

API- Programmierhandbuch

30. MRZ 2022

vRealize Operations 8.4

Die aktuellste technische Dokumentation finden Sie auf der VMware-Website unter:

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

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

VMware Global, Inc.
Zweigniederlassung Deutschland
Willy-Brandt-Platz 2
81829 München
Germany
Tel.: +49 (0) 89 3706 17 000
Fax: +49 (0) 89 3706 17 333
www.vmware.com/de

Copyright © 2022 VMware, Inc. Alle Rechte vorbehalten. [Urheberrechts- und Markenhinweise](#).

Inhalt

- 1 Informationen zum Programmierhandbuch 4**
- 2 Grundlegendes zur vRealize Operations Manager-API 5**
 - Funktionsweise der vRealize Operations Manager-API 5
 - Übersicht über den Client-Workflow 7
 - vRealize Operations Manager-API – REST-Anforderungen 8
 - vRealize Operations Manager-API – REST-Antworten 10
 - Verwenden der API mit vRealize Operations Manager 12
- 3 Erste Schritte mit der API 14**
 - Erwerben eines Authentifizierungs-Token 14
 - Auffinden des Adapter- und Objekttyps 16
 - Generieren einer Liste aller Metriken für das Objekt 18
- 4 Konfigurieren einer Adapterinstanz 20**
 - Übersicht über die Konfiguration von Adapterinstanzanforderungen 20
 - Ermitteln der Lösung und ihrer Adapter 21
 - Ermitteln, welche Objekttypen für einen Adapter benötigt werden 23
 - Erstellen der Adapterinstanz 24
 - Gültigkeit eines Zertifikats nachweisen 28
 - Starten der Überwachung einer neuen Adapterinstanz 32

Informationen zum Programmierhandbuch

1

Das Programmierhandbuch für die vRealize Operations Manager-API enthält Informationen über die REST APIs von vRealize Operations Manager, z. B. zur Nutzung von Ressourcen der REST APIs, zur Authentifizierung und zum Aufbau von REST API-Aufrufen.

Zielgruppe

Diese Informationen sind für Administratoren und Programmierer gedacht, die vRealize Operations Manager programmgesteuert mithilfe der REST API von vRealize Operations Manager konfigurieren und verwalten möchten. Im Mittelpunkt des Handbuchs stehen häufig auftretende Anwendungsfälle.

Grundlegendes zur vRealize Operations Manager-API

2

Entwickler können interaktive Clients von vRealize Operations Manager mithilfe der API erstellen. Die API folgt dem REST-Stil und steht allen lizenzierten Benutzern zur Verfügung.

vRealize Operations Manager-Clients kommunizieren über HTTP mit dem-Server und tauschen Darstellungen von vRealize Operations Manager-Objekten aus. Diese Darstellungen nehmen die Form von JSON- oder XML-Elementen an. Mit HTTP GET-Anforderungen rufen Sie die aktuelle Darstellung eines Objekts oder einer HTTP POST- bzw. PUT-Anforderung zur Erstellung oder Änderung eines Objekts bzw. einer HTTP DELETE-Anforderung zur Löschung eines Objekts ab.

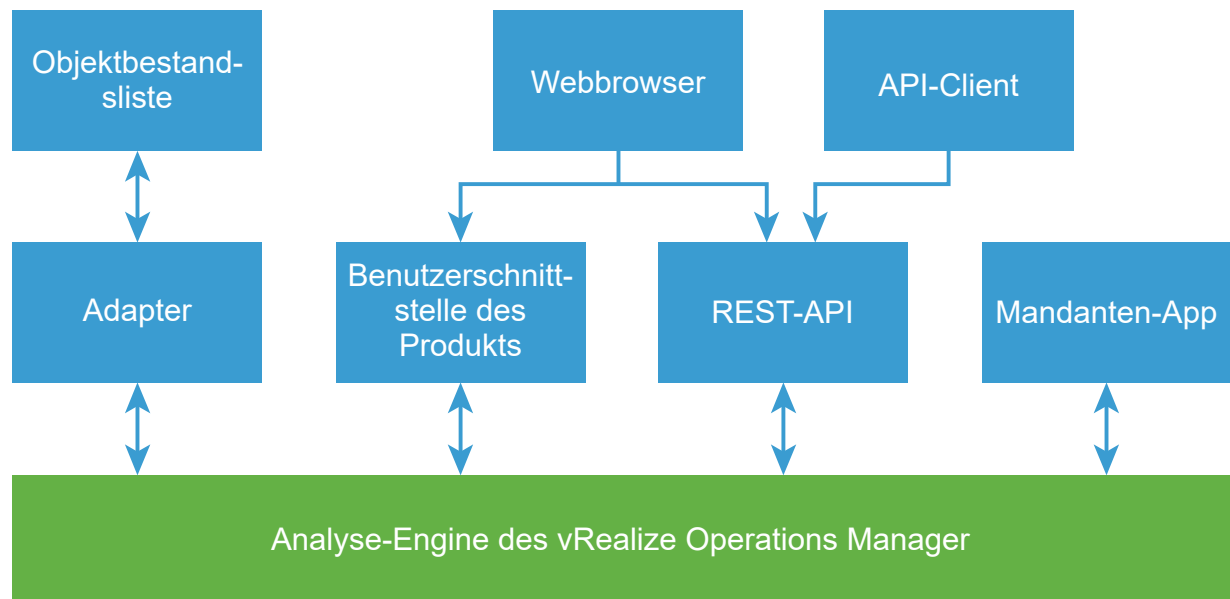
Dieses Kapitel enthält die folgenden Themen:

- [Funktionsweise der vRealize Operations Manager-API](#)
- [Übersicht über den Client-Workflow](#)
- [Verwenden der API mit vRealize Operations Manager](#)

Funktionsweise der vRealize Operations Manager-API

Für die Kommunikation mit der Analyse-Engine von vRealize Operations Manager verwenden Sie einen Webbrowser, entweder über die Benutzerschnittstelle des Produkts oder über API-Aufrufe.

