::SecretsManager::RotationSchedule
  - AWS::SecretsManager::Secret

- AWS::SecretsManager::SecretTargetAdjunto

- 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
- Familia 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 regional
  - AWS::WAFRegional::ByteMatchSet
  - AWS::WAFRegional::GeoMatchSet
  - AWS:WAFRegional:: IPSet
  - AWS::WAFRegional::RateBasedRegla
  - 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 Ingerir: ejemplos

Aquí encontrará algunos ejemplos detallados de cómo utilizar el tipo de cambio Create stack with CloudFormation template.

Para descargar un conjunto de CloudFormation plantillas de muestra por separado Región de AWS, consulte Plantillas de muestra.

Para obtener información de referencia sobre CloudFormation los recursos, consulte la referencia de tipos de recursos y propiedades de AWS. Sin embargo, AMS admite un conjunto de recursos más reducido, que se describen enCloudFormation Ingesta de AMS.

> **Note**
>
> AMS le recomienda que reúna todos los recursos de IAM u otros recursos relacionados con las políticas y los envíe en un único tipo de administración | Otros | Otros | Crear cambio (ct-1e1xtak34nx76). Por ejemplo, combine todas las funciones de IAM necesarias, los perfiles de instancia de IAM, las actualizaciones de las políticas de IAM para las funciones de IAM existentes, las políticas de bucket de S3, las SNS/SQS políticas, etc., y, a continuación, envíe una RFC ct-1e1xtak34nx76 para que se pueda hacer referencia a estos recursos preexistentes en las futuras plantillas de CFN Ingest.

Temas

- CloudFormation Ejemplos de ingesta: definición de recursos
- CloudFormation Ejemplos de ingesta: aplicación web de 3 niveles

CloudFormation Ejemplos de ingesta: definición de recursos

Cuando se utiliza AMS CloudFormation ingest, se personaliza una CloudFormation plantilla y se envía a AMS en una RFC con el tipo de cambio de ingesta (CloudFormation ct-36cn2avfrrj9v). Para crear una CloudFormation plantilla que se pueda reutilizar varias veces, se añaden los parámetros de configuración de la pila a la entrada de ejecución del tipo de cambio de CloudFormation ingesta, en lugar de codificarlos de forma rígida en la plantilla. CloudFormation La mayor ventaja es que puedes reutilizar la plantilla.

El esquema CloudFormation de entrada del tipo de cambio de ingesta de AMS le permite elegir hasta sesenta parámetros en una CloudFormation plantilla y proporcionar valores personalizados.

En este ejemplo, se muestra cómo definir una propiedad de recurso, que se puede utilizar en diversas CloudFormation plantillas, como parámetro en el AMS CloudFormation ingest CT. Los ejemplos de esta sección muestran específicamente el uso de los temas de SNS.

Temas

Ejemplo 1: Codificar de forma rígida la propiedad del CloudFormation SNSTopic recurso **`TopicName`**

En este ejemplo, se codifica de forma rígida la `TopicName` propiedad del CloudFormation SNSTopic recurso en la CloudFormation plantilla. Tenga en cuenta que la `Parameters` sección está vacía.

Para tener una CloudFormation plantilla que te permita cambiar el valor del SNSTopic nombre de una pila nueva sin tener que crear una CloudFormation plantilla nueva, puedes usar la `Parameters` sección AMS del tipo de cambio de CloudFormation ingesta para realizar esa configuración. De este modo, utilizarás la misma CloudFormation plantilla más adelante para crear una pila nueva con un `SNSTopic` nombre diferente.

```
{
```

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

Ejemplo 2: Utilice un SNSTopic recurso para hacer referencia a un parámetro del tipo de cambio AMS

En este ejemplo, se utiliza una `TopicName` propiedad de `SNSTopic` recurso definida en la CloudFormation plantilla para hacer referencia a una `Parameter` propiedad del tipo de cambio de 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"}
      }
    }
  }
}
```

Ejemplo 3: Cree un tema de SNS enviando un archivo de parámetros de ejecución JSON con el tipo de cambio de ingesta de AMS

En este ejemplo, envía un archivo de parámetros de ejecución de JSON con el AMS ingest CT que crea el tema de SNS. `TopicName` El tema de SNS debe definirse en la CloudFormation plantilla de la forma modificable que se muestra en este ejemplo.

```
{
  "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
}
```

Ejemplo 4: envíe un nuevo tipo de cambio que haga referencia a la misma plantilla CloudFormation

En este ejemplo de JSON se cambia el `TopicName` valor de SNS sin realizar ningún cambio en la CloudFormation plantilla. En su lugar, debe enviar un nuevo tipo de cambio Implementación | Ingestión | Pila a partir de CloudFormation plantilla | Crear que haga referencia a la misma plantilla de 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
}
```

Ejemplo 5: utilice los valores de los parámetros predeterminados en la plantilla CloudFormation

En este ejemplo, se crea el SNS `TopicName` = MyTopicName " porque no se proporcionó ningún `TopicName` valor en el parámetro de `Parameters` ejecución. Si no proporciona `Parameters` definiciones, se utilizan los valores de los parámetros predeterminados de la CloudFormation plantilla.

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

CloudFormation Ejemplos de ingesta: aplicación web de 3 niveles

Ingiera una CloudFormation plantilla para una aplicación web estándar de 3 niveles.

Esto incluye un Application Load Balancer, un grupo objetivo de Application Load Balancer, un grupo Auto Scaling, una plantilla de lanzamiento de grupos de Auto Scaling, Amazon Relational Database Service (RDS para SQL Server) con una base de datos MySQL, un almacén de parámetros SSM y AWS Secrets Manager. AWS Dedique entre 30 y 60 minutos a repasar este ejemplo.

Requisitos previos

- Cree un secreto que contenga un nombre de usuario y una contraseña con los valores correspondientes mediante el AWS Secrets Manager. Puedes consultar este ejemplo de plantilla JSON (archivo zip) que contiene el nombre `ams-shared/myapp/dev/dbsecrets` secreto y sustituirlo por tu nombre secreto. Para obtener información sobre el uso de AWS Secrets Manager con AMS, consulteUso de AWS Secrets Manager con los recursos de AMS.
- Configure los parámetros necesarios en el almacén de parámetros (PS) del AWS SSM. En este ejemplo, las `VPCId` subredes públicas y privadas se almacenan en el SSM PS en rutas como`/app/DemoApp/PublicSubnet1a`,`PublicSubnet1c`, `PrivateSubnet1a` y. `Subnet-Id` `PrivateSubnet1c` `VPCCidr` Actualice las rutas y los nombres y valores de los parámetros según sus necesidades.

- Cree un rol de EC2 instancia de Amazon de IAM con permisos de lectura para las rutas de AWS Secrets Manager y SSM Parameter Store (el rol de IAM creado y utilizado en estos ejemplos es). `customer-ec2_secrets_manager_instance_profile` Si crea políticas estándar de IAM, como el rol de perfil de instancia, el nombre del rol debe empezar por. `customer-` Para crear un nuevo rol de IAM (puede darle un nombre o cualquier otro nombre)`customer-ec2_secrets_manager_instance_profile`, utilice el tipo de cambio AMS Management | Applications | IAM instance profile | Create (ct-0ixp4ch2tiu04) CT y adjunte las políticas necesarias. Puede revisar las políticas estándar de IAM de AMS `customer_secrets_manager_policy` y`customer_systemsmanager_parameterstore_policy`, en la consola de IAM, utilizarlas tal cual o como referencia. AWS

Ingiera una CloudFormation plantilla para una aplicación web estándar de 3 niveles

1. Cargue la plantilla CloudFormation JSON de ejemplo adjunta en forma de archivo zip, en tres tier-cfn-ingest archivos.zip, en un bucket de S3 y genere una URL de S3 firmada para utilizarla en el RFC de CFN Ingest. Para obtener más información, consulte presign. La plantilla CFN también se puede incluir copy/pasted en el RFC de CFN Ingest cuando se envía el RFC a través de la consola AMS.
2. Cree un CloudFormation RFC de ingesta (implementación | Ingestión | Apila a partir de CloudFormation plantilla | Crear (ct-36cn2avfrrj9v)), ya sea mediante la consola AMS o la AMS CLI. El proceso de automatización CloudFormation de la ingesta valida la CloudFormation plantilla para garantizar que cuente con recursos válidos compatibles con AMS y cumpla con los estándares de seguridad.

   - Mediante la consola: para el tipo de cambio, seleccione Implementación -> Ingestión -> Apilar desde CloudFormation plantilla -> Crear y, a continuación, añada los siguientes parámetros como ejemplo (tenga en cuenta que el valor predeterminado de Multi es false): AZDatabase

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

- Uso del AWS CLI - Para obtener más información sobre cómo crear RFCs con el AWS CLI, consulte Creación RFCs. Por ejemplo, ejecute el siguiente comando:

```
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
```

Busque la URL de Application Load Balancer en el resultado de la ejecución de la CloudFormation RFC para acceder al sitio web. Para obtener información sobre cómo acceder a los recursos, consulta Acceder a las instancias.

Cree una pila de ingesta CloudFormation

Crear una pila de ingesta mediante CloudFormation la consola

Para crear una pila CloudFormation de ingesta mediante la consola

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs lista y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada Buscar tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir

la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.

- Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.

3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.

4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.

5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de una pila CloudFormation de ingesta mediante la CLI

Para crear una pila CloudFormation de ingesta mediante la CLI

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.

2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

1. Prepare la CloudFormation plantilla que utilizará para crear la pila y cárguela en su bucket de S3. Para obtener información importante, consulte las directrices, prácticas recomendadas y limitaciones de AWS CloudFormation Ingest.
2. Cree y envíe el RFC a AMS:

   - Cree y guarde el archivo JSON de los parámetros de ejecución e incluya los parámetros CloudFormation de la plantilla que desee. El siguiente ejemplo lo nombra CreateCfnParams .json.

     Ejemplo de CreateCfnParams archivo .json de pila de aplicaciones 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"
   }
  ],
}
```

Ejemplo de CreateCfnParams archivo.json del tema SNS:

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

3. Cree y guarde el archivo JSON de parámetros de RFC con el siguiente contenido. El siguiente ejemplo lo denomina CreateCfnRfc archivo.json:

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

4. Cree el RFC, especificando el CreateCfnRfc archivo y el CreateCfnParams archivo:

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

Recibirá el ID de la nueva RFC en la respuesta y podrá utilizarla para enviar y supervisar la RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

> **Note**
>
> Este tipo de cambio se encuentra en la versión 2.0 y es automático (no se ejecuta manualmente). Esto permite que la ejecución del CT sea más rápida y, con un nuevo parámetro CloudFormationTemplate, permite pegar en el RFC una CloudFormation plantilla personalizada. Además, en esta versión, no adjuntamos los grupos de seguridad AMS predeterminados si usted especifica sus propios grupos de seguridad. Si no especifica sus propios grupos de seguridad en la solicitud, AMS adjuntará los grupos de seguridad predeterminados de AMS. En la versión 1.0 de CFN Ingest, siempre agregábamos los grupos de seguridad predeterminados de AMS, independientemente de que usted proporcionara sus propios grupos de seguridad o no.
> AMS ha habilitado 17 servicios de AMS autoaprovisionados para su uso en este tipo de cambio. Para obtener información sobre los recursos compatibles, consulte CloudFormation Ingest Stack: Supported Resources.

> **Note**
>
> La versión 2.0 acepta un punto final de S3 que no sea una URL prefirmada.
> Si utiliza la versión anterior de este CT, el valor del parámetro CloudFormationTemplateS3Endpoint debe ser una URL prefirmada.
> Ejemplo de comando para generar una URL de bucket de S3 prefirmada (Mac/Linux):

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

> Ejemplo de comando para generar una URL de bucket de S3 prefirmada (Windows):

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

> Consulte también Creación de buckets prefirmados URLs para Amazon S3.

> **Note**
>
> Si el bucket de S3 existe en una cuenta de AMS, debe utilizar sus credenciales de AMS para este comando. Por ejemplo, es posible que tengas que añadirlas `--profile saml` después de obtener tus credenciales de AMS AWS Security Token Service (AWS STS).

Tipos de cambios relacionados:CloudFormation Aprueba un conjunto de cambios en la pila de ingesta, Actualice la pila de ingesta CloudFormation

Para obtener más información sobre AWS CloudFormation, consulte AWS CloudFormation. Para ver CloudFormation las plantillas, abra la AWS CloudFormation Template Reference.

Validar una ingesta CloudFormation

La plantilla se valida para garantizar que se pueda crear en una cuenta de AMS. Si pasa la validación, se actualiza para incluir todos los recursos o configuraciones necesarios para que se ajuste a los requisitos de AMS. Esto incluye añadir recursos como CloudWatch las alarmas de Amazon para permitir que AMS Operations supervise la pila.

La RFC se rechaza si se cumple alguna de las siguientes condiciones:

- La sintaxis JSON del RFC es incorrecta o no sigue el formato indicado.
- La URL prefirmada del bucket de S3 proporcionada no es válida.
- La plantilla no tiene una CloudFormation sintaxis válida.
- La plantilla no tiene valores predeterminados establecidos para todos los valores de los parámetros.
- La plantilla no pasa la validación de AMS. Para ver los pasos de validación de AMS, consulte la información que aparece más adelante en este tema.

La RFC falla si la CloudFormation pila no se crea debido a un problema de creación de recursos.

Para obtener más información sobre la validación y el validador de CFN, consulte Validación de plantillas e CloudFormation ingesta de pilas: ejemplos de validadores de CFN.

Actualice la pila de ingesta CloudFormation

Actualización de una pila CloudFormation de ingesta mediante la consola

Para actualizar una pila CloudFormation de ingesta mediante la consola

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs lista y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Actualización de una pila CloudFormation de ingesta mediante la CLI

Para actualizar una pila CloudFormation de ingesta mediante la CLI

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno

para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.

2. Envíe el `aws amscm submit-rfc --rfc-id` *ID* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *ID*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

1. Prepare la CloudFormation plantilla que desee usar para actualizar la pila y cárguela en su bucket de S3. Para obtener información importante, consulte las directrices, prácticas recomendadas y limitaciones de AWS CloudFormation Ingest.
2. Cree y envíe el RFC a AMS:

   - Cree y guarde el archivo JSON de los parámetros de ejecución e incluya los parámetros CloudFormation de la plantilla que desee. En este ejemplo se le asigna el nombre UpdateCfnParams .json.

     Ejemplo de UpdateCfnParams archivo.json con actualizaciones de parámetros integradas:

     ```
     {
       "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
}
```

Ejemplo de UpdateCfnParams archivo.json con un punto final de bucket de S3 que contiene una plantilla actualizada: CloudFormation

```
{
  "StackId": "stack-yjjoo9aicjyqw4ro2",
  "VpcId": "VPC_ID",
  "CloudFormationTemplateS3Endpoint": "s3_url",
  "TemplateParameters": [
    {
      "Key": "TopicName",
      "Value": "TopicNameCLI"
    },
    {
      "Key": "DisplayName",
      "Value": "DisplayNameCLI"
    }
  ],
  "TimeoutInMinutes": 1080
}
```

3. Cree y guarde el archivo JSON de parámetros de RFC con el siguiente contenido. Este ejemplo lo denomina UpdateCfnRfc archivo.json.

```
{
   "ChangeTypeId": "ct-361tlo1k7339x",
   "ChangeTypeVersion": "1.0",
```

```
   "Title": "cfn-ingest-template-update"
}
```

4. Cree el RFC, especificando el UpdateCfnRfc archivo y el UpdateCfnParams archivo:

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

   Recibirá el ID de la nueva RFC en la respuesta y podrá utilizarla para enviar y supervisar la RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

- Este tipo de cambio se encuentra ahora en la versión 2.0. Los cambios incluyen la eliminación del AutoApproveUpdateForResourcesparámetro, que se utilizó en la versión 1.0 de este CT, y la adición de dos parámetros nuevos: AutoApproveRiskyUpdatesy BypassDriftCheck.
- Si el bucket de S3 existe en una cuenta de AMS, debe utilizar sus credenciales de AMS para este comando. Por ejemplo, es posible que tengas que añadirlas `--profile saml` después de obtener tus credenciales de AMS AWS Security Token Service (AWS STS).
- Todos `Parameter` los valores de los recursos de la CloudFormation plantilla deben tener un valor, ya sea por defecto o personalizado en la sección de parámetros del CT. Puede anular el valor del parámetro estructurando los recursos de la CloudFormation plantilla para que hagan referencia a una clave de parámetros. Para ver ejemplos que muestran cómo hacerlo, consulta la pila de CloudFormation ingesta: ejemplos de validadores CFN.

  IMPORTANTE: Los parámetros que faltan no se proporcionan de forma explícita en el formulario; por defecto, son los valores establecidos actualmente en la pila o plantilla existente.
- Para ver una lista de los servicios autoaprovisionados que puede añadir mediante CloudFormation Ingest, consulte CloudFormation Ingest Stack: recursos compatibles.

  Para obtener más información CloudFormation, consulte AWS CloudFormation.

Validación de una ingesta CloudFormation

La plantilla se valida para garantizar que se pueda crear en una cuenta de AMS. Si pasa la validación, se actualiza para incluir todos los recursos o configuraciones necesarios para que se ajuste a los requisitos de AMS. Esto incluye añadir recursos como CloudWatch las alarmas de Amazon para permitir que AMS Operations supervise la pila.

La RFC se rechaza si se cumple alguna de las siguientes condiciones:

- La sintaxis JSON del RFC es incorrecta o no sigue el formato indicado.
- La URL prefirmada del bucket S3 proporcionada no es válida.
- La plantilla no tiene una CloudFormation sintaxis válida.
- La plantilla no tiene valores predeterminados establecidos para todos los valores de los parámetros.
- La plantilla no pasa la validación de AMS. Para ver los pasos de validación de AMS, consulte la información que aparece más adelante en este tema.

La RFC falla si la CloudFormation pila no se crea debido a un problema de creación de recursos.

Para obtener más información sobre la validación y el validador de CFN, consulte Validación de plantillas e CloudFormation ingesta de pilas: ejemplos de validadores de CFN.

CloudFormation Aprueba un conjunto de cambios en la pila de ingesta

Aprobar y actualizar una pila de CloudFormation ingesta mediante la consola

Para aprobar y actualizar una pila de CloudFormation ingesta mediante la consola

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs lista y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.

3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Aprobación y actualización de una pila CloudFormation de ingesta mediante la CLI

Para aprobar y actualizar una pila CloudFormation de ingesta mediante la CLI

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y de RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista

de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

1. Envía el esquema JSON de los parámetros de ejecución para este tipo de cambio a un archivo de tu carpeta actual. En este ejemplo se le asigna el nombre 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. Modifique y guarde el esquema de la siguiente manera:

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

Consejos

> **Note**
>
> Si hay varios recursos en una pila y desea eliminar solo un subconjunto de los recursos de la pila, utilice el botón de CloudFormation actualización; consulte CloudFormation Ingest Stack: Updating. También puedes enviar un caso de solicitud de servicio y los ingenieros de AMS te ayudarán a diseñar el conjunto de cambios, si es necesario.

Para obtener más información AWS CloudFormation, consulte AWS CloudFormation.

Protección de terminación CloudFormation de Update Stacks

Actualización de una pila de protección de CloudFormation terminaciones con la consola

A continuación se muestra este tipo de cambio en la consola AMS.

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Actualización de una protección de terminación de CloudFormation pila con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.

2. Envíe el `aws amscm submit-rfc --rfc-id` *ID* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *ID*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

Especifique únicamente los parámetros que desee cambiar. Los parámetros ausentes conservan los valores existentes.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON; este ejemplo lo nombra EnableTermPro CFNParams .json:

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

2. Modifique y guarde el EnableTermPro CFNParams archivo, conservando solo los parámetros que desee cambiar. Por ejemplo, puede reemplazar el contenido por algo como esto:

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

3. Envía la plantilla RFC a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre EnableTermPro CFNRfc .json:

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

4. Modifique y guarde el EnableTermPro CFNRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

5. Cree el RFC, especificando el EnableTermPro CFNRfc archivo y el EnableTermPro CFNParams archivo:

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

   Recibirá el ID de la nueva RFC en la respuesta y podrá utilizarla para enviar y supervisar la RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

> **Note**
>
> Hay un CT relacionado con Amazon EC2, EC2 pila: Actualización de la protección de rescisión.

Para obtener más información sobre la protección por terminación, consulta Cómo proteger una pila para que no se elimine.

Implementaciones de IAM automatizadas mediante la ingesta de CFN o la actualización de pilas en AMS CTs

Puedes usar estos tipos de cambios de AMS para implementar funciones de IAM (el `AWS::IAM::Role` recurso) tanto en la zona de aterrizaje multicuenta (MALZ) como en la zona de aterrizaje de una sola cuenta (SALZ):

- Despliegue | Ingestión | Apilación a partir de plantilla | Creación (ct-36cn2avfrrj9v CloudFormation)
- Administración | Pila personalizada | Apilación a partir de plantilla | Actualización (ct-361tlo1k7339x) CloudFormation
- Administración | Pila personalizada | Apilación a partir de plantilla | Aprobar y actualizar (ct-1404e21baa2ox) CloudFormation

Validaciones realizadas en las funciones de IAM de su plantilla CFN:

- ManagedPolicyArns: El atributo no ManagedPolicyArnsdebe existir en. `AWS::IAM::Role` La validación no permite adjuntar políticas administradas al rol que se está aprovisionando. En su lugar, los permisos del rol se pueden administrar mediante la política en línea a través de las políticas de propiedad.
- PermissionsBoundary: La política utilizada para establecer el límite de permisos para el rol solo puede ser la política gestionada por AMS:. `AWSManagedServices_IAM_PermissionsBoundary` Esta política actúa como una barrera que protege los recursos de la infraestructura de AMS para que no se modifiquen mediante el rol que se está aprovisionando. Con este límite de permisos predeterminado, se mantienen los beneficios de seguridad que proporciona AMS.

El `AWSManagedServices_IAM_PermissionsBoundary` (predeterminado) es obligatorio; sin él, la solicitud se rechaza.

- MaxSessionDuration: La duración máxima de sesión que se puede configurar para el rol de IAM es de 1 a 4 horas. El estándar técnico de AMS exige que el cliente asuma el riesgo de que la sesión dure más de 4 horas.
- RoleName: AMS conserva los siguientes espacios de nombres y no se pueden usar como prefijos de nombres de funciones de 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
```

- Políticas: la política integrada en la función de IAM solo puede incluir un conjunto de acciones de IAM aprobadas previamente por AMS. Este es el límite superior de todas las acciones de IAM con las que se puede crear un rol de IAM (política de control). La política de control consiste en:
  - Todas las acciones de la política AWS gestionada ReadOnlyAccess que proporcionan acceso de solo lectura a todos los Servicios de AWS recursos y
  - Las siguientes acciones, con la restricción de las acciones de S3 entre cuentas, es decir, las acciones de S3 permitidas, solo se pueden realizar en los recursos presentes en la misma cuenta en la que se encuentra el rol que se está creando:

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

Cualquier función de IAM creada o actualizada mediante la ingesta de CFN puede permitir las acciones enumeradas en esta política de control o acciones que tengan un alcance inferior (menos permisivo que) las acciones enumeradas en la política de control. Actualmente, permitimos estas acciones de IAM seguras, que pueden clasificarse como acciones de solo lectura, además de las acciones que no son de solo lectura mencionadas anteriormente, que no se pueden llevar a cabo y que están aprobadas previamente según la norma técnica de AMS. CTs

- AssumeRolePolicyDocument: Las siguientes entidades están preaprobadas y se pueden incluir en la política de confianza para que asuman la función que se va a crear:
  - Cualquier entidad de IAM (rol, usuario, usuario raíz, sesión con el rol asumido por STS) de la misma cuenta puede asumir el rol.
  - Las siguientes personas Servicios de AWS pueden asumir la función:

```
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
```

- El proveedor de SAML de la misma cuenta puede asumir la función. Actualmente, el único nombre de proveedor de SAML compatible es. `customer-saml`

Si una o más de las validaciones fallan, se rechaza la RFC. Un ejemplo de motivo de rechazo de un RFC es el siguiente:

```
{"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 necesita ayuda con una validación o ejecución fallida de la RFC, utilice la correspondencia de la RFC para ponerse en contacto con AMS. Para obtener instrucciones, consulte la correspondencia y el archivo adjunto de la RFC (consola). Para cualquier otra pregunta, envíe una solicitud de servicio. Para obtener información sobre cómo hacerlo, consulte Creación de una solicitud de servicio.

> **Note**
>
> Actualmente, no aplicamos ninguna de las mejores prácticas de IAM como parte de nuestras validaciones de IAM. Para conocer las prácticas recomendadas de IAM, consulte Prácticas recomendadas de seguridad en IAM.

Crear funciones de IAM con acciones más permisivas o aplicar las mejores prácticas de IAM

Cree sus entidades de IAM con los siguientes tipos de cambios manuales:

- Implementación | Componentes de pila avanzados | Identity and Access Management (IAM) | Crear entidad o política (ct-3dpd8mdd9jn1r)
- Administración | Componentes de pila avanzados | Identity and Access Management (IAM) | Actualizar entidad o política (ct-27tuth19k52b4)

Le recomendamos que lea y comprenda nuestras normas técnicas antes de archivar este manual. RFCs Para acceder a ellas, consulte Cómo acceder a las normas técnicas.

> **Note**
>
> Cada función de IAM creada directamente con estos tipos de cambios manuales pertenece a su propia pila individual y no reside en la misma pila en la que se crean los demás recursos de infraestructura mediante CFN Ingest CT.

La actualización de las funciones de IAM creadas con CFN se realiza mediante cambios manuales cuando las actualizaciones no se pueden realizar mediante tipos de cambios automatizados

Utilice el tipo de cambio Management | Advanced stack components | Identity and Access Management (IAM) | Actualizar entidad o política (ct-27tuth19k52b4).

> **Important**
>
> Las actualizaciones de las funciones de IAM realizadas mediante el CT manual no se reflejan en las plantillas de la pila de CFN y provocan una desviación de la pila. Una vez que el rol se haya actualizado mediante una solicitud manual a un estado que no supere nuestras validaciones, no se podrá volver a actualizar con el Stack Update CT (ct-361tlo1k7339x)

mientras siga sin cumplir con nuestras validaciones. El CT de actualización solo se puede usar si la plantilla de pila CFN cumple con nuestras validaciones. Sin embargo, la pila se puede seguir actualizando mediante el Stack Update CT (ct-361tlo1k7339x), siempre y cuando el recurso de IAM que no cumpla con nuestras validaciones no se actualice y la plantilla CFN las supere.

Eliminar las funciones de IAM creadas mediante la ingesta AWS CloudFormation

Si desea eliminar toda la pila, utilice el siguiente tipo de cambio automático de eliminación de la pila. Para obtener instrucciones, consulta Eliminar pila:

- ID de tipo de cambio: ct-0q0bic0ywqk6c
- Clasificación: Administración | Pilas estándar | Apilación | Eliminación y administración | Componentes de pila avanzados | Apilación | Eliminar

Si desea eliminar un rol de IAM sin eliminar toda la pila, puede eliminar el rol de IAM de la CloudFormation plantilla y utilizar la plantilla actualizada como entrada para el tipo de cambio de actualización automática de la pila:

- ID de tipo de cambio: ct-361tlo1k7339x
- Clasificación: Gestión | Pila personalizada | Apilación a partir de plantilla | Actualización CloudFormation

Para obtener instrucciones, consulte Actualizar la pila AWS CloudFormation de ingesta.

