Métodos de API de segurança

Element Software

NetApp
November 21, 2024

Índice

Métodos de API de segurança

Você pode integrar o software Element com serviços externos relacionados à segurança, como um servidor de gerenciamento de chaves externo. Esses métodos relacionados à segurança permitem configurar recursos de segurança do elemento, como gerenciamento de chaves externas para criptografia em repouso.

- AddKeyServerToProviderKmip
- CreateKeyProviderKmip
- CreateKeyServerKmip
- CreatePublicPrivateKeyPair
- DeleteKeyProviderKmip
- DeleteKeyServerKmip
- DisableEncryptionAtRest
- EnableEncryptionAtRest
- GetClientCertificateSignRequest
- GetKeyProviderKmip
- GetKeyServerKmip
- ListKeyProvidersKmip
- ListKeyServersKmip
- ModifyKeyServerKmip
- RemoveKeyServerFromProviderKmip
- SignSshKeys
- TestKeyProviderKmip
- TestKeyServerKmip

Encontre mais informações

- "Documentação do software SolidFire e Element"
- "Documentação para versões anteriores dos produtos NetApp SolidFire e Element"

AddKeyServerToProviderKmip

Você pode usar o `AddKeyServerToProviderKmip` método para atribuir um servidor de chave KMIP (Key Management Interoperability Protocol) ao provedor de chaves especificado. Durante a atribuição, o servidor é contatado para verificar a funcionalidade. Se o servidor de chaves especificado já estiver atribuído ao provedor de chaves especificado, nenhuma ação será tomada e nenhum erro será retornado. Você pode remover a atribuição usando o `RemoveKeyServerFromProviderKmip` método.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyProviderID	A ID do provedor de chaves para atribuir o servidor de chaves.	número inteiro	Nenhum	Sim
KeyServerID	A ID do servidor de chaves a atribuir.	número inteiro	Nenhum	Sim

Valores de retorno

Este método não tem valor de retorno. A atribuição é considerada bem-sucedida, desde que não haja nenhum erro retornado.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "AddKeyServerToProviderKmip",
  "params": {
    "keyProviderID": 1,
    "keyServerID": 15
    },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {}
    }
}
```

Novo desde a versão

11,7

CreateKeyProviderKmip

Você pode usar o `CreateKeyProviderKmip` método para criar um provedor de chaves KMIP (Key Management Interoperability Protocol) com o nome especificado. Um provedor de chaves define um mecanismo e um local para recuperar chaves de autenticação. Quando você cria um novo provedor de chaves KMIP, ele não tem nenhum servidor de chaves KMIP atribuído a ele. Para criar um servidor de chaves KMIP, use o `CreateKeyServerKmip` método. Para atribuí-lo a um provedor, `AddKeyServerToProviderKmip` consulte .

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyProvider Name (Nome do fornecedor)	O nome a ser associado ao provedor de chaves KMIP criado. Este nome é usado apenas para fins de exibição e não precisa ser exclusivo.	cadeia de carateres	Nenhum	Sim

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
KmipKeyProvider	Um objeto contendo detalhes sobre o provedor de chaves recém-criado.	"KeyProviderKmip"

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "CreateKeyProviderKmip",
  "params": {
    "keyProviderName": "ProviderName",
    },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {
    "kmipKeyProvider": {
        "keyProviderName": "ProviderName",
        "keyProviderIsActive": true,
        "kmipCapabilities": "SSL",
        "keyServerIDs": [
            15
        ],
        "keyProviderID": 1
    }
}
}
```

Novo desde a versão

11,7

CreateKeyServerKmip

Você pode usar o `CreateKeyServerKmip` método para criar um servidor de chave KMIP (Key Management Interoperability Protocol) com os atributos especificados. Durante a criação, o servidor não é contatado; ele não precisa existir antes de usar esse método. Para configurações de servidor de chave em cluster, você deve fornecer os nomes de host ou endereços IP de todos os nós de servidor no parâmetro kmipKeyServerHostnames. Você pode usar o `TestKeyServerKmip` método para testar um servidor de chaves.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KmipCaCertificate	O certificado de chave pública da CA raiz do servidor de chaves externo. Isso será usado para verificar o certificado apresentado pelo servidor de chaves externo na comunicação TLS. Para clusters de servidores-chave em que servidores individuais usam CAs diferentes, forneça uma cadeia de carateres concatenada contendo os certificados raiz de todas as CAs.	cadeia de carateres	Nenhum	Sim
KmipClientCertificate	Um certificado PKCS nº 10 X,509 codificado em formato PEM Base64 usado pelo cliente KMIP SolidFire.	cadeia de carateres	Nenhum	Sim
KmipKeyServerHostnames	Array dos nomes de host ou endereços IP associados a este servidor de chaves KMIP. Vários nomes de host ou endereços IP só devem ser fornecidos se os servidores-chave estiverem em uma configuração em cluster.	array de cadeia de carateres	Nenhum	Sim