Abbildung 2-1. vRealize Operations Manager – Vereinfachte Architektur



Die Adapterinstanz erfasst Daten von Objekten in Ihrer überwachten Umgebung. Die Analyse-Engine von vRealize Operations Manager erfasst die Daten und zeigt das komplette Modell in der grafischen Schnittstelle an.

Wozu Sie die API verwenden sollten

Die API ist besonders nützlich, wenn ein genau definierter Workflow automatisiert werden muss, z. B. wenn immer dieselben Aufgaben wiederholt werden, um die Zugriffssteuerung für neue Benutzer von vRealize Operations Manager zu konfigurieren. Die API ist auch bei Anfragen auf dem Daten-Repository von vRealize Operations Manager nützlich, z. B. zum Abrufen von Daten für bestimmte Assets in Ihrer virtuellen Umgebung. Außerdem können Sie mit der API alle Daten aus dem Daten-Repository von vRealize Operations Manager extrahieren und in ein separates Analysesystem laden.

vRealize Operations Manager – Terminologie

Die XML-Syntax, die Sie zur Beschreibung der Objekte für einen Adapter verwenden, entspricht der Syntax des API-Code, unterscheidet sich jedoch von dem, was Sie in der Benutzerschnittstelle finden. Die folgenden Begriffe tauchen in der Benutzerschnittstelle auf. Die Beschreibung jedes Begriffs enthält die entsprechende XML-Syntax, die bei einem API-Aufruf verwendet wird.

Adaptertypen	Legt fest, welcher Adapter zur Ermittlung bestimmter Objekttypen verwendet wird. Beispielsweise ermittelt der vCenter-Adapter Objekte, die mit vSphere-Rechenzentren verbunden sind. Der AWS-Adapter ermittelt AWS-Services und AWS-Objekte. XML-Syntax: <code>adapterkinds</code> .
Objekttypen	Die Klasse der Entitäten, die Objekte oder Informationsquellen darstellen. Objekte melden Daten an die Analyse-Engine von vRealize Operations Manager. Virtuelle Maschinen, Datenspeicher und Host-Systeme sind Beispiele für Objekttypen, die in einem vCenter-Adaptermodell definiert sind. XML-Syntax: <code>resourcekinds</code> .

Übersicht über den Client-Workflow

Clients der vRealize Operations Manager-API implementieren einen REST-Workflow, indem sie HTTP-Anforderungen an den Server senden und den Antworten des Servers die benötigten Informationen entnehmen.

Über REST

REST, ein Akronym für „Representational State Transfer“, beschreibt ein architektonisches Merkmal von Programmen, bei denen das Hypertext Transfer Protocol (HTTP) für den Austausch serialisierter Objektdarstellungen zwischen einem Client und einem Server verwendet wird. In der vRealize Operations Manager-API handelt es sich bei diesen Darstellungen um JSON- oder XML-Dokumente.

In einem REST-Workflow werden zwischen einem Client und einem Server Darstellungen von Objekten in der ausdrücklichen Annahme übergeben, dass keine Seite etwas anderes über ein Objekt wissen muss als das, was in einer einzelnen Anforderung oder Antwort dargestellt wird. Die URLs, unter denen diese Dokumente verfügbar sind, haben oft länger Bestand als die Anforderung oder die Antwort, in der sie enthalten sind.

REST API – Workflows

Zur Verwendung einer REST API erstellte Anwendungsprogramme verwenden HTTP-Anforderungen, die häufig von einem Skript oder einer anderen Sprache auf höherer Ebene ausgeführt werden, um aus der Ferne Prozesse zum Erstellen, Abrufen, Aktualisieren oder Löschen von der API festgelegter Objekte aufzurufen. In der REST API von vRealize Operations Manager werden diese Objekte durch eine Sammlung von XML-Schemata definiert. Bei den Vorgängen selbst handelt es sich um HTTP-Anforderungen. Sie sind daher für alle HTTP-Clients generisch.

Um eine Client-Anwendung für eine REST API zu schreiben, müssen Sie nur das HTTP-Protokoll und die Semantik von JSON bzw. XML (dem Übertragungsformat, das die vRealize Operations Manager-API verwendet) kennen. Um die API in einem solchen Client effektiv einzusetzen, müssen Sie sich mit den folgenden Konzepten vertraut machen.

- Dem Satz von Objekten, den die API unterstützt, und dem, was diese Objekte darstellen.
- Wie die API diese Objekte darstellt.
- Wie ein Client sich auf ein Objekt bezieht, mit dem ein Vorgang ausgeführt werden soll.

Die API-Referenz enthält eine vollständige Auflistung der API-Anforderungen. Weitere Informationen hierzu finden Sie unter [Informationen zur Schemareferenz](#).