CodeDeploy solicitudes

Puede usar AWS CodeDeploy para crear contenedores de aplicaciones que luego puede implementar a través de un grupo de CodeDeploy aplicaciones. Para obtener más información CodeDeploy, consulte la CodeDeploy documentación de AWS.

Trabajar con AWS CodeDeploy implica el siguiente proceso:

1. Cree una CodeDeploy aplicación. La CodeDeploy aplicación es un nombre o contenedor que se utiliza CodeDeploy para garantizar que se haga referencia a la revisión, la configuración de implementación y el grupo de implementación correctos durante una implementación.

2. Cree un grupo CodeDeploy de despliegues. Un grupo de CodeDeploy implementación define un conjunto de instancias individuales destinadas a una implementación. AMS tiene un tipo de cambio independiente para los grupos de CodeDeploy despliegue EC2.
3. Implemente la CodeDeploy aplicación a través del grupo CodeDeploy de implementación.

CodeDeploy aplicación

Cree o despliegue CodeDeploy aplicaciones.

Crear una CodeDeploy aplicación

Crear una CodeDeploy aplicación con la consola

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada Buscar tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.

4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de una CodeDeploy aplicación con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución del esquema JSON de la CodeDeploy aplicación CT a un archivo de tu carpeta actual; en este ejemplo se llama Create CDApp Params.json:

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

2. Modifique y guarde el archivo JSON de la siguiente manera. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

3. Envía la plantilla JSON CreateRfc a un archivo de tu carpeta actual; en este ejemplo se llama Create CDApp RFC.json:

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

4. Modifica y guarda el archivo JSON de la siguiente manera. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   ```

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

5. Cree el RFC, especificando el archivo Create CDApp Rfc y el archivo de parámetros de ejecución:

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

   Recibirá el ID del nuevo RFC en la respuesta y podrá usarlo para enviar y monitorear el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

Para obtener más información sobre AWS CodeDeploy, consulte Crear una aplicación con AWS CodeDeploy.

Implemente CodeDeploy una aplicación

Implementación de una CodeDeploy aplicación con la consola

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada Buscar tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.

- Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.

3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Implementación de una CodeDeploy aplicación con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando

cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}}"
```

CREACIÓN DE PLANTILLA:

1. Genera el esquema JSON de los parámetros de ejecución para el CT de despliegue de la CodeDeploy aplicación; este ejemplo lo denomina Deploy CDApp Params.json:

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

2. Modifique el archivo JSON de la siguiente manera. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "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. Envía la plantilla JSON CreateRfc a un archivo de tu carpeta actual; en este ejemplo se llama Deploy CDApp RFC.json:

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

4. Modifica y guarda el archivo Deploy CDApp rfc.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

5. Cree el RFC especificando el archivo de parámetros de ejecución y el archivo CDApp RFC de implementación:

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

   Recibirá el ID del nuevo RFC en la respuesta y podrá usarlo para enviar y monitorear el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

Para obtener más información, consulte Crear una implementación con CodeDeploy.

CodeDeploy grupos de despliegue

Cree grupos CodeDeploy de aplicaciones.

Cree un grupo CodeDeploy de despliegue

Crear un grupo CodeDeploy de despliegue con la consola

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada Buscar tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de un grupo CodeDeploy de implementación con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}}"
```

CREACIÓN DE PLANTILLA:

1. Envía el esquema JSON de los parámetros de ejecución a un archivo de tu carpeta actual; en este ejemplo se llama Create CDDep GroupParams .json:

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

2. Modifica y guarda el archivo JSON. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
"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. Envía la plantilla JSON CreateRfc a un archivo de tu carpeta actual; en este ejemplo se llama Create CDDep GroupRfc .json:

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

4. Modifica y guarda el archivo JSON. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
"ChangeTypeVersion":    "1.0",
```

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

5. Cree el RFC, especificando el archivo de creación y el CDDep GroupRfc archivo de parámetros de ejecución:

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

   Recibirá el ID de la nueva RFC en la respuesta y podrá utilizarla para enviar y supervisar la RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

Para obtener más información sobre los grupos de CodeDeploy implementación de AWS, consulte Crear un grupo de implementación con AWS CodeDeploy.

Cree un grupo de CodeDeploy implementación para EC2

Crear un grupo CodeDeploy de despliegue para EC2 con la consola

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada Buscar tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.

- Elegir por categoría: seleccione una categoría, subcategoría, elemento y operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.

3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Crear un grupo CodeDeploy de implementación para EC2 con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando

cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

CREACIÓN DE PLANTILLA:

1. Envía el esquema JSON de los parámetros de ejecución a un archivo; este ejemplo lo denomina Create CDDep GroupEc 2Params.json:

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

2. Modifique y guarde el archivo JSON. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "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. Envía la plantilla JSON CreateRfc a un archivo de tu carpeta actual; en este ejemplo se llama Create CDDep GroupEc 2rFC.json:

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

4. Modifica y guarda el archivo JSON. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

5. Cree el RFC, especificando el archivo Create CDDep GroupEc 2Rfc y el archivo de parámetros de ejecución:

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

   Recibirá el ID del nuevo RFC en la respuesta y podrá usarlo para enviar y monitorear el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

Para obtener más información sobre los grupos de CodeDeploy implementación de AWS, consulte Crear un grupo de implementación con AWS CodeDeploy.

AWS Database Migration Service (AWS DMS)

AWS Database Migration Service (AWS DMS) le ayuda a migrar las bases de datos a AMS de forma fácil y segura. Puede migrar datos hacia y desde la mayoría de las bases de datos comerciales y de

código abierto más utilizadas, como Oracle, MySQL y PostgreSQL. El servicio admite migraciones homogéneas, como Oracle a Oracle, y también migraciones heterogéneas entre diferentes plataformas de bases de datos, como Oracle a PostgreSQL o MySQL a Oracle. AWS DMS es un AWS servicio; los AMS le CTs ayudan a crear AWS DMS recursos en su cuenta gestionada por AMS

El siguiente gráfico muestra el flujo de trabajo de la migración de una base de datos.

Temas

AWS Database Migration Service (AWS DMS), antes de empezar

Al planificar una migración de base de datos mediante el AMS AWS DMS, tenga en cuenta lo siguiente:

- Puntos finales de origen y destino: debe saber qué información y tablas de la base de datos de origen deben migrarse a la base de datos de destino. AMS AWS DMS admite la migración básica de esquemas, incluida la creación de tablas y claves principales. Sin embargo, AMS AWS DMS no crea automáticamente índices secundarios, claves externas, cuentas, etc. en la base de datos de destino. Consulte Fuentes de migración de datos y Objetivos de la migración de datos para obtener más información.
- Migración de esquemas o códigos: AMS AWS DMS no realiza conversiones de esquemas o códigos. Puede utilizar herramientas como Oracle SQL Developer, MySQL Workbench o pgAdmin III para convertir el esquema. Si desea convertir un esquema existente en un motor de base de datos diferente, puede utilizar la herramienta de conversión de esquemas de AWS. Puede crear un esquema de destino y también generar y crear un esquema completo: tablas, índices, vistas y así sucesivamente. También puede utilizar la herramienta para convertir PL/SQL TSQL a PgSQL y otros formatos.
- Tipos de datos no compatibles: algunos tipos de datos de origen deben convertirse en tipos de datos equivalentes a los de la base de datos de destino.

AWS DMS escenarios a tener en cuenta

Los siguientes escenarios documentados pueden ayudarle a diseñar su propia ruta de migración de bases de datos.

- Migre datos de un servidor MySQL local a Amazon RDS MySQL: consulte la entrada del blog de AWS Migración de datos de MySQL locales a Amazon RDS (y viceversa)
- Migre datos de una base de datos Oracle a una base de datos Aurora PostgreSQL de Amazon RDS: consulte la entrada del blog de AWS Introducción rápida a la migración de una base de datos Oracle a una base de datos PostgreSQL de Amazon Aurora
- Migre datos de RDS MySQL a S3: consulte la entrada del blog de AWS Cómo archivar datos de bases de datos relacionales en Amazon Glacier con AWS DMS

Para migrar una base de datos, debe hacer lo siguiente:

- Planifique la migración de su base de datos, lo que incluye la configuración de un grupo de subredes de replicación.
- Asigne una instancia de replicación que realice todos los procesos de la migración.
- Especifique un punto final de base de datos de origen y uno de destino.
- Crear una tarea o conjunto de tareas para definir qué tablas y procesos de replicación desea utilizar.
- Cree el AWS DMS IAM `dms-cloudwatch-logs-role` y las `dms-vpc-role` funciones. Si utiliza Amazon Redshift como base de datos de destino, también debe crear y añadir el rol de IAM a `dms-access-for-endpoint` su cuenta de AWS. Para obtener más información, consulte Crear las funciones de IAM para usarlas con la AWS CLI y la API de AWS DMS.

Estos tutoriales proporcionan un ejemplo del uso de la consola AMS o la CLI de AMS para crear un AWS Database Migration Service ()AWS DMS. Se proporcionan comandos de CLI para crear la instancia de AWS DMS replicación, el grupo de subredes y la tarea, así como un punto final de AWS DMS origen y un punto final de destino.

Para obtener más información sobre AMS AWS DMS, consulte AWS Database Migration Servicela información general y las respuestas a AWS Database Migration Service FAQslas preguntas más frecuentes.

AWS DMS, datos necesarios para la configuración

Para cada uno de los siguientes AWS DMS tutoriales, se necesitan algunos datos en común.

- `Description`: Información significativa sobre el recurso, separada de otras opciones de parámetros`Description`.
- `VpcId`: La VPC que se va a utilizar. Para averiguarlo, ejecute el ListVpcSummaries funcionamiento de la API SKMS (`list-vpc-summaries`en la CLI) o consulte la VPCspágina de la consola AMS. Para ver la referencia de la API AMS SKMS, consulte la pestaña Informes de AWS Artifact Console.
- `Name`: un nombre para la pila o el componente de la pila; se convierte en el nombre de la pila.
- `TimeoutInMinutes`: Cuántos minutos se necesitan para crear la pila antes de que se produzca un error en la RFC. Esta configuración no retrasará la ejecución de la RFC, pero debes dedicarle tiempo suficiente (por ejemplo, no `"5"` especificarlo).
- `ChangeTypeIdChangeTypeVersion`, y`StackTemplateId`: Son obligatorios, pero varían según el CT y sus valores se indican en cada sección correspondiente, a continuación.

AWS DMS tareas de configuración

Configure AWS DMS con los siguientes tutoriales.

1: grupo de subredes de AWS DMS replicación: Crear

Puede usar la consola AMS o API/CLI crear un grupo de subredes de AWS DMS replicación AMS.

Cree un grupo de AWS DMS subredes de replicación

Crear un grupo de subredes de AWS DMS replicación con la consola

> **Note**
>
> Este CT falla si la función de `dms-vpc-role` IAM no existe en la cuenta.

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.

- Busque por tipo de cambio: puede hacer clic en un CT popular del área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

  Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
- Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.

3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de un grupo de subredes de AWS DMS replicación con la CLI

> **Note**
>
> Este CT falla si la función de `dms-vpc-role` IAM no existe en la cuenta.

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.

2. Envíe el `aws amscm submit-rfc --rfc-id` *ID* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *ID*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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
\"}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON; este ejemplo lo nombra CreateDmsRsgParams .json:

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

2. Modifique y guarde el CreateDmsRsgParams archivo.json de los parámetros de ejecución. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

3. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre CreateDmsRsgRfc .json:

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

4. Modifica y guarda el CreateDmsRsgRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

5. Cree el RFC especificando el archivo de parámetros de ejecución y el CreateDmsRsgRfc archivo:

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

Recibirá el ID del nuevo RFC en la respuesta y podrá utilizarlo para enviar y supervisar el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

- Este CT falla si la función de `dms-vpc-role` IAM no existe en la cuenta.
- Puede añadir hasta 50 etiquetas, pero para ello debe habilitar la vista de configuración adicional.

  Para obtener más información sobre las instancias de replicación y los grupos de subredes del DMS, consulte Configuración de una red para una instancia de replicación.

2: instancia de AWS DMS replicación: Crear

Puede usar la consola AMS o API/CLI crear una instancia de AWS DMS replicación de AMS.

Cree una instancia AWS DMS de replicación

Crear una instancia AWS DMS de replicación con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.

4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de una instancia de AWS DMS replicación con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}"
```

Mientras se está creando la instancia de replicación, puede especificar los almacenes de datos de origen y de destino. Los almacenes de datos de origen y destino pueden estar en una instancia de Amazon Elastic Compute Cloud (Amazon EC2), un bucket de AWS S3, una instancia de base de datos de Amazon Relational Database Service (Amazon RDS) o una base de datos local.

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON; este ejemplo lo nombra CreateDmsRiParams .json:

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

2. Modifique y guarde el CreateDmsRiParams archivo.json de los parámetros de ejecución. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "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. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre CreateDmsRiRfc .json:

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

4. Modifica y guarda el CreateDmsRiRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

5. Cree el RFC especificando el archivo de parámetros de ejecución y el CreateDmsRiRfc archivo:

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

   Recibirá el ID del nuevo RFC en la respuesta y podrá utilizarlo para enviar y supervisar el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

- Puede añadir hasta 50 etiquetas, pero para ello debe habilitar la vista de configuración adicional.
- Debe crear una instancia de replicación en una EC2 instancia de la VPC de AMS que tenga suficiente capacidad de almacenamiento y procesamiento para realizar las tareas que asigne y migrar los datos de la base de datos de origen a la base de datos de destino. El tamaño necesario para esta instancia varía en función de la cantidad de datos que deba migrar y las tareas que necesita que efectúe la instancia. La instancia de replicación proporciona alta disponibilidad y compatibilidad con la conmutación por error mediante un despliegue Multi-AZ al seleccionar la opción. `MultiAZ` Para obtener más información sobre las instancias de replicación, consulte Trabajar con una instancia de replicación de AWS DMS.

3: punto final de AWS DMS origen: crear, crear para Mongo DB, crear para S3

Puede utilizar la consola AMS o API/CLI crear un punto final de origen del AMS DMS para varias bases de datos. Le ofrecemos tres ejemplos.

Punto final de origen del DMS: creación

Creación de un punto final de origen del DMS con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.

5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de un punto final de origen de DMS con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON denominado CreateDmsSeParams .json.

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

2. Modifique y guarde los parámetros de ejecución en el archivo JSON. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "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. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre CreateDmsSeRfc .json:

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

4. Modifica y guarda el CreateDmsSeRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

5. Cree el RFC especificando el archivo de parámetros de ejecución y el CreateDmsSeRfc archivo:

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

Recibirá el ID del nuevo RFC en la respuesta y podrá utilizarlo para enviar y supervisar el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

Antes de crear el punto final del DMS, asegúrese de que la contraseña no contenga caracteres no admitidos. Para obtener más información, consulte Creación de puntos finales de origen y destino en la Guía del AWS Database Migration Service usuario.

Para obtener más información, consulte Fuentes para la migración de datos.

Para ver un punto final de origen de S3, consultePunto final de origen de DMS para S3: creación.

Para ver un punto final de origen de Mongo DB, consultePunto final de origen de DMS para MongoDB: creación.

Punto final de origen de DMS para MongoDB: creación

Creación de un punto final de origen de base de datos DMS Mongo con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.

- Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

  Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
- Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.

3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de un punto final de origen de base de datos DMS Mongo con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON denominado CreateDmsSeMongoParams .json.

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

2. Modifique y guarde los parámetros de ejecución en el archivo JSON. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
"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. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre CreateDmsSeMongoRfc .json:

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

4. Modifica y guarda el CreateDmsSeMongoRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

5. Cree el RFC especificando el archivo de parámetros de ejecución y el CreateDmsSeMongoRfc archivo:

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

Recibirá el ID del nuevo RFC en la respuesta y podrá utilizarlo para enviar y supervisar el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

> **Note**
>
> Puede añadir hasta 50 etiquetas, pero para ello debe habilitar la vista de configuración adicional.

AMS DMS puede utilizar Mongo o cualquier Relational Database Service (RDS) Relational Database Service (RDS) como punto final de origen. Para ver un punto final de origen de S3, consulte. Punto final de origen de DMS para S3: creación

Punto final de origen de DMS para S3: creación

Creación de un punto final de origen del DMS S3 con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.

4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de un punto final de origen DMS S3 con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON denominado CreateDmsSe S3Params.json.

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

2. Modifique y guarde el archivo JSON de los parámetros de ejecución. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "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. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre CreateDmsSe S3RFC.json:

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

4. Modifica y guarda el archivo S3RFC.json. CreateDmsSe Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2oxl37nphsrjz",
"Title":                "DMS_Source_S3"
}
```

5. Cree el RFC especificando el archivo de parámetros de ejecución y el archivo CreateDmsSe S3Rfc:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeS3Rfc.json --execution-
parameters file://CreateDmsSeS3Params.json
```

Recibirá el ID del nuevo RFC en la respuesta y podrá usarlo para enviar y monitorear el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

> **Note**
>
> Puede añadir hasta 50 etiquetas, pero para ello debe habilitar la vista de configuración adicional.

El AMS DMS puede utilizar S3 o cualquier punto final de origen del Relational Database Service (RDS). Para ver un punto final de origen de Mongo DB, consulte. Punto final de origen de DMS para MongoDB: creación

4: punto final de AWS DMS destino: crear, crear para S3

Puede utilizar la consola AMS o API/CLI crear un punto final de destino del AMS DMS para varias bases de datos. Le ofrecemos dos ejemplos.

Punto final de destino del DMS: creación

AMS DMS puede utilizar S3 o cualquier Relational Database Service (RDS) con MySQL, MariaDB, Oracle, Postgresql o Microsoft SQL como punto final de destino.

Creación de un punto final de destino de DMS con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir

la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.

- Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.

3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.

4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de un punto final de destino de DMS con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON denominado CreateDmsTeParams .json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3gf8dolbo8x9p" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeParams.json
   ```

2. Modifique y guarde los parámetros de ejecución en el archivo JSON. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "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. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre CreateDmsTeRfc .json:

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

4. Modifica y guarda el CreateDmsTeRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3gf8dolbo8x9p",
"Title":                "DB-DMS-Target-Endpoint"
}
```

5. Cree el RFC especificando el archivo de parámetros de ejecución y el CreateDmsTeRfc archivo:

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeRfc.json --execution-
parameters file://CreateDmsTeParams.json
```

Recibirá el ID del nuevo RFC en la respuesta y podrá utilizarlo para enviar y supervisar el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

- Este tipo de cambio se encuentra ahora en la versión 2.0.
- AMS DMS puede utilizar S3 o cualquier Relational Database Service (RDS) con MySQL, MariaDB, Oracle, Postgresql o Microsoft SQL como punto final de destino. Para ver un punto final de destino de S3, consulte. Punto final de DMS para S3: creación
- Para obtener más información, consulte Objetivos de la migración de datos.

- Puede añadir hasta 50 etiquetas, pero para ello debe habilitar la vista de configuración adicional.

Punto final de DMS para S3: creación

Creación de un punto final de destino de DMS S3 con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.

5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de un punto final de destino DMS S3 con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\"}}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON; este ejemplo lo denomina CreateDmsTe S3Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-05muqzievnxk5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeS3Params.json
   ```

2. Modifique y guarde los parámetros de ejecución en el archivo S3Params.json. CreateDmsTe Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   "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. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre CreateDmsTe S3RFC.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateDmsTeS3Rfc.json
   ```

4. Modifica y guarda el archivo S3RFC.json. CreateDmsTe Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
   ```

```
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-05muqzievnxk5",
"Title":                "DMS_Target_S3"
}
```

5. Cree el RFC especificando el archivo de parámetros de ejecución y el archivo CreateDmsTe S3Rfc:

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeS3Rfc.json --execution-
parameters file://CreateDmsTeS3Params.json
```

Recibirá el ID del nuevo RFC en la respuesta y podrá usarlo para enviar y monitorear el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

> **Note**
>
> Puede añadir hasta 50 etiquetas, pero para ello debe habilitar la vista de configuración adicional.

AMS proporciona un tipo de cambio independiente para crear un punto final de destino para S3. Para obtener más información, consulte Uso de Amazon S3 como destino para AWS Database Migration Service y Atributos de conexión adicionales al utilizar Amazon S3 como destino para AWS DMS.

5: tarea de AWS DMS replicación: crear

Puede utilizar la consola AMS o API/CLI crear una tarea de AWS DMS replicación de AMS.

Cree una tarea AWS DMS de replicación

Creación de una tarea de AWS DMS replicación con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.

2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Creación de una tarea de AWS DMS replicación con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON; este ejemplo lo nombra CreateDmsRtParams .json:

```
aws amscm get-change-type-version --change-type-id "ct-1d2fml15b9eth" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRtParams.json
```

2. Modifique y guarde el archivo JSON de los parámetros de ejecución. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
"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. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre CreateDmsRtRfc .json:

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

4. Modifica y guarda el CreateDmsRtRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1d2fml15b9eth",
"Title":                "DMS-RI-Create-RFC"
}
```

5. Cree el RFC especificando el archivo de parámetros de ejecución y el CreateDmsRtRfc archivo:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRtRfc.json --execution-
parameters file://CreateDmsRtParams.json
```

Recibirá el ID del nuevo RFC en la respuesta y podrá utilizarlo para enviar y supervisar el RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

Puede crear una AWS DMS tarea que capture tres tipos diferentes de cambios o datos. Para obtener más información, consulte Trabajar con tareas de AWS DMS, Crear una tarea y Crear tareas para la replicación continua mediante AWS DMS.

AWS DMS gestión

AWS DMS ejemplos de gestión.

Inicie la tarea AWS DMS de replicación

Iniciar una tarea de AWS DMS replicación con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.

- Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.

3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Inicio de una tarea de AWS DMS replicación con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.
2. Envíe el `aws amscm submit-rfc --rfc-id` *`ID`* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *`ID`*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando

cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
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\":[\"\"]}}"
```

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON; este ejemplo lo nombra StartDmsRtParams .json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1yq7hhqse71yg" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StartDmsRtParams.json
   ```

2. Modifique y guarde el archivo JSON de los parámetros de ejecución. Por ejemplo, puedes reemplazar el contenido por algo como esto:

   ```
   {
     "DocumentName": "AWSManagedServices-StartDmsTask",
     "Region": "us-east-1",
     "Parameters": {
         "ReplicationTaskArn": [
           "TASK_ARN"
         ],
         "StartReplicationTaskType": [
           "start-replication"
         ],
         "CdcStartPosition": [
   ```

```
        ""
      ],
      "CdcStopPosition": [
        ""
      ]
  }
}
```

3. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre StartDmsRtRfc .json:

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

4. Modifica y guarda el StartDmsRtRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
  "ChangeTypeId": "ct-1yq7hhqse71yg",
  "ChangeTypeVersion": "1.0",
  "Title": "Start DMS Replication Task"
}
```

5. Cree el RFC especificando el archivo de parámetros de ejecución y el StartDmsRtRfc archivo:

```
aws amscm create-rfc --cli-input-json file://StartDmsRtRfc.json --execution-
parameters file://StartDmsRtParams.json
```

   Recibirá el ID de la nueva RFC en la respuesta y podrá utilizarla para enviar y supervisar la RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

Puede iniciar una tarea de AWS DMS replicación mediante la consola AMS o la API/CLI de AMS. Para obtener más información, consulte Trabajar con tareas de AWS DMS.

Detenga la tarea AWS DMS de replicación

Detener una tarea AWS DMS de replicación con la consola

Captura de pantalla de este tipo de cambio en la consola AMS:

Cómo funciona:

1. Vaya a la página Crear RFC: en el panel de navegación izquierdo de la consola AMS, haga clic RFCspara abrir la página de RFCs listas y, a continuación, haga clic en Crear RFC.
2. Elija un tipo de cambio (CT) popular en la vista predeterminada de búsqueda de tipos de cambios o seleccione un CT en la vista Elegir por categoría.
   - Busque por tipo de cambio: puede hacer clic en un CT popular en el área de creación rápida para abrir inmediatamente la página Ejecutar RFC. Tenga en cuenta que no puede elegir una versión antigua de CT con Quick Create.

     Para ordenar CTs, utilice el área Todos los tipos de cambios en la vista de tarjeta o de tabla. En cualquiera de las vistas, seleccione una CT y, a continuación, haga clic en Crear RFC para abrir la página Ejecutar RFC. Si corresponde, aparece la opción Crear con una versión anterior junto al botón Crear RFC.
   - Elegir por categoría: seleccione una categoría, una subcategoría, un elemento y una operación, y se abrirá el cuadro de detalles del CT con la opción Crear con una versión anterior, si corresponde. Haga clic en Crear RFC para abrir la página Ejecutar RFC.
3. En la página Ejecutar RFC, abra el área del nombre del CT para ver el cuadro de detalles del CT. Se requiere un asunto (lo rellena automáticamente si elige su CT en la vista Buscar tipos de cambios). Abra el área de configuración adicional para añadir información sobre la RFC.

   En el área de configuración de ejecución, utilice las listas desplegables disponibles o introduzca valores para los parámetros necesarios. Para configurar los parámetros de ejecución opcionales, abra el área de configuración adicional.
4. Cuando haya terminado, haga clic en Ejecutar. Si no hay errores, aparecerá la página de la RFC creada correctamente con los detalles de la RFC enviada y el resultado inicial de la ejecución.
5. Abra el área de parámetros de ejecución para ver las configuraciones que envió. Actualice la página para actualizar el estado de ejecución de la RFC. Si lo desea, cancele la RFC o cree una copia de la misma con las opciones de la parte superior de la página.

Detener una tarea de AWS DMS replicación con la CLI

Cómo funciona:

1. Utilice la función de creación en línea (se emite un `create-rfc` comando con todos los parámetros de ejecución y RFC incluidos) o la de plantilla (se crean dos archivos JSON, uno

para los parámetros de RFC y otro para los parámetros de ejecución) y ejecute el `create-rfc` comando con los dos archivos como entrada. Ambos métodos se describen aquí.

2. Envíe el `aws amscm submit-rfc --rfc-id` *ID* comando RFC: con el ID de RFC devuelto.

   Supervise el comando RFC:. `aws amscm get-rfc --rfc-id` *ID*

Para comprobar la versión del tipo de cambio, utilice este comando:

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

> **Note**
>
> Puede utilizar cualquier `CreateRfc` parámetro con cualquier RFC, forme o no parte del esquema del tipo de cambio. Por ejemplo, para recibir notificaciones cuando cambie el estado de la RFC, añada esta línea `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` a la parte de los parámetros de la RFC de la solicitud (no a los parámetros de ejecución). Para ver una lista de todos los CreateRfc parámetros, consulta la referencia de la API de administración de cambios de AMS.