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KmipKeyServerName	O nome do servidor de chaves KMIP. Este nome é usado apenas para fins de exibição e não precisa ser exclusivo.	cadeia de carateres	Nenhum	Sim
KmipKeyServerPort	O número da porta associada a este servidor de chaves KMIP (normalmente 5696).	número inteiro	Nenhum	Não

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
KmipKeyServer	Um objeto contendo detalhes sobre o servidor de chaves recém-criado.	"KeyServerKmip"

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "CreateKeyServerKmip",
  "params": {
    "kmipCaCertificate": "MIICPDCCAaUCEDyRMcsf9tAbDpq40ES/E...",
    "kmipClientCertificate": "dKkkirWmnWXbj9T/UWZYB2oK0z5...",
    "kmipKeyServerHostnames" : ["server1.hostname.com",
"server2.hostname.com"],
    "kmipKeyServerName" : "keyserverName",
    "kmipKeyServerPort" : 5696
  },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {
        "kmipKeyServer": {
            "kmipCaCertificate":"MIICPDCCAaUCEDyRMcsf9tAbDpq40ES/E...",
            "kmipKeyServerHostnames":[
                "server1.hostname.com", "server2.hostname.com"
            ],
            "keyProviderID":1,
            "kmipKeyServerName":"keyserverName",
            "keyServerID":1
            "kmipKeyServerPort":1,
            "kmipClientCertificate":"dKkkirWmnWXbj9T/UWZYB2oK0z5...",
            "kmipAssignedProviderIsActive":true
        }
     }
}
```

Novo desde a versão

11,7

CreatePublicPrivateKeyPair

Você pode usar o `CreatePublicPrivateKeyPair` método para criar chaves SSL públicas e privadas. Você pode usar essas chaves para gerar solicitações de assinatura de certificado. Só pode haver um par de chaves em uso para cada cluster de armazenamento. Antes de usar esse método para substituir chaves existentes, verifique se as chaves não estão mais em uso por nenhum provedor.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
Nome comum	O campo X,509 Nome distinto **Nome Comum** (CN).	cadeia de carateres	Nenhum	Não
país	O campo X,509 Nome distinto **país** ©.	cadeia de carateres	Nenhum	Não

Nome	Descrição	Tipo	Valor padrão	Obrigatório
EMAILADDRESS	O campo X,509 Nome distinto **Endereço de e-mail** (CORREIO).	cadeia de carateres	Nenhum	Não
localidade	O campo X,509 Nome distinto **Nome da localidade** (L).	cadeia de carateres	Nenhum	Não
organização	O campo X,509 Nome distinto **Nome da Organização** (o).	cadeia de carateres	Nenhum	Não
Unidade organizacional	O campo Nome da Unidade organizacional* (ou) do nome distinto X,509.	cadeia de carateres	Nenhum	Não
estado	O campo X,509 Nome distinto **Estado** ou **Nome da Província** (ST ou SP ou S).	cadeia de carateres	Nenhum	Não

Valores de retorno

Este método não tem valores de retorno. Se não houver nenhum erro, a criação de chave é considerada bem-sucedida.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "CreatePublicPrivateKeyPair",
  "params": {
    "commonName": "Name",
    "country": "US",
    "emailAddress" : "email@domain.com"
   },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {}
}
```

Novo desde a versão

11,7

DeleteKeyProviderKmip

Você pode usar o `DeleteKeyProviderKmip` método para excluir o provedor de chaves KMIP (Inactive Key Management Interoperability Protocol) especificado.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyProviderID	A ID do fornecedor de chaves a eliminar.	número inteiro	Nenhum	Sim