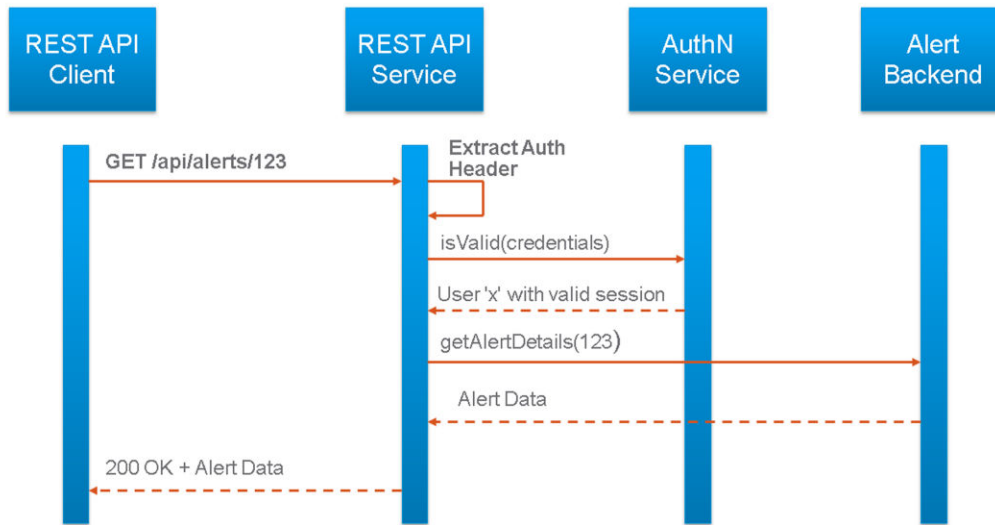
vRealize Operations Manager-API – REST-Anforderungen

Clients stellen HTTP-Anforderungen an Objektreferenzen, um Objektdarstellungen abzurufen.

Sicherheit

Der HTTP-Link zwischen einem API-Client und dem API-Server wird per SSL hergestellt. API-Clients konfigurieren die Token-Authentifizierung für die Kommunikation mit dem Server.

Abbildung 2-2. Szenario: Benutzeranmeldedaten angeben, um Informationen zu einer Warnung mit der ID 123 abzurufen



Bei der Token-Authentifizierung senden Sie eine POST-Anforderung an den API-Server von vRealize Operations Manager API und geben dabei gültige Benutzeranmeldedaten an, um ein Authentifizierungs-Token zu erhalten. Das folgende Beispiel zeigt ein Szenario mit Token-Authentifizierung.

- 1 Sie erhalten gültige Benutzeranmeldedaten für Ihre Instanz von vRealize Operations Manager.
- 2 Sie senden zur Authentifizierung eine POST-Anforderung an den REST-Endpoint.

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

Der Anforderungstext enthält den Benutzernamen, das Kennwort und die Authentifizierungsquelle.

- 3 Im Antworttext gibt der Endpoint das Token, das Ablaufdatum und die Uhrzeit an.
- 4 Während der folgenden Kommunikation geben Sie das Token-Objekt jeweils in folgendem Format in der Kopfzeile der Anforderung an:

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

Wenn Sie das Token von einer SSO-Quelle erworben haben, hat die Kopfzeile der Autorisierungsheader folgendes Format:

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

- 5 Sie können das Token vor dem Datum und der Uhrzeit, an dem es abläuft, ungültig machen, indem Sie eine POST-Anforderung an den Abmelde-Endpoint senden.

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

Anforderungskopfzeilen

Die folgenden HTTP-Kopfzeilen sind in der Regel in API-Anforderungen enthalten:

Accept-Language	<p>Mit der Anforderungskopfzeile <code>Accept-Language</code> können Sie angeben, in welcher Sprache die Antwort sein soll. Die Zeichenfolgen der Nachrichten in <code>ErrorResponse</code>-Antworten sind lokalisiert. Mit der folgenden Kopfzeile können Sie eine Antwort anfordern, deren Nachrichten in französischer Sprache lokalisierte Zeichenfolgen enthalten:</p> <pre>Accept-Language: fr-FR</pre>
Autorisierung	<p>Alle Anforderungen zur Erstellung einer API-Sitzung müssen eine <code>Authorization</code>-Kopfzeile des von dem Identitätsanbieter, den Ihre Organisation verwendet, vorgegebenen Formulars enthalten.</p>
Content-Type	<p>Anforderungen mit einem Text müssen eine entsprechende <code>Content-Type</code>-Kopfzeile in HTTP enthalten.</p> <ul style="list-style-type: none"> ■ Bei einem Anforderungstext in XML muss die Kopfzeile <code>Content-Type: application/xml</code> enthalten. ■ Bei einem Anforderungstext in JSON muss die Kopfzeile <code>Content-Type: application/json</code> enthalten.
Akzeptieren	<p>Um das gewünschte Antwortformat anzugeben, können Sie die Anforderung mit der Kopfzeile <code>Accept</code> versehen.</p> <ul style="list-style-type: none"> ■ Bei einer Antwort in XML muss die Kopfzeile <code>Accept: application/xml</code> enthalten. ■ Bei einer Antwort in JSON muss die Kopfzeile <code>Accept: application/json</code> enthalten.

