Méthodes API de sécurité

Element Software

NetApp
October 01, 2024

Sommaire

Méthodes API de sécurité

Vous pouvez intégrer le logiciel Element avec des services de sécurité externes, comme un serveur de gestion externe des clés. Ces méthodes de sécurité vous permettent de configurer les fonctionnalités de sécurité d'Element, telles que la gestion externe des clés pour le chiffrement des données au repos.

- AddKeyServerToProviderKmip
- CreateKeyProviderKmip
- CreateKeyServerKmip
- CreatePublicPrivateKeypair
- DeleteKeyProviderKmip
- DeleteKeyServerKmip
- DisableEncryptionAtRest
- EnableEncryptionAtRest
- GetClientCertificateSignRequest
- GetKeyProviderKmip
- GetKeyServerKmip
- ListeKeyProvidersKmip
- ListKeyServoersKmip
- ModityKeyServerKmip
- RemoveKeyServerFromProviderKmip
- SignSshKeys
- TestKeyProviderKmip
- TestKeyServerKmip

Trouvez plus d'informations

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

AddKeyServerToProviderKmip

Vous pouvez utiliser la `AddKeyServerToProviderKmip` méthode pour attribuer un serveur de clés KMIP (Key Management Interoperability Protocol) au fournisseur de clés spécifié. Lors de l'affectation, le serveur est contacté pour vérifier la fonctionnalité. Si le serveur de clés spécifié est déjà affecté au fournisseur de clés spécifié, aucune action n'est effectuée et aucune erreur n'est renvoyée. Vous pouvez supprimer l'affectation à l'aide de la `RemoveKeyServerFromProviderKmip` méthode.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyProviderID	ID du fournisseur de clés auquel affecter le serveur de clés.	entier	Aucune	Oui
KeyServerID	L'ID du serveur de clés à affecter.	entier	Aucune	Oui

Valeurs de retour

Cette méthode n'a pas de valeur de retour. L'affectation est considérée comme réussie tant qu'aucune erreur n'est renvoyée.

Exemple de demande

Les demandes 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":
      {}
    }
}
```

Nouveau depuis la version

11,7

CreateKeyProviderKmip

Vous pouvez utiliser cette `CreateKeyProviderKmip` méthode pour créer un fournisseur de clés KMIP (Key Management Interoperability Protocol) portant 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 n'est affecté. Pour créer un serveur de clés KMIP, utilisez la `CreateKeyServerKmip` méthode. Pour l'attribuer à un fournisseur, reportez-vousà la section `AddKeyServerToProviderKmip`.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyProviderName	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 a les valeurs de retour suivantes :

Nom	Description	Type
KmipKeyProvider	Objet contenant des détails sur le fournisseur de clés nouvellement créé.	"KeyProviderKmip"

Exemple de demande

Les demandes 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
    }
}
}
```

Nouveau depuis la version

11,7

CreateKeyServerKmip

Vous pouvez utiliser la `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'a pas besoin d'exister avant d'utiliser cette méthode. Pour les configurations de serveur à clé en cluster, vous devez fournir les noms d'hôte ou les adresses IP de tous les nœuds de serveur dans le paramètre kmipKeyServerHostnames. Vous pouvez utiliser la `TestKeyServerKmip` méthode pour tester un serveur de clés.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KmipCaCertificate	Certificat de clé publique de l'autorité de certification racine du serveur de clés externe. Cette option permet de vérifier le certificat présenté par le serveur de clés externe dans la communication TLS. Pour les grappes de serveurs de clés où des serveurs individuels utilisent des autorités de certification différentes, fournissez une chaîne concaténée contenant les certificats racine de toutes les autorités de certification.	chaîne	Aucune	Oui
KmipClientCertificate	Certificat PKCS#10 X.509 codé au format PEM utilisé par le client SolidFire KMIP.	chaîne	Aucune	Oui
Noms d'hôte kmipKeyServerHostNames	Tableau des noms d'hôte ou adresses IP associés à ce serveur de clés KMIP. Plusieurs noms d'hôte ou adresses IP ne doivent être fournis que si les serveurs clés sont dans une configuration en cluster.	tableau de chaînes	Aucune	Oui

