



Migración de grandes bases de datos MySQL o MariaDB de varios terabytes a
AWS

AWS Orientación prescriptiva



AWS Orientación prescriptiva: Migración de grandes bases de datos MySQL o MariaDB de varios terabytes a AWS

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

Introducción	1
Destinatarios previstos	2
Resultados empresariales específicos	2
Opciones de migración	3
Percona XtraBackup	4
Ventajas	6
Limitaciones	7
Prácticas recomendadas	8
MyDumper	8
Ventajas	11
Limitaciones	11
Prácticas recomendadas	11
mysqldump y mysqlpump	12
Ventajas	15
Limitaciones	15
Prácticas recomendadas	16
Copia de seguridad dividida	16
Amazon S3 File Gateway	19
Ventajas	20
Limitaciones	20
Prácticas recomendadas	21
Prácticas recomendadas	22
Recursos	24
Historial de documentos	26
.....	xxvii

Migración de grandes bases de datos MySQL o MariaDB de varios terabytes a AWS

Babaiah Valluru y Ankur Bhanawat, Amazon Web Services (AWS)

[Noviembre de](#) 2024 (historial del documento)

Muchas organizaciones que tienen servidores de bases de datos de MySQL y MariaDB en las instalaciones están interesadas en migrar sus cargas de trabajo de bases de datos a la Nube de AWS. Muchas eligen Amazon Relational Database Service (Amazon RDS) para MariaDB, Amazon RDS para MySQL o la edición compatible con MySQL de Amazon Aurora [Amazon RDS](#) se diseñó para facilitar la configuración, el funcionamiento y el escalado de bases de datos relacionales en la nube. [Amazon Aurora](#) forma parte de Amazon RDS y ofrece seguridad integrada, copias de seguridad continuas, computación sin servidores, hasta 15 réplicas de lectura, replicación multirregional automatizada e integración con otros Servicios de AWS.

Si bien la migración a uno de estos Servicios de AWS puede ofrecer muchos beneficios, la migración de bases de datos es una de las tareas críticas y que más tiempo requiere y que los administradores de bases de datos deben realizar. Para la migración de grandes bases de datos es necesaria una planificación e implementación precisas y garantizar que el rendimiento de la carga de trabajo migrada sea equivalente o mejorado. En esta guía, las bases de datos grandes pueden hacer referencia a una única base de datos de varios terabytes o a muchas bases de datos grandes que suman varios terabytes de datos. La selección de los servicios y de las herramientas de migración adecuados es fundamental para que la migración sea correcta. Existen dos enfoques comunes para migrar una base de datos: lógico y físico. Para más información sobre estos enfoques, consulte la documentación de [MySQL](#) y [MariaDB](#).

En esta guía se describen diversas herramientas de código abierto o de terceros que puede utilizar para migrar las bases de datos de MySQL y MariaDB grandes, en las instalaciones y de varios terabytes a Amazon RDS para MariaDB, Amazon RDS para MySQL o la edición compatible con MySQL de Amazon Aurora. Las opciones que se analizan en esta guía utilizan enfoques de migración lógicos o físicos. Cada opción incluye varios enfoques para transferir los archivos de copia de seguridad de las bases de datos grandes desde el centro de datos en las instalaciones a la nube, donde puede restaurar la base de datos desde el archivo de la copia de seguridad.

Destinatarios previstos

Esta guía está dirigida a los administradores bases de datos, ingenieros de bases de datos, gerentes de proyectos, y gerentes de operaciones o infraestructura que tienen previsto migrar sus bases de datos de MySQL o MariaDB a la Nube de AWS.

Resultados empresariales específicos

El objetivo de esta guía es ayudar a hacer lo siguiente:

- Elija el enfoque de migración de una base de datos grande que mejor se adapte a su caso de uso y entorno.
- Evite los retrasos y las pérdidas económicas que pueden producirse cuando la estrategia de migración no es la adecuada.
- Información acerca de las ventajas y limitaciones de cada opción de migración.
- Información de los distintos enfoques que puede utilizar para transferir archivos de copia de seguridad de bases de datos grandes desde su centro de datos en las instalaciones a la Nube de AWS.
- Revise las prácticas recomendadas generales para migrar las bases de datos grandes y revise también las prácticas recomendadas para cada herramienta, lo que puede serle útil para migrar la base de datos de manera más eficiente.

Opciones de migración de bases de datos de MySQL y MariaDB grandes

Puede elegir entre una amplia gama de opciones para migrar de bases de datos MySQL o MariaDB locales a instancias de bases de datos Amazon Relational Database Service (Amazon RDS) o Amazon Aurora Edition. MySQL-Compatible Elegir el enfoque y la herramienta de migración correctos es esencial para que la migración sea correcta. En esta guía, se evaluarán las opciones según sus requisitos de usabilidad, tamaño de datos y tiempo de inactividad.

A continuación, se muestran los enfoques y herramientas de migración más comunes que están disponibles para migrar las bases de datos de MySQL autoadministradas de varios terabytes de manera eficiente a las instancias de bases de datos de Amazon RDS, Aurora o Amazon Elastic Compute Cloud (Amazon EC2):

- [Percona XtraBackup](#) (física)
- [MyDumper](#) (lógica)
- [mysqldump y mysqlpump](#) (lógica)
- [Copia de seguridad dividida](#) (físicas, lógicas o ambas)

Los siguientes son los enfoques y herramientas de migración más comunes que están disponibles para migrar bases de datos de varios terabytes MySQL-compatible (como MariaDB) de manera eficiente a instancias de bases de datos de Amazon RDS, Aurora o Amazon EC2:

- [MyDumper](#) (lógica)
- [mysqldump y mysqlpump](#) (lógica)
- [Copia de seguridad dividida](#) (físicas, lógicas o ambas)

Para cada herramienta de migración, hay varios enfoques que puede utilizar para transferir el archivo de copia de seguridad de la base de datos grande a la Nube de AWS. Se proporcionan opciones para cada herramienta y también puede utilizar Amazon S3 File Gateway. Para obtener más información, consulte la sección [Uso de Amazon S3 File Gateway para transferir archivos de copia de seguridad](#) de esta guía.