CREACIÓN EN LÍNEA:

Ejecute el comando create RFC con los parámetros de ejecución incluidos en línea (comillas de escape al proporcionar los parámetros de ejecución en línea) y, a continuación, envíe el ID de RFC devuelto. Por ejemplo, puedes reemplazar el contenido por algo como esto:

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

CREACIÓN DE PLANTILLA:

1. Envía los parámetros de ejecución de este tipo de cambio a un archivo JSON; este ejemplo lo nombra StopDmsRtParams .json:

```
aws amscm get-change-type-version --change-type-id "ct-1vd3y4ygbqmfk" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > StopDmsRtParams.json
```

2. Modifique y guarde el archivo JSON de los parámetros de ejecución. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
  "DocumentName": "AWSManagedServices-StopDmsTask",
  "Region": "us-east-1",
  "Parameters": {
      "ReplicationTaskArn": [
        "TASK_ARN"
      ]
  }
}
```

3. Envía la plantilla JSON a un archivo de tu carpeta actual; en este ejemplo se le asigna el nombre StopDmsRtRfc .json:

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

4. Modifica y guarda el StopDmsRtRfc archivo.json. Por ejemplo, puedes reemplazar el contenido por algo como esto:

```
{
  "ChangeTypeId": "ct-1vd3y4ygbqmfk",
  "ChangeTypeVersion": "1.0",
  "Title": "Stop DMS Replication Task"
}
```

5. Cree el RFC especificando el archivo de parámetros de ejecución y el StopDmsRtRfc archivo:

```
aws amscm create-rfc --cli-input-json file://StopDmsRtRfc.json --execution-
parameters file://StopDmsRtParams.json
```

Recibirá el ID de la nueva RFC en la respuesta y podrá utilizarla para enviar y supervisar la RFC. Hasta que la envíe, la RFC permanece en estado de edición y no se inicia.

Consejos

Puede detener una tarea de replicación del DMS mediante la consola AMS o la API/CLI del AMS. Para obtener más información, consulte Trabajar con tareas de AWS DMS.

Importación de bases de datos (DB) a AMS RDS para Microsoft SQL Server

> **Note**
>
> Los puntos de enlace AMS API/CLI (amscm y amsskms) se encuentran en la región de AWS del Norte de Virginia,. `us-east-1` En función de cómo esté configurada su autenticación y de la región de AWS en la que se encuentren su cuenta y sus recursos, es posible que tenga que `--region us-east-1` añadirlos al emitir comandos. Es posible que también necesite añadirlo`--profile saml`, si ese es su método de autenticación.

El proceso de importación de bases de datos a AMS RDS para SQL Server se basa en los tipos de cambio de AMS (CTs) enviados como solicitudes de cambio (RFCs) y utiliza los parámetros de la API de Amazon RDS como entrada. MicroSoft SQL Server es un sistema de administración de bases de datos relacionales (RDBMS). Para obtener más información, consulte también: Amazon Relational Database Service (Amazon RDS) y la referencia de las API de RDS o Amazon RDS.

> **Note**
>
> Asegúrese de que cada RFC se complete correctamente antes de continuar con el siguiente paso.

Pasos de importación de alto nivel:

1. Realice una copia de seguridad de la base de datos MS SQL local de origen en un archivo.bak (copia de seguridad)
2. Copie el archivo.bak en el depósito de tránsito (cifrado) de Amazon Simple Storage Service (S3)
3. Importe el archivo .bak a una nueva base de datos en la instancia MS SQL de Amazon RDS de destino

Requisitos:

- Pila RDS de MS SQL en AMS
- Pila de RDS con opción de restauración () `SQLSERVER_BACKUP_RESTORE`
- Cubeta Transit S3
- Función de IAM con acceso al bucket, lo que permite a Amazon RDS asumir la función
- Una EC2 instancia con MS SQL Management Studio instalado para administrar el RDS (puede ser una estación de trabajo local)

Configuración

Complete estas tareas para iniciar el proceso de importación.

1. Envíe una RFC para crear una pila de RDS mediante Deployment | Advanced stack components | RDS database stack | Create (ct-2z60dyvto9g6c). No utilice el nombre de la base de datos de destino (*`RDSDBName`*parámetro) en la solicitud de creación, ya que la base de datos de destino se creará durante la importación. Asegúrese de dejar suficiente espacio (`RDSAllocatedStorage`parámetro). Para obtener más información sobre cómo hacerlo, consulte la Guía de administración de cambios de AMS (RDS DB Stack | Create).
2. Envíe una RFC para crear el depósito S3 de tránsito (si aún no existe) mediante Deployment | Advanced stack components | S3 storage | Create (ct-1a68ck03fn98r). Para obtener más información sobre cómo hacerlo, consulte la Guía de administración de cambios de AMS S3 Storage | Create.
3. Envíe un RFC de administración | Otros | Otros | Actualice (ct-1e1xtak34nx76) para implementarlo con estos detalles: `customer_rds_s3_role`

   En la consola de :

   - Asunto: «Para admitir la importación de bases de datos de MS SQL Server, impleméntelo en esta cuenta. `customer_rds_s3_role`
   - Nombre del bucket de Transit S3:*`BUCKET_NAME`*.
   - Información de contacto:*`EMAIL`*.

   Con un ImportDbParams archivo.json para la CLI:

```
{
        "Comment": "{"Transit S3 bucket name":"BUCKET_NAME"}",
```

```
        "Priority": "High"
      }
}
```

4. Envíe un RFC de administración | Otros | Actualice el RFC solicitando a AMS que establezca la `SQLSERVER_BACKUP_RESTORE` opción en el RDS creado en el paso 1 (utilice el ID de pila del resultado del paso 1 y el rol de `customer_rds_s3_role` IAM en esta solicitud, en esta solicitud).
5. Envía una RFC para crear una EC2 instancia (puedes usar cualquier estación de trabajo EC2 o instancias existentes o locales) e instala Microsoft SQL Management Studio en la instancia.

Importación de la base de datos

Para importar la base de datos (DB), siga estos pasos.

1. Realice una copia de seguridad de la base de datos local de origen mediante el backup y la restauración nativos de MS SQL (consulte Support for native backup and restore in SQL Server). Como resultado de ejecutar esa operación, debería disponer de un archivo.bak (copia de seguridad).
2. Cargue el archivo.bak en un depósito de S3 de tránsito existente mediante la CLI de AWS S3 o la consola de AWS S3. Para obtener información sobre los buckets S3 de tránsito, consulte Protección de datos mediante cifrado.
3. Importe el archivo.bak a una base de datos nueva en su instancia de RDS para MS SQL Server de destino (para obtener más información sobre los tipos, consulte los tipos de instancias de Amazon RDS for MySQL):

    a. Inicie sesión en la EC2 instancia (estación de trabajo local) y abra MS SQL Management Studio

    b. Conéctese a la instancia de RDS de destino creada como requisito previo en el paso #1. Siga este procedimiento para conectarse: Conexión a una instancia de base de datos que ejecute el motor de base de datos Microsoft SQL Server

    c. Inicie el trabajo de importación (restauración) con una nueva consulta de lenguaje de consulta estructurado (SQL) (para obtener más información sobre las consultas SQL, consulte Introducción a SQL). El nombre de la base de datos de destino debe ser nuevo (no utilice el mismo nombre que la base de datos que creó anteriormente). Ejemplo sin cifrado:

```
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. Compruebe periódicamente el estado del trabajo de importación ejecutando esta consulta en una ventana independiente:

```
exec msdb.dbo.rds_task_status;
```

Si el estado cambia a Fallado, busque los detalles del error en el mensaje.

Limpieza

Una vez importada la base de datos, es posible que desee eliminar los recursos innecesarios. Siga estos pasos.

1. Elimine el archivo de respaldo (.bak) del bucket de S3. Para ello, puede utilizar la consola S3. Para ver el comando de CLI para eliminar un objeto de un bucket de S3, consulte rm en la Referencia de comandos de la CLI de AWS.
2. Elimine el bucket de S3 si no piensa usarlo. Para ver los pasos para hacerlo, consulta Eliminar pila.
3. Si no tiene previsto realizar importaciones a MS SQL, envíe un RFC de administración | Otros | Otros | Actualización (ct-0xdawir96cy7k) y solicite a AMS que elimine la función de IAM. `customer_rds_s3_role`

Implementaciones de aplicaciones Tier y Tie en AMS

En una implementación de nivel y enlace, se crean, configuran y despliegan los recursos de una pila de forma independiente y por separado RFCs, y se utilizan los componentes IDs de la pila a medida que se avanza para asociarlos entre sí.

Por ejemplo, para implementar un sitio web de alta disponibilidad (redundante) detrás de un balanceador de carga y una base de datos, utilizando un enfoque de nivel y enlace, envíe RFCs una base de datos y un balanceador de carga y dos EC2 instancias o un grupo de Auto Scaling, y configure las EC2 instancias o el grupo de Auto Scaling con el ID del ELB que creó.

Una vez desplegados los recursos, puede enviar un cambio de creación de grupo de seguridad para permitir que los recursos se comuniquen con la base de datos. Para obtener más información sobre la creación de grupos de seguridad, consulte Crear un grupo de seguridad.

Implementaciones completas de aplicaciones en AMS

Una implementación completa consiste en enviar un RFC con un CT que crea y configura todo lo que necesita a la vez. Por ejemplo, para implementar el sitio web de alta disponibilidad que se acaba de describir (EC2 instancias, balanceador de carga y base de datos), usaría un CT que, en conjunto, crearía y configuraría un grupo de Auto Scaling, un balanceador de carga, una base de datos y la configuración del grupo de seguridad necesaria para que todas las instancias funcionen como una pila. A continuación se describen algunos ejemplos de dos AMS CTs que hacen esto.

- Pila de dos niveles de alta disponibilidad (ct-06mjngx5flwto): este tipo de cambio le permite crear una pila y configurar un Grupo de Auto Scaling, una base de datos respaldada por RDS, un Load Balancer y una aplicación y configuración. CodeDeploy Tenga en cuenta que el balanceador de carga no se considera un nivel, ya que se comparte entre varias aplicaciones como un dispositivo de red y las funciones también se consideran un dispositivo. CodeDeploy Además, crea un grupo de CodeDeploy implementación (con el nombre que le dé a la CodeDeploy aplicación) que se puede usar para implementar sus aplicaciones. La configuración del grupo de seguridad para permitir que los recursos funcionen juntos se crea automáticamente.
- Pila de un nivel de alta disponibilidad (ct-09t6q7j9v5hrn): este tipo de cambio le permite crear una pila y configurar un Grupo de Auto Scaling y un Application Load Balancer. La configuración del grupo de seguridad que permite que los recursos funcionen juntos se crea automáticamente.

Trabajar con tipos de cambios de aprovisionamiento () CTs

AMS es responsable de su infraestructura gestionada. Para realizar cambios, debe enviar una RFC con la clasificación CT correcta (categoría, subcategoría, artículo y operación). En esta sección se describe cómo encontrar CTs, determinar si alguno es adecuado para sus necesidades y solicitar un nuevo CT si no lo es.

Compruebe si una tomografía computarizada existente cumple con sus requisitos

Una vez que haya determinado qué es lo que desea implementar con AMS, el siguiente paso es estudiar CTs las CloudFormation plantillas existentes para ver si ya existe una solución.

Al crear una RFC, debe especificar la CT. Puede utilizar la Consola de administración de AWS API/CLI de AMS. A continuación se describen ejemplos del uso de ambas.

Puede utilizar la consola o la API/CLI para buscar un identificador de tipo de cambio (CT) o una versión. Existen dos métodos: buscar o seleccionar la clasificación. Para ambos tipos de selección, puede ordenar la búsqueda eligiendo Usado con más frecuencia, Usado más recientemente o Alfabético.

YouTube Vídeo: ¿Cómo creo un RFC mediante la CLI de AWS Managed Services y dónde puedo encontrar el esquema CT?

En la consola AMS, en la página RFCs-> Crear RFC:

- Con la opción Buscar por tipo de cambio seleccionada (opción predeterminada), puede hacer lo siguiente:
  - Usa el área de creación rápida para seleccionar una de las más populares de AMS CTs. Haga clic en una etiqueta y se abrirá la página Ejecutar RFC con la opción Asunto rellenada automáticamente. Complete las opciones restantes según sea necesario y haga clic en Ejecutar para enviar la RFC.
  - O bien, desplácese hacia abajo hasta el área Todos los tipos de cambios y comience a escribir un nombre de CT en el cuadro de opciones; no es necesario que tenga el nombre exacto o completo del tipo de cambio. También puede buscar una tomografía computarizada por ID de tipo de cambio, clasificación o modo de ejecución (automática o manual) introduciendo las palabras correspondientes.

    Con la vista de tarjetas predeterminada seleccionada, las tarjetas CT coincidentes aparecen a medida que escribes, seleccionas una carta y haces clic en Crear RFC. Con la vista de tabla seleccionada, elige el CT correspondiente y haz clic en Crear RFC. Ambos métodos abren la página Ejecutar RFC.
- Como alternativa, y para explorar las opciones de tipos de cambio, haga clic en Elegir por categoría en la parte superior de la página para abrir una serie de cuadros de opciones desplegables.
- Elija una categoría, una subcategoría, un elemento y una operación. Aparece el cuadro de información para ese tipo de cambio y aparece un panel en la parte inferior de la página.
- Cuando esté listo, presione Entrar y aparecerá una lista de los tipos de cambios coincidentes.
- Elige un tipo de cambio de la lista. El cuadro de información para ese tipo de cambio aparece en la parte inferior de la página.

- Una vez que tenga el tipo de cambio correcto, elija Crear RFC.

> **Note**
>
> La CLI de AMS debe estar instalada para que estos comandos funcionen. Para instalar la API o CLI de AMS, vaya a la página de recursos para desarrolladores de la consola AMS. Para obtener material de referencia sobre la API AMS CM o la API AMS SKMS, consulte la sección Recursos de información de AMS en la Guía del usuario. Puede que necesite añadir una `--profile` opción de autenticación; por ejemplo,`aws amsskms` *`ams-cli-command`* `--profile SAML`. Es posible que también tengas que añadir la `--region` opción, ya que todos los comandos de AMS se ejecutan desde us-east-1, por ejemplo. `aws amscm` *`ams-cli-command`* `--region=us-east-1`

> **Note**
>
> Los puntos de enlace AMS API/CLI (amscm y amsskms) se encuentran en la región de AWS del Norte de Virginia,. `us-east-1` En función de cómo esté configurada su autenticación y de la región de AWS en la que se encuentren su cuenta y sus recursos, es posible que tenga que `--region us-east-1` añadirlos al emitir comandos. Es posible que también necesite añadirlo`--profile saml`, si ese es su método de autenticación.

Para buscar un tipo de cambio mediante la API CM de AMS (consulte ListChangeTypeClassificationSummaries) o la CLI:

Puede utilizar un filtro o una consulta para realizar la búsqueda. La ListChangeTypeClassificationSummaries operación tiene opciones de filtros para `CategorySubcategory`,`Item`, y`Operation`, pero los valores deben coincidir exactamente con los valores existentes. Para obtener resultados más flexibles al usar la CLI, puede usar la `--query` opción.

Cambie el tipo de filtrado con la API/CLI AMS CM

Atributo	Valores válidos	Condición válida/predeterminada	Notas
ChangeTypeId	Cualquier cadena que represente a ChangeTypeId (por ejemplo: ct-abc123 xyz7890)	Igual a	Para ver el tipo de cambio, consulte la Referencia del tipo de cambio. IDs Para ver el tipo de cambio IDs, consulte Búsqueda de un tipo de cambio o CSIO.
Categoría	Cualquier texto de formato libre	Contiene	No se admiten las expresiones regulares en cada campo individual. Búsqueda sin distinción de mayúsculas y minúsculas
Subcategory			
Elemento			
Operation			

1. Estos son algunos ejemplos de clasificaciones de tipos de cambio de listas:

   El siguiente comando muestra todas las categorías de tipos de cambio.

   ```
   aws amscm list-change-type-categories
   ```

   El siguiente comando muestra las subcategorías que pertenecen a una categoría específica.

   ```
   aws amscm list-change-type-subcategories --category CATEGORY
   ```

   El siguiente comando muestra los elementos que pertenecen a una categoría y subcategoría especificadas.

   ```
   aws amscm list-change-type-items --category CATEGORY --subcategory SUBCATEGORY
   ```

2. Estos son algunos ejemplos de búsqueda de tipos de cambios con consultas CLI:

   El siguiente comando busca en los resúmenes de las clasificaciones de CT las que contienen «S3" en el nombre del elemento y crea el resultado de la categoría, la subcategoría, el elemento, la operación y el identificador del tipo de cambio en forma de tabla.

```
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. A continuación, puede utilizar el identificador del tipo de cambio para obtener el esquema CT y examinar los parámetros. El siguiente comando envía el esquema a un archivo JSON denominado Creates3Params.schema.json.

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateS3Params.schema.json
```

   Para obtener información sobre el uso de consultas CLI, consulte Cómo filtrar la salida con la opción --query y la referencia del lenguaje de consulta, JMESPath Especificación.

4. Una vez que tengas el ID del tipo de cambio, te recomendamos que compruebes la versión del tipo de cambio para asegurarte de que es la última. Usa este comando para buscar la versión de un tipo de cambio específico:

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

   Para `AutomationStatus` buscar el tipo de cambio específico, ejecute este comando:

```
aws amscm --profile saml get-change-type-version --change-type-id CHANGE_TYPE_ID --
query "ChangeTypeVersion.{AutomationStatus:AutomationStatus.Name}"
```

Para `ExpectedExecutionDurationInMinutes` buscar el tipo de cambio específico, ejecuta este comando:

```
aws amscm --profile saml get-change-type-version --change-type-id ct-14027q0sjyt1h
 --query "ChangeTypeVersion.{ExpectedDuration:ExpectedExecutionDurationInMinutes}"
```

Una vez que hayas encontrado el CT que consideres apropiado, observa los parámetros de ejecución (esquema JSON) asociado al mismo para saber si se adapta a tu caso de uso.

Utilice este comando para generar un esquema CT en un archivo JSON que lleve el nombre del CT; este ejemplo genera el esquema de almacenamiento Create S3:

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
--query "ChangeTypeVersion.ExecutionInputSchema" --output text >
CreateBucketParams.json
```

Veamos más de cerca lo que ofrece este esquema.

Esquema de creación de cubos de S3

```
{
  "$schema": "http://json-schema.org/draft-04/sch
ema#",
"name": "Create S3 Storage
"description": "Use to create an Amazon Simple
 Storage Service stack.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "The description of the
 stack.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to create the S3
 Bucket in, in the form vpc-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{17}$"
```

El esquema comienza con la CT («descripción»), que indica para qué sirve el esquema. En este caso, para crear una pila de almacenamiento de S3.

A continuación, tiene propiedades obligatorias y opcionales que puede especificar. Se proporcio nan los valores de propiedad predeterminados. Las propiedades obligatorias se muestran al final del esquema.

En el StackTemplateId área, verá que hay una plantilla de pila específica para este CT

```
    },
    "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",
            "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.",
```

y este esquema, y su identific ador es un valor de propiedad obligatorio.

El esquema le permite etiquetar la pila que está creando para fines de contabili dad interna. Además, algunas opciones, como la copia de seguridad, requieren una etiqueta de Key:Backup y Value:True. Para obtener información detallada, consulta Cómo etiquetar tus EC2 recursos de Amazon.

```
      "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,
          "maxLength": 63
        }
      },
      "additionalProperties": false,
      "required": [
        "AccessControl",
        "BucketName"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
```

En la sección de parámetros del esquema CT JSON se proporcionan los parámetros de ejecución.

Para este esquema, solo se requieren la ACL y BucketName los parámetros de ejecución.

```
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Solicite un nuevo CT

Tras examinar el esquema, puede decidir que no proporciona parámetros suficientes para crear la implementación que desea. Si ese es el caso, examine las CloudFormation plantillas existentes para encontrar una que se acerque más a lo que desea. Una vez que sepa qué parámetros adicionales necesita, envíe un documento de administración | Otros | Otros | Crear CT.

> **Note**
>
> Todos los demás | Other Create and Update CTs reciben la atención de un operador de AMS, quien se pondrá en contacto con usted para hablar sobre el nuevo CT.

Para enviar una solicitud de un nuevo CT, acceda a la consola AMS de forma habitual Consola de administración de AWSy, a continuación, siga estos pasos.

1. En el panel de navegación de la izquierda, haz clic en RFCs.

   Se abre la página RFCs del panel de control.
2. Haga clic en Crear.

   Se abre la página Crear una solicitud de cambio.
3. Seleccione Administración en la lista desplegable de categorías y Otros para la subcategoría y el artículo. Para la operación, elija Crear. La RFC necesitará aprobación antes de poder implementarse.
4. Introduzca la información que explique por qué desea el CT, por ejemplo: solicite un CT de almacenamiento Create S3 modificado que permita personalizarlo ACLs, en función del CT de almacenamiento Create S3 existente. Esto debería dar como resultado un nuevo CT: Deployment | Advanced Stack Components | S3 storage | Create S3 custom ACL. Este nuevo CT podría ser público.
5. Haga clic en Submit.

Su RFC aparece en el panel de control de RFC.

Pruebe la nueva tomografía computarizada

Una vez que AWS Managed Services haya creado ese nuevo CT, debe probarlo enviando un RFC con él. Si ha colaborado con AMS para aprobar previamente el nuevo CT, solo tiene que seguir una solicitud de RFC estándar y observar el resultado (para obtener más información sobre el envío RFCs, consulte Creación y envío de una RFC). Si el nuevo CT no está aprobado previamente (quiere asegurarse de que nunca se ejecute sin una aprobación explícita), tendrá que hablar con AMS sobre su implementación cada vez que desee publicarlo.

Arranques rápidos

Temas

Al utilizar una combinación de tipos de cambios de AMS, puede realizar tareas complejas.

Puedes usar el sistema de gestión de cambios AMS para configurar AMS Resource Scheduler, para una zona de aterrizaje multicuenta (MALZ) o para una cuenta de zona de aterrizaje con una sola cuenta (SALZ). El proceso varía. Además, para realizar transferencias de archivos e instantáneas entre cuentas.

Inicio rápido de AMS Resource Scheduler

Utilice esta guía de inicio rápido para implementar AMS Resource Scheduler, un programador de instancias basado en etiquetas para ahorrar costes en AMS Advanced.

El programador de recursos de AMS se basa en el programador de instancias de AWS.

Terminología del programador de recursos de AMS

Antes de empezar, es bueno familiarizarse con la terminología del programador de recursos de AMS:

- período: cada programa debe contener al menos un período que defina las horas en las que debe ejecutarse la instancia. Una programación puede contener más de un período. Cuando se utiliza más de un período en un programa, el programador de recursos aplica la acción de inicio adecuada cuando se cumple al menos una de las reglas del período.
- zona horaria: para obtener una lista de los valores de zona horaria aceptables que se pueden utilizar en el DefaultTimezoneparámetro al que se hace referencia más adelante, consulte la columna TZ de la lista de zonas horarias de la base de datos TZ.
- hibernar: cuando se establece en verdadero, las EC2 instancias que están habilitadas para la hibernación y cumplen con los requisitos de hibernación se hibernan (). suspend-to-disk Comprueba en la EC2 consola si tus instancias están habilitadas para la hibernación. Utilice la hibernación para las EC2 instancias de Amazon detenidas que ejecutan Amazon Linux.

- obligatorio: si se establece en true, según la programación definida, el programador de recursos detiene un recurso en ejecución si se inicia manualmente fuera del período de ejecución e inicia un recurso si se detiene manualmente durante el período de ejecución.
- retain_running: si se establece en true, impide que el programador de recursos detenga una instancia al final de un período de ejecución si la instancia se inició manualmente antes del comienzo del período. Por ejemplo, si una instancia con un período configurado que va de las 9 a. m. a las 5 p. m., se inicia manualmente antes de las 9 a. m., el programador de recursos no detiene la instancia a las 5 p. m.
- ssm-maintenance-window: añada una ventana de AWS Systems Manager mantenimiento como período de ejecución a una programación. Cuando especificas el nombre de un período de mantenimiento que existe en la misma cuenta y región de AWS que tu pila implementada para programar tus EC2 instancias de Amazon, el Programador de recursos iniciará la instancia antes del inicio del período de mantenimiento y la detendrá al final del período de mantenimiento, si ningún otro período de ejecución especifica que la instancia debe ejecutarse y si el evento de mantenimiento se ha completado.

  El Programador de recursos utiliza la AWS Lambda frecuencia que especificó durante la configuración inicial para determinar cuánto falta para iniciar la instancia hasta que finalice el período de mantenimiento. Si estableces el AWS CloudFormation parámetro Frecuencia en 10 minutos o menos, el Programador de recursos inicia la instancia 10 minutos antes del período de mantenimiento. Si estableces la frecuencia en más de 10 minutos, el programador de recursos inicia la instancia el mismo número de minutos que la frecuencia que especificaste. Por ejemplo, si establece la frecuencia del período de mantenimiento de Systems Manager en 30 minutos, Resource Schedulers inicia la instancia 30 minutos antes del período de mantenimiento.

  Para obtener más información, consulte Ventanas de AWS Systems Manager mantenimiento.
- estado de anulación: anula temporalmente las acciones de inicio y parada programadas configuradas por el programador de recursos. Si configura el campo en ejecución, el programador de recursos inicia, pero no detiene, la instancia correspondiente. La instancia se ejecuta hasta que la detengas manualmente. Si estableces el estado de anulación en Detenido, el Programador de recursos detiene pero no inicia la instancia correspondiente. La instancia no se ejecuta hasta que la inicies manualmente.

Implementación del programador de recursos de AMS

Para implementar una solución de programador de recursos de AMS, siga estos pasos.

1. Envíe un RFC de implementación | Programador de recursos de AMS | Solución | Implementación (ct-0ywnhc8e5k9z5) e indique los siguientes parámetros:

   - SchedulingActive: Sí para habilitar la programación de recursos, no para deshabilitarla. Está predeterminada en Sí.
   - ScheduledServices: Introduzca una lista de servicios separados por comas para los que programar los recursos. Los valores válidos incluyen una combinación de escalado automático, ec2 y rds. El valor predeterminado es autoscaling, ec2, rds.
   - TagName: el nombre de la clave de etiqueta que asocia los esquemas de programación de recursos con los recursos de servicio. El valor predeterminado es Schedule.

     > **Note**
     >
     > La implementación del programador de recursos solo funcionará con los recursos que tengan esta etiqueta.

   - DefaultTimezone: el nombre de la zona horaria, con el formato US/Pacific, que se utilizará como zona horaria predeterminada. El valor predeterminado es UTC.
2. Tras recibir la confirmación de que la RFC del primer paso se ha ejecutado correctamente, puede enviar el tipo de cambio Período | Añadir.
3. Por último, envíe una RFC para añadir un cronograma al período que se creó en el paso dos. Utilice el tipo de cambio Programar | Agregar.

Implementación y uso del programador de recursos AMS FAQs

Preguntas frecuentes sobre el AMS Resource Scheduler.

P: ¿Qué ocurre si habilito la hibernación pero la EC2 instancia no la admite?

