

vRealize Operations Manager API 프로그래밍 가 이드

2021년 5월 19일

vRealize Operations Manager 8.1

다음 VMware 웹 사이트에서 최신 기술 문서를 확인할 수 있습니다.

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

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

VMware 코리아
서울시 강남구
영동대로 517
아셈타워 13층
(우) 06164
전화: +82 2 3016 6500
팩스: +82 2 3016 6501
www.vmware.com/kr

목차

1	이 프로그래밍 가이드에 대한 정보	4
2	vRealize Operations Manager API 이해	5
	vRealize Operations Manager API의 작동 방식	5
	클라이언트 워크플로 개요	7
	vRealize Operations Manager API REST 요청	8
	vRealize Operations Manager API REST 응답	10
	vRealize Operations Manager로 API 사용	11
3	API 시작	13
	인증 토큰 획득	13
	어댑터 유형 및 개체 유형 찾기	15
	개체에 대한 모든 메트릭 목록 생성	16
4	어댑터 인스턴스 구성	19
	어댑터 인스턴스 구성 요청 요약	19
	솔루션 및 해당 어댑터 식별	20
	어댑터에 필요한 개체 유형 식별	22
	어댑터 인스턴스 생성	23
	인증서 유효성 증명 제공	27
	새 어댑터 인스턴스 모니터링 시작	30

이 프로그래밍 가이드에 대한 정보

1

vRealize Operations Manager API 프로그래밍 가이드에서는 REST API 리소스를 사용하고, 인증하고, REST API 호출을 생성하는 방법 등 vRealize Operations Manager REST API에 대한 정보를 제공합니다.

대상 사용자

이 정보는 vRealize Operations Manager REST API를 사용하여 프로그래밍 방식으로 vRealize Operations Manager를 구성하고 관리하려는 관리자 및 프로그래머를 대상으로 합니다. 이 가이드는 일반적인 사용 사례를 중점적으로 다룹니다.

vRealize Operations Manager API 이해

2

개발자는 API를 사용하여 vRealize Operations Manager의 대화형 클라이언트를 구축할 수 있습니다. API는 REST 스타일을 따르며 라이선스가 부여된 모든 사용자가 사용할 수 있습니다.

vRealize Operations Manager 클라이언트는 HTTP를 통해 서버와 통신하여 vRealize Operations Manager 개체의 표현을 교환합니다. 이러한 표현은 JSON 또는 XML 요소 형식입니다. HTTP GET 요청을 사용하여 개체의 현재 표현을 검색하고, HTTP POST 및 PUT 요청을 사용하여 개체를 생성 또는 수정하며 HTTP DELETE 요청을 사용하여 개체를 삭제합니다.

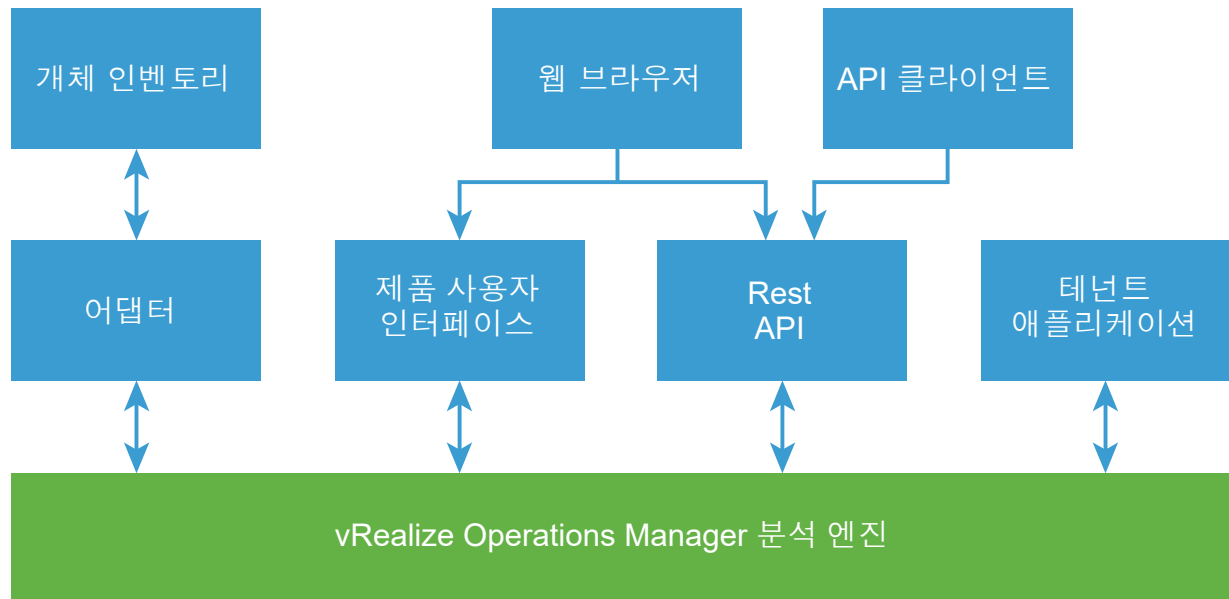
본 장은 다음 항목을 포함합니다.

