Méthodes de l'API de sécurité

Element Software

NetApp
November 18, 2025

Sommaire

Méthodes de l’API de sécurité

AjouterKeyServerToProviderKmip

Vous pouvez utiliser le `AddKeyServerToProviderKmip` méthode pour attribuer un serveur de clés KMIP (Key Management Interoperability Protocol) au fournisseur de clés spécifié. Lors de l’attribution, le serveur est contacté pour vérifier son bon fonctionnement. Si le serveur de clés spécifié est déjà attribué au fournisseur de clés spécifié, aucune action n’est entreprise et aucune erreur n’est renvoyée. Vous pouvez supprimer la tâche en utilisant le `RemoveKeyServerFromProviderKmip` méthode.

Paramètres

Cette méthode possède les paramètres d’entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du fournisseur de clés	L’identifiant du fournisseur de clés auquel attribuer le serveur de clés.	entier	Aucune	Oui
ID du serveur de clés	L’identifiant du serveur de clés à attribuer.	entier	Aucune	Oui

Valeurs de retour

Cette méthode ne renvoie aucune valeur. La tâche est considérée comme réussie tant qu’aucune erreur n’est renvoyée.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l’exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

CréerKeyProviderKmip

Vous pouvez utiliser le `CreateKeyProviderKmip` méthode pour créer un fournisseur de clés KMIP (Key Management Interoperability Protocol) avec le nom spécifié. Un fournisseur de clés définit un mécanisme et un emplacement pour récupérer les clés d'authentification. Lorsque vous créez un nouveau fournisseur de clés KMIP, aucun serveur de clés KMIP ne lui est attribué. Pour créer un serveur de clés KMIP, utilisez le `CreateKeyServerKmip` méthode. Pour l'attribuer à un fournisseur, voir `AddKeyServerToProviderKmip` .

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
Nom du fournisseur de clés	Le nom à associer au fournisseur de clés KMIP créé. Ce nom est utilisé uniquement à des fins d'affichage et n'a pas besoin d'être unique.	chaîne	Aucune	Oui

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type

kmipKeyProvider	Un objet contenant des informations sur le fournisseur de clés nouvellement créé.	"KeyProviderKmip"

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

CréerKeyServerKmip

Vous pouvez utiliser le `CreateKeyServerKmip` méthode pour créer un serveur de clés KMIP (Key Management Interoperability Protocol) avec les attributs spécifiés. Lors de la

création, le serveur n'est pas contacté ; il n'est pas nécessaire qu'il existe avant d'utiliser cette méthode. Pour les configurations de serveurs de clés en cluster, vous devez fournir les noms d'hôte ou les adresses IP de tous les nœuds du serveur dans le paramètre kmipKeyServerHostnames. Vous pouvez utiliser le `TestKeyServerKmip` méthode pour tester un serveur de clés.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
kmipCaCertificate	Le certificat de clé publique de l'autorité de certification racine du serveur de clés externe. Cela servira à vérifier le certificat présenté par le serveur de clés externe dans la communication TLS. Pour les clusters de serveurs critiques où chaque serveur utilise une autorité de certification différente, fournissez une chaîne concaténée contenant les certificats racine de toutes les autorités de certification.	chaîne	Aucune	Oui
Certificat client kmip	Un certificat X.509 PKCS#10 au format PEM encodé en Base64 utilisé par le client Solidfire KMIP.	chaîne	Aucune	Oui

Nom	Description	Type	Valeur par défaut	Obligatoire
Noms d'hôte du serveur de clés kmip	Tableau des noms d'hôtes ou des adresses IP associés à ce serveur de clés KMIP. Plusieurs noms d'hôtes ou adresses IP ne doivent être fournis que si les serveurs clés sont configurés en cluster.	tableau de chaînes	Aucune	Oui
Nom du serveur de clés kmip	Le nom du serveur de clés KMIP. Ce nom est utilisé uniquement à des fins d'affichage et n'a pas besoin d'être unique.	chaîne	Aucune	Oui
kmipKeyServerPort	Le numéro de port associé à ce serveur de clés KMIP (généralement 5696).	entier	Aucune	Non