Anforderungstext in XML

Bei einem in XML geschriebenen Anforderungstext validiert vRealize Operations Manager mit einem XML-Parser, ob die Reihenfolge und Anzahl der Elemente in einem Anforderungstext mit dem Schema übereinstimmt. Anforderungstexte werden als ungültig zurückgewiesen, wenn sie folgende Kriterien nicht erfüllen:

- Für alle durch Elemente in der Anforderung dargestellten Namensräume müssen XML-Namensraumattribute angegeben werden.
- Wenn in der Anforderung mehrere Namensräume dargestellt werden, müssen die XML-Namensraumattribute als Kennung ein Präfix enthalten, das für alle Elemente dieses Namensraums verwendet wird.
- Alle erforderlichen Elemente müssen im Anforderungstext erscheinen. Für alle Elemente, die im Anforderungstext erscheinen, ist die im Schema vorgesehene Reihenfolge einzuhalten. Für den Inhalt der Elemente gelten die im Schema angegebenen Typenbeschränkungen.

vRealize Operations Manager-API – REST-Antworten

Alle Antworten enthalten einen HTTP-Statuscode und, es sei denn, der Statuscode lautet 204 (kein Inhalt), eine `Accept`-Kopfzeile. Der Inhalt der Antwort hängt von der Anforderung ab. Manche Antworten enthalten einen Dokumententext, manche nur eine URL, und manche sind leer.

HTTP-Antwortcodes

Ein API-Client kann in einer Antwort eine Teilmenge von HTTP-Statuscodes erwarten.

Tabelle 2-1. Von der API ausgegebene HTTP-Statuscodes

Statuscode	Statusbeschreibung
200 OK	Die Anforderung ist gültig und wurde fertiggestellt. Die Antwort umfasst einen Dokumententext.
201 Created	Die Anforderung ist gültig. Das angeforderte Objekt wurde erzeugt und ist unter der in der Location-Kopfzeile angegebenen URL zu finden.
202 Accepted	Die Anforderung ist gültig, und es wurde eine Aufgabe erstellt, um Sie zu bearbeiten. Diese Antwort wird in der Regel von einem <code>TaskStatus</code> -Element begleitet.
204 No Content	Die Anforderung ist gültig und wurde abgeschlossen. Die Antwort enthält keinen Text.
400 Bad Request	Der Anforderungstext ist fehlerhaft, unvollständig oder aus anderen Gründen ungültig.
401 Unauthorized	Die Anmeldung ist fehlgeschlagen, oder die Gültigkeit des Authentifizierungs-Token ist abgelaufen.
403 Forbidden	Der Benutzer ist nicht authentifiziert oder hat keine ausreichende Berechtigung für den Zugriff auf ein oder mehrere in der Anforderung angegebene Objekte.
404 Not Found	Das in der Anforderung angegebene Objekt wurde nicht gefunden.
405 Method Not Allowed	Die in der Anforderung angegebene HTTP-Methode wird für dieses Objekt nicht unterstützt.
406 Not Acceptable	Die durch die Anforderung benannte Ressource kann keine Antwort von dem in der <code>Accept</code> -Kopfzeile der Anforderung angegebenen Typ generieren.
415 Unsupported Media Type	Die durch die Anforderung benannte Ressource unterstützt keine Anforderung mit dem angegebenen <code>Content-Type</code> und der angegebenen HTTP-Methode.
422 Not Found	Zeigt in der Regel an, dass die URL oder der Text der Anforderung fehlerhaft ist.
429 Too Many Requests	Ein Client hat zu viele Anforderungen gesendet, oder mehrere Clients senden zu viele gleichzeitige Anforderungen, die der Server aufgrund von Geschwindigkeitsbeschränkungen nicht verarbeiten kann. Um das Problem zu umgehen, versuchen Sie später, die Anforderung noch einmal senden.
500 Internal Server Error	Die Anforderung ist eingegangen, konnte aber wegen eines internen Serverfehlers nicht abgeschlossen werden.

Tabelle 2-1. Von der API ausgegebene HTTP-Statuscodes (Fortsetzung)

Statuscode	Statusbeschreibung
503 Service Unavailable	Der Server kann die Anforderung derzeit wegen einer temporären Bedingung nicht bearbeiten, z. B. weil die Ressourcen ausgeschöpft sind oder der Server gewartet wird.
504 Gateway Timeout	Der Server hat in seiner Funktion als Gateway bzw. Proxy keine rechtzeitige Antwort von dem durch die Anforderungs-URL angegebenen vorgelagerten Server erhalten.

Verwenden der API mit vRealize Operations Manager

Anforderungen versenden und Antworten empfangen können Sie mit einem Browser oder einem HTTP-Clientprogramm.

REST-Clientprogramme

Jede Clientanwendung, die HTTPS-Anforderungen versenden kann, ist ein geeignetes Tool zur Entwicklung von REST-Anwendungen mit der vRealize Operations Manager-API. Für die meisten Browser und viele IDEs stehen REST-Clients als Plug-ins zur Verfügung. Folgende Open-Source-Programme werden häufig verwendet:

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

Außerdem stellt VMware sprachspezifische Client-Bindings für die vRealize Operations Manager-API bereit. Weitere Informationen hierzu finden Sie unter [Informationen zur Schemareferenz](#).

Informationen zur Schemareferenz

Die Dokumentation der REST API für vRealize Operations Manager enthält Referenzmaterial für alle Elemente, Typen, Abfragen und Vorgänge in der vRealize Operations Manager-API. Sie enthält auch die Dateien mit den Schemadefinitionen.

Die Swagger-basierte API-Dokumentation ist zusammen mit dem Produkt erhältlich, mit der Möglichkeit, REST-API-Aufrufe direkt von der Startseite aus durchzuführen.

Verwenden Sie für den Zugriff auf die API-Dokumentation die URL Ihrer vRealize Operations Manager-Instanz. Beispiel: Wenn die URL Ihrer Instanz `https://vrealize.example.com` ist, finden Sie die API-Referenz unter: `https://vrealize.example.com/suite-api/doc/swagger-ui.html`.