- [vRealize Operations Manager API의 작동 방식](#)
- [클라이언트 워크플로 개요](#)
- [vRealize Operations Manager로 API 사용](#)

vRealize Operations Manager API의 작동 방식

웹 브라우저를 사용하여 제품 사용자 인터페이스 또는 API 호출을 통해 vRealize Operations Manager 분석 엔진과 통신합니다.

그림 2-1. vRealize Operations Manager 간소화된 아키텍처



어댑터 인스턴스는 모니터링되는 환경의 개체에서 데이터를 수집합니다. vRealize Operations Manager 분석 엔진은 데이터를 처리하고 그래픽 인터페이스에 전체 모델을 표시합니다.

API 사용 이유

API는 새 vRealize Operations Manager 사용자에게 대한 액세스 제어를 구성하는 것과 같은 동일 작업 반복과 같은 잘 정의된 워크플로를 자동화해야 할 때 가장 유용합니다. API는 가상 환경의 특정 자산에 대한 데이터 검색과 같은 vRealize Operations Manager 데이터 저장소에 대한 쿼리를 수행하는 경우에도 유용합니다. 또한 API를 사용하여 vRealize Operations Manager 데이터 저장소에서 모든 데이터를 추출하고 별도의 분석 시스템으로 로드할 수 있습니다.

vRealize Operations Manager 용어

어댑터의 개체를 설명하는 데 사용하는 XML 구문은 API 코드 구문에 해당하지만 사용자 인터페이스에서 찾은 것과는 다릅니다. 사용자 인터페이스에는 다음 용어가 표시됩니다. 각 용어에 대한 설명은 API 호출에 사용되는 해당 XML 구문에 포함되어 있습니다.

어댑터 유형 특정 개체 유형을 검색하는 데 사용되는 어댑터를 정의합니다. 예를 들어 vCenter 어댑터는 vSphere 데이터센터에 연결된 개체를 검색합니다. AWS 어댑터는 AWS 서비스 및 개체를 검색합니다.

XML 구문: `adapterkinds`입니다.

개체 유형 개체 또는 정보 소스를 나타내는 엔티티의 클래스입니다. 개체는 vRealize Operations Manager 분석 엔진에 데이터를 보고합니다. 가상 시스템, 데이터스토어 및 호스트 시스템은 vCenter 어댑터 모델에서 정의된 개체 유형의 예입니다.

XML 구문: `resourcekinds`입니다.

클라이언트 워크플로 개요

vRealize Operations Manager API 클라이언트는 REST 워크플로를 구현하여 서버에 대한 HTTP 요청을 수행하고 서버의 응답에서 해당 요청이 필요로 하는 정보를 검색합니다.