Percona XtraBackup

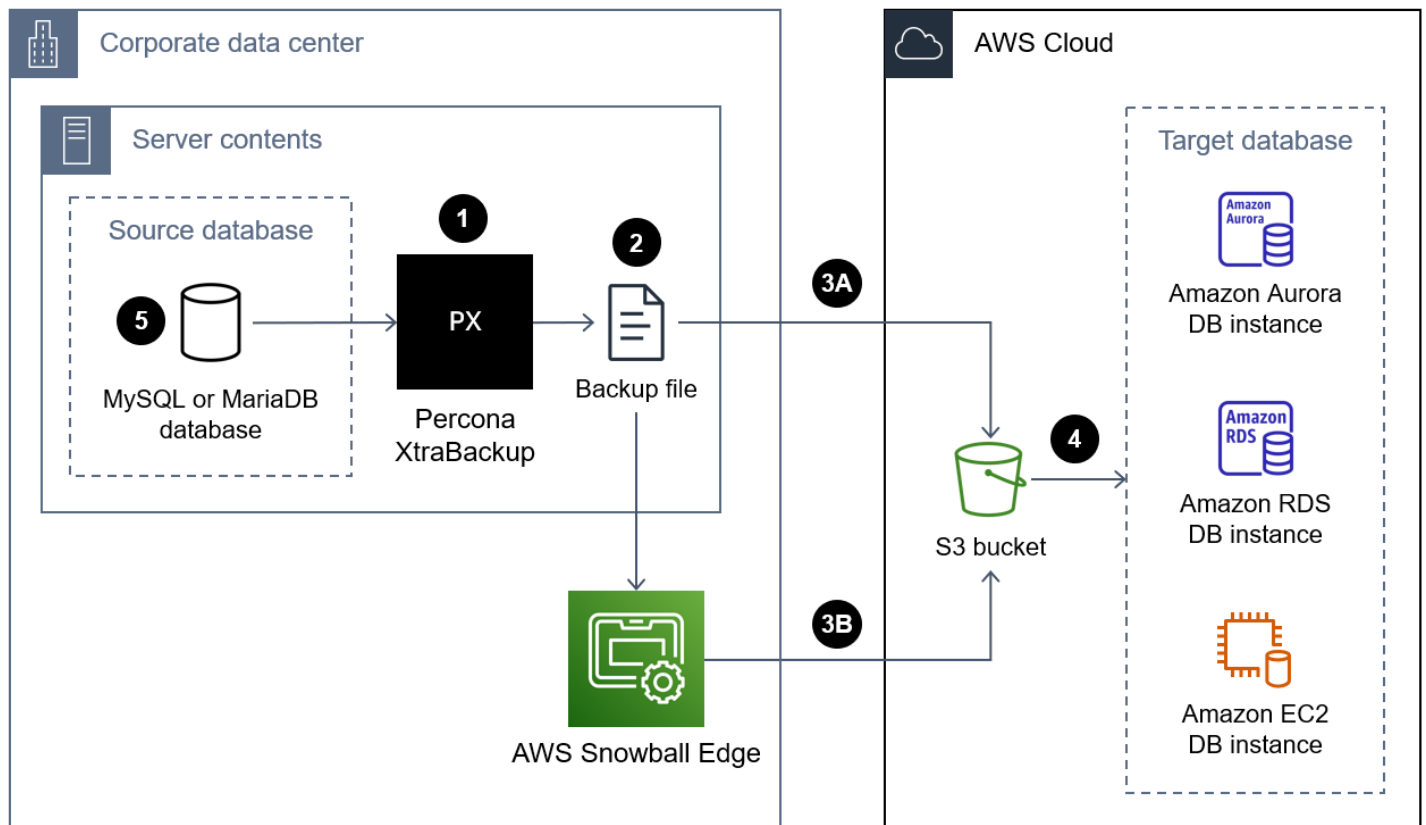
Important

Percona no XtraBackup es compatible con las versiones 10.3 o posteriores de MariaDB y solo es compatible parcialmente con las versiones 10.1 y 10.2.

[Percona XtraBackup](#) es un software común de copia de seguridad en caliente de código abierto para MySQL y MariaDB que realiza copias de seguridad sin bloqueo para los motores de almacenamiento InnoDB y XtraDB. Funciona con servidores MySQL o MariaDB. [Para obtener más información sobre la herramienta y algunas de sus características y ventajas, consulte Acerca de Percona en la documentación de Percona. XtraBackup](#) XtraBackup

Esta herramienta utiliza el enfoque de migración física. Copia de manera directa el directorio de datos de MySQL o MariaDB y los archivos que contiene. En el caso de las bases de datos grandes, como las de más de 100 GB, esto puede proporcionar un tiempo de restauración significativamente mejor que el de otras herramientas. Se crea una copia de seguridad de la base de datos de origen en las instalaciones, se migran los archivos de la copia de seguridad a la nube y, a continuación, se restaura la copia de seguridad en la nueva instancia de base de datos de destino.

El siguiente diagrama muestra los pasos de alto nivel necesarios para migrar una base de datos mediante un archivo de respaldo de XtraBackup Percona. Según el tamaño del archivo de copia de seguridad, hay dos opciones disponibles para transferir la copia de seguridad a un bucket de Amazon Simple Storage Service (Amazon S3) en la Nube de AWS.



Los siguientes son los pasos para usar Percona para XtraBackup migrar una base de datos a: Nube de AWS

1. Instale Percona XtraBackup en el servidor local. Si utiliza Amazon Aurora MySQL versión 2 o Amazon RDS, consulte [Instalación de Percona 2.4 XtraBackup](#). Si utiliza Amazon Aurora MySQL versión 3, consulte Instalación de [Percona XtraBackup 8.0](#) en la documentación de Percona XtraBackup.
2. Cree una copia de seguridad completa de la base de datos de MySQL o MariaDB de origen. [Para obtener instrucciones sobre Percona XtraBackup 2.4, consulte Copia de seguridad completa.](#) Para obtener instrucciones sobre Percona XtraBackup 8.0, consulte [Crear una copia de seguridad completa.](#)
3. Transfiera los archivos de respaldo a través de Internet mediante un servicio o una herramienta aprobados de su organización, como los siguientes:
 - [AWS Site-to-Site VPN](#)
 - [AWS Client VPN](#)
 - [AWS Direct Connect](#)

- [Amazon S3 File Gateway](#) (para más información, consulte [Uso de Amazon S3 File Gateway para transferir archivos de copia de seguridad](#) en esta guía.)
 - [AWS Command Line Interface \(AWS CLI\)](#)
4. Desde el bucket de Amazon S3, restaure los archivos de respaldo en la instancia de base de datos de destino. Para obtener instrucciones, consulte lo siguiente:
- Para Aurora MySQL-Compatible Edition, consulte [Migración de datos desde MySQL mediante un bucket de Amazon S3](#) en la documentación de Amazon RDS.
 - En el caso de Amazon RDS para MySQL o Amazon EC2, consulte [Importación de datos en una instancia de base de datos de MySQL](#).
 - En el caso de Amazon RDS para MariaDB o Amazon EC2, consulte [Importación de datos a una instancia de base de datos de MariaDB](#).
5. (Opcional) Puede configurar la replicación entre la base de datos de origen y la instancia de base de datos de destino. Puede utilizar la replicación del registro binario (binlog) para reducir el tiempo de inactividad. Para obtener más información, consulte los siguientes temas:
- [Setting the replication source configuration](#) en la documentación de MySQL
 - Para Amazon Aurora, consulte lo siguiente:
 - [Sincronización del clúster de base de datos de Amazon Aurora MySQL con la base de datos MySQL mediante replicación](#) en la documentación de Aurora
 - [Uso de la replicación de registro binario en Amazon Aurora](#) en la documentación de Aurora
 - Para Amazon RDS, consulte lo siguiente:
 - [Uso de la replicación de MySQL](#) en la documentación de Amazon RDS
 - [Uso de la replicación de MariaDB](#) en la documentación de Amazon RDS
 - Para Amazon EC2, consulte lo siguiente:
 - [Setting Up Binary Log File Position Based Replication](#) en la documentación de MySQL
 - [Setting Up Replicas](#) en la documentación de MySQL
 - [Setting Up Replication](#) en la documentación de MariaDB

Ventajas

- Como Percona XtraBackup utiliza un enfoque de migración física, el proceso de restauración suele ser más rápido que el de las herramientas que utilizan un enfoque de migración lógica. Esto se

debe a que el rendimiento está limitado por el rendimiento del disco o la red y no por los recursos de computación necesarios para el procesamiento de datos.

