

Guide de programmation au moyen d'API

12 MAI 2021

vRealize Operations 8.3

Vous trouverez la documentation technique la plus récente sur le site Web de VMware, à l'adresse :

<https://docs.vmware.com/fr/>

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

VMware France SAS.
Tour Franklin
100-101 Terrasse Boieldieu
92042 Paris La Défense 8 Cedex
France
www.vmware.com/fr

Copyright © 2021 VMware, Inc. Tous droits réservés. [Informations relatives aux copyrights et marques commerciales.](#)

Table des matières

1	À propos de ce guide de programmation	4
2	Compréhension de l'API vRealize Operations Manager	5
	Fonctionnement de l'API vRealize Operations Manager	5
	Vue d'ensemble du workflow client	7
	Demandes d'API REST vRealize Operations Manager	8
	Réponses d'API REST vRealize Operations Manager	10
	Utilisation de l'API avec vRealize Operations Manager	12
3	Démarrage avec l'API	14
	Obtenir un jeton d'authentification	14
	Rechercher le type d'adaptateur et le type d'objet	16
	Générer une liste de toutes les mesures pour l'objet	18
4	Configuration d'une instance de l'adaptateur	20
	Résumé de la configuration des demandes d'instance de l'adaptateur	20
	Identifier la solution et ses adaptateurs	21
	Identifier les types d'objet requis pour l'adaptateur	23
	Créer l'instance de l'adaptateur	24
	Fournir la preuve de la validité du certificat	28
	Commencer la surveillance de la nouvelle instance de l'adaptateur	32

À propos de ce guide de programmation

1

Le Guide de programmation au moyen d'API de vRealize Operations Manager fournit des informations sur les REST API de vRealize Operations Manager, et notamment sur l'utilisation des ressources REST API, l'authentification et la construction d'appels REST API.

Public cible

Ces informations sont destinées aux administrateurs et aux programmeurs qui souhaitent configurer et gérer vRealize Operations Manager par programmation à l'aide de REST API de vRealize Operations Manager. Ce guide traite des cas d'utilisation courants.

Compréhension de l'API vRealize Operations Manager

2

Les développeurs peuvent utiliser l'API pour créer des clients interactifs de vRealize Operations Manager. L'API suit le style REST et est disponible pour tous les utilisateurs disposant d'une licence.

Les clients vRealize Operations Manager communiquent avec le serveur via HTTP et échangent des représentations d'objets vRealize Operations Manager. Ces représentations prennent la forme d'éléments JSON ou XML. Vous utilisez des demandes HTTP GET pour récupérer la représentation actuelle d'un objet, des demandes HTTP POST et PUT pour créer ou modifier un objet, et des demandes HTTP DELETE pour supprimer un objet.

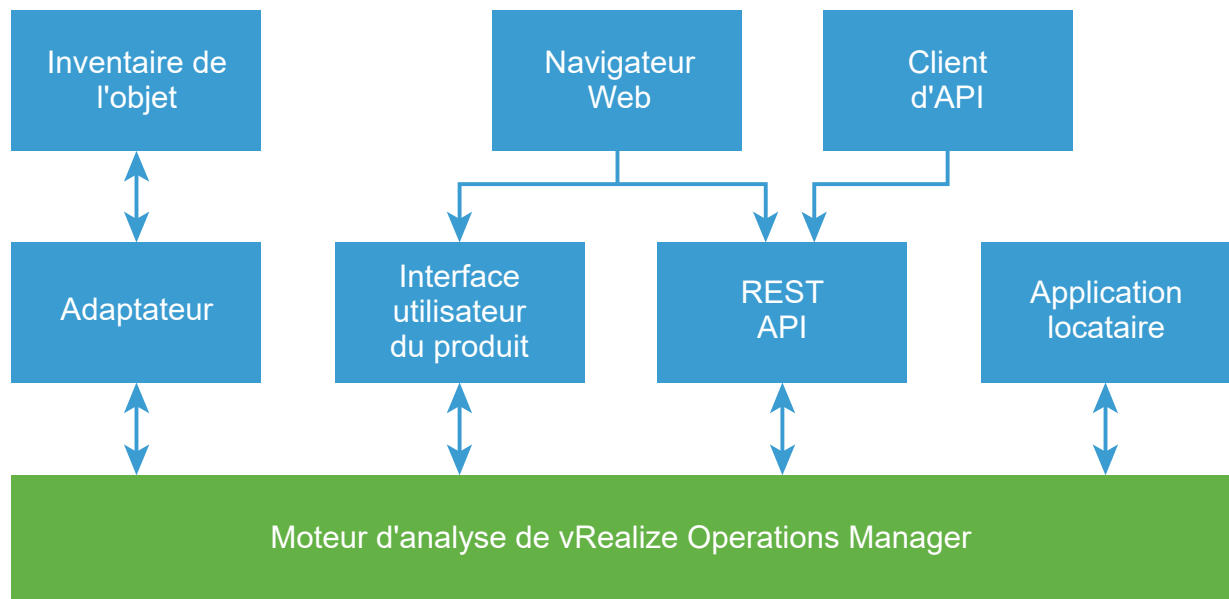
Ce chapitre contient les rubriques suivantes :

- [Fonctionnement de l'API vRealize Operations Manager](#)
- [Vue d'ensemble du workflow client](#)
- [Utilisation de l'API avec vRealize Operations Manager](#)

Fonctionnement de l'API vRealize Operations Manager

Vous utilisez un navigateur Web pour communiquer avec le moteur d'analyse vRealize Operations Manager, via l'interface utilisateur du produit ou via des appels d'API.

Figure 2-1. Architecture simplifiée de vRealize Operations Manager



L'instance de l'adaptateur collecte les données des objets présents dans votre environnement surveillé. Le moteur d'analyse de vRealize Operations Manager traite les données et affiche le modèle complet dans l'interface graphique.

Pourquoi utiliser l'API

L'API est particulièrement utile lorsque vous souhaitez automatiser un workflow bien défini, comme répéter les mêmes tâches pour configurer le contrôle d'accès des nouveaux utilisateurs de vRealize Operations Manager. L'API est également utile lors de l'exécution de requêtes sur le référentiel de données de vRealize Operations Manager, telles que la récupération de données pour certaines ressources de votre environnement virtuel. En outre, vous pouvez utiliser l'API pour extraire toutes les données du référentiel de données de vRealize Operations Manager et les charger dans un système d'analyse distinct.