REST 정보

REST(Representational State Transfer)는 HTTP(Hypertext Transfer Protocol)를 사용하여 클라이언트와 서버 간에 개체의 직렬화된 표현을 교환하는 프로그램의 아키텍처 스타일 특성을 설명합니다.

vRealize Operations Manager API에서 이러한 표현은 JSON 또는 XML 문서입니다.

REST 워크플로에서 개체의 표현은 단일 요청 또는 응답에 표시되는 것 이외의 당사자가 개체에 대해 어떠한 항목도 인식하지 못하는 것을 명시적으로 가정하고 클라이언트와 서버 간에 양방향으로 전달됩니다. 이러한 문서를 사용할 수 있는 URL은 이러한 문서가 포함된 요청 또는 응답의 수명 이상으로 지속되곤 합니다.

REST API 워크플로

REST API를 사용하도록 작성된 애플리케이션 프로그램은 API가 정의하는 개체를 생성, 검색, 업데이트 또는 삭제하는 원격 프로시저 호출을 수행하기 위해 스크립트나 기타 상위 언어로 자주 실행되는 HTTP 요청을 사용합니다. vRealize Operations Manager REST API에서 이러한 개체는 XML 스키마의 수집으로 정의됩니다. 작업 자체는 HTTP 요청이므로 모든 HTTP 클라이언트에 대해 일반적입니다.

REST API 클라이언트 애플리케이션을 작성하려면 HTTP 프로토콜, JSON 또는 XML의 의미, vRealize Operations Manager API에서 사용하는 전송 형식 등을 이해해야 합니다. 이러한 클라이언트에서 API를 효과적으로 사용하려면 다음 개념을 숙지해야 합니다.

- API에서 지원하는 개체 집합과 이러한 개체가 나타내는 것
- API가 이러한 개체를 나타내는 방식
- 클라이언트가 운영하려는 개체를 참조하는 방법

API 참조에는 전체 API 요청의 목록이 포함되어 있습니다. [스키마 참조 정보](#)를 참조하십시오.

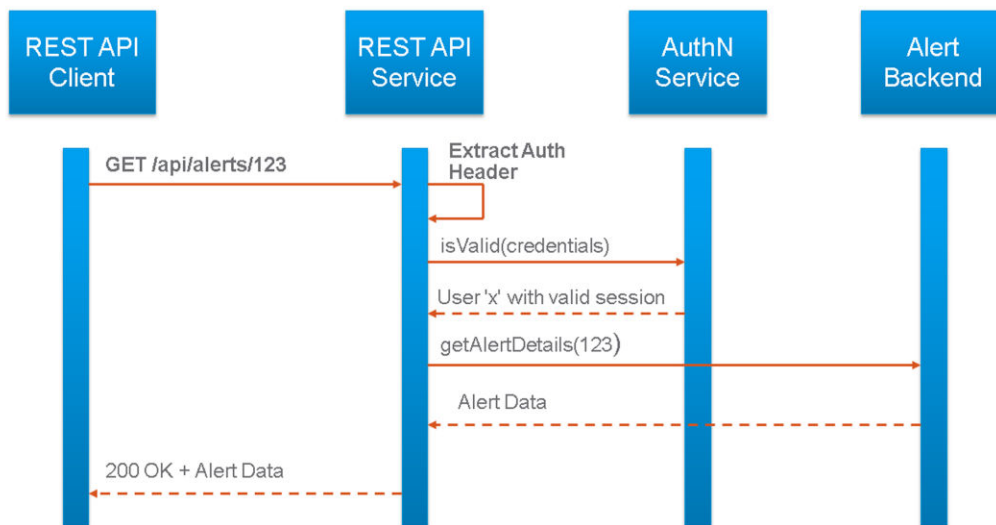
vRealize Operations Manager API REST 요청

개체 표현을 검색하려면 클라이언트는 개체 참조에 대해 HTTP 요청을 수행합니다.

보안

API 클라이언트와 서버 간의 HTTP 링크는 SSL을 사용하여 구축됩니다. API 클라이언트는 서버와 통신하도록 토큰 기반 인증을 구성합니다.