- Como el proceso de restauración consiste en una copia directa de los archivos del bucket de S3 a la instancia de base de datos de destino, los XtraBackup archivos de Percona suelen restaurarse más rápido que los archivos de backup creados con otras herramientas.
- Percona XtraBackup es adaptable. Por ejemplo, admite varios subprocesos para copiar archivos más rápido y admite la compresión para reducir el tamaño de la copia de seguridad.

Limitaciones

- La copia de seguridad sin conexión no es posible porque Percona XtraBackup debe tener acceso al servidor de la base de datos de origen.
- Percona solo se XtraBackup puede usar en sistemas con arquitecturas de sistema idénticas. Por ejemplo, no es posible restaurar una copia de seguridad de una base de datos de origen que se ejecuta en un servidor Intel para Windows en un servidor de destino ARM para Linux.
- Percona XtraBackup no es compatible con la versión 10.3 o posterior de MariaDB, y solo es compatible parcialmente con las versiones 10.2 y 10.1 de MariaDB. Para obtener más información, consulte [XtraBackup Descripción general de Percona: compatibilidad con MariaDB](#) en la base de conocimiento de MariaDB.
- No puede usar Percona XtraBackup para restaurar una base de datos MariaDB de origen en una instancia de base de datos MySQL de destino, como Amazon RDS for MySQL o Aurora. MySQL-Compatible
- El volumen total de datos y la cantidad de objetos que puede almacenar en un bucket de S3 son ilimitados. Sin embargo, el tamaño máximo del archivo es de 5 TB. Si el archivo de copia de seguridad supera los 5 TB, puede dividirlo en varios archivos más pequeños.
- Cuando la `innodb_file_per_table` configuración está desactivada, Percona XtraBackup no admite copias de seguridad parciales que utilicen `--tables`, `--tables-exclude`, `--tables-file` o `--databases` `--databases-exclude` `--databases-file` Para obtener más información sobre la XtraBackup versión 2.4 de Percona, consulte Copias de seguridad [parciales](#). Para obtener más información sobre la XtraBackup versión 8.0 de Percona, consulte [Crear una copia de seguridad parcial](#).

Prácticas recomendadas

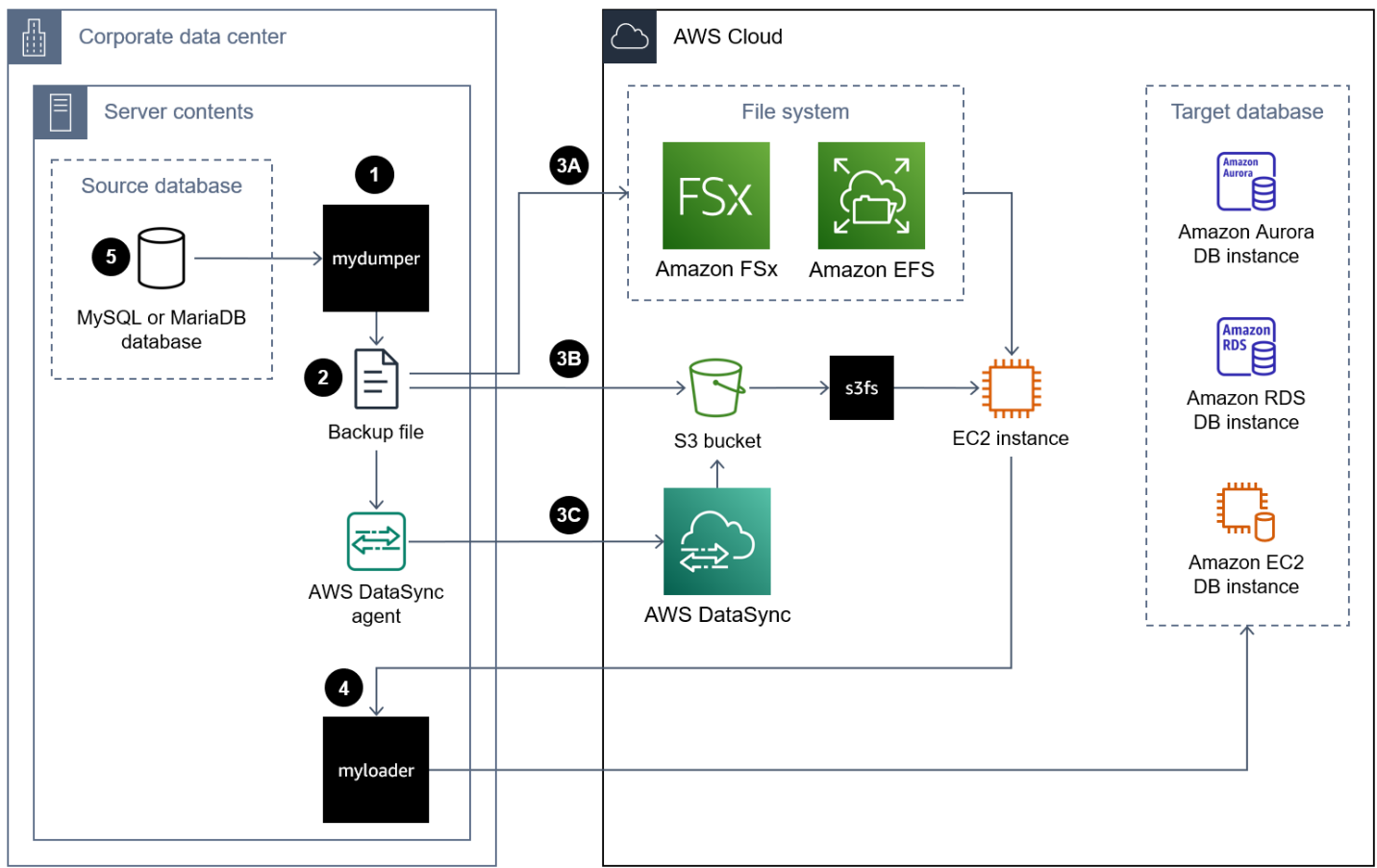
- Para mejorar el rendimiento del proceso de copia de seguridad, haga lo siguiente:
 - Copie varios archivos en paralelo mediante [`--parallel=<threads>`](#)
 - Comprima varios archivos en paralelo mediante [`--compress-threads=<threads>`](#)
 - Aumente la memoria mediante [`--use-memory=<size>`](#)
 - Cifre varios archivos en paralelo mediante [`--encrypt-threads=<threads>`](#)
- Asegúrese de que haya suficiente espacio suficiente en el servidor de origen para guardar los archivos de copia de seguridad de la base de datos.
- Genere la copia de seguridad de la base de datos con el archivo con formato (.xbstream) de Percona xstream. Para obtener más información, consulte [la descripción general del binario xstream](#) en la documentación de Percona. XtraBackup

MyDumper

[MyDumper](#) (GitHub) es una herramienta de migración lógica de código abierto que consta de dos utilidades:

- MyDumper exporta una copia de seguridad coherente de las bases de datos MySQL. Permite hacer copias de seguridad de la base de datos mediante el uso de varios subprocesos paralelos, hasta un subproceso por núcleo de CPU disponible.
- myloader lee los archivos de respaldo creados por MyDumper, se conecta a la instancia de base de datos de destino y, a continuación, restaura la base de datos.