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type
kmipKeyServer	Un objet contenant des informations sur le serveur de clés nouvellement créé.	"KeyServerKmip"

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

```
{
   "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
        }
    }
}
```

Nouveautés depuis la version

11,7

Créer une paire de clés publique/privée

Vous pouvez utiliser le `CreatePublicPrivateKeyPair` Méthode de création de clés SSL publiques et privées. Vous pouvez utiliser ces clés pour générer des demandes de

signature de certificat. Une seule paire de clés peut être utilisée pour chaque cluster de stockage. Avant d'utiliser cette méthode pour remplacer des clés existantes, assurez-vous que ces clés ne sont plus utilisées par aucun fournisseur.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
Nom commun	Le champ **Nom commun** (CN) du nom distinctif X.509.	chaîne	Aucune	Non
pays	Le champ de nom distinctif X.509 **Pays** ©.	chaîne	Aucune	Non
Adresse e-mail	Le champ **Adresse e-mail** du nom distinctif X.509 (MAIL).	chaîne	Aucune	Non
localité	Le champ **Nom de la localité** (L) du nom distinctif X.509.	chaîne	Aucune	Non
organisation	Le champ **Nom de l'organisation** du nom distinctif X.509 (O).	chaîne	Aucune	Non
unité organisationnelle	Le champ **Nom de l'unité organisationnelle** (OU) du nom distinctif X.509.	chaîne	Aucune	Non
État	Le champ du nom distinctif X.509 **État** ou **Nom de la province** (ST ou SP ou S).	chaîne	Aucune	Non

Valeurs de retour

Cette méthode ne renvoie aucune valeur. En l'absence d'erreur, la création de la clé est considérée comme réussie.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

DeleteKeyProviderKmip

Vous pouvez utiliser le `DeleteKeyProviderKmip` méthode pour supprimer le fournisseur de clés KMIP (Key Management Interoperability Protocol) inactif spécifié.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du fournisseur de clés	L'identifiant du fournisseur de clés à supprimer.	entier	Aucune	Oui

Valeurs de retour

Cette méthode ne renvoie aucune valeur. L'opération de suppression est considérée comme réussie tant qu'aucune erreur ne survient.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

SupprimerKeyServerKmip

Vous pouvez utiliser le `DeleteKeyServerKmip` méthode pour supprimer un serveur de clés KMIP (Key Management Interoperability Protocol) existant. Vous pouvez supprimer un serveur de clés, sauf s'il s'agit du dernier serveur attribué à son fournisseur et que ce fournisseur fournit des clés actuellement utilisées.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du serveur de clés	L'identifiant du serveur de clés KMIP à supprimer.	entier	Aucune	Oui

Valeurs de retour

Cette méthode ne renvoie aucune valeur. L'opération de suppression est considérée comme réussie en l'absence d'erreurs.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

Désactiver le chiffrement au repos

Vous pouvez utiliser le `DisableEncryptionAtRest` méthode pour supprimer le chiffrement qui était précédemment appliqué au cluster en utilisant la `EnableEncryptionAtRest` méthode. Cette méthode de désactivation est asynchrone et renvoie une réponse avant que le chiffrement ne soit désactivé. Vous pouvez utiliser le `GetClusterInfo` méthode permettant d'interroger le système pour savoir quand le processus est terminé.

- Cette méthode ne permet pas de désactiver le chiffrement logiciel au repos. Pour désactiver le chiffrement logiciel au repos, vous devez"créer un nouveau cluster" avec le chiffrement logiciel au repos désactivé.
- Pour consulter l'état actuel du chiffrement au repos, du chiffrement logiciel au repos ou des deux sur le cluster, utilisez "méthode d'obtention d'informations sur le cluster". Vous pouvez utiliser le `GetSoftwareEncryptionAtRestInfo` "méthode pour obtenir des informations sur la manière dont le cluster chiffre les données au repos".

Paramètres

Cette méthode ne requiert aucun paramètre d'entrée.