그림 2-2. 시나리오: ID 123의 경고에 대한 세부 정보를 가져오기 위해 사용자 자격 증명 제공



토큰 기반 인증을 사용하면 vRealize Operations Manager API 서버에 로그인 요청을 POST하고 유효한 사용자 자격 증명을 제공하여 인증 토큰을 얻을 수 있습니다. 다음 예에서는 토큰 기반 인증 시나리오를 제공합니다.

- 1 vRealize Operations Manager 인스턴스에 대한 유효한 사용자 자격 증명을 얻습니다.
- 2 인증을 위해 REST 끝점에 요청을 POST합니다.

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

요청 본문에는 사용자 이름, 비밀번호 및 인증 소스가 포함됩니다.

- 3 응답 본문에서 끝점은 토큰, 만료 날짜 및 시간을 반환합니다.
- 4 추가 통신을 위해 토큰 개체를 인증 헤더에 다음 형식으로 포함합니다.

```
Authorization: vRealizeOpsToken <vOps_token>
```

또는 SSO 소스에서 토큰을 얻은 경우 인증 헤더의 형식은 다음과 같습니다.

```
Authorization: SS02Token <SSO_SAML_TOKEN>
```

- 5 로그아웃 끝점에 POST 요청을 전송하여 만료 날짜 및 시간 전에 토큰을 무효화할 수 있습니다.

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

요청 헤더

다음 HTTP 헤더는 일반적으로 API 요청에 포함되어 있습니다.

수락-언어 응답에 원하는 언어를 지정하려면 **Accept-Language** 요청 헤더를 사용합니다. **ErrorType** 응답의 메시지 문자열은 현지화됩니다. 프랑스어로 현지화된 메시지 문자열로 응답을 요청하려면 다음 헤더를 사용합니다.

```
Accept-Language: fr-FR
```

권한 부여 API 세션을 생성하기 위한 모든 요청에는 조직에서 사용하는 ID 제공자에 의해 규정된 양식의 **Authorization** 헤더가 포함되어야 합니다.

컨텐츠-유형 본문을 포함하는 요청에는 적절한 **HTTP Content-Type** 헤더가 포함되어야 합니다.

- XML의 요청 본문의 경우 헤더에 **컨텐츠 유형**: 애플리케이션/XML이 포함되어야 합니다.
- JSON의 요청 본문의 경우 헤더에 **컨텐츠 유형**: 애플리케이션/JSON이 포함되어야 합니다.

허용 원하는 응답 형식을 지정하려면 **Accept** 요청 헤더를 포함합니다.

- XML의 응답의 경우 헤더에 **허용**: 애플리케이션/XML이 포함되어야 합니다.
- JSON의 응답의 경우 헤더에 **허용**: 애플리케이션/JSON이 포함되어야 합니다.

XML 형식의 본문 요청

XML로 작성된 요청 본문의 경우, vRealize Operations Manager는 요청 본문의 요소가 순서와 숫자에서 스키마와 호응하도록 요구하는 유효성 검사 XML 구문 분석을 사용합니다. 요청 본문은 다음 조건을 충족하지 않는 한 잘못된 것으로 거부됩니다.

- 요청에서 요소로 표시되는 모든 네임스페이스에 대해 XML 네임스페이스 특성을 제공해야 합니다.

- 요청에 네임스페이스가 여러 개 표시되는 경우, XML 네임스페이스 특성에 식별 접두사를 포함해야 하고 해당 접두사를 해당 네임스페이스의 모든 요소와 함께 사용해야 합니다.
- 모든 필수 요소가 요청 본문에 나타나야 합니다. 요청 본문에 표시되는 모든 요소는 스키마가 구축하는 순서와 스키마가 지정하는 유형 제약 조건을 준수하는 콘텐츠로 나타나야 합니다.

vRealize Operations Manager API REST 응답

모든 응답에는 HTTP 상태 코드가 포함되며, 상태 코드가 204(No Content)인 경우, 수락 헤더가 포함되지 않습니다. 응답 콘텐츠는 요청에 따라 다릅니다. 일부 응답에는 문서 본문이 포함되고 일부는 URL만 포함하고 일부는 비어 있습니다.