Nom	Description	Type	Valeur par défaut	Obligatoire
KmipKeyServerName	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	Numéro de port associé à ce serveur de clés KMIP (généralement 5696).	entier	Aucune	Non

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Nom	Description	Type
KmipKeyServer	Objet contenant des détails sur le nouveau serveur de clés créé.	"KeyServerKmip"

Exemple de demande

Les demandes 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
        }
     }
}
```

Nouveau depuis la version

11,7

CreatePublicPrivateKeypair

Vous pouvez utiliser la `CreatePublicPrivateKeyPair` méthode pour créer des clés SSL publiques et privées. Vous pouvez utiliser ces clés pour générer des demandes de signature de certificat. Seule une paire de clés peut être utilisée pour chaque cluster de stockage. Avant d'utiliser cette méthode pour remplacer les clés existantes, assurez-vous que les clés ne sont plus utilisées par aucun fournisseur.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
Nom CommonName	Le champ Nom distinctif X.509 **Nom commun** (CN).	chaîne	Aucune	Non
pays	Le champ Nom distinctif X.509 **pays** ©.	chaîne	Aucune	Non

Nom	Description	Type	Valeur par défaut	Obligatoire
Adresse électronique	Le champ Nom distinctif X.509 **adresse électronique** (MAIL).	chaîne	Aucune	Non
localité	Le champ L (Nom distinctif) de X.509 **Nom de localité**.	chaîne	Aucune	Non
entreprise	Le champ Nom distinctif X.509 **Nom de l'organisation** (O).	chaîne	Aucune	Non
Unité organisationnelle	Le champ Nom distinctif X.509 **Nom de l'unité organisationnelle** (ou).	chaîne	Aucune	Non
état	Le champ Nom distinctif X.509 **État** ou **Nom de province** (ST ou SP ou S).	chaîne	Aucune	Non

Valeurs de retour

Cette méthode n'a pas de valeurs de retour. En l'absence d'erreur, la création de clé est considérée comme réussie.

Exemple de demande

Les demandes 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":
      {}
}
```

Nouveau depuis la version

11,7

DeleteKeyProviderKmip

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

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

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

Valeurs de retour

Cette méthode n'a pas de valeurs de retour. L'opération de suppression est considérée comme réussie car il n'y a pas d'erreur.

Exemple de demande

Les demandes 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":
      {}
}
```

Nouveau depuis la version

11,7

DeleteKeyServerKmip

Vous pouvez utiliser cette `DeleteKeyServerKmip` méthode pour supprimer un serveur de clés KMIP (Key Management Interoperability Protocol) existant. Vous pouvez supprimer un serveur de clés à moins qu'il ne soit le dernier affecté à son fournisseur et que ce fournisseur fournit des clés actuellement utilisées.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

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

Valeurs de retour

Cette méthode a les valeurs no return. L'opération de suppression est considérée comme réussie en l'absence d'erreur.

Exemple de demande

Les demandes 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":
      {}
}
```

Nouveau depuis la version

11,7

DisableEncryptionAtRest

Vous pouvez utiliser la `DisableEncryptionAtRest` méthode pour supprimer le chiffrement précédemment appliqué au cluster à l'aide de la `EnableEncryptionAtRest` méthode. Cette méthode de désactivation est asynchrone et renvoie une réponse avant que le cryptage ne soit désactivé. Vous pouvez utiliser la `GetClusterInfo` méthode pour interroger le système pour voir quand le processus est terminé.

Pour afficher l'état actuel du chiffrement au repos et/ou du chiffrement logiciel au repos sur le cluster, utilisez le "obtenir la méthode d'information sur le cluster". Vous pouvez utiliser le `GetSoftwareEncryptionAtRestInfo` "méthode d'obtention des informations que le cluster utilise pour chiffrer les données au repos".

Vous ne pouvez pas utiliser cette méthode pour désactiver le chiffrement logiciel au repos. Pour désactiver le chiffrement logiciel au repos, vous devez "créez un nouveau cluster"activer le chiffrement logiciel au repos désactivé.

Paramètres

Cette méthode n'a pas de paramètres d'entrée.