Valores de retorno

Este método não tem valores de retorno. A operação de eliminação é considerada bem-sucedida, desde que não haja erro.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "DeleteKeyProviderKmip",
  "params": {
    "keyProviderID": "1"
    },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {}
}
```

Novo desde a versão

11,7

DeleteKeyServerKmip

Você pode usar o `DeleteKeyServerKmip` método para excluir um servidor de chaves KMIP (Key Management Interoperability Protocol) existente. Você pode excluir um servidor de chaves, a menos que seja o último atribuído a seu provedor, e esse provedor esteja fornecendo chaves que estão atualmente em uso.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyServerID	O ID do servidor de chaves KMIP a ser excluído.	número inteiro	Nenhum	Sim

Valores de retorno

Este método tem os valores sem retorno. A operação de exclusão é considerada bem-sucedida se não houver erros.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "DeleteKeyServerKmip",
  "params": {
     "keyServerID": 15
  },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {}
}
```

Novo desde a versão

11,7

DisableEncryptionAtRest

Você pode usar o `DisableEncryptionAtRest` método para remover a criptografia que foi aplicada anteriormente ao cluster usando o `EnableEncryptionAtRest` método. Este método de desativação é assíncrono e retorna uma resposta antes de a criptografia ser desativada. Você pode usar o `GetClusterInfo` método para poll o sistema para ver quando o processo foi concluído.

Para ver o status atual da criptografia em repouso e/ou criptografia de software em repouso no cluster, use o "obtenha o método de informações do cluster". Pode utilizar a `GetSoftwareEncryptionAtRestInfo` "método para obter informações que o cluster usa para criptografar dados em repouso".

Você não pode usar este método para desativar a criptografia de software em repouso. Para desativar a encriptação de software em repouso, tem de "crie um novo cluster"desativar a encriptação de software em repouso.

Parâmetros

Este método não tem parâmetros de entrada.

Valores de retorno

Este método não tem valores de retorno.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
   "method": "DisableEncryptionAtRest",
   "params": {},
   "id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id" : 1,
  "result" : {}
}
```

Novo desde a versão

9,6

Encontre mais informações

- "GetClusterInfo"
- "Documentação do software SolidFire e Element"
- "Documentação para versões anteriores dos produtos NetApp SolidFire e Element"

EnableEncryptionAtRest

Você pode usar o `EnableEncryptionAtRest` método para ativar a criptografia AES (Advanced Encryption Standard) de 256 bits em repouso no cluster para que o cluster possa gerenciar a chave de criptografia usada para as unidades em cada nó. Esta funcionalidade não está ativada por predefinição.

Para ver o status atual da criptografia em repouso e/ou criptografia de software em repouso no cluster, use o "obtenha o método de informações do cluster". Pode utilizar a `GetSoftwareEncryptionAtRestInfo` "método para obter informações que o cluster usa para criptografar dados em repouso".

Este método não ativa a encriptação de software em repouso. Isso só pode ser feito usando o "criar método de cluster" com `enableSoftwareEncryptionAtRest` definido como `true`.

Quando você ativa a criptografia em repouso, o cluster gerencia automaticamente as chaves de criptografia internamente para as unidades em cada nó do cluster.

Se um keyProviderID for especificado, a senha será gerada e recuperada de acordo com o tipo de provedor de chaves. Isso geralmente é feito usando um servidor de chave KMIP (Key Management Interoperability Protocol) no caso de um provedor de chaves KMIP. Após esta operação, o provedor especificado é considerado ativo e não pode ser excluído até que a criptografia em repouso seja desativada usando o `DisableEncryptionAtRest` método.

Se você tiver um tipo de nó com um número de modelo terminando em "-NE", a `EnableEncryptionAtRest` chamada de método falhará com uma resposta de "criptografia não permitida. Cluster detetado nó não criptografado".

Você só deve ativar ou desativar a criptografia quando o cluster estiver em execução e em um estado saudável. Você pode ativar ou desativar a criptografia a seu critério e sempre que precisar.

Este processo é assíncrono e retorna uma resposta antes de a criptografia ser ativada. Você pode usar o `GetClusterInfo` método para poll o sistema para ver quando o processo foi concluído.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyProviderID	O ID de um provedor de chaves KMIP a ser usado.	número inteiro	Nenhum	Não

Valores de retorno

Este método não tem valores de retorno.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
   "method": "EnableEncryptionAtRest",
   "params": {},
   "id": 1
}
```