HTTP 응답 코드

API 클라이언트는 응답에서 HTTP 상태 코드의 하위 집합을 예상할 수 있습니다.

표 2-1. API가 반환하는 HTTP 상태 코드

상태 코드	상태 설명
200 OK	요청이 유효하며 완료되었습니다. 응답에 문서 본문이 포함되어 있습니다.
201 Created	요청이 유효합니다. 요청된 개체가 생성되었으며 위치 헤더에 지정된 URL에서 찾을 수 있습니다.
202 Accepted	요청이 유효하고 이를 처리하기 위해 작업이 생성되었습니다. 이 응답은 일반적으로 TaskStatus 요소와 함께 제공됩니다.
204 No Content	요청이 유효하며 완료되었습니다. 응답에 본문이 포함되지 않습니다.
400 Bad Request	요청 본문이 잘못된 형식이거나 불완전하거나 유효하지 않습니다.
401 Unauthorized	로그인 실패 또는 인증 토큰이 만료되었습니다.
403 Forbidden	사용자가 인증되지 않았거나 요청에 지정된 하나 이상의 개체에 액세스하기 위한 적절한 권한이 없습니다.
404 Not Found	요청에 지정된 개체를 찾을 수 없습니다.
405 Method Not Allowed	요청에 지정된 HTTP 메서드가 이 개체에 대해 지원되지 않습니다.
406 Not Acceptable	요청으로 식별된 리소스가 요청의 Accept 헤더에 지정된 유형에 대한 응답을 생성할 수 없습니다.
415 Unsupported Media Type	요청으로 식별된 리소스가 지정된 Content-Type 및 HTTP 메서드의 요청을 지원하지 않습니다.
422 Not Found	일반적으로 잘못된 형식의 요청 URL 또는 요청 본문을 나타냅니다.
429 Too Many Requests	클라이언트가 너무 많은 요청을 전송했거나 여러 클라이언트가 동시에 요청을 너무 많이 전송하고 있으므로 서버가 속도 제한으로 인해 이를 처리할 수 없습니다. 이 문제를 해결하려면 나중에 요청을 다시 전송합니다.

표 2-1. API가 반환하는 HTTP 상태 코드 (계속)

상태 코드	상태 설명
500 Internal Server Error	요청을 받았지만 서버에서 내부 오류가 발생하여 완료할 수 없습니다.
503 Service Unavailable	서버가 리소스 소모 또는 서버 유지 보수와 같은 일시적인 상태로 인해 현재 요청을 처리할 수 없습니다.
504 Gateway Timeout	서버가 게이트웨이 또는 프록시로 작동하는 동안 요청 URL에서 지정된 업스트림 서버에서 적시에 응답을 받지 못했습니다.

vRealize Operations Manager로 API 사용

브라우저 또는 HTTP 클라이언트 프로그램을 사용하여 요청을 보내고 응답을 수신할 수 있습니다.

REST 클라이언트 프로그램

HTTPS 요청을 보낼 수 있는 모든 클라이언트 애플리케이션은 vRealize Operations Manager API를 사용하여 REST 애플리케이션을 개발하기 위한 적절한 도구입니다. 대부분의 브라우저 및 여러 IDE에 REST 클라이언트 플러그인을 사용할 수 있습니다. 다음은 일반적으로 사용되는 오픈 소스 프로그램입니다.

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

또한 VMware는 vRealize Operations Manager API에 대한 언어별 클라이언트 바인딩을 제공합니다. [스키마 참조 정보](#)를 참조하십시오.

스키마 참조 정보

vRealize Operations Manager REST API 설명서에는 vRealize Operations Manager API의 모든 요소, 유형, 쿼리 및 작업에 대한 참조 자료가 포함되어 있습니다. 또한 스키마 정의 파일도 포함됩니다.