Valeurs de retour

Cette méthode ne renvoie aucune valeur.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

9,6

Trouver plus d'informations

- "Obtenir les informations du cluster"
- "Documentation logicielle SolidFire et Element"
- "Documentation relative aux versions antérieures des produits NetApp SolidFire et Element"

Activer le chiffrement au repos

Vous pouvez utiliser le `EnableEncryptionAtRest` méthode permettant d'activer le chiffrement AES 256 bits au repos sur le cluster afin que celui-ci puisse gérer la clé de chiffrement utilisée pour les disques sur chaque nœud. Cette fonctionnalité n'est pas activée par défaut.

- Pour consulter l'état actuel du chiffrement au repos et/ou du chiffrement logiciel au repos sur le cluster, utilisez"méthode d'obtention d'informations sur le cluster" . Vous pouvez utiliser le `GetSoftwareEncryptionAtRestInfo` "méthode pour obtenir des informations sur la manière dont le cluster chiffre les données au repos".
- Cette méthode ne permet pas le chiffrement logiciel au repos. Cela ne peut être fait qu'en utilisant le"méthode de création de cluster" avec `enableSoftwareEncryptionAtRest` défini à `true` .

Lorsque vous activez le chiffrement au repos, le cluster gère automatiquement en interne les clés de chiffrement des disques sur chaque nœud du cluster.

Si un keyProviderID est spécifié, le mot de passe est généré et récupéré en fonction du type de fournisseur de clés. Cela se fait généralement à l'aide d'un serveur de clés KMIP (Key Management Interoperability Protocol) dans le cas d'un fournisseur de clés KMIP. Après cette opération, le fournisseur spécifié est considéré comme actif et ne peut être supprimé tant que le chiffrement au repos n'est pas désactivé à l'aide de `DisableEncryptionAtRest` méthode.

Si vous avez un type de nœud dont le numéro de modèle se termine par « -NE », le `EnableEncryptionAtRest` L'appel de méthode échouera avec la réponse « Chiffrement non autorisé ». Le cluster a détecté un nœud non cryptable.

Vous ne devez activer ou désactiver le chiffrement que lorsque le cluster est en fonctionnement et en bon état de fonctionnement. Vous pouvez activer ou désactiver le chiffrement à votre discrétion et aussi souvent que nécessaire.

Ce processus est asynchrone et renvoie une réponse avant que le chiffrement ne soit activé. Vous pouvez utiliser le `GetClusterInfo` méthode permettant d'interroger le système pour savoir quand le processus est terminé.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du fournisseur de clés	L'identifiant du fournisseur de clés KMIP à utiliser.	entier	Aucune	Non

Valeurs de retour

Cette méthode ne renvoie aucune valeur.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemples de réponses

Cette méthode renvoie une réponse similaire à l'exemple suivant de la méthode EnableEncryptionAtRest. Aucun résultat à signaler.

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

Lorsque le chiffrement au repos est activé sur un cluster, GetClusterInfo renvoie un résultat décrivant l'état du chiffrement au repos (« encryptionAtRestState ») comme étant « activé ». Une fois le chiffrement au repos entièrement activé, l'état renvoyé passe à « activé ».

```
{
   "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"
      }
   }
}
```


Nouveautés depuis la version

9,6

Trouver plus d'informations

- "SecureEraseDrives"
- "Obtenir les informations du cluster"
- "Documentation logicielle SolidFire et Element"
- "Documentation relative aux versions antérieures des produits NetApp SolidFire et Element"

Demande de signature du certificat client

Vous pouvez utiliser le `GetClientCertificateSignRequest` méthode permettant de générer une demande de signature de certificat pouvant être signée par une autorité de certification afin de générer un certificat client pour le cluster. Des certificats signés sont nécessaires pour établir une relation de confiance permettant d'interagir avec des services externes.

Paramètres

Cette méthode ne requiert aucun paramètre d'entrée.

