Dell™ PowerVault™ Encryption Key Manager

Guía del usuario

Dell™ PowerVault™ Encryption Key Manager

Guía del usuario

Contenido

Figuras

Tablas

Prefacio

Acerca de este manual

Este manual contiene la información y las instrucciones necesarias para la instalación y el funcionamiento de Dell™ Encryption Key Manager. Incluye conceptos y procedimientos que pertenecen a:

- Unidades de cintas LTO 4 y LTO 5 con posibilidades de cifrado
- Clave criptográfica
- Certificados digitales

A quién va dirigido este manual

Este manual va dirigido a los administradores de almacenamiento y seguridad responsables de la seguridad y la creación de copias de seguridad de los datos más importantes, así como a todos aquellos implicados en la configuración y el mantenimiento de los servidores de Encryption Key Manager del entorno operativo. Se presupone que el lector conoce los dispositivos y las redes de almacenamiento.

Convenios y terminología utilizados en este manual

Este manual utiliza los siguientes convenios tipográficos:

Tabla 1. Convenios tipográficos utilizados en este manual

Convenio	Utilización
negrita	Las palabras o los caracteres en **negrita** representan elementos del sistema que se deben utilizar literalmente, como nombres de mandatos, nombres de archivos, nombres de distintivos, nombres de vías de acceso y opciones de menú seleccionadas.
`monoespaciado`	Los ejemplos, el texto especificado por el usuario y la información que muestra el sistema aparecen con el tipo de letra `monoespaciado`.
cursiva	Las palabras o los caracteres en *cursiva* representan los valores de variables que el usuario debe proporcionar.
[elemento]	Indica los elementos opcionales.
{elemento}	Contiene una lista de la que se debe seleccionar un elemento en las descripciones de formato y sintaxis.
|	Una barra vertical separa los elementos de una lista de opciones.
<Tecla>	Indica la tecla que se debe pulsar.

Aviso de atención

Un aviso de atención indica la posibilidad de que se dañe un programa, dispositivo, sistema o datos. El aviso de atención puede ir acompañado de un signo de admiración, pero no es obligatorio. A continuación, se muestran algunos avisos de atención de ejemplo:

Atención: Si utiliza un destornillador eléctrico para realizar este procedimiento, puede dañar la cinta.

Publicaciones relacionadas

Consulte las publicaciones siguientes para obtener más información:

- *Iniciación a las bibliotecas de cintas Dell™ PowerVault™ TL2000 y TL4000* proporciona información sobre la instalación.
- *Dell™ PowerVault™ TL2000 Tape Library and TL4000 Tape Library SCSI Reference* proporciona los mandatos SCSI soportados y el protocolo que rige el comportamiento de la interfaz SCSI.

Información sobre Linux

Información sobre Red Hat

El URL siguiente está relacionado con los sistemas Red Hat Linux®:

- **http://www.redhat.com**

Información sobre SuSE

El URL siguiente está relacionado con los sistemas SuSE Linux:

- **http://www.suse.com**

Información sobre Microsoft Windows

El URL siguiente proporciona acceso a la información sobre sistemas Microsoft® Windows®:

- **http://www.microsoft.com**

Soporte en línea

Visite **http://support.dell.com** para obtener la siguiente publicación relacionada:

La publicación *Dell Encryption Key Manager Quick Start Guide* proporciona información para crear una configuración básica.

Visite **http://www.dell.com** para obtener la siguiente publicación relacionada:

El documento técnico *Library Managed Encryption for Tape* sugiere los métodos recomendados para el cifrado de cintas LTO.

Lea esto en primer lugar

Contacto con Dell

En caso de clientes de Estados Unidos, llamen al 800-WWW-DELL (800-999-3355).

Nota: Si no dispone de una conexión a Internet activa, encontrará información de contacto en la factura de compra, la hoja de embalaje, el recibo o el catálogo de productos de Dell.

Dell proporciona varias opciones en línea y telefónicas de servicio y soporte. La disponibilidad varía según el país y el producto, y puede que algunos servicios no estén disponibles en su zona. Para ponerse en contacto con Dell para incidencias de compras, servicio técnico o servicio al cliente:

1. Visite **http://support.dell.com**.
2. Elija su país o región en el menú desplegable **Choose A Country/Region** al final de la página.
3. Pulse **Contacto** en el lado izquierdo de la página.
4. Seleccione el servicio o enlace de soporte adecuados según sus necesidades.
5. Elija el método de contacto con Dell más apropiado para usted.

Capítulo 1. Visión general del cifrado de cintas

Los datos son unos de los recursos más valiosos en un entorno empresarial competitivo. Proteger estos datos, controlar el acceso a ellos y verificar su autenticidad mientras se mantiene su disponibilidad son prioridades en el mundo en que vivimos, tan preocupado por la seguridad. El cifrado de datos es una herramienta que responde a muchas de estas necesidades. Dell Encryption Key Manager (en lo sucesivo Encryption Key Manager) simplifica las tareas de cifrado.

Las unidades LTO 4 y LTO 5 también cifrar datos a medida que se graban a cualquier cartucho de datos LTO 4 y LTO 5. Esta nueva posibilidad añade una gran medida de seguridad a los datos almacenados, sin la sobrecarga de proceso ni la degradación de rendimiento asociados al cifrado realizado en el servidor, y sin el gasto de una aplicación dedicada a ello.

La solución de cifrado de unidad de cintas consta de tres elementos principales:

Unidad de cintas habilitada para cifrado
Todas las unidades de cintas LTO 4 y LTO 5 deben habilitarse a través de la interfaz de la biblioteca.

Consulte el apartado “Requisitos de hardware y software” en la página 2-2 para obtener más información sobre las unidades de cintas.

Gestión de claves de cifrado
El cifrado implica el uso de varios tipos de claves, en capas sucesivas. La generación, el mantenimiento, el control y la transmisión de estas claves dependen del entorno operativo en el que se haya instalado la unidad de cintas de cifrado. Algunas aplicaciones pueden realizar gestión de claves. Para aquellos entornos sin tales aplicaciones o para aquellos donde se prefiere utilizar una aplicación neutra, Dell Encryption Key Manager realiza todas las tareas de gestión de claves necesarias. El apartado “Gestión de cifrado” en la página 1-3 describe estas tareas en detalle.

Política de cifrado
Es el método utilizado para implementar el cifrado. Incluye las reglas que determinan qué volúmenes se cifran y el mecanismo para la selección de claves. Cómo y dónde se configuran estas reglas depende del entorno operativo. Consulte el apartado “Gestión de cifrado” en la página 1-3 para obtener más información.

Componentes

Encryption Key Manager forma parte del entorno de Java y utiliza los componentes de Java Security para sus funciones criptográficas. (Para obtener más información sobre los componentes de Java Security, consulte la sección de publicaciones relacionadas). Encryption Key Manager tiene tres componentes principales utilizados para controlar su comportamiento. Estos componentes son:

Almacén de claves de seguridad de Java
El almacén de claves se define como parte de Java Cryptography Extension (JCE) y un elemento de los componentes de seguridad de Java, que forman parte, a su vez, de Java Runtime Environment. Un almacén de claves guarda los certificados y claves (o punteros de los certificados y claves) utilizados por Encryption Key Manager para realizar operaciones

criptográficas. Varios tipos de almacenes de claves Java que ofrecen distintas características operativas para satisfacer sus necesidades están soportados. Estas características se analizan en profundidad en el apartado “Consideraciones sobre el almacén de claves” en la página 2-3.

Es imposible exagerar la importancia de proteger los datos del almacén de claves. Sin acceso al almacén de claves, no podrá descifrar las cintas cifradas. Lea con atención los temas siguientes para comprender los métodos disponibles para proteger los datos del almacén de claves.

Archivos de configuración

Los archivos de configuración permiten personalizar el comportamiento de Encryption Key Manager para satisfacer las necesidades de su organización. Estas opciones de comportamiento se describen ampliamente en este documento, primero en el Capítulo 2, “Planificación del entorno de Encryption Key Manager”, en la página 2-1, luego en el Capítulo 4, “Configuración de Encryption Key Manager”, en la página 4-1 y, por último, en el Apéndice B, donde se describe el conjunto completo de opciones de configuración.

Tabla de unidades de cintas

Encryption Key Manager utiliza la tabla de unidades de cintas para realizar el seguimiento de los dispositivos de cintas a los que da soporte. La tabla de unidades de cintas es un archivo binario no editable cuya ubicación se especifica en el archivo de configuración. Puede cambiar su ubicación para satisfacer sus necesidades.

Archivo KeyGroups.xml

Este archivo protegido por contraseña contiene los nombres de todos los grupos de claves de cifrado y los alias de las claves de cifrado asociadas con cada grupo de claves.

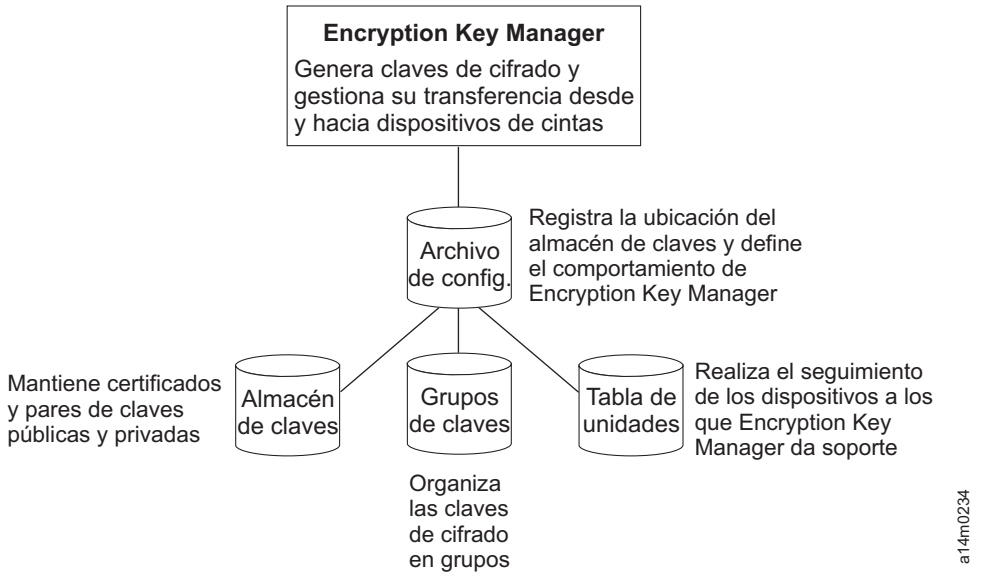

Figura 1-1. Los cuatro componentes principales de Encryption Key Manager

Gestión de cifrado

Dell Encryption Key Manager es un programa de software Java™ que le ayuda a las unidades de cintas con capacidad de cifrado a generar, proteger, almacenar y mantener claves de cifrado utilizadas para cifrar la información que se graba y descifrar la información que se lee de soportes de cintas (formatos de cinta y cartuchos). Encryption Key Manager funciona en Linux (SLES y RHEL) y Windows, y está diseñado para ejecutarse en segundo plano como un recurso compartido desplegado en varias ubicaciones de una empresa. Una interfaz de la línea de mandatos cliente proporciona un robusto conjunto de mandatos para personalizar Encryption Key Manager para su entorno y supervisar su funcionamiento. Hay muchas funciones de personalización y supervisión disponibles en la interfaz gráfica de usuario (GUI) de Dell Encryption Key Manager. Encryption Key Manager utiliza uno o más almacenes de claves para guardar los certificados y claves (o punteros a los certificados y claves) necesarios para todas las tareas de cifrado. Consulte el apartado “Consideraciones sobre el almacén de claves” en la página 2-3 para obtener información detallada.

IMPORTANTE Encryption Key Manager INFORMACIÓN DE CONFIGURACIÓN DEL SERVIDOR DE HOST: es recomendable que las máquinas que alojan el programa Dell Encryption Key Manager utilicen memoria ECC para minimizar el riesgo de pérdida de datos. Encryption Key Manager realiza la función de solicitar la generación de claves de cifrado y de pasar dichas claves a las unidades de cintas LTO 4 y LTO 5. El material de las claves, en formato empaquetado (cifrado) reside en la memoria del sistema mientras es procesado por Encryption Key Manager. Tenga en cuenta que el material de las claves debe ser transferido sin errores a la unidad de cintas correspondiente para que los datos grabados en un cartucho puedan ser recuperados (descifrados). Si, por algún motivo, el material de las claves (en formato empaquetado o no) resulta dañado debido a un error de bit en la memoria del sistema y ese material de claves se utiliza para grabar datos en un cartucho, los datos escritos en el cartucho no podrán ser recuperados (no podrán ser descifrados posteriormente). Existen métodos para asegurar que tales errores no se producen. Sin embargo, si la máquina que aloja Encryption Key Manager no está utilizando memoria ECC (Código de corrección de errores), existe la posibilidad de que el materia de las claves resulte dañado mientras está en la memoria del sistema y que los daños provoquen una pérdida de datos. La probabilidad de que esto suceda es baja, pero siempre es recomendable que las máquinas que alojan aplicaciones vitales (como Encryption Key Manager) utilicen memoria ECC.

Encryption Key Manager actúa como un proceso en segundo plano en espera de solicitudes de generación de clave o de recuperación de claves que le sean enviadas mediante una vía de comunicación TCP/IP entre la aplicación y la biblioteca de cintas. Cuando una unidad de cintas graba datos cifrados, primero solicita una clave de cifrado de Encryption Key Manager. Al recibir la solicitud, Encryption Key Manager realiza las siguientes tareas.

Encryption Key Manager captura una clave AES existente de un almacén de claves y la empaqueta para realizar una transferencia segura a la unidad de cintas donde se desempaqueta al llegar y se utiliza para cifrar los datos grabados en cinta.

Cuando una unidad LTO 4 o LTO 5 lee una cinta cifrada, Encryption Key Manager captura la clave requerida del almacén, utiliza la información en el ID de clave de la cinta, y la sirve a la unidad de cintas empaquetada para una transferencia segura.

hay dos métodos entre los que elegir para la gestión de cifrado. Estos métodos difieren del lugar donde reside el motor de política de cifrado, dónde se realiza la gestión de claves para su solución y cómo se conecta a la unidad Encryption Key Manager. El entorno operativo determina cuál es el que mejor se adapta en cada caso. La gestión de claves y el motor de políticas de cifrado se pueden encontrar en una de las siguientes dos capas del entorno.

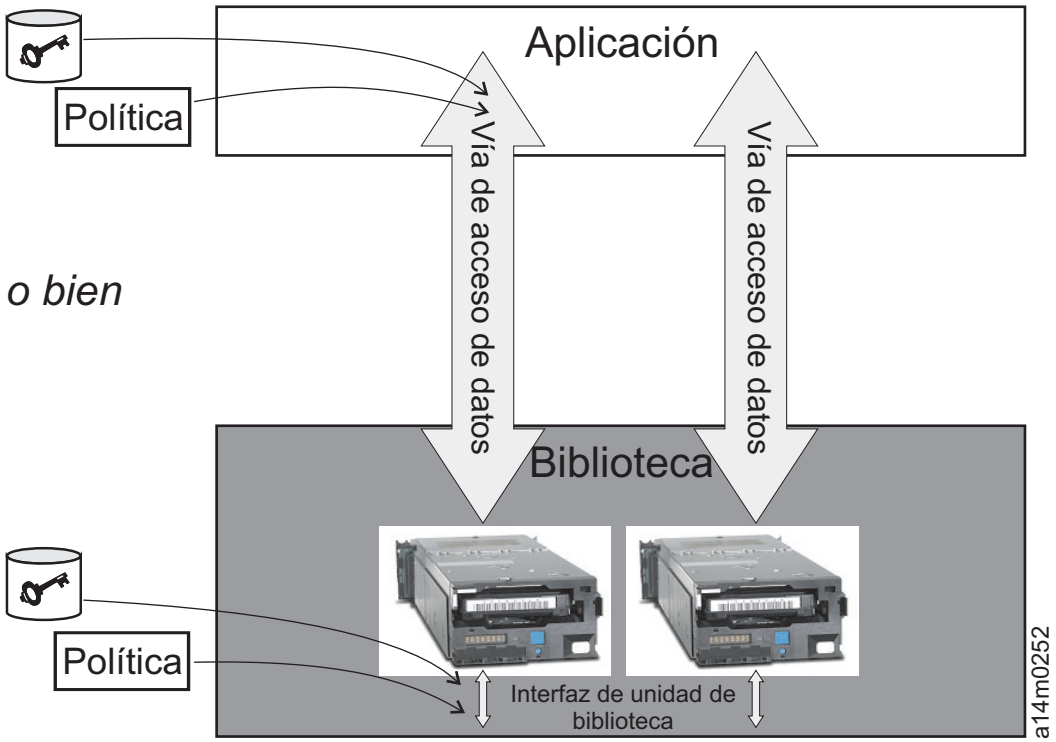

Figura 1-2. Hay dos posibles ubicaciones para el motor de políticas de cifrado y la gestión de claves.

Capa de la aplicación
Un programa de aplicación, independiente del gestor de claves, inicia una transferencia de datos para el almacenamiento en cinta. Consulte el apartado "Cifrado de cintas gestionado por aplicaciones" para ver las aplicaciones soportadas.

Capa de la biblioteca
El alojamiento del almacenamiento en cintas, como la biblioteca de cintas Dell PowerVault TL2000/TL4000 y la familia de productos ML6000. Una biblioteca de cintas moderna contiene una interfaz interna para cada una de sus unidades de cintas.

Cifrado de cintas gestionado por aplicaciones

Este método es el mejor cuando los sistemas operativos ejecutan una aplicación que ya es capaz de generar y gestionar políticas y claves de cifrado. Las políticas que especifican cuándo se debe utilizar el cifrado se definen mediante la interfaz de aplicación. Las políticas y las claves pasan por la vía de acceso de datos entre la capa de la aplicación y las unidades de cintas de cifrado. El cifrado es el resultado de la interacción entre la unidad de cintas habilitada para el cifrado y la aplicación, y no es necesario que se realicen cambios en las capas del sistema y de la biblioteca. Dado que la aplicación gestiona las claves de cifrado, los volúmenes

grabados y cifrados utilizando el método de la aplicación sólo se pueden leer utilizando el método de cifrado gestionado por aplicaciones, con la misma aplicación que los grabó.

El cifrado de cintas gestionado por aplicaciones no necesita, o no utiliza, Encryption Key Manager.

Las siguientes aplicaciones de versión mínima se pueden utilizar para gestionar el cifrado:

- CommVault Galaxy 7.0 SP1
- Symantec Backup Exec 12

Se da soporte al cifrado de cintas gestionado por aplicaciones en las unidades de cintas LTO 4 y LTO 5 en:

- Biblioteca de cintas Dell™ PowerVault™ TL2000
- Biblioteca de cintas Dell™ PowerVault™ TL4000
- Biblioteca de cintas Dell™ PowerVault™ ML6000

Consulte la documentación de su software de aplicación de copia de seguridad de cintas para aprender cómo gestionar políticas y claves de cifrado.

Cifrado de cintas gestionado por bibliotecas

Utilice este método para las unidades de cintas LTO 4 y LTO 5 en la la Biblioteca de cintas Dell™ PowerVault™ TL2000, la Biblioteca de cintas Dell™ PowerVault™ TL4000 o la Biblioteca de cintas Dell™ PowerVault™ ML6000conectada en abierto. La generación de claves y gestión es realizada por Encryption Key Manager, una aplicación Java que se ejecuta en un host conectado a la biblioteca. Las claves y el control de políticas pasan a través de la interfaz biblioteca a unidad, por lo que el cifrado es transparente para las aplicaciones.

Acerca de las claves de cifrado

Una clave de cifrado es una serie aleatoria de bits generada específicamente para desordenar y ordenar los datos. Las claves de cifrado se crean utilizando algoritmos diseñados para garantizar que cada clave sea única e imprevisible. Cuanto más larga sea la clave creada de este modo, más difícil será romper el código de cifrado. Los métodos de cifrado IBM y T10 utilizan claves de algoritmo AES de 256 bits para cifrar los datos. AES de 256 bits es el cifrado estándar que actualmente reconoce y recomienda el gobierno de Estados Unidos, que permite tres longitudes distintas. Las claves de 256 bits son las más largas permitidas por AES.

Encryption Key Manager utiliza dos tipos de algoritmos de cifrado: algoritmos simétricos y algoritmos asimétricos. El cifrado de claves secretas o simétricas utiliza una única clave para el cifrado o el descifrado. El cifrado de clave simétrico se utiliza, generalmente, para cifrar grandes cantidades de datos de una manera eficaz. Las claves AES de 256 bits son claves simétricas. El cifrado público/privado o asimétrico utiliza un par de claves. Los datos cifrados utilizando una clave sólo se pueden descifrar utilizando la otra clave en el par de claves público/privado. Cuando se genera un par de claves asimétricas, se utiliza la clave pública para cifrar y la clave privada para descifrar.

Encryption Key Manager utiliza claves simétricas y asimétricas: cifrado simétrico para el cifrado de alta velocidad de usuarios o datos del host, y cifrado asimétrico (más lento) para proteger la clave simétrica.

Las claves de cifrado pueden ser generadas para Encryption Key Manager por un programa de utilidad como keytool. La responsabilidad de generar claves AES y la manera en que se transmiten a la unidad de cintas dependen del método de gestión del cifrado. Sin embargo, puede que sea útil comprender la diferencia entre cómo Encryption Key Manager utiliza las claves de cifrado y cómo las usan otras aplicaciones.

Proceso de claves de cifrado por parte de Dell Encryption Key Manager

En el cifrado de cintas gestionado por bibliotecas, los datos no cifrados se envían a la unidad de cintas LTO 4 o LTO 5 y se convierten en texto cifrado utilizando una clave de datos (DK) simétrica generada previamente de un almacén de claves a disposición de Encryption Key Manager, para luego grabarse en cintas. Encryption Key Manager selecciona una clave de datos pregenerada con el método round robin. Las claves de datos se reutilizan en varios cartuchos de cinta cuando se ha generado previamente un número insuficiente de claves de datos. Encryption Key Manager envía la clave de datos a la unidad de cintas LTO 4 o LTO 5 en formato cifrado o empaquetado. Las unidades de cintas LTO 4 y LTO 5 desempaquetan esta clave de datos y la utilizan para llevar a cabo el cifrado o el descifrado. Sin embargo, no se almacenan claves empaquetadas en los cartuchos de cinta LTO 4 o LTO 5. Después de grabar el volumen cifrado, la clave de datos debe resultar accesible en función de la etiqueta de clave o el alias, y a disposición de Encryption Key Manager para que se pueda leer el volumen. En la Figura 1-3 en la página 1-7 se muestra este proceso.

Dell Encryption Key Manager le ofrece también la posibilidad de organizar las claves simétricas para el cifrado LTO en grupos de claves. De este modo, puede agrupar las claves según el tipo de datos que cifren, los usuarios que tengan acceso a ellas, o por otras características significativas. Consulte el apartado “Creación y gestión de grupos de claves” en la página 3-14 para obtener más información.

Proceso de claves de cifrado por parte de otras aplicaciones

En el cifrado de cintas gestionado por aplicaciones, los datos no cifrados se envían a las unidades de cintas LTO 4 o LTO 5 y se convierten en texto cifrado utilizando una clave de datos simétrica proporcionada por la aplicación, para luego grabarse en cintas. La clave de datos no se almacena en ningún lugar del cartucho de cinta. Después de grabar el volumen cifrado, la clave de datos se debe encontrar en una ubicación a la que tenga acceso la aplicación, como una base de datos de servidor, para que se pueda leer el volumen.

Las unidades de cintas LTO 4 y LTO 5 pueden utilizar aplicaciones como Yosemite (para las bibliotecas de cintas Dell PowerVault TL2000 y TL4000), CommVault y Symantec Backup Exec para el cifrado gestionado por aplicaciones.

Las aplicaciones que utilizan el conjunto de mandatos T10 para realizar el cifrado también pueden utilizar las unidades de cintas LTO 4 y LTO 5. El conjunto de mandatos T10 utiliza claves AES simétricas de 256 bits proporcionadas por la aplicación. T10 puede utilizar varias claves de datos exclusivas por cartucho de cinta, e incluso grabar datos cifrados y datos sin cifrar en el mismo cartucho de cinta. Cuando la aplicación cifra un cartucho de cinta, selecciona o genera una

clave de datos utilizando un método determinado por la aplicación y lo envía a la unidad de cintas. La clave **no** se empaqueta con una clave pública asimétrica y **no** se almacena en el cartucho de cinta. Después de grabar los datos cifrados en la cinta, la clave de datos se debe encontrar en una ubicación disponible para la aplicación para poder leer los datos.

El proceso del cifrado de cintas gestionado por bibliotecas y gestionado por aplicaciones se muestra en la Figura 1-3.

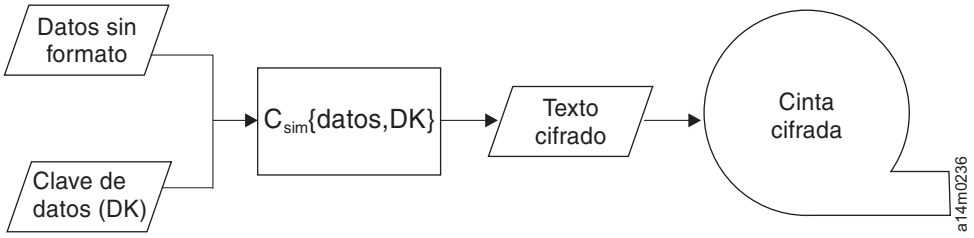

Figura 1-3. Cifrado utilizando claves de cifrado simétricas. Cifrado gestionado por bibliotecas y gestionado por aplicaciones en unidades de cintas LTO 4 y LTO 5.

En resumen

El número de claves de cifrado que se pueden utilizar para cada volumen depende de la unidad de cintas, el estándar de cifrado y el método utilizado para gestionar el cifrado. Para un cifrado transparente de LTO 4 y LTO 5, es decir, utilizando el cifrado gestionado por bibliotecas con Encryption Key Manager, la exclusividad de las claves de datos depende de la disponibilidad de un número suficiente de claves generadas previamente para Encryption Key Manager.

Tabla 1-1. Resumen de claves de cifrado

Método de gestión de cifrado	**Claves utilizadas por**	
	Cifrado de IBM	**Cifrado de T10**
Cifrado gestionado por bibliotecas	1 clave de datos / cartucho	N/A
Cifrado gestionado por aplicaciones	Varias claves de datos / cartuchos	Varias claves de datos / cartuchos
Clave de datos = Clave de datos AES simétrica de 256 bits		

Capítulo 2. Planificación del entorno de Encryption Key Manager

Esta sección está destinada a proporcionar información que le permitirá determinar la configuración de Encryption Key Manager que mejor se adapte a sus necesidades. Deben tenerse en cuenta muchos factores para planificar cómo configurar la estrategia de cifrado.

Las tareas de configuración de cifrado de un vistazo

Para poder utilizar la posibilidad de cifrado de la unidad de cintas, debe satisfacer determinados requisitos de software y hardware. El objetivo de las listas de comprobación siguientes es ayudarle a satisfacer estos requisitos.

Tareas de configuración de Encryption Key Manager

Antes de poder cifrar cintas, deberá configurar y ejecutar primero Encryption Key Manager para que pueda comunicarse con las unidades de cintas de cifrado. Encryption Key Manager no necesita estar ejecutándose mientras se instalan las unidades de cintas, pero debe estar ejecutándose para realizar el cifrado.

- Decida qué plataforma(s) utilizar como servidor(es) Encryption Key Manager.
- Actualice el sistema operativo del servidor, si es necesario. (Consulte el apartado “Requisitos de hardware y software” en la página 2-2).
- Instale los archivos de políticas sin restricciones de Java. (Consulte el apartado “Requisitos de hardware y software” en la página 2-2).
- Actualice el JAR de Encryption Key Manager. (Consulte “Descarga de la última versión de la imagen ISO del gestor de claves” en la página 3-1).
- Cree claves, certificados y grupos de claves.

  “Uso de la GUI para crear un archivo de configuración, un almacén de claves y certificados” en la página 3-5

  “Creación y gestión de grupos de claves” en la página 3-14
- Estos pasos no son necesarios si sigue el procedimiento del apartado “Uso de la GUI para crear un archivo de configuración, un almacén de claves y certificados” en la página 3-5, a menos que desee sacar provecho de las opciones de configuración adicionales:
  - Si fuese necesario, importe las claves y los certificados. (Consulte el apartado “Importación de claves de datos utilizando Keytool -importseckey ” en la página 3-12.)
  - Defina el archivo de propiedades de configuración. (Consulte el Capítulo 4, “Configuración de Encryption Key Manager”, en la página 4-1).
  - Defina las unidades de cintas en Encryption Key Manager o establecer el valor de la propiedad de configuración **drive.acceptUnknownDrives**. (Consulte el mandato “adddrive” en la página 5-8 para definir las unidades de manera explícita, o consulte el apartado “Actualización automática de la tabla de unidades de cintas” en la página 4-1).
  - Inicie el servidor Encryption Key Manager. (Consulte el apartado “Inicio, Renovación y Detención del Servidor del gestor de claves” en la página 5-1.)
  - Inicie el cliente de la interfaz de línea de mandatos. (Consulte el apartado “Cliente de la interfaz de línea de mandatos” en la página 5-5.)

Planificación para el cifrado de cintas gestionado por bibliotecas

Para realizar el cifrado, necesita:

- Unidades de cintas LTO 4 y LTO 5 con posibilidades de cifrado
- Almacén de claves
- Dell Encryption Key Manager

Tareas de cifrado de cintas gestionado por bibliotecas

1. Instale y conecte las unidades de cintas LTO 4 y LTO 5.
   - Actualice el firmware de la biblioteca (TL2000, TL4000, ML6000 cuando sea necesario). Visite **http://support.dell.com**.
     - Mínima versión de firmware de Biblioteca de cintas Dell™ PowerVault™ TL2000 necesaria = 5.xx.
     - Mínima versión de firmware de Biblioteca de cintas Dell™ PowerVault™ TL4000 necesaria = 5.xx.
     - Mínima versión de firmware necesaria para la familia de Biblioteca de cintas Dell™ PowerVault™ ML6000 = 415G.xxx.
   - Actualice el firmware de la unidad de cintas si es necesario. La versión mínima y necesaria del firmware es 77B5.
2. Habilite las unidades de cintas y la biblioteca de cintas LTO 4 y LTO 5 para el cifrado de cintas gestionado por bibliotecas (consulte la información de la biblioteca de cintas de Dell para obtener más información).
   - Añada las direcciones IP de Encryption Key Manager Server
3. Utilice las funciones de diagnóstico de biblioteca para verificar las vías de acceso de Encryption Key Manager y la configuración de cifrado (consulte la información de la biblioteca de cintas de Dell para obtener más detalles).

Requisitos de hardware y software

Nota: Sólo la versión IBM de Java Runtime Environment (JRE) para cada una de las siguientes plataformas da soporte a Encryption Key Manager.

Linux Solution Components

Sistemas operativos

- RHEL 4
- RHEL 5
- SLES 9
- SLES 10
- SLES 11

Encryption Key Manager (cuando se ejecuta en Linux)

Tabla 2-1. Requisitos mínimos de software para Linux

Plataforma	IBM Software Developer Kit	Disponible en:
AMD/Opteron/EM64T de 64 bits	Java 6.0 SR5	**http://support.dell.com**
Compatible con Intel® de 32 bits		

Bibliotecas de cintas

Para la biblioteca de cintas Dell PowerVault TL2000, la biblioteca de cintas TL4000 y la biblioteca de cintas ML6000, asegúrese de que el nivel de firmware es el último disponible. Para actualizar el firmware, visite **http://support.dell.com**.

Unidad de cintas

Para las unidades de cintas LTO 4 y LTO 5, asegúrese de que el nivel de firmware es el último disponible. Para actualizar el firmware, visite **http://support.dell.com**.