Terminologie de vRealize Operations Manager

La syntaxe XML que vous utilisez pour décrire les objets d'un adaptateur correspond à la syntaxe du code API, mais elle est différente de celle que vous trouvez dans l'interface utilisateur. Les termes suivants s'affichent dans l'interface utilisateur. La description de chaque terme comprend la syntaxe XML correspondante utilisée dans un appel d'API.

Types d'adaptateur	Définit l'adaptateur utilisé pour détecter certains types d'objet. Par exemple, l'adaptateur vCenter détecte des objets connectés aux centres de données vSphere. L'adaptateur AWS détecte les services et les objets AWS. Syntaxe XML : <code>adapterkinds</code> .
Types d'objets	Classe d'entités qui représentent des objets ou des sources d'informations. Les objets rapportent les données au moteur d'analyse de vRealize Operations Manager. Les machines virtuelles, les banques de données et les systèmes hôtes sont des exemples de types d'objet définis dans un modèle de l'adaptateur vCenter. Syntaxe XML : <code>resourcekinds</code> .

Vue d'ensemble du workflow client

Les clients d'API vRealize Operations Manager mettent en œuvre un workflow REST, en envoyant des demandes HTTP au serveur et en récupérant les informations dont ils ont besoin à partir des réponses du serveur.

À propos de REST

REST, acronyme de Representational State Transfer, décrit une caractéristique de style architectural des programmes qui utilisent le protocole HTTP (Hypertext Transfer Protocol) pour échanger des représentations sérialisées d'objets entre un client et un serveur. Dans l'API vRealize Operations Manager, ces représentations sont des documents JSON ou XML.

Dans un workflow REST, les représentations d'objets sont échangées entre un client et un serveur sur le principe explicite qu'aucune des parties n'a besoin d'en savoir plus sur un objet que ce qui est présenté dans une demande ou une réponse unique. Les URL auxquelles ces documents sont disponibles sont souvent conservées au-delà de la durée de vie de la demande ou de la réponse qui les comprend.

Workflows REST API

Les programmes d'application écrits pour utiliser REST API utilisent des demandes HTTP fréquemment exécutées par un script ou un autre langage de niveau supérieur pour effectuer des appels de procédure distante qui créent, récupèrent, mettent à jour ou suppriment des objets définis par l'API. Dans REST API vRealize Operations Manager, ces objets sont définis par un ensemble de schémas XML. Les opérations elles-mêmes sont des demandes HTTP et sont donc génériques pour tous les clients HTTP.

Pour écrire une application client REST API, vous devez comprendre uniquement le protocole HTTP et la sémantique de JSON ou XML, c'est-à-dire le format de transfert que l'API vRealize Operations Manager utilise. Pour utiliser efficacement l'API dans ce type de client, vous devez vous familiariser avec les concepts suivants.

- L'ensemble des objets pris en charge par l'API et ce qu'ils représentent.
- La façon dont l'API représente ces objets.
- La façon dont un client fait référence à un objet qu'il souhaite utiliser.

La référence d'API comprend une liste complète de demandes d'API. Reportez-vous à [À propos des références de schéma](#).

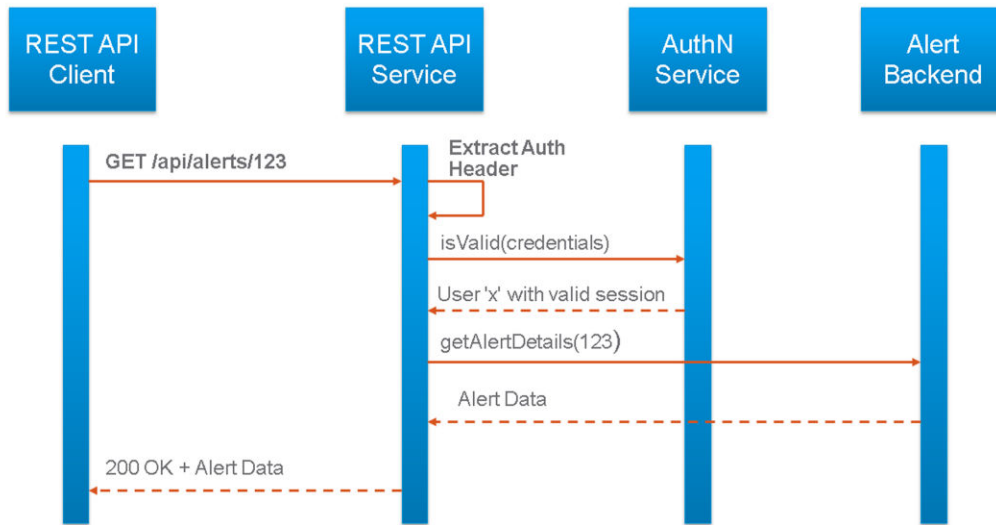
Demandes d'API REST vRealize Operations Manager

Pour récupérer des représentations d'objets, les clients envoient des demandes HTTP aux références d'objets.

Sécurité

Le lien HTTP entre un client et un serveur d'API est établi à l'aide de SSL. Les clients d'API configurent l'authentification basée sur un jeton pour communiquer avec le serveur.

Figure 2-2. Scénario : Fournir des informations d'identification d'utilisateur pour obtenir des détails concernant l'alerte avec l'ID 123



Avec l'authentification basée sur un jeton, vous envoyez une demande POST de connexion au serveur d'API vRealize Operations Manager, en fournissant des informations d'identification d'utilisateur valides pour obtenir un jeton d'authentification. L'exemple suivant présente un scénario d'authentification basée sur un jeton.

- 1 Vous obtenez des informations d'identification d'utilisateur valides pour votre instance de vRealize Operations Manager.
- 2 Envoyez une demande POST au point de terminaison REST pour l'authentification.

```
https://RESTendpoint.example.com/suite-api/api/token/acquire
```

Le corps de la demande comporte le nom d'utilisateur, le mot de passe et la source d'authentification.