참조는 HTML 형식으로 제공되며 vRealize Operations Manager 인스턴스와 함께 설치됩니다. 예를 들어, vRealize Operations Manager 인스턴스 URL이 <https://vrealize.example.com>인 경우 API 참조는 <https://vrealize.example.com/suite-api/docs/rest/index.html>에서 사용할 수 있습니다.

언어별 클라이언트 바인딩은 다음에서 사용할 수 있습니다.

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

vRealize Operations Manager API 예 정보

모든 예에는 HTTP 요청 및 응답이 포함됩니다. 다음 예에서는 모니터링되는 환경의 개체에 대한 정보 생성 및 쿼리와 같은 작업과 관련된 워크플로와 콘텐츠를 보여 줍니다.

예제 요청 본문은 JSON 형식입니다. vRealize Operations Manager API에 필요한 요청 헤더는 더 큰 예의 일부가 아닌 예제 요청에 포함되어 있습니다.

대부분의 예제 응답은 고찰 중인 작업과 관련된 이러한 요소 및 특성만 표시합니다. 줄임표(...)는 응답 본문 내에 생략된 콘텐츠를 나타냅니다.

API 시작

3

API 클라이언트 및 vRealize Operations Manager 서버는 HTTPS를 통해 통신하며 API 개체의 XML 표현을 교환합니다.

REST API 워크플로의 이 간단한 예에서는 VMware vCenter® 어댑터의 모델 정의에 포함된 가상 시스템 개체 유형에 대한 모든 메트릭 목록을 가져오는 방법을 보여 줍니다. API를 사용하여 모든 개체 유형에 대해 사용 가능한 메트릭의 전체 목록을 가져올 수 있습니다.

본 장은 다음 항목을 포함합니다.

- 인증 토큰 획득
- 어댑터 유형 및 개체 유형 찾기
- 개체에 대한 모든 메트릭 목록 생성

인증 토큰 획득

vRealize Operations Manager에는 인증되어야 하는 API 요청이 필요합니다. 이 워크플로의 첫 번째 단계는 인증 토큰을 가져오는 것입니다.

인증 토큰을 가져오기 위해 로그인 요청이 기본 HTTP 인증에 필요한 양식으로 사용자 자격 증명을 제공합니다. 이 예에서는 사용자가 URL <https://vrealize.example.com/>을 사용하여 vRealize Operations Manager 인스턴스에 로그인합니다.

참고 이 예에서는 토큰 기반 인증을 사용합니다. 인증 메커니즘에 대한 자세한 내용은 [보안](#)을 참조하십시오.

사전 요구 사항

- 웹 브라우저와 vRealize Operations Manager 서버 간의 채널을 보호합니다. 브라우저를 열고 다음과 같은 vRealize Operations Manager 인스턴스의 URL을 입력합니다.

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

시스템에서 연결이 비공개가 아니라고 경고합니다. 클릭하여 보안 예외를 확인하고 SSL 핸드셰이크를 구축합니다.

- API에 액세스할 수 있는지 확인합니다. 다음과 같이 끝에 `suite-api/docs/rest/index.html`이 추가된 vRealize Operations Manager 인스턴스의 URL을 입력합니다.

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

- vRealize Operations Manager 인스턴스의 사용자에게 대한 로그인 자격 증명이 있는지 확인합니다.

절차

- 1 토큰을 가져오기 위해 로그인 URL에 요청을 POST합니다.

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

로그인 요청 및 응답을 참조하십시오.

- 2 응답을 검토합니다.

요청에 성공하면 후속 API 요청에 포함해야 하는 ops 인증 토큰이 반환됩니다.

예제: 로그인 요청 및 응답

이 예에서는 로그인 사용자 이름이 **vRealize-user**이고 비밀번호가 **vRealize-dummy-password**인 사용자에게 대한 요청 및 응답을 보여줍니다.