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type
Demande de signature de certificat client	Demande de signature de certificat client X.509 PKCS#10 au format PEM encodé en Base64.	chaîne

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

GetKeyProviderKmip

Vous pouvez utiliser le `GetKeyProviderKmip` méthode permettant de récupérer des informations sur le fournisseur de clés du protocole d'interopérabilité de gestion de clés (KMIP) spécifié.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du fournisseur de clés	L'identifiant de l'objet fournisseur de clés KMIP à renvoyer.	entier	Aucune	Oui

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type
kmipKeyProvider	Un objet contenant des informations sur le fournisseur de clés demandé.	"KeyProviderKmip"

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

GetKeyServerKmip

Vous pouvez utiliser le `GetKeyServerKmip` méthode permettant de renvoyer des informations sur le serveur de clés KMIP (Key Management Interoperability Protocol) spécifié.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du serveur de clés	L'identifiant du serveur de clés KMIP sur lequel renvoyer des informations.	entier	Aucune	Oui

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type
kmipKeyServer	Un objet contenant des informations sur le serveur de clés demandé.	"KeyServerKmip"

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

```
{
   "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
        }
    }
}
```

Nouveautés depuis la version

11,7

GetSoftwareEncryptionAtRestInfo

Vous pouvez utiliser le `GetSoftwareEncryptionAtRestInfo` méthode permettant d'obtenir des informations sur le chiffrement logiciel au repos utilisé par le cluster pour chiffrer les données au repos.

Paramètres

Cette méthode ne requiert aucun paramètre d'entrée.

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Paramètre	Description	Type	Facultatif
masterKeyInfo	Informations concernant la clé principale actuelle de chiffrement au repos du logiciel.	Informations sur la clé de chiffrement	Vrai
rekeyMasterKeyAsyncResultID	L'identifiant du résultat asynchrone de l'opération de renouvellement de clé actuelle ou la plus récente (le cas échéant), s'il n'a pas encore été supprimé. `GetAsyncResult` Le résultat comprendra un `newKey` champ contenant des informations sur la nouvelle clé principale et un `keyToDecommission` champ contenant des informations sur l'ancienne clé.	entier	Vrai
État	État actuel du chiffrement logiciel au repos. Les valeurs possibles sont `disabled` ou `enabled` .	chaîne	FAUX
version	Un numéro de version incrémenté à chaque activation du chiffrement logiciel au repos.	entier	FAUX

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

12,3

Trouver plus d'informations

- "Documentation logicielle SolidFire et Element"
- "Documentation relative aux versions antérieures des produits NetApp SolidFire et Element"

ListKeyProvidersKmip

Vous pouvez utiliser le `ListKeyProvidersKmip` méthode permettant de récupérer une liste de tous les fournisseurs de clés existants du protocole d'interopérabilité de gestion des clés (KMIP). Vous pouvez filtrer la liste en spécifiant des paramètres supplémentaires.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
keyProviderIsActive	Les filtres ont renvoyé les objets serveur de clés KMIP en fonction de leur état d'activité. Valeurs possibles : • vrai : renvoie uniquement les fournisseurs de clés KMIP actifs (fournissant des clés actuellement utilisées). • false : Renvoie uniquement les fournisseurs de clés KMIP inactifs (ne fournissant aucune clé et pouvant être supprimés). Si cette option est omise, les fournisseurs de clés KMIP renvoyés ne sont pas filtrés en fonction de leur statut (actif ou non).	booléen	Aucune	Non

Nom	Description	Type	Valeur par défaut	Obligatoire
kmipKeyProviderHasServerAssigned	Les filtres ont renvoyé les fournisseurs de clés KMIP en fonction de la présence ou non d'un serveur de clés KMIP attribué. Valeurs possibles : • true : Renvoie uniquement les fournisseurs de clés KMIP auxquels un serveur de clés KMIP est attribué. • false : Renvoie uniquement les fournisseurs de clés KMIP qui n'ont pas de serveur de clés KMIP attribué. Si cette option est omise, les fournisseurs de clés KMIP renvoyés ne sont pas filtrés en fonction de la présence ou non d'un serveur de clés KMIP attribué.	booléen	Aucune	Non