- 3 Dans le corps de la réponse, le point de terminaison renvoie le jeton, la date d'expiration et l'heure.
- 4 Pour une communication plus poussée, vous devez inclure l'objet du jeton dans l'en-tête d'autorisation au format suivant :

```
Authorization: vRealizeOpsToken <vROps_token>
```

Si vous avez obtenu le jeton auprès d'une source d'authentification unique, l'en-tête d'autorisation est au format suivant :

```
Authorization: SSO2Token <SSO_SAML_TOKEN>
```

- 5 Vous pouvez invalider le jeton avant la date et l'heure d'expiration en envoyant une demande POST au point de terminaison de déconnexion.

```
POST https://RESTendpoint.example.com/api/token/release
```

En-têtes de la demande

Les en-têtes HTTP suivants sont généralement inclus dans les demandes d'API :

Accept-Language	<p>Pour indiquer la langue souhaitée dans les réponses, utilisez l'en-tête de la demande <code>Accept-Language</code>. Les chaînes de caractères dans les réponses <code>ErrorType</code> sont localisées. Pour demander une réponse avec les chaînes de caractères localisées en français, utilisez l'en-tête suivant :</p> <pre>Accept-Language: fr-FR</pre>
Autorisation	Toutes les demandes de création d'une session d'API doivent inclure un en-tête <code>Authorization</code> du formulaire recommandé par le fournisseur d'identité utilisé par votre organisation
Content-Type	<p>Les demandes qui incluent un corps doivent inclure un en-tête HTTP <code>Content-Type</code> approprié.</p> <ul style="list-style-type: none"> ■ Pour un corps de demande au format XML, l'en-tête doit inclure <code>Content-Type: application/xml</code> ■ Pour un corps de demande au format JSON, l'en-tête doit inclure <code>Content-Type: application/json</code>
Accepter	<p>Pour indiquer le format de réponse souhaité, vous devez inclure l'en-tête de demande <code>Accept</code>.</p> <ul style="list-style-type: none"> ■ Pour une réponse au format XML, l'en-tête doit inclure <code>Accept: application/xml</code> ■ Pour une réponse au format JSON, l'en-tête doit inclure <code>Accept: application/json</code>

Corps de la demande au format XML

Pour un corps de demande écrit en XML, vRealize Operations Manager utilise un analyseur XML de validation qui exige que les éléments du corps de la demande respectent le schéma selon l'ordre et le nombre. Les corps de la demande sont rejetés et non valides, sauf s'ils répondent aux critères suivants :

- Les attributs d'espace de noms XML doivent être fournis pour tous les espaces de noms représentés par des éléments dans la demande.
- Si plusieurs espaces de noms sont représentés dans la demande, les attributs d'espace de noms XML doivent inclure un préfixe d'identification, et ce préfixe doit être utilisé avec tous les éléments de cet espace de noms.
- Tous les éléments requis doivent apparaître dans les corps de la demande. Tous les éléments qui s'affichent dans les corps de demande doivent s'afficher dans l'ordre indiqué sur le schéma, et avec le contenu conforme à la contrainte de type spécifiée par le schéma.

Réponses d'API REST vRealize Operations Manager

Toutes les réponses incluent un code d'état HTTP et, sauf si le code d'état est 204 (aucun contenu), un en-tête `Accept`. Le contenu de la réponse dépend de la demande. Certaines réponses incluent un corps de document, certaines incluent uniquement une URL, et certaines sont vides.

Codes de réponse HTTP

Un client d'API peut s'attendre à recevoir un sous-ensemble de codes d'état HTTP dans une réponse.

Tableau 2-1. Codes d'état HTTP renvoyés par l'API

Code d'état	Description de l'état
200 OK	La demande est valide et a été effectuée. La réponse comprend un corps de document.
201 Créé	La demande est valide. L'objet demandé a été créé et se situe à l'URL spécifiée dans l'en-tête d'emplacement.
202 Accepté	La demande est valide et une tâche a été créée pour la gérer. Cette réponse est généralement accompagnée d'un élément <code>TaskStatus</code> .
204 Aucun contenu	La demande est valide et a été effectuée. La réponse ne comprend pas de corps.
400 Demande incorrecte	Le corps de la demande est incorrect, incomplet ou non valide.
401 Non autorisé	La connexion a échoué ou le jeton d'authentification a expiré.
403 Interdit	L'utilisateur n'est pas authentifié ou ne dispose pas des droits suffisants pour accéder à un ou plusieurs objets spécifiés dans la demande.
404 Non trouvé	L'objet spécifié dans la demande est introuvable.
405 Méthode non autorisée	La méthode HTTP spécifiée dans la demande n'est pas prise en charge pour cet objet.
406 Non acceptable	La ressource identifiée par la demande n'est pas en mesure de générer une réponse du type spécifié dans l'en-tête <code>Accept</code> de la demande.
415 Type de support non pris en charge	La ressource identifiée par la demande ne prend pas en charge une demande du <code>Content-Type</code> et de la méthode HTTP spécifiés.
422 Non trouvé	Indique généralement une URL de demande ou un corps de demande incorrect.
429 Trop de demandes	Un client a envoyé trop de demandes ou plusieurs clients envoient trop de demandes simultanément, et le serveur ne peut pas les traiter en raison du débit. Pour résoudre ce problème, renvoyez la demande ultérieurement.
500 Erreur de serveur interne	La demande a été reçue, mais n'a pas pu être exécutée en raison d'une erreur interne sur le serveur.

Tableau 2-1. Codes d'état HTTP renvoyés par l'API (suite)

Code d'état	Description de l'état
503 Service non disponible	Le serveur ne peut actuellement pas traiter la demande en raison d'une condition temporaire telle que l'épuisement des ressources ou la maintenance du serveur.
504 Délai d'expiration de la passerelle	Le serveur, tout en agissant en tant que passerelle ou proxy, n'a pas reçu de réponse en temps opportun du serveur en amont spécifié par l'URL de la demande.

Utilisation de l'API avec vRealize Operations Manager

Vous pouvez utiliser un navigateur ou un programme client HTTP pour envoyer des demandes et recevoir des réponses.

Programmes clients REST

Toute application client pouvant envoyer des demandes HTTPS peut être utilisée pour développer des applications REST avec l'API vRealize Operations Manager. Les plug-ins du client REST sont disponibles pour la plupart des navigateurs et de nombreux IDE. Les programmes Open Source suivants sont souvent utilisés :

- cURL. <http://curl.haxx.se>
- Application Postman. <http://www.getpostman.com>

En outre, VMware fournit des bindings client spécifiques au langage pour l'API vRealize Operations Manager. Reportez-vous à [À propos des références de schéma](#).

À propos des références de schéma

La documentation REST API vRealize Operations Manager comprend du matériel de référence pour tous les éléments, types, requêtes et opérations dans l'API vRealize Operations Manager. Elle comprend également les fichiers de définition de schéma.

La documentation de l'API basée sur Swagger est disponible avec le produit, avec la possibilité d'effectuer des appels d'API REST à partir de la page de lancement.

Pour accéder à la documentation de l'API, utilisez l'URL de votre instance de vRealize Operations Manager. Par exemple, si l'URL de votre instance est `https://vrealize.example.com`, la référence d'API est disponible à partir de : `https://vrealize.example.com/suite-api/doc/swagger-ui.html`.