Windows Solution Components

Sistemas operativos

Windows Server 2003, 2008 y 2008 R2

Dell Encryption Key Manager

La versión mínima requerida de Encryption Key Manager es la 2.1 con una fecha de compilación de 20070914 o posterior, y uno de los siguientes IBM Runtime Environments:

Tabla 2-2. Requisitos mínimos de software para Windows

Sistema operativo	IBM Runtime Environment
Windows 2003	• Entorno de ejecución IBM® de 64 bits para Windows en la arquitectura AMD64/EM64T, Java 2 Technology Edition, Versión 5.0 SR5 • Entorno de ejecución IBM de 32 bits para Windows, Java 2 Technology Edition, Versión 5.0 SR5
Windows 2008 y 2008 R2	Entorno de ejecución IBM de 64 para Windows en la arquitectura AMD64/EM64T, Java 2 Technology Edition, Versión 6.0 SR5

Bibliotecas de cintas

Para Biblioteca de cintas Dell™ PowerVault™ TL2000, Biblioteca de cintas Dell™ PowerVault™ TL4000 y Biblioteca de cintas Dell™ PowerVault™ ML6000, asegúrese de que el nivel de firmware es el último disponible. Para actualizar el firmware, visite **http://support.dell.com**.

Unidad de cintas

Para las unidades de cintas LTO 4 y LTO 5, asegúrese de que el nivel de firmware es el último disponible. Para actualizar el firmware, visite **http://support.dell.com**.

Consideraciones sobre el almacén de claves

Es imposible exagerar la importancia de preservar sus datos del almacén de claves. Sin acceso al almacén de claves, no podrá descifrar las cintas cifradas. Lea con atención los temas siguientes para comprender los métodos disponibles para proteger los datos del almacén de claves.

Almacén de claves JCEKS

EKM da soporte al tipo de almacén de claves JCEKS.

JCEKS (basado en archivos de Unix System Services) es un almacén de claves basado en archivos al que se da soporte en todas las plataformas en las que se ejecuta EKM. Por lo tanto, resulta bastante fácil copiar el contenido de este almacén de claves para realizar copias de seguridad o recuperaciones, así como mantener dos instancias de EKM sincronizadas para la migración tras error. JCEKS proporciona protección basada en contraseña del contenido del almacén de claves de seguridad, y su rendimiento es bastante bueno. Se pueden utilizar métodos de copia de archivos, como FTP.

Claves de cifrado y las unidades de cintas LTO 4 y LTO 5

Dell Encryption Key Manager y sus unidades de cintas soportadas utilizan claves simétricas AES de 256 bits para cifrar datos. Este tema explica lo que debe saber acerca de estas claves y certificados.

Cuando se realizan tareas de cifrado en las unidades de cinta LTO 4 o LTO 5 para cartuchos de cinta LTO, Encryption Key Manager sólo utiliza claves de datos simétricas AES de 256 bits.

Cuando LTO 4 o LTO 5 solicitan una clave, Encryption Key Manager utiliza el alias especificado para la unidad de cintas. Si no se ha especifica ningún alias para la unidad de cintas, se utiliza un alias de un grupo de claves, una lista de alias de claves o un intervalo de alias de claves especificados en la propiedad de configuración symmetricKeySet. Si falta un alias específico de la unidad de cintas, se seleccionan alias de las otras entidades con el método round robin para equilibrar el uso de claves.

El alias seleccionado se asocia con una clave de datos (DK) simétrica que ha cargado previamente en el almacén de claves. Encryption Key Manager envía esta clave de datos, empaquetada con una clave distinta que la unidad de cintas puede descifrar, a la unidad de cintas LTO 4 o LTO 5 para cifrar los datos. La clave de datos no se transmite mediante TCP/IP en la copia no cifrada. El alias seleccionado también se convierte en una unidad denominada identificador de claves de datos (DKi), que se graba en una cinta con los datos cifrados. De esta manera, Encryption Key Manager puede utilizar el identificador de claves de datos para identificar la clave de datos correcta necesaria para descifrar los datos cuando se lee la cinta LTO 4 o LTO 5.

En los temas **adddrive** y **moddrive** del apartado “Mandatos CLI” en la página 5-8 se muestra cómo especificar un alias para una unidad de cintas. Consulte el apartado “Generación de claves y alias para el cifrado en LTO 4 y LTO 5” en la página 3-9, que incluye información sobre la importación de claves, la exportación de claves y la especificación de alias predeterminados en la propiedad de configuración symmetricKeySet. En el apartado “Creación y gestión de grupos de claves” en la página 3-14 se muestra cómo definir un grupo de claves y rellenarlo con alias del almacén de claves.

La Figura 2-1 en la página 2-5 muestra cómo se procesan las claves para las operaciones de grabación cifrada.

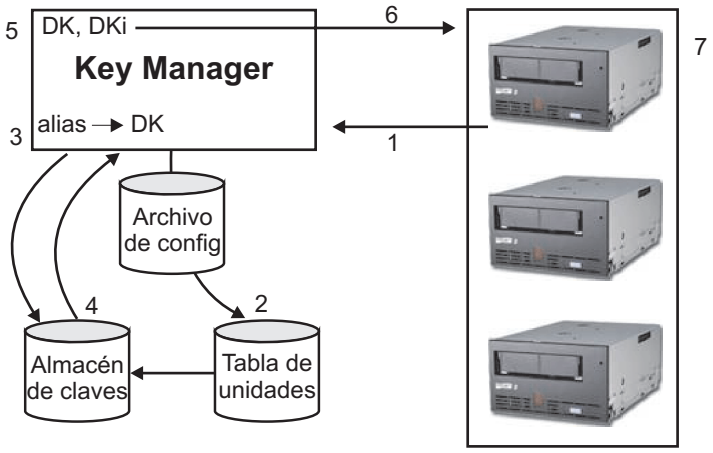

Figura 2-1. Solicitud de la unidad de cintas LTO 4 o LTO 5 para una operación de grabación cifrada

1. La unidad de cintas solicita una clave para cifrar cintas
2. Encryption Key Manager verifica el dispositivo de cintas en la tabla de unidades
3. Si no se especifica ningún alias en la solicitud ni tampoco en la tabla de unidades, Encryption Key Manager selecciona un alias del conjunto de alias o del grupo de claves en keyAliasList
4. Encryption Key Manager captura una clave de datos correspondiente del almacén de claves
5. Encryption Key Manager convierte el alias en un identificador de claves de datos y empaqueta la clave de datos con una clave que la unidad puede descifrar
6. Encryption Key Manager envía la clave de datos y el identificador de claves de datos a la unidad de cintas
7. La unidad de cintas desempaqueta la clave de datos y graba los datos cifrados y el identificador de claves de datos en la cinta

La Figura 2-2 muestra cómo se procesan las claves para las operaciones de lectura cifrada.

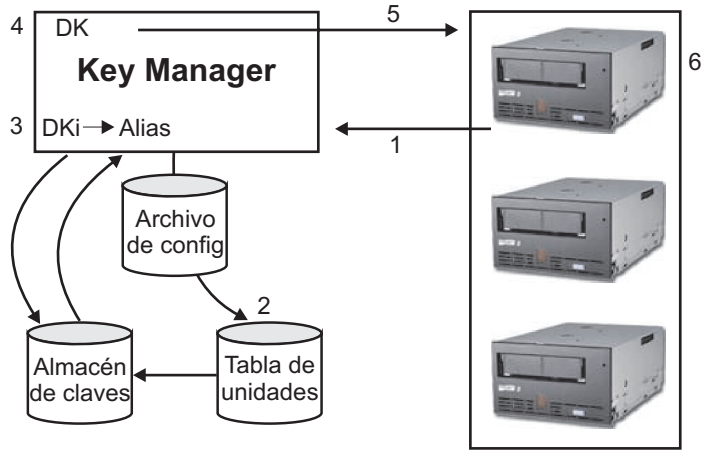

Figura 2-2. Solicitud de la unidad de cintas LTO 4 o LTO 5 para una operación de lectura de cifrado

1. La unidad de cintas recibe la solicitud de lectura y envía el identificador de claves de datos a Encryption Key Manager
2. Encryption Key Manager verifica el dispositivo de cintas en la tabla de unidades
3. Encryption Key Manager convierte el identificador de claves de datos en un alias y captura la clave de datos correspondiente del almacén de claves
4. Encryption Key Manager empaqueta la clave de datos con una clave que la unidad puede descifrar
5. Encryption Key Manager envía la clave de datos empaquetada a la unidad de cintas
6. La unidad de cintas desempaqueta la clave de datos y la utiliza para descifrar los datos

Creación de copias de seguridad de los datos del almacén de claves

Nota: Debido a la importancia de las claves del almacén de claves, es fundamental realizar copias de seguridad de estos datos en un dispositivo no cifrado, para poder recuperarlos si lo necesita y poder acceder a las cintas cifradas utilizando dichos certificados asociados a la biblioteca o la unidad de cintas. Si no realiza correctamente copias de seguridad del almacén de claves, perderá de manera irrevocable el acceso a todos los datos cifrados.

Hay muchas maneras de realizar una copia de seguridad de esta información del almacén de claves. Cada tipo de almacén de claves tiene sus propias características. Estas directrices generales se aplican en todos los casos:

- Guarde una copia de todos los certificados cargados en el almacén de claves (por lo general, un archivo de formato PKCS12).
- Utilice las funciones de copia de seguridad del sistema (como RACF) para crear una copia de seguridad de la información del almacén de claves (tenga cuidado de no cifrar esta copia utilizando las unidades de cintas cifradas, ya que imposibilitaría el descifrado y la recuperación).
- Mantenga un Encryption Key Manager primario y secundario y una copia del almacén de claves (para copias de seguridad y como redundancia en caso de migración tras error). Realice una copia de seguridad del almacén de claves primario y secundario, para una mayor redundancia.
- Para un almacén de claves JCEKS, sólo tiene que copiar el archivo de almacén de claves y almacenar la copia clara (sin cifrar) en una ubicación segura, como una caja fuerte (tenga cuidado de no descifrar esta copia utilizando las unidades de cintas cifradas, ya que sería imposible descifrarlas y recuperarlas).

Como mínimo, debe realizar una copia de seguridad de los datos del almacén de claves siempre que los modifique. Encryption Key Manager no modifica los datos del almacén de claves. Los únicos cambios que se realizan sobre el almacén de claves son aquellos que se aplican, así que debe tener cuidado y copiar el almacén de claves en cuanto lo modifique.

Copias de seguridad de los archivos utilizando la GUI

1. Abra la GUI si todavía no la ha iniciado:

   En Windows
   Vaya a c:\ekm\gui y pulse **LaunchEKMGui.bat**

En plataformas Linux

Vaya a /var/ekm/gui y especifique `. ./LaunchEKMGui.sh`

2. Seleccione **Backup Critical Files** en el navegador en la parte izquierda de la GUI de Encryption Key Manager.
3. Especifique la vía de acceso de los datos con copia de seguridad en el diálogo que aparece (Figura 2-3).

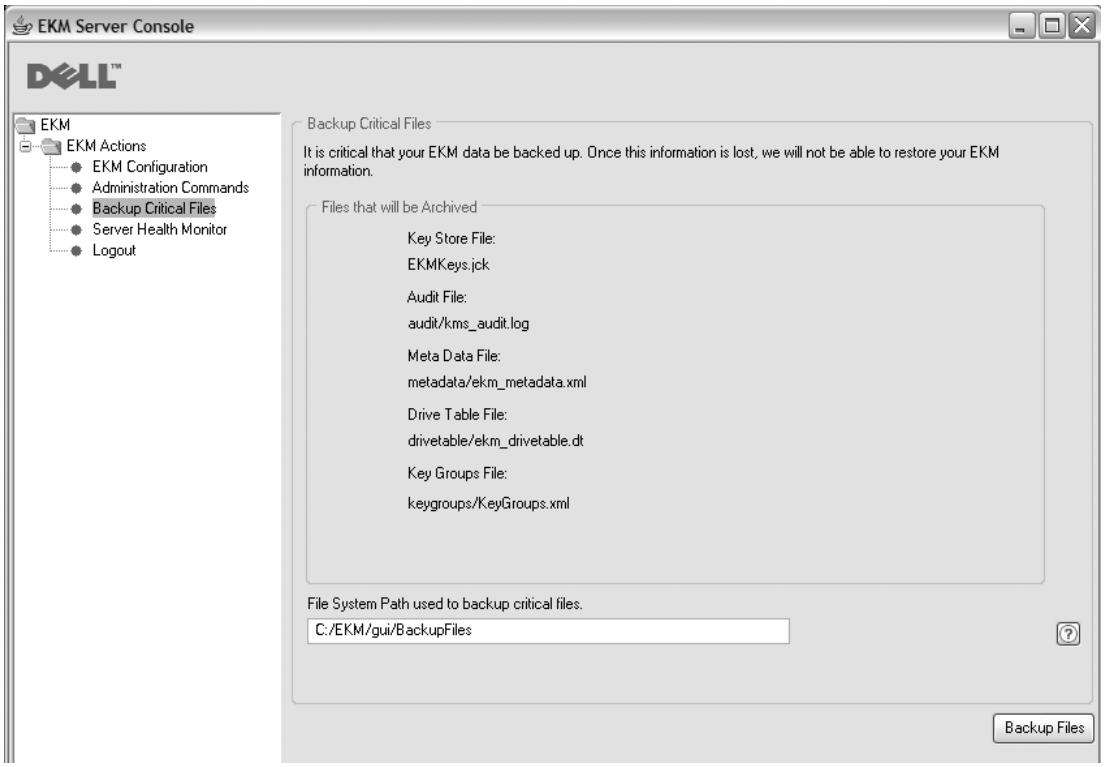

Figura 2-3. Ventana de copia de seguridad de archivos fundamentales

4. Pulse **Backup Files**.
5. Aparece un mensaje de información con los resultados obtenidos.

Varios gestores de claves para obtener redundancia

Encryption Key Manager está diseñado para funcionar con unidades de cintas y bibliotecas para permitir la redundancia y, por tanto, la alta disponibilidad, por lo que es posible tener varios gestores de claves dando servicio al mismo tipo de unidades y bibliotecas de cintas. Además, estos gestores de claves no se tienen que encontrar en los mismos sistemas que las bibliotecas y las unidades de cintas. El número máximo de gestores de claves depende de la biblioteca o el proxy. El único requisito es que la clave esté a disposición de las unidades de cintas por medio de conectividad TCP/IP.

Esto permite tener dos instancias de Encryption Key Manager que son imágenes duplicadas entre sí con una copia de seguridad integrada de la información crítica acerca de sus almacenes de claves, así como una migración tras error en caso de que uno de los gestores de clave deje de estar disponible. Cuando configure el dispositivo (o proxy), puede hacer que señale a dos gestores de claves. Si un gestor de claves deja de estar disponible por cualquier motivo, el dispositivo (o biblioteca de) se limitará a utilizar el gestor de claves alternativo.

Tiene también la capacidad de mantener los dos Encryption Key Manager sincronizados. Es muy importante que utilice esta función cuando sea necesario, tanto por la copia de seguridad inherente de datos fundamentales como por la función de migración tras error, que le ayudará a evitar interrupciones en el funcionamiento de las cintas. Consulte el apartado “Sincronización de datos entre dos servidores del gestor de claves” en la página 4-2.

Nota: La sincronización no incluye los almacenes de claves. Se deben copiar a mano.

Configuraciones del servidor Encryption Key Manager

Encryption Key Manager puede instalarse en uno o varios servidores. En los ejemplos siguientes se muestran configuraciones de un gestor de claves y de dos, pero es posible que la biblioteca admita más.

Configuración de servidor individual

Una configuración de con un solo servidor, como la que se muestra en la Figura 2-4, es la configuración de Encryption Key Manager más sencilla. Sin embargo, dada la falta de redundancia, no es recomendable. En esta configuración, todas las unidades de cintas se basan en un servidor individual del gestor de claves sin copia de seguridad. Si el servidor se cuelga, el almacén de claves, el archivo de configuración, el archivo KeyGroups.xml y la tabla de unidades dejarán de estar disponibles, lo que impediría leer la cinta cifrada. En una configuración con un solo servidor deberá comprobar que las copias de seguridad del almacén de claves, archivo de configuración, archivo KeyGroups.xml y tabla de unidades se guardan en lugar seguro, separadas de Encryption Key Manager, para que pueda reconstruirse su función en un servidor de sustitución si se pierden las copias del servidor.

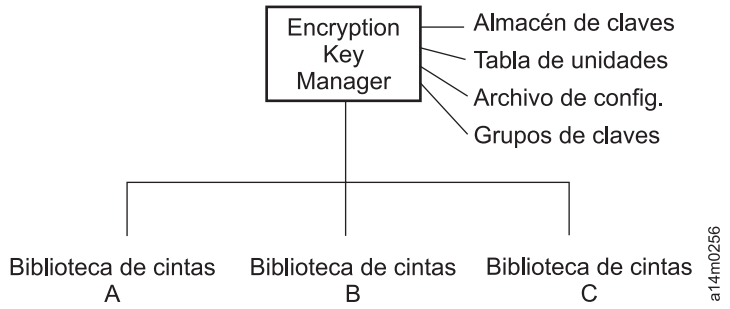

Figura 2-4. Configuración de servidor individual

Configuraciones de dos servidores

Se recomienda una configuración de dos servidores. Esta configuración de Encryption Key Manager realizará automáticamente una migración tras error al gestor de claves secundario si no puede accederse al primario por cualquier motivo.

Nota: Cuando se utilizan distintos servidores Encryption Key Manager para gestionar solicitudes del mismo conjunto de unidades de cintas, la información asociada en los almacenes de claves DEBE ser idéntica. Esto es necesario para que, al margen del servidor del gestor de claves con el que se establezca contacto, la información necesaria esté disponible para dar soporte a las solicitudes de las unidades de cintas.

Configuraciones idénticas: En un entorno con dos servidores Encryption Key Manager con configuraciones idénticas, como los que aparecen en la Figura 2-5, el proceso realizará automáticamente una migración tras error al gestor de claves secundario si el primario falla. En una configuración de este tipo, es fundamental que los dos servidores del gestor de claves estén sincronizados. Las actualizaciones sobre el archivo de configuración y la tabla de unidades de un servidor del gestor de claves se pueden reproducir en el otro automáticamente, utilizando el mandato **sync**, pero las actualizaciones de un almacén de claves se deben copiar en el otro utilizando métodos específicos de los almacenes de claves que se están utilizando. El archivo XML de los almacenes de claves y los grupos de claves deben copiarse manualmente. Consulte el apartado "Sincronización de datos entre dos servidores del gestor de claves" en la página 4-2 para obtener más información.

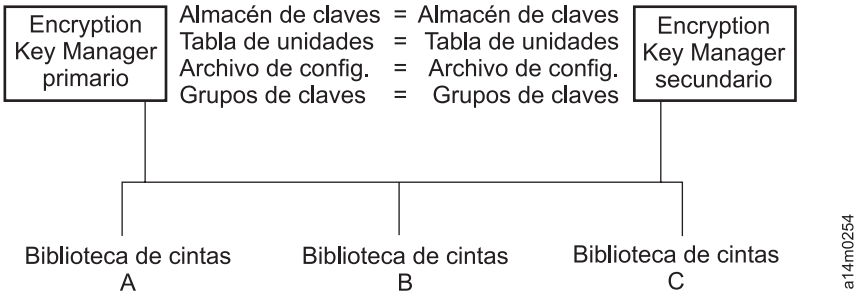

Figura 2-5. Dos servidores con configuraciones compartidas

Configuraciones individuales: Dos servidores Encryption Key Manager pueden compartir un almacén de claves y tabla de unidades común y tener dos archivos de configuración diferentes y dos conjuntos distintos de grupos de claves definidos en sus archivos XML. El único requisito es que las claves utilizadas para dar servicio a las unidades de cintas comunes sean las mismas para cada servidor. Esto permite que cada servidor del gestor de claves tenga su propio conjunto de propiedades. En este tipo de configuración, que se ve en la Figura 2-6, sólo se debe sincronizar la tabla de unidades entre los servidores del gestor de claves. (Consulte el apartado "Sincronización de datos entre dos servidores del gestor de claves" en la página 4-2 para obtener más información). Asegúrese de especificar `sync.type = drivetab` (no especifique config o all) para evitar que se sobrescriban los archivos de configuración.

Nota: No hay manera de compartir parcialmente la configuración entre servidores.

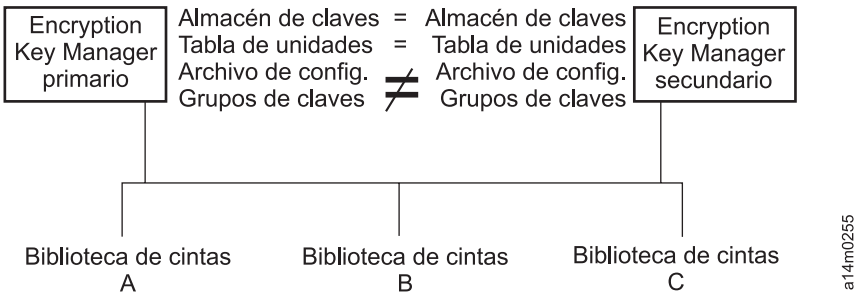

Figura 2-6. Dos servidores con configuraciones distintas accediendo a los mismos dispositivos

Consideraciones sobre el sitio de recuperación en caso de error

Si planea utilizar un sitio de recuperación en caso de error (DR), Encryption Key Manager proporciona un número de opciones para permitir que dicho sitio pueda leer y grabar cintas cifradas. Son las siguientes:

- Cree un Encryption Key Manager duplicado en el sitio DR.

  Establezca un Encryption Key Manager duplicado en el sitio DR con la misma información que en el Encryption Key Manager local (archivo de configuración, tabla de unidades de cintas, archivo XML de grupos de claves y el almacén de claves). Este gestor de claves está ahora preparado para encargarse de que uno de los gestores de claves de producción existentes lea y grabe cintas cifradas.
- Cree una copia de seguridad de los tres archivos de datos de Encryption Key Manager para poder recuperarlos en caso necesario.

  Si crea una copia actual de los cuatro elementos de datos necesarios para Encryption Key Manager (archivo de configuración, tabla de unidades de cintas, archivo XML de grupos clave y almacén de claves), podrá iniciar un gestor de claves en cualquier momento para que actúe como duplicado en el sitio DR. (Recuerde que no debe utilizar Encryption Key Manager para cifrar las copias de estos archivos ya que no podrá cifrarlas sin un gestor de claves operativo). Si el sitio de recuperación en caso de error utiliza distintas unidades de cintas del sitio primario, el archivo de configuración y la tabla de unidades deben contener la información correcta para el sitio de recuperación en caso de error.

Consideraciones para compartir cintas cifradas fuera del sitio

Nota: Es importante verificar la validez de los certificados recibidos de un business partner comprobando la cadena de confianza de dicho certificado hasta la entidad emisora de certificados (CA) que lo firmó. Si confía en la CA, puede fiarse del certificado. La validez del certificado también se puede verificar si se ha protegido con seguridad durante el tránsito. Si no se verifica la validez de un certificado de una de estas maneras se puede recibir un ataque de tipo “Man-in-the-Middle”.

Compartir cintas LTO 4 y LTO 5

Para compartir datos cifrados en una cinta LTO 4 o LTO 5, es necesario que la otra organización disponga de una copia de la clave simétrica utilizada para cifrar los datos en la cinta, lo que le permitirá leer la cinta. Para compartir la clave simétrica, la otra organización debe compartir su clave pública con usted. Esta clave pública será utilizada para empaquetar la clave simétrica cuando se exporta desde el almacén de claves de Encryption Key Manager utilizando keytool (consulte el apartado “Exportación de claves de datos utilizando Keytool -exportseckey ” en la página 3-13). Cuando la otra organización importa la clave simétrica en su almacén de claves de Encryption Key Manager, será desempaquetada utilizando su clave privada correspondiente (consulte el apartado “Importación de claves de datos utilizando Keytool -importseckey ” en la página 3-12). Así se garantiza que la clave simétrica esté segura durante el tránsito, dado que sólo el poseedor de la clave privada puede desempaquetar la clave simétrica. Con la clave simétrica utilizada para cifrar los datos en su almacén de claves de Encryption Key Manager, la otra organización podrá leer los datos en la cinta.

Consideraciones sobre el Estándar federal de procesamiento de la información (FIPS) 140-2

El Estándar federal de procesamiento de la información 140-2 ha ganado importancia ahora que el gobierno federal exige que todos sus proveedores de criptografía posean el certificado FIPS 140. Este estándar también se adopta cada vez en más empresas del sector privado. La certificación de las funciones criptográficas por parte de un tercero de acuerdo con los estándares gubernamentales ha mejorado la calidad en un mundo preocupado por la seguridad.

Encryption Key Manager no proporciona funciones criptográficas por sí mismo y, por lo tanto, no necesita ni puede obtener la certificación FIPS 140-2. Sin embargo, Encryption Key Manager utiliza las funciones criptográficas de IBM JVM en el componente IBM Java Cryptographic Extension y permite la selección y uso del proveedor criptográfico IBMJCEFIPS, que cuenta con una certificación FIPS 140-2 de nivel 1. Al establecer el parámetro de configuración **fips** en **on** en el archivo de propiedades de configuración, Encryption Key Manager utilizará el proveedor IBMJCEFIPS para todas las funciones criptográficas.

Consulte la documentación de proveedores específicos de hardware y software para obtener información sobre si sus productos han sido certificados por FIPS 140-2.

Capítulo 3. Instalación de Encryption Key Manager y almacenes de claves

Encryption Key Manager incluye la instalación de IBM Java Virtual Machine y requiere IBM Software Developer Kit para Linux, e IBM Runtime Environment para Windows (consulte el apartado "Requisitos de hardware y software" en la página 2-2). Siga el procedimiento indicado para el sistema operativo:

- "Instalación de Encryption Key Manager en Linux"
- "Instalación de Encryption Key Manager en Windows" en la página 3-2

Si no está seguro de tener la última versión de Encryption Key Manager, "Descarga de la última versión de la imagen ISO del gestor de claves" explica cómo determinar si existe una versión más reciente. Es una buena idea obtener la versión más reciente de Encryption Key Manager, que puede que no esté en su instalación de Java. Visite **http://support.dell.com** para obtener más información.

IMPORTANTE Encryption Key Manager INFORMACIÓN DE CONFIGURACIÓN DEL SERVIDOR DE HOST: es recomendable que las máquinas que alojan el programa Dell Encryption Key Manager utilicen memoria ECC para minimizar el riesgo de pérdida de datos. Encryption Key Manager realiza la función de solicitar la generación de claves de cifrado y de pasar dichas claves a las unidades de cintas LTO 4 y LTO 5. El material de las claves, en formato empaquetado (cifrado) reside en la memoria del sistema mientras es procesado por Encryption Key Manager. Tenga en cuenta que el material de las claves debe ser transferido sin errores a la unidad de cintas correspondiente para que los datos grabados en un cartucho puedan ser recuperados (descifrados). Si, por algún motivo, el material de las claves (en formato empaquetado o no) resulta dañado debido a un error de bit en la memoria del sistema y ese material de claves se utiliza para grabar datos en un cartucho, los datos escritos en el cartucho no podrán ser recuperados (no podrán ser descifrados posteriormente). Existen métodos para asegurar que tales errores no se producen. Sin embargo, si la máquina que aloja Encryption Key Manager no está utilizando memoria ECC (Código de corrección de errores), existe la posibilidad de que el materia de las claves resulte dañado mientras está en la memoria del sistema y que los daños provoquen una pérdida de datos. La probabilidad de que esto suceda es baja, pero siempre es recomendable que las máquinas que alojan aplicaciones vitales (como Encryption Key Manager) utilicen memoria ECC.

Descarga de la última versión de la imagen ISO del gestor de claves

Para descargar la última versión de la imagen ISO de Dell, vaya a http://support.dell.com.

Instalación de Encryption Key Manager en Linux

Instalación de Encryption Key Manager en Linux desde el CD

1. Inserte el CD de Dell Encryption Key Manager y especifique `Install_Linux` desde el directorio raíz del CD.

   La instalación copia todo el contenido (documentación, archivos de la GUI y archivos de propiedades de configuración) correspondiente al sistema operativo

desde el CD a la unidad de disco duro. Durante la instalación, se revisa el sistema para buscar el IBM Java Runtime Environment correcto. Si no se encuentra, se instala automáticamente.

Cuando la instalación se haya completado, se inicia la interfaz gráfica de usuario (GUI).

Instalación manual del Software Developer Kit en Linux

Siga estos pasos si no está realizando la instalación desde el CD.

1. Desde **http://support.dell.com**, descargue el entorno de ejecución correcto de para Java en función del sistema operativo:
   - Java 6 SR 5 (32 bits) o posterior
   - Java 6 SR 5 (64 bits) o posterior
2. Coloque el archivo rpm linux de Java en un directorio de trabajo:

```
mordor:~ #/tape/Encryption/java/1.6.0# pwd
/tape/Encryption/java/1.6.0
mordor:~ #/tape/Encryption/java/1.6.0# ls
ibm-java-i386-jre-6.0-5.0.i386.rpm
```

3. Instale el paquete rpm:

```
mordor:~ #rpm -ivh -nodeps ibm-java-i386-jre-6.0-5.0.i386.rpm
```

   Así, los archivos se colocarán en el directorio **/opt/ibm/java-i386-60/**:

```
mordor:~ #/opt/ibm/java-i386-60/jre # ls
.systemPrefs  bin  javaws  lib
```

4. Edite el archivo **/etc/profile.local** (o créelo si fuese necesario) con las variables JAVA_HOME, CLASSPATH y el directorio bin de la versión de Java instalada. Añada estas tres líneas:

```
JAVA_HOME=/opt/ibm/java-i386-60/jre
CLASSPATH=/opt/ibm/java-i386-60/jre/lib
PATH=$JAVA_HOME:opt/ibm/java-i386-60/jre/bin/:$PATH
```

5. Cierre la sesión y vuelva a iniciarla el host para que las entradas **/etc/profile.local** entren en vigor o emitan los mandatos de exportación de la línea de mandatos:

```
mordor:~ # export JAVA_HOME=/opt/ibm/java-i386-60/jre
mordor:~ # export CLASSPATH=/opt/ibm/java-i386-60/jre/lib
mordor:~ # export PATH=/opt/ibm/java-i386-60/jre/bin/:$PATH
```

6. Después de volver a iniciar sesión, emita el mandato **java -version**. Debería obtener estos resultados:

```
mordor:~ # java -version
java version "1.6.0"
Java(TM) SE Runtime Environment (build pmz60sr5-20090529(SR5))
IBM J9 VM (build 2.4, J2RE 1.6.0 IBM J9 2.4 Linux x86-32 jvmxi3260-20090519_35743 (JIT enabled)
...
mordor:~ # which java
/opt/ibm/java-i386-60/jre/bin/java
```

Instalación de Encryption Key Manager en Windows

1. Inserte el CD Dell Encryption Key Manager CD.

   La instalación copia todo el contenido (documentación, archivos de la GUI y archivos de propiedades de configuración) correspondiente al sistema operativo desde el CD a la unidad de disco duro. Durante la instalación, se revisa el sistema para buscar el IBM Java Runtime Environment correcto. Si no se encuentra, se instala automáticamente.

Cuando la instalación se haya completado, se inicia la interfaz gráfica de usuario (GUI).

2. Cuando se abra el asistente InstallShield Wizard, pulse **Next**.
3. Lea el Acuerdo de licencia y pulse **Yes**.
4. Cuando se abra la ventana Choose Destination Location (Figura 3-1), seleccione una carpeta y recuérdela. Necesitará esta vía de acceso de Java para iniciar Encryption Key Manager.

Figura 3-1. Ventana Choose Destination Location

Pulse **Next**.

5. Se abre una ventana que le pregunta si desea este Java Runtime Environment como JVM predeterminada del sistema (Figura 3-2).

Figura 3-2. Establezca esta versión de JVM como predeterminada

Pulse **No**.