Exemplos de resposta

Este método retorna uma resposta semelhante ao exemplo a seguir do método EnableEncryptionAtRest. Não há nenhum resultado para relatar.

```
{
   "id": 1,
   "result": {}
}
```

Enquanto a criptografia em repouso está sendo ativada em um cluster, GetClusterInfo retorna um resultado descrevendo o estado de criptografia em repouso ("criptoporAtRestState") como "habilitando". Depois que a encriptação em repouso estiver totalmente ativada, o estado devolvido muda para "activado".

```
{
   "id": 1,
      "result": {
         "clusterInfo": {
            "attributes": { },
               "encryptionAtRestState": "enabling",
            "ensemble": [
               "10.10.5.94",
               "10.10.5.107",
               "10.10.5.108"
            ],
            "mvip": "192.168.138.209",
            "mvipNodeID": 1,
            "name": "Marshall",
            "repCount": 2,
            "svip": "10.10.7.209",
            "svipNodeID": 1,
            "uniqueID": "91dt"
      }
   }
}
```

Novo desde a versão

9,6

Encontre mais informações

- "SecureEraseDrives"
- "GetClusterInfo"
- "Documentação do software SolidFire e Element"

- "Documentação para versões anteriores dos produtos NetApp SolidFire e Element"

GetClientCertificateSignRequest

Você pode usar o `GetClientCertificateSignRequest` método para gerar uma solicitação de assinatura de certificado que pode ser assinada por uma autoridade de certificação para gerar um certificado de cliente para o cluster. Certificados assinados são necessários para estabelecer um relacionamento de confiança para interagir com serviços externos.

Parâmetros

Este método não tem parâmetros de entrada.

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
ClientCertificateSignRequest	Um pedido de sinal de certificado de cliente PKCS nº 10 X,509 codificado em formato PEM Base64.	cadeia de carateres

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "GetClientCertificateSignRequest",
  "params": {
  },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {
        "clientCertificateSignRequest":
"MIIByjCCATMCAQAwgYkxCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpDYWxpZm9ybm..."
   }
}
```

Novo desde a versão

11,7

GetKeyProviderKmip

Você pode usar o `GetKeyProviderKmip` método para recuperar informações sobre o provedor de chaves KMIP (Key Management Interoperability Protocol) especificado.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyProviderID	A ID do objeto do provedor de chaves KMIP a retornar.	número inteiro	Nenhum	Sim

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
KmipKeyProvider	Um objeto contendo detalhes sobre o provedor de chaves solicitado.	"KeyProviderKmip"

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "GetKeyProviderKmip",
  "params": {
    "keyProviderID": 15
    },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
    {
    "kmipKeyProvider": {
        "keyProviderID": 15,
        "kmipCapabilities": "SSL",
        "keyProviderIsActive": true,
        "keyServerIDs": [
            1
        ],
        "keyProviderName": "ProviderName"
    }
   }
}
```

Novo desde a versão

11,7

GetKeyServerKmip

Você pode usar o `GetKeyServerKmip` método para retornar informações sobre o servidor de chaves KMIP (Protocolo de interoperabilidade de Gerenciamento de chaves especificado).

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyServerID	O ID do servidor de chaves KMIP para retornar informações.	número inteiro	Nenhum	Sim

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
KmipKeyServer	Um objeto contendo detalhes sobre o servidor de chaves solicitado.	"KeyServerKmip"