Valeurs de retour

Cette méthode n'a pas de valeurs de retour.

Exemple de demande

Les demandes 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" : {}
}
```

Nouveau depuis la version

9,6

Trouvez plus d'informations

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

EnableEncryptionAtRest

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

Pour afficher l'état actuel du chiffrement au repos et/ou du chiffrement logiciel au repos sur le cluster, utilisez le "obtenir la méthode d'information sur le cluster". Vous pouvez utiliser le `GetSoftwareEncryptionAtRestInfo` "méthode d'obtention des informations que le cluster utilise pour chiffrer les données au repos".

Cette méthode n'active pas le chiffrement logiciel au repos. Cette opération ne peut être effectuée qu'à l'aide "créer une méthode de cluster"du avec `enableSoftwareEncryptionAtRest` défini sur `true`.

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

Si un keyProviderID est spécifié, le mot de passe est généré et récupéré selon le type de fournisseur de clés. Cette opération est généralement effectuée 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 pas être supprimé tant que le chiffrement au repos n'est pas désactivé à l'aide de la `DisableEncryptionAtRest` méthode.

Si vous avez un type de nœud avec un numéro de modèle se terminant par "-ne", l' `EnableEncryptionAtRest`appel de méthode échouera avec une réponse de "chiffrement non autorisé. Cluster détecté noeud non encryptable ».

Vous devez activer ou désactiver le chiffrement uniquement lorsque le cluster est en cours d'exécution et qu'il est en état de santé. Vous pouvez activer ou désactiver le cryptage à votre discrétion et aussi souvent que vous le souhaitez.

Ce processus est asynchrone et renvoie une réponse avant l'activation du chiffrement. Vous pouvez utiliser la `GetClusterInfo` méthode pour interroger le système pour voir quand le processus est terminé.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyProviderID	ID d'un fournisseur de clés KMIP à utiliser.	entier	Aucune	Non

Valeurs de retour

Cette méthode n'a pas de valeurs de retour.

Exemple de demande

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

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

Exemples de réponse

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

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

Bien que le chiffrement au repos soit activé sur un cluster, GetClusterInfo renvoie un résultat décrivant l'état du chiffrement au repos (« EncryptionAtRestState ») comme « activation ». Une fois le chiffrement au repos entièrement activé, l'état renvoyé est « 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"
      }
   }
}
```

Nouveau depuis la version

9,6

Trouvez plus d'informations

- "SecureEraseDrives"
- "GetClusterInfo"
- "Documentation SolidFire et Element"

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

GetClientCertificateSignRequest

Vous pouvez utiliser la `GetClientCertificateSignRequest` méthode pour générer une demande de signature de certificat qui peut être signée par une autorité de certification afin de générer un certificat client pour le cluster. Les certificats signés sont nécessaires pour établir une relation de confiance pour interagir avec les services externes.

Paramètres

Cette méthode n'a pas de paramètres d'entrée.

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

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

Exemple de demande

Les demandes 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..."
   }
}
```

Nouveau depuis la version

11,7

GetKeyProviderKmip

Vous pouvez utiliser la `GetKeyProviderKmip` méthode pour récupérer des informations sur le fournisseur de clés KMIP (Key Management Interoperability Protocol) spécifié.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

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

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Nom	Description	Type
KmipKeyProvider	Objet contenant des détails sur le fournisseur de clés demandé.	"KeyProviderKmip"

Exemple de demande

Les demandes 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"
    }
   }
}
```

Nouveau depuis la version

11,7

GetKeyServerKmip

Vous pouvez utiliser la `GetKeyServerKmip` méthode pour renvoyer des informations relatives au serveur de clés KMIP (Key Management Interoperability Protocol) spécifié.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyServerID	ID du serveur de clés KMIP pour renvoyer des informations sur.	entier	Aucune	Oui

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Nom	Description	Type
KmipKeyServer	Objet contenant des détails sur le serveur de clés demandé.	"KeyServerKmip"