Les bindings client spécifiques au langage sont disponibles à partir de :

```
https://vrealize.example.com/suite-api/
```

À propos des exemples d'API vRealize Operations Manager

Tous les exemples incluent des demandes HTTP et leurs réponses. Ces exemples présentent le workflow et le contenu associés à des opérations telles que la création et la demande d'informations sur des objets de votre environnement surveillé.

Les exemples de corps de demande sont au format JSON. Les en-têtes de demande requis par l'API vRealize Operations Manager sont inclus dans les exemples de demandes qui ne sont pas des fragments d'un exemple plus complet.

La plupart des exemples de réponses affichent uniquement les éléments et les attributs pertinents pour l'opération décrite. Les points de suspension (...) indiquent le contenu omis dans les corps de réponse.

Démarrage avec l'API

3

Les clients d'API et les serveurs vRealize Operations Manager communiquent via HTTPS, et s'échangent des représentations XML d'objets d'API.

Cet exemple simple de workflow REST API montre comment obtenir une liste de toutes les mesures pour un type d'objet de machine virtuelle qui est inclus dans la définition de modèle de l'adaptateur VMware vCenter®. À l'aide de l'API, vous pouvez obtenir la liste complète des mesures disponibles pour n'importe quel type d'objet.

Ce chapitre contient les rubriques suivantes :

- [Obtenir un jeton d'authentification](#)
- [Rechercher le type d'adaptateur et le type d'objet](#)
- [Générer une liste de toutes les mesures pour l'objet](#)

Obtenir un jeton d'authentification

vRealize Operations Manager requiert l'authentification des demandes d'API. La première étape de ce workflow consiste à obtenir un jeton d'authentification.

Pour obtenir un jeton d'authentification, la demande de connexion fournit les informations d'identification de l'utilisateur sous la forme requise par l'authentification HTTP simple. Dans cet exemple, l'utilisateur se connecte à une instance de vRealize Operations Manager avec l'URL `https://vrealize.example.com/`.

Note Cet exemple utilise l'authentification basée sur un jeton. Pour plus d'informations sur les mécanismes d'authentification, reportez-vous à la section [Sécurité](#).

Grâce à `authSource`, vous pouvez importer et authentifier les informations sur les utilisateurs et les groupes d'utilisateurs résidant sur une autre machine. Par exemple, vous pouvez authentifier des utilisateurs avec LDAP, Active Directory, VMware Identity Manager, Single Sign-On, etc. Lorsque vous importez des informations de comptes d'utilisateurs résidant sur un autre ordinateur, vous devez définir les critères utilisés pour importer les comptes d'utilisateurs de la machine source.

Après avoir créé une source d'authentification, vous pouvez l'utiliser pour acquérir un jeton en spécifiant son nom. Le type de source d'authentification par défaut est LOCAL.

Conditions préalables

- Sécurisez un canal entre le navigateur Web et le serveur vRealize Operations Manager. Ouvrez un navigateur et entrez l'URL d'une instance de vRealize Operations Manager telle que :

```
https://vrealize.example.com
```

Le système vous avertit que votre connexion n'est pas privée. Cliquez pour confirmer l'exception de sécurité et établir une liaison SSL.

- Vérifiez que vous pouvez accéder aux API. Entrez l'URL de votre instance de vRealize Operations Manager en ajoutant `suite-api/docs/rest/index.html` à la fin, par exemple :

```
https://vrealize.example.com/suite-api/docs/rest/index.html
```

- Vérifiez que vous disposez des informations d'identification de connexion d'un utilisateur de votre instance de vRealize Operations Manager.

Procédure

- 1 Envoyez une demande POST à l'URL de connexion pour obtenir un jeton.

```
POST https://vrealize.example.com/suite-api/api/auth/token/acquire
```

Reportez-vous à [Demande de connexion et réponse](#).

- 2 Examinez la réponse.

Une fois réussie, la demande renvoie un jeton d'autorisation `ops`, que vous devez inclure dans les demandes d'API suivantes.

Exemple : Demande de connexion et réponse

Cet exemple présente une demande et une réponse pour un utilisateur dont le nom d'utilisateur de connexion est : **vRealize-user** et le mot de passe est : **vRealize-dummy-password**.

En-tête de la demande :

```
POST https://vrealize.example.com/suite-api/api/auth/token/acquire
Content-Type: application/json
Accept: application/json
```

Corps de la demande au format JSON :

```
{
  "username" : "vRealize-user",
  "password" : "vRealize-dummy-password"
}
```

Réponse au format JSON :

```
200 OK
```

```
{
  "token": "8f868cca-27cc-43d6-a838-c5467e73ec45::77cea9b2-1e87-490e-b626-e878beea23b",
  "validity": 1470421325035,
  "expiresAt": "Friday, August 5, 2016 6:22:05 PM UTC",
  "roles": []
}
```

Le code de réponse indique si la demande a réussi ou pourquoi elle a échoué.

- Si la demande a réussi, le serveur renvoie le code de réponse HTTP 200 (OK) et un jeton d'autorisation `ops` réutilisable, qui expire après six heures. Ce jeton doit être inclus dans toutes les demandes d'API suivantes.
- Si l'en-tête d'autorisation manque pour cette demande, le serveur renvoie le code de réponse HTTP 403.
- Si les informations d'identification fournies dans l'en-tête d'autorisation ne sont pas valides, le serveur renvoie le code de réponse HTTP 401.

Rechercher le type d'adaptateur et le type d'objet

Votre instance de vRealize Operations Manager comprend plusieurs types d'adaptateur. Pour trouver le type d'adaptateur pour l'adaptateur vCenter, effectuez une demande GET afin de récupérer la liste de tous les types d'adaptateur. La réponse de l'API comprend tous les types d'objet que l'adaptateur surveille.

Conditions préalables

Vérifiez que vous êtes connecté à l'instance de vRealize Operations Manager.

Procédure

- 1 Effectuez une demande GET pour tous les types d'adaptateur.

```
GET https://vrealize.example.com/suite-api/api/adaptekinds
```

- 2 Examinez la réponse pour trouver l'adaptateur vCenter et la liste des types d'objet surveillés.
Reportez-vous à la section réponse de [Déterminer le type d'adaptateur et les types d'objet pour l'adaptateur vCenter](#).

Exemple : Déterminer le type d'adaptateur et les types d'objet pour l'adaptateur vCenter

Cet exemple permet de rechercher le type d'adaptateur pour l'adaptateur vCenter et tous les types d'objet inclus dans la définition du modèle d'adaptateur.