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "GetKeyServerKmip",
  "params": {
     "keyServerID": 15
  },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {
        "kmipKeyServer": {
            "kmipCaCertificate":"MIICPDCCAaUCEDyRMcsf9tAbDpq40ES/E...",
            "kmipKeyServerHostnames":[
                "server1.hostname.com", "server2.hostname.com"
            ],
            "keyProviderID":1,
            "kmipKeyServerName":"keyserverName",
            "keyServerID":15
            "kmipKeyServerPort":1,
            "kmipClientCertificate":"dKkkirWmnWXbj9T/UWZYB2oK0z5...",
            "kmipAssignedProviderIsActive":true
        }
     }
}
```

Novo desde a versão

11,7

GetSoftwareEncryptionAtRestInfo

Você pode usar o `GetSoftwareEncryptionAtRestInfo` método para obter informações de criptografia de software em repouso que o cluster usa para criptografar dados em repouso.

Parâmetros

Este método não tem parâmetros de entrada.

Valores de retorno

Este método tem os seguintes valores de retorno:

Parâmetro	Descrição	Tipo	Opcional
MasterKeyInfo	Informações sobre a chave mestra de criptografia em repouso do software atual.	EncryptionKeyInfo	Verdadeiro

Parâmetro	Descrição	Tipo	Opcional
RekeyMasterKeyAsyncResultID	O ID de resultado assíncrono da operação de rechavear atual ou mais recente (se houver), se ainda não tiver sido excluído. `GetAsyncResult` a saída incluirá um `newKey` campo que contém informações sobre a nova chave mestra e um `keyToDecommission` campo que contém informações sobre a chave antiga.	número inteiro	Verdadeiro
estado	O estado atual de criptografia em repouso do software. Os valores possíveis são `disabled` ou `enabled`.	cadeia de carateres	Falso
versão	Um número de versão que é incrementado sempre que a criptografia de software em repouso é ativada.	número inteiro	Falso

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
 "method": "getsoftwareencryptionatrestinfo"
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
    "id": 1,
    "result": {
        "masterKeyInfo": {
            "keyCreatedTime": "2021-09-20T23:15:56Z",
            "keyID": "4d80a629-a11b-40ab-8b30-d66dd5647cfd",
            "keyManagementType": "internal"
        },
        "state": "enabled",
        "version": 1
    }
}
```

Novo desde a versão

12,3

Encontre mais informações

- "Documentação do software SolidFire e Element"
- "Documentação para versões anteriores dos produtos NetApp SolidFire e Element"

ListKeyProvidersKmip

Você pode usar o `ListKeyProvidersKmip` método para recuperar uma lista de todos os provedores de chave KMIP (Key Management Interoperability Protocol) existentes. Você pode filtrar a lista especificando parâmetros adicionais.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyProviderIsActive	Os filtros devolveram objetos de servidor de chave KMIP com base se eles estão ativos. Valores possíveis: • Verdadeiro: Retorna apenas provedores de chave KMIP que estão ativos (fornecendo chaves que estão atualmente em uso). • Falso: Retorna apenas provedores de chave KMIP que estão inativos (não fornecendo nenhuma chave e capazes de ser excluídos). Se omitido, os provedores de chave KMIP retornados não são filtrados com base se estão ativos.	booleano	Nenhum	Não

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KmipKeyProviderHasServerAssigned	Os filtros devolveram provedores de chave KMIP com base se eles têm um servidor de chave KMIP atribuído. Valores possíveis: • True: Retorna apenas provedores de chave KMIP que têm um servidor de chave KMIP atribuído. • Falso: Retorna apenas provedores de chave KMIP que não têm um servidor de chave KMIP atribuído. Se omitido, os provedores de chave KMIP retornados não são filtrados com base se eles têm um servidor de chave KMIP atribuído.	booleano	Nenhum	Não

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
KmipKeyProviders	Uma lista de provedores de chave KMIP que foram criados.	"KeyProviderKmip" array