6. Se abre la ventana Start Copying Files (Figura 3-3). Asegúrese de haber tomado nota del directorio de destino.

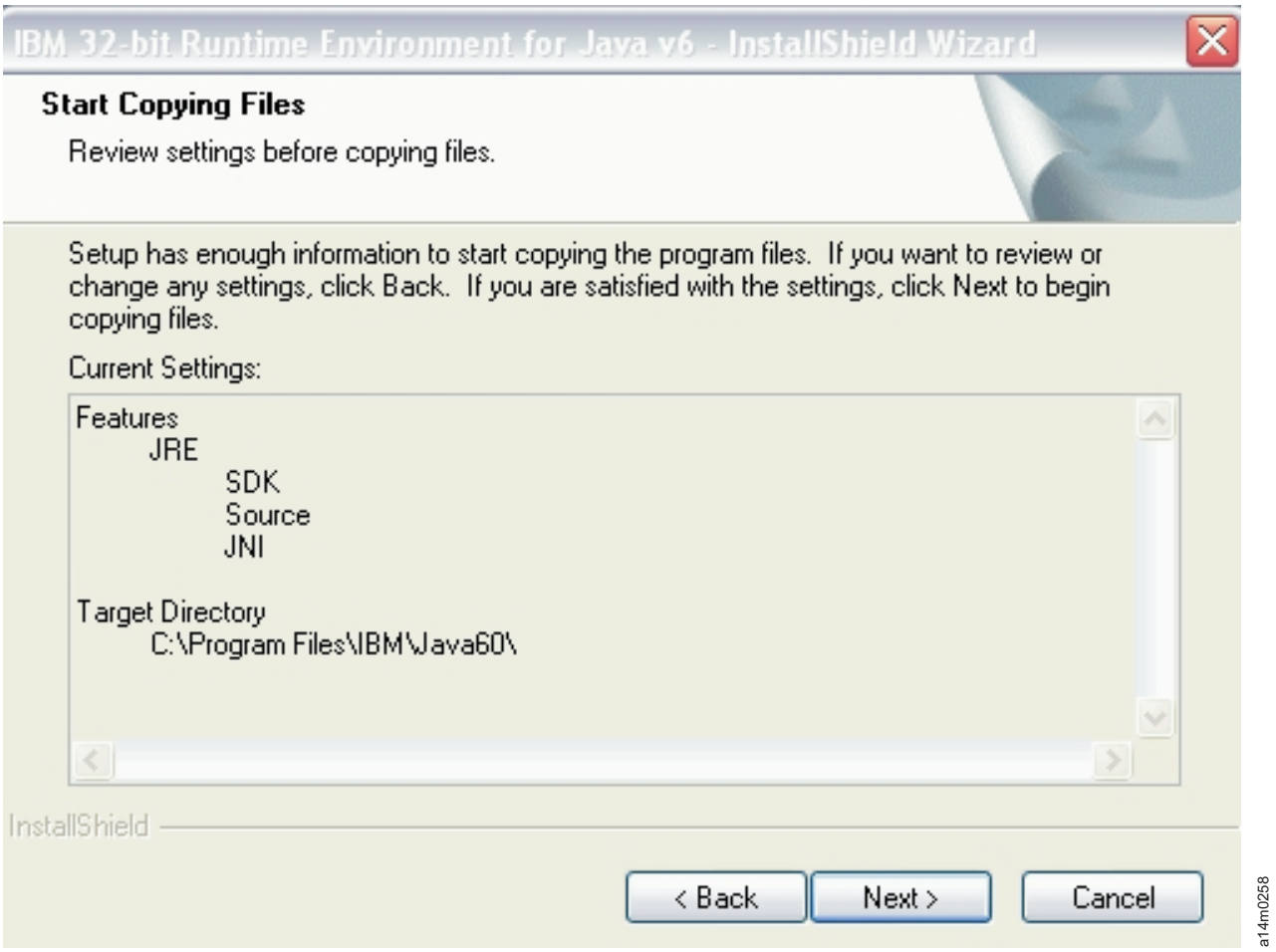

Figura 3-3. Ventana Start Copying Files

Pulse **Next**.

7. La ventana de estado indica el progreso de la instalación.
8. Se abre la ventana Browser Registration. Seleccione un navegador que utilizar con Encryption Key Manager. Pulse **Next**.
9. Cuando se abra la ventana InstallShield Wizard Complete, pulse **Finalizar**.

   Tras la instalación, puede abrir un indicador de mandatos para consultar la versión de Java instalada:

```
C:\WinEKM>C:\"Archivos de programa"\IBM\Java60\jre\bin\java -version
java version "1.6.0"
Java(TM) SE Runtime Environment (build pwi3260sr5-20090529_04(SR5))
IBM J9 VM (build 2.4, J2RE 1.6.0 IBM J9 2.4 Windows Server 2003 x86-32 j9vmwi3223-20090
519_35743 (JIT enabled, AOT enabled)
...
```

10. Actualice la variable PATH de la forma siguiente (necesario para Encryption Key Manager 2.1 pero opcional para si la fecha de compilación es 05032007 o anterior):

    Si va a invocar el SDK de Java desde una ventana de mandatos, es posible que desee establecer la variable PATH si desea tener la posibilidad de ejecutar los archivos ejecutables de JRE de Java (java.exe) desde cualquier directorio,

sin necesidad de especificar la vía de acceso completa del mandato. Si no establece la variable PATH, debe especificar la vía de acceso completa en el ejecutable cada vez que lo ejecute, como por ejemplo:

```
C:>\Archivos de programa\IBM\Java60\jre\bin\java ...
```

Para establecer PATH permanentemente (requerido para Encryption Key Manager 2.1), añada la vía de acceso completa al directorio bin de java a la variable PATH. Por lo general, esta vía de acceso completa tiene el siguiente aspecto:

```
C:\Archivos de programa\IBM\Java60\jre\bin
```

Para establecer la variable PATH de manera permanente en Microsoft Windows 2003, 2008 y 2008 R2:

Nota: El establecimiento de la variable PATH desde la línea de mandatos no funcionará.

a. En el menú Inicio, seleccione **Configuración** y, a continuación, **Panel de control**.
b. Efectúe una doble pulsación en **Sistema**.
c. Pulse en la pestaña **Avanzadas**.
d. Pulse **Variables de entorno**.
e. Desplace hacia abajo la lista Variables del sistema hasta encontrar la variable Path y pulse **Editar**.
f. Añada la vía de acceso de IBM JVM al principio de la variable PATH.

   El directorio de instalación predeterminado es C:\PROGRA~1\IBM\Java60\jre\bin.

   IMPORTANTE: inserte un punto y coma al final de la vía de acceso para diferenciarla del resto de los directorios en la lista de vías de acceso.
g. Pulse **Aceptar**.

Uso de la GUI para crear un archivo de configuración, un almacén de claves y certificados

Antes de iniciar Encryption Key Manager, deberá cerrar al menos un nuevo almacén de claves y al menos un certificado autofirmado. Puede utilizar la interfaz gráfica de usuario (GUI) de Dell Encryption Key Manager para crear el archivo de propiedades de configuración de Encryption Key Manager, un almacén de claves, certificado(s) y clave(s). También se crea un archivo de propiedades de configuración CLI sencillo como resultado de este proceso.

1. Abra la GUI si todavía no la ha iniciado:

   En Windows
   Vaya a c:\ekm\gui y pulse **LaunchEKMGui.bat**

   En plataformas Linux
   Vaya a /var/ekm/gui y especifique `. ./LaunchEKMGui.sh`
2. Seleccione **Configuración de EKM** en el navegador de la izquierda de la interfaz gráfica de usuario (GUI).

3. En la página “EKM Server Configuration” (Figura 3-4), especifique los datos en todos los campos obligatorios (indicados con un asterisco *). Algunos campos ya se han rellenado, por su comodidad. Pulse el signo de interrogación a la derecha de cualquier campo de datos para obtener una descripción. Pulse **Next**.

   Nota: Una vez que haya establecido la contraseña del almacén de claves, **no la cambie**, salvo que su seguridad haya sido vulnerada. Las contraseñas se ocultan para eliminar los riesgos de seguridad. El cambio de la contraseña del almacén de claves exige que cada contraseña de dicho almacén de claves se cambie individualmente utilizando el mandato **keytool**. Consulte el apartado “Cambio de contraseñas del almacén de claves” en la página 3-12.

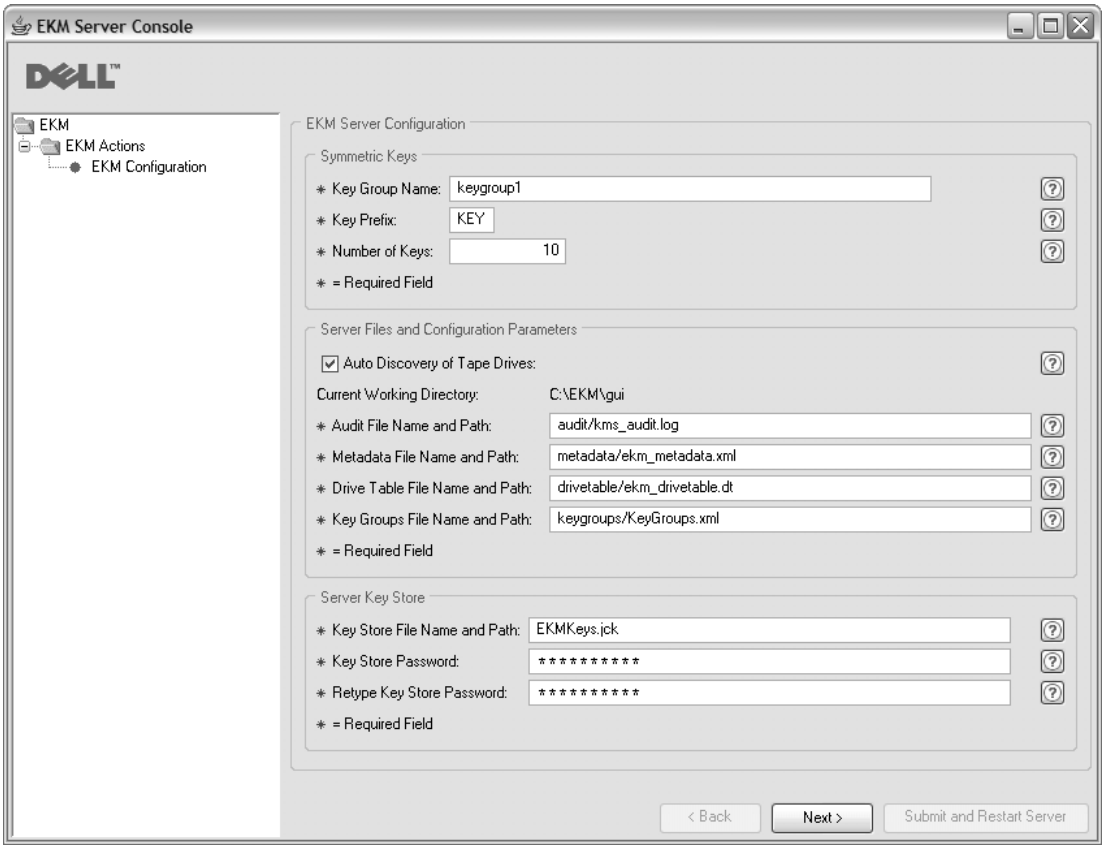

Figura 3-4. Página de configuración del servidor EKM

Aunque el número de claves que pueden generarse para el almacén de claves de Dell Encryption Key Manager es ilimitado, el tiempo necesario para generar las claves aumentará dependiendo del número de claves requerido. Encryption Key Manager tarda 15 segundos en generar 10 claves y unos 30 minutos en generar 10.000 claves. Tenga en cuenta que el número de claves está limitado por los recursos del servidor host (memoria en el servidor). La aplicación Encryption Key Manager mantiene la lista del almacén de claves en la memoria del sistema mientras se ejecuta para poder acceder rápidamente a las claves cuando la biblioteca envía una solicitud desde la unidad.

Nota: Interrumpir la GUI de Encryption Key Manager durante la generación de claves requerirá la reinstalación de Encryption Key Manager.

El archivo del almacén de claves resultará dañado si detiene el proceso de generación de claves de Encryption Key Manager antes de que termine. Para recuperarse de este suceso, siga estos pasos:

- Si se ha interrumpido Encryption Key Manager durante la instalación inicial de Encryption Key Manager, navegue hasta el directorio donde se encuentra Encryption Key Manager (por ejemplo, x:\ekm). Suprima este directorio y reinicie la instalación.
- Si Encryption Key Manager ha sido interrumpido cuando estaba añadiendo un nuevo grupo de claves, detenga el servidor Encryption Key Manager, restaure el archivo del almacén de claves con el almacén de claves de copia de seguridad más reciente (este archivo se encuentra en la carpeta x:\ekm\gui\backupfiles). Tenga en cuenta que el archivo de copia de seguridad contiene la indicación de la fecha y hora como parte del nombre de archivo (por ejemplo, 2007_11_19_16_38_31_EKMKeys.jck). La indicación de la fecha y hora deberá eliminarse una vez se copie el archivo en el directorio x:\ekm\gui. Reinicie el servidor Encryption Key Manager y añada el grupo de claves interrumpido anteriormente.

4. En la página "EKM Server Certificate Configuration", (Figura 3-5), especifique el alias del almacén de claves y cualquier otra información adicional que desee incluir. Pulse **Submit and Restart Server**.

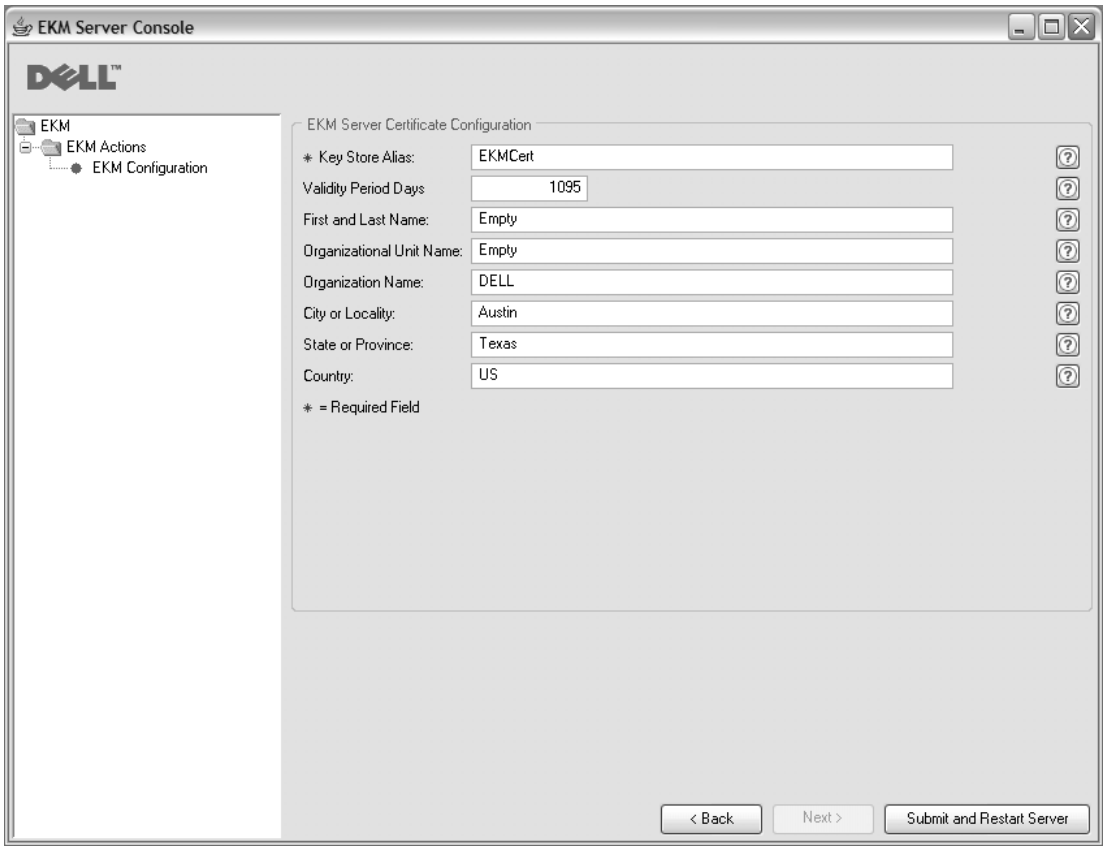

Figura 3-5. Página de configuración del certificado del servidor EKM

5. Se abrirá una ventana "Backup Critical Files" (Figura 3-6 en la página 3-8) que le recordará que debe realizar una copia de seguridad de sus archivos de datos de Encryption Key Manager.

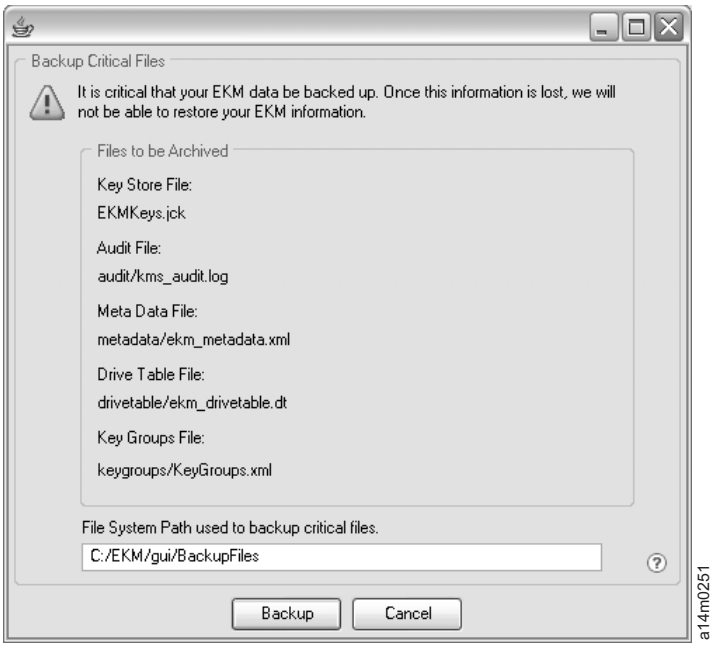

Figura 3-6. Ventana de copia de seguridad de archivos fundamentales

Verifique la vía de acceso y pulse **Backup**. El servidor Dell Encryption Key Manager se inicia en segundo plano.

Encryption Key Manager genera un conjunto de archivos de copia de seguridad cada vez que se pulsa **OK** al cambiar la configuración del servidor Encryption Key Manager o **Backup** en la ventana "Backup Critical Files". Los archivos listados como *Files to be Archived* se guardan en el directorio c:/ekm/gui/BackupFlies. Cada nombre de archivo lleva antepuesto la fecha y hora. Por ejemplo, un conjunto de archivos creados el 26 de noviembre de 2007 a las 14:58 y 46 segundos mostrará la siguiente indicación fecha y hora al antes de su nombre "2007_11_26_14_58_46_NombreArchivo. Los archivos de copia de seguridad no se sobregraban.

6. Seleccione **Server Health Monitor** en la ventana del navegador de la GUI para verificar que el servidor Encryption Key Manager está activo.

Para añadir claves a un almacén de claves existente, consulte el apartado "Utilización de la GUI para definir grupos de claves y crear claves" en la página 3-15.

Cómo encontrar la dirección IP del host correcta:

Es posible que las limitaciones en la GUI actual de Encryption Key Manager impidan que aparezca la dirección IP del host de Encryption Key Manager en Server Health Monitor:

- Si el host está configurado con una dirección IPv6, la aplicación Encryption Key Manager no podrá mostrar la dirección IP.
- Si la aplicación Encryption Key Manager está instalada en un sistema Linux, la aplicación Encryption Key Manager mostrará la dirección del host local y no el puerto IP activo real.

1. Para recuperar la dirección IP real del sistema host, encuentre la dirección del puerto IP accediendo a la configuración de red.
   - En un sistema Windows, abra una ventana de mandatos y escriba `ipconfig`.

- En Linux, escriba `isconfig`.

Cómo identificar el puerto SSL de EKM

1. Inicie el servidor Encryption Key Manager utilizando la línea de mandatos.
   - En Windows, haga cd `c:\ekm` y pulse **startServer.bat**
   - En las plataformas Linux, navegue a `/var/ekm` y especifique `startServer.sh`
   - Consulte el apartado "Inicio, Renovación y Detención del Servidor del gestor de claves" en la página 5-1 para obtener más información.
2. Inicie el cliente CLI utilizando la línea de mandatos.
   - En Windows, haga cd `c:\ekm` y seleccione **startClient.bat**
   - En las plataformas Linux, navegue a `/var/ekm` y especifique `startClient.sh`
   - Consulte el apartado "Cliente de la interfaz de línea de mandatos" en la página 5-5 para obtener más información.
3. Inicie sesión en un cliente CLI en el servidor Encryption Key Manager utilizando el siguiente mandato:

   `login –ekmuser` *`IDUsuario`* `–ekmpassword` *`contraseña`*

   donde *ID_usuario* = EKMAdmin y *contraseña* = changeME (esta es la contraseña predeterminada. Si ha cambiado anteriormente la contraseña predeterminada, utilice la nueva contraseña).

   Una vez haya iniciado correctamente la sesión, aparece el mensaje `User successfully logged in`.
4. Identifique el puerto SSL especificando el siguiente mandato:

   `status`

   La respuesta en pantalla debería ser similar a la siguiente: `server is running. TCP port: 3801, SSL port: 443`.

   Anote el puerto configurado SSL y compruebe que es el puerto utilizado para configurar los valores de cifrado de gestionados por biblioteca.
5. Cierre la sesión de la línea de mandatos. Especifique el siguiente mandato:

   `exit`

   Cierre la ventana de mandatos.

Generación de claves y alias para el cifrado en LTO 4 y LTO 5

La GUI del servidor Dell Encryption Key Manager es la manera más sencilla de generar claves de cifrado simétricas (consulte el apartado "Uso de la GUI para crear un archivo de configuración, un almacén de claves y certificados" en la página 3-5). También puede utilizar el programa de utilidad Keytool para generar las claves de cifrado simétrico. La solución Keytool resulta especialmente útil para importar y exportar claves entre distintos almacenes de claves. Consulte los apartados "Importación de claves de datos utilizando Keytool -importseckey " en la página 3-12 y "Exportación de claves de datos utilizando Keytool -exportseckey " en la página 3-13 para obtener información.

Keytool es un programa de utilidad para gestionar claves, certificados y alias. Le permite generar, importar y exportar las claves de datos de cifrado y almacenarlas en un almacén de claves.

A cada clave de datos del almacén de claves se accede mediante un alias exclusivo. Un alias es una serie de caracteres, como `123456tape`. En almacenes de claves

JCEKS, `123456Tape` sería equivalente a `123456tape` y permite acceder a la misma entrada en el almacén de claves. Cuando utilice el mandato **keytool -genseckey** para generar una clave de datos, especifique el alias correspondiente en el mismo mandato. El alias le permite identificar la clave correcta, en el grupo de claves y el almacén de claves correctos, para utilizarla en los datos cifrados de grabación y lectura en la cinta LTO 4 y LTO 5.

Nota: Los alias individuales y los intervalos de alias deben ser exclusivos. Esta condición se fuerza cuando se generan claves en una instancia de almacén de claves/Encryption Key Manager determinada. Sin embargo, en un entorno con varios Encryption Key Manager/almacenes de claves, debe utilizar un convenio de denominación que mantenga la exclusividad de los nombres en varias instancias en caso de que desee transportar claves entre instancias manteniendo la exclusividad de la referencia.

Después de generar claves y alias, actualizar la propiedad symmetricKeySet en el archivo KeyManagerConfig.properties para especificar el nuevo alias, el intervalo de alias o el GroupID del grupo de claves, el nombre de archivo bajo el que se almacenan las claves simétricas y el nombre de archivo en el que se definen los grupos de claves. (Consulte el apartado "Creación y gestión de grupos de claves" en la página 3-14 para obtener más información). Sólo las claves designadas en symmetricKeySet se validarán (se comprobará si hay un alias y una clave simétrica con el tamaño y el algoritmo adecuados). Si se especifica una clave no válida en esta propiedad, el gestor de claves no se inicia y se crea un registro de auditoría.

El programa de utilidad keytool facilita también la importación y exportación de claves de datos a otros almacenes de claves y desde ellos. A continuación se muestra una visión general de cada tarea. Puede emitir **keytool -ekmhelp** para visualizar todos los parámetros relacionados con el gestor de claves de los que se habla en los temas siguientes.

Edición de archivos de propiedades de configuración

Para realizar cambios en el archivo KeyManagerConfig.properties o ClientKeyManagerConfig.properties:

1. Detenga el servidor Encryption Key Manager.
2. Utilizando su editor de texto preferido, abra el archivo KeyManagerConfig.properties para realizar cambios en la configuración del servidor o el archivo ClientKeyManagerConfig.properties para realizar cambios en la configuración del cliente. No utilice Windows para editar el archivo de una máquina Linux debido a ^M. Si utiliza Windows, edite el archivo con gvim/vim.
3. Cambie el o los valores de propiedad de acuerdo con las instrucciones proporcionadas en este documento.
4. Guarde el archivo.
5. Reinicie el servidor Encryption Key Manager.

Si no utiliza Keytool

Si no utiliza keytool ni la GUI para generar claves y alias, no puede generar intervalos de claves compatibles con Encryption Key Manager. Para generar claves individuales compatibles con Encryption Key Manager, asegúrese de especificar los alias utilizando uno de los formatos siguientes:

- 12 caracteres imprimibles o menos (por ejemplo, `abcdefghijk`)

- 3 caracteres imprimibles, seguidos de dos ceros, seguidos de 16 dígitos hexadecimales (por ejemplo, ABC000000000000000001) para alcanzar un total de exactamente 21 caracteres

Generación de claves de datos y alias utilizando Keytool -genseckey

Nota: Antes de utilizar el mandato **keytool** por primera vez en una sesión, ejecute el script updatePath para establecer el entorno correcto.

En Windows
Vaya a cd c:\ekm y pulse **updatePath.bat**

En plataformas Linux
Vaya a /var/ekm y especifique **. ./updatePath.sh**

El programa de utilidad Keytool genera alias y claves simétricas para el cifrado en unidades de cintas LTO 4 y LTO 5 utilizando una cinta LTO 4 y LTO 5. Utilice el mandato **keytool -genseckey** para generar una o varias claves secretas y almacenarlas en un almacén de claves específico. **keytool -genseckey** adopta los siguientes parámetros:

```
-genseckey      [-v] [-protected]
                [-alias <alias> | aliasrange <intervalo_alias>] [-keypass <ctr_clave>]
                [-keyalg <alg_clave>] [-keysize <tmñ_clave>]
                [-keystore <almacén_claves>] [-storepass <ctr_almacén>]
                [-storetype <tipo_almacén>] [-providerName <nombre>]
                [-providerClass <nombre_clase_proveedor> [-providerArg <arg>] ...
                [-providerPath <lista_vías_acceso>]
```

Estos parámetros son especialmente importantes al generar claves de datos para que Encryption Key Manager las sirva a las unidades LTO 4 y LTO 5 para el cifrado de cintas:

-alias
Especifique un valor *alias* para una única clave de datos con hasta 12 caracteres de impresión (por ejemplo, abcfrg o key123tape).

-aliasrange
Cuando se generan varias claves de datos, *aliasrange* se especifica como un prefijo alfabético de 3 caracteres seguido de límites inferiores y superiores para una serie de cadenas de 16 caracteres (hexadecimales) con los ceros iniciales colocados automáticamente para crear alias de 21 caracteres de longitud. Por ejemplo, al especificar key1-a se produciría una serie de alias de KEY000000000000000001 a KEY00000000000000000A. Al especificar un valor *aliasrange* de xyz01-FF se produciría XYZ000000000000000001 a XYZ0000000000000000FF, lo que generaría 255 claves simétricas.

-keypass
Especifica una contraseña utilizada para proteger la clave de datos. Esta contraseña **debe ser idéntica** a la contraseña del almacén de claves. Si no especifica ninguna contraseña, se le solicitará. Si pulsa **Intro** en la solicitud, la contraseña de la clave se establecerá en la misma contraseña utilizada para el almacén de claves. *keypass* debe tener al menos seis caracteres de longitud.

Nota: Una vez que haya establecido la contraseña del almacén de claves, **no la cambie**, salvo que su seguridad haya sido vulnerada. Consulte el apartado "Cambio de contraseñas del almacén de claves".

-keyalg
Especifica el algoritmo que se utilizará para generar la clave de datos. Este valor se debe especificar como AES.

-keysize
Especifica el tamaño de la clave de datos que se generará. como 256.

Algunos ejemplos de alias aceptables que se pueden asociar con las claves simétricas son los siguientes:

```
abc000000000000000001
abc00a0120fa000000001
```

Algunos ejemplos de alias que no aceptará el gestor de claves son los siguientes:

```
abcefghij1234567 ? wrong length
abcg0000000000000001 ? prefix is longer than 3 characters
```

Si ya existe un alias en el almacén de claves, keytool genera una excepción y se detiene.

Cambio de contraseñas del almacén de claves

Nota: Una vez que haya establecido la contraseña del almacén de claves, **no la cambie**, salvo que su seguridad haya sido vulnerada. Las contraseñas se ocultan para eliminar los riesgos de seguridad. El cambio de la contraseña del almacén de claves exige que la contraseña de cada clave en dicho almacén de claves se cambie individualmente utilizando el siguiente mandato **keytool**.

Para cambiar la contraseña del almacén de claves, especifique:

```
keytool -keypasswd  -keypass antigua_contraseña -new nueva_contraseña -alias alias
        -keystore nombre_almacén_claves -storetype tipo_almacén_claves
```

Debe editar también el archivo KeyManagerConfig.properties para cambiar la contraseña del almacén de claves en cada archivo de configuración del servidor en el que se haya especificado utilizando uno de estos métodos:

- Suprima toda la contraseña oculta y permita que Encryption Key Manager le solicite una durante el siguiente inicio.
- Suprima la contraseña oculta completa y escriba la nueva contraseña en el espacio en blanco. Se ocultará la próxima vez que se inicie.

Importación de claves de datos utilizando Keytool -importseckey

Utilice el mandato keytool -importseckey para importar una clave secreta o un lote de claves secretas de un archivo de importación. **keytool -importseckey** adopta los siguientes parámetros:

```
-importseckey          [-v]
                [-keyalias <alias_clave>] [-keypass <cnt_clave>]
                [-keystore <almacén_claves>] [-storepass <ctr_almacén>]
```

[-storetype <tipo_almacén>] [-providerName <nombre>]
[-importfile <archivo_importación>] [-providerClass <nombre_clase_proveedor>]
[providerArg <arg>]

Estos parámetros son especialmente importantes al importar claves de datos para que Encryption Key Manager las sirva a las unidades LTO 4 y LTO 5 para el cifrado de cintas:

-keyalias
Especifica el alias de una clave privada del almacén de claves para descifrar todas las claves de datos de *importfile*.

-importfile
Especifica el archivo que contiene las claves de datos que se importarán.

Exportación de claves de datos utilizando Keytool -exportseckey

Utilice el mandato keytool -exportseckey para exportar una clave secreta o un lote de claves secretas a un archivo de exportación. **keytool -exportseckey** adopta los siguientes parámetros:

-exportseckey [-v]
[-alias <alias> | aliasrange <intervalo_alias>] [-keyalias <ctr_clave>]
[-keystore <almacén_claves>] [-storepass <ctr_almacén>]
[-storetype <tipo_almacén>] [-providerName <nombre>]
[-exportfile <archivo_exportación>] [-providerClass <nombre_clase_proveedor>]
[providerArg <arg>]

Estos parámetros son especialmente importantes al exportar claves de datos para que Encryption Key Manager las sirva a las unidades LTO 4 y LTO 5 para el cifrado de cintas:

-alias
Especifique un valor *alias* para una única clave de datos con hasta 12 caracteres de impresión (por ejemplo, `abcfrg` o `key123tape`).