En-tête de la demande :

```
GET https://vRealize.example.com/suite-api/api/adapterkinds
Content-Type: application/json
Authorization: vRealizeOpsToken <vROps_token>
Accept: application/json
```

Où *vROps_token* est le jeton que vous avez obtenu à partir de la réponse dans [Obtenir un jeton d'authentification](#).

Extrait de la réponse au format JSON pour l'adaptateur vCenter :

```
200 OK
```

```
{
  "key": "VMWARE",
  "name": "vCenter Adapter",
  "description": "Provides the connection information and credentials required...",
  "adapterKindType": "GENERAL",
  "describeVersion": 573,
  "identifiers": [],
  "resourceKinds": [
    "ClusterComputeResource",
    "ComputeResource",
    "CustomDatacenter",
    "Datacenter",
    "Datastore",
    "StoragePod",
    "DatastoreFolder",
    "VM Entity Status",
    "Folder",
    "HostFolder",
    "HostSystem",
    "NetworkFolder",
    "ResourcePool",
    "VMwareAdapter Instance",
    "VirtualMachine",
    "VMFolder",
    "DistributedVirtualPortgroup",
    "VmwareDistributedVirtualSwitch",
    "vSphere World"
  ],
  ...
}
```

Pour l'adaptateur vCenter, la clé `adapter-kind` est `VMWARE`. Les `resourceKinds` sont les types d'objet surveillés par l'adaptateur vCenter. Pour le type d'objet de machine virtuelle, le `resourceKinds` est `VirtualMachine`.

Générer une liste de toutes les mesures pour l'objet

Pour générer une liste complète de mesures pour toutes les machines virtuelles définies dans le modèle de l'adaptateur vCenter, vous devez envoyer une demande GET à l'URL avec le type d'adaptateur et le type d'objet.

Conditions préalables

Vérifiez que les conditions suivantes sont réunies :

- Vous êtes connecté à l'instance de vRealize Operations Manager.
- Vous connaissez la valeur `adapterKind` de l'adaptateur vCenter et la valeur `resourceKind` de la machine virtuelle. Reportez-vous à [Déterminer le type d'adaptateur et les types d'objet pour l'adaptateur vCenter](#)

Procédure

- 1 Envoyez une demande GET pour obtenir les métadonnées des mesures.

```
GET https://vrealize.example.com/suite-api/api/adapterkinds/VMWARE/resourcekinds/
VirtualMachine/statkeys
```

- 2 Comparez les mesures répertoriées dans la réponse aux mesures affichées dans l'interface utilisateur. Reportez-vous à [Mesures de machine virtuelle à partir de l'API et dans l'interface utilisateur](#)

Exemple : Mesures de machine virtuelle à partir de l'API et dans l'interface utilisateur

Cet exemple décrit la façon dont les mesures de machine virtuelle répertoriées dans la réponse XML peuvent être comparées aux mesures affichées dans l'interface utilisateur de vRealize Operations Manager.

Demande :

```
GET https://vrealize.example.com/suite-api/api/adapterkinds/VMWARE/resourcekinds/
VirtualMachine/statkeys
Content-Type: application/json
Authorization: vRealizeOpsToken <vROps_token>
Accept: application/json
```

Où :

- `VMWARE` est `adapterKindKey`.
- `VirtualMachine` est `resourceKindKey`.
- `vROps_token` est le jeton que vous avez obtenu à partir de la réponse dans [Obtenir un jeton d'authentification](#)

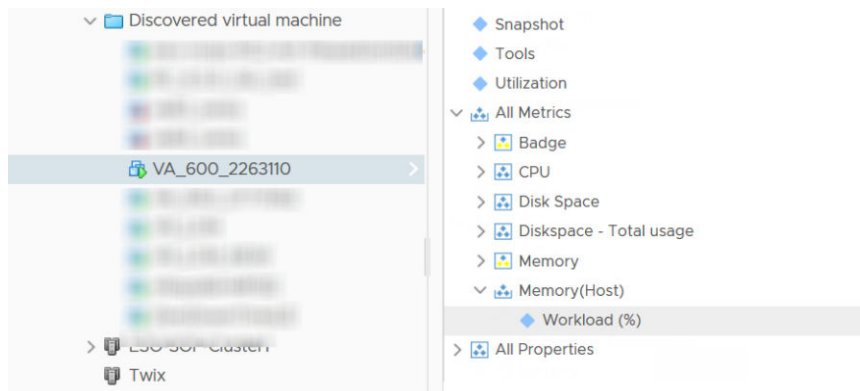
Extrait de la réponse au format JSON :

200 OK

```
{
  "resourceTypeAttributes": [
    ...
    {
      "key": "mem|host_workload",
      "name": "Memory|Host Workload",
      "description": "Host Workload (%)",
      "defaultMonitored": false,
      "rollupType": "AVG",
      "instanceType": "INSTANCED",
      "unit": "%",
      "dataType2": "FLOAT",
      "monitoring": false,
      "property": false
    },
    ...
  ]
}
```

Chaque `resourceTypeAttribute` dans la réponse est une mesure avec des métadonnées pour un objet de machine virtuelle. `name` correspond au texte affiché dans l'interface utilisateur de vRealize Operations Manager. Dans cet exemple, l'extrait répertorie les mesures pour Mémoire et Charge de travail de l'hôte.

Pour comparer les mesures obtenues dans la réponse aux mesures dans l'interface utilisateur, connectez-vous à l'instance de vRealize Operations Manager exécutée sur `vrealize.example.com` et accédez aux mesures d'une machine virtuelle. L'exemple suivant montre où vous pouvez trouver les mesures pour Mémoire (hôte) et Charge de travail.



L'exemple montre comment récupérer des mesures pour le type d'objet de machine virtuelle. Pour récupérer des mesures pour d'autres types d'objet, remplacez `VirtualMachine` dans la demande GET par d'autres `resourceKinds`.

Configuration d'une instance de l'adaptateur

4

Après avoir installé une solution incluant un module de gestion avec un adaptateur, vous devez configurer une instance de l'adaptateur pour collecter des données auprès des objets dans la définition du modèle de l'adaptateur. Vous pouvez utiliser l'API vRealize Operations Manager pour configurer une instance de l'adaptateur.