Sprachspezifische Client-Bindings finden Sie unter:

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

Informationen zu den API-Beispielen für vRealize Operations Manager

Alle Beispiele enthalten HTTP-Anforderungen und HTTP-Antworten. Diese Beispiele zeigen den mit Vorgängen, wie z. B. der Erzeugung von Objekten in Ihrer überwachten Umgebung und der Abfrage diesbezüglicher Informationen, verbundenen Workflow und Inhalt.

Die Texte der Beispielanforderungen sind in JSON. Die von der vRealize Operations Manager-API verlangten Kopfzeilen der Anforderungen sind in den Beispielanforderungen enthalten, die keine Fragmente eines größeren Beispiels sind.

In den meisten Beispiellantworten werden nur die für den jeweils behandelten Vorgang relevanten Elemente und Attribute angezeigt. Die Auslassungspunkte (...) zeigen an, dass Inhalte im Antworttext ausgelassen wurden.

Erste Schritte mit der API

3

API-Clients und vRealize Operations Manager -Server kommunizieren über HTTPS und tauschen XML-Darstellungen von API-Objekten aus.

Dieses einfache Beispiel für den Workflow einer REST API zeigt, wie Sie eine Liste aller Metriken für einen Objekttyp der virtuellen Maschine abrufen können, der in der Modelldefinition des VMware vCenter®-Adapters enthalten ist. Mithilfe der API können Sie für jeden Objekttyp die vollständige Auflistung aller verfügbaren Metriken abrufen.

Dieses Kapitel enthält die folgenden Themen:

- [Erwerben eines Authentifizierungs-Token](#)
- [Auffinden des Adapter- und Objekttyps](#)
- [Generieren einer Liste aller Metriken für das Objekt](#)

Erwerben eines Authentifizierungs-Token

vRealize Operations Manager erfordert, dass-API-Anforderungen authentifiziert werden. Im ersten Schritt dieses Workflows rufen Sie ein Authentifizierungs-Token ab.

Um einen Authentifizierungs-Token abzurufen, werden bei der Login-Anfrage die Anmeldedaten des Benutzers in der zur HTTP-Basisauthentifizierung erforderlichen Form angegeben. In diesem Beispiel meldet sich der Benutzer bei einer Instanz von vRealize Operations Manager mit der URL `https://vrealize.example.com/ an`.

Hinweis Im Beispiel wird zur Authentifizierung ein Token verwendet. Weitere Informationen zu Authentifizierungsmechanismen finden Sie unter [Sicherheit](#).

Mithilfe von `authSource` können Sie Benutzer und Benutzergruppeninformationen importieren und authentifizieren, die sich auf einer anderen Maschine befinden. Sie können beispielsweise Benutzer aus LDAP, Active Directory, VMware Identity Manager, Single Sign-On usw. authentifizieren. Beim Importieren von Benutzerkontoinformationen, die sich auf einer anderen Maschine befindet, müssen die Kriterien für den Import der Benutzerkonten aus der Quellmaschine definiert werden.

Nach dem Erstellen einer Authentifizierungsquelle können Sie sie zum Abrufen eines Tokens verwenden, indem Sie den Namen eingeben. „LOKAL“ wird als Standardtyp der Authentifizierungsquelle verwendet.

Voraussetzungen

- Sichern Sie einen Kanal zwischen dem Webbrowser und dem Server von vRealize Operations Manager. Öffnen Sie einen Browser und geben Sie die URL einer Instanz von vRealize Operations Manager ein, z. B.:

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

Das System warnt Sie, dass Ihre Verbindung nicht privat ist. Klicken Sie sich durch die Bestätigung der Sicherheitsausnahme und die Festlegung eines SSL-Handshake.

- Vergewissern Sie sich, dass Sie Zugriff auf die APIs haben. Geben Sie die URL Ihrer Instanz von vRealize Operations Manager ein und fügen Sie am Ende `suite-api/docs/rest/index.html` hinzu, z. B.:

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

- Vergewissern Sie sich, dass Sie Anmeldedaten für einen Benutzer Ihrer Instanz von vRealize Operations Manager haben.

Verfahren

- 1 Senden Sie eine Anforderung an die Login-URL, um ein Token zu erwerben.

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

Weitere Informationen hierzu finden Sie unter [Login-Anforderung und Antwort](#).

- 2 Sehen Sie sich die Antwort an.

Wenn die Anforderung erfolgreich war, wird ein Autorisierungs-Token für `ops` ausgegeben, den Sie allen folgenden API-Anforderungen beifügen müssen.

Beispiel: Login-Anforderung und Antwort

Dieses Beispiel zeigt eine Anforderung und eine Antwort für einen Benutzer mit dem Anmeldenamen **vRealize** und dem Kennwort **vRealize-dummy-password**.

Kopfzeile der Anforderung:

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

Anforderungstext im Format JSON:

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

Antwort in JSON:

```
200 OK
```

```
{
  "token": "8f868cca-27cc-43d6-a838-c5467e73ec45::77cea9b2-1e87-490e-b626-e878beaa23b",
  "validity": 1470421325035,
  "expiresAt": "Friday, August 5, 2016 6:22:05 PM UTC",
  "roles": []
}
```

Der Antwortcode gibt an, ob die Anforderung erfolgreich war oder (falls sie fehlgeschlagen ist) worin der Fehler bestand.