-aliasrange
Cuando se exportan varias claves de datos, *aliasrange* se especifica como un prefijo alfabético de 3 caracteres seguido de límites inferiores y superiores para una serie de cadenas de 16 caracteres (hexadecimales) con los ceros iniciales colocados automáticamente para crear alias de 21 caracteres de longitud. Por ejemplo, al especificar `key1-a` se produciría una serie de alias de `KEY000000000000000001` a `KEY00000000000000000A`. Al especificar un valor *aliasrange* de `xyz01-FF` se produciría `XYZ000000000000000001` a `XYZ0000000000000000FF`

-exportfile
Especifica el archivo donde almacenar las claves de datos cuando se exportan.

-keyalias
Especifica el alias de una clave pública en el almacén de claves para cifrar todas las claves de datos. Asegúrese de que el almacén de claves al que se van a importar las claves (datos) simétricas contenga la clave privada correspondiente.

Configuración de alias de ejemplo y clave simétrica para el cifrado LTO 4 y LTO 5 utilizando un almacén de claves JCEKS

Invoque **KeyTool** con la opción `-aliasrange`.

Tenga en cuenta que se debe especificar el algoritmo de clave (-keyalg) como AES y el tamaño de clave (–keysize) se debe especificar como 256, tal y como se indica a continuación:

```
/bin/keytool –genseckey –v –aliasrange AES01-FF –keyalg AES –keysize 256
–keypass contraseña -storetype jceks –keystore vía_acceso/nombre_archivo.jceks
```

Estas invocaciones de KeyTool generan 255 alias secuenciales en el intervalo AES0000000000000000001 a AES0000000000000000FF y claves simétricas AES asociadas de 256 bits. Se pueden repetir de manera acumulativa tantas veces como sea necesario para configurar el número completo de alias de claves autónomos del intervalo que se desea obtener para un funcionamiento sólido del gestor de claves. Por ejemplo, para generar un alias adicional y una clave simétrica para LTO 4 y LTO 5:

```
/bin/keytool –genseckey –v –alias abcfrg –keyalg AES –keysize 256
–keypass contraseña -storetype jceks –keystore vía_acceso/nombre_archivo.jceks
```

Esta invocación añade el alias autónomo abcfrg de manera acumulativa al almacén de claves designado, que ya contiene 255 alias de la invocación anterior, lo que produce que haya 256 claves simétricas en el archivo jceks designado en la opción –keystore.

Actualice la propiedad symmetricKeySet del archivo KeyManagerConfig.properties para añadir la siguiente línea y que coincida con alguno o con todos los intervalos de alias utilizados más arriba, y el nombre de archivo bajo el que se almacenaron las claves simétricas. Tenga en cuenta que es posible que Encryption Key Manager no se inicie si se especifica un alias no válido. Otros motivos para que falle la comprobación de validación pueden ser un tamaño de bits incorrecto (para AES el tamaño de clave DEBE ser 256) o un algoritmo no válido para la plataforma. -keyalg debe ser AES y -keysize debe ser 256. El nombre de archivo especificado en **config.keystore.file** debe coincidir con el nombre especificado en –keystore <nombre_archivo> en la invocación de KeyTool:

```
symmetricKeySet = AES01-FF,abcfrg
config.keystore.file = <nombre_archivo>.jceks
```

Sólo las claves designadas en symmetricKeySet se validarán (se comprobará si hay un alias y una clave simétrica con el tamaño y el algoritmo adecuados). Si se especifica una clave no válida en esta propiedad, Encryption Key Manager no se iniciará y se creará un registro de auditoría.

Creación y gestión de grupos de claves

Encryption Key Manager proporciona la capacidad de organizar las claves simétricas para el cifrado de LTO 4 y LTO 5 en grupos de claves. De este modo, puede agrupar las claves según el tipo de datos que cifren, los usuarios que tengan acceso a ellas, o por otras características significativas. Cuando se crea un grupo de claves, se puede asociar con una unidad de cintas específica utilizando la palabra clave -symrec en el mandato **adddrive**. Consulte el mandato “adddrive” en la página 5-8 para ver la sintaxis.

Para crear un grupo de claves, debe definirlo en el archivo KeyGroups.xml. Si ha seguido el procedimiento del apartado “Uso de la GUI para crear un archivo de configuración, un almacén de claves y certificados” en la página 3-5, la ubicación de este archivo se habrá especificado en la página de configuración de EKM. Si

está creando el archivo de configuración de manera manual, la ubicación del archivo KeyGroups.xml se especifica en el archivo de propiedades de configuración tal y como se indica a continuación:

```
config.keygroup.xml.file = FILE:KeyGroups.xml
```

Si no se especifica este parámetro, el comportamiento predeterminado es utilizar el archivo KeyGroups.xml en el directorio de trabajo de la ubicación de inicio de Encryption Key Manager. Si este archivo no existe, se creará un archivo KeyGroups.xml vacío. En los siguientes inicios del servidor Encryption Key Manager, es posible que aparezca el siguiente mensaje en **native_stderr.log**: `[Fatal Error] :-1:-1: Premature end of file.` Este es un error que se produce al analizar el archivo KeyGroups.xml vacío y no impide que el servidor Encryption Key Manager se inicie a no ser que el servidor Encryption Key Manager haya sido configurado para utilizar grupos de claves.

Los grupos de claves se crean utilizando la GUI del servidor Dell Encryption Key Manager o utilizando los siguientes mandatos del cliente CLI (consulte el apartado "Mandatos CLI" en la página 5-8 para obtener la sintaxis):

Utilización de la GUI para definir grupos de claves y crear claves

Puede utilizar la GUI para realizar todas las tareas necesarias para gestionar grupos de claves. También puede utilizarla para crear claves adicionales.

Nota: Si pulsa **Submit Changes** al realizar cualquiera de las siguientes tareas, se abrirá una ventana de diálogo de copias de seguridad (Figura 3-6 en la página 3-8) recordándole que realice una copia de seguridad de sus archivos de datos de Encryption Key Manager. Especifique una vía de acceso para guardar los datos de la copia de seguridad. Pulse **Submit**. A continuación, verifique la vía de acceso de la copia de seguridad y pulse **OK**.

Para crear un grupo de claves y llenarlo con claves, o para añadir claves a un grupo de claves existente:

1. Abra la GUI si todavía no la ha iniciado:

   En Windows
   Vaya a c:\ekm\gui y pulse **LaunchEKMGui.bat**

   En plataformas Linux
   Vaya a /var/ekm/gui y especifique `. ./LaunchEKMGui.sh`
2. Seleccione **Administration Commands** en el navegador situado a la izquierda de la GUI.
3. Pulse **Create a Group of Keys** en la parte inferior de la ventana (Figura 3-7 en la página 3-16).

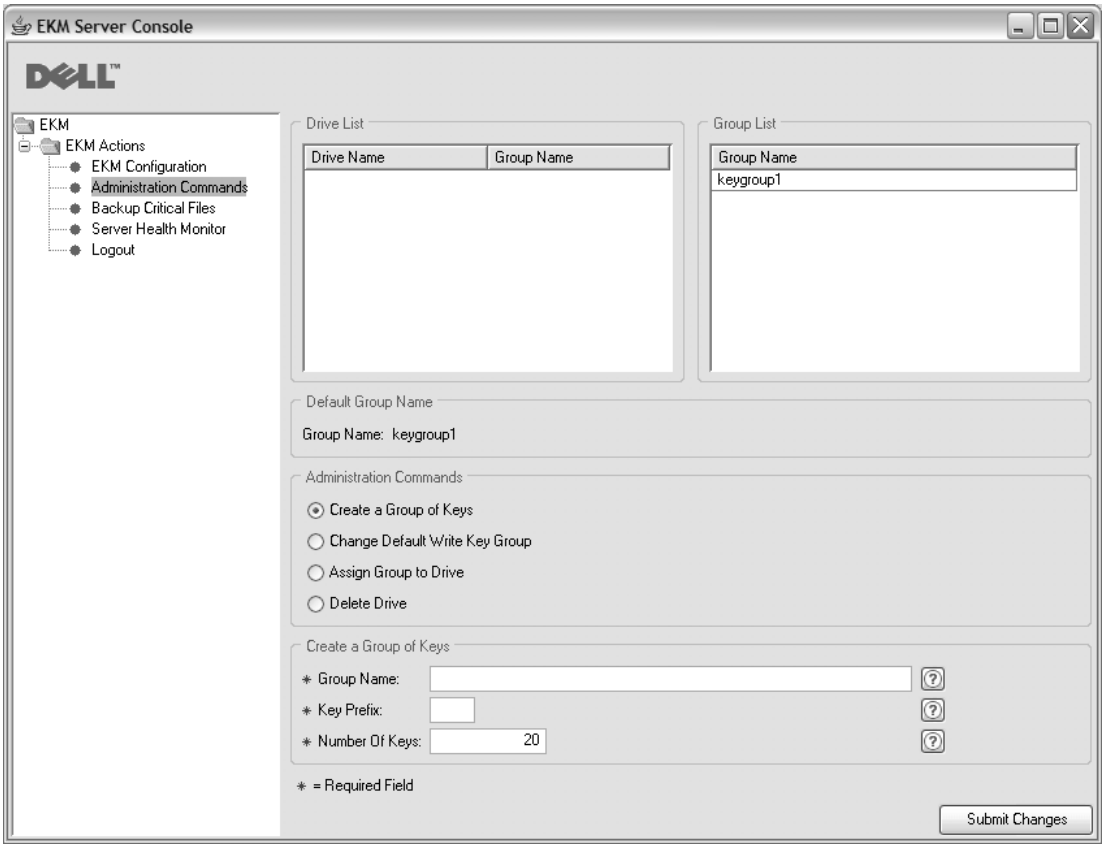

Figura 3-7. Creación de un grupo de claves

4. Especifique el nombre del nuevo grupo de claves, el prefijo que se utilizará para los alias de claves y el número de claves que va a contener el grupo. Pulse **Submit Changes**.

Para cambiar el grupo de claves predeterminado

1. Seleccione **Administration Commands** en el navegador situado a la izquierda de la GUI.
2. Pulse **Change Default Write Key Group** en la parte inferior de la ventana (Figura 3-8 en la página 3-17).

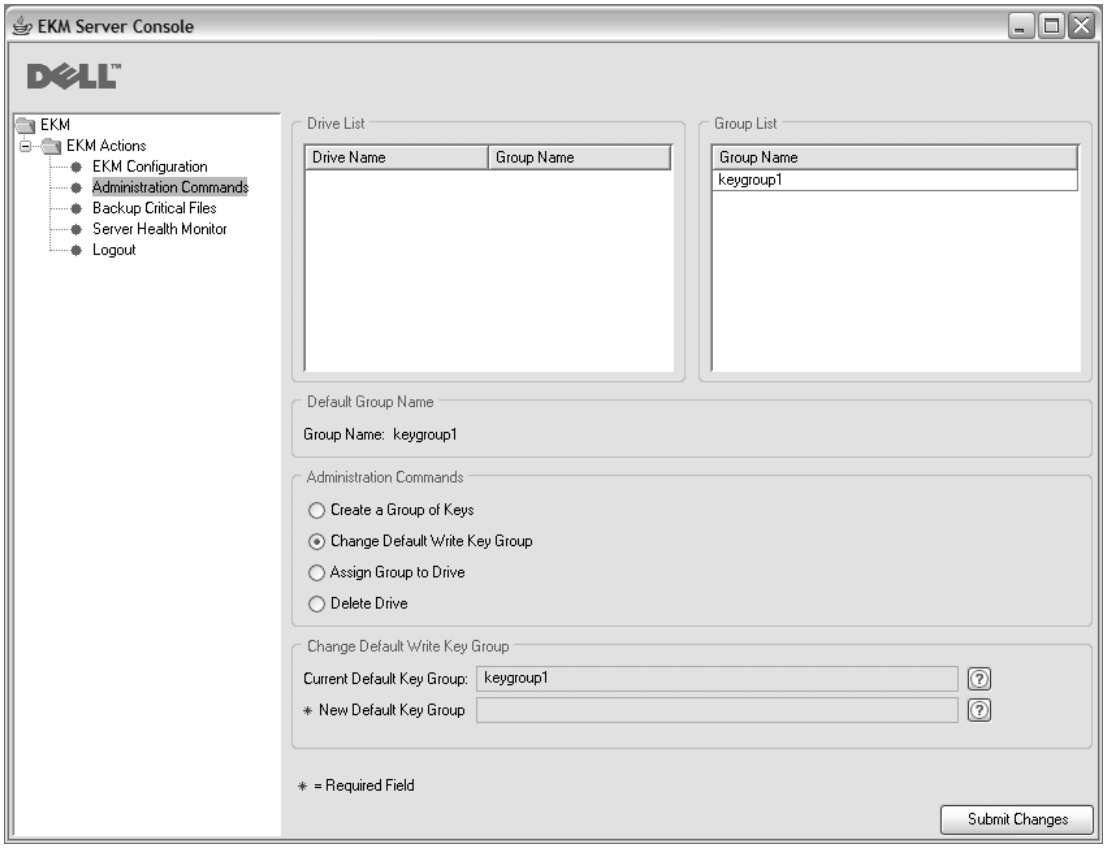

Figura 3-8. Cambio del grupo de claves de escritura predeterminado

3. Seleccione el nuevo grupo de claves predeterminado de la lista Group List, a la derecha.
4. Verifique el grupo de claves actual y el nuevo y predeterminado, en la parte inferior de la ventana, y pulse **Submit Changes**.

Para asignar un grupo de claves específico a una unidad de cintas específica:

1. Seleccione **Administration Commands** en el navegador situado a la izquierda de la GUI.
2. Pulse **Assign Group to Drive** en la parte inferior de la ventana (Figura 3-9 en la página 3-18).

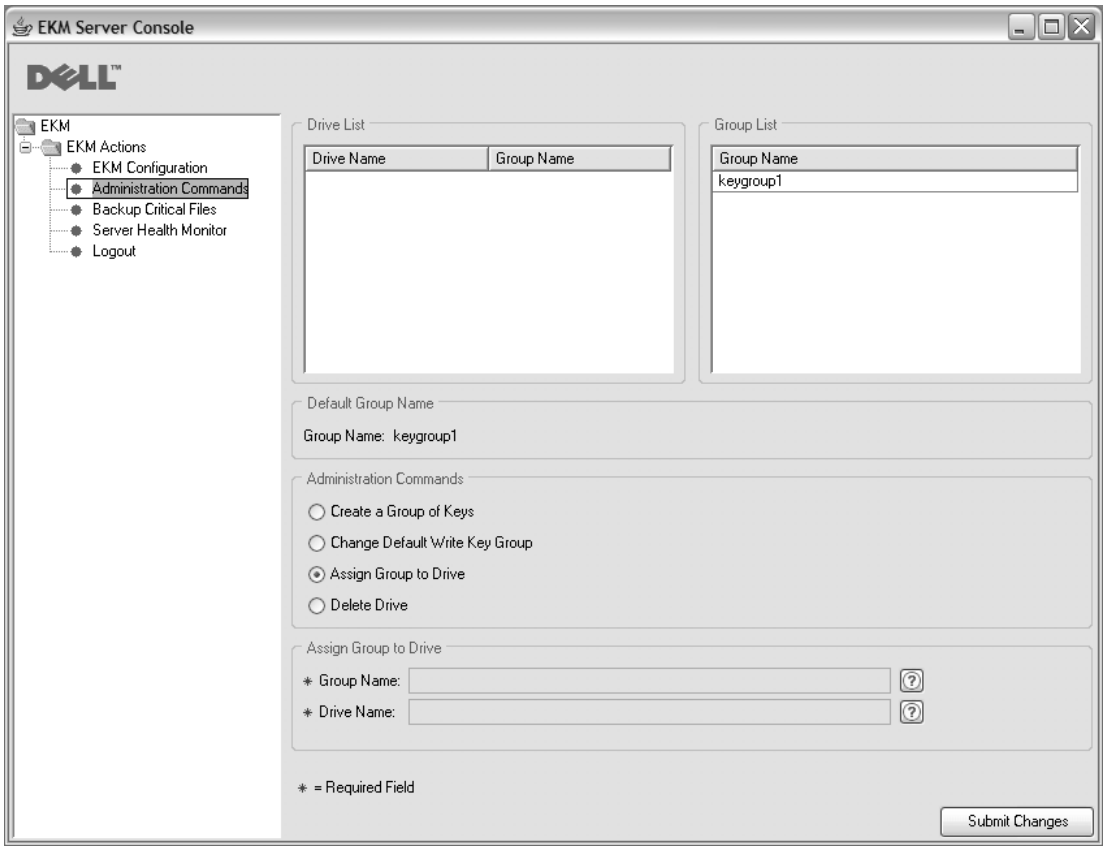

Figura 3-9. Asignación de un grupo a una unidad

3. Seleccione la unidad de cintas de la lista Drive List.
4. Seleccione el grupo de claves de la lista Group List.
5. Verifique el grupo de unidades y claves en la parte inferior de la ventana y pulse **Submit Changes**.

Para suprimir una unidad de cintas de la tabla de unidades:

1. Seleccione **Administration Commands** en el navegador situado a la izquierda de la GUI.
2. Pulse **Delete Drive** en la parte inferior de la ventana (Figura 3-10 en la página 3-19).

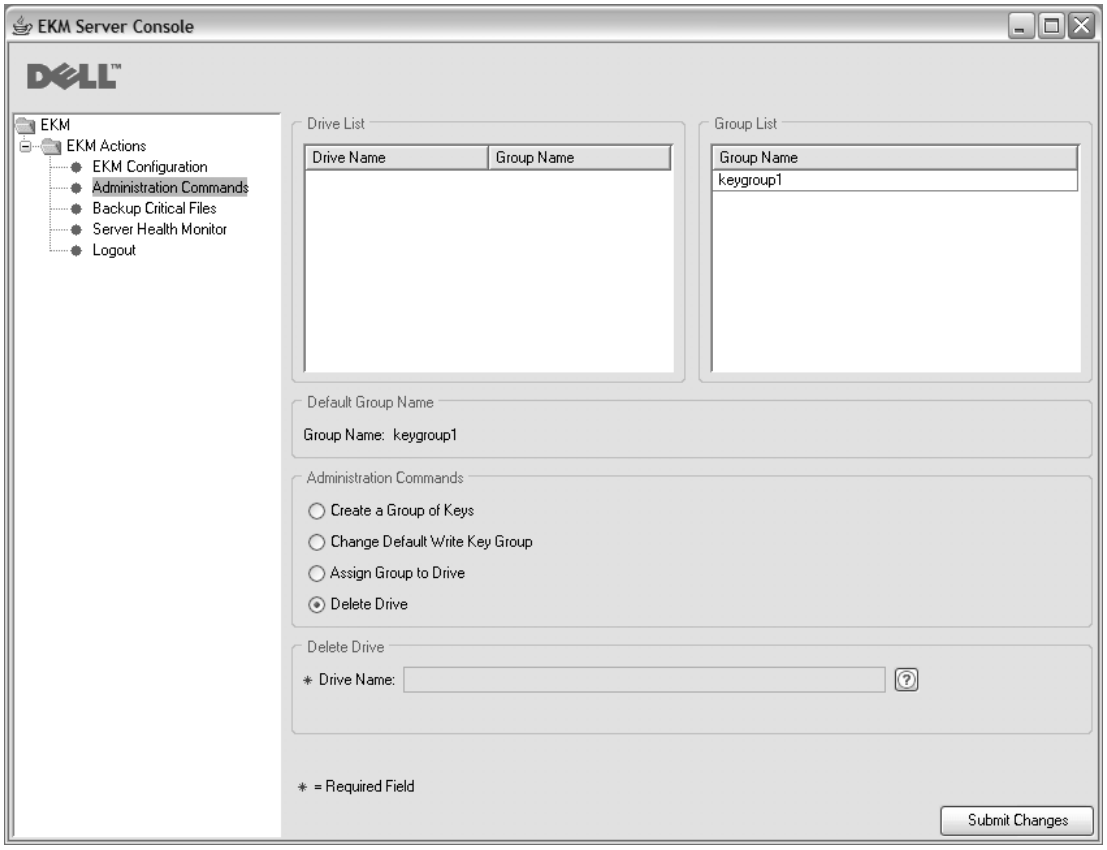

Figura 3-10. Supresión de una unidad

3. Seleccione la unidad de cintas de la lista Drive List.
4. Verifique el nombre de la unidad en la parte inferior de la ventana y pulse **Submit Changes**.

Utilización de los mandatos CLI para definir grupos de claves

Encryption Key Manager tiene una característica de grupo de claves que permite agrupar conjuntos de claves.

Una vez instalada y configurada la aplicación Encryption Key Manager (almacén de claves y claves generadas) y se ha iniciado el servidor Encryption Key Manager, inicie la sesión en el servidor utilizando el cliente y siga estos pasos:

1. Ejecute el mandato **createkeygroup**.

   Este mandato crea el objeto de grupo de claves inicial en el archivo KeyGroups.xml. Ejecútelo sólo una vez.

   Sintaxis: **createkeygroup -password** *contraseña*

   -password
   Contraseña utilizada para cifrar la contraseña del almacén de claves en el archivo KeyGroups.xml, con el fin de recuperarla más adelante. El almacén de claves cifra la clave del grupo de claves, que cifra a cambio cada contraseña del alias de grupo de claves individual. Por lo tanto, ninguna de las claves del archivo KeyGroups.xml se encuentra en la copia no cifrada.

   Ejemplo: createkeygroup -password a75xynrd
2. Ejecute el mandato **addkeygroup**.

Este mandato crea una instancia de un grupo de claves con un ID de grupo exclusivo en el archivo KeyGroups.xml.

Sintaxis: **addkeygroup -groupID** *nombre_grupo*

-groupID
Nombre_grupo exclusivo utilizado para identificar el grupo en el archivo KeyGroups.xml.

Ejemplo: addkeygroup -groupID keygroup1

3. Ejecute el mandato **addkeygroupalias**.

   Este mandato crea un nuevo alias para un alias de clave existente en el almacén de claves, para añadirlo a un ID de grupo de claves específico.

   Sintaxis: **addkeygroupalias -alias** *nombre_alias* **-groupID** *nombre_grupo*

   -alias
   Nuevo *nombre_alias* de la clave. Debe ser el nombre de clave completo, es decir Key00 se debe especificar como key000000000000000000.

   -groupID
   Nombre_grupo exclusivo utilizado para identificar el grupo en el archivo KeyGroups.xml.

   Ejemplo: addkeygroupalias -alias key000000000000000000 -groupID keygroup1

   Nota: Si utiliza el mandato CLI sólo puede añadir una clave a la vez. Este mandato se debe ejecutar para cada clave individual que necesite añadirse al grupo de claves.

4. Asocie un grupo de claves a una unidad de cintas nueva o existente.
   a. Ejecute el mandato **moddrive** para asociar un grupo de claves a una unidad de cintas existente.

      Este mandato modifica la información de la unidad de cintas en la tabla de unidades.

      Sintaxis: **moddrive -drivename** *nombre_unidad* **-symrec** *alias*

      -drivename
      nombre_unidad especifica el número de serie de la unidad de cintas.

      -symrec
      Especifica un *alias* (de la clave simétrica) o un nombre de grupo de claves para la unidad de cintas.

      Ejemplo: moddrive -drivename 000123456789 -symrec keygroup1

   b. Ejecute el mandato **adddrive** para añadir una unidad de cintas a la tabla de unidades y asociarla a un grupo de claves.

      Este mandato le permite añadir una unidad y asociarla a un grupo de claves específico.

      Sintaxis: **adddrive -drivename** *nombre_unidad* **-symrec** *alias*

      -drivename
      nombre_unidad especifica el número de serie de 12 dígitos de la unidad que se va a añadir.

      Nota: Debe añadir dos ceros (0) iniciales delante del número de serie de 10 dígitos para alcanzar un total de 12 dígitos.

      -symrec
      Especifica un *alias* (de la clave simétrica) o un ID de grupo para la unidad de cintas.

      Ejemplo: adddrive -drivename 000123456789 -symrec keygroup1

Para especificar un grupo de claves como predeterminado para su uso cuando no se haya definido ningún alias para una unidad de cintas, establezca la propiedad symmetrickeySet del archivo de propiedades de configuración en el GroupID del grupo de claves que desee utilizar. Por ejemplo,

```
symmetricKeySet = keygroup1
```

GroupID debe coincidir con un ID de grupo de claves existente en el archivo KeyGroups.xml. De lo contrario, el servidor Encryption Key Manager no se iniciará. Encryption Key Manager rastrea el uso de las claves dentro de un grupo de claves. Si especifica un GroupID válido, Encryption Key Manager registrará qué clave fue utilizada por última vez y, a continuación, selecciona una clave aleatoria en el interior del grupo de claves.

Copia de claves de un grupo de claves a otro

Ejecute el mandato **addaliastogroup**.

Este mandato copia un alias específico de un grupo de claves existente (origen) a un nuevo grupo de claves (destino).

Sintaxis: **addaliastogroup -aliasID** *nombre_alias* **-sourceGroupID** *nombre_grupo* **-targetGroupID** *nombre_grupo*

-aliasID
Nombre_alias de la clave que se va a añadir.

-sourceGroupID
Nombre_grupo exclusivo utilizado para identificar el grupo desde el que se va a copiar el alias.

-targetGroupID
Nombre_grupo exclusivo utilizado para identificar el grupo al que se va a añadir el alias.

Ejemplo: addaliastogroup -aliasID aliasname -sourceGroupID keygroup1 -targetGroupID keygroup2

Nota: La clave está disponible en ambos grupos de claves.

Capítulo 4. Configuración de Encryption Key Manager

Utilización de la GUI para configurar Encryption Key Manager

La manera más sencilla de crear su archivo de propiedades de configuración es utilizar la GUI de Dell Encryption Key Manager siguiendo el procedimiento del apartado "Uso de la GUI para crear un archivo de configuración, un almacén de claves y certificados" en la página 3-5. Si lo ha hecho, ya ha creado el archivo de configuración y no necesita realizar ninguna tarea de configuración adicional. La siguiente información puede resultarle útil si desea utilizar opciones de configuración adicionales de Encryption Key Manager.

Estrategias de configuración

Algunos valores de configuración del archivo KeyManagerConfig.properties ofrecen atajos que pueden tener efectos que el usuario debe conocer.

Actualización automática de la tabla de unidades de cintas

Encryption Key Manager proporciona una variable en el archivo de configuración (drive.acceptUnknownDrives) que, cuando se establece en un valor de `true`, rellena automáticamente la tabla de unidades de cintas cuando una nueva unidad de cintas contacta con Dell Encryption Key Manager. Así se elimina la necesidad de utilizar el mandato **adddrive** para cada biblioteca o unidad de cintas. De este modo, no es necesario especificar el número de serie de 10 dígitos de cada uno de estos dispositivos utilizando los mandatos de cliente CLI. Las nuevas unidades pasan por el intercambio normal de criptografía de clave pública/privada para verificar la identidad del dispositivo de cintas. Cuando se haya completado esta verificación, el nuevo dispositivo podrá leer las cintas existentes en función de los ID de clave almacenados en ellas (se presupone que la información de clave correspondiente se encuentra en el almacén de claves configurado).

Nota: El servidor Encryption Key Manager se debe renovar utilizando la GUI o el mandato "refresh" en la página 5-14 después de añadir automáticamente unidades para comprobar que se han almacenado en la tabla de unidades.

Para las unidades LTO 4 y LTO 5, puede establecer la agrupación de clave simétrica predeterminada (symmetricKeySet) para el cifrado de los dispositivos recién añadidos. En otras palabras, puede hacer que Encryption Key Manager configure completamente el dispositivo con el material de claves asociado cuando el dispositivo realiza el contacto. Si decide no hacer esto cuando el dispositivo se añada a la tabla de unidades, puede hacerlo después de añadir la unidad de cintas a la tabla de unidades de cintas, utilizando el mandato **moddrive**.

Además de evitar que el administrador tenga que especificar el número de serie de 10 dígitos para cada una de las unidades de cintas a las que dará servicio Encryption Key Manager, también permite crear un entorno predeterminado para configuraciones de sistemas de gran tamaño.

Debe tenerse en cuenta que este tipo de comodidad se obtiene a cambio de una seguridad menor. Dado que los dispositivos se añaden automáticamente y se pueden asociar con un alias de certificado (capaz de grabar una cinta con dicho alias de certificado), la comprobación de seguridad añadida verifica que la tarea

que el administrador realizaría al añadir dispositivos manualmente se omita. Es importante que evalúe las ventajas y desventajas de esta opción para determinar si añadir automáticamente la información de la unidad de cintas a la tabla de unidades y otorgar implícitamente a los nuevos dispositivos acceso a la información del certificado es un riesgo de seguridad aceptable.

Nota: La propiedad drive.acceptUnknownDrives se establece, de manera predeterminada, en `false`. Por lo tanto, Encryption Key Manager no añadirá nuevas unidades a la tabla de unidades automáticamente. Seleccione la modalidad en la que desee trabajar y cambie la configuración en consecuencia. Consulte el Apéndice B para obtener más información.

Sincronización de datos entre dos servidores del gestor de claves

Es posible sincronizar la tabla de unidades y el archivo de propiedades de configuración entre dos servidores Encryption Key Manager. Esto se puede hacer de manera manual utilizando el mandato **sync** del cliente CLI, o automáticamente, estableciendo cuatro propiedades en el archivo KeyManagerConfig.properties.

Notas

Ningún método de sincronización actúa en el archivo XML del almacén de claves o el grupo de claves. Se deben copiar a mano.

La función de sincronización automática sólo se habilita cuando se especifica una dirección IP válida en la propiedad sync.ipaddress del archivo KeyManagerConfig.properties. Consulte el apartado "Sincronización automática" en la página 4-3.

Sincronización manual

El método manual implica la ejecución del mandato **sync** del cliente CLI. La sintaxis es la siguiente:

sync {**-all** | **-config** | **-drivetab**} **-ipaddr** *dir_ip :puerto_ssl* [**-merge** | **-rewrite**]

Este mandato envía las propiedades del archivo de configuración, la información de la tabla de unidades, o ambas, desde el servidor de origen (o emisor) al servidor de destino (o receptor) especificado por el parámetro **–ipaddr**. El servidor Encryption Key Manager receptor debe estar activo y en ejecución.

Campos obligatorios

-all
Enviar el archivo de propiedades de configuración y la información de la tabla de unidades al servidor especificado por **-ipaddr**.

-config
Enviar sólo el archivo de propiedades de configuración al servidor especificado por **-ipaddr**.

-drivetab
Enviar solo la información de la unidad de tabla al servidor especificado por **-ipaddr**.

-ipaddr
dir_ip:puertossl especifica la dirección y el puerto ssl del servidor receptor.

puertossl debe coincidir con el valor especificado para "TransportListener.ssl.port" en el archivo KeyManagerConfig.properties del servidor receptor.

Campos opcionales

-merge
Fusionar (añadir) los nuevos datos de la tabla de unidades con los datos actuales en el servidor receptor. (El archivo de configuración siempre es una regrabación). Este es el valor predeterminado.

-rewrite
Sustituir los datos actuales del servidor receptor por datos nuevos

Sincronización automática

La tabla de unidades y el archivo de propiedades se pueden enviar automáticamente desde un servidor del gestor de claves primario a un servidor secundario. El servidor secundario se debe estar ejecutando para que se produzca la sincronización de datos. Para sincronizar automáticamente los datos del servidor primario al secundario, se deben especificar las cuatro siguientes propiedades en el archivo KeyManagerConfig.properties del servidor primario. No es necesario realizar cambios sobre el archivo de propiedades del servidor secundario o receptor.

sync.ipaddress
Especifica la dirección y el puerto SSL del servidor receptor, por ejemplo,

```
sync.ipaddress = backupekm.server.ibm.com:1443
```

Si esta propiedad no se especifica o se especifica de manera incorrecta, se inhabilita la sincronización automática.

sync.action
Fusionar o regrabar los datos existentes en el servidor receptor. Los valores válidos son **merge** (predeterminado) y **rewrite**. La sincronización de las propiedades de configuración siempre produce una regrabación.

sync.timeinhours
Frecuencia con la que deben enviarse los datos. El valor se especifica en números enteros (horas). El intervalo de tiempo empieza cuando se inicia el servidor, es decir, la sincronización se producirá después de que el servidor se haya ejecutado durante un determinado número de horas. El valor predeterminado es 24.

sync.type
Datos que se deben enviar. Los valores válidos son **drivetab** (predeterminado), **config** y **all**.