Exemple de demande

Les demandes 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
        }
     }
}
```


Nouveau depuis la version

11,7

GetSoftwareEncryptionAtRestInfo

Vous pouvez utiliser cette `GetSoftwareEncryptionAtRestInfo` méthode pour obtenir les informations de chiffrement logiciel au repos que le cluster utilise pour chiffrer les données au repos.

Paramètres

Cette méthode n'a pas de paramètres d'entrée.

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Paramètre	Description	Type	Facultatif
Master KeyInfo	Informations sur la clé principale de chiffrement logiciel au repos actuelle.	Crypter_KeyInfo	Vrai

Paramètre	Description	Type	Facultatif
RekeyMasterKeyAsyncResultID	ID de résultat asynchrone de l'opération de renouvellement de clés en cours ou la plus récente (le cas échéant), s'il n'a pas encore été supprimé. `GetAsyncResult` le résultat inclut 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 logiciel de chiffrement des données au repos. Les valeurs possibles sont `disabled` ou `enabled`.	chaîne	Faux
version	Un numéro de version incrémenté à chaque fois que le chiffrement logiciel au repos est activé.	entier	Faux

Exemple de demande

Les demandes 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
    }
}
```


Nouveau depuis la version

12,3

Trouvez plus d'informations

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

ListeKeyProvidersKmip

Vous pouvez utiliser cette `ListKeyProvidersKmip` méthode pour obtenir la liste de tous les fournisseurs de clés KMIP (Key Management Interoperability Protocol) existants. Vous pouvez filtrer la liste en spécifiant des paramètres supplémentaires.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyProviderIsActive	Les filtres ont renvoyé des objets de serveur de clés KMIP en fonction de leur état d'activité. Valeurs possibles : • Vrai : seuls les fournisseurs de clés KMIP actifs (avec des clés en cours d'utilisation) sont renvoyées. • Faux : renvoie uniquement les fournisseurs de clés KMIP qui sont inactifs (aucune clé ne peut être supprimée). Si vous ne l'omettez pas, les fournisseurs de clés KMIP renvoyés ne sont pas filtrés en fonction de la condition qu'ils soient actifs.	booléen	Aucune	Non

Nom	Description	Type	Valeur par défaut	Obligatoire
KmipKeyProviderHasServerAssigned	Les filtres ont renvoyé les fournisseurs de clés KMIP en fonction du serveur de clés KMIP qui est attribué ou non. Valeurs possibles : • Vrai : ne renvoie que les fournisseurs de clés KMIP qui disposent d'un serveur de clés KMIP attribué. • FALSE : renvoie uniquement les fournisseurs de clés KMIP qui ne disposent pas d'un serveur de clés KMIP attribué. En cas d'omission, les fournisseurs de clés KMIP renvoyés ne sont pas filtrés selon que le serveur de clés KMIP est attribué ou non.	booléen	Aucune	Non

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Nom	Description	Type
KmipKeyProviders	Liste de fournisseurs de clés KMIP créés.	"KeyProviderKmip" baie

Exemple de demande

Les demandes 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"
        }
    ]
}
}
```

Nouveau depuis la version

11,7

ListKeyServoersKmip

Vous pouvez utiliser cette `ListKeyServersKmip` méthode pour répertorier 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 présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyProviderID	Lorsqu'elle est spécifiée, cette méthode renvoie uniquement les serveurs de clés KMIP qui sont affectés au fournisseur de clés KMIP spécifié. Si vous omettez le paramètre, les serveurs de clés KMIP renvoyés ne sont pas filtrés selon que ils sont attribués au fournisseur de clés KMIP spécifié ou non.	entier	Aucune	Non
KmipAssignedProviderIsActive	Les filtres ont renvoyé des objets de serveur de clés KMIP en fonction de leur état d'activité. Valeurs possibles : • Vrai : seuls les serveurs de clés KMIP qui sont actifs (avec des clés en cours d'utilisation) sont renvoyées. • Faux : renvoie uniquement les serveurs de clés KMIP qui sont inactifs (ne fournissant aucune clé et ne pouvant être supprimés). Si vous omettez le paramètre, les serveurs de clés KMIP renvoyés ne sont pas filtrés en fonction de la condition qu'ils soient actifs.	booléen	Aucune	Non

Nom	Description	Type	Valeur par défaut	Obligatoire
KmipHasProviderAssigned	Les filtres ont renvoyé des serveurs de clés KMIP en fonction du fournisseur de clés KMIP qui leur est attribué ou non. Valeurs possibles : • Vrai : seuls les serveurs de clés KMIP qui sont affectés par un fournisseur de clés KMIP. • FALSE : renvoie uniquement les serveurs de clés KMIP qui ne disposent pas d'un fournisseur de clés KMIP attribué. En cas d'omission, les serveurs de clés KMIP renvoyés ne sont pas filtrés selon que le fournisseur de clés KMIP est attribué ou non.	booléen	Aucune	Non

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Nom	Description	Type
KmipKeyServers	La liste complète des serveurs de clés KMIP qui ont été créés.	"KeyServerKmip" baie

Exemple de demande

Les demandes 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
        }
    ]
}
```