요청 헤더:

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

JSON 형식의 요청 본문:

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

JSON의 응답:

```
200 OK
```

```
{
  "token": "8f868cca-27cc-43d6-a838-c5467e73ec45::77cea9b2-1e87-490e-b626-e878beead23b",
  "validity": 1470421325035,
  "expiresAt": "Friday, August 5, 2016 6:22:05 PM UTC",
  "roles": []
}
```

이 응답 코드는 요청이 성공했는지 여부 또는 어떻게 실패했는지를 나타냅니다.

- 요청이 성공하면 서버는 HTTP 응답 코드 200(OK)와 6시간 후에 만료되는 다시 사용 가능한 ops 인증 토큰을 반환합니다. 이 토큰은 각 후속 API 요청에 포함되어야 합니다.

- 요청에 대한 인증 헤더가 누락된 경우, 서버는 HTTP 응답 코드 403을 반환합니다.
- 인증 헤더에 제공된 자격 증명이 잘못된 경우, 서버는 HTTP 응답 코드 401을 반환합니다.

어댑터 유형 및 개체 유형 찾기

vRealize Operations Manager 인스턴스에는 여러 어댑터 유형이 포함되어 있습니다. vCenter 어댑터의 어댑터 유형을 찾으려면 GET 요청을 수행하여 모든 어댑터 유형의 목록을 검색합니다. API 응답에는 어댑터가 모니터링하는 모든 개체 유형이 포함됩니다.

사전 요구 사항

vRealize Operations Manager 인스턴스에 로그인했는지 확인합니다.

절차

- 1 모든 어댑터 유형에 대해 GET 요청을 수행합니다.

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

- 2 응답을 검토하여 vCenter 어댑터와 모니터링되는 개체 유형의 목록을 찾습니다.

[vCenter 어댑터의 어댑터 유형 및 개체 유형 확인](#)의 응답 부분을 참조하십시오.

예제: vCenter 어댑터의 어댑터 유형 및 개체 유형 확인

이 예에서는 vCenter 어댑터의 유형 및 어댑터 모델 정의에 포함된 모든 개체 유형을 찾습니다.

요청 헤더:

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

여기서 *vROps_token*은 [인증 토큰 획득](#)의 응답에서 가져온 토큰입니다.

vCenter 어댑터에 대한 JSON의 응답 코드 조각:

```
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"
  ],
  ...
}

```

vCenter 어댑터의 경우, `adapter-kind` 키는 **VMWARE**입니다. `resourceKinds`은(는) vCenter 어댑터가 모니터링하는 개체 유형입니다. 가상 시스템 개체 유형의 경우, `resourceKinds`은(는) **VirtualMachine**입니다.

개체에 대한 모든 메트릭 목록 생성

vCenter 어댑터 모델에서 정의된 모든 가상 시스템에 대한 전체 메트릭 목록을 생성하려면 어댑터 유형과 개체 유형을 사용하여 URL에 대한 GET 요청을 수행합니다.

사전 요구 사항

다음 요구 사항이 충족되는지 확인합니다.

- vRealize Operations Manager 인스턴스에 로그인했습니다.
- vCenter 어댑터의 `adapterKind` 값과 가상 시스템의 `resourceKinds` 값을 알고 있습니다. [vCenter 어댑터의 어댑터 유형 및 개체 유형 확인](#) 항목을 참조하십시오.

절차

- 1 메트릭의 메타데이터를 가져오기 위해 GET 요청을 수행합니다.

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

- 2 응답에 나열된 메트릭을 사용자 인터페이스에 표시되는 메트릭과 비교합니다. [API 및 사용자 인터페이스의 가상 시스템 메트릭](#) 항목을 참조하십시오.

예제: API 및 사용자 인터페이스의 가상 시스템 메트릭

이 예에서는 XML 응답에 나열된 가상 시스템 메트릭이 vRealize Operations Manager 사용자 인터페이스에 표시된 메트릭과 비교되는 방식을 보여 줍니다.

요청:

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

여기서:

- VMWARE는 adapterKindKey입니다.
- VirtualMachine은 resourceKindKey입니다.
- vROps_token은 인증 토큰 획득의 응답에서 가져온 토큰입니다.