Datos generales de la configuración

Nota: Si ha seguido el procedimiento del apartado "Uso de la GUI para crear un archivo de configuración, un almacén de claves y certificados" en la página 3-5, se habrá creado una configuración básica y no es necesario realizar ninguno de estos pasos. Esta información muestra cómo realizar estas tareas sin utilizar la GUI, y puede ser útil si desea sacar provecho de las opciones de configuración adicionales.

Nota para los usuarios de Windows: Windows no acepta mandatos con vías de acceso de directorios que contienen espacios en blanco. Cuando se especifican mandatos, puede que sea necesario especificar el nombre abreviado generado para dichos directorios, por ejemplo `Archiv~1` en lugar de `Archivos de programa`. Para listar los nombres abreviados de directorios, emita el mandato **dir /x**.

Este procedimiento contiene los pasos mínimos necesarios para configurar Encryption Key Manager. El Apéndice A incluye ejemplos de los archivos de propiedades de configuración del servidor. Consulte el Apéndice B para obtener una lista completa de todas las propiedades de configuración del cliente y del servidor.

1. Utilice **keytool** para gestionar los almacenes de claves JCEKS. Cuando cree el almacén de claves, tome nota de la vía de acceso y del nombre de archivo, así como de los nombres dados a los certificados y las claves. Esta información se utilizará más adelante.
2. Cree un almacén de claves si no existe. Añada o importe los certificados y las claves que utilizarán las unidades de cintas a este nuevo almacén de claves. (Consulte el apartado “Generación de claves y alias para el cifrado en LTO 4 y LTO 5” en la página 3-9). Tome nota de los nombres dados a los certificados y las claves. Esta información se utilizará más adelante.
3. Cree grupos de claves y rellénelos con alias de claves. Consulte el apartado “Creación y gestión de grupos de claves” en la página 3-14.
4. Utilizando su editor de texto preferido, abra el archivo **KeyManagerConfig.properties** para especificar las siguientes propiedades. Tenga en cuenta que el diseño actual del servidor es muy estricto. No utilice Windows para editar el archivo para una máquina de Linux, debido a ^M. Si utiliza Windows, edite el archivo con gvim/vim.

   Nota para los usuarios de Windows: El SDK de Java utiliza barras inclinadas, incluso cuando se ejecuta en Windows. Cuando especifique vías de acceso en el archivo **KeyManagerConfig.properties**, asegúrese de utilizar barras inclinadas. Cuando especifique un nombre completo de vía de acceso en la ventana de mandatos, utilice barras inclinadas invertidas de la manera habitual en Windows.

   a. **Audit.Handler.File.Directory**: especifique una ubicación para almacenar los registros de auditoría.
   b. **Audit.metadata.file.name**: especifique un nombre de archivo y una vía de acceso completos para el archivo XML de metadatos.
   c. **Config.drivetable.file.url** – especifique una ubicación para la información sobre unidades conocidas para Encryption Key Manager. Este archivo no es necesario antes de iniciar el servidor o el cliente CLI. Si no existe, será creado durante el cierre del servidor Encryption Key Manager .
   d. **TransportListener.ssl.keystore.name**: especifique la vía de acceso y el nombre de archivo del almacén de claves creado en el paso 1.
   e. **TransportListener.ssl.truststore.name**: especifique la vía de acceso y el nombre de archivo del almacén de claves creados en el paso 1.

f. **Admin.ssl.keystore.name**: especifique la vía de acceso y el nombre de archivo del almacén de claves creado en el paso 1.

g. **Admin.ssl.truststore.name**: especifique la vía de acceso y el nombre de archivo del almacén de claves creado en el paso 1.

h. **config.keystore.file**: especifique la vía de acceso y el nombre de archivo del almacén de claves creado en el paso 1.

i. **drive.acceptUnknownDrives**: especifique `true` o `false`. Un valor true permite que las nuevas unidades de cintas que contacten con Encryption Key Manager sean añadidas automáticamente a la tabla de unidades. El valor predeterminado es false.

5. Las siguientes entradas opcionales de contraseña se pueden añadir u omitir. Si estas entradas no se especifican en **KeyManagerConfig.properties**, Encryption Key Manager solicitará la clave del almacén de claves durante el arranque del servidor.

   a. **Admin.ssl.keystore.password**: especifique la contraseña del almacén de claves creado en el paso 1.

   b. **config.keystore.password**: especifique la contraseña del almacén de claves creado en el paso 1.

   c. **TransportListener.ssl.keystore.password**: especifique la contraseña del almacén de claves creado en el paso 1.

   Cuando se añade al archivo **KeyManagerConfig.properties**, Encryption Key Manager oculta estas contraseñas por motivos de seguridad.

6. Si lo desea, puede establecer la propiedad **Server.authMechanism** en el valor `LocalOS` si la autenticación de cliente CLI se va a realizar contra el registro del sistema operativo local. Si no se ha especificado (o se establece en `EKM`), el valor predeterminado es que el cliente CLI inicie sesión en el servidor del gestor de claves utilizando EKMAdmin/changeME como usr/passwd. (Esta contraseña se puede cambiar con el mandato **chgpasswd**).

   Cuando la propiedad **Server.authMechanism** está establecida en LocalOS, será necesaria una configuración adicional para las plataformas Linux. Para obtener más información, consulte el archivo Readme en **http://support.dell.com** o el soporte de Dell Encryption Key Manager proporcionado con el producto. El apartado "Autenticación de usuarios del cliente CLI" en la página 5-5 contiene más información.

7. Guarde los cambios en **KeyManagerConfig.properties**.

8. Inicie el servidor Encryption Key Manager. Para iniciar el servidor sin la GUI,

   En Windows
   Vaya a cd c:\ekm\ekmserver y pulse **startServer.bat**

   En plataformas Linux
   Vaya a /var/ekm/ekmserver y especifique `. ./startServer.sh`

   Consulte el apartado "Inicio, Renovación y Detención del Servidor del gestor de claves" en la página 5-1 para obtener más información.

9. Inicie el cliente CLI:

   En Windows
   Vaya a cd c:\ekm\ekmclient y pulse **startClient.bat**

   En plataformas Linux
   Vaya a /var/ekm/ekmclient y especifique `. ./startClient.sh`

Consulte el apartado “Cliente de la interfaz de línea de mandatos” en la página 5-5 para obtener más información.

10. Si ha especificado **drive.acceptUnknownDrives = false** en el paso 4(i), configure una unidad especificando lo siguiente en la solicitud #:

    ```
    adddrive -drivename nombre_unidad -recl nombre_cert -rec2
    nombre_cert
    ```

    Por ejemplo:

    ```
    # adddrive -drivename 000001365054 -rec1 key1c1 -rec2 key1c2
    ```

    seguido de

    ```
    # listdrives -drivename 000001365054
    ```

    resulta en

    ```
    Entry Key:  SerialNumber = 000001365054

    Entry Key:  AliasTwo = key1c2

    Entry Key:  AliasOne = key1c1
     Deleted : false
     Updated : true
     TimeStamp : Sun Jul 03 17:34:44 MST 2007
    ```

11. Especifique el mandato **listdrives** en la solicitud # para asegurarse de que la unidad se ha añadido correctamente.

Capítulo 5. Administración de Encryption Key Manager

Inicio, Renovación y Detención del Servidor del gestor de claves

El servidor Encryption Key Manager es muy fácil de iniciar y detener.

Renovar el servidor hace que Encryption Key Manager vuelve el contenido actual del almacén de claves, tabla de unidades e información de configuración que tiene en su memoria a los archivos correspondientes y, a continuación, lo recarga en la memoria. Realizar una renovación es útil después de realizar cambios sobre estos componentes utilizando el cliente CLI. Aunque tales cambios se guardan automáticamente al cerrar el servidor Encryption Key Manager, renovar el servidor evita que estos cambios se pierdan en caso de que se produzca un cuelgue del sistema o un fallo en la alimentación.

Inicie el servidor Encryption Key Manager desde la GUI de Dell Encryption Key Manager:

1. Abra la GUI si todavía no la ha iniciado:

   En Windows
   Vaya a c:\ekm\gui y pulse **LaunchEKMGui.bat**

   En plataformas Linux
   Vaya a /var/ekm/gui y especifique `. ./LaunchEKMGui.sh`
2. Pulse **Server Health Monitor** > en el navegador situado a la izquierda de la GUI.
3. En la página "Server Status" (Figura 5-1), pulse **Start Server** o **Refresh Server**.

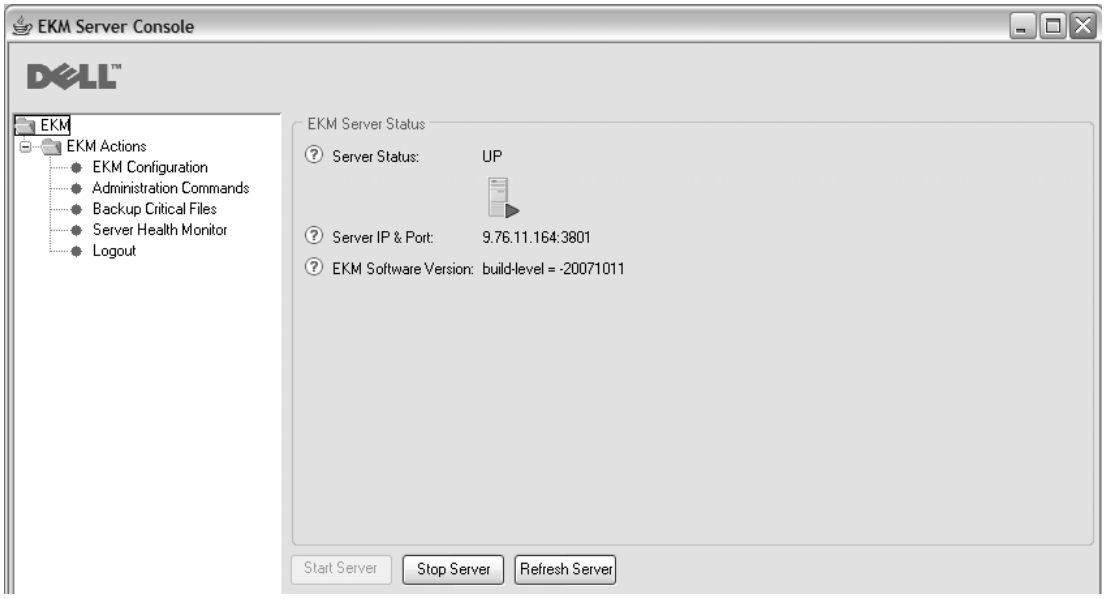

Figura 5-1. Estado del servidor

4. El cambio en el estado del servidor se refleja en la ventana Server Status. Consulte el apartado Figura 5-1.

5. Aparece la ventana Login (Figura 5-2).

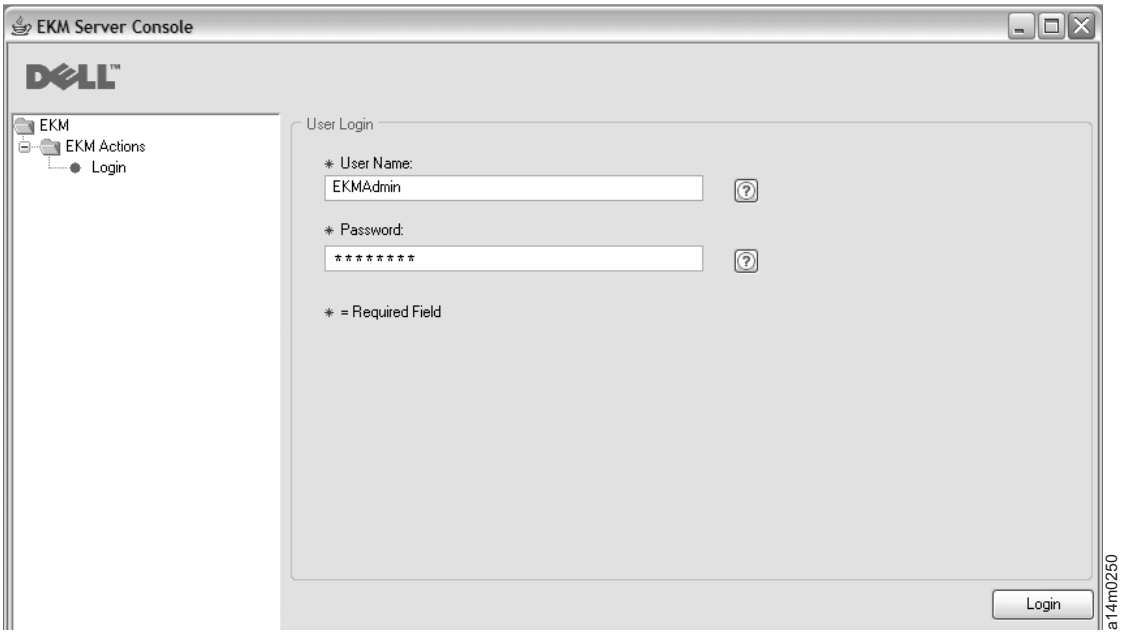

Figura 5-2. Ventana de inicio de sesión

Especifique `EKMAdmin` como User Name. La contraseña inicial es `changeME`. Después de haberse registrado, puede utilizar el mandato **chgpasswd** para cambiar la contraseña. Consulte el mandato "chgpasswd" en la página 5-9.

Nota: • Es posible que la GUI de Dell Encryption Key Manager no pueda mostrar la dirección IP del servidor

Existen dos limitaciones en la GUI actual que impiden que se muestre la dirección IP de host actual de Encryption Key Manager en **Server Health Monitor** :

- La aplicación actual no reconoce IPV6. Si el host está configurado con una dirección IPV6, la aplicación Encryption Key Manager no podrá mostrar la dirección IP.
- Si la aplicación Encryption Key Manager está instalada en un sistema Linux, la aplicación mostrará la dirección del host local y no el puerto IP activo real.

Para recuperar la dirección IP real del sistema host, encuentre la dirección del puerto IP accediendo a la configuración de red. En un sistema Windows, abra una ventana de mandatos y escriba `ipconfig`. En Linux, escriba `isconfig`.

6. Pulse **Login**.

Utilice la misma página, Server Status, para detener el servidor.

Inicio del Servidor del gestor de claves utilizando un script

En Windows

Vaya a cd c:\ekm\ekmserver y pulse **startServer.bat**

En plataformas Linux

Vaya a /var/ekm/ekmserver y especifique `. ./startServer.sh`

Para detener el servidor, emita el mandato **stopekm** utilizando cualquiera de los métodos descritos a continuación en el apartado "Cliente de la interfaz de línea de mandatos" en la página 5-5. Otro método consiste en enviar **sigterm** al proceso del gestor de claves. Esto permitirá al servidor cerrarse y terminar de una manera limpia. No envíe **sigkill** al proceso del gestor de claves. **sigkill** no cerrará el proceso de una manera limpia. Por ejemplo, en los sistemas Linux, especifique `kill -SIGTERM pid` o `kill -15 pid`.

Inicio y detención del Servidor del gestor de claves desde el indicador de mandatos

Para iniciar el servidor Encryption Key Manager desde cualquier ventana de mandatos o shell , especifique:

```
java com.ibm.keymanager.EKMLaunch KeymanagerConfig.properties
```

Esto inicia el servidor Encryption Key Manager en segundo plano. Cuando se ha iniciado correctamente, puede mostrar el proceso Java de Encryption Key Manager con el mandato `ps -ef | grep java` (plataformas Linux) o bien utilizando el Administrador de tareas de Windows. Cuando se ejecuta como servicio de Windows, aparece como LaunchEKMService.

Para detener el servidor, emita el mandato **stopekm** utilizando cualquiera de los métodos descritos a continuación en el apartado "Cliente de la interfaz de línea de mandatos" en la página 5-5. Otro método consiste en enviar **sigterm** al proceso del gestor de claves. Esto permitirá al servidor cerrarse y terminar de una manera limpia. No envíe **sigkill** al proceso del gestor de claves. **sigkill** no cerrará el proceso de una manera limpia. Por ejemplo, en los sistemas Linux, especifique `kill -SIGTERM pid` o `kill -15 pid`.

En plataformas Windows, cuando se ha iniciado Dell Encryption Key Manager como un servicio de Windows, puede detenerse desde el Panel de control.

Instalación del Servidor del gestor de claves como servicio de Windows

Instalar el servidor Encryption Key Manager como un servicio en el servidor host asegura que la aplicación del servidor Encryption Key Manager se iniciará cuando se reinicie el servidor host.

1. Extraiga el archivo ejecutable LaunchEKMService.exe del release descargado del sitio web de soporte de Dell (**http://support.dell.com**) a un directorio temporal.
2. Para que el servicio se ejecute correctamente, es necesario establecer algunas variables de entorno:
   a. En el menú Inicio, pulse en **Panel de control**.
   b. Pulse dos veces en **Sistema**.
   c. Pulse la pestaña **Avanzadas**.
   d. Pulse **Variables de entorno**.
   e. En la lista variables del sistema, pulse **Nuevo**.
   f. Especifique `JAVA_HOME` como el nombre de la variable y escriba el directorio de IBM JVM. El directorio de instalación predeterminado es `C:\PROGRA~1\IBM\Java60`
   g. Pulse **Aceptar**.
3. Edite la variable PATH del sistema utilizando este procedimiento.

Nota: El establecimiento de la variable PATH desde la línea de mandatos no funcionará.

a. En el menú Inicio, pulse en **Panel de control**.
b. Pulse dos veces en **Sistema**.
c. Pulse en la pestaña **Avanzadas**.
d. Pulse **Variables de entorno**.
e. Desplácese por la lista Variables del sistema hasta la variable **Path** y pulse **Editar**.
f. Añada la vía de acceso de `IBM JVM` al principio de la variable Path. El directorio de instalación predeterminado es `C:\PROGRA~1\IBM\Java60\jre\bin`

   Nota: Inserte un punto y coma al final de la vía de acceso para diferenciarla del resto de los directorios en la lista de vías de acceso.

g. Pulse **Aceptar**.

4. Compruebe que las vías de acceso en el archivo de configuración del servidor Encryption Key Manager son completas. Este archivo se llama KeyManagerConfig.properties y se encuentra en el directorio `C:\ekm\gui`. Deberán comprobarse y actualizarse todas las vías de acceso siguientes en el archivo para asegurarse de que tiene una vía de acceso completa (por ejemplo, utilice `c:\ekm\gui\EKMKeys.jck` y no `gui\EKMKeys.jck`). Consulte los siguientes ejemplos para ver cómo cambiar las vías de acceso al utilizar una instalación predeterminada.

   Estas son las propiedades y las vías de acceso completas a las que deben apuntar cuando se utiliza la instalación predeterminada y los nombres de almacén de claves. Encontrará cada una de estas entradas en el archivo KeyManagerConfig.properties.

   config.keygroup.xml.file
   Deberá cambiar la vía de acceso a: FILE:C:/ekm/gui/keygroups/KeyGroups.xml

   Admin.ssl.keystore.name
   Deberá cambiar la vía de acceso a: C:/ekm/gui/EKMKeys.jck

   TransportListener.ssl.truststore.name
   Deberá cambiar la vía de acceso a: C:/ekm/gui/EKMKeys.jck

   Audit.metadata.file.name
   Deberá cambiar la vía de acceso a: C:/ekm/gui/metadata/ekm_metadata.xml

   Audit.handler.file.directory
   Deberá cambiar la vía de acceso a: C:/ekm/gui/audit

   config.keystore.file
   Deberá cambiar la vía de acceso a: C:/ekm/gui/EKMKeys.jck

   TransportListener.ssl.keystore.name
   Deberá cambiar la vía de acceso a: C:/ekm/gui/EKMKeys.jck

   config.drivetable.file.url
   Deberá cambiar la vía de acceso a: FILE:C:/ekm/gui/drivetable/ekm_drivetable.dt

   Admin.ssl.truststore.name
   Deberá cambiar la vía de acceso a: C:/ekm/gui/EKMKeys.jck

5. El archivo **LaunchEKMServices.exe** debe ejecutarse desde un indicador de mandatos. En Windows, lo encontrará en **Inicio > Programas > Accesorios > Indicador de mandatos**.
6. Desde el indicador de mandatos, navegue hasta el directorio temporal donde se ha extraído **LaunchEKMService.exe**. Ejecute el archivo **LaunchEKMService.exe** utilizando las siguientes opciones como referencia.

   LaunchEKMService {**-help** | **-i** *archivo_config* | **-u**}

   -help
   Muestra la información de uso.

   -i Instala Encryption Key Manager como un servicio Windows. Esta opción necesita que se pase como un argumento el nombre de vía de acceso completo del archivo de propiedades de configuración. La vía de acceso y nombre de archivo son `C:\ekm\gui\KeyManagerConfig.properties`.

   -u Desinstala el servicio de Windows del gestor de claves si ya no necesita ejecutarlo como un servicio. Tenga en cuenta que debe detenerse el servicio EKMServer antes de desinstalarlo. Cuando se ejecuta este mandato, puede que también vea el siguiente mensaje de error: `Could not remove EKMServer. Error 0`. El servicio habrá sido, no obstante, desinstalado.

   Para instalar Encryption Key Manager como un servicio Windows, emita:

   `LaunchEKMService.exe -i archivo_config`
7. Una vez instalado el servicio con el mandato anterior, EKMServer aparecerá en el panel de control de servicios y podrá iniciar y detener Encryption Key Manager utilizando el panel de control de servicios.

   Nota: Deberá iniciar el servicio manualmente la primera vez que se utilice usando el panel de control.

Cliente de la interfaz de línea de mandatos

Una vez iniciado el servidor Encryption Key Manager, puede emitir mandatos CLI utilizando la interfaz cliente local o remotamente. Para emitir mandatos CLI, antes debe iniciar el cliente CLI.

Autenticación de usuarios del cliente CLI

La propiedad Server.authMechanism del archivo de configuración especifica el mecanismo de autenticación que se utilizará con los clientes locales/remotos. Cuando el valor se establezca en `EKM`, el usuario del cliente CLI debe iniciar la sesión en el servidor utilizando EKMAdmin/changeME como usuario/contraseña. (Esta contraseña se puede cambiar con el mandato **chgpasswd**. Consulte el apartado "chgpasswd" en la página 5-9). El valor predeterminado de la propiedad Server.authMechanism es `EKM`.

Cuando el valor de la propiedad Server.authMechanism se especifica como `LocalOS` en el archivo KeyManagerConfig.properties, la autenticación del cliente se realiza contra el registro del sistema operativo local. El usuario de cliente CLI debe iniciar la sesión en el servidor con los valores de usuario/contraseña del sistema operativo. Tenga en cuenta que el único ID de usuario/contraseña que puede iniciar la sesión y enviar mandatos al servidor es el ID de usuario con el que el servidor se está ejecutando y que tenga también autorización de superusuario/root.

Importante: el servidor Encryption Key Manager debe estar desactivado y la GUI debe estar cerrada al realizar estos cambios en el archivo de configuración de Encryption Key Manager

Para la autenticación local basada en el SO en Windows, establezca Server.authMechanism=LocalOS en el archivo KeyManagerConfig.properties como se describe a continuación:

1. Encuentre el archivo KeyManagerConfig.properties (directorio `c:\ekm\gui`).
2. Abra el archivo con su editor de texto preferido (es recomendable usar WordPad).
3. Encuentre la serie Server.authMechanism. Si esta serie no está presente, añádala al archivo exactamente de esta manera `Server.authMechanism=LocalOS`.
4. Guarde el archivo.

Ahora su ID de usuario y contraseña para el servidor Encryption Key Manager coinciden con la cuenta de usuario del sistema operativo. Tenga en cuenta que aquellos usuarios que tienen autorización para iniciar sesión y emitir mandatos en el servidor y que cuentan con privilegios de administrador pueden gestionar el servidor Encryption Key Manager

Para la autenticación local basada en sistema operativo en plataformas Linux , es necesario realizar pasos adicionales:

1. Descargue Dell Release R175158 (EKMServicesAndSamples) desde **http://support.dell.com** y extraiga los archivos en un directorio de su elección.
2. Encuentre el directorio LocalOS en la descarga.
3. Copie el archivo libjaasauth.so desde el directorio JVM-JaasSetup correspondiente a su plataforma en *java_home*/jre/bin.
   - En entornos Linux Intel de 32 bits, copie el archivo LocalOS-setup/linux_ia32/libjaasauth.so al directorio *java_home*/jre/bin/, donde *java_home* suele ser *java_install_path*/IBMJava-i386-60 para un kernel Linux Intel de 32 bits ejecutando la versión 1.6 de JVM.
   - En entornos Linux AMD64 de 64 bits, copie el archivo LocalOS-setup/linux-x86_64/libjaasauth.so al directorio *java_home*/jre/bin/, donde *java_home* suele ser *java_install_path*/IBMJava-x86_64-60 para un kernel Linux de 64 bits ejecutando la versión 1.6 de JVM.

Para plataformas Windows este archivo no es necesario.

Una vez terminada la instalación puede iniciar el servidor Encryption Key Manager. El cliente Encryption Key Manager puede ahora iniciar la sesión utilizando un nombre de usuario/contraseña del sistema operativo. Tenga en cuenta que el único ID de usuario que puede iniciar la sesión y enviar mandatos al servidor es el ID de usuario con el que el servidor se está ejecutando y que tenga también autorización de superusuario/root.

El soporte de su producto Dell incluye un archivo readme , también está disponible en **http://support.dell.com**, proporciona más detalles sobre la instalación.

Inicio del cliente de la interfaz de línea de mandatos

Nota: Las propiedades TransportListener.ssl.port en el servidor Encryption Key Manager y en los archivos de propiedades del cliente CLI de Encryption Key Manager deben estar establecidas con el mismo valor o no podrán

comunicarse. Consulte el apartado “Depuración de problemas de comunicación entre el cliente CLI y el servidor EKM” en la página 6-2 si se producen problemas.

El cliente CLI de Encryption Key Manager y el servidor Encryption Key Manager utilizan SSL para proteger sus comunicaciones. Cuando se utiliza la configuración JSSE predeterminada de no utilizar autenticación cliente, los certificados en TransportListener.ssl.keystore en el servidor Encryption Key Manager deben estar presentes en TransportListener.ssl.truststore. De esta manera, el cliente sabe que puede confiar en el servidor. Si el cliente CLI de Encryption Key Manager se está ejecutando en el mismo sistema que el servidor Encryption Key Manager, podrá utilizarse el mismo archivo de propiedades de configuración. Esto permite que el cliente CLI de Encryption Key Manager utilice la misma configuración de almacén de claves/almacén de confianza que el servidor Encryption Key Manager. Si no está en el mismo sistema o si desea que el cliente utilice distintos almacenes de claves, deberá exportar los certificados desde TransportListener.ssl.keystore especificados en el archivo de propiedades de configuración del servidor Encryption Key Manager. Estos certificados deben importarse en el almacén de confianza especificado por TransportListener.ssl.truststore en el archivo de propiedades CLI de Encryption Key Manager.

Puede iniciar el cliente CLI y emitir mandatos CLI de cuatro maneras. Independientemente de la que seleccione, debe especificar el nombre de un archivo de configuración de CLI. Consulte el Apéndice B para obtener información.

Con un script

En Windows
Vaya a cd c:\ekm\ekmclient y pulse **startClient.bat**

En plataformas Linux
Vaya a /var/ekm/ekmclient y especifique `. ./startClient.sh`

De manera interactiva
Para ejecutar los mandatos de manera interactiva desde cualquier shell o ventana de mandatos, especifique:

```
java com.ibm.keymanager.KMSAdminCmd CLIconfiglfile_name -i
```

Aparece la solicitud #. Antes de enviar mandatos, debe registrar el cliente CLI en el servidor del gestor de claves con el siguiente mandato:

```
#login –ekmuser EKMAdmin –ekmpassword changeME
```

Cuando el cliente CLI esté correctamente registrado en el servidor del gestor de claves, podrá ejecutar cualquier mandato CLI. Utilice el mandato **quit** o **logout** para cerrar el cliente CLI cuando haya acabado. De forma predeterminada, el servidor Encryption Key Manager cierra el socket de comunicación con un cliente que no se utilice transcurridos diez minutos. Cualquier intento de especificar un mandato después de diez minutos provocará la salida del cliente. Para especificar un periodo de tiempo de espera más largo para el socket cliente-servidor de Encryption Key Manager, modifique la propiedad theTransportListener.ssl.timeout en el archivo KeyManagerConfig.properties.

Con un archivo de mandato
Para enviar un lote de mandatos de un archivo al servidor del gestor de claves, cree un archivo que contenga los mandatos que desea emitir, por ejemplo, *clifile*. El primer mandato de este archivo debe ser el mandato **login**, porque es

necesario que el cliente inicie la sesión antes de ejecutar algún mandato. Por ejemplo, clifile puede contener lo siguiente:

```
login -ekmuser EKMAdmin -ekmpassword changeME
listdrives
```

Para ejecutar este archivo de mandato, inicie el cliente CLI:

```
java com.ibm.keymanager.admin.KMSAdminCmd CLIconfiglfile_name –filename clifile
```

Con un mandato cada vez

Puede ejecutar un único mandato cada vez si especifica la contraseña y el ID de usuario de CLI de cada mandato. Desde cualquier shell o ventana de mandatos, especifique:

```
java com.ibm.keymanager.KMSAdminCmd ClientConfig.properties_name -listdrives
    -ekmuser EKMAdmin -ekmpassword changeME
```

(Esta contraseña se puede cambiar con el mandato **chgpasswd**). El mandato se ejecutará y terminará la sesión de cliente.

Mandatos CLI

Encryption Key Manager proporciona un conjunto de mandatos que puede utilizarse para interactuar con el servidor Encryption Key Manager desde un cliente de interfaz de línea de mandatos, que incluye los siguientes mandatos.

addaliastogroup

Copiar un alias específico de un grupo de claves existente (origen) a un nuevo grupo de claves (destino). Esto resulta útil cuando se desea añadir un alias que ya existe en un grupo de claves a un grupo de claves distinto.

addaliastogroup -aliasID *nombre_alias* **-sourceGroupID** *nombre_grupo* **-targetGroupID** *nombre_grupo*

-aliasID
Nombre_alias de la clave que se va a añadir.

-sourceGroupID
Nombre_grupo exclusivo utilizado para identificar el grupo desde el que se va a copiar el alias.

-targetGroupID
Nombre_grupo exclusivo utilizado para identificar el grupo al que se va a añadir el alias.

Ejemplo: addaliastogroup -aliasID aliasname -sourceGroupID keygroup1 -targetGroupID keygroup2

adddrive

Añadir una nueva unidad a la tabla de unidades del gestor de claves. Consulte el apartado “Actualización automática de la tabla de unidades de cintas” en la página 4-1 para obtener información sobre cómo añadir automáticamente unidades de cintas a la tabla de unidades. Consulte los apartados “Claves de cifrado y las unidades de cintas LTO 4 y LTO 5” en la página 2-4 para obtener información sobre los requisitos de alias.

adddrive -drivename *nombre_unidad* [**-rec1** *alias*] [**-rec2** *alias*][**-symrec** *alias*]

-drivename
nombre_unidad especifica el número de serie de 12 dígitos de la unidad que se va a añadir.

Nota: Debe añadir dos ceros (0) iniciales delante del número de serie de 10 dígitos para alcanzar un total de 12 dígitos.

-rec1
Especifica el *alias* (o etiqueta de clave) del certificado de la unidad.

-rec2
Especifica un segundo *alias* (o etiqueta de clave) del certificado de la unidad.

-symrec
Especifica un *alias* (de la clave simétrica) o un nombre de grupo de claves para la unidad de cintas.

Ejemplo: adddrive -drivename 000123456789 -rec1 alias1 -rec2 alias2

addkeygroup

Crear una instancia de un grupo de claves con un ID de grupo exclusivo en el archivo XML KeyGroup.

addkeygroup -groupID *nombre_grupo*

-groupID
Nombre_grupo exclusivo utilizado para identificar el grupo en el archivo XML KeyGroup.