Nouveau depuis la version

11,7

ModityKeyServerKmip

Vous pouvez utiliser la `ModifyKeyServerKmip` méthode pour modifier un serveur de clés KMIP (Key Management Interoperability Protocol) existant en fonction des attributs spécifiés. Bien que le seul paramètre requis soit le keyServerID, une requête contenant uniquement le keyServerID ne prend aucune action et ne renvoie aucune erreur. Tous les autres paramètres que vous spécifiez remplaceront les valeurs existantes pour le serveur de clés par l'ID keyServerID spécifié. Le serveur principal est contacté au cours de l'opération pour s'assurer qu'il est fonctionnel. Vous pouvez fournir plusieurs noms d'hôte ou adresses IP avec le paramètre kmipKeyServerHostnames, mais uniquement si les serveurs de clés sont dans une configuration en cluster.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyServerID	ID du serveur de clés KMIP à modifier.	entier	Aucune	Oui
KmipCaCertificate	Certificat de clé publique de l'autorité de certification racine du serveur de clés externe. Cette option permet de vérifier le certificat présenté par le serveur de clés externe dans la communication TLS. Pour les grappes de serveurs de clés où des serveurs individuels utilisent des autorités de certification différentes, fournissez une chaîne concaténée contenant les certificats racine de toutes les autorités de certification.	chaîne	Aucune	Non
KmipClientCertificate	Certificat PKCS#10 X.509 codé au format PEM utilisé par le client SolidFire KMIP.	chaîne	Aucune	Non

Noms d'hôte kmipKeyServerHostNames	Tableau des noms d'hôte ou adresses IP associés à ce serveur de clés KMIP. Plusieurs noms d'hôte ou adresses IP ne doivent être fournis que si les serveurs clés sont dans une configuration en cluster.	tableau de chaînes	Aucune	Non
KmipKeyServerName	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	Numéro de port associé à ce serveur de clés KMIP (généralement 5696).	entier	Aucune	Non

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Nom	Description	Type
KmipKeyServer	Objet contenant des détails sur le serveur de clés nouvellement modifié.	"KeyServerKmip"

Exemple de demande

Les demandes 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
        }
    }
}
```

Nouveau depuis la version

11,7

RekeySoftwareEncryptionAtResteMasterKey

Vous pouvez utiliser `RekeySoftwareEncryptionAtRestMasterKey` la méthode pour

redéfinir la clé principale de chiffrement logiciel au repos utilisée pour chiffrer les clés DEK (Data Encryption Keys). Lors de la création du cluster, le chiffrement logiciel au repos est configuré pour utiliser la gestion interne des clés (IKM). Cette méthode de renouvellement peut être utilisée après la création du cluster pour utiliser IKM ou External Key Management (EKM).

Paramètres

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

Paramètre	Description	Type	Facultatif
Type de gestion de clés	Type de gestion des clés utilisé pour gérer la clé principale. Les valeurs possibles sont : `Internal`: Rekey using Internal Key Management. `External`: Rekey à l'aide de la gestion externe des clés. Si ce paramètre n'est pas spécifié, l'opération de renouvellement de clés est effectuée à l'aide de la configuration existante de gestion des clés.	chaîne	Vrai
KeyProviderID	ID du fournisseur de clés à utiliser. Il s'agit d'une valeur unique renvoyée dans le cadre de l'une des `CreateKeyProvider` méthodes. L'ID n'est requis que si `keyManagementType` est `External` et est autrement invalide.	entier	Vrai

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Paramètre	Description	Type	Facultatif
Asynchrone	Déterminez l'état de l'opération de renouvellement de clé à l'aide de cette `asyncHandle` valeur avec `GetAsyncResult`. `GetAsyncResult` le résultat inclut 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 demandes 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
}
```