El siguiente diagrama muestra los pasos de alto nivel necesarios para migrar una base de datos mediante un archivo de MyDumper respaldo. En este diagrama de arquitectura se incluyen tres opciones para migrar el archivo de copia de seguridad del centro de datos en las instalaciones a una instancia de EC2 en la Nube de AWS.



Los siguientes son los pasos que se deben seguir para MyDumper migrar una base de datos a: Nube de AWS

1. Install MyDumper y myloader. Para obtener instrucciones, consulte [Cómo instalar mydumper/myloader](#) (GitHub).
2. Se utiliza MyDumper para crear una copia de seguridad de la base de datos MySQL o MariaDB de origen. Para obtener instrucciones, consulte [Cómo usarlo](#). MyDumper
3. Mueva el archivo de respaldo a una instancia EC2 de la siguiente manera: Nube de AWS

Enfoque 3A: monte un sistema de archivos de [Amazon FSx](#) o [Amazon Elastic File System \(Amazon EFS\)](#) en el servidor en las instalaciones que ejecuta la instancia de base de datos. Puede usar AWS Direct Connect o Site-to-Site VPN establecer la conexión. Puede hacer una copia de seguridad de la base de datos de manera directa en el recurso compartido de archivos montado o puede hacer la copia de seguridad en dos pasos: haga una copia de seguridad de la base de datos en un sistema de archivos local y, a continuación, cárguela en el volumen de FSx o

EFS montado. A continuación, monte el sistema de archivos de Amazon FSx o Amazon EFS, que también está montado en el servidor en las instalaciones, en una instancia de EC2.

Método 3B: utilice el AWS CLI AWS SDK o la API REST de Amazon S3 para mover directamente el archivo de respaldo del servidor local a un bucket de S3. Si el depósito S3 de destino se encuentra en un Región de AWS lugar alejado del centro de datos, puede utilizar [Amazon S3 Transfer Acceleration](#) para transferir el archivo con mayor rapidez. Utilice el sistema de archivos [s3fs-fuse](#) para montar el bucket S3 en la instancia de EC2.

Enfoque 3C: instale el agente de AWS DataSync en el centro de datos en las instalaciones y utilice [AWS DataSync](#) para mover el archivo de copia de seguridad a un bucket de Amazon S3. Utilice el sistema de archivos [s3fs-fuse](#) para montar el bucket S3 en la instancia de EC2.

 Note

También puede utilizar Amazon S3 File Gateway para transferir los archivos de copia de seguridad de una base de datos grande a un bucket de S3 en la Nube de AWS. Para obtener más información, consulte la sección [Uso de Amazon S3 File Gateway para transferir archivos de copia de seguridad](#) de esta guía.

4. Utilice myloader para restaurar la copia de seguridad en la instancia de base de datos de destino. Para obtener instrucciones, consulte [myloader usage](#) (GitHub).
5. (Opcional) Puede configurar la replicación entre la base de datos de origen y la instancia de base de datos de destino. Puede utilizar la replicación del registro binario (binlog) para reducir el tiempo de inactividad. Para obtener más información, consulte los siguientes temas:
 - [Setting the replication source configuration](#) en la documentación de MySQL
 - Para Amazon Aurora, consulte lo siguiente:
 - [Sincronización del clúster de base de datos de Amazon Aurora MySQL con la base de datos MySQL mediante replicación](#) en la documentación de Aurora
 - [Uso de la replicación de registro binario en Amazon Aurora](#) en la documentación de Aurora
 - Para Amazon RDS, consulte lo siguiente:
 - [Uso de la replicación de MySQL](#) en la documentación de Amazon RDS
 - [Uso de la replicación de MariaDB](#) en la documentación de Amazon RDS
 - Para Amazon EC2, consulte lo siguiente:
 - [Setting Up Binary Log File Position Based Replication](#) en la documentación de MySQL

- [Setting Up Replicas](#) en la documentación de MySQL
- [Setting Up Replication](#) en la documentación de MariaDB

Ventajas

- MyDumper admite el paralelismo mediante el uso de subprocesos múltiples, lo que mejora la velocidad de las operaciones de copia de seguridad y restauración.
- MyDumper evita costosas rutinas de conversión de conjuntos de caracteres, lo que ayuda a garantizar que el código sea altamente eficiente.
- MyDumper simplifica la visualización y el análisis de los datos mediante la descarga de archivos separados para las tablas y los metadatos.
- MyDumper mantiene instantáneas en todos los subprocesos y proporciona posiciones precisas de los registros principales y secundarios.
- Puede utilizar las expresiones regulares compatibles con Perl (PCRE) para especificar si quiere incluir o excluir las tablas o las bases de datos.

Limitaciones

- Puede elegir otra herramienta si los procesos de transformación de datos requieren archivos de volcado intermedios en formato plano en lugar de formato SQL.
- myloader no importa de manera automática las cuentas de usuario de la base de datos. Si va a restaurar la copia de seguridad en Amazon RDS o Aurora, vuelva a crear los usuarios con los permisos necesarios. Para más información, consulte [Privilegios de la cuenta de usuario maestro](#) en la documentación de Amazon RDS. Si va a restaurar la copia de seguridad en una instancia de base de datos de Amazon EC2, puede exportar de manera manual las cuentas de usuario de la base de datos de origen e importarlás a la instancia de EC2.