Cet exemple de cas d'utilisation décrit la configuration d'une instance de l'adaptateur pour une solution vSphere et comprend les éléments suivants :

- résumé des opérations avec demande, corps de la demande et réponse pour chacune d'elles
- procédure spécifique pour chaque opération

Ce chapitre contient les rubriques suivantes :

- [Résumé de la configuration des demandes d'instance de l'adaptateur](#)
- [Identifier la solution et ses adaptateurs](#)
- [Identifier les types d'objet requis pour l'adaptateur](#)
- [Créer l'instance de l'adaptateur](#)
- [Fournir la preuve de la validité du certificat](#)
- [Commencer la surveillance de la nouvelle instance de l'adaptateur](#)

Résumé de la configuration des demandes d'instance de l'adaptateur

Effectuez des demandes d'API séquentielles pour configurer une instance de l'adaptateur. Les réponses provenant des demandes antérieures génèrent des informations nécessaires pour les demandes suivantes.

Tableau 4-1. Résumé des demandes

Opération	Demande	Corps de la demande	Réponse
Obtenir toutes les solutions enregistrées avec le produit et identifier les types d'adaptateur.	GET <URL de l'API>/suite-api/api/solutions	Aucun	adapterkindkeys
Obtenir tous les types d'objet d'un type d'adaptateur particulier.	GET <URL de l'API>/suite-api/api/adapterkinds/{key}/resourcekinds	Aucun	resourceIdentifierTypes
Créer un objet d'instance de l'adaptateur	POST <URL de l'API>/suite-api/api/adapters	Valeurs pour resourceIdentifiers et credential	adapterid
Corriger une instance de l'adaptateur pour confirmer le certificat présenté	PATCH <URL de l'API>/suite-api/api/adapters	Réponse de l'API de POST <URL de l'API>/suite-api/api/adapters	Réponse de l'API de POST <URL de l'API>/suite-api/api/adapters sans adapter-cerfificates
Démarrer la surveillance de l'adaptateur	PUT <URL de l'API>/suite-api/api/adapters/{adapterid}/monitoringstate/start	Aucun	200 OK

Identifier la solution et ses adaptateurs

Votre instance de vRealize Operations Manager peut avoir plusieurs solutions installées. Pour trouver la solution vSphere et ses types d'adaptateur, effectuez une demande GET pour récupérer une liste de toutes les solutions. La réponse comprend tous les adaptateurs inclus avec la solution.

Dans cet exemple de cas d'utilisation, l'*URL de l'API* pour l'instance de vRealize Operations Manager est `vrealize.example.com`.

Conditions préalables

Vérifiez que vous pouvez vous connecter à l'URL de l'API pour une instance de vRealize Operations Manager. Reportez-vous à [Obtenir un jeton d'authentification](#).

Procédure

- 1 Effectuez une demande GET pour répertorier toutes les solutions.

```
GET https://vrealize.example.com/suite-api/api/solutions
```

- 2 Examinez la réponse pour trouver la solution vSphere et ses types d'adaptateur.

Reportez-vous à la section réponse de [Types d'adaptateur pour la solution vSphere](#).

Exemple : Types d'adaptateur pour la solution vSphere

Cet exemple répertorie toutes les solutions installées et les types d'adaptateur pour chacune d'elles.

En-tête de la demande :

```
GET https://vrealize.example.com/suite-api/api/solutions
```

Réponse au format JSON :

200 OK

```
{
  "solution":
    [
      {
        "id": "MPforLogInsight",
        "name": "VMware vRealize Operations Management Pack for Log Insight",
        "version": "6.0.3171089",
        "description": "VMware vRealize Operations Management Pack for Log Insight... ",
        "vendor": "VMware Inc.",
        "adapterKindKeys": [
          "LogInsightAdapter"
        ]
      },
      {
        "id": "ep-ops-os-and-availability",
        "name": "Operating Systems / Remote Service Monitoring",
        "version": "1.0.4071095",
        "description": "The End Point Operations Management Solution for Operating... ",
        "vendor": "VMware Inc.",
        "adapterKindKeys": [
          "ep-ops-os-and-availability-kind"
        ]
      },
      {
        "id": "VMware vSphere",
        "name": "VMware vSphere",
        "version": "6.0.7496664",
        "description": "Manages vSphere objects such as Clusters, Hosts...",
        "vendor": "VMware Inc.",
        "adapterKindKeys": [
          "VMWARE",
          "PythonRemediationVcenterAdapter"
        ]
      }
    ]
}
```

La réponse indique trois que solutions sont installées :

- Solution Management Pack for Log Insight
- Solution Opérations de point de terminaison

- Solution vSphere

La solution vSphere dispose de deux types d'adaptateur :

- VMWARE
- PythonRemediationVcenterAdapter

Pour l'adaptateur vCenter, le type d'adaptateur est VMWARE.

Identifier les types d'objet requis pour l'adaptateur

Une fois que vous savez que vous souhaitez créer une instance de l'adaptateur vCenter, vous devez identifier les types d'objet requis pour cet adaptateur. Effectuez une demande GET pour récupérer une liste de tous les types d'objet de l'adaptateur vCenter.

Conditions préalables

Vérifiez que vous connaissez le type de l'adaptateur vCenter.

Procédure

- 1 Effectuez une demande GET pour répertorier tous les types d'objet de l'adaptateur vCenter.

```
GET https://vrealize.example.com/suite-api/api/adapterkinds/VMWARE/resourcekinds
```

- 2 Examinez la réponse pour identifier les types d'objet requis.

Reportez-vous à la section réponse de [Types d'objet requis pour l'adaptateur vCenter](#).

Exemple : Types d'objet requis pour l'adaptateur vCenter

Cet exemple permet de trouver tous les types d'objet de l'adaptateur vCenter.

En-tête de la demande :

```
GET https://vrealize.example.com/suite-api/api/adapterkinds/VMWARE/resourcekinds
```

Extrait de la réponse au format JSON :

```
200 OK
```

```
{
  "key": "VMwareAdapter Instance",
  "name": "vCenter Server",
  "adapterKind": "VMWARE",
  "resourceKindType": "ADAPTER_INSTANCE",
  "resourceKindSubType": "NONE",
  "resourceIdentifierTypes": [
    {
      "name": "AUTODISCOVERY",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
  ],
}
```

```

{
  "name": "DISABLE_COMPUTATION_BASED_ON_CONSUMERS",
  "dataType": "STRING",
  "isPartOfUniqueness": false
},
{
  "name": "DV_PORT_GROUP_DISABLED",
  "dataType": "STRING",
  "isPartOfUniqueness": false
},
{
  "name": "DVS_DISABLED",
  "dataType": "STRING",
  "isPartOfUniqueness": false
},
{
  "name": "PROCESSCHANGEEVENTS",
  "dataType": "STRING",
  "isPartOfUniqueness": false
},
{
  "name": "VCURL",
  "dataType": "STRING",
  "isPartOfUniqueness": true
},
...
{
  "name": "VM_LIMIT",
  "dataType": "INTEGER",
  "isPartOfUniqueness": false
}
],
...
}

```

Cet extrait indique le type de ressource avec l'attribut `"resourceKindType"`: `"ADAPTER_INSTANCE"`. Tout type d'objet disposant de l'identifiant de ressource `"isPartOfUniqueness":true` doit être accompagné d'une valeur pour ce type d'objet avec la demande d'API pour créer l'instance de l'adaptateur.