R: La hibernación guarda el contenido de la memoria de la instancia (RAM) en el volumen raíz de Amazon Elastic Block Store (Amazon EBS). Si este campo está establecido en true, las instancias pasan a hibernación cuando Resource Scheduler las detiene.

Si configuras el Programador de recursos para que utilice la hibernación, pero tus instancias no están habilitadas para la hibernación o no cumplen los requisitos previos de hibernación, el Programador de recursos registra una advertencia y las instancias se detienen sin hibernación. Para obtener más información, consulta Hibernar la instancia.

P: ¿Qué ocurre si configuro override_status y forced?

R: Si estableces override_status en ejecución y forzadas en true (evita que una instancia se inicie manualmente fuera de un período de ejecución), Resource Scheduler detiene la instancia.

Si estableces override_status como detenido y forzado como true (evita que una instancia se detenga manualmente durante un período de ejecución), el Programador de recursos reinicia la instancia.

> **Note**
>
> Si forced es falsa, se aplica el comportamiento de anulación configurado.

P: Una vez implementado el programador de recursos de AMS, ¿cómo puedo inhabilitar o habilitar el programador de recursos en mi cuenta?

R: Para activar o desactivar el programador de recursos de AMS:

- Para deshabilitar: cree un RFC mediante State | Disable. Asegúrese de configurar la opción SchedulerStateDESHABILITAR
- Para habilitarlo: cree un RFC mediante State | Enable. Asegúrese de configurar la opción SchedulerStateHABILITAR

P: ¿Qué ocurre si el período del programador de recursos de AMS se encuentra dentro de mi período de mantenimiento de los parches?

R: El programador de recursos funciona en función de sus programas configurados. Si está configurado para detener una instancia mientras se está aplicando el parche, detiene la instancia a menos que se añada la ventana de aplicación de parches como un período a la programación antes de que comience la aplicación de parches. En otras palabras, Resource Scheduler no inicia automáticamente ninguna instancia detenida para aplicar parches a menos que se configure un período designado. Para evitar conflictos con el período de mantenimiento de los parches, añada el intervalo de tiempo asignado a la aplicación de parches al programa del Programador de recursos como un período. Para añadir un período a la programación existente, cree una RFC mediante Período | Añadir.

P: Si necesito tener un horario diferente para distintas EC2 instancias, ¿puedo configurar más de un horario en mi cuenta?

R: Sí, puedes crear varios horarios. Cada programa puede tener varios períodos según el requisito. Cuando el programador de recursos de AMS está activado en la cuenta, se configura una clave de etiqueta. Por ejemplo, si la clave de la etiqueta es «Programación», el valor de la etiqueta puede variar en función de los distintos programas, lo que corresponde al nombre del programa del programador de recursos de AMS. Para añadir una nueva programación, puede crear una RFC utilizando el tipo de cambio Management | AMS Resource Scheduler | Schedule | Add (ct-2bxelbn765ive), consulte Programación | Añadir.

P: ¿Dónde puedo encontrar los distintos tipos de cambios compatibles con el Programador de recursos de AMS?

R: AMS dispone de varios tipos de cambios en el Programador de recursos para implementarlo en su cuenta; activarlo o deshabilitarlo; definir, añadir, actualizar y eliminar programas y períodos para usarlos con él; y describir (obtener una descripción detallada de) los programas y períodos.

Configuración de copias de seguridad entre cuentas (dentro de la región)

AWS Backup admite la posibilidad de copiar instantáneas de una cuenta a otra dentro de la misma región de AWS siempre que las dos cuentas estén dentro de la misma organización de AWS. Por ejemplo, en la zona de aterrizaje multicuenta (MALZ) de AMS Advanced, puede configurar una copia de instantáneas entre cuentas dentro de la misma región de AWS mediante este inicio rápido.

Para obtener más información, consulte AWS Backup y AWS Organizations incorporan la función de respaldo multicuenta

Las instantáneas se copian entre cuentas para la recuperación ante desastres (DR). Para proteger los datos, es posible que tenga requisitos para conservar las instantáneas en la misma región de AWS, pero fuera de los límites de la cuenta.

Información general:

En términos generales, estos son los pasos para realizar copias de seguridad entre cuentas en AMS:

- Cree una cuenta de destino para alojar las copias de seguridad en la región de AWS en la que está alojada su landing zone de AMS (paso 1)
- Cree una clave KMS para cifrar las copias de seguridad en la cuenta de destino (paso 3)

- Cree una bóveda de respaldo en la cuenta de destino de la misma región que su zona de aterrizaje de AMS Advanced (paso 4)
- Habilite la configuración de cuentas cruzadas en su cuenta de administración (paso 5)
- Cree o modifique el plan y las reglas de respaldo de la cuenta de origen (paso 6)

> **Note**
>
> Asegúrese de que las cuentas de origen y destino estén en la misma región. Si desea copiar las copias de seguridad de una región a otra, póngase en contacto con su CA o CSDM.

Para habilitar y configurar las copias de seguridad entre cuentas:

1. Cree una cuenta de destino para alojar las copias de seguridad; si ya tiene una cuenta de este tipo, puede omitir este paso. Para crear la cuenta, envíe un RFC desde su cuenta de Management Payer mediante el tipo de cambio Implementación | Managed landing zone | Cuenta de administración | Crear cuenta de aplicación (con VPC) (ct-1zdasmc2ewzrs).
2. [Opcional] Si los recursos o las instantáneas están cifrados en la cuenta de origen (por ejemplo, Prod), comparta la clave KMS utilizada para el cifrado con la cuenta de destino. Para ello, envíe una RFC con la opción Administración | Componentes de pila avanzados | Clave KMS | Tipo de cambio de actualización (ct-3ovo7px2vsa6n).
3. En la cuenta de destino, cree una clave KMS para utilizarla en el cifrado de Backup Vault. Para ello, envíe una RFC mediante la opción Implementación | Componentes de pila avanzados | Clave KMS | Crear tipo de cambio (auto) (ct-1d84keiri1jhg).
4. En la cuenta de destino, cree un Backup Vault con la clave creada anteriormente. Los almacenes de AWS Backup se pueden crear mediante el tipo de cambio automático de ingesta CFN, Deployment | Ingestion | Stack from CloudFormation Template | Create (ct-36cn2avfrrj9v). En la misma solicitud, es necesario modificar la política de acceso al almacén para permitir que las cuentas de origen accedan al almacén. A continuación, se muestra un ejemplo de política:

   Ejemplo CloudFormation de plantilla para 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. Desde su cuenta de Management Payer, active la copia de seguridad entre cuentas. Para ello, envíe un RFC mediante el tipo de cambio Management | AWS Backup | Backup plan | Enable cross-account copy (cuenta de administración) (ct-2yja7ihh30ply).
6. Por último, desde la cuenta de origen en la que se originan las copias de seguridad, cree la regla o reglas del plan de copias de seguridad que rigen las copias de seguridad para copiar instantáneas entre cuentas. Para ello, envíe una RFC mediante el tipo de cambio Deployment | AWS Backup | Backup plan | Create (ct-2hyozbpa0sx0m). Si necesita actualizar un plan de respaldo existente, envíe una RFC utilizando el tipo de cambio Administración | Otros | Otros | Actualización (ct-0xdawir96cy7k) con la siguiente información:

    1. El nombre del plan de respaldo y el nombre de la regla que se va a actualizar.
    2. El ARN de la bóveda de respaldo de la destination/ICE cuenta.
    3. La retención para la days/months que desea conservar las instantáneas en la bóveda del ICE objetivo.

Tutoriales

Temas

Los siguientes tutoriales detallan los pasos para crear una pila de dos niveles con High Availability (ct-06mjngx5flwto), usar la CLI y usar la consola e implementar un grupo Amazon Auto Scaling (ASG) de Linux o RHEL. EC2 A cada uno tier-and-tie le sigue un tutorial similar (uno para la consola y otro para la CLI), que utiliza recursos separados CTs y creados en un orden tal que permiten unir los recursos a medida que se crean.

Las descripciones de todas las opciones de CT, incluidas, ChangeTypeId se encuentran en la referencia managedservices/latest/ctref /Change Type.

Tutorial de consola: pila de dos niveles de alta disponibilidad (Linux/RHEL)

En esta sección se describe cómo implementar un WordPress sitio de alta disponibilidad (HA) en un entorno AMS mediante la consola AMS.

> **Note**
>
> Este tutorial de implementación se ha probado en entornos AMZN Linux y RHEL.

Resumen de las tareas y requisitos: RFCs

1. Cree una infraestructura (pila de dos niveles de alta disponibilidad)
2. Cree un bucket de S3 para aplicaciones CodeDeploy
3. Cree el paquete de WordPress aplicaciones y cárguelo en el bucket de S3
4. Implemente la aplicación con CodeDeploy
5. Acceda al WordPress sitio e inicie sesión para validar la implementación

6. Destruya la implementación

Las descripciones de todas las opciones de tomografía computarizada ChangeTypeId, incluidas las disponibles, se encuentran en la referencia de cambios de tipo de AMS.

Antes de empezar

Deployment | Advanced Stack Components | High Availability Two Tier Stack | Create CT crea un grupo de Auto Scaling, un balanceador de cargas, una base de datos y un nombre de CodeDeploy aplicación y un grupo de implementación (con el mismo nombre que le dé a la aplicación). Para obtener más información, CodeDeploy consulte ¿Qué es? CodeDeploy

En este tutorial se utiliza una RFC de pila de dos niveles de alta disponibilidad que incluye `UserData` y también describe cómo crear un WordPress paquete que CodeDeploy se pueda implementar.

Lo que `UserData` se muestra en el ejemplo obtiene los metadatos de la instancia, como el ID de la instancia, la región, etc., de una instancia en ejecución consultando el servicio de metadatos de la EC2 instancia disponible en http://169.254.169.254/latest/meta-data/. Esta línea del script de datos de usuario:`REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed 's/[a-z]$//')`, recupera el nombre de la zona de disponibilidad del servicio de metadatos y lo coloca en la variable $REGION de las regiones admitidas y lo usa para completar la URL del depósito de S3 donde se descarga el agente. CodeDeploy La IP 169.254.169.254 solo se puede enrutar dentro de la VPC (todos pueden consultar el servicio). VPCs Para obtener información sobre el servicio, consulte Metadatos de instancia y datos de usuario. Tenga en cuenta también que los scripts introducidos como UserData se ejecutan como usuario «root» y no necesitan usar el comando «sudo».

Este tutorial deja los siguientes parámetros en el valor predeterminado (que se muestra):

- Grupo 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 datos:`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`.
- Solicitud:`DeploymentConfigName=CodeDeployDefault.OneAtATime`.

Parámetros variables:

La consola proporciona una opción ASAP para la hora de inicio y en este tutorial se recomienda usarla. ASAP hace que la RFC se ejecute tan pronto como se aprueben las aprobaciones.

> **Note**
>
> Hay muchos parámetros que puede elegir configurar de forma diferente a la que se muestra. Los valores de los parámetros que se muestran en el ejemplo se han probado, pero es posible que no sean adecuados para usted. En los ejemplos solo se muestran los valores obligatorios. Los valores de la *`replaceable`* fuente deben cambiarse, ya que son específicos de su cuenta.

Cree la infraestructura

Este procedimiento utiliza el CT de pila de dos niveles de alta disponibilidad seguido del CT de almacenamiento Create S3.

Si recopila los siguientes datos antes de empezar, la implementación será más rápida.

LOS DATOS NECESARIOS TIENEN UNA PILA:

- AutoScalingGroup:
  - UserData: Este valor se proporciona en este tutorial. Incluye comandos para configurar el recurso CodeDeploy e iniciar el CodeDeploy agente.

  - AMI-ID: este valor determina el sistema operativo de las EC2 instancias que activará su grupo de Auto Scaling (ASG). Seleccione una AMI en su cuenta que comience por «customer-» y que sea del sistema operativo que desee. Busque AMI IDs en la consola AMS VPCs -> página de VPCs detalles. Este tutorial es para ASGs configurar el uso de una AMI de Amazon Linux o RHEL.
- Base de datos:
  - Estos parámetros, DBEngineEngineVersion, y LicenseModeldeben configurarse de acuerdo con su situación, aunque se han probado los valores que se muestran en el ejemplo. El tutorial usa estos valores, respectivamente:*MySQL*,*8.0.16*,*general-public-license*.
  - Estos parámetros, DBNameMasterUserPassword, y MasterUsernameson necesarios al implementar el paquete de aplicaciones. El tutorial usa estos valores, respectivamente:*wordpressDB*,*p4ssw0rd*,*admin*. Tenga en cuenta que solo DBName puede contener caracteres alfanuméricos.
  - Cuando introduzca la MasterUsernamepara la base de datos de RDS, aparecerá en texto sin cifrar, así que inicie sesión en la base de datos lo antes posible y cambie la contraseña para garantizar su seguridad.
  - Para los RDSSubnetID, utilice dos subredes privadas. Introdúzcalas una por una, pulsando «Entrar» después de cada una. Busque la subred IDs con la referencia de la API SKMS de AMS, consulte la pestaña Informes de la consola AWS Artifact Console. Funcionamiento (CLI list-subnet-summaries:) o en la página de detalles de la consola AMS VPCs -> VPC.
- LoadBalancer:
  - Establezca este parámetro, Public, en true, porque el tutorial utiliza subredes ELB públicas.
  - ELBSubnetIdentificadores: utilice dos subredes públicas. Introdúzcalas una por una, pulsando «Entrar» después de cada una. Busque la subred IDs con la referencia de la API SKMS de AMS, consulte la pestaña Informes de la consola AWS Artifact Console. Funcionamiento (CLI list-subnet-summaries:) o en la página de detalles de la consola AMS VPCs -> VPC.
- Aplicación: el ApplicationNamevalor establece el nombre de la CodeDeploy aplicación y el nombre del grupo de implementación. CodeDeploy Se usa para implementar la aplicación. Debe ser único en la cuenta. Para comprobar los CodeDeploy nombres de su cuenta, consulte la CodeDeploy consola. El ejemplo usa *WordPress* pero, si va a usar ese valor, asegúrese de que aún no esté en uso.

1. Lance la pila de alta disponibilidad.

a. En la página Crear RFC, seleccione de la lista la categoría Despliegue, la subcategoría Standard Stacks, el elemento Pila de dos niveles de alta disponibilidad y la operación Create.

b. IMPORTANTE: Seleccione Avanzado y defina los valores tal y como se muestra.

   Solo tiene que introducir valores para las opciones marcadas con un asterisco (*); en el ejemplo se muestran los valores comprobados; puede dejar en blanco las opciones vacías que no sean obligatorias.

c. Para la sección de descripción de la RFC:

```
Subject: WP-HA-2-Tier-RFC
```

d. En la sección de información sobre recursos, defina los parámetros para la base de datos AutoScalingGroupLoadBalancer, la aplicación y las etiquetas.

   Además, el objetivo de la clave de etiqueta AppName «» es poder buscar fácilmente las instancias de ASG en la EC2 consola; puede llamar a esta clave de etiqueta «Nombre» o cualquier otro nombre de clave que desee. Tenga en cuenta que puede añadir hasta 50 etiquetas.

```
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. Haz clic en Enviar cuando hayas terminado.

2. Inicie sesión en la base de datos que creó y cambie la contraseña.
3. Lance un bucket Stack de S3.

   Si recopila los siguientes datos antes de empezar, la implementación será más rápida.

   DEPÓSITO S3 DE DATOS NECESARIO:

   - VPC-ID: este valor determina dónde estará su bucket de S3. Busque la VPC IDs con la referencia para obtener información sobre la API SKMS de AMS, consulte la pestaña Informes de la consola AWS Artifact Console. Operation (CLI: list-vpc-summaries) o en la página de la consola AMS. VPCs
   - BucketName: Este valor establece el nombre del bucket de S3, que se utiliza para cargar el paquete de aplicaciones. Debe ser único en toda la región de la cuenta y no puede incluir letras mayúsculas. No BucketName es obligatorio incluir tu ID de cuenta como parte de él, pero te permitirá identificar el segmento más adelante con más facilidad. Para ver qué nombres de bucket de S3 existen en la cuenta, vaya a la consola de Amazon S3 de su cuenta.

   a. En la página Crear RFC, seleccione la categoría Implementación, la subcategoría Componentes de pila avanzados, el elemento S3 Storage y la operación Create en la lista de selección de RFC CT.

   b. Mantenga la opción básica predeterminada y establezca los valores como se muestra.

```
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. Haga clic en Enviar cuando haya terminado. El depósito implementado con este tipo de cambio permite el read/write acceso total a toda la cuenta.

Crear, cargar e implementar la aplicación

En primer lugar, cree un paquete de WordPress aplicaciones y, a continuación, utilice el CodeDeploy CTs para crear e implementar la aplicación.

1. Descargue WordPress, extraiga los archivos y cree un. directorio /scripts.

   Comando de Linux:

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows: pégalo `https://github.com/WordPress/WordPress/archive/master.zip` en una ventana del navegador y descarga el archivo zip.

   Cree un directorio temporal en el que ensamblar el paquete.

   Linux:

   ```
   mkdir /tmp/WordPress
   ```

   Windows: cree un directorio WordPress «», utilizará la ruta del directorio más adelante.

2. Extraiga la WordPress fuente al directorio WordPress «» y cree un. directorio /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: vaya al directorio «WordPress» que creó y cree allí un directorio de «scripts».

Si se encuentra en un entorno Windows, asegúrese de establecer el tipo de interrupción de los archivos de script en Unix (LF). En Notepad ++, esta opción se encuentra en la parte inferior derecha de la ventana.

3. Cree el archivo CodeDeploy appspec.yml en el WordPress directorio (si va a copiar el ejemplo, compruebe la hendidura, cada espacio cuenta). IMPORTANTE: Asegúrese de que la ruta «fuente» sea correcta para copiar los WordPress archivos (en este caso, en su WordPress directorio) al destino esperado (/). var/www/html/WordPress En el ejemplo, el archivo appspec.yml está en el directorio con los WordPress archivos, por lo que solo se necesita «/». Además, aunque haya utilizado una AMI de RHEL para su grupo de Auto Scaling, deje la línea «os: linux» tal como está. Ejemplo de archivo 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. Cree scripts de archivos bash en. WordPress directorio /scripts.

En primer lugar, cree `config_wordpress.sh` con el siguiente contenido (si lo prefiere, puede editar el archivo wp-config.php directamente).

> **Note**
>
> *DBName*Sustitúyalo por el valor indicado en el RFC de HA Stack (por ejemplo,`wordpress`).
> *DB_MasterUsername*Sustitúyalo por el `MasterUsername` valor indicado en el RFC de HA Stack (por ejemplo,`admin`).
> *DB_MasterUserPassword*Sustitúyalo por el `MasterUserPassword` valor indicado en el RFC de HA Stack (por ejemplo,`p4ssw0rd`).
> *DB_ENDPOINT*Sustitúyalo por el nombre DNS del punto final en los resultados de ejecución del RFC de la pila HA (por ejemplo,`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Puede encontrarlo en la GetRfcoperación (CLI: get-rfc --rfc-id RFC_ID) o en la página de detalles del RFC de la consola AMS para el RFC de la pila de alta disponibilidad que envió anteriormente.

```
#!/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. En `install_dependencies.sh` el mismo directorio, cree con el siguiente contenido:

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS se instala como parte de los datos del usuario en el momento del lanzamiento para permitir que las comprobaciones de estado funcionen desde el principio.

6. En el mismo directorio, cree `start_server.sh` con el siguiente contenido:

   - Para las instancias de Amazon Linux, usa lo siguiente:

   ```
   #!/bin/bash
   service httpd start
   ```

   - Para las instancias de RHEL, usa esto (los comandos adicionales son políticas que permiten que SELINUX las acepte): 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. En el mismo directorio, cree `stop_server.sh` con el siguiente contenido:

   ```
   #!/bin/bash
   service httpd stop
   ```

8. Cree el paquete zip.

   Linux:

   ```
   $ cd /tmp/WordPress
   $ zip -r wordpress.zip .
   ```

   Windows: Ve al directorio WordPress «», selecciona todos los archivos y crea un archivo zip, asegúrate de llamarlo wordpress.zip.

1. Cargue el paquete de aplicaciones en el bucket de S3

El paquete debe estar en su lugar para poder seguir desplegando la pila.

Tendrá acceso automáticamente a cualquier instancia de bucket de S3 que cree. Puede acceder a él a través de sus Bastions (consulte Acceso a las instancias) o a través de la consola S3 y cargar el CodeDeploy paquete con él drag-and-drop, o bien navegando hasta el archivo y seleccionándolo.

También puede usar el siguiente comando en una ventana de shell; asegúrese de tener la ruta correcta al archivo zip:

```
aws s3 cp wordpress/wordpress.zip s3://BUCKET_NAME/
```

2. Implemente el paquete WordPress CodeDeploy de aplicaciones

   DESPLIEGUE DE CÓDIGO DE DATOS REQUERIDO: DESPLIEGUE DE APLICACIONES:

   - CodeDeployApplicationName: el nombre que le diste a la CodeDeploy aplicación.
   - CodeDeployGroupName: Dado que tanto la CodeDeploy aplicación como el grupo se crearon a partir del nombre que le diste a la CodeDeploy aplicación en el RFC de la pila HA, es el mismo nombre que el CodeDeployApplicationName.
   - S3Bucket: el nombre que le diste al depósito S3.
   - S3 BundleType y S3Key: forman parte del paquete de WordPress aplicaciones que implementó.
   - VpcId: El VPC correspondiente.

   a. En la página Crear RFC, seleccione la categoría Despliegue, la subcategoría Aplicaciones, el elemento CodeDeploy Aplicación y operación Despliegue en la lista de selección de RFC CT.

   b. Mantenga la opción básica predeterminada y establezca los valores como se muestra.

   > **Note**
   >
   > Haga referencia a la CodeDeploy aplicación, el grupo de CodeDeploy implementación, el depósito de S3 y el paquete creados anteriormente.

```
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. Haga clic en Enviar cuando haya terminado.

Validar el despliegue de la aplicación

Navegue hasta el punto final (LoadBalancerCName) del balanceador de cargas creado anteriormente, con la ruta implementada:/. WordPress WordPress Por ejemplo:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Deberías ver una página como esta:

Elimine la implementación de alta disponibilidad

Para reducir la implementación, debe enviar el Delete Stack CT, comparándolo con el conjunto de alta disponibilidad de dos niveles y el bucket S3, y puede solicitar que se eliminen las instantáneas de RDS (se eliminan automáticamente al cabo de diez días, pero mientras están ahí cuestan un pequeño importe). Reúna la pila IDs de alta disponibilidad y el depósito de S3 y, a continuación, siga estos pasos. Consulte Stack | Delete.

Tutorial de consola: Implementación de un WordPress sitio web Tier and Tie

En esta sección se describe cómo implementar un WordPress sitio de alta disponibilidad (HA) en un entorno AMS mediante la consola AMS. Este conjunto de instrucciones incluye un ejemplo de cómo crear el archivo WordPress CodeDeploy de paquete compatible necesario (por ejemplo, zip). El aprovisionamiento de los recursos sigue un orden que permite unirlos para formar «niveles».

> **Note**
>
> Este tutorial de implementación está diseñado para usarse con un sistema operativo Linux AMZN.
> Los parámetros de las variables esenciales están anotados como*`replaceable`*; sin embargo, es posible que desee modificar otros parámetros para adaptarlos a su situación.

Resumen de las tareas y requisitos: RFCs

1. Cree la infraestructura:
   a. Crear un clúster de base de datos MySQL RDS
   b. Cree un equilibrador de carga
   c. Cree un grupo de escalado automático y vincúlelo al balanceador de cargas
   d. Cree un bucket de S3 para las aplicaciones CodeDeploy
2. Cree un paquete de WordPress aplicaciones (no requiere un RFC)
3. Implemente el paquete WordPress de aplicaciones con CodeDeploy:
   a. Cree una CodeDeploy aplicación
   b. Cree un grupo CodeDeploy de despliegue
   c. Cargue el paquete de WordPress aplicaciones en el bucket de S3 (no requiere un RFC)
   d. Implemente la aplicación CodeDeploy
4. Valide el despliegue
5. Destruya el despliegue

Las descripciones de todas las opciones de tomografía computarizada, incluidas las disponibles, ChangeTypeId se encuentran en la referencia de cambios de tipo de AMS.

Creación de un RFC mediante la consola (conceptos básicos)

Estos son algunos pasos que debe seguir cada vez que cree una RFC con la consola.

1. Haga clic RFCsen el panel de navegación izquierdo para abrir la página de la RFCs lista y, a continuación, haga clic en Crear RFC.

   Se abre la página Crear RFC.
2. Seleccione Buscar tipos de cambios (opción predeterminada) o Elegir por categoría.
3. Examine los tipos de cambios:

   a. Haga clic en una opción de creación rápida para iniciar una RFC con uno de los tipos de cambios más utilizados.

      Se abre el área de configuración general para ese tipo de cambio y se completa la línea de asunto. Para ver los detalles del tipo de cambio, abra el área en la parte superior de la página.

   b. Utilice el área Todos los tipos de cambios.

      Filtra, cambia entre una vista de cartas o de tabla u ordena los tipos de cambios. Cuando encuentres el que buscas, selecciónalo y haz clic en Crear RFC en la parte superior de la página.

      Se abre el área de configuración general para ese tipo de cambio y se completa la línea de asunto. Para ver los detalles del tipo de cambio, abra el área en la parte superior de la página.
4. Elija por categoría:

   a. Seleccione la categoría, la subcategoría, el artículo y la operación correspondientes.

      El cuadro de detalles del tipo de cambio aparece en la parte inferior de la página.

   b. Haga clic en Crear RFC en la parte inferior de la página.

   c. Se abre el área de configuración general para ese tipo de cambio y se rellena la línea de asunto. Para ver los detalles del tipo de cambio, abra el área en la parte superior de la página.
5. Para asegurarse de que determinadas personas reciban notificaciones sobre el progreso de la RFC, rellene las direcciones de correo electrónico. Para añadir detalles sobre el tipo de cambio,

rellena la descripción. Abra el área de configuración adicional para agregar más detalles sobre la RFC.

6. Para programar, seleccione Ejecutar este cambio lo antes posible o Programar este cambio. Si selecciona Ejecutar este cambio lo antes posible, su RFC se ejecutará en cuanto se aprueben las aprobaciones. Si selecciona Programar este tipo de cambio, aparecerá un calendario de selección, hora y zona horaria y su RFC comenzará, tras su envío, según lo programado.
7. En el área de configuración de ejecución, configure los parámetros del tipo de cambio. Para ver los parámetros opcionales, abra el área de configuración adicional.
8. Cuando esté listo, haga clic en Ejecutar.

Creación de la infraestructura

Inicie sesión en la consola de AWS de la cuenta de AMS de destino y, a continuación, en la consola de AMS de la cuenta.

Los siguientes procedimientos describen la creación de una base de datos de RDS, un balanceador de carga y un grupo de Auto Scaling de tal manera que se utilice el recurso IDs para construir la infraestructura.

Crear una pila de RDS

Consulte Pila de RDS | Crear.

Cree una pila ELB

Lanza un ELB público.