Prácticas recomendadas

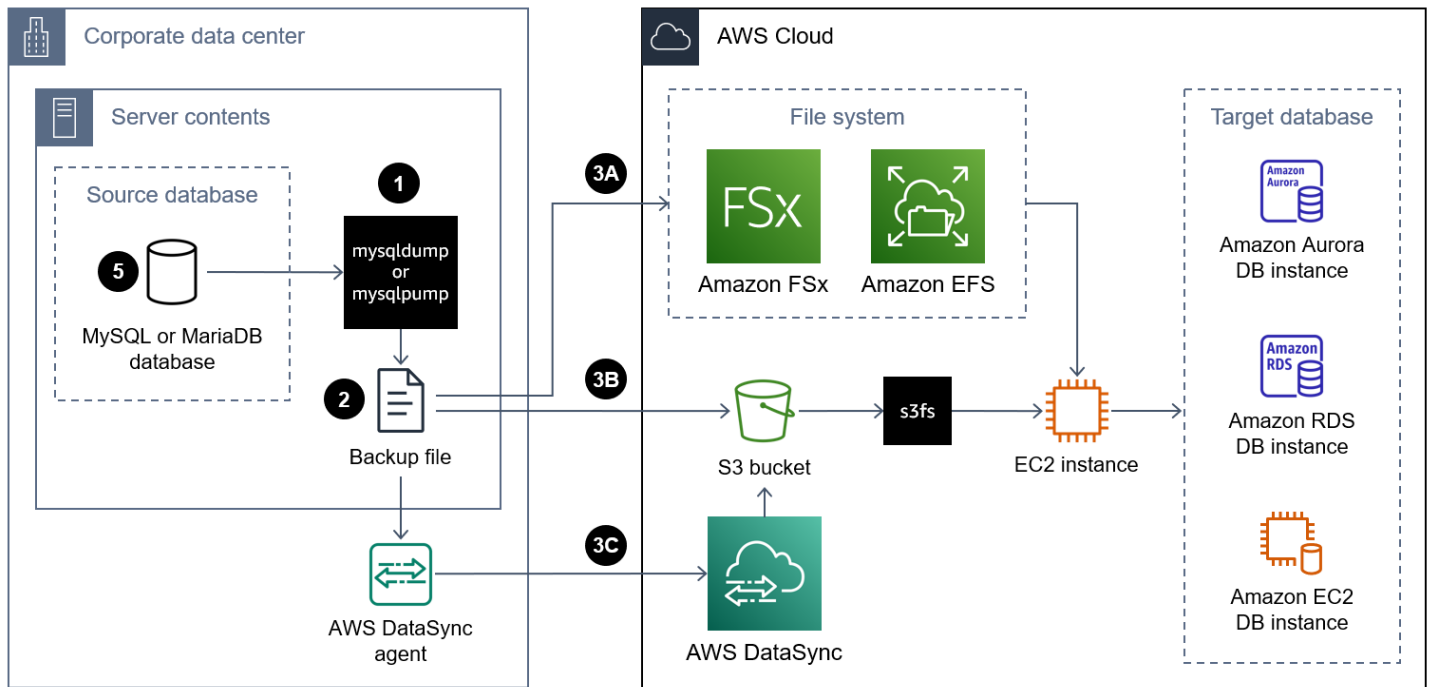
- Configure MyDumper para dividir cada tabla en segmentos, por ejemplo, 10 000 filas en cada segmento, y escriba cada segmento en un archivo independiente. Esto permite importar los datos en paralelo más tarde.

- Si utiliza el motor de InnoDB, utilice la opción `--trx-consistency-only` para minimizar el bloqueo.
- El uso de la base de datos MyDumper para exportar puede resultar intensivo en lectura y el proceso puede afectar al rendimiento general de la base de datos de producción. Si tiene una instancia de base de datos de réplica, ejecute el proceso de exportación desde la réplica. Antes de ejecutar la exportación desde la réplica, detenga el subproceso de SQL de replicación. De este modo, el proceso de exportación se ejecute con mayor rapidez.
- No exporte la base de datos durante las horas de mayor actividad. Evitar las horas de mayor actividad puede estabilizar el rendimiento de la base de datos de producción principal durante la exportación de la base de datos.
- Amazon RDS para MySQL no admite el complemento `keyring_aws`. Para más información, consulte [Known issues and limitations](#). Para migrar las tablas cifradas en las instalaciones a la instancia de Amazon RDS, en los scripts de copia de seguridad, debe eliminar `ENCRYPTION` o `DEFAULT ENCRYPTION` de la sintaxis `CREATE TABLE`. Para el cifrado en reposo, puede utilizar una clave AWS Key Management Service (AWS KMS). Para obtener más información, consulte [Cifrado de recursos de Amazon RDS](#).

mysqldump y mysqlpump

[mysqldump](#) y [mysqlpump](#) son herramientas de copia de seguridad de bases de datos nativas para MySQL. MariaDB es compatible con `mysqldump`, pero no con `mysqlpump`. Ambas herramientas crean copias de seguridad lógicas y forman parte de los programas cliente de MySQL. `mysqldump` admite el procesamiento de un solo subproceso. `mysqlpump` admite el procesamiento paralelo de bases de datos y objetos en las bases de datos para acelerar el proceso de volcado. Se introdujo en la versión 5.7.8 de MySQL. `mysqlpump` se eliminó en la versión 8.4 de MySQL.

En el diagrama siguiente se muestran los pasos generales que implica migrar una base de datos con un archivo de copia de seguridad de `mysqldump` o `mysqlpump`.



A continuación, se muestran los pasos para utilizar mysqldump o mysqlpump para migrar una base de datos a la Nube de AWS.

1. Instale MySQL Shell en el servidor en las instalaciones. Para ver las instrucciones, consulte [Installing MySQL Shell](#) en la documentación de MySQL. Esto instala mysqldump y mysqlpump.
2. Con mysqldump o mysqlpump, cree una copia de seguridad de la base de datos de origen en las instalaciones. Para obtener instrucciones, consulte [mysqldump](#) and [mysqlpump](#) en la documentación de MySQL, o consulte [Making Backups with mysqldump](#) en la documentación de MariaDB. Para más información sobre la invocación de programas de MySQL y la especificación de opciones, consulte [Using MySQL programs](#).
3. Mueva el archivo de respaldo a una instancia EC2 de la siguiente manera: Nube de AWS

Método 3A: monte un sistema de archivos [Amazon FSx](#) o [Amazon Elastic File System \(Amazon EFS\)](#) en el servidor local que ejecuta la instancia de base de datos. Puede usar AWS Direct Connect o establecer Site-to-Site VPN la conexión. Puede hacer una copia de seguridad de la base de datos de manera directa en el recurso compartido de archivos montado o puede hacer la copia de seguridad en dos pasos: haga una copia de seguridad de la base de datos en un sistema de archivos local y, a continuación, cárguela en el volumen de FSx o EFS montado. A continuación, monte el sistema de archivos de Amazon FSx o Amazon EFS, que también está montado en el servidor en las instalaciones, en una instancia de EC2.

Método 3B: utilice el AWS CLI AWS SDK o la API REST de Amazon S3 para mover directamente el archivo de respaldo del servidor local a un bucket de S3. Si el depósito S3 de destino se encuentra en un Región de AWS lugar alejado del centro de datos, puede utilizar [Amazon S3 Transfer Acceleration](#) para transferir el archivo con mayor rapidez. Utilice el sistema de archivos [s3fs-fuse](#) para montar el bucket S3 en la instancia de EC2.

Enfoque 3C: instale el agente de AWS DataSync en el centro de datos en las instalaciones y utilice [AWS DataSync](#) para mover el archivo de copia de seguridad a un bucket de Amazon S3. Utilice el sistema de archivos [s3fs-fuse](#) para montar el bucket S3 en la instancia de EC2.