- Wenn die Anforderung erfolgreich ist, gibt der Server den HTTP-Antwortcode 200 (OK) und einen wiederverwendbaren Autorisierungs-Token `ops`, der nach sechs Stunden abläuft, aus. Dieser Token muss in jeder nachfolgenden-API-Anforderung enthalten sein.
- Wenn die Kopfzeile der Autorisierung für die Anforderung fehlt, gibt der Server den HTTP-Antwortcode 403 aus.
- Wenn die in der Kopfzeile der Autorisierung angegebenen Anmeldedaten ungültig sind, gibt der Server den HTTP-Antwortcode 401 aus.

Auffinden des Adapter- und Objekttyps

Ihre Instanz von vRealize Operations Manager enthält mehrere Adaptertypen. Um den Adaptertyp für den vCenter-Adapter zu finden, rufen Sie mit einer GET-Anforderung eine Liste aller Adaptertypen auf. Die API-Antwort umfasst alle Objekttypen, die vom Adapter überwacht werden.

Voraussetzungen

Vergewissern Sie sich, dass Sie bei der Instanz von vRealize Operations Manager angemeldet sind.

Verfahren

- 1 Erzeugen Sie eine GET-Anforderung für alle Adaptertypen.

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

- 2 Suchen Sie in der Antwort nach dem vCenter-Adapter und der Liste mit überwachten Objekttypen.

Weitere Informationen hierzu finden Sie im Antwortteil von [Ermitteln des Adapter- und Objekttyps für den vCenter-Adapter](#).

Beispiel: Ermitteln des Adapter- und Objekttyps für den vCenter-Adapter

In diesem Beispiel werden der Adaptertyp für den vCenter-Adapter und alle in der Definition des Adaptermodells enthaltenen Objekttypen ermittelt.

Kopfzeile der Anforderung:

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

Wobei *vROps_token* das Token ist, das Sie der Antwort in [Erwerben eines Authentifizierungstoken](#) entnommen haben.

Ausschnitt der Antwort in JSON für den vCenter-Adapter:

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"
  ],
  ...
}
```

Für den vCenter-Adapter ist der `adapter-kind`-Schlüssel `VMware`. Die `resourceKinds` sind die Objekttypen, die der vCenter-Adapter überwacht. Für den Objekttyp „virtuelle Maschine“ ist `resourceKinds VirtualMachine`.

Generieren einer Liste aller Metriken für das Objekt

Um eine Liste aller Metriken für eine im vCenter-Adaptermodell definierte virtuelle Maschine zu generieren, senden Sie eine GET-Anforderung an die URL mit dem Adapter- und dem Objekttyp.

Voraussetzungen

Vergewissern Sie sich, dass die folgenden Voraussetzungen erfüllt sind:

- Sie sind bei der Instanz von vRealize Operations Manager angemeldet.
- Sie kennen den `adapterKind`-Wert für den vCenter-Adapter und den `resourceKinds`-Wert für die virtuelle Maschine. Siehe [Ermitteln des Adapter- und Objekttyps für den vCenter-Adapter](#)

Verfahren

- 1 Erzeugen Sie eine GET-Anforderung, um die Metadaten für Metriken abzurufen.

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

- 2 Vergleichen Sie die in der Antwort aufgeführten mit den in der Benutzeroberfläche angezeigten Metriken. Siehe [Metriken für die virtuelle Maschine aus der API und in der Benutzeroberfläche](#)

Beispiel: Metriken für die virtuelle Maschine aus der API und in der Benutzeroberfläche

Dieses Beispiel gleicht die in der XML-Antwort aufgelisteten Metriken der virtuellen Maschine mit den in der Benutzeroberfläche von vRealize Operations Manager angezeigten Metriken ab.

Anforderung:

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

Wobei:

- `VMware` der `adapterKindKey` ist.
- `VirtualMachine` der `resourceKindKey` ist.
- `vROps_token` das Token ist, das Sie der Antwort in [Erwerben eines Authentifizierungstoken](#)

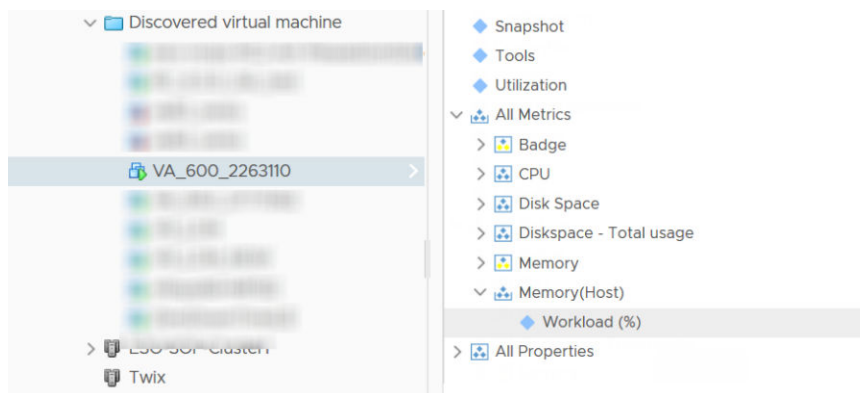
Ausschnitt aus der Antwort in JSON:

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
    },
    ...
  ]
}
```

Jedes `resourceTypeAttribute` in der Antwort ist eine Metrik mit Metadaten für ein Objekt der virtuellen Maschine. Der `name` entspricht dem in der Benutzeroberfläche von vRealize Operations Manager angezeigten Text. In diesem Beispiel listet der Ausschnitt Metriken für den Arbeitsspeicher und die Host-Arbeitslast auf.

Um Metriken in der Antwort mit Metriken in der-Benutzeroberfläche zu vergleichen, melden Sie sich bei der Instanz von vRealize Operations Manager an, die auf `vrealize.example.com` läuft, und navigieren zu den Metriken für eine virtuelle Maschine. Das folgende Beispiel zeigt, wo Sie Metriken für den Arbeitsspeicher (Host) und die Arbeitslast finden.