DATOS NECESARIOS:

- `VpcId`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.
- `ELBSubnetIds`: Una matriz de subredes a través de las cuales el balanceador de cargas distribuirá el tráfico. Elija subredes públicas o privadas. Busque la subred IDs con la referencia de la API SKMS de AMS, consulte la pestaña Informes de la consola AWS Artifact Console. Funcionamiento (CLI list-subnet-summaries:) o en la página de detalles de la consola AMS VPCs -> VPC.
- `VpcId`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.

1. En la página Crear RFC, seleccione la categoría Despliegue, la subcategoría Componentes de pila avanzados, pila de balanceadores de carga (ELB) de elementos y haga clic en Crear. Elija Avanzado y acepte todos los valores predeterminados (incluidos los que no tienen ningún valor) excepto los que se muestran a continuación.

```
Subject:                        WP-ELB-RFC
ELBSubnetIds:                   PUBLIC_AZ1
                              PUBLIC_AZ2
ELBScheme                       true
ELBCookieExpirationPeriod       600
VpcId:                          VPC_ID
Name:                           WP-Public-ELB
```

2. Haga clic en Enviar cuando haya terminado.

Crear una pila de grupos de Auto Scaling

Lance un grupo de escalado automático.

DATOS NECESARIOS:

- `VpcId`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.
- `AMI-ID`: Este valor determina qué tipo de EC2 instancias activará su grupo de Auto Scaling (ASG). Asegúrese de seleccionar una AMI en su cuenta que comience por «customer-» y que sea del sistema operativo que desee. Busque la AMI IDs con la referencia de la API SKMS de AMS, consulte la pestaña Informes de la consola AWS Artifact Console. Operation (CLI: list-amis) o en la consola AMS -> página de detalles. VPCs VPCs Este tutorial es para ASGs configurar el uso de una AMI de Linux.
- `ASGLoadBalancerNames`: El balanceador de carga que creó anteriormente. Busque el nombre consultando EC2 Consola -> Equilibradores de carga (en el menú de navegación de la izquierda). Ten en cuenta que este no es el «nombre» que especificaste cuando creaste el ELB anteriormente.

1. En la página Crear RFC, seleccione la categoría Despliegue, la subcategoría Componentes de pila avanzados, el elemento Grupo de escalado automático y haga clic en Crear. Elija Avanzado y acepte todos los valores predeterminados (incluidos los que no tienen ningún valor) excepto los que se muestran a continuación.

> **Note**
>
> Especifique la AMI de AMS más reciente. Especifique el ELB creado anteriormente.

```
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. Haga clic en Enviar cuando haya terminado.

Crear una pila S3

Lanza un depósito de S3. El depósito de S3 es el lugar donde se carga el paquete de aplicaciones que ha creado.

DATOS NECESARIOS:

- `VPC-ID`: Este valor determina dónde estará su bucket de S3, que debe ser el mismo que el de la VPC utilizada anteriormente.

- `AccessControl`: Las opciones de la AccessControl lista preestablecida (ACL) son`Private`, y. `PublicRead` Para obtener más información, consulte Amazon Simple Storage Service Canned ACL.
- `BucketName`: Este valor establece el nombre del bucket de S3, que se utiliza para cargar el paquete de aplicaciones. Debe ser único en la región de la cuenta y no puede incluir letras mayúsculas. No BucketName es obligatorio incluir tu ID de cuenta como parte de él, pero te permitirá identificar el segmento más adelante con más facilidad. Para ver qué nombres de bucket de S3 existen en la cuenta, vaya a la consola de Amazon S3 de su cuenta.

1. En la página Crear RFC, seleccione la categoría Implementación, la subcategoría Advanced Stack Components, el elemento S3 Storage y haga clic en Crear.

   Puede dejar la opción de parámetro predeterminada en Básico para aceptar los valores predeterminados tal como se describe. Para establecer valores diferentes, seleccione Avanzado.

   > **Note**
   >
   > El segmento implementado con este tipo de cambio permite el read/write acceso total a toda la cuenta, por lo que es posible que se necesiten nuevos tipos de cambios para permitir permisos de acceso más restringidos.

   ```
   Subject:            S3-Bucket-RFC
   BucketName:         ACCOUNT_ID-codedeploy-bundles
   AccessControl:      Private

   VpcId:              VPC_ID
   Name:               S3BucketForWP
   ```

2. Haz clic en Enviar cuando hayas terminado.

Crea un WordPress CodeDeploy paquete

En la sección se proporciona un ejemplo de cómo crear un paquete de despliegue de aplicaciones.

1. Descargue WordPress, extraiga los archivos y cree un. directorio /scripts.

   Comando de Linux:

```
wget https://github.com/WordPress/WordPress/archive/master.zip
```

Windows: pégalo `https://github.com/WordPress/WordPress/archive/master.zip` en una ventana del navegador y descarga el archivo zip.

Cree un directorio temporal en el que ensamblar el paquete.

Linux:

```
mkdir /tmp/WordPress
```

Windows: cree un directorio WordPress «», utilizará la ruta del directorio más adelante.

2. Extraiga la WordPress fuente al directorio WordPress «» y cree un. directorio /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: vaya al directorio «WordPress» que creó y cree allí un directorio de «scripts».

   Si se encuentra en un entorno Windows, asegúrese de establecer el tipo de interrupción de los archivos de script en Unix (LF). En Notepad ++, esta opción se encuentra en la parte inferior derecha de la ventana.

3. Cree el archivo CodeDeploy appspec.yml en el WordPress directorio (si va a copiar el ejemplo, compruebe la hendidura, cada espacio cuenta). IMPORTANTE: Asegúrese de que la ruta «fuente» sea correcta para copiar los WordPress archivos (en este caso, en su WordPress directorio) al destino esperado (/). var/www/html/WordPress En el ejemplo, el archivo appspec.yml está en el directorio con los WordPress archivos, por lo que solo se necesita «/». Además, aunque haya utilizado una AMI de RHEL para su grupo de Auto Scaling, deje la línea «os: linux» tal como está. Ejemplo de archivo 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. Cree scripts de archivos bash en. WordPress directorio /scripts.

   En primer lugar, cree `config_wordpress.sh` con el siguiente contenido (si lo prefiere, puede editar el archivo wp-config.php directamente).

   > **Note**
   >
   > *DBName*Sustitúyalo por el valor indicado en el RFC de HA Stack (por ejemplo,`wordpress`).
   > *DB_MasterUsername*Sustitúyalo por el `MasterUsername` valor indicado en el RFC de HA Stack (por ejemplo,`admin`).
   > *DB_MasterUserPassword*Sustitúyalo por el `MasterUserPassword` valor indicado en el RFC de HA Stack (por ejemplo,`p4ssw0rd`).
   > *DB_ENDPOINT*Sustitúyalo por el nombre DNS del punto final en los resultados de ejecución del RFC de la pila HA (por ejemplo,`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Puede encontrarlo en la GetRfcoperación (CLI: get-rfc --rfc-id RFC_ID) o en la página de detalles del RFC de la consola AMS para el RFC de la pila de alta disponibilidad que envió anteriormente.

```
#!/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. En `install_dependencies.sh` el mismo directorio, cree con el siguiente contenido:

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS se instala como parte de los datos del usuario en el momento del lanzamiento para permitir que las comprobaciones de estado funcionen desde el principio.

6. En el mismo directorio, cree `start_server.sh` con el siguiente contenido:

   - Para las instancias de Amazon Linux, usa lo siguiente:

   ```
   #!/bin/bash
   service httpd start
   ```

   - Para las instancias de RHEL, usa esto (los comandos adicionales son políticas que permiten que SELINUX las acepte): 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. En el mismo directorio, cree `stop_server.sh` con el siguiente contenido:

```
#!/bin/bash
service httpd stop
```

8. Cree el paquete zip.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows: Ve a tu directorio WordPress «», selecciona todos los archivos y crea un archivo zip, asegúrate de llamarlo wordpress.zip.

Implemente el paquete de WordPress aplicaciones con CodeDeploy

CodeDeploy Es un servicio de implementación de AWS que automatiza las implementaciones de aplicaciones en las instancias de Amazon EC2 . Esta parte del proceso implica la creación de una CodeDeploy aplicación, la creación de un grupo de CodeDeploy implementación y, a continuación, la implementación de la aplicación mediante. CodeDeploy

Crear una CodeDeploy aplicación

La CodeDeploy aplicación es simplemente un nombre o contenedor que AWS utiliza CodeDeploy para garantizar que se haga referencia a la revisión, la configuración de implementación y el grupo de implementación correctos durante una implementación. La configuración de despliegue, en este caso, es el WordPress paquete que creó anteriormente.

DATOS NECESARIOS:

- `VpcId`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.
- `CodeDeployApplicationName`: Debe ser única en la cuenta. Consulte la CodeDeploy consola para comprobar los nombres de las aplicaciones existentes.

1. Cree la CodeDeploy aplicación para WordPress

En la página Crear RFC, seleccione la categoría Despliegue, la subcategoría Aplicaciones y el elemento CodeDeploy Aplicación y operación Crear en la lista de selección de RFC CT. Elija Básico y defina los valores como se muestra. Haga clic en Enviar cuando haya terminado.

```
Subject:                        CD-WP-App-RFC
CodeDeployApplicationName:      WordPress
VpcId:                          VPC_ID
Name:                           WP-CD-App
```

2. Haga clic en Enviar cuando haya terminado.

Crear un grupo CodeDeploy de implementación

Cree el grupo CodeDeploy de despliegue.

Un grupo de CodeDeploy implementación define un conjunto de instancias individuales destinadas a una implementación.

DATOS NECESARIOS:

- `VpcId`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.
- `CodeDeployApplicationName`: Utilice el valor que creó anteriormente.
- `CodeDeployAutoScalingGroups`: Use el nombre del grupo de Auto Scaling que creó anteriormente.
- `CodeDeployDeploymentGroupName`: un nombre para el grupo de implementación. Este nombre debe ser único para cada aplicación asociada al grupo de implementaciones.
- `CodeDeployServiceRoleArn`: Utilice la fórmula que se proporciona en el ejemplo.

1. En la página Crear RFC, seleccione la categoría Despliegue, la subcategoría Aplicaciones, el elemento, el grupo de CodeDeploy despliegue y la operación Crear en la lista de selección de RFC CT. Elija Avanzado y defina los valores como se muestra (solo se necesita un asunto para la RFC). Haga clic en Enviar cuando haya terminado.

> **Note**
>
> Haga referencia al CodeDeploy ARN de la función de servicio en este formato `"arn:aws:iam::085398962942:role/aws-codedeploy-role"` y utilice el nombre del grupo de escalado automático creado anteriormente para «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. Haga clic en Enviar cuando haya terminado.

Cargue la WordPress aplicación

Tendrá acceso automáticamente a cualquier instancia de bucket de S3 que cree. Puede acceder a él a través de sus Bastions (consulte Acceso a las instancias) o a través de la consola S3 y cargar el CodeDeploy paquete. El paquete debe estar en su lugar para poder seguir desplegando la pila. En el ejemplo se usa el nombre del bucket creado anteriormente.

Puedes usar este comando de AWS para comprimir el paquete:

```
aws s3 cp wordpress/wordpress.zip s3://ACCOUNT_ID-codedeploy-bundles/
```

Implemente la WordPress aplicación con CodeDeploy

Implemente la CodeDeploy aplicación.

DATOS NECESARIOS:

- `VPC-ID`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.
- `CodeDeployApplicationName`: Use el nombre de la CodeDeploy aplicación que creó anteriormente.

- `CodeDeployDeploymentGroupName`: utilice el nombre del grupo de CodeDeploy despliegue que creó anteriormente.
- `S3Location`(donde cargó el paquete de aplicaciones)`S3Bucket`:: El BucketName que creó anteriormente `S3BundleType` y`S3Key`: el tipo y el nombre del paquete que colocó en su tienda de S3.

1. Implemente el paquete WordPress CodeDeploy de aplicaciones

   En la página Crear RFC, seleccione la categoría Implementación, la subcategoría Aplicaciones, el elemento CodeDeploy Aplicación y operación Implementación en la lista de selección de RFC CT. Elija Básico y defina los valores como se muestra. Haga clic en Enviar cuando haya terminado.

   > **Note**
   >
   > Haga referencia a la CodeDeploy aplicación, el grupo de CodeDeploy implementación, el bucket de S3 y el paquete creados anteriormente.

   ```
   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. Haga clic en Enviar cuando haya terminado.

Validar el despliegue de la aplicación

Navegue hasta el punto final (ELB CName) del balanceador de cargas creado anteriormente, con la ruta implementada:/. WordPress WordPress Por ejemplo:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Elimine la implementación de aplicaciones

Para desglosar la implementación, debe enviar el Delete Stack CT a la pila de bases de datos de RDS, el balanceador de carga de aplicaciones, el grupo Auto Scaling, el bucket S3 y la aplicación y el grupo Code Deploy (seis en total). RFCs Además, puede enviar una solicitud de servicio para eliminar las instantáneas de RDS (se eliminan automáticamente al cabo de diez días, pero su permanencia en ellas cuesta un pequeño importe). Reúna la pila IDs para todas y, a continuación, siga estos pasos. Consulte Apilar | Eliminar.

Tutorial de CLI: pila de dos niveles de alta disponibilidad (Linux/RHEL)

En esta sección se describe cómo implementar una pila de dos niveles de alta disponibilidad (HA) en un entorno AMS mediante la CLI de AMS.

> **Note**
>
> Este tutorial de implementación se ha probado en entornos AMZN Linux y RHEL.

Resumen de las tareas y requisitos: RFCs

1. Cree una infraestructura (pila de dos niveles de alta disponibilidad)
2. Cree un bucket de S3 para aplicaciones CodeDeploy
3. Cree el paquete de WordPress aplicaciones y cárguelo en el bucket de S3
4. Implemente la aplicación con CodeDeploy
5. Acceda al WordPress sitio e inicie sesión para validar la implementación

Antes de empezar

Deployment | Advanced Stack Components | High Availability Two Tier Stack Advanced | Create CT crea un grupo de Auto Scaling, un balanceador de cargas, una base de datos y un nombre de CodeDeploy aplicación y un grupo de implementación (con el mismo nombre que le dé a la aplicación). Para obtener más información, CodeDeploy consulte ¿Qué es? CodeDeploy

Este tutorial utiliza una RFC de dos niveles (avanzada) de alta disponibilidad que incluye UserData y también describe cómo crear un WordPress paquete que CodeDeploy se pueda implementar.

Lo que `UserData` se muestra en el ejemplo obtiene los metadatos de la instancia, como el ID de la instancia, la región, etc., de una instancia en ejecución consultando el servicio de metadatos de la EC2 instancia disponible en http://169.254.169.254/latest/meta-data/. Esta línea del script de datos de usuario:`REGION=$(curl 169.254.169.254/latest/meta-data/placement/availability-zone/ | sed 's/[a-z]$//')`, recupera el nombre de la zona de disponibilidad del servicio de metadatos y lo coloca en la variable $REGION de las regiones admitidas y lo usa para completar la URL del depósito de S3 donde se descarga el agente. CodeDeploy La IP 169.254.169.254 solo se puede enrutar dentro de la VPC (todos pueden consultar el servicio). VPCs Para obtener información sobre el servicio, consulte Metadatos de instancia y datos de usuario. Tenga en cuenta también que los scripts introducidos como UserData se ejecutan como usuario «root» y no necesitan usar el comando «sudo».

Este tutorial deja los siguientes parámetros en el valor predeterminado (que se muestra):

- Grupo 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 datos:`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`.
- Solicitud:`DeploymentConfigName=CodeDeployDefault.OneAtATime`.
- Cubeta S3:`AccessControl=Private`.

AJUSTES ADICIONALES:

`RequestedStartTime`y `RequestedEndTime` si quieres programar tu RFC: puedes usar Time.is para determinar la hora UTC correcta. Los ejemplos proporcionados deben ajustarse adecuadamente. Una RFC no puede continuar si ha pasado la hora de inicio. Como alternativa, puede omitir esos valores para crear un RFC ASAP que se ejecute en cuanto se aprueben las aprobaciones.

> **Note**
>
> Hay muchos parámetros que puede elegir configurar de forma diferente a la que se muestra. Los valores de los parámetros que se muestran en el ejemplo se han probado, pero es posible que no sean adecuados para usted.

Cree la infraestructura

Si recopila los siguientes datos antes de empezar, la implementación será más rápida.

LOS DATOS NECESARIOS TIENEN UNA PILA:

- AutoScalingGroup:
  - `UserData`: Este valor se proporciona en este tutorial. Incluye comandos para configurar el recurso CodeDeploy e iniciar el CodeDeploy agente.
  - `AMI-ID`: Este valor determina qué tipo de EC2 instancias activará su grupo de Auto Scaling (ASG). Asegúrese de seleccionar una AMI en su cuenta que comience por «customer-» y que sea del sistema operativo que desee. Busque la AMI IDs con la referencia de la API SKMS de AMS, consulte la pestaña Informes de la consola AWS Artifact Console. Operation (CLI: list-amis) o en la consola AMS -> página de detalles. VPCs VPCs Este tutorial es para ASGs configurar el uso de una AMI de Linux.
- Base de datos:
  - Estos parámetros, `DBEngineEngineVersion`, y `LicenseModel` deben configurarse de acuerdo con su situación, aunque se han probado los valores que se muestran en el ejemplo.
  - Estos parámetros, `RDSSubnetIds` `DBNameMasterUsername`, y `MasterUserPassword` son necesarios para implementar el paquete de aplicaciones. Para RDSSubnet los identificadores, utilice dos subredes privadas.
- LoadBalancer:
  - Estos parámetros, `DBEngineEngineVersion`, y `LicenseModel` deben configurarse de acuerdo con su situación, aunque se han probado los valores que se muestran en el ejemplo.

  - `ELBSubnetIds`: Utilice dos subredes públicas.
- Aplicación: el `ApplicationName` valor establece el nombre de la CodeDeploy aplicación y el nombre del grupo de CodeDeploy implementación. Se usa para implementar la aplicación. Debe ser único en la cuenta. Para comprobar los CodeDeploy nombres de su cuenta, consulte la CodeDeploy consola. En el ejemplo se usa WordPress "" pero, si va a utilizar ese valor, asegúrese de que no esté ya en uso.

Este procedimiento utiliza el CT de pila de dos niveles (avanzado) de alta disponibilidad (ct-06mjngx5flwto) y el CT de almacenamiento Create S3 (ct-1a68ck03fn98r). Desde su cuenta autenticada, siga estos pasos en la línea de comandos.

1. Lanza la pila de infraestructuras.

   a. Envía los parámetros de ejecución del esquema JSON de la pila de dos niveles de alta disponibilidad (CT) a un archivo de tu carpeta actual denominado CreateStackParams .json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-06mjngx5flwto"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateStackParams.json
   ```

   b. Modifique el esquema. *`variables`*Sustitúyala según proceda. Por ejemplo, utilice el sistema operativo que desee para las EC2 instancias que cree el ASG. Guárdelo`ApplicationName`, ya que lo usará más adelante para implementar la aplicación. Tenga en cuenta que puede añadir hasta 50 etiquetas.

   ```
   {
   "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. Envía la plantilla CreateRfc JSON a un archivo de tu carpeta actual denominado CreateStackRfc .json:

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

d. Modifique la plantilla RFC de la siguiente manera y guárdela; puede eliminar y reemplazar el contenido. Tenga en cuenta que `RequestedStartTime` ahora `RequestedEndTime` son opcionales; al excluirlos, se crea un RFC lo antes posible que se ejecuta tan pronto como se aprueba (lo que normalmente ocurre automáticamente). Para enviar una RFC programada, añada esos valores.

```
{
"ChangeTypeVersion":    "3.0",
```

```
"ChangeTypeId":         "ct-06mjngx5flwto",
"Title":                "HA-Stack-For-WP-RFC"
}
```

e. Cree el RFC especificando el CreateStackRfc archivo.json y el archivo de parámetros de ejecución CreateStackParams .json:

```
aws amscm create-rfc --cli-input-json file://CreateStackRfc.json --execution-
parameters file://CreateStackParams.json
```

Recibirás el ID de RFC en la respuesta. Guarde el ID para los pasos siguientes.

f. Envíe el RFC:

```
aws amscm submit-rfc --rfc-id  RFC_ID
```

Si la RFC es correcta, no recibirá ningún resultado.

g. Para comprobar el estado del RFC, ejecute

```
aws amscm get-rfc --rfc-id RFC_ID
```

Anote el ID de RFC.

2. Lance un bucket de S3

Si recopila los siguientes datos antes de empezar, la implementación será más rápida.

DEPÓSITO S3 DE DATOS NECESARIO:

- `VPC-ID`: Este valor determina dónde estará su bucket de S3. Use el mismo ID de VPC que utilizó anteriormente.
- `BucketName`: Este valor establece el nombre del bucket de S3, que se utiliza para cargar el paquete de aplicaciones. Debe ser único en toda la región de la cuenta y no puede incluir letras mayúsculas. No BucketName es obligatorio incluir tu ID de cuenta como parte de él, pero te permitirá identificar el segmento más adelante con más facilidad. Para ver qué nombres de bucket de S3 existen en la cuenta, vaya a la consola de Amazon S3 de su cuenta.

a. Genere los parámetros de ejecución del esquema JSON para el CT de creación del almacenamiento de S3 en un archivo JSON denominado CreateS3 StoreParams .json.

```
aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateS3StoreParams.json
```

b. Modifique el esquema de la siguiente manera, puede eliminar y reemplazar el contenido. Reemplace *VPC_ID* adecuadamente. Los valores del ejemplo se han probado, pero es posible que no sean adecuados para usted.

> **Tip**
>
> `BucketName`Deben ser únicos en la región de la cuenta y no pueden incluir letras mayúsculas. No BucketName es obligatorio incluir tu ID de cuenta como parte de él, pero te permitirá identificar el segmento más adelante con más facilidad. Para ver qué nombres de bucket de S3 existen en la cuenta, vaya a la consola de Amazon S3 de su cuenta.

```
{
"Description":      "S3BucketForWordPressBundle",
"VpcId":            "VPC_ID",
"StackTemplateId":  "stm-s2b72beb000000000",
"Name":             "S3BucketForWP",
"TimeoutInMinutes":  60,
"Parameters":   {
    "AccessControl":    "Private",
    "BucketName":       "ACCOUNT_ID-BUCKET_NAME"
    }
}
```

c. Envía la plantilla JSON CreateRfc a un archivo, en tu carpeta actual, denominado CreateS3 StoreRfc .json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateS3StoreRfc.json
```

d. Modifique y guarde el archivo CreateS3 StoreRfc .json. Puede eliminar y reemplazar el contenido. Tenga en cuenta que `RequestedStartTime` ahora `RequestedEndTime` son opcionales; al excluirlos, se crea un RFC ASAP que se ejecuta tan pronto como se aprueba (lo que normalmente ocurre automáticamente). Para enviar una RFC programada, añada esos valores.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1a68ck03fn98r",
"Title":                "S3-Stack-For-WP-RFC"
}
```

e. Cree el RFC especificando el archivo CreateS3 StoreRfc .json y el archivo de parámetros de ejecución StoreParams CreateS3 .json:

```
aws amscm create-rfc --cli-input-json file://CreateS3StoreRfc.json  --
execution-parameters file://CreateS3StoreParams.json
```

Recibirá el nuevo RFC en RfcId la respuesta. Guarde el ID para los pasos siguientes.

f. Envíe el RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Si la RFC es correcta, no recibirá ningún resultado.

g. Para comprobar el estado del RFC, ejecute

```
aws amscm get-rfc --rfc-id RFC_ID
```

Crear, cargar e implementar la aplicación

En primer lugar, cree un paquete de WordPress aplicaciones y, a continuación, utilice el CodeDeploy CTs para crear e implementar la aplicación.

1. Descargue WordPress, extraiga los archivos y cree un. directorio /scripts.

   Comando de Linux:

   ```
   wget https://github.com/WordPress/WordPress/archive/master.zip
   ```

   Windows: pégalo `https://github.com/WordPress/WordPress/archive/master.zip` en una ventana del navegador y descarga el archivo zip.

   Cree un directorio temporal en el que ensamblar el paquete.

Linux:

```
mkdir /tmp/WordPress
```

Windows: cree un directorio WordPress «», utilizará la ruta del directorio más adelante.

2. Extraiga la WordPress fuente al directorio WordPress «» y cree un. directorio /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: vaya al directorio «WordPress» que creó y cree allí un directorio de «scripts».

   Si se encuentra en un entorno Windows, asegúrese de establecer el tipo de interrupción de los archivos de script en Unix (LF). En Notepad ++, esta opción se encuentra en la parte inferior derecha de la ventana.

3. Cree el archivo CodeDeploy appspec.yml en el WordPress directorio (si va a copiar el ejemplo, compruebe la hendidura, cada espacio cuenta). IMPORTANTE: Asegúrese de que la ruta «fuente» sea correcta para copiar los WordPress archivos (en este caso, en su WordPress directorio) al destino esperado (/). var/www/html/WordPress En el ejemplo, el archivo appspec.yml está en el directorio con los WordPress archivos, por lo que solo se necesita «/». Además, aunque haya utilizado una AMI de RHEL para su grupo de Auto Scaling, deje la línea «os: linux» tal como está. Ejemplo de archivo 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. Cree scripts de archivos bash en. WordPress directorio /scripts.

   En primer lugar, cree `config_wordpress.sh` con el siguiente contenido (si lo prefiere, puede editar el archivo wp-config.php directamente).

> **Note**
>
> *DBName*Sustitúyalo por el valor indicado en el RFC de HA Stack (por ejemplo,`wordpress`).
> *DB_MasterUsername*Sustitúyalo por el `MasterUsername` valor indicado en el RFC de HA Stack (por ejemplo,`admin`).
> *DB_MasterUserPassword*Sustitúyalo por el `MasterUserPassword` valor indicado en el RFC de HA Stack (por ejemplo,`p4ssw0rd`).
> *DB_ENDPOINT*Sustitúyalo por el nombre DNS del punto final en los resultados de ejecución del RFC de la pila HA (por ejemplo,`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Puede encontrarlo en la GetRfcoperación (CLI: get-rfc --rfc-id RFC_ID) o en la página de detalles del RFC de la consola AMS para el RFC de la pila de alta disponibilidad que envió anteriormente.

```
#!/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. En `install_dependencies.sh` el mismo directorio, cree con el siguiente contenido:

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS se instala como parte de los datos del usuario en el momento del lanzamiento para permitir que las comprobaciones de estado funcionen desde el principio.

6. En el mismo directorio, cree `start_server.sh` con el siguiente contenido:

   - Para las instancias de Amazon Linux, usa lo siguiente:

   ```
   #!/bin/bash
   service httpd start
   ```

   - Para las instancias de RHEL, usa esto (los comandos adicionales son políticas que permiten que SELINUX las acepte): 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. En el mismo directorio, cree `stop_server.sh` con el siguiente contenido:

```
#!/bin/bash
service httpd stop
```

8. Cree el paquete zip.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

Windows: Ve a tu directorio WordPress «», selecciona todos los archivos y crea un archivo zip, asegúrate de llamarlo wordpress.zip.

1. Cargue el paquete de aplicaciones en el bucket de S3.

   El paquete debe estar en su lugar para poder seguir desplegando la pila.