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type
Fournisseurs de clés kmip	Liste des fournisseurs de clés KMIP créés.	"KeyProviderKmip"tableau

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

ListKeyServersKmip

Vous pouvez utiliser le `ListKeyServersKmip` méthode permettant de lister tous les serveurs de clés KMIP (Key Management Interoperability Protocol) qui ont été créés. Vous pouvez filtrer les résultats en spécifiant des paramètres supplémentaires.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du fournisseur de clés	Lorsqu'elle est spécifiée, la méthode ne renvoie que les serveurs de clés KMIP affectés au fournisseur de clés KMIP spécifié. Si cette option est omise, les serveurs de clés KMIP renvoyés ne seront pas filtrés en fonction de leur appartenance ou non au fournisseur de clés KMIP spécifié.	entier	Aucune	Non
kmipAssignedProviderIsActive	Les filtres ont renvoyé les objets serveur de clés KMIP en fonction de leur état d'activité. Valeurs possibles : • vrai : renvoie uniquement les serveurs de clés KMIP actifs (fournissant les clés actuellement utilisées). • false : Renvoie uniquement les serveurs de clés KMIP inactifs (ne fournissant aucune clé et pouvant être supprimés). Si cette option est omise, les serveurs de clés KMIP renvoyés ne sont pas filtrés en fonction de leur état (actif ou non).	booléen	Aucune	Non

Nom	Description	Type	Valeur par défaut	Obligatoire
kmipHasProviderAssigned	Les filtres ont renvoyé les serveurs de clés KMIP en fonction de la présence ou non d'un fournisseur de clés KMIP attribué. Valeurs possibles : • vrai : renvoie uniquement les serveurs de clés KMIP auxquels un fournisseur de clés KMIP est attribué. • false : Renvoie uniquement les serveurs de clés KMIP auxquels aucun fournisseur de clés KMIP n'est attribué. Si cette option est omise, les serveurs de clés KMIP renvoyés ne sont pas filtrés en fonction de la présence ou non d'un fournisseur de clés KMIP attribué.	booléen	Aucune	Non

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type
Serveurs de clés kmip	Liste complète des serveurs de clés KMIP créés.	"KeyServerKmip"tableau

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

ModifierKeyServerKmip

Vous pouvez utiliser le `ModifyKeyServerKmip` méthode permettant de modifier un serveur de clés KMIP (Key Management Interoperability Protocol) existant selon les attributs spécifiés. Bien que le seul paramètre requis soit le keyServerID, une requête ne contenant que le keyServerID ne fera rien et ne renverra aucune erreur. Tout autre paramètre que vous spécifiez remplacera les valeurs existantes pour le serveur de clés par l'identifiant keyServerID spécifié. Le serveur clé est contacté pendant l'opération afin de s'assurer de son bon fonctionnement. Vous pouvez fournir plusieurs noms d'hôtes ou adresses IP avec le paramètre kmipKeyServerHostnames, mais uniquement si les serveurs de clés sont configurés en cluster.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du serveur de clés	L'identifiant du serveur de clés KMIP à modifier.	entier	Aucune	Oui
kmipCaCertificate	Le certificat de clé publique de l'autorité de certification racine du serveur de clés externe. Cela servira à vérifier le certificat présenté par le serveur de clés externe dans la communication TLS. Pour les clusters de serveurs critiques où chaque serveur utilise une autorité de certification différente, fournissez une chaîne concaténée contenant les certificats racine de toutes les autorités de certification.	chaîne	Aucune	Non
Certificat client kmip	Un certificat X.509 PKCS#10 au format PEM encodé en Base64 utilisé par le client Solidfire KMIP.	chaîne	Aucune	Non
Noms d'hôte du serveur de clés kmip	Tableau des noms d'hôtes ou des adresses IP associés à ce serveur de clés KMIP. Plusieurs noms d'hôtes ou adresses IP ne doivent être fournis que si les serveurs clés sont configurés en cluster.	tableau de chaînes	Aucune	Non