Das Beispiel zeigt, wie Sie Metriken für den Objekttyp der virtuellen Maschine abrufen. Um Metriken für andere Objekttypen abzurufen, ersetzen Sie `VirtualMachine` in der GET-Anforderung durch andere `resourceKinds`.

Konfigurieren einer Adapterinstanz

4

Nach der Installation einer Lösung, die ein Management-Paket mit einem Adapter beinhaltet, müssen Sie eine Adapterinstanz konfigurieren, um Daten aus den Objekten in der Modelldefinition des Adapters zu erfassen. Sie können zur Konfiguration einer Adapterinstanz die vRealize Operations Manager -API verwenden.

In diesem Anwendungsbeispiel wird beschrieben, wie Sie eine Adapterinstanz für eine vSphere-Lösung konfigurieren. Es beinhaltet Folgendes:

- Übersicht der Vorgänge, jeweils mit Anforderung, Anforderungstext und Antwort
- Spezifische Vorgehensweise für jeden Vorgang

Dieses Kapitel enthält die folgenden Themen:

- [Übersicht über die Konfiguration von Adapterinstanzanforderungen](#)
- [Ermitteln der Lösung und ihrer Adapter](#)
- [Ermitteln, welche Objekttypen für einen Adapter benötigt werden](#)
- [Erstellen der Adapterinstanz](#)
- [Gültigkeit eines Zertifikats nachweisen](#)
- [Starten der Überwachung einer neuen Adapterinstanz](#)

Übersicht über die Konfiguration von Adapterinstanzanforderungen

Sie erzeugen sequenzielle API-Anforderungen, um eine Adapterinstanz zu konfigurieren. Die Antworten auf die vorherigen Anforderungen enthalten Informationen, die Sie für nachfolgende Anforderungen benötigen.

Tabelle 4-1. Anforderungsübersicht

Vorgang	Anforderung	Anforderungstext	Antwort
Alle für das Produkt registrierten Lösungen abrufen und die Adaptertypen ermitteln	GET <API-URL>/suite-api/api/solutions	Keine	adapterkindkeys
Rufen Sie alle Objekttypen für einen bestimmten Adaptertyp ab.	GET <API-URL>/suite-api/api/adapterkinds/{key}/resourcekinds	Keine	resourceIdentifierTypes
Ein Objekt für eine Adapterinstanz erzeugen	POST <API-URL>/suite-api/api/adapters	Werte für resourceIdentifiers und credential	adapterid
Eine Adapterinstanz patchen, um das vorgelegte Zertifikat anzuerkennen	PATCH <API-URL>/suite-api/api/adapters	API-Antwort auf POST <API-URL>/suite-api/api/adapters	API-Antwort auf POST <API-URL>/suite-api/api/adapters ohne adapter-certificates
Mit Überwachung des Adapters beginnen	PUT <API-URL>/suite-api/api/adapters/{adapterid}/monitoringstate/start	Keine	200 OK

Ermitteln der Lösung und ihrer Adapter

Für Ihre Instanz von vRealize Operations Manager sind möglicherweise mehrere Lösungen installiert. Um die vSphere-Lösung und ihre Adaptertypen zu finden, erstellen Sie eine GET-Anforderung, mit der Sie eine Liste aller Lösungen abrufen. Die Antwort umfasst alle Adapter, die in der-Lösung enthalten sind.

Für dieses Anwendungsbeispiel ist die *API-URL* für die Instanz von vRealize Operations Manager `vrealize.example.com`.

Voraussetzungen

Vergewissern Sie sich, dass Sie sich bei der-API-URL für eine Instanz von vRealize Operations Manager anmelden können. Weitere Informationen hierzu finden Sie unter [Erwerben eines Authentifizierungstoken](#).

Verfahren

- 1 Erstellen Sie eine GET-Anforderung, um eine Liste aller Lösungen zu erhalten.

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

- 2 Sehen Sie sich die Antwort an, um die vSphere-Lösung und ihre Adaptertypen zu finden.
Mehr darüber erfahren Sie im Antwortteil von [Adaptertypen für die vSphere Lösung](#).

Beispiel: Adaptertypen für die vSphere Lösung

In diesem Beispiel werden alle installierten Lösungen und die jeweiligen Adaptertypen aufgelistet.

Kopfzeile der Anforderung:

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

Die Antwort in 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"
      ]
    }
  ]
}
```

Die Antwort zeigt drei installierte Lösungen:

- Management-Paket für die Lösung Log Insight
- Lösung End Point Operations
- Lösung vSphere

Die Lösung vSphere hat zwei Adaptertypen:

- VMWARE
- PythonRemediationVcenterAdapter

Der Adaptertyp für den vCenter-Adapter ist VMWARE.

Ermitteln, welche Objekttypen für einen Adapter benötigt werden

Nachdem Sie festgestellt haben, dass Sie eine Instanz des vCenter-Adapters erstellen möchten, müssen Sie herausfinden, welche Objekttypen für diesen Adapter benötigt werden. Sie rufen mit einer GET-Anforderung eine Liste aller Objekttypen für den vCenter-Adapter ab.

Voraussetzungen

Vergewissern Sie sich, dass Sie den Adaptertyp für den vCenter-Adapter kennen.

Verfahren

- 1 Rufen Sie mit einer GET-Anforderung eine Liste aller Objekttypen für den vCenter-Adapter ab.

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

- 2 Lesen Sie die Antwort, um herauszufinden, welche Objekttypen benötigt werden.

Mehr darüber erfahren Sie im Antwortteil von [Für den vCenter-Adapter benötigte Objekttypen](#).

Beispiel: Für den vCenter-Adapter benötigte Objekttypen

In diesem Beispiel werden alle Objekttypen für den vCenter-Adapter ermittelt.

Kopfzeile der Anforderung:

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

Ausschnitt der Antwort in 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
    }
  ],
  ...
}

