Comience con la gestión de claves externas

Element Software

NetApp
April 17, 2024

Tabla de contenidos

Comience con la gestión de claves externas

La gestión de claves externas (EKM) ofrece gestión de claves de autenticación seguras (AK) en combinación con un servidor de claves externo (EKS) fuera de clúster. El AKS se utiliza para bloquear y desbloquear unidades de cifrado automático (SED) cuando "cifrado en reposo" está habilitado en el clúster. El EKS proporciona una generación y almacenamiento seguros del AKS. El clúster utiliza el protocolo de interoperabilidad de gestión de claves (KMIP, en inglés "Key Management Interoperability Protocol"), un protocolo estándar definido de OASIS para comunicarse con el EKS.

- "Configurar la administración externa"
- "Vuelva a obtener el cifrado de software en la clave maestra de REST"
- "Recuperación de claves de autenticación no válidas o inaccesibles"
- "Comandos de API de gestión de claves externas"

Obtenga más información

- "CreateCluster API que se puede usar para habilitar el cifrado de software en reposo"
- "Documentación de SolidFire y el software Element"
- "Documentación para versiones anteriores de SolidFire de NetApp y los productos Element"

Configure la gestión de claves externas

Puede seguir estos pasos y usar los métodos API de Element que aparecen para configurar la función de gestión de claves externa.

Lo que necesitará

- Si va a configurar la gestión de claves externas en combinación con el cifrado de software en reposo, debe habilitar el cifrado de software en reposo con el "CreateCluster" método en un nuevo clúster que no contiene volúmenes.

Pasos

1. Establecer una relación de confianza con el servidor de claves externo (EKS).
   a. Cree un par de claves público/privado para el clúster de Element que se utilice para establecer una relación de confianza con el servidor de claves llamando al siguiente método de API: "CreatePublicPrivateKeyPair"
   b. Obtenga la solicitud de firma de certificado (CSR) que la entidad de certificación debe firmar. La CSR permite que el servidor de claves verifique que el clúster de Element que tendrá acceso a las claves se autentique como clúster de Element. Llame al siguiente método API: "GetClientCertificateSignRequest"
   c. Utilice la autoridad EKS/Certificate para firmar la CSR recuperada. Consulte la documentación de terceros para obtener más información.
2. Cree un servidor y un proveedor en el clúster para comunicarse con el EKS. Un proveedor de claves define dónde se debe obtener una clave y un servidor define los atributos específicos del EKS con los que se comunicará.

a. Cree un proveedor de claves en el que residirán los detalles del servidor de claves llamando al siguiente método de API: "CreateKeyProviderKmip"

b. Cree un servidor de claves que proporcione el certificado firmado y el certificado de clave pública de la entidad emisora de certificados llamando a los siguientes métodos API: "CreateKeyServerKmip" "TestKeyServerKmip"

   Si la prueba falla, verifique la configuración y la conectividad del servidor. A continuación, repita la prueba.

c. Para agregar el servidor de claves al contenedor de proveedor de claves, llame a los siguientes métodos API:"AddKeyServerToProviderKmip" "TestKeyProviderKmip"

   Si la prueba falla, verifique la configuración y la conectividad del servidor. A continuación, repita la prueba.

3. Realice una de las siguientes acciones como siguiente paso para el cifrado en reposo:

   a. (Para el cifrado de hardware en reposo) Habilitar "cifrado de hardware en reposo" Mediante la identificación del proveedor de claves que contiene el servidor de claves utilizado para almacenar las claves, llame al "EnableEncryptionAtest" Método API.

   > Debe habilitar el cifrado en reposo a través del "API". Si se habilita el cifrado en reposo con el botón existente de interfaz de usuario de Element, la función volverá al uso de claves generadas internamente.

   b. (Para el cifrado de software en reposo) en orden de "cifrado de software en reposo" Para utilizar el proveedor de claves recién creado, pase el ID de proveedor de claves al "RekeySoftwareEncryptionAtRestMasterKey" Método API.

Obtenga más información

- "Habilite y deshabilite el cifrado de un clúster"
- "Documentación de SolidFire y el software Element"
- "Documentación para versiones anteriores de SolidFire de NetApp y los productos Element"

Vuelva a obtener el cifrado de software en la clave maestra de REST

Es posible usar la API de Element para volver a introducir una clave existente. Este proceso crea una nueva clave maestra de reemplazo para el servidor de gestión de claves externo. Las claves maestras siempre se sustituyen por claves maestras nuevas y nunca se duplican ni se sobrescriben.

Es posible que deba volver a introducir la clave como parte de uno de los siguientes procedimientos:

- Cree una nueva clave como parte de un cambio de la gestión de claves interna a la gestión de claves externas.
- Cree una nueva clave como reacción o como protección ante un evento relacionado con la seguridad.

Este proceso es asíncrono y devuelve una respuesta antes de que se complete la operación de reclave. Puede utilizar el "GetAsyncResult" método para sondear el sistema para ver cuándo se ha completado el proceso.