Nom du serveur de clés kmip	Le nom du serveur de clés KMIP. Ce nom est utilisé uniquement à des fins d'affichage et n'a pas besoin d'être unique.	chaîne	Aucune	Non
kmipKeyServerPort	Le numéro de port associé à ce serveur de clés KMIP (généralement 5696).	entier	Aucune	Non

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type
kmipKeyServer	Un objet contenant des informations sur le serveur de clés nouvellement modifié.	"KeyServerKmip"

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

```
{
   "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
        }
     }
}
```

Nouveautés depuis la version

11,7

RekeySoftwareEncryptionAtRestMasterKey

Vous pouvez utiliser le `RekeySoftwareEncryptionAtRestMasterKey` méthode de renouvellement de la clé principale de chiffrement au repos du logiciel utilisée pour chiffrer les DEK (clés de chiffrement de données). Lors de la création du cluster, le chiffrement logiciel au repos est configuré pour utiliser la gestion des clés internes (IKM). Cette méthode de renouvellement de clé peut être utilisée après la création du cluster pour utiliser soit IKM, soit la gestion de clés externes (EKM).

Paramètres

Cette méthode possède les paramètres d'entrée suivants. Si le `keyManagementType` Si aucun paramètre n'est spécifié, l'opération de renouvellement de clé est effectuée à l'aide de la configuration de gestion des clés existante. Si le `keyManagementType` est spécifié et le fournisseur de clés est externe, le `keyProviderID` Ce paramètre doit également être utilisé.

Paramètre	Description	Type	Facultatif
Type de gestion des clés	Le type de gestion des clés utilisé pour gérer la clé principale. Les valeurs possibles sont : `Internal` : Renouveler la clé à l'aide du système de gestion des clés interne. `External` : Renouveler la clé à l'aide d'un système de gestion de clés externe. Si ce paramètre n'est pas spécifié, l'opération de renouvellement de clé est effectuée en utilisant la configuration de gestion des clés existante.	chaîne	Vrai
ID du fournisseur de clés	L'identifiant du fournisseur de clés à utiliser. Il s'agit d'une valeur unique renvoyée dans le cadre de l'une des `CreateKeyProvider` méthodes. La pièce d'identité n'est requise que lorsque `keyManagementType` est `External` et est par ailleurs invalide.	entier	Vrai

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Paramètre	Description	Type	Facultatif
asyncHandle	Déterminez l'état de l'opération de renouvellement de clé à l'aide de ceci `asyncHandle` valeur avec `GetAsyncResult` . `GetAsyncResult` Le résultat comprendra un `newKey` champ contenant des informations sur la nouvelle clé principale et un `keyToDecommission` champ contenant des informations sur l'ancienne clé.	entier	FAUX

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

12,3

Trouver plus d'informations

- "Documentation logicielle SolidFire et Element"
- "Documentation relative aux versions antérieures des produits NetApp SolidFire et Element"

SupprimerKeyServerFromProviderKmip

Vous pouvez utiliser le `RemoveKeyServerFromProviderKmip` méthode permettant de désaffecter le serveur de clés KMIP (Key Management Interoperability Protocol) spécifié du fournisseur auquel il a été affecté. Vous pouvez désassocier un serveur de clés de son fournisseur, sauf s'il s'agit du dernier et que son fournisseur est actif (fournissant des clés actuellement utilisées). Si le serveur de clés spécifié n'est pas attribué à un fournisseur, aucune action n'est entreprise et aucune erreur n'est renvoyée.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du serveur de clés	L'identifiant du serveur de clés KMIP à désattribuer.	entier	Aucune	Oui

Valeurs de retour

Cette méthode ne renvoie aucune valeur. La suppression est considérée comme réussie tant qu'aucune erreur n'est renvoyée.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

SignSshKeys

Une fois SSH activé sur le cluster à l'aide de"Méthode EnableSSH" , vous pouvez utiliser le `SignSshKeys` méthode pour accéder à un shell sur un nœud.