Ejemplo: addkeygroup -groupID keygroup1

addkeygroupalias

Crear un nuevo alias para un alias de clave existente en el almacén de claves para añadirlo a un ID de grupo de claves específico.

addkeygroupalias -alias *nombre_alias* **-groupID** *nombre_grupo*

-alias
Nuevo *nombre_alias* de la clave.

-groupID
Nombre_grupo exclusivo utilizado para identificar el grupo en el archivo XML KeyGroup.

Ejemplo: addkeygroupalias -alias aliasname -groupID keygroup1

chgpasswd

Cambiar la contraseña predeterminada del usuario del cliente CLI (EKMAdmin).

chgpasswd -new *contraseña*

-new
Nueva *contraseña* que sustituye a la contraseña anterior.

Ejemplo: chgpasswd -new ebw74jxr

createkeygroup

Crear el objeto de grupo de claves inicial en el archivo KeyGroups.xml. Ejecútelo sólo una vez.

createkeygroup -password *contraseña*

-password
Contraseña utilizada para cifrar la contraseña del almacén de claves en el archivo KeyGroups.xml, con el fin de recuperarla más adelante. El almacén de claves cifra la clave del grupo de claves, que cifra a cambio cada contraseña del alias de grupo de claves individual. Por lo tanto, ninguna de las claves del archivo KeyGroups.xml se encuentra en la copia no cifrada.

Ejemplo: createkeygroup -password password

deletedrive

Suprimir una unidad de la tabla de unidades del gestor de claves. Los mandatos equivalentes son **deldrive** y **removedrive**.

deletedrive -drivename *nombre_unidad*

-drivename
Nombre_unidad especifica el número de serie de la unidad que se va a suprimir.

Ejemplo: deletedrive -drivename 000123456789

delgroupalias

Suprimir un alias de clave del grupo de claves.

delgroupalias -groupID *nombre_grupo* **-alias** *nombre_alias*

-groupID
Nombre_grupo exclusivo utilizado para identificar el grupo en el archivo KeyGroups.xml.

-alias
Nombre_alias del alias de clave que se eliminará.

Ejemplo: delgroupalias -groupID keygroup1 -alias aliasname

delkeygroup

Suprimir un grupo de claves completo.

delkeygroup -groupID *nombre_grupo*

-groupID
Nombre_grupo exclusivo utilizado para identificar el grupo en el archivo KeyGroups.xml.

Ejemplo: delkeygroup -groupID keygroup1

exit

Salga del cliente CLI y detenga el servidor Encryption Key Manager. Un mandato equivalente es **quit**.

Ejemplo: exit

export

Exportar una tabla de unidades o un archivo de configuración de un servidor Encryption Key Manager a un URL especificado.

export {**-drivetab** | **-config**} **-url** *nombre_url*

-drivetab
Exportar la tabla de unidades.

-config
Exportar el archivo de configuración del servidor Encryption Key Manager.

-url
nombre_url especifica la ubicación en la que se va a escribir el archivo.

Ejemplo: export -drivetab -url FILE:///keymanager/data/export.table

help

Visualizar la sintaxis y los nombres de mandato de la interfaz de línea de mandatos. El mandato equivalente es **?**.

help

import

Importar una tabla de unidades o un archivo de configuración de un URL especificado.

import {**-merge** | **-rewrite**} {**-drivetab** | **-config**} **-url** *nombre_url*

-merge
Fusionar los nuevos datos con los datos actuales.

-rewrite
Sustituir los datos actuales con datos nuevos.

-drivetab
Importar la tabla de unidades.

-config
Importar el archivo de configuración.

-url
nombre_url especifica la ubicación de la que se tomarán los nuevos datos.

Ejemplo: import -merge -drivetab -url FILE:///keymanager/data/export.table

list

Listar los certificados contenidos en el almacén de claves designado por la propiedad config.keystore.file.

list [-cert |-key|-keysym][-alias *alias* **-verbose |-v]**

-cert
Listar los certificados del almacén de claves especificado.

-key
Listar todas las claves del almacén de claves especificado.

-keysym
Listar las claves simétricas del almacén de claves especificado.

-alias
alias especifica un certificado específico que listar.

-verbose|-v
Mostrar más información sobre los certificados.

Ejemplos:
`list -v` lista todo lo que aparece en el almacén de claves.

`list -alias mycert -v` lista todos los datos de variables del alias mycert si este existe en el almacén de claves config.keystore.file.

listcerts

Listar los certificados contenidos en el almacén de claves designado por la propiedad config.keystore.file.

listcerts [-alias *alias* **-verbose |-v]**

-alias
alias especifica un certificado específico que listar.

-verbose|-v
Mostrar más información sobre los certificados.

Ejemplo: listcerts -alias alias1 -v

listconfig

Lista las propiedades de configuración del servidor Encryption Key Manager en memoria, reflejando el contenido actual del archivo KeyManagerConfig.properties así como cualquier actualización realizada con el mandato **modconfig**.

listconfig

listdrives

Listar las unidades de la tabla de unidades.

listdrives [-drivename *nombre_unidad* **]**

-drivename
Nombre_unidad especifica el número de serie de la unidad de cintas que listar.

-verbose|-v
Mostrar más información sobre las unidades de cintas.

Ejemplo: listdrives -drivename 000123456789

login

Iniciar sesión en un cliente CLI en el servidor Encryption Key Manager.

login -ekmuser *ID_usuario* **-ekmpassword** *contraseña*

-ekmuser
Especifique el valor de ID de usuario `EKMadmin` o localOS para *ID_usuario*, en función del tipo de autenticación utilizado (consulte el apartado "Autenticación de usuarios del cliente CLI" en la página 5-5).

-ekmpassword
Contraseña válida para el ID de usuario.

Ejemplo: login -ekmuser EKMAdmin -ekmpassword changeME

logout

Termina la sesión del usuario actual. Un mandato equivalente es **logoff**. Estos mandatos sólo son útiles cuando la sesión de cliente está habilitada.

Ejemplo: logout

modconfig

Modificar una propiedad en el archivo de propiedades de configuración del servidor Encryption Key Manager, KeyManagerConfig.properties. Un mandato equivalente es **modifyconfig**.

modconfig {**-set** | **-unset**} **-property** *nombre* **-value** *valor*

-set
Establezca la propiedad especificada en el valor especificado.

-unset
Elimine la propiedad especificada.

-property
nombre especifica el nombre de la propiedad de destino.

-value
valor especifica el nuevo valor de la propiedad de destino cuando se especifica **-set**.

Ejemplo: modconfig -set -property sync.timeinhours -value 24

moddrive

Modifique la información de la unidad en la tabla de unidades. Un mandato equivalente es **modifydrive**.

moddrive -drivename *nombre_unidad* {**-rec1** [*alias*] | **-rec2** [*alias*] | **-symrec** [*alias*]}

-drivename
Nombre_unidad especifica el número de serie de la unidad de cintas.

-rec1
Especifica el *alias* (o etiqueta de clave) del certificado de la unidad.

-rec2
Especifica un segundo *alias* (o etiqueta de clave) del certificado de la unidad.

-symrec
Especifica un *alias* (de la clave simétrica) o un nombre de grupo de claves para la unidad de cintas.

Ejemplo: moddrive -drivename 000123456789 -rec1 newalias1

refresh

Indica a Encryption Key Manager que renueve los valores de auditoría, depuración y tabla de unidades con los parámetros de configuración más recientes.

Ejemplo: refresh

refreshks

Renueva el almacén de claves. Utilice esta opción para recargar el almacén de claves especificado en **config.keystore.file** si ha sido modificado mientras se estaba ejecutando el servidor Encryption Key Manager. Utilice este mandato sólo cuando sea necesario, ya que puede empeorar el rendimiento.

Ejemplo: refreshks

status

Muestra si el servidor del gestor de claves está iniciado o detenido.

Ejemplo: status

stopekm

Detiene el servidor Encryption Key Manager.

Ejemplo: stopekm

sync

Sincroniza las propiedades del archivo de configuración, la información de tabla de unidades o ambas en otro servidor Encryption Key Manager con las del servidor del gestor de claves que emiten el mandato.

Nota: Ningún método de sincronización actúa en el almacén de claves o en el archivo KeyGroups.xml. Se deben copiar manualmente.

sync {**-all** | **-config** | **-drivetab**} **-ipaddr** *dir_ip* :*ssl*:*puerto* [**-merge** | **-rewrite**]

-all
Envía el archivo de propiedades de configuración y la información de tabla de unidades al servidor Encryption Key Manager especificado por **-ipaddr**.

-config
Enviar sólo el archivo de propiedades de configuración al servidor Encryption Key Manager especificado por **-ipaddr**.

-drivetab
Envía sólo la información de tabla de unidades al servidor Encryption Key Manager especificado por **-ipaddr**.

-ipaddr
dir_ip:ssl:puerto especifica la dirección y puerto SSL del servidor Encryption Key Manager receptor. *ssl:puerto* debe coincidir con el valor especificado para "TransportListener.ssl.port" en el archivo KeyManagerConfig.properties del servidor receptor.

-merge
Fusionar la nueva tabla de unidades con los datos actuales. (El archivo de configuración siempre es una regrabación). Este es el valor predeterminado.

-rewrite
Sustituir los datos actuales con datos nuevos.

Ejemplo: sync -drivetab -ipaddr remoteekm.ibm.com:443 -merge

version

Muestra la versión del servidor Encryption Key Manager.

Ejemplo: version

Capítulo 6. Determinación de problemas

Puede habilitar la depuración para un componente individual, varios componentes o todos los componentes de Encryption Key Manager.

Archivos importantes que comprobar para solucionar problemas del servidor Encryption Key Manager

Cuando Encryption Key Manager no se inicia, existen tres archivos que comprobar para determinar la causa del problema.

- **native_stdout.log** y **native_stderr.log**
  - El servidor Encryption Key Manager se ejecuta en un proceso en segundo plano por lo que no tiene ninguna consola que muestra sus mensajes normales informativos y los de error. Estos mensajes se registran en estos dos archivos.
  - Si el archivo de propiedades del servidor Encryption Key Manager contiene la propiedad **debug.output.file**, estos dos archivos serán creados en el mismo directorio que el registro de depuración.
  - Si el archivo de propiedades del servidor Encryption Key Manager no contiene la propiedad **debug.output.file**, estos dos archivos serán creados en el mismo directorio que el registro de depuración.
  - Estos dos archivos se suprimen y vuelven a crear cada vez que se inicia el servidor Encryption Key Manager.
- **Registro de auditorías**
  - El registro de auditorías contiene registros creados cuando se procesaba Encryption Key Manager.
  - Dos propiedades especifican la ubicación de este archivo en **KeyManagerConfig.properties**, el archivo de propiedades de configuración del servidor Encryption Key Manager:
    - Audit.handler.file.directory – especifica en qué directorio debe encontrarse el registro de auditorías
    - Audit.handler.file.name – especifica el nombre de archivo del registro de auditoría.
  - Para obtener más información sobre la auditoría, consulte el Capítulo 7, “Registros de auditoría”, en la página 7-1.

Entradas de registro para contraseñas de almacén de claves de más de 127 caracteres

Cuando Encryption Key Manager se instale como servicio de Windows y las contraseñas de almacén de claves del archivo KeyManagerConfig.properties tengan 128 caracteres de longitud o más, Encryption Key Manager no podrá iniciarse porque no puede solicitar una contraseña de longitud aceptable. Los registros nativos de Encryption Key Manager incluirán entradas similares a las siguientes:

native_stdout.log

```
Server initialized
Default keystore failed to load
```

native_stderr.log

```
at com.ibm.keymanager.KeyManagerException: Default keystore failed to load
at com.ibm.keymanager.keygroups.KeyGroupManager.loadDefaultKeyStore(KeyGroupManager.java:145)
at com.ibm.keymanager.keygroups.KeyGroupManager.init(KeyGroupManager.java:605)
at com.ibm.keymanager.EKMServer.c(EKMServer.java:243)
at com.ibm.keymanager.EKMServer.<init>(EKMServer.java:753)
at com.ibm.keymanager.EKMServer.a(EKMServer.java:716)
at com.ibm.keymanager.EKMServer.main(EKMServer.java:129)
```

Depuración de problemas de comunicación entre el cliente CLI y el servidor EKM

La comunicación entre el cliente CLI de EKM y el servidor EKM se realiza a través de los puertos especificados en la propiedad TransportListener.ssl.port en los archivos de propiedades de configuración del cliente y el servidor, y está protegido por SSL.

A continuación, se muestra una lista de las posibles razones por las que es posible que el cliente no conecte con el servidor EKM. Incluye pasos que muestran cómo determinar el problema y cómo corregirlo.

- El servidor EKM no se está ejecutando, por lo que el cliente no tiene con quién comunicarse.
  1. Emita **netstat –an** desde una ventana de mandatos y confirme que se visualicen los puertos especificados por las propiedades TransportListener.ssl.port y TransportListener.tcp.port del archivo de propiedades del servidor EKM. Si los puertos no se visualizan, el servidor no se está ejecutando
- La propiedad TransportListener.ssl.host del archivo de propiedades del cliente CLI de EKM no señala al host correcto en el que se está ejecutando el servidor EKM.
  1. El valor de la propiedad TransportListener.ssl.host en el archivo de propiedades del cliente CLI de EKM se establece de manera predeterminada en `localhost`. Modifique el valor de esta propiedad para que señale al host correcto.
- El servidor EKM y el cliente CLI de EKM no se encuentran en el mismo puerto.
  1. Compruebe las propiedades TransportListener.ssl.port de los archivos de propiedades del servidor EKM y el cliente CLI de EKM para asegurarse de que están establecidos en el mismo valor.
- El servidor EKM y el cliente CLI de EKM no pueden encontrar un certificado común para utilizarlo y proteger la comunicación.
  1. Asegúrese de que los almacenes de claves especificados en las propiedades TransportListener.ssl.keystore y TransportListener.ssl.truststore del cliente CLI contengan los mismos certificados que los almacenes de claves Admin.ssl.keystore y Admin.ssl.truststore en las propiedades del servidor.
  2. Asegúrese de que la contraseña TransportListener.ssl.keystore.password de las propiedades de cliente sea la correcta.
  3. Asegúrese de que ninguno de los certificados de estos almacenes de claves haya caducado. JSSE no utilizará certificados caducados para proteger las comunicaciones.
- El archivo de propiedades del cliente CLI de EKM es de sólo lectura.
  1. Revise los atributos o los permisos del archivo para asegurarse de que el usuario que ejecuta el cliente CLI de EKM CLI tiene permiso para acceder al archivo y modificarlo.

- El archivo de propiedades del servidor EKM tiene Server.authMechanism = LocalOS, pero el archivo necesario del paquete EKMServicesAndSamples no ha sido instalado o se ha instalado en la ubicación equivocada.
  1. Consulte el archivo léame incluido en el paquete EKMServiceAndSamples para obtener más información sobre la autenticación.

Depuración de problemas del servidor del gestor de claves

La mayoría de los problemas relacionados con el gestor de claves implican la configuración o el arranque del servidor del gestor de claves. Consulte el Apéndice B, Archivo de configuración predeterminado, para obtener información sobre cómo especificar la propiedad de depuración.

Si Encryption Key Manager falla al arrancarse, compruebe si existe un cortafuegos.

Es posible que un cortafuegos de software o de hardware estén bloqueando el acceso al puerto de Encryption Key Manager.

El servidor EKM no se ha iniciado. El archivo de configuración EKM.properties no se ha cargado o encontrado.

1. Este error se produce al iniciar KMSAdminCmd o EKMLaunch sin especificar la vía de acceso completa de **KeyManagerConfig.properties** cuando el archivo de propiedades no se encuentra en la vía de acceso predeterminada.

   La vía de acceso predeterminada en Windows es **C:/Archivos de programa/IBM/KeyManagerServer/**

   La vía de acceso predeterminada en plataformas Linux es **/opt/ibm/KeyManagerServer/**
2. Vuelva a especificar el mandato para iniciar KMSAdminCmd e incluya la vía de acceso completa del archivo **KeyManagerConfig.properties**. Consulte el Apéndice B, "Encryption Key Manager Archivos de propiedades de configuración" para obtener más información.

El servidor EKM no se ha iniciado. El nombre de archivo del archivo de metadatos XML se tiene que especificar en el archivo de configuración.

Falta la entrada Audit.metadata.file.name del archivo de configuración.

Para corregir este problema, añada la propiedad Audit.metadata.file.name al archivo de configuración **KeyManagerConfig.properties**.

No se ha podido iniciar EKM.Mykeys. El sistema no puede encontrar el archivo especificado.

1. Este mensaje de error se produce cuando las entradas del almacén de claves en **KeyManagerConfig.properties** no señalan a un archivo existente.
2. Para corregir este problema, asegúrese de que las siguientes entradas del archivo **KeyManagerConfig.properties** señalen a archivos de almacén de claves existentes y válidos:

   Admin.ssl.keystore.name

   TransportListener.ssl.truststore.name

   TransportListener.ssl.keystore.name

   Admin.ssl.truststore.name

Consulte el Apéndice B, “Encryption Key Manager Archivos de propiedades de configuración” para obtener más información.

No se ha podido iniciar EKM. El archivo no existe = safkeyring://xxx/yyy

El error puede estar causado por haber especificado el proveedor incorrecto en la variable IJO en el script de shell del entorno de Encryption Key Manager.

Para almacenes de claves JCECCARACFKS, utilice:

```
-Djava.protocol.handler.pkgs=com.ibm.crypto.hdwrCCA.provider
```

y para los almacenes de claves JCERACFKS, utilice:

```
-Djava.protocol.handler.pkgs=com.ibm.crypto.provider
```

No se ha podido iniciar EKM. Se ha intentado forzar el almacén de claves o la contraseña no es válida.

1. Este error se produce si una o más de estas entradas en el archivo de propiedades (consulte el Apéndice B, “Encryption Key Manager Archivos de propiedades de configuración”) tiene el valor incorrecto:

   config.keystore.password (corresponde a config.keystore.file)

   admin.keystore.password (corresponde a admin.keystore.name)

   transportListener.keystore.password (corresponde a transportListener.keystore.name)
2. Este error también se puede producir si se especifica la contraseña equivocada en el indicador de solicitud de contraseña o al iniciar el servidor.
3. Si ninguna de las contraseñas está en la configuración, se le preguntará hasta tres veces si las tres entradas del almacén de claves del archivo de propiedades son exclusivas. Si todas las entradas de las propiedades son las mismas, se le preguntará una vez.

No se ha podido iniciar EKM. El formato del almacén de claves no es válido

1. Este error se puede producir cuando se especifica el tipo de almacén de claves erróneo para una de las entradas del almacén de claves en el archivo de propiedades.
2. Si todas las entradas de almacenes de claves en el archivo de propiedades apuntan al mismo archivo, Encryption Key Manager utilizará el valor config.keystore.type como el tipo de almacén de claves de todos los almacenes de claves.
3. Cuando no hay ningún tipo de entrada en el archivo de propiedades para un tipo de almacén de claves en particular, Encryption Key Manager asume que el tipo es `jceks`.

No se ha podido iniciar el servidor. La hebra del escucha no está activa ni en ejecución.

Este error se puede producir por varios motivos:

1. Las dos siguientes entradas del archivo **KeyManagerConfig.properties** señalan al mismo puerto:

   TransportListener.ssl.port

   TransportListener.tcp.port

Cada uno de los escuchas de transporte debe configurarse para escuchar en su propio puerto.

2. Una o más entradas han sido configuradas en un puerto que ya está utilizando otro servicio que se ejecuta en la misma máquina que el servidor del gestor de claves. Busque los puertos que no esté utilizando otro servicio y utilícelos para configurar el servidor del gestor de claves.
3. En sistemas que ejecuten sistemas operativos Linux de tipo , este error se puede producir si uno o varios puertos están por debajo de 1024 y el usuario que inicia el servidor del gestor de claves no es un usuario raíz. Modifique las entradas del escucha de transporte en el archivo **KeyManagerConfig.properties** para utilizar puertos por encima de 1024.

Mensaje "[Fatal Error] :-1:-1: Premature end of file." en native_stderr.log.

Este mensaje se produce cuando Encryption Key Manager carga un archivo de grupos de claves vacío. Este mensaje procede del analizador XML y no impide que Encryption Key Manager se inicie a no ser que esté configurado para utilizar grupos de claves y el archivo especificado por la propiedad config.keygroup.xml.file en **KeyManagerConfig.properties**, el archivo de propiedades del servidor Encryption Key Manager, esté dañado.

Error: No se ha podido encontrar Secretkey en el almacén de claves de configuración con alias:MyKey.

La entrada symmetricKeySet del archivo de propiedades contiene un alias de clave que no existe en la entrada config.keystore.file.

Para corregir este problema, modifique la entrada symmetricKeySet del archivo de configuración para que sólo contenga aquellos alias que existen en el archivo de almacén de claves designado por la entrada config.keystore.file en **KeyManagerConfig.properties** O añada la clave simétrica que falta al almacén de claves. Consulte el Apéndice B, "Encryption Key Manager Archivos de propiedades de configuración" para obtener más información.

No hay claves simétricas en symmetricKeySet, no se da soporte a las unidades LTO.

Se trata de un mensaje informativo. El servidor Encryption Key Manager se iniciará, pero las unidades LTO no estarán soportadas en esta instancia de Encryption Key Manager. Esto no supone un problema si no hay ninguna unidad LTO configurada para comunicarse con este Encryption Key Manager.

Errores notificados por Encryption Key Manager

Este apartado define mensajes de error notificados por Encryption Key Manager y devueltos en los datos de detección de la unidad. Por lo general, se denominan códigos de síntoma de anomalía o FSC. La tabla incluye un número de error, una breve descripción de la anomalía y acciones correctivas. Consulte el Apéndice B, Archivo de configuración predeterminado, para obtener información sobre cómo especificar la propiedad de depuración.

Tabla 6-1. Errores comunicados por el gestor de claves de cifrado

Número de error	Descripción	Acción
EE02	Anomalía en el mensaje de lectura de cifrado: DriverErrorNotifyParameterError: "Bad ASC & ASCQ received. ASC & ASCQ does not match with either of Key Creation/Key Translation/Key Aquisition operation."	La unidad de cintas ha solicitado una acción no soportada. Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado “Descarga de la última versión de la imagen ISO del gestor de claves” en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Compruebe las versiones del firmware del servidor proxy o la unidad y actualícelas al último release, si es necesario. Habilite el rastreo de depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en “Contactar con Dell” en el apartado “Lea esto en primer lugar” al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.
EE0F	Error de lógica de cifrado: Error interno: "Unexpected error. Internal programming error in EKM."	Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado “Descarga de la última versión de la imagen ISO del gestor de claves” en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Compruebe las versiones del firmware del servidor proxy o la unidad y actualícelas al último release, si es necesario. Habilite el rastreo de depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en “Contactar con Dell” en el apartado “Lea esto en primer lugar” al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.
	Error: Error de hardware de la llamada CSNDDSV returnCode 12 reasonCode 0.	Si utiliza criptografía de hardware, asegúrese de iniciar ICSF.
EE23	Anomalía en el mensaje de lectura de cifrado: Error interno: "Unexpected error....."	El mensaje recibido de la unidad o el servidor proxy no se ha podido analizar debido a un error general. Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado “Descarga de la última versión de la imagen ISO del gestor de claves” en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Habilite la depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en “Contactar con Dell” en el apartado “Lea esto en primer lugar” al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.

Tabla 6-1. Errores comunicados por el gestor de claves de cifrado (continuación)

Número de error	Descripción	Acción
EE25	Problema de configuración del cifrado: Se han producido errores relacionados con la tabla de unidades.	Asegúrese de que el parámetro config.drivetable.file.url sea correcto en el archivo KeyManagerConfig.properties, si se ha proporcionado. Ejecute el mandato `listdrives -drivename <nombre_unidad>` en el servidor Encryption Key Manager para verificar si la unidad está configurada correctamente (por ejemplo, si el número de serie de la unidad, el alias y los certificados son correctos). Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado "Descarga de la última versión de la imagen ISO del gestor de claves" en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Compruebe las versiones del firmware del servidor proxy o la unidad y actualícelas al último release, si es necesario. Habilite el rastreo de depuración y repita la operación. Si el problema persiste, en "Contactar con Dell" en el apartado "Lea esto en primer lugar" al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.
EE29	Anomalía en el mensaje de lectura de cifrado: Firma no válida	El mensaje recibido de la unidad o del servidor proxy no coincide con la firma que hay en él. Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado "Descarga de la última versión de la imagen ISO del gestor de claves" en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Habilite la depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en "Contactar con Dell" en el apartado "Lea esto en primer lugar" al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.

Tabla 6-1. Errores comunicados por el gestor de claves de cifrado (continuación)

Número de error	Descripción	Acción
EE2B	Anomalía en el mensaje de lectura de cifrado: Error interno: "Either no signature in DSK or signature in DSK can not be verified."	Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado “Descarga de la última versión de la imagen ISO del gestor de claves” en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Compruebe las versiones del firmware del servidor proxy o la unidad y actualícelas al último release, si es necesario. Habilite el rastreo de depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en “Contactar con Dell” en el apartado “Lea esto en primer lugar” al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.
EE2C	Anomalía en el mensaje de lectura de cifrado: QueryDSKParameterError: "Error parsing a QueryDSKMessage from a device. Unexpected dsk count or unexpected payload."	La unidad de cintas ha solicitado que Encryption Key Manager lleve a cabo una acción no soportada. Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado “Descarga de la última versión de la imagen ISO del gestor de claves” en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Compruebe las versiones del firmware del servidor proxy o la unidad y actualícelas al último release, si es necesario. Habilite el rastreo de depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en “Contactar con Dell” en el apartado “Lea esto en primer lugar” al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.
EE2D	Anomalía en el mensaje de lectura de cifrado: Tipo de mensaje no válido	Encryption Key Manager ha recibido un mensaje fuera de secuencia o bien ha recibido un mensaje que no sabe cómo gestionar. Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado “Descarga de la última versión de la imagen ISO del gestor de claves” en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Habilite la depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en “Contactar con Dell” en el apartado “Lea esto en primer lugar” al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.

Tabla 6-1. Errores comunicados por el gestor de claves de cifrado (continuación)

Número de error	Descripción	Acción
EE2E	Anomalía en el mensaje de lectura de cifrado: Error interno: Tipo de firma no válido	El mensaje recibido de la unidad o el servidor proxy no tiene un tipo de firma válido. Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado "Descarga de la última versión de la imagen ISO del gestor de claves" en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Habilite la depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en "Contactar con Dell" en el apartado "Lea esto en primer lugar" al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.
EE30	Solicitud prohibida.	Se ha solicitado una operación no soportada para una unidad de cintas. Especifique el mandato soportado correcto para la unidad de cintas de destino.
EE31	Problema de configuración del cifrado: se han producido errores relacionados con el almacén de claves.	Compruebe las etiquetas de clave que está intentando utilizar o configurar como predeterminadas. Puede listar los certificados disponibles para Encryption Key Manager utilizando el mandato listcerts. Si sabe que está intentando utilizar los valores predeterminados, ejecute el mandato listdrives -drivename *nombre_unidad* en el servidor Encryption Key Manager para verificar si la unidad está correctamente configurada (si, por ejemplo, el número de serie de la unidad y los alias asociados y etiquetas de clave sin correctos). Si la unidad en cuestión no tiene etiquetas de clave/alias asociadas, compruebe los valores de default.drive.alias1 y default.drive.alias2. Si esto no le ayuda o el alias/etiqueta de clave no existe, recopile los registros de depuración y póngase en en "Contactar con Dell" en el apartado "Lea esto en primer lugar" al comienzo de esta publicación en el apartado "Lea esto en primer lugar" al principio de esta publicación para obtener información sobre cómo obtener asistencia técnica.
EE32	Problema relacionado con el almacén de claves.	La causa más probable es que la cinta se cifró utilizando un Encryption Key Manager distinto con claves diferentes o que se ha cambiado el nombre de la clave utilizada para cifrar esta cinta o ha sido suprimida del almacén de claves. Emita el mandato `list -keysym` y compruebe que el alias de la solicitud está en el almacén de claves.

Tabla 6-1. Errores comunicados por el gestor de claves de cifrado (continuación)

Número de error	Descripción	Acción
EEE1	Error de lógica de cifrado: Error interno: "Unexpected error: EK/EEDK flags conflict with subpage."	Compruebe que está ejecutando la versión más reciente de Encryption Key Manager (consulte el apartado “Descarga de la última versión de la imagen ISO del gestor de claves” en la página 3-1 para saber cómo determinar cuál es la versión más reciente). Compruebe las versiones del firmware del servidor proxy o la unidad y actualícelas al último release, si es necesario. Habilite la depuración en el servidor del gestor de claves. Intente recrear el problema y recopilar los registros de depuración. Si el problema persiste, en “Contactar con Dell” en el apartado “Lea esto en primer lugar” al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.
EF01	Problema de configuración de cifrado: "Drive not configured."	La unidad que está intentando comunicarse con Encryption Key Manager no está presente en la tabla de unidades. Asegúrese de que el parámetro config.drivetable.file.url sea correcto en el archivo KeyManagerConfig.properties, si se ha proporcionado. Ejecute el mandato `listdrives` para comprobar si la unidad se encuentra en la lista. De no ser así, configure la unidad manualmente utilizando el mandato `adddrive` con la información de unidad correcta, o establezca la propiedad "drive.acceptUnknownDrives" en true con el mandato `modconfig`. Habilite el rastreo de depuración y repita la operación. Si el problema persiste, en “Contactar con Dell” en el apartado “Lea esto en primer lugar” al comienzo de esta publicación para obtener información sobre cómo obtener asistencia técnica.

Mensajes

Puede que Encryption Key Manager genere y muestre los siguientes mensajes en la consola del administrador.

No se ha especificado el archivo de configuración

Texto

```
Configuration file not specified:  KeyManager Configuration file not
specified when starting EKM.
```

Explicación

El mandato KMSAdmin necesita que el archivo de configuración se pase como un parámetro de la línea de mandatos.

Respuesta del sistema

El programa se detiene.

Respuesta del operador

Proporcione el archivo de configuración y reintente el mandato.

No se ha podido añadir la unidad

Texto

```
Failed to add drive. Drive already exists.
```

Explicación

El mandato **adddrive** ha fallado porque la unidad ya está configurada con el Encryption Key Manager y existe en la unidad de tabla.

Respuesta del operador

Ejecute el mandato **listdrives** para ver si la unidad está ya configurada con Encryption Key Manager. Si la unidad ya existe, la configuración de la unidad se puede cambiar utilizando el mandato **moddrive**. Ejecute **help** para obtener más información.

No se ha podido archivar el archivo de registro

Texto

```
Failed to archive the log file.
```

Explicación

No se ha podido cambiar el nombre del archivo de registro.

Respuesta del operador

Compruebe los permisos del archivo y el espacio en la unidad.

No se ha podido suprimir la configuración

Texto

```
"modconfig" command failed.
```

Explicación

No se ha podido suprimir la configuración de Encryption Key Manager utilizando el mandato **modconfig**.

Respuesta del operador

Verifique la sintaxis del mandato utilizando **help** y asegúrese de que los parámetros proporcionados sean los correctos. Consulte los registros de auditoría para obtener más información.

No se ha podido suprimir la entrada de la unidad

Texto

```
"deldrive" command failed.
```

Explicación

El mandato **deldrive** no ha podido suprimir la entrada de la unidad de la tabla de unidades.

Respuesta del operador

Verifique la sintaxis del mandato utilizando **help** y asegúrese de que los parámetros proporcionados sean los correctos. Compruebe que esta unidad está configurada con Encryption Key Manager utilizando el mandato **listdrives**. Consulte los registros de auditoría para obtener más información.

No se ha podido realizar la importación

Texto

```
"import" command failed.
```

Explicación

No se ha podido importar la tabla de unidades o los archivos de configuración.