Une instance d'adaptateur pour l'adaptateur vCenter nécessite une valeur pour `VCURL` ou l'URL du vCenter.

Créer l'instance de l'adaptateur

Une fois que vous avez identifié les types d'objet requis pour l'adaptateur, indiquez les valeurs de paramètre pour les types d'objet afin de créer une instance de l'adaptateur. Votre demande POST comprend le corps de la demande avec les paramètres requis.

Le paramètre `VCURL` est obligatoire pour créer une instance de l'adaptateur.

Conditions préalables

Vérifiez que vous disposez d'une adresse IP et d'informations d'identification pour un vCenter.

Procédure

- 1 Effectuez une demande POST pour créer l'instance de l'adaptateur.

```
POST https://vrealize.example.com/suite-api/api/adapters
```

- 2 Examinez la réponse pour trouver le nom de la solution vSphere et ses types d'adaptateur.
Reportez-vous à la section réponse de [Instance de l'adaptateur](#).

Exemple : Instance de l'adaptateur

Cet exemple crée l'instance de l'adaptateur pour un vCenter avec les paramètres suivants :

- Nom complet : instance de l'adaptateur VC
- Description : instance de l'adaptateur vCenter pour VC 12.345.678.9
- Adresse IP de vCenter Server : https://12.345.678.9
- Nom des informations d'identification : VC-Credentials-1
- Nom d'utilisateur : administrator@vsphere.local
- Mot de passe : VC-dummy-passwd

AUTODISCOVERY et PROCESSCHANGEEVENTS sont facultatifs, mais sont tout de même inclus pour afficher des exemples supplémentaires d'identifiants de ressources dans le corps de la demande et dans la réponse.

En-tête de la demande :

```
POST https://vrealize.example.com/suite-api/api/adapters
```

Corps de la demande au format JSON :

```
{
  "name" : "VC Adapter Instance",
  "description" : "A vCenter Adapter Instance for VC 12.345.678.9",
  "collectorId" : "1",
  "adapterKindKey" : "VMWARE",
  "resourceIdentifiers" : [
    {
      "name" : "AUTODISCOVERY",
      "value" : "true"
    },
    {
      "name" : "PROCESSCHANGEEVENTS",
      "value" : "true"
    },
    {
      "name" : "VCURL",
```

```

        "value" : "https://12.345.678.9"
    }
],
"credential" : {
    "id" : null,
    "name" : "VC-Credential-1",
    "adapterKindKey" : "VMWARE",
    "credentialKindKey" : "PRINCIPALCREDENTIAL",
    "fields" : [
        {
            "name" : "USER",
            "value" : "administrator@vsphere.local"
        },
        {
            "name" : "PASSWORD",
            "value" : "VC-dummy-passwd"
        }
    ]
},
},
}

```

Extrait de la réponse au format JSON :

201 Created

```

{
  "resourceKey": {
    "name": "VC Adapter Instance",
    "adapterKindKey": "VMWARE",
    "resourceKindKey": "VMwareAdapter Instance",
    "resourceIdentifiers": [
      {
        "identifierType": {
          "name": "AUTODISCOVERY",
          "dataType": "STRING",
          "isPartOfUniqueness": false
        },
        "value": "true"
      },
      {
        "identifierType": {
          "name": "DISABLE_COMPUTATION_BASED_ON_CONSUMERS",
          "dataType": "STRING",
          "isPartOfUniqueness": false
        },
        "value": ""
      },
      {
        "identifierType": {
          "name": "DV_PORT_GROUP_DISABLED",
          "dataType": "STRING",
          "isPartOfUniqueness": false
        },
        "value": ""
      }
    ]
  }
}

```

```

    },
    {
      "identifierType": {
        "name": "DVS_DISABLED",
        "dataType": "STRING",
        "isPartOfUniqueness": false
      },
      "value": ""
    },
    {
      "identifierType": {
        "name": "PROCESSCHANGEEVENTS",
        "dataType": "STRING",
        "isPartOfUniqueness": false
      },
      "value": "true"
    },
    {
      "identifierType": {
        "name": "VCURL",
        "dataType": "STRING",
        "isPartOfUniqueness": true
      },
      "value": "https://12.345.678.9"
    },
    {
      "identifierType": {
        "name": "VM_FOLDER_DISABLED",
        "dataType": "STRING",
        "isPartOfUniqueness": false
      },
      "value": ""
    },
    {
      "identifierType": {
        "name": "VM_LIMIT",
        "dataType": "STRING",
        "isPartOfUniqueness": false
      },
      "value": ""
    }
  ]
},
"description": "A vCenter Adapter Instance for VC 12.345.678.9",
"collectorId": 1,
"collectorGroupId": "909c2fbf-2c2c-4957-9a75-21bf2a887d31",
"credentialInstanceId": "65081a8d-d462-43b2-b4e0-596eaf3d497e",
"monitoringInterval": 5,
"adapter-certificates": [
  {
    "thumbprint": "2520fb4351bc91ee7b82ef7cc54a8d88fa893da9",
    "certificateDetails": "[
      Version: V3 Subject: C=US, CN=12.345.678.9
      Signature Algorithm: SHA256withRSA, OID = 1.2.840.113549.1.1.11
      Key: Sun RSA public key, 2048 bits modulus: ...
    ]"
  }
]

```

```

    Validity: [From: Wed Jul 15 19:26:51 UTC 2015, To: Tue Jul 08 11:26:30 UTC 2025]
    Issuer: O=W12R2UINanduVC, C=US, DC=local, DC=vsphere, CN=CA ...
    ...
  ]"
}
],
...
"id": "a97bd204-e3e5-404b-a219-e2b20cf158d2"
}

```

L'API crée un nouvel adaptateur avec un UUID généré en interne qui identifie l'objet de manière unique. La réponse de l'API comporte les certificats présentés par vCenter 12.345.678.9. La valeur de l'ID d'instance de l'adaptateur est utilisée pour démarrer la surveillance et la collecte des données.