 Note

También puede utilizar Amazon S3 File Gateway para transferir los archivos de copia de seguridad de una base de datos grande a un bucket de S3 en la Nube de AWS. Para obtener más información, consulte la sección [Uso de Amazon S3 File Gateway para transferir archivos de copia de seguridad](#) de esta guía.

4. Utilice el método de restauración nativo para restaurar la copia de seguridad en la base de datos de destino. Para obtener instrucciones, consulte [Recargar SQL-Format copias de seguridad](#) en la documentación de MySQL o consulte [Restauración de datos de archivos de volcado](#) en la documentación de MariaDB.
5. (Opcional) Puede configurar la replicación entre la base de datos de origen y la instancia de base de datos de destino. Puede utilizar la replicación del registro binario (binlog) para reducir el tiempo de inactividad. Para obtener más información, consulte los siguientes temas:
 - [Setting the replication source configuration](#) en la documentación de MySQL
 - Para Amazon Aurora, consulte lo siguiente:
 - [Sincronización del clúster de base de datos de Amazon Aurora MySQL con la base de datos MySQL mediante replicación](#) en la documentación de Aurora
 - [Uso de la replicación de registro binario en Amazon Aurora](#) en la documentación de Aurora
 - Para Amazon RDS, consulte lo siguiente:
 - [Uso de la replicación de MySQL](#) en la documentación de Amazon RDS
 - [Uso de la replicación de MariaDB](#) en la documentación de Amazon RDS
 - Para Amazon EC2, consulte lo siguiente:
 - [Setting Up Binary Log File Position Based Replication](#) en la documentación de MySQL

- [Setting Up Replicas](#) en la documentación de MySQL
- [Setting Up Replication](#) en la documentación de MariaDB

Ventajas

- mysqldump y mysqlpump se incluyen en la instalación de MySQL Server
- Los archivos de copia de seguridad que generan estas herramientas están en un formato más legible.
- Antes de restaurar el archivo de copia de seguridad, puede modificar el archivo .sql resultante mediante un editor de texto estándar.
- Puede hacer una copia de seguridad de una tabla, base de datos o incluso de una selección de datos concreta.
- mysqldump y mysqlpump son independientes de la arquitectura de la máquina.

Limitaciones

- mysqldump es un proceso de copias de seguridad de un solo subprocesso. El rendimiento al hacer una copia de seguridad es bueno para bases de datos pequeñas, pero puede resultar ineficiente cuando el tamaño de la copia de seguridad es superior a 10 GB.
- Los archivos de copia de seguridad en formato lógico son voluminosos, sobre todo cuando se guardan como texto, y suelen tardar en crearse y restaurarse.
- La restauración de los datos puede resultar lenta porque volver a aplicar sentencias SQL en la instancia de base de datos de destino implica un procesamiento intensivo del disco I/O y la CPU para su inserción, creación de índices y aplicación de las restricciones de integridad referencial.
- La utilidad mysqlpump no es compatible con las versiones de MySQL anteriores a la 5.7.8 ni a las versiones 8.4 y posteriores.
- De manera predeterminada, mysqlpump no hace copias de seguridad de las bases de datos del sistema, por ejemplo, performance_schema o sys. Para hacer una copia de seguridad de una parte de la base de datos del sistema, asígnele un nombre explícito en la línea de comandos.
- mysqldump no hace copias de seguridad de las instrucciones CREATE TABLESPACE de InnoDB

 Note

Las copias de seguridad de las instrucciones CREATE TABLESPACE y las bases de datos del sistema solo son útiles cuando se restauran las copias de seguridad de las bases de datos de MySQL o MariaDB en una instancia de EC2. Estas copias de seguridad no se utilizan para Amazon RDS ni Aurora.

Prácticas recomendadas

- Cuando restaure la copia de seguridad de la base de datos, desactive las comprobaciones clave, por ejemplo FOREIGN_KEY_CHECKS, de la sesión en la base de datos de destino. Esto aumenta la velocidad de restauración.
- Asegúrese de que el usuario de la base de datos tenga los [privilegios](#) suficientes para crear y restaurar la copia de seguridad.

Copia de seguridad dividida

Una estrategia de división de copia de seguridad consiste en migrar un servidor de bases de datos grande al dividir la copia de seguridad en varias partes. Podría utilizar enfoques distintos para migrar cada parte de la copia de seguridad. Esta puede ser la mejor opción para los casos de uso siguientes:

- Servidor de bases de datos grandes pero bases de datos individuales pequeñas: este es un buen enfoque cuando el tamaño total del servidor de bases de datos es de varios TB, pero el tamaño de cada base de datos de usuario individual e independiente es inferior a 1 TB. Para reducir el periodo de migración general, puede migrar bases de datos individuales por separado y en paralelo.

Utilicemos un ejemplo de un servidor de base de datos en las instalaciones de 2 TB. Este servidor consta de cuatro bases de datos de 0,5 TB cada una. Puede hacer copias de seguridad de cada base de datos individual por separado. Al restaurar la copia de seguridad, puede restaurar todas las bases de datos de una instancia en paralelo o, si las bases de datos son independientes, puede restaurar cada copia de seguridad en una instancia independiente. Se recomienda restaurar las bases de datos independientes en instancias distintas, en lugar de restaurarlas en la misma instancia. Para más información, consulte la sección Prácticas recomendadas de guía.

- Servidor de bases de datos grandes pero tablas de bases de datos individuales pequeñas: este es un buen enfoque cuando el tamaño total del servidor de bases de datos es de varios TB, pero el tamaño de cada base de datos de usuario tabla de bases de datos independiente es inferior a 1 TB. Para reducir el periodo de migración general, puede migrar tablas independientes de manera individual.

Utilicemos un ejemplo de una base de datos de un solo usuario de 1 TB y que es la única base de datos de un servidor de bases de datos en las instalaciones. Hay 10 tablas en la base de datos y cada una tiene 100 GB. Puede hacer copias de seguridad de cada tabla individual por separado. Al restaurar la copia de seguridad, puede restaurar todas las tablas de una instancia en paralelo.

- Una base de datos contiene tablas de cargas de trabajo transaccionales y no transaccionales: al igual que en el caso de uso anterior, puede utilizar un enfoque de copia de seguridad dividida cuando tiene tablas de cargas de trabajo transaccionales y no transaccionales en la misma base de datos.

Utilicemos un ejemplo de una base de datos de 2 TB que consta de 0,5 TB de tablas de cargas de trabajo críticas que se utilizan para el procesamiento de transacciones en línea (OLTP) y una sola tabla de 1,5 TB que se utiliza para archivar datos antiguos. Puede hacer la copia de seguridad de todos los objetos de la base de datos, excepto la tabla de archivo, como una copia de seguridad coherente y de una sola transacción. A continuación, haga otra copia de seguridad independiente de la tabla de archivo únicamente. Para la copia de seguridad de la tabla de archivo, también puede considerar la posibilidad de hacer varias copias de seguridad paralelas con condiciones para dividir el número de filas del archivo de copia de seguridad. A continuación, se muestra un ejemplo:

```
mysqldump -p your_db1 --tables your_table1 --where="column1 between 1 and 1000000 " >
your_table1_part1.sql
mysqldump -p your_db1 --tables your_table1 --where="column1 between 1000001 and
2000000 " > your_table1_part2.sql
mysqldump -p your_db1 --tables your_table1 --where="column1 > 2000000 " >
your_table1_part3.sql
```

Al restaurar los archivos de copia de seguridad, puede restaurar la copia de seguridad de la carga de trabajo transaccional y la copia de seguridad de la tabla de archivo en paralelo.