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "ListKeyProvidersKmip",
  "params": {},
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
    {
    "kmipKeyProviders": [
        {
            "keyProviderID": 15,
            "kmipCapabilities": "SSL",
            "keyProviderIsActive": true,
            "keyServerIDs": [
                1
            ],
            "keyProviderName": "KeyProvider1"
        }
    ]
}
}
```

Novo desde a versão

11,7

ListKeyServersKmip

Você pode usar o `ListKeyServersKmip` método para listar todos os servidores de chave KMIP (Key Management Interoperability Protocol) que foram criados. Você pode filtrar os resultados especificando parâmetros adicionais.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyProviderID	Quando especificado, o método retorna somente servidores de chave KMIP atribuídos ao provedor de chaves KMIP especificado. Se omitido, os servidores de chave KMIP retornados não serão filtrados com base se eles são atribuídos ao provedor de chaves KMIP especificado.	número inteiro	Nenhum	Não
KmipAssignedProviderIsActive	Os filtros devolveram objetos de servidor de chave KMIP com base se eles estão ativos. Valores possíveis: • True: Retorna apenas servidores de chave KMIP que estão ativos (fornecendo chaves que estão atualmente em uso). • False: Retorna apenas servidores de chave KMIP que estão inativos (não fornecendo nenhuma chave e capazes de ser excluídos). Se omitido, os servidores de chave KMIP retornados não são filtrados com base se estão ativos.	booleano	Nenhum	Não

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KmipHasProviderAssigned	Os filtros devolveram servidores de chave KMIP com base se eles têm um provedor de chaves KMIP atribuído. Valores possíveis: • Verdadeiro: Retorna apenas servidores de chave KMIP que têm um provedor de chaves KMIP atribuído. • Falso: Retorna apenas servidores de chave KMIP que não têm um provedor de chave KMIP atribuído. Se omitido, os servidores de chave KMIP retornados não são filtrados com base se têm um provedor de chaves KMIP atribuído.	booleano	Nenhum	Não

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
KmipKeyServers	A lista completa de servidores de chave KMIP que foram criados.	"KeyServerKmip" array

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "ListKeyServersKmip",
  "params": {},
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
    "kmipKeyServers": [
        {
            "kmipKeyServerName": "keyserverName",
            "kmipClientCertificate": "dKkkirWmnWXbj9T/UWZYB2oK0z5...",
            "keyServerID": 15,
            "kmipAssignedProviderIsActive": true,
            "kmipKeyServerPort": 5696,
            "kmipCaCertificate": "MIICPDCCAaUCEDyRMcsf9tAbDpq40ES/E...",
            "kmipKeyServerHostnames": [
                "server1.hostname.com", "server2.hostname.com"
            ],
            "keyProviderID": 1
        }
    ]
}
```

Novo desde a versão

11,7

ModifyKeyServerKmip

Você pode usar o `ModifyKeyServerKmip` método para modificar um servidor de chave KMIP (Key Management Interoperability Protocol) existente para os atributos especificados. Embora o único parâmetro necessário seja o keyServerID, uma solicitação contendo apenas o keyServerID não tomará nenhuma ação e não retornará nenhum erro. Quaisquer outros parâmetros especificados substituirão os valores existentes para o servidor de chaves pelo keyServerID especificado. O servidor de chaves é contatado durante a operação para garantir que ele esteja funcional. Você pode fornecer vários nomes de host ou endereços IP com o parâmetro kmipKeyServerHostnames, mas apenas se os servidores-chave estiverem em uma configuração em cluster.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyServerID	O ID do KMIP Key Server a modificar.	número inteiro	Nenhum	Sim
KmipCaCertificate	O certificado de chave pública da CA raiz do servidor de chaves externo. Isso será usado para verificar o certificado apresentado pelo servidor de chaves externo na comunicação TLS. Para clusters de servidores-chave em que servidores individuais usam CAs diferentes, forneça uma cadeia de carateres concatenada contendo os certificados raiz de todas as CAs.	cadeia de carateres	Nenhum	Não
KmipClientCertificate	Um certificado PKCS nº 10 X,509 codificado em formato PEM Base64 usado pelo cliente KMIP SolidFire.	cadeia de carateres	Nenhum	Não
KmipKeyServerHostnames	Array dos nomes de host ou endereços IP associados a este servidor de chaves KMIP. Vários nomes de host ou endereços IP só devem ser fornecidos se os servidores-chave estiverem em uma configuração em cluster.	array de cadeia de carateres	Nenhum	Não

KmipKeyServerName	O nome do servidor de chaves KMIP. Este nome é usado apenas para fins de exibição e não precisa ser exclusivo.	cadeia de carateres	Nenhum	Não
KmipKeyServerPort	O número da porta associada a este servidor de chaves KMIP (normalmente 5696).	número inteiro	Nenhum	Não

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
KmipKeyServer	Um objeto contendo detalhes sobre o servidor de chaves recém-modificado.	"KeyServerKmip"