   Tendrá acceso automáticamente a cualquier instancia de bucket de S3 que cree. Puedes acceder a él a través de tus bastiones o a través de la consola S3 y cargar el WordPress paquete con el archivo zip drag-and-drop o buscarlo y seleccionarlo.

   También puedes usar el siguiente comando en una ventana de shell; asegúrate de tener la ruta correcta al archivo zip:

   ```
   aws s3 cp wordpress.zip s3://BUCKET_NAME/
   ```

2. Implemente el paquete de WordPress aplicaciones.

   Si recopila los siguientes datos antes de empezar, la implementación será más rápida.

   DATOS NECESARIOS:

   - `VPC-ID`: Este valor determina dónde estará su bucket de S3. Use el mismo ID de VPC que utilizó anteriormente.
   - `CodeDeployApplicationNameyCodeDeployApplicationName`: El `ApplicationName` valor que utilizó en el RFC de pila de 2 niveles de HA estableció el y el CodeDeployApplicationName . CodeDeployDeploymentGroupName En el ejemplo se utiliza «WordPress», pero es posible que haya utilizado un valor diferente.
   - `S3Location`: Para`S3Bucket`, usa el `BucketName` que creaste anteriormente. Los `S3BundleType` y `S3Key` son del paquete que pusiste en tu tienda S3.

   a. Genera el esquema JSON de los parámetros de ejecución para la CodeDeploy aplicación Deploy Params.json en un archivo JSON denominado Deploy CDApp Params.json.

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

b. Modifique el esquema de la siguiente manera y guárdelo como, puede eliminar y reemplazar el contenido.

```
{
"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. Envía la plantilla JSON CreateRfc a un archivo, en tu carpeta actual, denominado Deploy CDApp RFC.json:

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

d. Modifica y guarda el archivo Deploy CDApp RFC.json. Puedes eliminar y reemplazar el contenido. Tenga en cuenta que `RequestedStartTime` ahora `RequestedEndTime` son opcionales; al excluirlos, se crea un RFC ASAP que se ejecuta tan pronto como se aprueba (lo que normalmente ocurre automáticamente). Para enviar una RFC programada, añada esos valores.

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-WP-RFC"
```

```
}
```

e. Cree la RFC especificando el archivo CDApp Rfc de despliegue y el archivo de parámetros de ejecución de Deploy CDApp Params:

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

Recibirá el RfcId nuevo RFC en la respuesta. Guarde el ID para los pasos siguientes.

f. Envíe el RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Si la RFC es correcta, no recibirá ningún resultado.

g. Para comprobar el estado del RFC, ejecute

```
aws amscm get-rfc --rfc-id RFC_ID
```

Validar el despliegue de la aplicación

Navegue hasta el punto final (ELB CName) del balanceador de cargas creado anteriormente, con la ruta implementada:/. WordPress WordPress Por ejemplo:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Destruya la implementación de aplicaciones

Cuando termines con el tutorial, querrás desmantelar el despliegue para que no te cobren por los recursos.

La siguiente es una operación genérica de eliminación de pilas. Deberá enviarla dos veces, una para la pila de 2 niveles de alta disponibilidad y otra para la pila de cubos de S3. Como seguimiento final, envíe una solicitud de servicio para que se eliminen todas las instantáneas del bucket de S3 (incluya el ID de la pila de S3 en la solicitud de servicio). Se eliminan automáticamente al cabo de 10 días, pero eliminarlas anticipadamente supone un pequeño ahorro de costes.

Este tutorial proporciona un ejemplo del uso de la consola AMS para eliminar una pila S3; este procedimiento se aplica a la eliminación de cualquier pila mediante la consola AMS.

> **Note**
>
> Si borras un depósito de S3, primero debes vaciarlo de objetos.

DATOS NECESARIOS:

- `StackId`: La pila que se va a utilizar. Puedes encontrarlo consultando la página de pilas de consolas de AMS, disponible a través de un enlace en la barra de navegación izquierda. Con la API/CLI de AMS SKMS, ejecute la operación Para la referencia de la API de AMS SKMS, consulte la pestaña Informes de la AWS Artifact Console. (en la CLI). `list-stack-summaries`
- El identificador de tipo de cambio para este tutorial es que la versión es `ct-0q0bic0ywqk6c` «1.0". Para averiguar cuál es la versión más reciente, ejecute este comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0q0bic0ywqk6c
```

CREACIÓN EN LÍNEA:

- Ejecute el comando create RFC con los parámetros de ejecución proporcionados en línea (comillas de escape cuando se proporcionan los parámetros de ejecución en línea). E

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0"
 --title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

- Envíe la RFC mediante el ID de RFC devuelto en la operación de creación de la RFC. Hasta que se envíe, la RFC permanece en ese `Editing` estado y no se toman medidas al respecto.

```
aws amscm submit-rfc --rfc-id RFC_ID
```

- Supervise el estado de la RFC y vea el resultado de la ejecución:

```
aws amscm get-rfc --rfc-id RFC_ID
```

CREACIÓN DE PLANTILLA:

1. Envía la plantilla RFC a un archivo de tu carpeta actual; el ejemplo la nombra DeleteStackRfc .json:

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

2. Modifique y guarde el DeleteStackRfc archivo.json. Como la eliminación de una pila solo tiene un parámetro de ejecución, los parámetros de ejecución pueden estar en el propio DeleteStackRfc archivo.json (no es necesario crear un archivo JSON independiente con los parámetros de ejecución).

   Las comillas internas de la extensión ExecutionParameters JSON deben ir precedidas de una barra invertida (\). Ejemplo sin hora de inicio y finalización:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0q0bic0ywqk6c",
"Title":                "Delete-My-Stack-RFC"
"ExecutionParameters":  "{
        \"StackId\":\"STACK_ID\"}"
}
```

3. Cree el RFC:

```
aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
```

   Recibirá el RfcId nuevo RFC en la respuesta. Por ejemplo:

```
{
"RfcId": "daaa1867-ffc5-1473-192a-842f6b326102"
}
```

   Guarde el ID para los pasos siguientes.

4. Envíe el RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Si la RFC se realiza correctamente, no recibirá ninguna confirmación en la línea de comandos.

5. Para supervisar el estado de la solicitud y ver el resultado de la ejecución:

```
aws amscm get-rfc --rfc-id RFC_ID --query "Rfc.
{Status:Status.Name,Exec:ExecutionOutput}" --output table
```

Tutorial de CLI: Implementación de un WordPress sitio web Tier and Tie

En esta sección se describe cómo implementar un WordPress sitio de alta disponibilidad (HA) en un entorno AMS mediante la CLI de AMS. Este conjunto de instrucciones incluye un ejemplo de cómo crear el archivo WordPress CodeDeploy de paquete compatible necesario (por ejemplo, zip).

> **Note**
>
> Este tutorial de implementación está diseñado para usarse en un entorno Linux de AMZN. Los parámetros de las variables esenciales están anotados como*replaceable*; sin embargo, es posible que desee modificar otros parámetros para adaptarlos a su situación.

Resumen de las tareas y requisitos: RFCs

1. Cree la infraestructura:
   a. Crear una pila de RDS (CLI)
   b. Cree un equilibrador de carga
   c. Cree un grupo de escalado automático y vincúlelo al balanceador de cargas
   d. Cree un bucket de S3 para las aplicaciones CodeDeploy
2. Cree un paquete de WordPress aplicaciones (no requiere un RFC)
3. Implemente el paquete WordPress de aplicaciones con CodeDeploy:
   a. Cree una CodeDeploy aplicación
   b. Cree un grupo CodeDeploy de despliegue
   c. Cargue el paquete de WordPress aplicaciones en el bucket de S3 (no requiere un RFC)
   d. Implemente la aplicación CodeDeploy
4. Valide el despliegue
5. Destruya el despliegue

Siga todos los pasos de la línea de comandos desde su cuenta autenticada.

Creación de un RFC mediante la CLI

Para obtener información detallada sobre la creación RFCs, consulte Creación RFCs; para obtener una explicación de los parámetros de RFC comunes, consulte Parámetros comunes de RFC.

Cree la infraestructura

Los siguientes procedimientos describen la creación de una base de datos de RDS, un balanceador de carga y un grupo de Auto Scaling de tal manera que se utilice el recurso IDs para construir la infraestructura.

Crear una pila de RDS (CLI)

Consulte Pila de RDS | Crear.

Cree una pila ELB

Lanza un balanceador de cargas público (ELB). Consulte Load Balancer (ELB) Stack | Create.

Crear una pila de grupos de Auto Scaling

Lance un grupo de escalado automático.

Consulte Auto Scaling Group | Crear.

Cree una tienda S3

Lanza un depósito de S3. El depósito de S3 es el lugar donde se carga el paquete de aplicaciones que ha creado. Consulte S3 Storage | Create.

Cree un paquete de WordPress aplicaciones para CodeDeploy

En esta sección se proporciona un ejemplo de cómo crear un paquete de despliegue de aplicaciones.

1. Descargue WordPress, extraiga los archivos y cree un. directorio /scripts.

   Comando de Linux:

```
wget https://github.com/WordPress/WordPress/archive/master.zip
```

Windows: pégalo `https://github.com/WordPress/WordPress/archive/master.zip` en una ventana del navegador y descarga el archivo zip.

Cree un directorio temporal en el que ensamblar el paquete.

Linux:

```
mkdir /tmp/WordPress
```

Windows: cree un directorio WordPress «», utilizará la ruta del directorio más adelante.

2. Extraiga la WordPress fuente al directorio WordPress «» y cree un. directorio /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: vaya al directorio «WordPress» que creó y cree allí un directorio de «scripts».

   Si se encuentra en un entorno Windows, asegúrese de establecer el tipo de interrupción de los archivos de script en Unix (LF). En Notepad ++, esta opción se encuentra en la parte inferior derecha de la ventana.

3. Cree el archivo CodeDeploy appspec.yml en el WordPress directorio (si va a copiar el ejemplo, compruebe la hendidura, cada espacio cuenta). IMPORTANTE: Asegúrese de que la ruta «fuente» sea correcta para copiar los WordPress archivos (en este caso, en su WordPress directorio) al destino esperado (/). var/www/html/WordPress En el ejemplo, el archivo appspec.yml está en el directorio con los WordPress archivos, por lo que solo se necesita «/». Además, aunque haya utilizado una AMI de RHEL para su grupo de Auto Scaling, deje la línea «os: linux» tal como está. Ejemplo de archivo 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. Cree scripts de archivos bash en. WordPress directorio /scripts.

   En primer lugar, cree `config_wordpress.sh` con el siguiente contenido (si lo prefiere, puede editar el archivo wp-config.php directamente).

   > **Note**
   >
   > *DBName*Sustitúyalo por el valor indicado en el RFC de HA Stack (por ejemplo,`wordpress`).
   > *DB_MasterUsername*Sustitúyalo por el `MasterUsername` valor indicado en el RFC de HA Stack (por ejemplo,`admin`).
   > *DB_MasterUserPassword*Sustitúyalo por el `MasterUserPassword` valor indicado en el RFC de HA Stack (por ejemplo,`p4ssw0rd`).
   > *DB_ENDPOINT*Sustitúyalo por el nombre DNS del punto final en los resultados de ejecución del RFC de la pila HA (por ejemplo,`srt1cz23n45sfg.clgvd67uvydk.us-east-1.rds.amazonaws.com`). Puede encontrarlo en la GetRfcoperación (CLI: get-rfc --rfc-id RFC_ID) o en la página de detalles del RFC de la consola AMS para el RFC de la pila de alta disponibilidad que envió anteriormente.

```
#!/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. En `install_dependencies.sh` el mismo directorio, cree con el siguiente contenido:

```
#!/bin/bash
yum install -y php
yum install -y php-mysql
yum install -y mysql
service httpd restart
```

> **Note**
>
> HTTPS se instala como parte de los datos del usuario en el momento del lanzamiento para permitir que las comprobaciones de estado funcionen desde el principio.

6. En el mismo directorio, cree `start_server.sh` con el siguiente contenido:

   - Para las instancias de Amazon Linux, usa lo siguiente:

   ```
   #!/bin/bash
   service httpd start
   ```

   - Para las instancias de RHEL, usa esto (los comandos adicionales son políticas que permiten que SELINUX las acepte): 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. En el mismo directorio, cree `stop_server.sh` con el siguiente contenido:

```
#!/bin/bash
service httpd stop
```

8. Cree el paquete zip.

   Linux:

```
$ cd /tmp/WordPress
$ zip -r wordpress.zip .
```

   Windows: Ve a tu directorio WordPress «», selecciona todos los archivos y crea un archivo zip, asegúrate de llamarlo wordpress.zip.

Implemente el paquete de WordPress aplicaciones con CodeDeploy

CodeDeploy Es un servicio de implementación de AWS que automatiza las implementaciones de aplicaciones en las instancias de Amazon EC2 . Esta parte del proceso implica la creación de una CodeDeploy aplicación, la creación de un grupo de CodeDeploy implementación y, a continuación, la implementación de la aplicación mediante. CodeDeploy

Cree una CodeDeploy aplicación

La CodeDeploy aplicación es simplemente un nombre o contenedor que AWS utiliza CodeDeploy para garantizar que se haga referencia a la revisión, la configuración de implementación y el grupo de implementación correctos durante una implementación. La configuración de despliegue, en este caso, es el WordPress paquete que creó anteriormente.

DATOS NECESARIOS:

- `VpcId`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.
- `CodeDeployApplicationName`: Debe ser única en la cuenta. Consulte la CodeDeploy consola para comprobar los nombres de las aplicaciones existentes.
- `ChangeTypeIdyChangeTypeVersion`: El identificador de tipo de cambio para este tutorial `esct-0ah3gwb9seqk2`: para encontrar la última versión, ejecute este comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0ah3gwb9seqk2
```

1. Envía el esquema JSON de los parámetros de ejecución de la CodeDeploy aplicación CT a un archivo de tu carpeta actual; en el ejemplo, se llama Create CDApp Params.json.

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

2. Modifique y guarde el archivo JSON de la siguiente manera; puede eliminar y reemplazar el contenido.

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

3. Envía la plantilla JSON CreateRfc a un archivo de tu carpeta actual; en el ejemplo, se llama Create CDApp RFC.json.

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

4. Modifica y guarda el archivo JSON de la siguiente manera; puedes eliminar y reemplazar el contenido. Tenga en cuenta que `RequestedStartTime` ahora `RequestedEndTime` son opcionales; si se excluyen, el RFC se ejecuta tan pronto como se aprueba (lo que normalmente ocurre automáticamente). Para enviar una RFC «programada», añada esos valores.

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

5. Cree el RFC especificando el archivo de creación de CDApp RFC y el archivo de parámetros de ejecución:

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

   Recibirá el ID de RFC de la nueva RFC en la respuesta. Guarde el ID para los pasos siguientes.

6. Envíe el RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Si la RFC se realiza correctamente, no recibirá ningún resultado.

7. Envíe el RFC:

```
aws amscm get-rfc --rfc-id RFC_ID
```

Cree un grupo de CodeDeploy implementación

Cree el grupo CodeDeploy de despliegue.

Un grupo de CodeDeploy implementación define un conjunto de instancias individuales destinadas a una implementación.

DATOS NECESARIOS:

- `VpcId`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.
- `CodeDeployApplicationName`: Utilice el valor que creó anteriormente.
- `CodeDeployAutoScalingGroups`: Use el nombre del grupo de Auto Scaling que creó anteriormente.
- `CodeDeployDeploymentGroupName`: un nombre para el grupo de implementación. Este nombre debe ser único para cada aplicación asociada al grupo de implementaciones.
- `CodeDeployServiceRoleArn`: Utilice la fórmula que se proporciona en el ejemplo.
- `ChangeTypeIdyChangeTypeVersion`: El identificador de tipo de cambio para este tutorial `esct-2gd0u847qd9d2`, para encontrar la última versión, ejecute este comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-2gd0u847qd9d2
```

1. Envía el esquema JSON de los parámetros de ejecución a un archivo de tu carpeta actual; en el ejemplo, se llama Create CDDep GroupParams .json.

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

2. Modifique y guarde el archivo JSON de la siguiente manera; puede eliminar y reemplazar el contenido.

```
{
"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. Envía la plantilla JSON CreateRfc a un archivo de tu carpeta actual; en el ejemplo, se llama Create CDDep GroupRfc .json.

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

4. Modifica y guarda el archivo JSON de la siguiente manera; puedes eliminar y reemplazar el contenido. Tenga en cuenta que `RequestedStartTime` ahora `RequestedEndTime` son opcionales; si se excluyen, el RFC se ejecuta tan pronto como se aprueba (lo que normalmente ocurre automáticamente). Para enviar una RFC «programada», añada esos valores.

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

5. Cree el RFC especificando el archivo de creación y el CDDep GroupRfc archivo de parámetros de ejecución:

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

   Recibirá el ID de RFC de la nueva RFC en la respuesta. Guarde el ID para los pasos siguientes.

6. Envíe el RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Si la RFC se realiza correctamente, no recibirá ningún resultado.

7. Compruebe el estado del RFC:

```
aws amscm get-rfc --rfc-id RFC_ID
```

Cargue la solicitud WordPress

Tendrá acceso automáticamente a cualquier instancia de bucket de S3 que cree. Puede acceder a él a través de sus Bastions (consulte Acceso a las instancias) o a través de la consola S3 y cargar el CodeDeploy paquete. El paquete debe estar en su lugar para poder seguir desplegando la pila. En el ejemplo, se usa el nombre del bucket creado anteriormente.

```
aws s3 cp wordpress/wordpress.zip s3://ACCOUNT_ID-codedeploy-bundles/
```

Implemente la WordPress aplicación con CodeDeploy

Implemente la CodeDeploy aplicación.

Una vez que tenga el paquete de CodeDeploy aplicaciones y el grupo de implementación, utilice esta RFC para implementar la aplicación.

DATOS NECESARIOS:

- `VPC-ID`: La VPC que está utilizando debe ser la misma que la VPC utilizada anteriormente.
- `CodeDeployApplicationName`: Use el nombre de la CodeDeploy aplicación que creó anteriormente.
- `CodeDeployDeploymentGroupName`: utilice el nombre del grupo de CodeDeploy despliegue que creó anteriormente.
- `S3Location`(donde cargó el paquete de aplicaciones)`S3Bucket`:: El BucketName que creó anteriormente `S3BundleType` y`S3Key`: el tipo y el nombre del paquete que colocó en su tienda de S3.
- `ChangeTypeIdyChangeTypeVersion`: El identificador de tipo de cambio de este tutorial `esct-2edc3sd1sqmrb`: Para encontrar la versión más reciente, ejecute este comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-2edc3sd1sqmrb
```

1. Envía el esquema JSON de los parámetros de ejecución para el CT de despliegue de la CodeDeploy aplicación a un archivo de tu carpeta actual; en el ejemplo, se llama Deploy CDApp Params.json.

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

2. Modifique el archivo JSON de la siguiente manera; puede eliminar y reemplazar el contenido. Para `S3Bucket` ello, usa el `BucketName` que creaste anteriormente.

```
{
"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. Envía la plantilla JSON CreateRfc a un archivo de tu carpeta actual; en el ejemplo, se llama Deploy CDApp RFC.json:

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

4. Modifica y guarda el archivo Deploy CDApp rfc.json; puedes eliminar y reemplazar el contenido.

```
{
"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. Cree el RFC especificando el archivo de parámetros de ejecución y el archivo RFC de despliegueCDApp:

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

   Recibirá el RfcId nuevo RFC en la respuesta. Guarde el ID para los pasos siguientes.

6. Envíe el RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

   Si la RFC se realiza correctamente, no recibirá ningún resultado.

Valide la implementación de la aplicación

Navegue hasta el punto final (ELB CName) del balanceador de cargas creado anteriormente, con la ruta WordPress implementada:/. WordPress Por ejemplo:

```
http://stack-ID-FOR-ELB.us-east-1.elb.amazonaws.com/WordPress
```

Destruya el despliegue de la aplicación

Para desglosar la implementación, debe enviar el Delete Stack CT a la pila de bases de datos de RDS, el balanceador de carga de aplicaciones, el grupo Auto Scaling, el bucket S3 y la aplicación y el grupo Code Deploy (seis en total). RFCs Además, puede enviar una solicitud de servicio para eliminar las instantáneas de RDS (se eliminan automáticamente al cabo de diez días, pero su permanencia en ellas cuesta un pequeño importe). Reúna la pila IDs para todas y, a continuación, siga estos pasos.

Este tutorial proporciona un ejemplo del uso de la consola AMS para eliminar una pila S3; este procedimiento se aplica a la eliminación de cualquier pila mediante la consola AMS.

> **Note**
>
> Si borras un depósito de S3, primero debes vaciarlo de objetos.

DATOS NECESARIOS:

- `StackId`: La pila que se va a utilizar. Puedes encontrarlo consultando la página de pilas de consolas de AMS, disponible a través de un enlace en la barra de navegación izquierda. Con la API/CLI de AMS SKMS, ejecute la operación Para la referencia de la API de AMS SKMS, consulte la pestaña Informes de la AWS Artifact Console. (en la CLI). `list-stack-summaries`
- El identificador de tipo de cambio para este tutorial es que la versión es `ct-0q0bic0ywqk6c` «1.0". Para averiguar cuál es la versión más reciente, ejecute este comando:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=ct-0q0bic0ywqk6c
```

CREACIÓN EN LÍNEA:

- Ejecute el comando create RFC con los parámetros de ejecución proporcionados en línea (comillas de escape cuando se proporcionan los parámetros de ejecución en línea). E

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0"
 --title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

- Envíe la RFC mediante el ID de RFC devuelto en la operación de creación de la RFC. Hasta que se envíe, la RFC permanece en ese `Editing` estado y no se toman medidas al respecto.

```
aws amscm submit-rfc --rfc-id RFC_ID
```

- Supervise el estado de la RFC y vea el resultado de la ejecución:

```
aws amscm get-rfc --rfc-id RFC_ID
```

CREACIÓN DE PLANTILLA:

1. Envía la plantilla RFC a un archivo de tu carpeta actual; el ejemplo la nombra DeleteStackRfc .json:

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

2. Modifique y guarde el DeleteStackRfc archivo.json. Como la eliminación de una pila solo tiene un parámetro de ejecución, los parámetros de ejecución pueden estar en el propio DeleteStackRfc archivo.json (no es necesario crear un archivo JSON independiente con los parámetros de ejecución).

   Las comillas internas de la extensión ExecutionParameters JSON deben ir precedidas de una barra invertida (\). Ejemplo sin hora de inicio y finalización:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0q0bic0ywqk6c",
   "Title":                "Delete-My-Stack-RFC"
   "ExecutionParameters":  "{
           \"StackId\":\"STACK_ID\"}"
   }
   ```

3. Cree el RFC:

   ```
   aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
   ```

Recibirá el RfcId nuevo RFC en la respuesta. Por ejemplo:

```
{
"RfcId": "daaa1867-ffc5-1473-192a-842f6b326102"
}
```

Guarde el ID para los pasos siguientes.

4. Envíe el RFC:

```
aws amscm submit-rfc --rfc-id RFC_ID
```

Si la RFC se realiza correctamente, no recibirá ninguna confirmación en la línea de comandos.

5. Para supervisar el estado de la solicitud y ver el resultado de la ejecución:

```
aws amscm get-rfc --rfc-id RFC_ID --query "Rfc.
{Status:Status.Name,Exec:ExecutionOutput}" --output table
```

Mantenimiento de aplicaciones

Una vez implementada la infraestructura, el desafío es actualizarla de manera uniforme en todos los entornos de AMS, desde el control de calidad hasta la puesta en escena y la producción.

En esta sección, se ofrece una descripción general del proceso de incorporación de la carga de trabajo de AMS y algunos ejemplos de los distintos métodos que puede utilizar para mantener actualizada la capa de infraestructura de la nube.

Estrategias de mantenimiento de aplicaciones

La forma en que se despliegan las aplicaciones afecta a la forma en que se mantienen. En esta sección se proporcionan algunas estrategias para el mantenimiento de las aplicaciones.

Las actualizaciones del entorno pueden implicar cualquiera de estos cambios:

- Actualizaciones de seguridad
- Nuevas versiones de sus aplicaciones
- Cambios en la configuración de la aplicación
- Actualizaciones de las dependencias

> **Note**
>
> Para el despliegue de cualquier aplicación, sea cual sea el método, siempre presente una solicitud de servicio de antemano para informar a AMS de que va a implementar una aplicación.

Ejemplos de instalación de aplicaciones inmutables o mutables

Mutabilidad de instancias de cómputo	Método de instalación de la aplicación	AMI
Mutable	¿Con CodeDeploy	Proporcionado por AMS
	Manualmente	

Mutabilidad de instancias de cómputo	Método de instalación de la aplicación	AMI
	Con un chef o una marioneta, a base de carne	
	Con Ansible o Salt, Push-Based	
Inmutable	Con un AMI dorado	Personalizada (basada en la proporcionada por AMS)

Implementación mutable con una AMI CodeDeploy habilitada

AWS CodeDeploy es un servicio que automatiza las implementaciones de código en cualquier instancia, incluidas las instancias de Amazon y EC2 las instancias que se ejecutan en las instalaciones. Puede usarlo CodeDeploy con AMS para crear e implementar una aplicación. CodeDeploy Tenga en cuenta que AMS proporciona un perfil de instancia predeterminado para CodeDeploy las aplicaciones.

- Amazon Linux (versión 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

Antes de usarlo CodeDeploy por primera vez, debe completar una serie de pasos de configuración:

1. Instalación o actualización de la AWS CLI
2. Cree un rol de servicio para AWS CodeDeploy y utilice el ARN del rol de servicio en la implementación

IDs para ver todas las opciones de CT, consulte la referencia del tipo de cambio.

> **Note**
>
> Actualmente, debe utilizar el almacenamiento de Amazon S3 con esta solución.

Los pasos básicos se describen aquí y el procedimiento se detalla en la Guía del usuario de AMS.

1. Cree un depósito de almacenamiento de Amazon S3. CT: ct-1a68ck03fn98r. El bucket de S3 debe tener habilitado el control de versiones (para obtener información sobre cómo hacerlo, consulte Habilitar el control de versiones de buckets).
2. Coloca los CodeDeploy artefactos incluidos en él. Puede hacerlo con la consola Amazon S3 sin solicitar acceso a través de AMS. O bien, utilizando una variante de este comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Busque una `customer-` AMI de AMS; utilice una de las siguientes opciones:
   - Consola AMS: la página de detalles de la VPC correspondiente
   - API AMS Para obtener información sobre la API AMS SKMS, consulte la pestaña Informes de la AWS Artifact Console. O CLI: `aws amsskms list-amis`
4. Cree un grupo de escalado automático (ASG). CT: ct-2tylseo8rxfsc. Especifique la AMI de AMS, configure el equilibrador de carga para que tenga puertos abiertos y especifique `customer-mc-ec2-instance-profile` la`ASGIAMInstanceProfile`.
5. Cree su CodeDeploy aplicación. CT: ct-0ah3gwb9seqk2. Los parámetros incluyen el nombre de una aplicación, por ejemplo. `WordpressProd`
6. Cree su grupo CodeDeploy de despliegue. CT: ct-2gd0u847qd9d2. Los parámetros incluyen el nombre de CodeDeploy la aplicación, el nombre del ASG, el nombre del tipo de configuración y el ARN del rol de servicio.
7. Implemente la aplicación CodeDeploy . CT: ct-2edc3sd1sqmrb. Los parámetros incluyen el nombre de CodeDeploy la aplicación, el nombre del tipo de configuración, el nombre del grupo de implementación, el tipo de revisión y la ubicación del depósito de S3 donde se encuentran los artefactos. CodeDeploy

Implementación mutable, instancias de aplicaciones configuradas y actualizadas manualmente

Esta estrategia de despliegue de aplicaciones consiste en una actualización sencilla y manual de las instancias de la aplicación. Estos son los pasos básicos.

IDs para ver todas las opciones de tomografía computarizada, consulte la referencia del tipo de cambio.

> **Note**
>
> Actualmente, debe utilizar el almacenamiento de Amazon S3 con esta solución.

Los pasos básicos se describen aquí; los distintos procedimientos se detallan en la Guía del usuario de AMS.

1. Cree un depósito de almacenamiento de Amazon S3. CT: ct-1a68ck03fn98r. El bucket de S3 debe tener habilitado el control de versiones (para obtener información sobre cómo hacerlo, consulte Habilitar el control de versiones de buckets).
2. Coloca los artefactos de la aplicación incluidos en el paquete (todo lo que la aplicación necesita para iniciarse y funcionar). Puede hacerlo con la consola Amazon S3 sin solicitar acceso a través de AMS. O bien, utilizando una variante de este comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Encuentra un AMI de AMS, todos lo tendrán CodeDeploy encima. Para encontrar una AMI «dirigida a un cliente», utilice una de las siguientes opciones:

   - Consola AMS: la página de detalles de la VPC correspondiente
   - API AMS Para obtener información sobre la API AMS SKMS, consulte la pestaña Informes de la AWS Artifact Console. O CLI: `aws amsskms list-amis`

4. Cree una EC2 instancia con esa AMI. CT: ct-14027q0sjyt1h. Especifique la AMI de AMS, establezca una etiqueta `Key=backup, Value=true` y especifique la `customer-mc-ec2-instance-profile` para el `InstanceProfile` parámetro. Anote el ID de instancia que se devuelve.

5. Solicita acceso de administrador a la instancia. CT: ct-1dmlg9g1l91h6. Necesitarás el FQDN de tu cuenta. Si no estás seguro de cuál es tu FQDN, puedes encontrarlo de la siguiente manera:
   - Uso de la pestaña Nombre del directorio de AWS Management Console for Directory Services (en Seguridad e identidad).
   - Ejecute uno de estos comandos (devolver clases de directorio; DC+DC+DC=FQDN): Windows: o Linux:. `whoami /fqdn` `hostname --fqdn`
6. Inicie sesión en la instancia; consulte Acceso a las instancias mediante bastiones en la Guía del usuario de AMS.
7. Descargue los archivos de aplicación empaquetados desde su bucket de S3 a la instancia.
8. Solicita una copia de seguridad inmediata con una solicitud de servicio a AMS. Necesitarás saber el ID de la instancia.
9. Cuando necesite actualizar la aplicación, cargue nuevos archivos en el bucket de S3 y, a continuación, siga los pasos del 3 al 8.

Implementación mutable con una AMI configurada mediante una herramienta de implementación basada en extracciones

Esta estrategia se basa en el `InstanceUserData` parámetro de Managed Services Create EC2 CT. Para obtener más información sobre el uso de este parámetro, consulte Configuración de instancias con datos de usuario. En este ejemplo, se presupone una herramienta de despliegue de aplicaciones basada en extracciones, como Chef o Puppet.

El CodeDeploy agente es compatible con todos los AMS. AMIs Esta es la lista de los compatibles AMIs:

- Amazon Linux (versión 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

IDs para ver todas las opciones de CT, consulte la Referencia de tipos de cambios.

> **Note**
>
> Actualmente, debe utilizar el almacenamiento de Amazon S3 con esta solución.

Los pasos básicos se describen aquí y el procedimiento se detalla en la Guía del usuario de AMS.

1. Cree un depósito de almacenamiento de Amazon S3. CT: ct-1a68ck03fn98r. El bucket de S3 debe tener habilitado el control de versiones (para obtener información sobre cómo hacerlo, consulte Habilitar el control de versiones de buckets).
2. Coloca los CodeDeploy artefactos incluidos en él. Puede hacerlo con la consola Amazon S3 sin solicitar acceso a través de AMS. O bien, utilizando una variante de este comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Busque una `customer-` AMI de AMS; utilice una de las siguientes opciones:
   - Consola AMS: la página de detalles de la VPC correspondiente
   - API AMS Para obtener información sobre la API AMS SKMS, consulte la pestaña Informes de la AWS Artifact Console. O CLI: `aws amsskms list-amis`
4. Cree una instancia. EC2 CT: ct-14027q0sjyt1h; establece una etiqueta `Key=backup, Value=true` y usa el `InstanceUserData` parámetro para especificar un bootstrap y otros scripts (Chef/Puppet agente de descarga, etc.) e incluye las claves de autorización necesarias. Puede encontrar un ejemplo de cómo hacerlo en la guía del usuario de AMS, en la sección de gestión de cambios, ejemplos de cómo crear un despliegue de alta disponibilidad en dos niveles. Como alternativa, solicite acceso a la instancia e inicie sesión en ella y configúrela con los elementos de despliegue necesarios. Recuerde que los comandos de despliegue basados en extracciones van desde los agentes de las instancias hasta el servidor maestro corporativo y es posible que necesiten autorización para pasar por los bastiones. Es posible que necesite una solicitud de servicio a AMS para solicitar el acceso a los group/AD grupos de seguridad sin bastiones.
5. Repita el paso 4 para crear otra EC2 instancia y configurarla con el servidor maestro de la herramienta de despliegue.
6. Cuando necesite actualizar la aplicación, utilice la herramienta de implementación para implementar las actualizaciones en las instancias.

Implementación mutable con una AMI configurada mediante una herramienta de implementación push

Esta estrategia se basa en el `InstanceUserData` parámetro de Managed Services Create EC2 CT. Para obtener más información sobre el uso de este parámetro, consulte Configuración de instancias con datos de usuario. En este ejemplo, se presupone una herramienta de despliegue de aplicaciones basada en extracciones, como Chef o Puppet.

IDs para ver todas las opciones de CT, consulte la referencia de tipos de cambio.

> **Note**
>
> Actualmente, debe utilizar el almacenamiento de Amazon S3 con esta solución.

Los pasos básicos se describen aquí y el procedimiento se detalla en la Guía del usuario de AMS.

1. Cree un depósito de almacenamiento de Amazon S3. CT: ct-1a68ck03fn98r. El bucket de S3 debe tener habilitado el control de versiones (para obtener información sobre cómo hacerlo, consulte Habilitar el control de versiones de buckets).
2. Coloca los CodeDeploy artefactos incluidos en él. Puede hacerlo con la consola Amazon S3 sin solicitar acceso a través de AMS. O bien, utilizando una variante de este comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Encuentra una AMI de AMS, todas las tendrán CodeDeploy en su contra. Para encontrar una AMI «dirigida a un cliente», utilice una de las siguientes opciones:

   - Consola AMS: la página de detalles de la VPC correspondiente
   - API AMS Para obtener información sobre la API AMS SKMS, consulte la pestaña Informes de la AWS Artifact Console. O CLI: `aws amsskms list-amis`

4. Cree una instancia. EC2 CT: ct-14027q0sjyt1h; establezca una etiqueta y utilice el `InstanceUserData` parámetro para ejecutar un bootstrap y otros scriptsKey=backup, `Value=true`, incluidas las claves de autorización, la pila SALT (arranca un minion; para obtener más información, consulte Bootstrapping Salt en Linux EC2 con Cloud-Init) o Ansible (instale un par de claves; para obtener más información, consulte Introducción a Ansible y Dynamic Amazon Inventory Management). EC2 Como alternativa, solicite acceso a la instancia e inicie

sesión en ella y configúrela con los artefactos de implementación necesarios. Recuerde que los comandos push provienen de la subred corporativa y llegan a las instancias y es posible que necesite configurar la autorización para que pasen por los bastiones. Es posible que necesite una solicitud de servicio a AMS para solicitar el acceso a un group/AD grupo de seguridad sin bastiones.

5. Repita el paso 4 para crear otra EC2 instancia y configurarla con el servidor maestro de la herramienta de despliegue.
6. Cuando necesite actualizar la aplicación, utilice la herramienta de implementación para implementar las actualizaciones en las instancias.

Despliegue inmutable con una AMI dorada

Esta estrategia emplea una AMI «dorada» que ha configurado para que se comporte como desea que lo hagan todas las instancias de la aplicación. Por ejemplo, las instancias creadas con esta AMI dorada se unirían automáticamente al dominio y al DNS correctos, se configurarían automáticamente, se reiniciarían y lanzarían todos los sistemas necesarios. Cuando desee actualizar las instancias de aplicación, vuelva a crear la AMI dorada y, con ella, lance instancias de aplicaciones completamente nuevas.

El CodeDeploy agente es compatible con todos los AMS. AMIs Esta es la lista de los compatibles AMIs:

- Amazon Linux (versión 1)
- Amazon Linux 2
- RedHat 7
- CentOS 7

IDs para ver todas las opciones de TC, consulte la referencia del tipo de cambio.

> **Note**
>
> Actualmente, debe utilizar el almacenamiento de Amazon S3 con esta solución.

1. Cree un depósito de almacenamiento de Amazon S3. CT: ct-1a68ck03fn98r. El bucket de S3 debe tener habilitado el control de versiones (para obtener información sobre cómo hacerlo, consulte Habilitar el control de versiones de buckets).
2. Coloca en él los artefactos de la aplicación incluidos (todo lo que la aplicación necesita para iniciarse y funcionar). Puede hacerlo con la consola Amazon S3 sin solicitar acceso a través de AMS. O bien, utilizando una variante de este comando:

   ```
   aws s3 cp ZIP_FILEPATH_AND_NAME s3://S3BUCKET_NAME/
   ```

3. Busque una `customer-` AMI de AMS; utilice una de las siguientes opciones:

   - Consola AMS: la página de detalles de la VPC correspondiente
   - API AMS Para obtener información sobre la API AMS SKMS, consulte la pestaña Informes de la AWS Artifact Console. O CLI: `aws amsskms list-amis`

4. Cree una EC2 instancia con esa AMI. CT: ct-14027q0sjyt1h. Especifique la AMI de AMS, establezca una etiqueta `Key=backup, Value=true` y especifique `customer-mc-ec2-instance-profile` la`InstanceProfile`. Anote el ID de instancia que se devuelve.
5. Solicita acceso de administrador a la instancia. CT: ct-1dmlg9g1l91h6. Necesitarás el FQDN de tu cuenta. Si no estás seguro de cuál es tu FQDN, puedes encontrarlo de la siguiente manera:

   - Uso de la pestaña Nombre del directorio de AWS Management Console for Directory Services (en Seguridad e identidad).
   - Ejecute uno de estos comandos (devolver clases de directorio; DC+DC+DC=FQDN): Windows: o Linux:. `whoami /fqdn` `hostname --fqdn`

6. Inicie sesión en la instancia; consulte Acceso a las instancias en la Guía del usuario de AMS.
7. Descargue en la instancia los archivos de aplicación incluidos desde su bucket de S3. Configure la instancia para que, al arrancar, despliegue automáticamente la aplicación en pleno funcionamiento.
8. Crea el AMI dorado en la instancia. CT: ct-3rqqu43krekby. Para obtener más información, consulte AMI | Create.
9. Configure un grupo de Auto Scaling para crear nuevas instancias con esa AMI. CT: ct-2tylseo8rxfsc. Cuando necesite actualizar su solicitud, siga este procedimiento y solicite a AMS que actualice el ASG para utilizar la nueva AMI dorada; utilice un CT de administración | Otros | Otros | Actualización para ello.

Estrategias de actualización

Existen varias estrategias diferentes que puede emplear para actualizar sus aplicaciones o instancias en su entorno administrado por AMS.

- Tiempo de inactividad programado: esta sencilla estrategia implica programar el tiempo para que la aplicación esté fuera de línea y se actualice manualmente. Para ello, envíe una solicitud de administración | Otros | Otros | Actualización CT (ct-0xdawir96cy7k) para detener las instancias necesarias. Realice las actualizaciones necesarias y, a continuación, envíe otra solicitud de administración | Otros | Otros | Actualización de CT (ct-0xdawir96cy7k) para iniciar las instancias.

- Azul/Verde: esta estrategia requiere disponer de un entorno redundante (dos entornos completamente funcionales) y desconectar uno de ellos mediante actualizaciones del sistema de nombres de dominio (DNS) o del firewall web (WAF) para redirigir el tráfico. Actualice un entorno y, a continuación, vuelva a redirigirlo para actualizar el otro entorno.

  Para obtener más información, consulte AWS CodeDeploy presenta Blue/Green las implementaciones.
- Actualización progresiva con una nueva AMI: aquí es donde tiene una nueva AMI que puede personalizar (consulte Crear una AMI) y, a continuación, solicitar que AMS la despliegue en su grupo de Auto Scaling. Para ello, utilice un CT de administración | Otros | Otros | Actualización (ct-0xdawir96cy7k).

Programador de recursos de AWS Managed Services

Utilice el programador de recursos de AWS Managed Services (AMS) para programar el inicio y la parada automáticos de AutoScaling los grupos, las EC2 instancias de Amazon y las instancias de RDS de su cuenta. Esto ayuda a reducir los costos de infraestructura cuando los recursos no están destinados a funcionar las 24 horas del día, los 7 días de la semana. La solución se basa en Instance Scheduler AWS, pero contiene funciones y personalizaciones adicionales específicas para las necesidades de AMS.

> **Note**
>
> De forma predeterminada, el programador de recursos de AMS no interactúa con los recursos que no forman parte de una pila. AWS CloudFormation El recurso debe formar parte de una pila que comience por «stack-», «sc-» o «SC-». Para programar los recursos

que no forman parte de una CloudFormation pila, puede actualizar el parámetro de pila del Programador de recursos `ScheduleNonStackResources` a`Yes`.

El programador de recursos de AMS utiliza períodos y programas:

- Los períodos definen las horas en las que se ejecuta el Programador de recursos, como la hora de inicio, la hora de finalización y los días del mes.
- Los programas contienen los períodos definidos, junto con configuraciones adicionales, como el período de mantenimiento del SSM, la zona horaria, la configuración de hibernación, etc., y especifican cuándo deben ejecutarse los recursos, según las reglas de período configuradas.

Puede configurar estos períodos y programas mediante los tipos de cambios automatizados del Programador de recursos de AMS (). CTs

Para obtener más información sobre los ajustes disponibles para el programador de recursos de AMS, consulte la documentación correspondiente del programador de AWS instancias en Componentes de la solución. Para obtener una vista de la arquitectura de la solución, consulte la documentación correspondiente del programador de AWS instancias en Architecture overview.html.

Implementación del programador de recursos AMS

Para implementar el programador de recursos de AMS, utilice el tipo de cambio automático (CT): Despliegue | Programador de recursos de AMS | Solución | Implementación (ct-0ywnhc8e5k9z5) para generar una RFC que, a continuación, despliegue la solución en su cuenta. Una vez ejecutada la RFC, se aprovisiona automáticamente en su cuenta una pila que contiene los recursos del programador de recursos de AMS con CloudFormation la configuración predeterminada. Para obtener más información sobre los tipos de cambios del Programador de recursos, consulte el Programador de recursos de AMS.

> **Note**
>
> Para saber si AMS Resource Scheduler ya está implementado en su cuenta, consulte la consola AWS Lambda de esa cuenta y busque AMSResourcela función Scheduler.

Una vez aprovisionado el programador de recursos de AMS en su cuenta, le recomendamos que revise la configuración predeterminada y, si es necesario, que personalice las configuraciones, como

la clave de etiqueta, la zona horaria, los servicios programados, etc., en función de sus preferencias. Para obtener más información sobre las personalizaciones recomendadas, consulte a continuación. Personalización del programador de recursos AMS

Para realizar las configuraciones personalizadas o simplemente confirmar la configuración del Resource Scheduler,

Personalización del programador de recursos AMS

Le recomendamos que personalice las siguientes propiedades del programador de recursos de AMS mediante los tipos de cambio actualizados del programador de recursos de AMS; consulte el programador de recursos de AMS.

- Nombre de etiqueta: el nombre de la etiqueta que el Programador de recursos utilizará para asociar los cronogramas de instancias a los recursos. El valor predeterminado es Schedule.
- Servicios programados: lista de servicios separados por comas que Resource Scheduler puede administrar. El valor predeterminado es «ec2, rds, autoscaling». Los valores válidos son «ec2", «rds» y «autoscaling»
- Zona horaria predeterminada: especifique la zona horaria predeterminada que utilizará el programador de recursos. El valor predeterminado es UTC.
- Utilice CMK: una lista separada por comas de la clave gestionada por el cliente (CMK) de Amazon KMS a la ARNs que se le pueden conceder permisos a Resource Scheduler.
- Uso LicenseManager: se pueden conceder permisos para ese programador de recursos a una lista separada por comas de ARNs los administradores de AWS licencias.

> **Note**
>
> AMS puede, de vez en cuando, publicar funciones y correcciones para mantener el Programador de recursos de AMS actualizado en su cuenta. Cuando esto ocurre, se conservan todas las personalizaciones que realices en el programador de recursos de AMS.

Uso del programador de recursos AMS

Para configurar el programador de recursos de AMS una vez implementada la solución, utilice el programador de recursos automatizado CTs para crear, eliminar, actualizar y describir (obtener detalles sobre) los períodos del programador de recursos de AMS (las horas en que se ejecuta

el programador de recursos) y los programas (los períodos configurados y otras opciones). Para ver un ejemplo del uso de los tipos de cambio del programador de recursos de AMS, consulte el programador de recursos de AMS.

Para seleccionar los recursos que gestionará el programador de recursos de AMS, tras la creación del despliegue y el programa, utilice la etiqueta AMS Create CTs para etiquetar los grupos de Auto Scaling, las pilas de Amazon RDS y EC2 los recursos de Amazon con la clave de etiqueta que proporcionó durante la implementación y el programa definido como valor de etiqueta. Una vez etiquetados los recursos, se programa su inicio o finalización según el cronograma definido por el programador de recursos.

El uso del programador de recursos de AMS no conlleva ningún coste adicional. Sin embargo, la solución utiliza varios Servicios de AWS y se le cobrará por estos recursos a medida que se utilicen. Para obtener más información, consulte Descripción general de la arquitectura.

Para excluirse del programador de recursos de AMS:

- Para excluirse o deshabilitarse temporalmente: envíe una RFC mediante el tipo de cambio automatizado Management | AMS Resource Scheduler | State | Disable (ct-14v49adibs4db)
- Para su eliminación definitiva, envíe un RFC de administración | Otros | Otros | Actualice (es necesario revisar) (ct-0xdawir96cy7k) solicitando su eliminación del sistema automatizado de versiones de Resource Scheduler

Estimador de costos de AMS Resource Scheduler

Para hacer un seguimiento del ahorro de costes, el programador de recursos de AMS incluye un componente que calcula cada hora el ahorro de costes estimado para los recursos de Amazon EC2 y RDS gestionados por el programador. Estos datos de ahorro de costes se publican luego como una CloudWatch métrica (`AMS/ResourceScheduler`) para ayudarle a realizar un seguimiento de los mismos. El estimador de ahorros de costos solo estima los ahorros en las horas de funcionamiento de una instancia. No tiene en cuenta ningún otro coste, como los costes de transferencia de datos asociados a un recurso.

El estimador de ahorro de costes está activado con Resource Scheduler. Funciona cada hora y recupera datos de costos y uso de. AWS Cost Explorer A partir de esos datos, calcula el costo promedio por hora para cada tipo de instancia y, a continuación, proyecta el costo de un día completo si se ejecutó sin programarse. El ahorro de costos es la diferencia entre el costo proyectado y el costo real informado por Cost Explorer para un día determinado.

Por ejemplo, si la instancia A está configurada con el Programador de recursos para que se ejecute de 9 a. m. a 5 p. m., son ocho horas de un día determinado. Cost Explorer informa que el costo es de 1 dólar y el uso de 8. Por lo tanto, el costo promedio por hora es de 0,125 USD. Si la instancia no estaba programada con Resource Scheduler, se ejecutaría 24 horas ese día. En ese caso, el costo habría sido de 24 x 0,125 = 3$. Resource Scheduler le ayudó a ahorrar 2 dólares en costos.

Para que el estimador de ahorros de costos recupere el costo y el uso solo de los recursos administrados por Resource Scheduler desde Cost Explorer, la clave de etiqueta que el Programador de recursos utiliza para destinar los recursos debe activarse como etiqueta de asignación de costos en el panel de facturación. Si la cuenta pertenece a una organización, la clave de etiqueta debe estar activada en la cuenta de administración de la organización. Para obtener información sobre cómo hacerlo, consulte Activación de etiquetas de asignación de costes definidas por el usuario y etiquetas de asignación de costes definidas por el usuario

Una vez que la clave de etiqueta se activa como etiqueta de asignación de costos, la AWS facturación comienza a hacer un seguimiento del costo y el uso de los recursos administrados por el programador de recursos y, una vez disponibles los datos, la calculadora de ahorros de costos comienza a calcular los ahorros de costos y a publicar los datos en el espacio de nombres de las métricas en. `AMS/ResourceScheduler` CloudWatch

Consejos para la estimación de costos

Cost Savings Estimator no acepta descuentos como instancias reservadas, planes de ahorro, etc., para su cálculo. El estimador toma los costos de uso de Cost Explorer y calcula el costo promedio por hora de los recursos. Para obtener más información, consulte Cómo entender sus conjuntos de datos de AWS costos: una hoja de referencia

Para que el estimador de ahorros de costos recupere el costo y el uso solo de los recursos administrados por Resource Scheduler desde Cost Explorer, la clave de etiqueta que el Programador de recursos utiliza para destinar los recursos debe activarse como etiqueta de asignación de costos en el panel de facturación. Si la cuenta pertenece a una organización, la clave de etiqueta debe estar activada en la cuenta de administración de la organización. Para obtener información sobre cómo hacerlo, consulte Etiquetas de asignación de costes definidas por el usuario. Si la etiqueta de asignación de costes no está activada, el estimador no podrá calcular los ahorros ni publicar la métrica, aunque esté habilitada.

Mejores prácticas de AMS Resource Scheduler

Programación de Amazon EC2 Instances

- El comportamiento de cierre de instancias debe estar configurado en `stop` y no `enterminate`. Está preestablecido `stop` para las instancias que se crean con el tipo de cambio automatizado AMS Amazon EC2 Create (ct-14027q0sjyt1h) y se puede configurar para las instancias de EC2 Amazon creadas con la ingestión, estableciendo la propiedad en. AWS CloudFormation `InstanceInitiatedShutdownBehavior` `stop` Si el comportamiento de cierre de las instancias está establecido `enterminate`, finalizarán cuando el programador de recursos las detenga y el programador no pueda volver a iniciarlas.
- AMS Resource Scheduler no procesa individualmente las EC2 instancias de Amazon que forman parte de un grupo de Auto Scaling, aunque estén etiquetadas.
- Si el volumen raíz de la instancia de destino está cifrado con una clave maestra de cliente (CMK) de KMS, es necesario añadir un `kms:CreateGrant` permiso adicional a la función de IAM del programador de recursos para que el programador pueda iniciar dichas instancias. Este permiso no se añade a la función de forma predeterminada para mejorar la seguridad. Si necesita este permiso, envíe una RFC al tipo de cambio Management | AMS Resource Scheduler | Solution | Update y especifique una lista ARNs del KMS separada por comas. CMKs

Programación de grupos de Auto Scaling

- El programador de recursos de AMS inicia o detiene el escalado automático de los grupos de Auto Scaling, no de las instancias individuales del grupo. Es decir, el planificador restaura el tamaño del grupo de Auto Scaling (inicio) o establece el tamaño en 0 (parada).
- Etiquete el AutoScaling grupo con la etiqueta especificada y no las instancias del grupo.
- Durante la parada, el programador de recursos AMS almacena los valores de capacidad mínima, deseada y máxima del grupo Auto Scaling y establece la capacidad mínima y deseada en 0. Durante el inicio, el planificador restaura el tamaño del grupo de Auto Scaling tal como estaba durante la parada. Por lo tanto, las instancias de grupo de Auto Scaling deben usar una configuración de capacidad adecuada para que la finalización y el relanzamiento de las instancias no afecten a ninguna aplicación que se ejecute en el grupo Auto Scaling.
- Si se modifica el grupo de Auto Scaling (la capacidad mínima o máxima) durante un período de ejecución, el planificador almacena el nuevo tamaño del grupo de Auto Scaling y lo usa para restaurar el grupo al final de una programación de paradas.

Programación de instancias de Amazon RDS

- El programador puede tomar una instantánea antes de detener las instancias de RDS (no se aplica al clúster de base de datos Aurora). Esta función está activada de forma predeterminada con el parámetro de CloudFormation plantilla Crear instantánea de instancia de RDS establecido en true. La instantánea se conserva hasta la próxima vez que se detenga la instancia de Amazon RDS y se cree una nueva instantánea.

  El programador puede ser una instancia de start/stop Amazon RDS que forme parte de un clúster o una base de datos Aurora de Amazon RDS o que se encuentre en una configuración de zona de disponibilidad múltiple (Multi-AZ). Sin embargo, compruebe las limitaciones de Amazon RDS cuando el programador no pueda detener la instancia de Amazon RDS, especialmente las instancias Multi-AZ. Para programar el inicio o la parada del Clúster de Aurora, utilice el parámetro de plantilla Programar cúmulos de Aurora (el valor predeterminado es true). El clúster Aurora (no las instancias individuales del clúster) debe etiquetarse con la clave de etiqueta definida durante la configuración inicial y el nombre del programa como valor de etiqueta para programar ese clúster.

  Cada instancia de Amazon RDS tiene un período de mantenimiento semanal durante el cual se aplican los cambios en el sistema. Durante el período de mantenimiento, Amazon RDS iniciará automáticamente las instancias que hayan estado detenidas durante más de siete días para aplicar el mantenimiento. Tenga en cuenta que Amazon RDS no detendrá la instancia una vez que se complete el evento de mantenimiento.

  El programador permite especificar si se debe añadir el período de mantenimiento preferido de una instancia de Amazon RDS como período de ejecución a su programación. La solución iniciará la instancia al principio del período de mantenimiento y la detendrá al final del período de mantenimiento si ningún otro período de ejecución especifica que la instancia debe ejecutarse y si el evento de mantenimiento se ha completado.

  Si el evento de mantenimiento no se completa al final del período de mantenimiento, la instancia se ejecutará hasta el intervalo de programación posterior a la finalización del evento de mantenimiento.

> **Note**
>
> El programador no valida que un recurso se haya iniciado o detenido. Realiza la llamada a la API y sigue adelante. Si la llamada a la API falla, registra el error para su investigación.

Consideraciones de seguridad de las aplicaciones

La seguridad de las aplicaciones incluye considerar qué permisos necesitará la aplicación para ejecutarse, qué reglas de firewall y qué funciones de IAM deben habilitarse para acceder a la aplicación.

Para comprender mejor la AWS seguridad general, consulte las prácticas recomendadas en materia de seguridad, identidad y conformidad.

Acceso para la administración de la configuración

AWS Managed Services (AMS) busca proporcionarle una infraestructura sin complicaciones para que no tenga que preocuparse por problemas de seguridad, problemas de parches, problemas de respaldo, etc. Para ello, AMS recomienda que las funciones de IAM sean mínimas y que solo un grupo específico o un servidor maestro, si utilizan una herramienta de despliegue de aplicaciones, puedan acceder a las instancias en las que se ejecuta la aplicación.

Reglas de firewall de acceso a las aplicaciones

Al igual que el sistema operativo (SO), el acceso a todas las aplicaciones debe regirse mediante grupos de Active Directory (AD). Si utiliza Amazon Relational Database Service (Amazon RDS) como ejemplo, debe romper el espejo (replicación) para añadir un nuevo usuario. El mejor enfoque es crear un grupo en AD y añadirlo en el momento de crear la base de datos. Tener los grupos en su AD de AMS significa que puede crearlos CTs para el acceso a las aplicaciones. Para obtener información sobre la estrategia de agrupamiento oficial de AD, consulte Uso de la estrategia de anidación de grupos: prácticas recomendadas de AD para la estrategia de grupos.

Para obtener más información sobre los árboles de parent/child dominios y los dominios, consulte Cómo funcionan los dominios y los bosques.

Las siguientes reglas ilustran una solución adecuada para una confianza forestal de varios dominios con usuarios ubicados en dominios secundarios.

Instancias de Windows

Estas son las reglas que debe configurar para los controladores de dominio principales y secundarios de Windows.

Controlador de dominio principal, Windows

DE: controladores de dominio principales A: subredes apiladas y de servicios compartidos de Windows

Puerto de origen	Puerto de destino	Protocolo
88	49152 - 65535	TCP
389	49152 - 65535	UDP

DESDE: apile subredes, incluidos los servicios compartidos, HASTA: controladores de dominio raíz de bosques de Windows

Puerto de origen	Puerto de destino	Protocolo
49152 - 65535	88	TCP
49152 - 65535	389	UDP

Controlador de dominio secundario, Windows

DE: controladores de dominio secundarios A: controladores de dominio AWS de Windows

Puerto de origen	Puerto de destino	Protocolo
49152 - 65535	53	TCP
49152 - 65535	88	TCP
49152 - 65535	389	UDP

DE: controladores de dominio secundarios A: subredes apiladas y de servicios compartidos de Windows

Puerto de origen	Puerto de destino	Protocolo
88	49152 - 65535	TCP

Puerto de origen	Puerto de destino	Protocolo
135	49152 - 65535	TCP
389	49152 - 65535	TCP
389	49152 - 65535	UDP
445	49152 - 65535	TCP
49152 - 65535	49152 - 65535	TCP

DESDE: Apile subredes, incluidos los servicios compartidos, HASTA: controladores de dominio secundarios de Windows

Puerto de origen	Puerto de destino	Protocolo
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

Instancias de Linux

Estas son las reglas que debe configurar para los controladores de dominio principales y secundarios de Linux.

Todas las pruebas se realizaron con Amazon Linux. Si bien el rango de puertos dinámicos para Windows es del 49152 al 65535, muchos núcleos de Linux utilizan el rango de puertos del 32768 al 61000. Ejecute el siguiente comando para ver el rango de puertos IP.