Respuesta del sistema

El servidor Encryption Key Manager no se inicia start.

Respuesta del operador

Asegúrese de que el URL especificado exista y tenga permisos de lectura. Verifique la sintaxis del mandato utilizando **help**. Asegúrese de que los parámetros sean correctos e inténtelo de nuevo.

No se ha podido modificar la configuración

Texto

```
"modconfig" command failed.
```

Explicación

No se ha podido modificar la configuración de Encryption Key Manager utilizando el mandato **modconfig**.

Respuesta del operador

Verifique la sintaxis del mandato utilizando **help** y asegúrese de que los parámetros proporcionados sean los correctos. Consulte los registros de auditoría para obtener más información.

El nombre de archivo no puede ser nulo

Texto

```
File name was not supplied for audit log file.
```

Explicación

No se ha proporcionado el nombre del archivo de auditoría mediante las propiedades de configuración de Encryption Key Manager. Este parámetro es un parámetro de configuración necesario.

Respuesta del sistema

El programa se detiene.

Respuesta del operador

Compruebe que la propiedad `Audit.handler.file.name` está definida en el archivo de propiedades de configuración proporcionado a Encryption Key Manager e intente reiniciarlo.

El límite del tamaño de archivos no puede ser un número negativo

Texto

```
Maximum file size for audit log can not be a negative number.
```

Explicación

El valor de la propiedad `Audit.handler.file.size` en el archivo de configuración Encryption Key Manager debe ser un número positivo.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Especifique un número válido para `Audit.handler.file.size` e intente reiniciar Encryption Key Manager.

No hay datos que sincronizar

Texto

```
No data can be found to be synchronized with "sync".
```

Explicación

El mandato de sincronización no puede identificar datos que sincronizar.

Respuesta del operador

Asegúrese de que existe el archivo de configuración proporcionado y de que la tabla de unidades esté correctamente configurada en el archivo de configuración

utilizando *config.drivetable.file.url*. Verifique la sintaxis utilizando **help** y vuelva a intentar ejecutar el mandato **sync**.

Entrada no válida

Texto

```
Invalid input parameters for the CLI.
```

Explicación

Es posible que la sintaxis del mandato específico no sea correcta.

Respuesta del operador

Asegúrese de que el mandato especificado sea correcto. Verifique la sintaxis del mandato utilizando **help**. Asegúrese de que los parámetros proporcionados sean los correctos e inténtelo de nuevo.

Número de puerto SSL no válido en el archivo de configuración

Texto

```
Invalid SSL port number specified in the EKM configuration file.
```

Explicación

El número de puerto SSL proporcionado en el archivo de configuración no es un número válido.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Especifique un número de puerto válido para la propiedad `TransportListener.ssl.port` en el archivo de configuración al iniciar Encryption Key Manager e intente reiniciar.

Número de puerto TCP no válido en el archivo de configuración

Texto

```
Invalid TCP port number specified in the EKM configuration file.
```

Explicación

El número de puerto TCP proporcionado en el archivo de configuración no es un número válido.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Especifique un número de puerto válido para la propiedad `TransportListener.tcp.port` en el archivo de configuración al iniciar Encryption Key Manager e intente reiniciar. El número de puerto TCP predeterminado es 3801.

Se debe especificar el número de puerto SSL en el archivo de configuración

Texto

```
SSL port number is not configured in the properties file.
```

Explicación

El número de puerto SSL es una propiedad que se debe configurar en el archivo de propiedades de configuración. Se utiliza para la comunicación entre servidores Encryption Key Manager en un entorno con varios servidores.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Especifique un número de puerto válido para la propiedad `TransportListener.ssl.port` e intente reiniciar Encryption Key Manager.

Se debe especificar el número de puerto TCP en el archivo de configuración

Texto

```
TCP port number is not configured in the properties file.
```

Explicación

El número de puerto TCP es una propiedad que se debe configurar en el archivo de propiedades de configuración. Se utiliza para la comunicación entre la unidad y Encryption Key Manager.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Especifique un número de puerto válido en la propiedad `TransportListener.tcp.port` e intente reiniciar Encryption Key Manager. El número de puerto TCP predeterminado es 3801.

No se ha podido iniciar el servidor

Texto

```
EKM server failed to start.
```

Explicación

El servidor Encryption Key Manager no puede iniciarse debido a problemas de configuración.

Respuesta del operador

Verifique los parámetros en el archivo de configuración proporcionado. Revise los registros para obtener más información.

La sincronización ha fallado

Texto

```
"sync" command failed.
```

Explicación

La operación de sincronización de datos entre los dos servidores Encryption Key Manager ha fallado.

Respuesta del operador

Compruebe que la dirección IP especificada para el servidor Encryption Key Manager remoto es la correcta y que puede acceder al sistema. Asegúrese de que el archivo de configuración existe y contiene la información de tabla de unidades correcta. Verifique la sintaxis del mandato **sync** utilizando **help**. Revise los registros para obtener más información.

El archivo de registro de auditoría especificado es de sólo lectura

Texto

```
No es posible abrir el archivo de auditoría para escribir en él.
```

Explicación

No es posible abrir el archivo de registro de auditoría en la configuración de Encryption Key Manager especificado por la propiedad `Audit.handler.file.name` para escribir en él.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Compruebe los permisos del archivo y directorio de auditoría especificado y vuelva a intentar reiniciar Encryption Key Manager.

No se ha podido cargar el almacén de claves del administrador

Texto

```
Keystore for Admin cannot be loaded.
```

Explicación

No puede cargarse el almacén de claves del administrador proporcionado a Encryption Key Manager. El almacén de claves del administrador se utiliza entre servidores Encryption Key Manager para la comunicación por parte del servidor en un entorno con varios servidores.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Compruebe los valores del archivo de configuración. Compruebe que las propiedades `admin.keystore.file`, `admin.keystore.provider` y `admin.keystore.type` en el archivo de configuración de Encryption Key Manager son correctas (consulte el Apéndice B) y que el archivo de almacén de claves existe y tiene permiso de lectura. Asegúrese de que la contraseña proporcionada para el almacén de claves del administrador a través de la propiedad `admin.keystore.password` o de la línea de mandatos sea la correcta. Intente reiniciar Encryption Key Manager.

No se ha podido cargar el almacén de claves

Texto

```
Keystore for EKM can not be loaded.
```

Explicación

No puede cargarse el almacén de claves especificado a Encryption Key Manager.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Compruebe los valores del archivo de configuración. Compruebe que las propiedades `config.keystore.file`, `config.keystore.provider` y `config.keystore.type` en el archivo de configuración de Encryption Key Manager son correctas y que el archivo del almacén de claves existe y tiene permiso de lectura. Compruebe que la contraseña proporcionada para el almacén de claves de Encryption Key Manager mediante la propiedad `config.keystore.password` o especificada utilizando la línea de mandatos es correcta. Intente reiniciar.

No se ha podido cargar el almacén de claves de transporte

Texto

```
Transport keystore cannot be loaded.
```

Explicación

No es posible cargar el almacén de claves de transporte proporcionado a Encryption Key Manager. El almacén de claves de transporte se utiliza entre

servidores Encryption Key Manager para la comunicación por parte del cliente en entornos de varios servidores.

Respuesta del sistema

Encryption Key Manager no se inicia.

Respuesta del operador

Compruebe los valores del archivo de configuración. Compruebe que las propiedades `transport.keystore.file`, `transport.keystore.provider` y `transport.keystore.type` en el archivo de configuración de Encryption Key Manager son correctas y que el archivo de almacén de claves existe y tiene permiso de lectura. Asegúrese de que la contraseña proporcionada para el almacén de claves del administrador a través de la propiedad `transport.keystore.password` o de la línea de mandatos sea la correcta. Intente reiniciar Encryption Key Manager.

Acción no soportada

Texto

```
User entered action for the CLI which is not supported for EKM.
```

Explicación

Encryption Key Manager no soporta o no comprende la acción proporcionada por el mandato **sync**. Las acciones válidas son fusionar y regrabar.

Respuesta del operador

Verifique la sintaxis del mandato utilizando **help** y vuelva a intentarlo.

Capítulo 7. Registros de auditoría

Nota: Los formatos del registro de auditoría descritos en este capítulo no se consideran interfaces de programación. El formato de estos registros puede cambiar de un release a otro. El formato se documenta en este capítulo, por si se desea realizar algún análisis de los registros de auditoría.

Visión general de la auditoría

El subsistema de auditoría graba registros de auditoría textuales en un conjunto de archivos secuenciales a medida que se producen distintos sucesos auditables mientras Encryption Key Manager procesa solicitudes. El subsistema de auditoría se graba en un archivo (el nombre del directorio y el archivo se pueden configurar). El tamaño de estos archivos también se puede configurar. A medida que se escriben registros en el archivo, y que el tamaño del archivo alcanza el tamaño configurable, el archivo se cierra, su nombre se cambia en función de la indicación de fecha y hora actual y se abre otro archivo en el que se graban los nuevos registros. El registro general de registros de auditoría se separa en archivos de tamaño configurable, con nombres separados por la indicación de fecha y hora del momento en que el tamaño del archivo supera el tamaño configurable.

Para evitar que la cantidad de información del registro de auditoría general (distribuyendo todos los archivos secuenciales creados) crezca demasiado y supere el espacio disponible en el sistema de archivos, puede crear un script o un programa para supervisar el conjunto de archivos del directorio/carpeta/contenedor de auditoría configurado. Según se van cerrando y denominando los archivos en función de la indicación de fecha y hora, el contenido del archivo debera copiarse y adjuntarse en la ubicación de registros continua y duradera y, a continuación, borrarse. Tenga cuidado de no eliminar o alterar el archivo en el que Encryption Key Manager está grabando los registros durante su ejecución (este archivo no tiene una indicación de la hora en el nombre de archivo).

Parámetros de configuración de auditoría

Los siguientes parámetros se utilizan en el archivo de configuración de Encryption Key Manager para controlar qué sucesos se registran en el registro de auditoría, donde se graban los archivos de auditoría, así como el tamaño máximo de los archivos de registro de auditoría.

Audit.event.types

Sintaxis

Audit.event.types={*tipo*[*;tipo*]}

Utilización

Se utiliza para especificar los tipos de auditoría que se deben enviar al registro de auditoría. Los valores posibles para el parámetro de configuración son:

all	Todos los tipos de sucesos
authentication	Sucesos de autenticación

data_synchronization	Sucesos que se producen durante la sincronización de información entre servidores Encryption Key Manager
runtime	Sucesos que se producen como parte de las operaciones de proceso y solicitudes enviadas a Encryption Key Manager
configuration_management	Sucesos que se producen cuando se realizan cambios de configuración
resource_management	Sucesos que se producen cuando se modifican valores de recursos (unidades de cintas) en Encryption Key Manager

Ejemplos

Una especificación de ejemplo para este valor de configuración es:

```
Audit.event.types=all
```

Otro ejemplo es:

```
Audit.event.types=authentication;runtime;resource_management
```

Audit.event.outcome

Sintaxis

Audit.event.outcome={*resultado*[*;resultado*]}

Utilización

Se utiliza para indicar si los sucesos se producen como resultado de operaciones correctas, operaciones incorrectas, o si se deben auditar ambos tipos. Especifique **correcto** para registrar los sucesos que se producen como resultado de operaciones correctas. Especifique **anomalía** para registrar los sucesos que se producen como resultado de operaciones incorrectas.

Ejemplos

Una especificación de ejemplo para este valor de configuración es:

```
Audit.event.outcome=failure
```

Para habilitar los casos correctos e incorrectos:

```
Audit.event.outcome=success;failure
```

Audit.eventQueue.max

Sintaxis

Audit.eventQueue.max=*número_sucesos*

Utilización

Se utiliza para establecer el número máximo de objetos de suceso que se mantienen en la cola de memoria. Este parámetro es opcional, pero se recomienda utilizarlo. El valor predeterminado es cero.

Ejemplo

```
Audit.eventQueue.max=8
```

Audit.handler.file.directory

Sintaxis

Audit.handler.file.directory=*nombre_directorio*

Utilización

Este parámetro se utiliza para indicar en qué directorio se pueden grabar los archivos del registro de auditoría. Tenga en cuenta que si no existe el directorio, Encryption Key Manager intentará crear el directorio. Si no puede hacerlo, sin embargo, no se iniciará Encryption Key Manager. Es recomendable que el directorio exista antes de ejecutar Encryption Key Manager. Tenga en cuenta que el ID de usuario con el que se ejecuta Encryption Key Manager debe tener acceso de grabación al directorio especificado.

Ejemplos

Para establecer el directorio en **/var/ekm/ekm1/audit**:

```
Audit.handler.file.directory=/var/ekm/ekm1/audit
```

Audit.handler.file.size

Sintaxis

Audit.handler.file.size=*tamañoEnKiloBytes*

Utilización

Este parámetro se utiliza para indicar el límite de tamaño en el que se cierra un archivo de auditoría y se empieza a grabar en un nuevo archivo de auditoría. Tenga en cuenta que el tamaño actual del archivo de auditoría resultante puede superar este valor en varios bytes si el archivo se cierra después de que se supere el límite de tamaño.

Ejemplos

Para establecer el tamaño máximo del archivo en 2 megabytes, especifique:

```
Audit.handler.file.size=2000
```

Audit.handler.file.name

Sintaxis

Audit.handler.file.name=*nombre_archivo*

Utilización

Utilice este parámetro para especificar el nombre de archivo base, en el directorio de auditoría especificado para utilizarlo como nombre base al crear archivos de registro de auditoría. Tenga en cuenta que este parámetro debe contener sólo el nombre de archivo base y no el nombre de vía de acceso calificado al completo. El nombre completo del archivo de registro de auditoría tendrá el valor correspondiente a la hora en la que se grabó el archivo junto al nombre.

Para ilustrar este punto, considere un ejemplo en el que el valor Audit.handler.file.name se establezca en **ekm.log**. El nombre completo de los

archivos será similar a: `ekm.log.2315003554`. La serie adjunta se puede utilizar para determinar el orden en el que se han creado los archivos de registro de auditoría: los números más altos indican archivos de registro de auditoría más recientes.

Ejemplos

Un ejemplo del establecimiento del nombre base en **ekm.log** es:

```
Audit.handler.file.name=ekm.log
```

Audit.handler.file.multithreads

Sintaxis

Audit.handler.file.multithreads={**yes** | **true** | **no** | **false**}

Utilización

Si se especifica **true**, se utiliza una hebra diferente para grabar los datos de sucesos en el registro de auditoría, lo que permite que la hebra actual de ejecución (operación) continúe sin esperar a que se complete la grabación en el registro de auditoría. El comportamiento predeterminado es utilizar varias hebras.

Ejemplos

Un ejemplo del establecimiento del nombre base en **true** es:

```
Audit.handler.file.multithreads=true
```

Audit.handler.file.threadlifespan

Sintaxis

Audit.handler.file.threadlifespan=*tiempoEnSegundos*

Utilización

Este parámetro se utiliza para especificar el tiempo máximo que debe necesitar una hebra para grabar una entrada en el registro de auditoría. Este valor se utiliza durante el proceso de limpieza para permitir que las hebras completen su trabajo antes de interrumpirlas. Si una hebra de fondo no ha completado su trabajo en el tiempo asignado por el parámetro threadlifespan, la hebra se interrumpirá con el proceso de limpieza.

Ejemplos

Para establecer el tiempo que necesita una hebra para grabar en el registro de auditoría en 10 segundos, especifique:

```
Audit.handler.file.threadlifespan=10
```

Formato del registro de auditoría

Todos los registros de auditoría utilizan un formato de resultado parecido, que se describe a continuación. Todos los registros de auditoría contienen información común que incluye la indicación de fecha y hora y el tipo de registro, junto con la información específica del suceso de auditoría que se ha producido. El formato general para los registros de auditoría se muestra a continuación:

```
TipoRegistroAuditoría:[
  timestamp=indicación de fecha y hora
  Nombre del atributo=Valor de atributo
  ...
  ]
```

Cada registro distribuye varias líneas en el archivo. La primera línea del registro empieza con el tipo de registro de auditoría, que comienza con el primer carácter de la línea, seguido de un punto y coma (;) y un corchete izquierdo de apertura ([). Las líneas subsiguientes asociadas al mismo registro de auditoría están sangradas en dos (2) espacios para facilitar la lectura de los datos del registro. La última línea de un único registro de auditoría contiene un corchete derecho de cierre (]) sangrado en dos (2) espacios. El número de líneas de cada registro de auditoría varía en función del tipo de registro de auditoría y la información adicional del atributo que se proporciona con el registro de auditoría.

La indicación de la hora de los registros de auditoría se basa en el reloj del sistema del sistema en el que se está ejecutando Encryption Key Manager. Si estos registros se deben correlacionar en función de la indicación de fecha y hora con los sucesos que se producen en otros sistemas, se debe utilizar algún tipo de sincronización temporal para garantizar que los relojes de los diversos sistemas del entorno estén sincronizados con un nivel de precisión adecuado.

Puntos de auditoría en Encryption Key Manager

Encryption Key Manager puede grabar registros de auditoría basados en la configuración de muchos sucesos que se producen durante el proceso de solicitudes. En esta sección, se describe el conjunto de sucesos que se pueden auditar junto con la categoría de configuración del registro de auditoría, que se debe habilitar para que estos registros de auditoría se graben en los archivos de auditoría (consulte la Tabla 7-1).

Tabla 7-1. Tipos de registros de auditoría que Encryption Key Manager graba para auditar archivos

Tipo de registro de auditoría	Tipo de auditoría	Descripción
Autenticación	`authentication`	Se utiliza para registrar los sucesos de autenticación
Sincronización de datos	`data_synchronization`	Se utiliza para registrar el proceso de sincronización de datos
Tiempo de ejecución	`runtime`	Utilizado para registrar varios sucesos de proceso importantes que se producen dentro del servidor de Encryption Key Manager al manejar solicitudes
Gestión de recursos	`resource_management`	Utilizado para registrar los cambios al modo en que los recursos se configuran en Encryption Key Manager
Gestión de configuración	`configuration_management`	Utilizado para registrar los cambios en la configuración del servidor Encryption Key Manager

Atributos del registro de auditoría

En las listas siguientes se muestran los atributos disponibles para cada uno de los tipos de registro de auditoría.

Suceso de autenticación

El formato de estos registros es:

```
Authentication event:[
  timestamp=indicación de fecha y hora
  event source=origen
  outcome=resultado
  event type=SECURITY_AUTHN
  message=mensaje
  authentication type=tipo
  users=usuarios
  ]
```

Tenga en cuenta que el valor `message` sólo aparece si hay información disponible.

Suceso de sincronización de datos

El formato de estos registros es:

```
Data synchronization event:
  timestamp=indicación de fecha y hora
  event source=origen
  outcome=resultado
  event type=SECURITY_DATA_SYNC
  message=mensaje
  action=acción
  resource=recurso
  user=usuario
  ]
```

Tenga en cuenta que los valores `message` y `user` sólo aparecen si hay información disponible.

Suceso de tiempo de ejecución

El formato de estos registros es:

```
Runtime event:
  timestamp=indicación de fecha y hora
  event source=origen
  outcome=resultado
  event type=SECURITY_RUNTIME
  message=mensaje
  resource=recurso
  action=acción
  user=usuario
  ]
```

Tenga en cuenta que los valores `message` y `user` sólo aparecen si hay información disponible.

Suceso de gestión de recursos

El formato de estos registros es:

```
Resource management event:
  timestamp=indicación de fecha y hora
  event source=origen
  outcome=resultado
  event type=SECURITY_MGMT_RESOURCE
  message=mensaje
```

```
  action=acción
  user=usuario
  resource=recurso
  ]
```

Tenga en cuenta que el valor `message` sólo aparece si hay información disponible.

Suceso de gestión de configuración

El formato de estos registros es:

```
Configuration management event:
  timestamp=indicación de fecha y hora
  event source=origen
  outcome=resultado
  event type=SECURITY_MGMT_CONFIG
  message=mensaje
  action=acción
  command type=tipo
  user=usuario
  ]
```

Tenga en cuenta que el valor `message` sólo aparece si hay información disponible.

Sucesos auditados

La Tabla 7-2 describe los sucesos que hacen que se creen los registros de auditoría. En la tabla se muestra el tipo de registro de auditoría que se registra cuando se produce este suceso.

Tabla 7-2. Tipos de registro de auditoría por suceso auditado

Suceso auditado	Tipo de registro de auditoría
Usuario correctamente autenticado	authentication
Autenticación de usuario fallida	authentication
Datos enviados correctamente a otro EKM	data_synchronization
Error al enviar datos a otro EKM	data_synchronization
mandato de sincronización procesado	data_synchronization
Error al procesar mandatos de sincronización	data_synchronization
Proceso de línea de mandatos iniciado	runtime
Mandato de salida recibido	runtime
Mandato desconocido especificado	runtime
Mensaje de la unidad recibido	runtime
Error al procesar el mensaje de la unidad	runtime
Error del mensaje recibido de la unidad	runtime
Error al actualizar la tabla de la unidad con información recibida de la unidad	runtime
Error al recuperar información de la tabla de unidades	runtime
Error al recuperar información del almacén de claves	runtime
Error al procesar certificados del almacén de claves	runtime

Tabla 7-2. Tipos de registro de auditoría por suceso auditado (continuación)

Suceso auditado	Tipo de registro de auditoría
Error al buscar la clave privada en el almacén de claves	runtime
Error al sumar los valores criptográficos	runtime
El intercambio de mensajes se ha procesado correctamente	runtime
El proceso de mensajes ha empezado	runtime
Proceso de línea de mandatos iniciado	runtime
Se ha encontrado un problema al utilizar los servicios criptográficos	runtime
Se ha descubierto una unidad nueva	runtime
Error al configurar la tabla unidad a unidad	runtime
El proceso de mensajes se ha iniciado correctamente desde la unidad	runtime
Mandato stopekm recibido y procesado	runtime
Unidad eliminada de la tabla de unidades	resource_management
Error al eliminar la unidad de la tabla de unidades	resource_management
La importación de la tabla de unidades ha sido correcta	resource_management
Error al importar la tabla de unidades	resource_management
La exportación de la tabla de unidades ha sido correcta	resource_management
Error al exportar la tabla de unidades	resource_management
mandato listcerts correcto	resource_management
La adición de la unidad a la tabla de unidades ha sido correcta	resource_management
Error al añadir una unidad a la tabla de unidades	resource_management
mandato listdrives correcto	resource_management
Error al procesar el mandato listdrives	resource_management
La modificación de la tabla de unidades ha sido correcta	resource_management
Error al modificar la tabla de unidades	resource_management
El almacén de claves se ha abierto correctamente	resource_management
Error al abrir el almacén de claves	resource_management
Propiedad de configuración modificada	configuration_management
Error al cambiar la propiedad de configuración	configuration_management
Propiedad de configuración suprimida	configuration_management
Error al suprimir la propiedad de configuración	configuration_management
La importación de la configuración ha sido correcta	configuration_management

Tabla 7-2. Tipos de registro de auditoría por suceso auditado (continuación)

Suceso auditado	**Tipo de registro de auditoría**
Error al importar la configuración	configuration_management
La exportación de la configuración ha sido correcta	configuration_management
Error al exportar la configuración	configuration_management
mandato listconfig correcto	configuration_management

Capítulo 8. Utilización de metadatos

Encryption Key Manager debe configurarse para crear un archivo XML que capture información vital a medida que se cifran y graban los datos en la cinta. Este archivo puede ser consultado por el número de serie del volumen para mostrar el alias o la etiqueta de clave utilizados en el volumen. Por el contrario, el alias puede consultar el archivo para visualizar todos los volúmenes asociados a dicho alias/etiqueta de clave.

Nota: Si no configura un archivo de metadatos, Encryption Key Manager no se iniciará.

Se realiza un proceso de cifrado, Encryption Key Manager recopila los siguientes datos:

- Número de serie de la unidad
- Nombre de ámbito mundial de la unidad
- Fecha de creación
- Alias de clave 1
- Alias de clave 2
- DKi
- VolSer

Cuando los datos recopilados alcancen un determinado límite, se graban en un archivo XML. El límite predeterminado, que puede establecerse en el archivo de propiedades de Encryption Key Manager (KeyManagerConfig.properties), es de 100 registros. Una vez grabado el archivo, podrá consultarse siempre que Encryption Key Manager se esté ejecutando. Para evitar que el archivo se haga demasiado grande, se desvía automáticamente a un nuevo archivo cuando se alcanza un tamaño máximo de archivo. El tamaño de archivo máximo predeterminado para el desvío, que también puede establecerse en el archivo de propiedades de Encryption Key Manager, es de 1 MB. Sólo se guarda la versión del archivo previa y la actual. Los valores que establecer en el archivo de propiedades de configuración de Encryption Key Manager son:

Audit.metadata.file.name
Nombre del archivo XML donde se guardan los metadatos. Es obligatorio.

Audit.metadata.file.size
El tamaño de archivo máximo, especificado en kilobytes, antes de desviar el archivo de la versión actual a la anterior. Es opcional. El valor predeterminado es 1024 (1MB).

Audit.metadata.file.cachecount
Número de registros que se almacenarán en la memoria caché antes de grabar el archivo de metadatos. Es opcional. El valor predeterminado es 100.

Formato de archivo XML

El archivo contiene registros en el formato siguiente.

```
<KeyUsageEvent>
 <DriveSSN>FVTDRIVE0000</driveSSN>          -Drive Serial Number
 <VolSer>TESTER</volSer>                    -Volume Serial
```

```
 <DriveWWN>57574E414D453030</driveWWN>       -drive  WWN
 <keyAlias2>cert2</keyAlias2>               -Key  Alias1
 <keyAlias1>cert1</keyAlias1>               - keyAlias2
 <dateTime>Tue  Feb  20  09:18:07  CST  2007</dateTime>    - creation  date
</KeyUsageEvent>
```

Nota: para las unidades LTO 4 y LTO 5 sólo existirá el registro <keyAlias1></keyAlias1> y se grabará el identificador de claves de datos.

Consulta del archivo XML de metadatos

Utilice la herramienta EKMDataParser para consultar el archivo de metadatos. Esta herramienta analiza el archivo XML utilizando técnicas DOM (Modelo de objeto de documento) y no puede ejecutarse desde la interfaz de la línea de mandatos de Encryption Key Manager. Se invoca como se indica a continuación:

java com.ibm.keymanager.tools.EKMDataParser -filename
vía_acceso_completa_archivo_metadatos {**-volser** *volser* | **-keyalias** *alias*}

vía_acceso_metadatos
Se trata de la misma vía de acceso de directorio especificada para el archivo de metadatos en Audit.metadata.file.name, en el archivo **KeyManagerConfig.properties**.

-filename
nombre_archivo es obligatorio y debe ser el nombre del archivo de metadatos XML. Por lo general, coincide con el nombre especificado en la propiedad Audit.metadata.file.name del archivo **KeyManagerConfig.properties**.

-volser
El número de serie del volumen del cartucho de cinta que está buscando en el archivo XML. Deben especificarse **-volser** o **-keyalias**.

-keyalias
Alias o etiqueta de la clave que se está buscando en el archivo XML. Deben especificarse **-volser** o **-keyalias**.

Ejemplo

Asumiendo que la propiedad del nombre de archivo de metadatos (Audit.metadata.file.name) en **KeyManagerConfig.properties** está establecida en un valor de `metadata` y que el archivo que se encuentra en el directorio local donde se ejecuta Encryption Key Manager, el siguiente mandato filtraría (en pantalla) sólo los registros XML relacionados con el volser `72448`:

```
<jvm_path>/bin/java com.ibm.keymanager.tools.EKMDataParser -filename metadata -volser 72448
```

El resultado tendrá el formato siguiente:

Tabla 8-1. Formato de salida de consulta de metadatos

keyalias1	keyalias2	volSer	dateTime	driveSSN	dki
cert1	cert2	72448	Wed Mar 14 10:31:32 CDT 2007	FVTDRIVE0004	

Recuperación de un archivo de metadatos dañado

El archivo de metadatos de Encryption Key Manager puede dañarse si se cierra incorrectamente Encryption Key Manager o si el sistema en el que se está ejecutando Encryption Key Manager se cuelga. Una edición o modificación

incorrecta del archivo de metadatos puede también dañarlo. Estos daños pasarán desapercibidos hasta que EKMDataParser analiza el archivo de metadatos. EKMDataParser puede fallar con un error similar al siguiente:

```
[Fatal Error] EKMData.xml:290:16: The end-tag for element type "KeyUsageEvent" must
 end with a '>' delimiter.
org.xml.sax.SAXParseException: The end-tag for element type "KeyUsageEvent" must
 end with a '>' delimiter.
at org.apache.xerces.parsers.DOMParser.parse(Unknown Source)
at org.apache.xerces.jaxp.DocumentBuilderImpl.parse(Unknown Source)
at javax.xml.parsers.DocumentBuilder.parse(Unknown Source)
at com.ibm.keymanager.tools.EKMDataParser.a(EKMDataParser.java:136)
at com.ibm.keymanager.tools.EKMDataParser.a(EKMDataParser.java:26)
at com.ibm.keymanager.tools.EKMDataParser.main(EKMDataParser.java:93)
```

Si se produce este error, se debe a que falta un código final XML en un elemento. El archivo de metadatos de Encryption Key Manager puede recuperarse para permitir que EKMDataParser analice el archivo de nuevo.

1. Haga una copia de seguridad del archivo de metadatos de Encryption Key Manager.
2. Edite el archivo de metadatos de Encryption Key Manager.
3. En XML, debe haber un código inicial y un código final correspondiente para cada parte de datos o sucesos.
   - Algunos ejemplos de un código inicial serían:
     - <KeyUsageEvent>
     - <driveSSN>
     - <keyAlias1>
   - Algunos ejemplos de un código final serían:
     - </KeyUsageEvent>
     - </driveSSN>
     - </keyAlias1>
4. Analice el archivo y busque códigos sin correspondencia. El mensaje de error de EKMDataParser lista a qué código le falta su código final. Esto debería facilitar este proceso.
5. Cuando encuentre un código sin correspondencia, suprima temporalmente el suceso o agregue los códigos necesarios para completar el suceso.
   - Por ejemplo, el siguiente extracto de un archivo de metadatos de Encryption Key Manager muestra el primer KeyUsageEvent sin código final:

     ```
     <KeyUsageEvent>
     <driveSSN>001310000109</driveSSN>
     <volSer>      </volSer>
     <driveWWN>5005076312418B07</driveWWN>
     <keyAlias1>key000000000000000F</keyAlias1>
     <dki>6B657900000000000000000F</dki>
     <dateTime>Thu Aug 30 09:50:53 MDT 2007</dateTime>
     <KeyUsageEvent>
     <driveSSN>001310000100</driveSSN>
     <volSer>      </volSer>
     <driveWWN>5005076312418ABB</driveWWN>
     <keyAlias1>key00000000000000000</keyAlias1>
     <dki>6B6579000000000000000000</dki>
     <dateTime>Thu Sep 06 16:49:39 MDT 2007</dateTime>
     </KeyUsageEvent>
     ```

     Al añadir un </KeyUsageEvent> entre las líneas <dateTime>Thu Aug 30 09:50:53 MDT 2007</dateTime> y <KeyUsageEvent> se completa el primer <KeyUsageEvent>.

Reparar los daños en el archivo permitirá que EKMDataParser analice satisfactoriamente los datos.

Apéndice A. Archivos de ejemplo

Script del daemon de arranque de ejemplo

Atención: Es imposible exagerar la importancia de proteger los datos del almacén de claves. Sin acceso al almacén de claves, no podrá descifrar las cintas cifradas. Asegúrese de guardar la información de contraseña y almacén de claves.