À partir de l'élément 12.5, `sfreadonly` Il s'agit d'un nouveau compte système qui permet un dépannage de base sur un nœud. Cette API permet l'accès SSH via `sfreadonly` Compte système sur tous les nœuds du cluster.

Sauf avis contraire du support NetApp , toute modification du système n'est pas prise en charge, annule votre contrat de support et peut entraîner une instabilité ou une inaccessibilité des données.

Après avoir utilisé cette méthode, vous devez copier le trousseau de clés à partir de la réponse, l'enregistrer sur le système qui initiera la connexion SSH, puis exécuter la commande suivante :

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

```
`identity_file`est un fichier à partir duquel l'identité (clé privée) pour l'authentification par clé publique est lue et `node_ip` est l'adresse IP du nœud.  Pour plus d'informations sur `identity_file` , voir la page de manuel SSH.
```

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
durée	Entier compris entre 1 et 24 indiquant le nombre d'heures de validité de la clé signée. Si la durée n'est pas spécifiée, la valeur par défaut est utilisée.	entier	1	Non

Nom	Description	Type	Valeur par défaut	Obligatoire
clé publique	Si ce paramètre est fourni, il ne renverra que la clé publique signée au lieu de créer un trousseau complet pour l'utilisateur.  Clés publiques soumises via la barre d'adresse d'un navigateur avec + sont interprétés comme des signes espacés et des signes de rupture.	chaîne	Nul	Non
sfadmin	Permet d'accéder au compte shell sfadmin lorsque vous effectuez l'appel API avec un accès au cluster supportAdmin, ou lorsque le nœud n'est pas dans un cluster.	booléen	FAUX	Non

Valeurs de retour

Cette méthode renvoie les valeurs suivantes :

Nom	Description	Type
statut de génération de clés	Contient l'identité de la clé signée, les principaux autorisés et les dates de début et de fin de validité de la clé.	chaîne
clé privée	La valeur d'une clé SSH privée n'est renvoyée que si l'API génère un trousseau complet pour l'utilisateur final. La valeur est encodée en Base64 ; vous devez la décoder lorsqu'elle est écrite dans un fichier pour vous assurer qu'elle est lue comme une clé privée valide.	chaîne
clé publique	Une valeur de clé SSH publique n'est renvoyée que si l'API génère un trousseau complet pour l'utilisateur final. Lorsque vous transmettez un paramètre public_key à la méthode API, seule la `signed_public_key` La valeur est renvoyée dans la réponse.	chaîne
clé publique signée	La clé publique SSH résultant de la signature de la clé publique, qu'elle ait été fournie par l'utilisateur ou générée par l'API.	chaîne

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Dans cet exemple, une clé publique est signée et renvoyée, valable pour une durée de 1 à 24 heures.

Nouveautés depuis la version

12,5

TestKeyProviderKmip

Vous pouvez utiliser le `TestKeyProviderKmip` méthode permettant de tester si le fournisseur de clés KMIP (Key Management Interoperability Protocol) spécifié est joignable et fonctionne normalement.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du fournisseur de clés	L'identifiant du fournisseur de clés à tester.	entier	Aucune	Oui

Valeurs de retour

Cette méthode ne renvoie aucune valeur. Le test est considéré comme réussi tant qu'aucune erreur n'est renvoyée.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7

TestKeyServerKmip

Vous pouvez utiliser le `TestKeyServerKmip` méthode permettant de tester si le serveur de clés KMIP (Key Management Interoperability Protocol) spécifié est accessible et fonctionne normalement.

Paramètres

Cette méthode possède les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
ID du serveur de clés	L'identifiant du serveur de clés KMIP à tester.	entier	Aucune	Oui

Valeurs de retour

Cette méthode ne renvoie aucune valeur. Le test est considéré comme réussi si aucune erreur n'est renvoyée.

Exemple de demande

Les requêtes pour cette méthode sont similaires à l'exemple suivant :

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

Exemple de réponse

Cette méthode renvoie une réponse similaire à l'exemple suivant :

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

Nouveautés depuis la version

11,7