```
cat /proc/sys/net/ipv4/ip_local_port_range
```

Controlador de dominio principal, Linux

DE: controladores de dominio principales A: subredes apiladas y de servicios compartidos de Linux

Puerto de origen	Puerto de destino	Protocolo
389	32768 - 61000	UDP
88	32768 - 61000	TCP

DESDE: Apile subredes, incluidos los servicios compartidos, HASTA: controladores de dominio raíz de bosques de Linux

Puerto de origen	Puerto de destino	Protocolo
32768 - 61000	88	TCP
32768 - 61000	389	UDP

Controlador de dominio secundario, Linux

DE: controladores de dominio secundarios A: controladores de dominio AWS para Linux

Puerto de origen	Puerto de destino	Protocolo
49152 - 65535	53	TCP
49152 - 65535	88	TCP
389	49152 - 65535	UDP
49152 - 65535	389	UDP

DESDE: controladores de dominio secundarios HASTA: subredes apiladas y de servicios compartidos de Linux

Puerto de origen	Puerto de destino	Protocolo
88	32768 - 61000	TCP
389	32768 - 61000	UDP

DESDE: Apile subredes, incluidos los servicios compartidos, HASTA: controlador de dominio secundario de Linux

Puerto de origen	Puerto de destino	Protocolo
32768 - 61000	88	TCP
32768 - 61000	389	UDP

Gestión del tráfico de salida AMS

De forma predeterminada, la ruta con un CIDR de destino de 0.0.0.0/0 para las subredes privadas y de aplicaciones de cliente de AMS tiene como destino una puerta de enlace de traducción de direcciones de red (NAT). Los servicios TrendMicro y los parches de AMS son componentes que deben tener acceso de salida a Internet para que AMS pueda prestar su servicio y los sistemas operativos puedan obtener actualizaciones. TrendMicro

AMS permite desviar el tráfico de salida a Internet a través de un dispositivo de salida gestionado por el cliente siempre que:

- Actúa como un proxy implícito (por ejemplo, transparente).

  y

- Permite las dependencias HTTP y HTTPS de AMS (enumeradas en esta sección) para permitir la aplicación continua de parches y el mantenimiento de la infraestructura gestionada por AMS.

Algunos ejemplos son:

- La pasarela de tránsito (TGW) tiene una ruta predeterminada que apunta al firewall local administrado por el cliente a través de la conexión AWS Direct Connect en la cuenta de red de zona de aterrizaje multicuenta.
- El TGW tiene una ruta predeterminada que apunta a un punto de enlace de AWS en la VPC de salida de múltiples cuentas que aprovecha AWS y apunta a un proxy administrado por el cliente en otra cuenta de PrivateLink AWS.
- El TGW tiene una ruta predeterminada que apunta a un firewall administrado por el cliente en otra cuenta de AWS, con una conexión site-to-site VPN adjunta a la TGW de la zona de destino multicuenta.

AMS ha identificado las dependencias HTTP y HTTPS correspondientes de AMS, y las desarrolla y perfecciona de forma continua. Consulte egressMgmt.zip. Junto con el archivo JSON, el ZIP contiene un archivo README.

> **Note**
>
> - Esta información no es exhaustiva; algunos sitios externos obligatorios no aparecen aquí.
> - No utilices esta lista como parte de una lista de rechazados o una estrategia de bloqueo.
> - Esta lista pretende ser un punto de partida para un conjunto de reglas de filtrado de egreso, con la expectativa de que se utilicen herramientas de generación de informes para determinar con precisión dónde se diferencia el tráfico real de la lista.

Para solicitar información sobre cómo filtrar el tráfico de salida, envíe un correo electrónico a su CSDM: ams-csdm@amazon.com.

Grupos de seguridad

En AWS VPCs, los grupos de seguridad de AWS actúan como firewalls virtuales y controlan el tráfico de una o más pilas (una instancia o un conjunto de instancias). Cuando se lanza una pila, se asocia a uno o más grupos de seguridad, que determinan qué tráfico puede llegar a ella:

- En el caso de las pilas de las subredes públicas, los grupos de seguridad predeterminados aceptan el tráfico de HTTP (80) y HTTPS (443) procedente de todas las ubicaciones (Internet). Las pilas también aceptan tráfico SSH y RDP interno de su red corporativa y de los bastiones de AWS.

Luego, esas pilas pueden salir a Internet a través de cualquier puerto. También pueden salir a las subredes privadas y a otras pilas de la subred pública.

- Las pilas de tus subredes privadas pueden salir a cualquier otra pila de tu subred privada, y las instancias de una pila pueden comunicarse completamente entre sí a través de cualquier protocolo.

> **Important**
>
> El grupo de seguridad predeterminado para las pilas de las subredes privadas permite que todas las pilas de la subred privada se comuniquen con otras pilas de esa subred privada. Si desea restringir las comunicaciones entre las pilas de una subred privada, debe crear nuevos grupos de seguridad que describan la restricción. Por ejemplo, si desea restringir las comunicaciones a un servidor de base de datos para que las pilas de esa subred privada solo puedan comunicarse desde un servidor de aplicaciones específico a través de un puerto específico, solicite un grupo de seguridad especial. En esta sección se describe cómo hacerlo.

Grupos de seguridad predeterminados

MALZ

En la siguiente tabla se describe la configuración predeterminada del grupo de seguridad entrante (SG) para sus pilas. El SG se denomina «SentinelDefaultSecurityGroupPrivateOnly-VPC-ID» y es *ID* un ID de VPC en tu cuenta de zona de landing zone multicuenta de AMS. Se permite que todo el tráfico salga a «mc-initial-garden-SentinelDefaultSecurityGroupPrivateOnly» a través de este grupo de seguridad (se permite todo el tráfico local dentro de las subredes apiladas).

Un segundo grupo de seguridad "» permite que todo el tráfico salga a 0.0.0.0/0. SentinelDefaultSecurityGroupPrivateOnly

> **Tip**
>
> Si elige un grupo de seguridad para un tipo de cambio de AMS, como EC2 crear o OpenSearch crear un dominio, debe utilizar uno de los grupos de seguridad predeterminados que se describen aquí o un grupo de seguridad que haya creado. Puede

encontrar la lista de grupos de seguridad, por VPC, en la consola de AWS o en la EC2 consola de VPC.

Hay otros grupos de seguridad predeterminados que se utilizan con fines internos de AMS.

Grupos de seguridad predeterminados de AMS (tráfico entrante)

Tipo	Protocolo	Intervalo de puertos	Origen
Todo el tráfico	Todos	Todos	SentinelDefaultSecurityGroupPrivateOnly (restringe el tráfico saliente a los miembros del mismo grupo de seguridad)
Todo el tráfico	Todos	Todos	SentinelDefaultSecurityGroupPrivateOnlyEgressAll (no restringe el tráfico saliente)
HTTP, HTTPS, SSH, RDP	TCP	80/443 (fuente 0.0.0.0/0) Se permite el acceso SSH y RDP desde los bastiones	SentinelDefaultSecurityGroupPublic (no restringe el tráfico saliente)
Bastiones de MALZ:			
SSH	TCP	22	SharedServices CIDR de VPC y CIDR de VPC DMZ, además de entornos locales proporcionados por el cliente CIDRs
SSH	TCP	22	
RDP	TCP	3389	
RDP	TCP	3389	
Bastiones de SALZ:			
SSH	TCP	22	mc-initial-garden- SG LinuxBastion
SSH	TCP	22	mc-initial-garden- LinuxBastion DMZSG

Tipo	Protocolo	Intervalo de puertos	Origen
RDP	TCP	3389	mc-initial-garden- SG WindowsBastion
RDP	TCP	3389	mc-initial-garden- WindowsBastion DMZSG

SALZ

En la siguiente tabla se describe la configuración predeterminada del grupo de seguridad entrante (SG) para sus pilas. El SG se denomina «mc-initial-garden-SentinelDefaultSecurityGroupPrivateOnly -*ID*» y *ID* es un identificador único. Se permite que todo el tráfico salga a «mc-initial-garden-SentinelDefaultSecurityGroupPrivateOnly» a través de este grupo de seguridad (se permite todo el tráfico local dentro de las subredes apiladas).

Un segundo grupo de seguridad, "- -», permite que todo el tráfico salga a 0.0.0.0/0. mc-initial-garden SentinelDefaultSecurityGroupPrivateOnlyEgressAll *ID*

> **Tip**
>
> Si elige un grupo de seguridad para un tipo de cambio de AMS, como EC2 crear o OpenSearch crear un dominio, debe utilizar uno de los grupos de seguridad predeterminados que se describen aquí o un grupo de seguridad que usted haya creado. Puede encontrar la lista de grupos de seguridad, por VPC, en la consola de AWS o en la EC2 consola de VPC.

Hay otros grupos de seguridad predeterminados que se utilizan con fines internos de AMS.

Grupos de seguridad predeterminados de AMS (tráfico entrante)

Tipo	Protocolo	Intervalo de puertos	Origen
Todo el tráfico	Todos	Todos	SentinelDefaultSecurityGroupPrivateOnly (restringe el tráfico saliente a los miembros del mismo grupo de seguridad)

Tipo	Protocolo	Intervalo de puertos	Origen
Todo el tráfico	Todos	Todos	SentinelDefaultSecurityGroupPrivateOnlyEgressAll (no restringe el tráfico saliente)
HTTP, HTTPS, SSH, RDP	TCP	80/443 (fuente 0.0.0.0/0) Se permite el acceso SSH y RDP desde los bastiones	SentinelDefaultSecurityGroupPublic (no restringe el tráfico saliente)
Bastiones de MALZ:			
SSH	TCP	22	SharedServices CIDR de VPC y CIDR de VPC DMZ, además de entornos locales proporcionados por el cliente CIDRs
SSH	TCP	22	
RDP	TCP	3389	
RDP	TCP	3389	
Bastiones de SALZ:			
SSH	TCP	22	mc-initial-garden- SG LinuxBastion
SSH	TCP	22	mc-initial-garden- LinuxBastion DMZSG
RDP	TCP	3389	mc-initial-garden- SG WindowsBastion
RDP	TCP	3389	mc-initial-garden- WindowsBastion DMZSG

Crear, cambiar o eliminar grupos de seguridad

Puede solicitar grupos de seguridad personalizados. En los casos en que los grupos de seguridad predeterminados no satisfagan las necesidades de sus aplicaciones o de su organización, puede modificar o crear nuevos grupos de seguridad. Esta solicitud se consideraría necesaria para su aprobación y sería examinada por el equipo de operaciones de AMS.

Para crear un grupo de seguridad fuera de las pilas VPCs, envíe una RFC con el tipo de `Deployment | Advanced stack components | Security group | Create (review required)` cambio (ct-1oxx2g2d7hc90).

Para las modificaciones de los grupos de seguridad de Active Directory (AD), utilice los siguientes tipos de cambios:

- Para añadir un usuario: envíe un RFC mediante Management | Directory Service | Usuarios y grupos | Añadir usuario al grupo [ct-24pi85mjtza8k]
- Para eliminar un usuario: envíe un RFC mediante Management | Directory Service | Users and groups | Eliminar usuario del grupo [ct-2019s9y3nfml4]

> **Note**
>
> Cuando utilice la opción «revisión obligatoria» CTs, AMS recomienda que utilice la opción de programación lo antes posible (elija ASAP en la consola y deje en blanco la hora de inicio y finalización en la API/CLI), ya que CTs requiere que un operador de AMS examine la RFC y, posiblemente, se comunique con usted antes de que pueda aprobarse y ejecutarse. Si las programa RFCs, asegúrese de esperar al menos 24 horas. Si la aprobación no se produce antes de la hora de inicio programada, la RFC se rechaza automáticamente.

Busque grupos de seguridad

Para buscar los grupos de seguridad adjuntos a una pila o instancia, utilice la EC2 consola. Tras encontrar la pila o la instancia, podrá ver todos los grupos de seguridad asociados a ella.

Para obtener información sobre cómo buscar grupos de seguridad en la línea de comandos y filtrar los resultados, consulte `describe-security-groups`.

Apéndice: Cuestionario de incorporación de solicitudes

Utilice este cuestionario para describir los elementos y la estructura de la implementación, de modo que AMS pueda determinar qué componentes de infraestructura son necesarios. Los requisitos de incorporación de las aplicaciones Line-of-Business (LoB) son significativamente diferentes a los de las aplicaciones de productos, por lo que este cuestionario está diseñado para abordar ambos.

Temas

Resumen de despliegue

Descripción de la implementación. Por ejemplo:

- Esta cuenta es para el despliegue de una aplicación Line-of-Business (LoB) (a diferencia del despliegue de una aplicación de producto).
- La implementación implica un ARP (proxy inverso autenticado) escalado automáticamente dentro de la subred de la cuenta. public/DMZ
- Los servidores web y de aplicaciones se implementarán en la subred privada de la cuenta.
- También se implementará una instancia de Amazon RDS (Amazon Relational Database Service) en la subred privada de la cuenta.
- Los servidores (ARP, web, aplicación, base de datos, balanceador de carga, etc.) están separados en grupos de seguridad distintos.
- La cuenta requiere un diseño de alta disponibilidad (HA) distribuido en todas las zonas de disponibilidad (), es decir, en zonas de disponibilidad múltiples (Multi-AZAZs).

Componentes de despliegue de infraestructura

¿Cuáles son los distintos componentes que deberán configurarse para ser compatibles con su aplicación?

- Región: ¿Qué Región de AWS o qué regiones se necesitan?
- Alta disponibilidad (HA): ¿Qué zonas de disponibilidad se utilizarán?
- Virtual Private Cloud (VPC): ¿Qué es el bloque CIDR de la VPC?
- ¿Qué instancias de servidor se necesitan?
  - Proxy inverso autenticado (ARP): ¿sistema operativo, AMI, tipo de instancia, ID de subred, grupo de seguridad, puerto de entrada?
  - Servidor de herramientas de implementación de aplicaciones: ¿sistema operativo, AMI, tipo de instancia, ID de subred, grupo de seguridad, puerto de entrada (Chef, Puppet) o puerto de salida (Ansible, Saltstack)?
  - Amazon RDS con MySQL: versión de base de datos, tipo de uso, clase de instancia, ID de subred, grupo de seguridad, ID de instancia de base de datos, tamaño de almacenamiento, Multi-AZ, tipo de autenticación, ¿cifrado?
  - Almacenamiento: ¿su aplicación no tiene estado? ¿Necesita depósitos S3? ¿Necesitas almacenamiento persistente? ¿Necesita cifrar los datos en reposo en sus volúmenes de EBS? ¿Necesita el cifrado de la base de datos?
  - Puntos finales de servidor externos (a la VPC de Managed Services): ¿SMTP? ¿LDAP?
  - Requisitos de red: ¿filtrado de red (¿basado en grupos de seguridad?)? Inspección del tráfico web (¿entrante? saliente?)?
- Etiquetado: ¿Qué etiquetas deberían usarse para agrupar los recursos en colecciones lógicas? Por ejemplo, todos los recursos de una pila de aplicaciones. Seleccione etiquetas para su caso de uso; por ejemplo, `backup=true` para habilitar las copias de seguridad. Además, debes usar la etiqueta `name=value` para que cualquier EC2 instancia que crees muestre un nombre en la consola.
- Grupos de seguridad:
  - ¿Qué grupos de seguridad se necesitan?
  - ¿Reglas de ingreso de grupos de seguridad?
  - ¿Reglas de salida de los grupos de seguridad?

Plataforma de alojamiento de aplicaciones

Para su plataforma de alojamiento de aplicaciones, tenga en cuenta los siguientes requisitos posibles:

- ¿Base de datos cifrada?
- ¿Quién administra las claves de cifrado?
- ¿Están cifrados todos los datos en tránsito y en reposo?
- ¿Todos los usuarios acceden al sistema a través de HTTPS?
- ¿Todas system-to-system las interacciones fueron aprobadas por su equipo de operaciones de seguridad?

Modelo de despliegue de aplicaciones

Consideraciones sobre cómo planificar las implementaciones de sus aplicaciones. Consulte ¿Cuál es mi modelo operativo?

- ¿Automatizado o manual? Sin automatización del despliegue, no hay escalabilidad automática. Si solicita acceso e inicia sesión y actualiza manualmente su aplicación, la actualización falla. AMS espera que anules la actualización o que nos avises mediante una solicitud de servicio para que podamos ayudarte.
- Si es automática, ¿cuál es el marco? ¿Guiones? ¿Basado en agentes ()? puppet/chef)? Agentless (SALT/Ansible CodeDeploy? Las herramientas de despliegue basadas en agentes y sin agentes requieren que se cree e implemente una instancia independiente como servidor maestro para las herramientas. AMS espera que conozca todos los elementos necesarios para el éxito de las herramientas de despliegue de aplicaciones; sin embargo, estaremos encantados de ayudarle con las cuestiones relacionadas con la infraestructura.
- ¿Sus Line-of-Business aplicaciones (las aplicaciones que utiliza para crear y gestionar sus aplicaciones) requieren la aplicación de parches?

Dependencias de aplicaciones

¿Necesita instancias para aplicaciones Line-of-Business (LoB)? ¿Para aplicaciones de productos?

¿Qué necesitan las aplicaciones de sus productos para funcionar correctamente?

- Dependencias a nivel de red: por ejemplo, Direct Connect
- Package dependencies: por ejemplo, `pip`
- Aplicaciones de las que depende esta aplicación: por ejemplo, MySql
- ¿Dependencias del firewall?

¿Qué necesitan sus aplicaciones de LoB para funcionar correctamente?

- Dependencias a nivel de red: por ejemplo, Direct Connect
- Package dependencies: por ejemplo, Firefox Saucy
- Aplicaciones de las que depende esta aplicación: por ejemplo, MySql
- ¿Dependencias del firewall?

Certificados SSL para aplicaciones de productos

¿Qué certificados SSL necesitarán sus servidores para que sus aplicaciones (LoB y producto) puedan acceder a todo lo que necesitan para funcionar y ser accesibles?

- ¿Grupo Auto Scaling?
- ¿Base de datos (Amazon RDS)?
- ¿Load Balancer?
- ¿Servidor de herramientas de despliegue?
- Firewall de aplicaciones web (AWS WAF)?
- ¿Otras instancias?

Por ejemplo, para cada una de las instancias enumeradas anteriormente, es posible que necesite los siguientes certificados:

WAF (certificado 1) - > ELB-ext (certificado 2) - > ARP (certificado 3) - > ELB-int (certificado 4) -> Sitio web (certificado 5) -> ELB-int (certificado 6) -> Servicio web (certificado 7).

Historial del documento

En la siguiente tabla se describe la documentación de esta versión de AMS.

- Versión de API: 21 de mayo de 2019
- Última actualización de la documentación: 16 de febrero de 2023

Cambio	Descripción	Enlace
Se ha eliminado el enlace TOC	Se ha eliminado el enlace al AWS glosario de TOC.	8 de agosto de 2025
Contenido actualizado: Migración de cargas de trabajo: validación previa a la ingesta de Windows	Sección actualizada para incluir los pasos detallados para usar el script de WIGs prevalidación a fin de validar que su instancia de Windows está lista para ser incorporada a su cuenta de AMS;.	Migración de cargas de trabajo: validación previa a la ingestión de Windows
Contenido actualizado, configuración de DMS	una nota importante sobre el rol requerido, dms-vpc-role.	1: grupo de subredes de AWS DMS replicación: Crear
Contenido actualizado, recursos compatibles con CFN Ingest	Añadido. OpenSearch	Recursos admitidos
Contenido actualizado, migración de cargas de trabajo	Instrucciones actualizadas para la validación previa a la ingesta.	Migración de cargas de trabajo: validación previa a la

Cambio	Descripción	Enlace
		ingestión de Windows
Contenido actualizado, CFN Ingest.	Se eliminaron los «recursos compatibles» restringidos del contenido de CFN Ingest.	CloudFormation Ingest Stack: recursos compatibles
Versiones de Windows compatibles actualizadas	Se agregó soporte para Windows Server 2022.	Imágenes de máquinas AMS Amazon (AMIs), Migración de cargas de trabajo: requisitos previos para Linux y Windows, y Migración de cargas de trabajo: validación previa a la ingestión de Windows
Contenido actualizado, Programador de recursos.	Instrucciones actualizadas para usar el CT de despliegue dedicado, ct-0ywnhc8e5k9z5, aplicables tanto a SALZ como a MALZ.	Inicio rápido de AMS Resource Scheduler

Cambio	Descripción	Enlace
Contenido actualizado, Workload Ingest.	Se han actualizado las versiones compatibles de SUSE Linux.	Migración de cargas de trabajo: requisitos previos para Linux y Windows
Contenido actualizado, Database Migration Service.	Se agregó a los requisitos previos y se realizaron varios cambios para mayor utilidad y facilidad de uso.	AWS Database Migration Service (AWS DMS)
Contenido actualizado, Workload Ingest.	Se ha actualizado el archivo zip de validación anterior al WIGS de Linux.	Migración de cargas de trabajo: requisitos previos para Linux y Windows
Contenido actualizado.	Se actualizó el archivo zip de validación anterior a WIGS para Linux. Además, se agregó Windows Server 2008 R2 como sistema operativo compatible.	Migración de cargas de trabajo: requisitos previos para Linux y Windows
Contenido nuevo	Los inicios rápidos y los tutoriales se han trasladado aquí desde la Guía avanzada de administración de cambios de AMS, que ya no estaba disponible.	Arranques rápidos, Tutoriales.

Cambio	Descripción	Enlace
Contenido actualizado	Despliegue | Componentes de pila avanzados | Database Migration Service (DMS) | Iniciar la tarea de replicación (ct-1yq7hhqse71yg) Se actualizó para indicar que los parámetros DocumentNamey la región son obligatorios; anteriormente, aparecían erróneamente como opcionales.	Database Migration Service (DMS) | Iniciar la tarea de replicación
Contenido actualizado	CloudFormation Ingerir Se actualizó para indicar dos nuevos recursos compatibles AWS::Route53Resolver::ResolverRuleAssociation y AWS::Route53Resolver::ResolverRule.	Recursos admitidos
Contenido actualizado	Migración de cargas de trabajo: validación previa a la ingestión de Windows	La información de Sysprep se actualizó con más detalles. Migración de cargas de trabajo: validación previa a la ingestión de Windows

Cambio	Descripción	Enlace
Contenido actualizado	Administración | Pila personalizada | Apila a partir de una CloudFormation plantilla | Aprobar el conjunto de cambios y actualizar (ct-1404e 21baa2ox) La descripción detallada del parámetro en CT se ha actualizado con información adicional. ChangeSetName	Apilar a partir de CloudFormation una plantilla | Aprobar el conjunto de cambios y actualizar
	Están disponibles Ubuntu 18.04 y Oracle Linux 8.3	Migración de cargas de trabajo: requisitos previos para Linux y Windows
Contenido nuevo:	Implementaciones de IAM a través de CFN Ingest y Stack Update. CTs	10 de febrero de 2022
Tareas de replicación del Database Migration Service (DMS)	Los tipos de cambios se actualizaron para que las expresiones regulares permitan realizar tareas ARNs que contengan guiones. Inicie la tarea AWS DMS de replicacióny Database Migration Service (DMS) | Detenga la tarea de replicación.	13 de enero de 2022
Validación previa a la ingestión de Linux WIGS	Se actualizó el archivo zip. Migración de cargas de trabajo: validación previa a la ingesta de Linux.	13 de enero de 2022

Cambio	Descripción	Enlace
Enlaces fijos	La Configuración sección Importación de bases de datos (DB) a AMS SQL RDS -> tenía algunos enlaces defectuosos.	13 de enero de 2022

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.