Plataformas Linux

A continuación, se muestra un script de ejemplo que permite que EKM se arranque en segundo plano, en modo de prueba. Este script inicia EKM y pasa la contraseña del almacén de claves, *contraseña_almacén_claves*, a través del script. De este modo, la contraseña del almacén de claves no tiene que estar en el archivo de configuración de EKM. (vea la nota siguiente). Se debe tener en cuenta lo siguiente en el archivo de script:

```
java com.ibm.keymanager.KMSAdminCmd KeyManagerConfig.properties <<EOF
startekm
contraseña_almacén_claves
status
EOF
```

Nota: Si la contraseña del almacén de claves se especifica en EKM por medio de un script, (es decir, el archivo de configuración de EKM no contiene la contraseña del almacén de claves), cuando se realiza la copia de seguridad de EKM, los archivos (archivo de configuración, tabla de unidades y archivo de copia de seguridad del almacén de claves) no se tienen que tratar como archivos secretos, sino que el script que contiene la contraseña del almacén de claves se **debe** almacenar de una manera segura y fácil de recuperar (por ejemplo, varias copias en varias ubicaciones). La contraseña del almacén de claves es información confidencial y debe ser tratada como tal. Para realizar copias de seguridad del archivo de script existen las mismas opciones que para realizar la copia de seguridad del archivo de configuración que contiene la contraseña del almacén de claves. Puede que se realicen copias de seguridad de los scripts, o que estos se almacenen/transmitan en secreto y por separado de los archivos de copia de seguridad de EKM, lo que añadiría una cierta falta de seguridad. Por último, debemos resaltar que independientemente de dónde se almacene la contraseña del almacén de claves (en un script o en el archivo de configuración de EKM), se debe almacenar de una manera segura y fácil de recuperar, de manera que siempre se pueda recuperar la contraseña del almacén de claves. **La pérdida de todas las copias de la contraseña del almacén de claves haría que se perdiesen todas las claves del almacén de claves y no habría manera de recuperarlas**.

Archivos de configuración de ejemplo

A continuación, se muestra un archivo de propiedades EKM de ejemplo, con todas las entradas del almacén de claves señalando al mismo almacén de claves de software:

```
Admin.ssl.keystore.name = /keymanager/testkeys
Admin.ssl.keystore.type = jceks
Admin.ssl.truststore.name = /keymanager/testkeys
Admin.ssl.truststore.type = jceks
Audit.event.outcome = success,failure
Audit.event.types = all
Audit.eventQueue.max = 0
Audit.handler.file.directory = /keymanager/audit
Audit.handler.file.name = kms_audit.log
Audit.handler.file.size = 10000
Audit.metadata.file.name = /keymanager/metafile.xml
config.drivetable.file.url = FILE:///keymanager/drivetable
config.keystore.file = /keymanager/testkeys
config.keystore.provider = IBMJCE
config.keystore.type = jceks
fips = Off
TransportListener.ssl.ciphersuites = JSSE_ALL
TransportListener.ssl.clientauthentication = 0
TransportListener.ssl.keystore.name = /keymanager/testkeys
TransportListener.ssl.keystore.type = jceks
TransportListener.ssl.port = 443
TransportListener.ssl.protocols = SSL_TLS
TransportListener.ssl.truststore.name = /keymanager/testkeys
TransportListener.ssl.truststore.type = jceks
TransportListener.tcp.port = 3801
```

Éste es un archivo de propiedades EKM de ejemplo, con todas las entradas del almacén de claves señalando a un almacén de claves distinto. Las entradas en negrita difieren del primer archivo de propiedades de ejemplo anterior.

```
Admin.ssl.keystore.name = /keymanager/adminkeys.jceks
Admin.ssl.keystore.type = jceks
Admin.ssl.truststore.name = /keymanager/admintrustkeys
Admin.ssl.truststore.type = jceks
Audit.event.outcome = success,failure
Audit.event.types = all
Audit.eventQueue.max = 0
Audit.handler.file.directory = /keymanager/audit
Audit.handler.file.name = kms_audit.log
Audit.handler.file.size = 10000
Audit.metadata.file.name = /keymanager/metafile.xml
config.drivetable.file.url = FILE:///keymanager/drivetable
config.keystore.file = /keymanager/drive.keys
config.keystore.provider = IBMJCE
config.keystore.type = jceks
fips = Off
TransportListener.ssl.ciphersuites = JSSE_ALL
TransportListener.ssl.clientauthentication = 0
TransportListener.ssl.keystore.name = /keymanager/sslkeys
TransportListener.ssl.keystore.type = jceks
TransportListener.ssl.port = 443
TransportListener.ssl.protocols = SSL_TLS
TransportListener.ssl.truststore.name = /keymanager/ssltrustkeys
TransportListener.ssl.truststore.type = jceks
TransportListener.tcp.port = 3801
```

Apéndice B. Archivos de propiedades de configuración de Encryption Key Manager

Encryption Key Manager Necesita dos archivos de propiedades de configuración: uno para el servidor Encryption Key Manager y uno para el cliente CLI. Cada archivo se trata y analiza como un archivo de carga Java.util.Properties, que establece determinadas restricciones en el formato y en la especificación de propiedades:

- Se registra una propiedad de configuración por línea. El valor o los valores de una propiedad determinada se amplían hasta el final de la línea.
- No es necesario que los valores de propiedad, como las contraseñas, que contienen espacios se encuentren entre comillas.
- Las contraseñas de almacén de claves no deben tener más de 127 caracteres de longitud.
- Un espacio en blanco accidental al final de una línea puede interpretarse como parte de un valor de propiedad.

Los archivos de propiedades de configuración de ejemplo se pueden descargar en **http://support.dell.com** en el archivo EKMServicesandSamples.

Archivo de propiedades de configuración del servidor Encryption Key Manager

A continuación se muestra el conjunto completo de propiedades en el archivo de configuración del servidor Encryption Key Manager (KeyManagerConfig.properties). El orden de los valores de propiedad en el archivo no es importante. Aparecerán comentarios en el archivo. Para añadir un comentario, utilice el símbolo "#" en la primera columna de la línea.

Nota: Los cambios realizados sobre el archivo KeyManagerConfig.properties se pueden perder durante la conclusión. Por lo tanto, compruebe que el servidor Encryption Key Manager no se está ejecutando antes de editar las propiedades de configuración. Para detener el servidor Encryption Key Manager, emita el mandato **stopekm** desde el cliente CLI. Los cambios se activarán cuando se reinicie el servidor Encryption Key Manager.

Admin.ssl.ciphersuites = *valor*

Especifica las suites de cifrado que se utilizarán para comunicarse entre los servidores Encryption Key Manager. Una suite de cifrado describe los algoritmos criptográficos y los protocolos de reconocimiento que utilizan TLS (Transport Layer Security) y SSL (Secure Sockets Layer) para la transferencia de datos.

Necesario Opcional.

Valores Los valores posibles son las suites de cifrado a las que da soporte IBMJSSE2.

Predeterminado
JSSE_ALL

Admin.ssl.keystore.name = *valor*

Este es el nombre de la base de datos de pares de claves y los certificados

utilizados para operaciones de cliente SSL como, por ejemplo, mandatos **sync** entre servidores Encryption Key Manager. En una operación de sincronización, el certificado que presenta el cliente SSL al servidor SSL proviene de este almacén de claves.

Necesario Opcional. Sólo se utiliza con el mandato **sync**. El valor predeterminado es el de la propiedad **config.keystore.file**.

Admin.ssl.keystore.password = password
Contraseña para acceder a Admin.ssl.keystore.name

Necesario Opcional. Si no se proporciona, puede que se solicite al iniciar Encryption Key Manager. Si se especifica, el valor de esta propiedad se oculta por razones de seguridad y el nombre de la stanza en el archivo de propiedades se sustituirá por una nueva stanza denominada 'Admin.ssl.keystore.password.obfuscated.'

Admin.ssl.keystore.type = *valor*
Tipo de almacén de claves utilizado.

Necesario Opcional.

Predeterminado
jceks

Admin.ssl.protocols = *valor*
Protocolos de seguridad.

Necesario Opcional.

Valores SSL_TLS | SSL | TLS

Predeterminado
SSL_TLS

Admin.ssl.timeout = *valor*
Especifica cuánto tiempo espera un socket el mandato read() antes de emitir una excepción de tipo SocketTimeoutException.

Necesario Opcional.

Valores Especificado en minutos. 0 significa que no hay tiempo de espera excedido

Predeterminado
1

Admin.ssl.truststore.name = *valor*
Es el nombre del archivo de base de datos utilizado para comprobar la fiabilidad del certificado del servidor SSL que presenta el servidor al cliente SSL.

Necesario Opcional. Sólo se utiliza con el mandato **sync**. El valor predeterminado es el de la propiedad **config.keystore.file**.

Admin.ssl.truststore.type = *valor*
Tipo de almacén de claves utilizado.

Necesario Opcional.

Predeterminado
jceks

Audit.event.outcome = *valor*

Sólo se registran los sucesos de auditoría que han producido el resultado especificado

Necesario Sí.

Valores success | failure. Ambos se pueden especificar separados por una coma o un punto y coma.

Predeterminado
success

Audit.event.Queue.max = 0

Número máximo de objetos de suceso en la cola de memoria de auditoría antes de que se copien en un archivo.

Necesario Opcional. Recomendado.

Valores 0 - ? (0 indica copia inmediata).

Predeterminado
0

Audit.event.types = *valor*

Sólo se registran los sucesos de auditoría que han producido el resultado especificado

Necesario Sí.

Valores all | authentication | authorization | data synchronization | runtime | audit management | authorization terminate | configuration management | resource management | none. Se pueden especificar varios valores separados por una coma o un punto y coma.

Predeterminado
all

Audit.handler.file.directory = ../audit

Directorio en el que se ubicará Audit.handler.file.name

Necesario Opcional. Recomendado.

Audit.handler.file.multithreads = *valor*

Especifica si el manejador de auditoría debe enviar hebras separadas para procesar los registros de auditoría.

Necesario Opcional.

Valores true | false

Predeterminado
true

Audit.handler.file.name = kms_audit.log

Nombre del archivo donde se registrarán las entradas de auditoría.

Necesario Sí.

Audit.handler.file.size = 100

Tamaño que alcanzará Audit.Handler.file.name antes de empezar a sobrescribirse

Necesario Opcional. Recomendado.

Valores 0 - ? (especificado en kilobytes).

Predeterminado
100

Audit.handler.file.threadlifespan = *valor*
Limita la duración de una hebra de proceso del registro de auditoría. Sólo es útil si audit.handler.file.multithreads= true.

Necesario Opcional.

Valores Especificado en milisegundos.

Predeterminado
10000

Audit.metadata.file.cachecount = 100
Especifica el número de registros que almacenar en la memoria antes de grabar el archivo de metadatos.

Necesario No

Predeterminado
100

Audit.metadata.file.name = *valor*
Especifica el nombre del archivo XML en el que se deben guardar los registros de metadatos.

Necesario Sí.

Audit.metadata.file.size = 1024
Especifica el tamaño de archivo máximo, especificado en KB, que puede alcanzar el archivo de metadatos XML antes de que el archivo se cierre y se inicie un nuevo archivo. Sólo se guardan la versión actual y la anterior del archivo.

Necesario No

Predeterminado
1024

config.drivetable.file.url = FILE:../filedrive.table
Archivo que contiene información referente a la unidad de cintas, como el número de serie, los certificados, etc.

Necesario Sí.

config.keygroup.xml.file = *valor*
Especifica el nombre del archivo XML en el que los grupos de claves almacenan alias individuales.

Necesario Opcional.

config.keystore.file = *valor*
Especifica el almacén de claves que se utilizará.

Necesario Sí.

config.keystore.password = password
Contraseña para acceder a config.keystore.file. Si se especifica, el valor de esta propiedad se oculta por razones de seguridad y el nombre de la stanza en el archivo de propiedades se sustituirá por una nueva stanza denominada 'config.keystore.password.obfuscated'.

Necesario Opcional. Si no se proporciona, es posible que se solicite al iniciar Encryption Key Manager.

config.keystore.provider = IBMJCE

Necesario Opcional.

config.keystore.type = jceks

Necesario Opcional. Recomendado.

Predeterminado
jceks

debug = *valor*
Permite la depuración del componente de Encryption Key Manager especificado.

Necesario Opcional.

Valores all | audit | server | drivetable | config | admin | transport | logic | keystore | console | none. Puede tomar varios valores separados por comas.

Predeterminado
ninguno

debug.output = *valor*
Direcciona la salida de la depuración a la ubicación especificada.

Necesario Opcional.

Valores simple_file | console (no se recomienda).

debug.output.file = debug
Vía de acceso y nombre de archivo donde se debe grabar la salida de depuración.

Necesario Opcional. Necesario cuando debug.output = simple_file. Debe existir una vía de acceso al archivo.

drive.acceptUnknownDrives = *valor*
Añade automáticamente una nueva unidad que pone Encryption Key Manager en contacto con la tabla de unidades

Necesario Sí.

Valores true | false

Predeterminado
false

Nota de seguridad - Este valor, combinado con un valor drive.default.alias1 válido, permite añadir y poner en funcionamiento las unidades de cintas conectadas a Encryption Key Manager sin que un administrador valide dicha adición. Consulte el apartado "Actualización automática de la tabla de unidades de cintas", en el capítulo 3, para obtener más información.

fips = *valor*
Estándar federal de procesamiento de la información. Consulte el apartado "Consideraciones sobre el Estándar federal de procesamiento de la información (FIPS) 140-2" del capítulo 2 para obtener más información.

Necesario Opcional.

Valores on | off

Predeterminado
off

maximum.threads = 200

Número máximo de hebras que Encryption Key Manager puede crear.

Necesario Opcional.

Server.authMechanism = *valor*

Especifica el mecanismo de autenticación que se utilizará con clientes locales/remotos. Cuando el valor se establezca en `EKM`, el usuario del cliente CLI debe iniciar la sesión en el servidor utilizando EKMAdmin/changeME como usr/passwd. (Esta contraseña se puede cambiar con el mandato chgpasswd). Cuando el valor se especifica como `LocalOS`, la autenticación de cliente se realiza contra el registro del sistema operativo local. (Compruebe que ha cerrado el servidor Encryption Key Manager antes de modificar el archivo KeyManagerConfig.properties). El usuario de cliente CLI debe iniciar la sesión en el servidor con usr/passwd del sistema operativo. Para la autenticación basada en el sistema operativo local plataformas basadas en Linux , es necesario realizar pasos adicionales:

1. Descargue Dell Release R175158 (EKMServicesAndSamples) desde **http://support.dell.com** y extraiga los archivos en un directorio de su elección.
2. Extraiga el contenido de EKMServiceAndSamples.jar (incluido en el soporte de producto Dell y disponible en **http://support.dell.com**) en un directorio temporal
3. Copie el archivo libjaasauth.so desde el directorio LocalOS-setup correspondiente a su plataforma en *java_home*/jre/bin.
   - En entornos Linux Intel de 32 bits, copie el archivo LocalOS-setup/linux_ia32/libjaasauth.so al directorio *java_home*/jre/bin/, donde *java_home* suele ser *java_install_path*/IBMJava2-i386-142 para un kernel Linux Intel de 32 bits ejecutando la versión 1.4.2 de JVM.
   - En entornos Linux AMD64 de 64 bits, copie el archivo LocalOS-setup/linux-x86_64/libjaasauth.so al directorio *java_home*/jre/bin/, donde *java_home* suele ser *java_install_path*/IBMJava2-amd64-142 para un kernel Linux AMD de 64 bits ejecutando la versión 1.4.2 de JVM.

Para plataformas Windows este archivo no es necesario.

Una vez terminada la instalación puede iniciar el servidor Encryption Key Manager. El cliente Encryption Key Manager puede ahora iniciar la sesión utilizando un nombre de usuario/contraseña del sistema operativo. Tenga en cuenta que el único ID de usuario que puede iniciar la sesión y enviar mandatos al servidor es el ID de usuario con el que el servidor se está ejecutando y que tenga también autorización de superusuario/root.

El soporte del producto de Dell incluye un archivo readme, también está disponible en **http://support.dell.com**, que proporciona más detalles sobre la instalación.

Necesario Opcional.

Valores EKM | LocalOS

Predeterminado
EKM

Server.password = *valor*

Propiedad interna. No la edite.

symmetricKeySet = {GroupID | keyAliasList [, keyAliasList,]}

Especifica los alias de la clave simétrica y los grupos de claves que se utilizarán para las unidades de cintas LTO 4 y LTO 5.

Necesario Opcional. Se aplica sólo a los cartuchos de cinta LTO 4 y LTO 5.

Valores

Especifique un valor para *GroupID* o uno o varios valores para *keyAliasList*.

GroupID especifica un nombre de grupo de clave que alimente la lista de claves simétricas y se pueda utilizar como predeterminada cuando no se especifique ningún alias para la unidad de cintas. *GroupID* debe coincidir con un ID de grupo de claves existente en el archivo KeyGroups.xml. Si no, se devuelve la excepción KeyManageException. Si se especifica más de un *GroupID*, se devuelve una excepción KeyManagerException. Si especifica un *GroupID* válido, se rastrea la última clave utilizada en el archivo XML de grupos de claves y se utiliza una selección aleatoria para la clave siguiente cada vez que se invoque getKey desde el archivo KeyGroups.xml para la lista de claves simétricas.Cada especificación de *keyAliasList* contiene un valor para *keyAlias* o *keyAliasRange*.

keyAlias especifica la notación de Backus-Naur (BNF) para un nombre o alias de una clave simétrica del almacén de claves, de hasta 12 caracteres de longitud, o un sequentialKeyID de exactamente 21 caracteres de longitud.

keyAliasRange especifica un sequentialKeyID y dígitos hexadecimales de hasta 18 caracteres, separados por un guión (-). Si se especifican 18 caracteres, los dos primeros deben ser 00. Se debe especificar en una línea y no debe contener retorno de carro ni avance de línea.

GroupID especifica el nombre de un grupo de alias.

Ejemplo `symmetricKeySet = KMA0238ab34,KMB0000034acd2345678a,THZ001-FF` Esto indica a Encryption Key Manager que utilice los alias KMA0238ab34, KMB0000034acd2345678a y el intervalo de alias de THZ000000000000000001 hasta THZ0000000000000000FF al servir claves a unidades de cintas LTO 4 y LTO 5. Estas claves deben existir en el almacén de claves especificado por **config.keystore.file** en el archivo de propiedades.

sync.action = *valor*

Especifica qué se debe hacer con los datos durante una sincronización automática.

Necesario Opcional.

Valores rewrite | merge

Predeterminado

merge

Nota: fusionar la información de configuración es igual que volver a grabarla.

sync.ipaddress = *dir_ip:ssl*
Especifica la dirección IP y el puerto del Encryption Key Manager remoto para la sincronización automática.

Necesario Opcional. Si esta propiedad no se especifica o se especifica de manera incorrecta, se inhabilita la función de sincronización.

Valores Dirección IP del servidor remoto:número de puerto SSL

sync.timeinhours = *valor*
Especifica cuántas horas esperar antes de realizar una sincronización automática con un Encryption Key Manager remoto.

Necesario Opcional.

Valores Especificado en horas.

Predeterminado
24

sync.type = *valor*
Especifica qué datos se sincronizarán automáticamente.

Necesario Opcional.

Valores config | drivetab | all

Predeterminado
drivetab

TransportListener.ssl.ciphersuites = JSSE_ALL
Suites de cifrado que se utilizarán para la comunicación entre servidores Encryption Key Manager. Una suite de cifrado describe los algoritmos criptográficos y los protocolos de reconocimiento que utilizan TLS (Transport Layer Security) y SSL (Secure Sockets Layer) para la transferencia de datos.

Necesario Opcional.

Valores Valores: cualquier suite de cifrado a la que dé soporte IBMJSSE2.

TransportListener.ssl.clientauthentication = 0
Es necesaria la autenticación SSL para la comunicación entre servidores Encryption Key Manager.

Necesario Opcional.

Valores 0: sin autenticación de cliente (predeterminado)
1: el servidor quiere realizar la autenticación de cliente con el cliente
2: el servidor debe realizar la autenticación de cliente con el cliente

TransportListener.ssl.keystore.name = *valor*
el nombre de la base de datos utilizada por el servidor Encryption Key Manager para guardar los certificados y claves privadas del servidor SSL. Este certificado se da al cliente SSL para la autenticación y la comprobación de confianza. Este almacén de claves es también utilizado por el cliente Encryption Key Manager para comunicarse con el servidor Encryption Key Manager y funciona como un cliente SSL.

Necesario Sí.

TransportListener.ssl.keystore.password = password
Contraseña para acceder a TransportListener.ssl.keystore.name. Si se especifica, el valor de esta propiedad se oculta por razones de seguridad y el nombre de la stanza en el archivo de propiedades se sustituirá por una nueva stanza denominada 'TransportListener.ssl.keystore.password.obfuscated'.

Necesario Opcional.

TransportListener.ssl.keystore.type = jceks

Necesario Opcional. Recomendado.

Valores JCEKS

TransportListener.ssl.port = *valor*
Puerto en el que el servidor Encryption Key Manager escuchará solicitudes de otros servidores Encryption Key Manager o del cliente CLI de Encryption Key Manager.

Necesario Sí.

Valores Número de puerto, por ejemplo 443. Debe coincidir con la propiedad TransportListener.ssl.port del archivo de propiedades de configuración del cliente CLI.

TransportListener.ssl.protocols = SSL_TLS
Protocolos de seguridad

Necesario Opcional.

Valores SSL_TLS (default) | SSL | TLS

TransportListener.ssl.timeout = 10
Especifica cuánto tiempo espera un socket el mandato read() antes de emitir una excepción de tipo SocketTimeoutException.

Necesario Opcional.

Valor Especificado en minutos.

Predeterminado
1

TransportListener.ssl.truststore.name = *valor*
Nombre de la base de datos de claves públicas y certificados firmados utilizados para verificar las identidades de otros clientes y servidores. Si la propiedad TransportListener.ssl.clientauthentication **no** está establecida en el valor predeterminado de 0 (sin autenticación de cliente), el servidor Encryption Key Manager, actuando como el servidor SSL, deberá autenticar el cliente utilizando este archivo. Este almacén de confianza también es utilizado por el cliente Encryption Key Manager para comunicarse con el servidor Encryption Key Manager y actuar como cliente SSL.

Necesario Sí.

TransportListener.ssl.truststore.type = jceks

Necesario Opcional. Recomendado.

Valores JCEKS

TransportListener.tcp.port = *valor*

El puerto en el que el servidor Encryption Key Manager escuchará solicitudes de las unidades de cintas. El número de puerto TCP predeterminado es 3801.

Necesario Sí.

Valores Número de puerto, 10 por ejemplo.

TransportListener.tcp.timeout = *valor*

Especifica cuánto tiempo espera un socket el mandato read() antes de emitir una excepción de tipo SocketTimeoutException.

Necesario Opcional.

Valores Especificado en minutos. 0 significa que no hay tiempo de espera excedido.

Predeterminado
10

Archivo de propiedades de configuración del cliente CLI

Este archivo, ClientKeyManagerConfig.properties, contiene un subconjunto de las propiedades contenidas en el archivo KeyManagerConfig.properties. Este subconjunto incluye las siguientes propiedades.

TransportListener.ssl.ciphersuites = JSSE_ALL

Las suites de cifrado que se utilizarán entre servidores Encryption Key Manager y clientes CLI. Una suite de cifrado describe los algoritmos criptográficos y los protocolos de reconocimiento que utilizan TLS (Transport Layer Security) y SSL (Secure Sockets Layer) para la transferencia de datos.

Necesario Opcional.

Valores Este valor debe coincidir con el valor especificado por TransportListener.ssl.ciphersuites en el archivo de propiedades de Encryption Key Manager, KeyManagerConfig.properties.

TransportListener.ssl.host = *valor*

Identifica el servidor Encryption Key Manager con el cliente CLI de Encryption Key Manager.

Necesario Opcional.

Valores Dirección IP o nombre de host

Predeterminado
localhost

Ejemplos
```
TransportListener.ssl.host = 9.24.136.444
TransportListener.ssl.host = ekmsvr02
```

Nota: No se utiliza en el archivo KeyManagerConfig.properties.

TransportListener.ssl.keystore.name = *valor*

Este almacén de claves es también utilizado por el cliente Encryption Key Manager para comunicarse con el servidor Encryption Key Manager y actuar como un cliente SSL.

Necesario Sí.

TransportListener.ssl.keystore.type = jceks
Tipo de almacén de claves.

Necesario Opcional. Recomendado.

Predeterminado
jceks

TransportListener.ssl.port = *valor*
Este es el puerto que utilizará el cliente CLI para comunicarse con los servidores Encryption Key Manager.

Necesario Sí.

Valores Este valor debe coincidir con el valor especificado por TransportListener.ssl.port en el archivo de propiedades del servidor Encryption Key Manager, KeyManagerConfig.properties.

TransportListener.ssl.protocols = SSL_TLS
Protocolos de seguridad

Necesario Opcional.

Valores Este valor debe coincidir con el valor especificado en TransportListener.ssl.protocols en el archivo de propiedades del servidor Encryption Key Manager, KeyManagerConfig.properties.

TransportListener.ssl.truststore.name = *valor*
Nombre de la base de datos de claves públicas y certificados firmados utilizados para verificar las identidades de otros clientes y servidores.

Necesario Sí.

TransportListener.ssl.truststore.type = jceks
Tipo de almacén de confianza.

Necesario Opcional. Recomendado.

Predeterminado
jceks

Existen archivos de configuración de ejemplo disponibles para su descarga en el archivo EKMServicesAndSamples desde el sitio web de **http://support.dell.com**.

Apéndice C. Preguntas frecuentes

¿Se puede utilizar alguna combinación de gestión de claves basada en aplicaciones y cifrado gestionado por bibliotecas?

No. Cuando se utiliza el cifrado gestionado por aplicaciones, el cifrado es transparente en las capas del la biblioteca. Del mismo modo, cuando se utiliza el cifrado gestionado por bibliotecas, el proceso es transparente en las otras capas. Cada método de gestión de cifrado es independiente de los otros. Para el cifrado gestionado por bibliotecas, la aplicación no necesita ser modificada de ninguna manera.

¿Es necesario tener Encryption Key Manager instalado y ejecutándose en todos los sistemas que puedan generar una solicitud para cifrar o descifrar una cinta?

Tanto con el cifrado gestionado por biblioteca-, el sistema desde el cual se origina la solicitud de grabación de unidad de cintas NO tiene que ser necesariamente el sistema en el que se está ejecutando Encryption Key Manager. Es más, no es necesario que se esté ejecutando una instancia de Encryption Key Manager en todos los sistemas desde los que se accede a una unidad de cintas cifrada.

Si incluyo el parámetro "drive.acceptUnknownDrives = True", ¿debo incluir de todos modos el parámetro "config.drivetable.file.url = FILE:/nombre_archivo" en el archivo de configuración?

config.drivetable.file.url debe especificarse siempre. Ahí es donde se encontrará la información de la unidad. Si establece `drive.acceptUnknownDrives = True`, deberá especificar también las variables `drive.default.alias1` y `drive.default.alias2` en la etiqueta correcta de clave/alias de certificado.

¿Es FILE:/nombre_archivo la sintaxis correcta para la propiedad `config.drivetable.file.url`? FILE:///filename aparece en el archivo de ejemplo y FILE:../ en la descripción.

Los ejemplos son correctos. Se trata de una especificación de URL y no es lo que se suele esperar de una especificación de estructura de directorios

¿Debo utilizar barras inclinadas o barras inclinadas invertidas al especificar vías de acceso completas en el archivo KeyManagerConfig.properties para una instancia de Encryption Key Manager que se ejecute en Windows?

Dado que KeyManagerConfig.properties es un archivo de propiedades de Java, sólo se reconocen las barras inclinadas en los nombres de vías de acceso, incluso en Windows. Si utiliza barras inclinadas invertidas en el archivo KeyManagerConfig.properties, se producirán errores.

¿Realiza Encryption Key Manager alguna comprobación del tipo Lista de revocación de certificados (CRL)?

No, Encryption Key Manager no realiza ninguna comprobación tipo CRL

¿Qué sucede cuándo el certificado que se está utilizando para cifrar las cintas caduca? ¿Leerá Encryption Key Manager cintas que hayan sido cifradas anteriormente?

Para Encryption Key Manager no es relevante si el certificado ha caducado. Continuará respetando estos certificados y leyendo las cintas cifradas anteriormente. Sin embargo, el certificado caducado debe permanecer en el almacén de claves para que se puedan leer o adjuntar las cintas cifradas anteriormente.

¿Necesitará Encryption Key Manager renombrar un certificado al renovarlo?

Encryption Key Manager está configurado para respetar las nuevas solicitudes de claves con certificados caducados. Cuando Encryption Key Manager está configurado de esta manera la renovación de certificados no es necesaria. Si esta función está inhabilitada y necesita utilizar el par de clave privada y certificado para las nuevas solicitudes, el usuario deberá renovar el certificado. Sólo se renovaría el certificado (fechas de validación) pero no las claves asociadas.

¿Seguirán leyendo las cintas creadas con versiones anteriores las versiones futuras de Encryption Key Manager?

Sí. Encryption Key Manager respetará los certificados independientemente del release.

Avisos

Marcas registradas

Marcas registradas utilizadas en este documento: Dell, el logotipo de Dell y PowerVault son marcas registradas de Dell Inc. Microsoft y Windows son marcas registradas de Microsoft Corporation. Es posible que en este documento se utilicen otras marcas registradas y nombres comerciales para hacer referencia a las entidades poseedoras de la marca y del nombre o de sus productos. Dell Inc. declina cualquier interés de propiedad en las marcas registradas y nombre comerciales que no sean propios de Dell Inc.

Glosario

Este glosario define los términos, abreviaturas y acrónimos especiales utilizados en esta publicación y en otras publicaciones relacionadas.

AES. Estándar de cifrado avanzado. Cifrado de bloque adoptado como estándar de cifrado por el gobierno de EE.UU.

alias. Véase etiqueta de clave.

almacén de certificados. Véase almacén de claves.

almacén de claves. Base de datos de claves privadas y sus cadenas de certificados digitales X.509 asociados, utilizada para autenticar las claves públicas correspondientes. También se denomina almacén de certificados o conjunto de claves en algunos entornos.

certificado. Documento digital que enlaza una clave pública con la identidad del propietario del certificado, lo que permite que el propietario del certificado se autentique.

cifrado. Conversión de datos en una cifra. En necesaria una clave para cifrar y descifrar los datos. El cifrado proporciona protección frente a personas o software que tratan de acceder a los datos sin tener la clave.

clave privada. Una clave de un par de claves asimétrico, por lo general utilizada para el descifrado. Encryption Key Manager utiliza claves privadas para desempaquetar las claves de datos AES antes del descifrado.

clave pública. Una clave de un par de claves asimétrico, por lo general utilizada para el cifrado. Encryption Key Manager utiliza claves públicas para empaquetar (proteger)

conjunto de claves. Véase almacén de claves.

DK. Clave de datos. Serie alfanumérica utilizada para cifrar datos.

EEDK. Clave de datos cifrados externamente. Clave de datos que ha sido cifrada (empaquetada) por una clave de cifrado de claves antes de ser almacenada en el cartucho de datos. Véase KEK.

etiqueta de certificado. Véase etiqueta de clave.

etiqueta de clave. Identificador exclusivo utilizado para correlacionar la EEDK con la clave privada (KEK) necesaria para desempaquetar la clave de datos simétrica protegida. Se denomina también alias o etiqueta de certificados, en función del almacén de claves utilizado.

KEK. Clave de cifrado de claves. Clave alfanumérica asimétrica utilizada para cifrar la Clave de datos. Véase EEDK.

PKDS. Conjunto de datos de claves públicas. También el conjunto de datos de claves criptográficas PKA.

RSA. Algoritmo Rivest-Shamir-Adleman. Sistema para la criptografía asimétrica de clave pública utilizado para el cifrado y la autenticación. Lo inventaron en 1977 Ron Rivest, Adi Shamir y Leonard Adleman. La seguridad del sistema depende de la dificultad de factorizar el producto de dos grandes números primos.

volver a poner en clave. Proceso de cambiar la clave de cifrado de claves (KEK) asimétrica que protege la clave de datos (DK) almacenada en una cinta ya cifrada, lo que permite que varias entidades accedan a los datos.

Índice

P

R

S

T

U

V

W