- Limitaciones de recursos informáticos: si tiene recursos informáticos limitados en el servidor local, como la CPU, la memoria o el disco I/O, esto puede afectar a la estabilidad y el rendimiento al

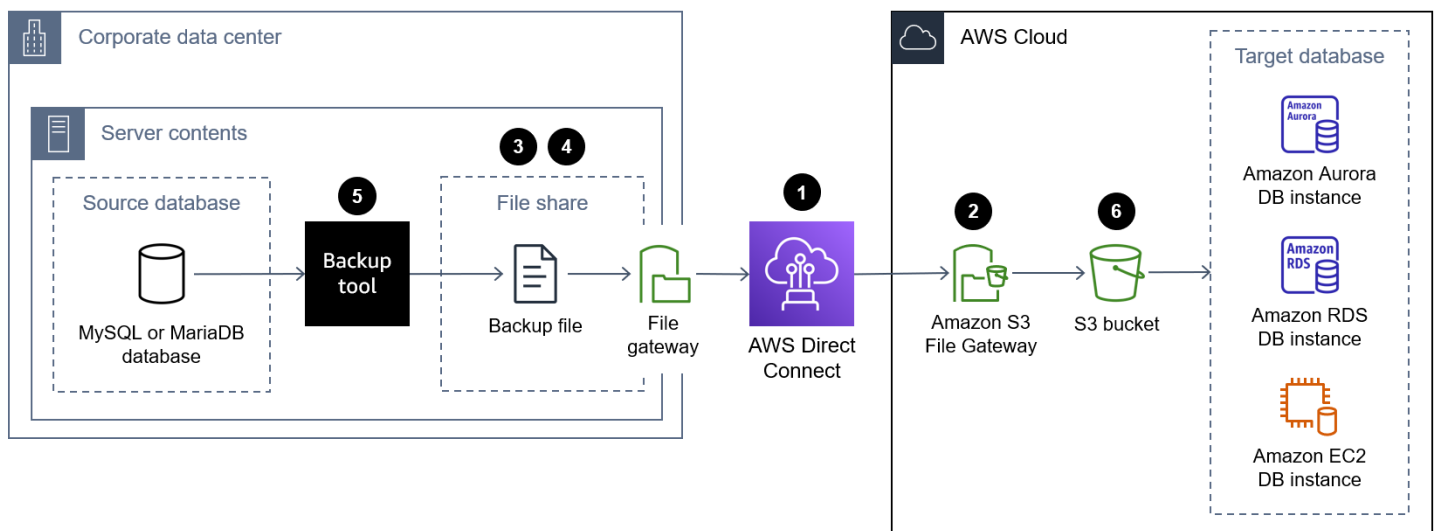
realizar la copia de seguridad. En lugar de hacer una copia de seguridad completa, puede dividirla en partes.

Por ejemplo, un servidor de producción en las instalaciones puede estar muy cargado de cargas de trabajo y tener recursos de CPU limitados. Si hace una copia de seguridad de una sola ejecución de una base de datos de varios terabytes en este servidor, puede consumir recursos de CPU adicionales y afectar de manera negativa al servidor de producción. En lugar de hacer la copia de seguridad completa de la base de datos, divida la copia de seguridad en varias partes, por ejemplo, de 2 a 3 tablas cada una.

Uso de Amazon S3 File Gateway para transferir archivos de copia de seguridad

[Amazon S3 File Gateway](#) conecta el entorno en las instalaciones con Amazon Simple Storage Service (Amazon S3) a través de una interfaz de archivos para que pueda almacenar y recuperar objetos de Amazon S3 mediante protocolos de archivos estándar del sector, como Network File System (NFS) y bloque de mensajes de servidor (SMB). Está diseñado para ser una solución rentable y escalable para almacenar datos en la nube. Como puede utilizarlo para almacenar archivos de copia de seguridad de bases de datos, este servicio puede serle útil para migrar bases de datos grandes en las instalaciones a la Nube de AWS. Por ejemplo, podría utilizar Amazon S3 File Gateway y la herramienta de copia de seguridad de bases de datos que prefiera para hacer una copia de seguridad de la base de datos grande de MySQL o MariaDB de manera directa en un bucket de Amazon S3. A continuación, puede montar el bucket de S3 en la instancia de destino y restaurar la copia de seguridad.

En el diagrama siguiente se muestran los pasos generales necesarios para utilizar Amazon S3 File Gateway para transferir el archivo de copia de seguridad de una base de datos en las instalaciones a un bucket de S3 de la Nube de AWS.



A continuación, se muestran los pasos para utilizar Amazon S3 File Gateway para transferir un archivo de copia de seguridad de base de datos desde un centro de datos en las instalaciones a un bucket de S3 en la Nube de AWS:

1. Conecte el centro de datos local al Nube de AWS mediante un servicio como, por ejemplo, AWS Direct Connect AWS Site-to-Site VPN o mediante una conexión pública a Internet.
2. Cree una puerta de enlace de archivo de S3. Para obtener instrucciones, consulte [Creating your gateway](#).
3. Cree un recurso compartido de archivos NFS o SMB alojado en la puerta de enlace de archivo de S3. Para obtener instrucciones, consulte [Create a file share](#).
4. Monte el recurso compartido de archivos NFS o SMB en el servidor en las instalaciones que aloja la base de datos de MySQL o MariaDB. Para obtener instrucciones, consulte [Mount and use your file share](#).
5. Haga una copia de seguridad de la base de datos de MySQL o MariaDB en las instalaciones en el directorio donde está montado el recurso compartido de archivos NFS. Puede utilizar cualquiera de las herramientas de copia de seguridad que se analizan en esta guía.
6. Restaure la copia de seguridad de la base de datos en la instancia de base de datos de destino con cualquiera de los enfoques que se describen en esta guía.

Ventajas

- Al crear copias de seguridad de la base de datos de manera directa en el bucket de S3 y restaurar la copia de seguridad en la instancia de base de datos de destino directamente desde el mismo bucket de S3, puede acelerar de manera en gran medida el proceso de migración integral.
- Los archivos de copia de seguridad de las bases de datos se almacenan de manera duradera en Amazon S3. Usted elige la política de administración del ciclo de vida y la clase de almacenamiento de S3.

Limitaciones

Al utilizar los recursos compartidos de archivo de Amazon S3 File Gateway se aplican las limitaciones siguientes:

- El número máximo de recursos compartidos de archivos por puerta de enlace es 50.
- Para evitar conflictos de lectura y escritura cuando varios recursos compartidos de archivos utilizan el mismo bucket de S3, debe configurar cada recurso compartido de archivos para que utilice un nombre de prefijo único.