Nouveau depuis la version

12,3

Trouvez plus d'informations

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

RemoveKeyServerFromProviderKmip

Vous pouvez utiliser la `RemoveKeyServerFromProviderKmip` méthode pour annuler l'affectation du serveur de clés KMIP (Key Management Interoperability Protocol) spécifié au fournisseur auquel il a été attribué. Vous pouvez annuler l'affectation d'un serveur de clés de son fournisseur, sauf s'il s'agit du dernier serveur et de son fournisseur actif (fournissant les clés actuellement utilisées). Si le serveur de clés spécifié n'est pas affecté à un fournisseur, aucune action n'est effectuée et aucune erreur n'est renvoyée.

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyServerID	ID du serveur de clés KMIP à désaffecter.	entier	Aucune	Oui

Valeurs de retour

Cette méthode n'a pas de valeurs de retour. La suppression est considérée comme réussie tant qu'aucune erreur n'est renvoyée.

Exemple de demande

Les demandes 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":
      {}
    }
}
```

Nouveau depuis la version

11,7

SignSshKeys

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

Depuis Element 12.5, `sfreadonly` est un nouveau compte système qui permet un dépannage de base sur un nœud. Cette API active l'accès SSH en utilisant le `sfreadonly` compte système sur tous les nœuds du cluster.

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

Après avoir utilisé la méthode, vous devez copier le trousseau 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) de
l'authentification par clé publique est lue et `node_ip` qui correspond à
l'adresse IP du nœud. Pour plus d'informations sur `identity_file`,
consultez la page de manuel SSH.
```

Paramètres

Cette méthode présente 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
Publickey	S'il est fourni, ce paramètre renvoie uniquement la clé_publique_signée au lieu de créer un trousseau complet pour l'utilisateur. Les clés publiques soumises à l'aide de la barre d'URL dans un navigateur avec + sont interprétées comme étant espacées et rompez la signature.	chaîne	Nul	Non

Nom	Description	Type	Valeur par défaut	Obligatoire
sfadmin	Permet d'accéder au compte sfadmin shell lorsque vous passez l'appel API avec l'accès au cluster supportAdmin ou lorsque le nœud ne se trouve pas dans un cluster.	booléen	Faux	Non

Valeurs de retour

Cette méthode a les valeurs de retour suivantes :

Nom	Description	Type
keygen_status	Contient l'identité dans la clé signée, les entités autorisées et les dates de début et de fin valides pour la clé.	chaîne
clé_privée	Une valeur de 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 décoder la valeur lorsqu'elle est écrite dans un fichier pour vous assurer qu'elle est lue comme une clé privée valide.	chaîne

Nom	Description	Type
touche_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` valeur est renvoyée dans la réponse.	chaîne
clé_publique_signée	Clé publique SSH résultant de la signature de la clé publique, que l'utilisateur ait fourni ou généré par l'API.	chaîne

Exemple de demande

Les demandes 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 et valide pour la durée (1-24 heures).

Nouveau depuis la version

12,5

TestKeyProviderKmip

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

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

Nom	Description	Type	Valeur par défaut	Obligatoire
KeyProviderID	ID du fournisseur de clé à tester.	entier	Aucune	Oui

Valeurs de retour

Cette méthode n'a pas de valeurs de retour. Le test est considéré comme réussi tant qu'aucune erreur n'est renvoyée.

Exemple de demande

Les demandes 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":
      {}
    }
}
```

Nouveau depuis la version

11,7

TestKeyServerKmip

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

Paramètres

Cette méthode présente les paramètres d'entrée suivants :

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

Valeurs de retour

Cette méthode n'a pas de valeurs de retour. Le test est considéré comme réussi s'il n'y a pas d'erreur renvoyée.

Exemple de demande

Les demandes 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":
      {}
    }
```

Nouveau depuis la version

11,7