Fournir la preuve de la validité du certificat

Avant que vRealize Operations Manager ne puisse se connecter à vCenter et commencer à collecter des données, il doit vérifier que les sources de données détectées par l'instance de l'adaptateur présentent des certificats valides. Votre demande PATCH fournit la preuve que les certificats sont valides en incluant un corps de demande qui est la réponse à la demande POST utilisée pour créer l'adaptateur.

Conditions préalables

Vérifiez que vous disposez de la réponse à la demande POST utilisée pour créer l'adaptateur. Consultez la réponse dans [Instance de l'adaptateur](#).

Procédure

- ◆ Effectuez une demande PATCH pour notifier au système que l'utilisateur a accepté le certificat présenté par le vCenter.

```
PATCH https://vrealize.example.com/suite-api/api/adapters
```

Exemple : Validation du certificat

Dans cet exemple, le corps de la demande PATCH est identique à la réponse à la demande POST utilisée pour créer l'instance de l'adaptateur.

En-tête de la demande :

```
PATCH https://vrealize.example.com/suite-api/api/adapters
```

Corps de la demande au format JSON :

```

{
  "resourceKey": {
    "name": "VC Adapter Instance",
    "adapterKindKey": "VMWARE",
    "resourceKindKey": "VMwareAdapter Instance",

```

```

"resourceIdentifiers": [
  {
    "identifierType": {
      "name": "AUTODISCOVERY",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": "true"
  },
  {
    "identifierType": {
      "name": "DISABLE_COMPUTATION_BASED_ON_CONSUMERS",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": ""
  },
  {
    "identifierType": {
      "name": "DV_PORT_GROUP_DISABLED",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": ""
  },
  {
    "identifierType": {
      "name": "DVS_DISABLED",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": ""
  },
  {
    "identifierType": {
      "name": "PROCESSCHANGEEVENTS",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": "true"
  },
  {
    "identifierType": {
      "name": "VCURL",
      "dataType": "STRING",
      "isPartOfUniqueness": true
    },
    "value": "https://12.345.678.9"
  },
  {
    "identifierType": {
      "name": "VM_FOLDER_DISABLED",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": ""
  }
]

```

```

        "value": ""
    },
    {
        "identifierType": {
            "name": "VM_LIMIT",
            "dataType": "STRING",
            "isPartOfUniqueness": false
        },
        "value": ""
    }
]
},
"description": "A vCenter Adapter Instance for VC 12.345.678.9",
"collectorId": 1,
"collectorGroupId": "909c2fbf-2c2c-4957-9a75-21bf2a887d31",
"credentialInstanceId": "65081a8d-d462-43b2-b4e0-596eaf3d497e",
"monitoringInterval": 5,
"adapter-certificates": [
    {
        "thumbprint": "2520fb4351bc91ee7b82ef7cc54a8d88fa893da9",
        "certificateDetails": "[
            Version: V3 Subject: C=US, CN=12.345.678.9
            Signature Algorithm: SHA256withRSA, OID = 1.2.840.113549.1.1.11
            Key: Sun RSA public key, 2048 bits modulus: ...
            Validity: [From: Wed Jul 15 19:26:51 UTC 2015, To: Tue Jul 08 11:26:30 UTC 2025]
            Issuer: O=W12R2UINanduVC, C=US, DC=local, DC=vsphere, CN=CA ...
            ...
        ]"
    }
],
...
"id": "a97bd204-e3e5-404b-a219-e2b20cf158d2"
}

```

Réponse au format JSON :

```

{
    "resourceKey": {
        "name": "VC Adapter Instance",
        "adapterKindKey": "VMWARE",
        "resourceKindKey": "VMwareAdapter Instance",
        "resourceIdentifiers": [
            {
                "identifierType": {
                    "name": "AUTODISCOVERY",
                    "dataType": "STRING",
                    "isPartOfUniqueness": false
                },
                "value": "true"
            },
            {
                "identifierType": {
                    "name": "DISABLE_COMPUTATION_BASED_ON_CONSUMERS",
                    "dataType": "STRING",
                    "isPartOfUniqueness": false
                }
            }
        ]
    }
}

```

```

    },
    "value": ""
  },
  {
    "identifierType": {
      "name": "DV_PORT_GROUP_DISABLED",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": ""
  },
  {
    "identifierType": {
      "name": "DVS_DISABLED",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": ""
  },
  {
    "identifierType": {
      "name": "PROCESSCHANGEEVENTS",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": "true"
  },
  {
    "identifierType": {
      "name": "VCURL",
      "dataType": "STRING",
      "isPartOfUniqueness": true
    },
    "value": "https://12.345.678.9"
  },
  {
    "identifierType": {
      "name": "VM_FOLDER_DISABLED",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": ""
  },
  {
    "identifierType": {
      "name": "VM_LIMIT",
      "dataType": "STRING",
      "isPartOfUniqueness": false
    },
    "value": ""
  }
]
},
"description": "A vCenter Adapter Instance for VC 12.345.678.9",
"collectorId": 1,

```

```

"collectorGroupId": "909c2fbf-2c2c-4957-9a75-21bf2a887d31",
"credentialInstanceId": "65081a8d-d462-43b2-b4e0-596eaf3d497e",
"monitoringInterval": 5,
...
"id": "a97bd204-e3e5-404b-a219-e2b20cf158d2"
}

```

La réponse est identique au corps de la demande sans la section `adapter-certificates`.

Commencer la surveillance de la nouvelle instance de l'adaptateur

Après la création de l'instance de l'adaptateur et la configuration de vRealize Operations Manager pour confirmer un certificat valide, commencez à surveiller et à collecter les données. Votre demande PUT fournit l'UUID de l'instance de l'adaptateur utilisée pour détecter de nouveaux objets.

Conditions préalables

Vérifiez que vous disposez de l'UUID de l'instance de l'adaptateur récemment créée. Consultez la réponse dans [Instance de l'adaptateur](#).

Procédure

- ◆ Effectuez une demande PUT pour démarrer la surveillance avec la nouvelle instance de l'adaptateur.

```

PUT https://vrealize.example.com/suite-api/api/adapters/<adapter_UUID>/monitoringstate/
start

```

Exemple : Détecter des objets et collecter des données

Cet exemple permet de démarrer le processus de surveillance de l'adaptateur à l'aide de l'ID de l'instance de l'adaptateur provenant de la demande PUT qui a créé l'instance de l'adaptateur.

En-tête de la demande :

```

PUT https://vrealize.example.com/suite-api/api/adapters/a97bd204-e3e5-404b-a219-e2b20cf158d2/
monitoringstate/start

```