- El tamaño máximo de un archivo individual es 5 TB, que es el tamaño máximo de los objetos individuales en Amazon S3.
- La longitud máxima de la ruta es de 1024 caracteres.
- Las ACL de Windows se admiten solo en recursos compartidos de archivos habilitados para Active Directory cuando se utilizan clientes SMB de Windows para tener acceso a los recursos compartidos de archivos.
- Amazon S3 File Gateway admite un máximo de 10 entradas de ACL para cada archivo y directorio.
- La configuración de la ACL raíz de los recursos compartidos de archivos SMB solo se encuentra en la puerta de enlace. Esta configuración es persistente a lo largo de las actualizaciones y reinicios de la puerta de enlace.

 Note

Si configura las ACL en la carpeta raíz en lugar de la carpeta principal situada bajo la carpeta raíz, los permisos no son persistentes en Amazon S3.

Prácticas recomendadas

Para más información sobre las prácticas recomendadas de la puerta de enlace de archivo de Amazon S3, consulte [Best practices](#) en la documentación de Amazon S3 File Gateway.

Prácticas recomendadas para la migración de las bases de datos de MySQL y MariaDB grandes

Además de las prácticas recomendadas específicas de cada herramienta que se muestran para cada opción de migración, revise las siguientes prácticas recomendadas generales. Estas prácticas recomendadas se aplican al migrar las bases de datos de MySQL y MariaDB grandes y con varios terabytes, sin importar la herramienta que seleccione:

- Asegúrese de que haya espacio suficiente en las bases de datos de origen y destino para hacer y restaurar la copia de seguridad.
- No cree índices secundarios en la instancia de base de datos de destino hasta que finalice la migración. Los índices secundarios agregan una sobrecarga de mantenimiento durante la importación y pueden ralentizar el proceso de importación.
- Si utiliza un enfoque de varios subprocesos, elija el número correcto de subprocesos. Para la exportación, le recomendamos utilizar un subproceso para cada núcleo de la CPU. Para la importación, le recomendamos utilizar un subproceso por cada dos núcleos de la CPU.
- Los volcados de datos suelen hacerse desde los servidores de las bases de datos activos que forman parte de un entorno de producción crítico. Si el volcado de datos afecta gravemente al rendimiento y esto no es aceptable en su entorno, tenga en cuenta una de las opciones siguientes:
 - El servidor de origen tiene réplicas, puede volcar los datos de una de las réplicas.
 - Los procedimientos habituales de copia de seguridad se aplican al servidor de origen:
 - Si el formato de la copia de seguridad es adecuado para la importación directa a la base de datos de destino, utilice los datos de la copia de seguridad como entrada para el proceso de importación.
 - Si el formato de la copia de seguridad no es adecuado para la importación directa a la base de datos de destino, utilice la copia de seguridad para aprovisionar una base de datos temporal y volcar los datos de esta.
 - Si las réplicas y las copias de seguridad no están disponibles, haga lo siguiente:
 - Haga volcados durante las horas de menor actividad, cuando el tráfico de producción es más bajo.
 - Reduzca la simultaneidad de las operaciones de descarga para que el servidor tenga suficiente capacidad adicional para gestionar el tráfico de producción.
- Cree volcados solo de bases de datos que hayan creado los usuarios.

- Vuelva a crear los usuarios en la base de datos de destino y configure sus permisos. Para más información, consulte [Administración de la identidad y el acceso en Amazon RDS](#), [Administración de la identidad y el acceso en Amazon Aurora](#) o [Identity and Access Management para Amazon EC2](#).
- Al migrar un servidor de bases de datos grande que consta de varias bases de datos independientes, cree una instancia independiente para cada base de datos. Esto es útil para administrar la base de datos de manera más eficiente y puede mejorar el aprovisionamiento de los recursos, mientras que los recursos de computación independientes pueden mejorar el rendimiento de la base de datos.

Recursos

AWS Recomendaciones de

- [Portfolio playbook for AWS large migrations](#)
- [Estrategia de migración para bases de datos relacionales](#)
- [Migración de una base de datos MySQL en las instalaciones a Amazon RDS para MySQL](#)
- [Configure la replicación de datos entre Amazon RDS para MySQL y MySQL en Amazon EC2 mediante GTID](#)
- [Migración de una base de datos de MariaDB en las instalaciones hasta Amazon RDS para MariaDB mediante herramientas nativas](#)

Publicaciones de blog de AWS

- [Security best practices for Amazon RDS for MySQL and MariaDB instances](#)
- [Migrate self-managed MariaDB to Amazon Aurora MySQL](#)

Recursos para restaurar la copia de seguridad

- [Creación de un bucket](#) (documentación de Amazon S3)
- [Conexión a la instancia de Linux mediante SSH](#) (documentación de Amazon EC2)
- [Configuración de AWS CLI](#) (documentación de AWS CLI)
- [comando sync](#) (referencia del comando de AWS CLI)
- [Creación de una política de IAM para acceder a los recursos de Amazon S3](#) (documentación de Aurora)
- [Requisitos previos del clúster de base de datos](#) (documentación de Aurora)
- [Uso de los grupos de subredes de base de datos](#) (documentación de Aurora)
- [Creación de un grupo de seguridad de VPC para un clúster de base de datos privada](#) (documentación de Aurora)
- [Restauración de un clúster de base de datos de Aurora MySQL desde un bucket de Amazon S3](#) (documentación de Aurora)
- [Configuración de la replicación con MySQL o con otro clúster de base de datos de Aurora](#) (documentación de Aurora)

- [Procedimiento rds_set_external_master](#) (documentación de Amazon RDS)
- [Procedimiento rds_start_replication](#) (documentación de Amazon RDS)

AWS Marketing de

- [Amazon Aurora](#)
- [Amazon RDS para MariaDB](#)
- [Amazon RDS para MySQL](#)
- [Amazon S3 File Gateway](#)

Otros recursos

- [Información de XtraBackup](#)
- [MyDumper](#)
- [mysqldump](#)
- [mysqlpump](#)

Historial de documentos

En la siguiente tabla, se describen cambios significativos de esta guía. Si quiere recibir notificaciones de futuras actualizaciones, puede suscribirse a las [notificaciones RSS](#).

Cambio	Descripción	Fecha
disponibilidad de mysqlpump	mysqlpump se eliminó en la versión 8.4 de MySQL. Actualizamos las secciones mysqldump y mysqlpump para reflejar este cambio de disponibilidad.	21 de noviembre de 2024
Prácticas recomendadas de MyDumper	Actualizamos las prácticas recomendadas de MyDumper para agregar información sobre la migración de tablas de bases de datos cifradas.	24 de octubre de 2024
Versiones de Percona XtraBackup	En la sección Percona XtraBackup , actualizamos las instrucciones para reflejar las versiones de Percona XtraBackup compatibles con Amazon Aurora MySQL y Amazon RDS.	3 de agosto de 2023
Publicación inicial	—	6 de abril de 2023

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.