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "ModifyKeyServerKmip",
  "params": {
    "keyServerID": 15
    "kmipCaCertificate": "CPDCCAaUCEDyRMcsf9tAbDpq40ES/E...",
    "kmipClientCertificate": "kirWmnWXbj9T/UWZYB2oK0z5...",
    "kmipKeyServerHostnames" : ["server1.hostname.com",
"server2.hostname.com"],
    "kmipKeyServerName" : "keyserverName",
    "kmipKeyServerPort" : 5696
  },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {
        "kmipKeyServer": {
            "kmipCaCertificate":"CPDCCAaUCEDyRMcsf9tAbDpq40ES/E...",
            "kmipKeyServerHostnames":[
                "server1.hostname.com", "server2.hostname.com"
            ],
            "keyProviderID":1,
            "kmipKeyServerName":"keyserverName",
            "keyServerID":1
            "kmipKeyServerPort":1,
            "kmipClientCertificate":"kirWmnWXbj9T/UWZYB2oK0z5...",
            "kmipAssignedProviderIsActive":true
        }
     }
}
```


Novo desde a versão

11,7

RekeySoftwareEncryptionAtRestMasterKey

Você pode usar o `RekeySoftwareEncryptionAtRestMasterKey` método para rechavear a chave mestra de criptografia em repouso usada para criptografar DEKs (chaves de criptografia de dados). Durante a criação do cluster, a criptografia de software em repouso é configurada para usar o gerenciamento de chave interna (IKM). Este método de rechavear pode ser usado após a criação do cluster para usar o IKM ou o EKM (External Key Management).

Parâmetros

Este método tem os seguintes parâmetros de entrada. Se o `keyManagementType` parâmetro não for especificado, a operação de rechavear é executada usando a configuração de gerenciamento de chaves existente. Se o `keyManagementType` for especificado e o provedor de chaves for externo, o `keyProviderID` parâmetro também deve ser usado.

Parâmetro	Descrição	Tipo	Opcional
KeyManagementType	O tipo de gerenciamento de chaves usado para gerenciar a chave mestra. Os valores possíveis são `Internal`: : Rechavear utilizando a gestão de chaves internas. `External`: Rechavear utilizando a gestão de chaves externas. Se este parâmetro não for especificado, a operação de rechavear é executada utilizando a configuração de gestão de chaves existente.	cadeia de carateres	Verdadeiro
KeyProviderID	A ID do fornecedor de chaves a utilizar. Este é um valor único retornado como parte de um dos `CreateKeyProvider` métodos. A ID só é necessária quando `keyManagementType` é `External` e é inválida.	número inteiro	Verdadeiro

Valores de retorno

Este método tem os seguintes valores de retorno:

Parâmetro	Descrição	Tipo	Opcional
AsyncHandle	Determine o estado da operação de rechavear utilizando este `asyncHandle` valor com `GetAsyncResult`. `GetAsyncResult` a saída incluirá um `newKey` campo que contém informações sobre a nova chave mestra e um `keyToDecommission` campo que contém informações sobre a chave antiga.	número inteiro	Falso

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

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

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
    "asyncHandle": 1
}
```

Novo desde a versão

12,3

Encontre mais informações

- "Documentação do software SolidFire e Element"
- "Documentação para versões anteriores dos produtos NetApp SolidFire e Element"

RemoveKeyServerFromProviderKmip

Você pode usar o `RemoveKeyServerFromProviderKmip` método para anular a atribuição do servidor de chave KMIP (Key Management Interoperability Protocol) especificado do provedor ao qual foi atribuído. Você pode anular a atribuição de um servidor de chaves de seu provedor, a menos que seja o último e seu provedor esteja ativo (fornecendo chaves que estão atualmente em uso). Se o servidor de chaves especificado não for atribuído a um provedor, nenhuma ação será tomada e nenhum erro será retornado.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyServerID	A ID do servidor de chaves KMIP a ser desatribuída.	número inteiro	Nenhum	Sim

Valores de retorno

Este método não tem valores de retorno. A remoção é considerada bem-sucedida, desde que nenhum erro seja retornado.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "RemoveKeyServerFromProviderKmip",
  "params": {
    "keyServerID": 1
   },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {}
    }
}
```

Novo desde a versão

11,7

SignSshKeys

Depois que o SSH estiver ativado no cluster usando o "Método EnableSSH", você poderá usar o `SignSshKeys` método para obter acesso a um shell em um nó.

Começando com o elemento 12,5, `sfreadonly` é uma nova conta de sistema que permite a solução de problemas básicos em um nó. Esta API permite o acesso SSH usando a `sfreadonly` conta do sistema em todos os nós do cluster.

> A menos que seja aconselhado pelo suporte da NetApp, quaisquer alterações no sistema não são suportadas, anulando o contrato de suporte e pode resultar em instabilidade ou inacessibilidade dos dados.

Depois de usar o método, você deve copiar o keychain da resposta, salvá-lo no sistema que estará iniciando a

conexão SSH e, em seguida, executar o seguinte comando:

```
ssh -i <identity_file> sfreadonly@<node_ip>
```

```
`identity_file` É um arquivo do qual a identidade (chave privada) para autenticação de chave pública é lida e `node_ip` é o endereço IP do nó. Para obter mais informações sobre `identity_file`o , consulte a página man do SSH.
```

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
duração	Número inteiro de 1 a 24 refletindo o número de horas para a chave assinada ser válido. Se a duração não for especificada, o padrão será usado.	número inteiro	1	Não

Nome	Descrição	Tipo	Valor padrão	Obrigatório
PublicKey	Se fornecido, esse parâmetro retornará somente a signed_public_key em vez de criar um keychain completo para o usuário.  Chaves públicas enviadas usando a barra de URL em um navegador com + são interpretadas como espaçadas e quebras de assinatura.	cadeia de carateres	Nulo	Não
sfadmin	Permite o acesso à conta de shell sfadmin quando você faz a chamada de API com acesso de cluster supportAdmin ou quando o nó não está em um cluster.	booleano	Falso	Não

Valores de retorno

Este método tem os seguintes valores de retorno:

Nome	Descrição	Tipo
keygen_status	Contém a identidade na chave assinada, os principais permitidos e as datas de início e fim válidas para a chave.	cadeia de carateres
chave_privada	Um valor de chave SSH privada só será retornado se a API estiver gerando um keychain completo para o usuário final. O valor é Base64 codificado; você deve decodificar o valor quando ele é gravado em um arquivo para garantir que ele seja lido como uma chave privada válida.	cadeia de carateres
public_key	Um valor de chave SSH pública só será retornado se a API estiver gerando um keychain completo para o usuário final. Quando você passa um parâmetro public_key para o método API, apenas o `signed_public_key` valor é retornado na resposta.	cadeia de carateres
signed_public_key	A chave pública SSH que resulta da assinatura da chave pública, seja ela fornecida pelo usuário ou gerada pela API.	cadeia de carateres

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
    "method": "SignSshKeys",
    "params": {
        "duration": 2,
        "publicKey":<string>
    },
    "id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
  "id": null,
  "result": {
    "signedKeys": {
      "keygen_status": <keygen_status>,
      "signed_public_key": <signed_public_key>
    }
  }
}
```

Neste exemplo, uma chave pública é assinada e retornada válida para a duração (1-24 horas).

Novo desde a versão

12,5

TestKeyProviderKmip

Você pode usar o `TestKeyProviderKmip` método para testar se o provedor de chaves KMIP (Key Management Interoperability Protocol) especificado está acessível e funcionando normalmente.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyProviderID	A ID do fornecedor de chaves a testar.	número inteiro	Nenhum	Sim

Valores de retorno

Este método não tem valores de retorno. O teste é considerado bem-sucedido desde que nenhum erro seja retornado.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "TestKeyProviderKmip",
  "params": {
    "keyProviderID": 15
  },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {}
    }
}
```

Novo desde a versão

11,7

TestKeyServerKmip

Você pode usar o `TestKeyServerKmip` método para testar se o servidor de chaves KMIP (Key Management Interoperability Protocol) especificado está acessível e funcionando normalmente.

Parâmetros

Este método tem os seguintes parâmetros de entrada:

Nome	Descrição	Tipo	Valor padrão	Obrigatório
KeyServerID	O ID do servidor de chaves KMIP a testar.	número inteiro	Nenhum	Sim

Valores de retorno

Este método não tem valores de retorno. O teste é considerado bem-sucedido se não houver erros retornados.

Exemplo de solicitação

As solicitações para este método são semelhantes ao seguinte exemplo:

```
{
  "method": "TestKeyServerKmip",
  "params": {
    "keyServerID": 15
  },
"id": 1
}
```

Exemplo de resposta

Este método retorna uma resposta semelhante ao seguinte exemplo:

```
{
   "id": 1,
   "result":
      {}
    }
```

Novo desde a versão

11,7

Informações sobre direitos autorais

Informações sobre marcas comerciais