Lo que necesitará

- Habilitó el cifrado de software en reposo mediante el "CreateCluster" Método en un nuevo clúster que no contiene volúmenes y no tiene I/O. Utilice el enlace:../api/reference_element_api_getsoftwareencryptionatrestinfo.html[`GetSoftwareEncryptionatRestInfo`] para confirmar que el estado es `enabled` antes de continuar.
- Ya tienes "estableció una relación de confianza" Entre el clúster de SolidFire y un servidor de claves externo (EKS). Ejecute el "TestKeyProviderKmip" método para verificar que se ha establecido una conexión con el proveedor de claves.

Pasos

1. Ejecute el "ListKeyProvidersKmip" Y copie el ID del proveedor de claves (`keyProviderID`).
2. Ejecute el "RekeySoftwareEncryptionAtRestMasterKey" con la `keyManagementType` parámetro como `external` y.. `keyProviderID` Como el número de ID del proveedor de claves del paso anterior:

```
{
 "method": "rekeysoftwareencryptionatrestmasterkey",
 "params": {
   "keyManagementType": "external",
   "keyProviderID": "<ID number>"
 }
}
```

3. Copie el `asyncHandle` valor de `RekeySoftwareEncryptionAtRestMasterKey` respuesta del comando.
4. Ejecute el "GetAsyncResult" con el `asyncHandle` valor del paso anterior para confirmar el cambio en la configuración. Desde la respuesta del comando, debe ver que la configuración de la clave maestra anterior se ha actualizado con información de clave nueva. Copie el nuevo ID del proveedor de claves para usarlo en un paso posterior.

```
{
   "id": null,
   "result": {
     "createTime": "2021-01-01T22:29:18Z",
     "lastUpdateTime": "2021-01-01T22:45:51Z",
     "result": {
       "keyToDecommission": {
         "keyID": "<value>",
         "keyManagementType": "internal"
     },
     "newKey": {
       "keyID": "<value>",
       "keyManagementType": "external",
       "keyProviderID": <value>
     },
     "operation": "Rekeying Master Key. Master Key management being
transferred from Internal Key Management to External Key Management with
keyProviderID=<value>",
     "state": "Ready"
   },
   "resultType": "RekeySoftwareEncryptionAtRestMasterKey",
   "status": "complete"
}
```

5. Ejecute el `GetSoftwareEncryptionatRestInfo` comando para confirmar la información de la nueva clave, incluida la `keyProviderID`, se han actualizado.

```
{
   "id": null,
   "result": {
     "masterKeyInfo": {
       "keyCreatedTime": "2021-01-01T22:29:18Z",
       "keyID": "<updated value>",
       "keyManagementType": "external",
       "keyProviderID": <value>
     },
     "rekeyMasterKeyAsyncResultID": <value>
     "status": "enabled",
     "version": 1
   },
}
```

Obtenga más información

- "Gestione el almacenamiento con la API de Element"
- "Documentación de SolidFire y el software Element"
- "Documentación para versiones anteriores de SolidFire de NetApp y los productos Element"

Recuperación de claves de autenticación no válidas o inaccesibles

Ocasionalmente, puede producirse un error que requiere la intervención del usuario. En caso de error, se generará un error del clúster (denominado código de avería del clúster). Los dos casos más probables se describen aquí.

El clúster no puede desbloquear las unidades debido a un fallo en el clúster KmipServerFault.

Esto puede suceder cuando el clúster se inicia por primera vez y no se puede acceder al servidor de claves o la clave requerida no está disponible.

1. Siga los pasos de recuperación indicados en los códigos de fallo del clúster (si los hubiera).

Se puede configurar un error slicServiceUnhealthy porque las unidades de metadatos se han marcado como un error y se han colocado en el estado "Available".

Pasos para borrar:

1. Vuelva a añadir las unidades.
2. Después de 3 a 4 minutos, verificar que el `sliceServiceUnhealthy` se borró el error.

Consulte "códigos de error de clúster" si quiere más información.

Comandos de API de gestión de claves externas

Lista de todas las API disponibles para administrar y configurar EKM.

Se utiliza para establecer una relación de confianza entre el clúster y los servidores externos propiedad del cliente:

- CreatePublicPrivateKeyPair
- GetClientCertificateSignRequest

Se utiliza para definir los detalles específicos de los servidores externos propiedad del cliente:

- CreateKeyServerKmip
- ModifyKeyServerKmip
- DeleteKeyServerKmip
- GetKeyServerKmip
- ListKeyServersKmip

- TestKeyServerKmip

Se utiliza para crear y mantener proveedores de claves que gestionan servidores de claves externos:

- CreateKeyProviderKmip
- DeleteKeyProviderKmip
- AddKeyServerToProviderKmip
- RemoveKeyServerFromProviderKmip
- GetKeyProviderKmip
- ListKeyProvidersKmip
- RekeySoftwareEncryptionAtRestMasterKey
- TestKeyProviderKmip

Para obtener información sobre los métodos de API, consulte "Información de referencia de API".