```

Dieser Ausschnitt zeigt den Ressourcentyp mit dem Attribut "resourceKindType":

"ADAPTER_INSTANCE". Jeder Objekttyp mit dem Ressourcenbezeichner

"isPartOfUniqueness":true benötigt einen Wert für diesen Objekttyp mit der API-Anforderung zur Erzeugung der Adapterinstanz.

Für eine Adapterinstanz des vCenter-Adapters wird ein Wert für VCURL oder die URL des vCenter benötigt.

Erstellen der Adapterinstanz

Nachdem Sie die für den Adapter benötigten Objekttypen ermittelt haben, geben Sie Parameterwerte für die Objekttypen an, um eine Adapterinstanz zu erstellen. Ihre POST-Anforderung beinhaltet einen Anforderungstext mit den benötigten Parametern.

Bei der Erstellung einer Adapterinstanz ist die Einstellung VCURL obligatorisch.

Voraussetzungen

Vergewissern Sie sich, dass Sie über die IP-Adresse eines vCenter und die dazugehörigen Anmeldedaten verfügen.

Verfahren

- 1 Erstellen Sie eine POST-Anforderung, um die Adapterinstanz zu erstellen.

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

- 2 Sehen Sie sich die Antwort an, um den Namen für die vSphere-Lösung und ihre Adaptertypen zu finden.

Mehr darüber erfahren Sie im Antwortteil von [Adapterinstanz](#).

Beispiel: Adapterinstanz

In diesem Beispiel wird die Adapterinstanz für ein vCenter mit den folgenden Parametern erstellt:

- Anzeigename: VC-Adapterinstanz
- Beschreibung: eine vCenter-Adapterinstanz für VC 12.345.678.9
- IP-Adresse des vCenter-Servers: https://12.345.678.9
- Anmeldename: VC-Credential-1
- Benutzername: administrator@vsphere.local
- Kennwort: VC-dummy-passwd

AUTODISCOVERY and PROCESSCHANGEEVENTS sind optional, werden aber als zusätzliche Beispiele für Ressourcenkennungen im Anforderungstext und in der Antwort einbezogen.

Kopfzeile der Anforderung:

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

Anforderungstext im 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"
        }
    ]
},
},
}

```

Ausschnitt der Antwort in 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"
}

```

Die API erstellt einen neuen Adapter mit einer intern generierten UUID, die das Objekt eindeutig identifiziert. Die API-Antwort enthält die von vCenter 12.345.678.9 bereitgestellten Zertifikate. Der Wert der Adapterinstanz-ID dient zum Starten der Überwachung und zur Datenerfassung.

Gültigkeit eines Zertifikats nachweisen

Bevor vRealize Operations Manager eine Verbindung zum vCenter herstellen und mit der Datenerfassung beginnen kann, muss überprüft werden, ob die von der Adapterinstanz erkannten Datenquellen gültige Zertifikate vorlegen. Ihre PATCH-Anforderung liefert den Nachweis, dass die Zertifikate gültig sind, weil sie einen Anforderungstext mit der Antwort auf die zur Erzeugung des Adapters verwendete POST-Anforderung enthält.

Voraussetzungen

Vergewissern Sie sich, dass Sie die Antwort auf die zur Erzeugung des Adapters verwendete POST-Anforderung haben. Sie finden die Antwort unter [Adapterinstanz](#).

Verfahren

- ◆ Erzeugen Sie eine PATCH-Anforderung, um das System darüber zu informieren, dass der Benutzer das vom vCenter vorgelegte Zertifikat akzeptiert hat.

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

Beispiel: Validierung des Zertifikats

In diesem Beispiel ist der Anforderungstext für die PATCH-Anforderung identisch mit der Antwort auf die zur Erzeugung der Adapterinstanz verwendete POST-Anforderung.

Kopfzeile der Anforderung:

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

Anforderungstext im 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"
}

```

Antwort in 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"
}

```

Die Antwort ist identisch mit dem Anforderungstext ohne den Abschnitt `adapter-certificates`.

Starten der Überwachung einer neuen Adapterinstanz

Nach der Erzeugung der Adapterinstanz und der Konfiguration von vRealize Operations Manager zur Ermittlung eines gültigen Zertifikats beginnen Sie mit der Datenüberwachung und -erfassung. Mit Ihrer PUT-Anforderung erhalten Sie die UUID der zur Ermittlung neuer Objekte verwendeten Adapterinstanz.

Voraussetzungen

Vergewissern Sie sich, dass Sie die UUID der zur Ermittlung neuer Objekte verwendeten Adapterinstanz haben. Sie finden die Antwort unter [Adapterinstanz](#).

Verfahren

- ◆ Erzeugen Sie eine PUT-Anforderung, um die Überwachung mit der neuen Adapterinstanz zu beginnen.

```

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

```

Beispiel: Objekte erkennen und Daten erfassen

In diesem Beispiel startet die Überwachung des Adapters mit der Adapterinstanz-ID der PUT-Anforderung, die die Adapterinstanz erzeugt hat.

Kopfzeile der Anforderung:

```

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

```