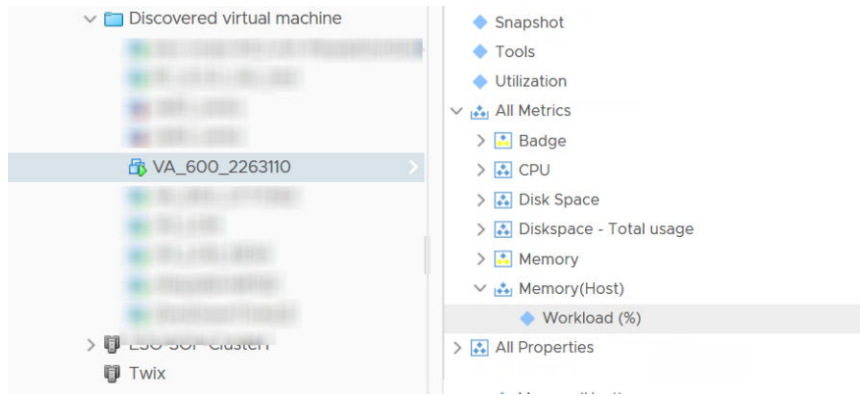
JSON의 응답 코드 조각:

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
    },
    ...
  ]
}
```

응답의 모든 resourceTypeAttribute은(는) 가상 시스템 개체에 대한 메타데이터가 포함된 메트릭입니다. name은(는) vRealize Operations Manager 사용자 인터페이스에 표시되는 텍스트에 해당합니다. 이 예에서 코드 조각은 메모리 및 호스트 워크로드에 대한 메트릭을 나열합니다.

응답의 메트릭을 사용자 인터페이스의 메트릭과 비교하려면 **vrealize.example.com**에서 실행되는 vRealize Operations Manager 인스턴스에 로그인하고 가상 시스템의 메트릭으로 이동합니다. 다음 예는 메모리(호스트) 및 워크로드에 대한 메트릭을 찾을 수 있는 위치를 보여 줍니다.



이 예는 가상 시스템 개체 유형에 대한 메트릭을 검색하는 방법을 보여 줍니다. 다른 개체 유형에 대한 메트릭을 검색하려면 GET 요청의 **VirtualMachine**을 다른 **resourceKinds(으)**로 바꿉니다.

어댑터 인스턴스 구성

4

어댑터가 있는 관리 팩이 포함된 솔루션을 설치한 후, 어댑터 모델 정의의 개체에서 데이터를 수집하도록 어댑터 인스턴스를 구성해야 합니다. vRealize Operations Manager API를 사용하여 어댑터 인스턴스를 구성할 수 있습니다.

이 사용 사례 예제는 vSphere 솔루션에 대한 어댑터 인스턴스를 구성하는 방법을 보여 주며 다음을 포함합니다.

- 요청, 요청 본문 및 각 작업에 대한 각 특정 절차의
- 응답이 있는 작업 요약

본 장은 다음 항목을 포함합니다.

- 어댑터 인스턴스 구성 요청 요약
- 솔루션 및 해당 어댑터 식별
- 어댑터에 필요한 개체 유형 식별
- 어댑터 인스턴스 생성
- 인증서 유효성 증명 제공
- 새 어댑터 인스턴스 모니터링 시작

어댑터 인스턴스 구성 요청 요약

어댑터 인스턴스를 구성하기 위해 순차적 API 요청을 수행합니다. 이전 요청의 응답은 후속 요청에 필요한 정보를 생성합니다.

표 4-1. 요청 요약

작업	요청	요청 본문	응답
제품에 등록된 모든 솔루션을 가져오고 어댑터 유형 식별	GET <API-URL>/suite-api/api/solutions	없음	adapterkindkeys
특정 어댑터 유형에 대한 모든 개체 유형을 가져옵니다.	GET <API-URL>/suite-api/api/adapterkinds/{key}/resourcekinds	없음	resourceIdentifierTypes

표 4-1. 요청 요약 (계속)

작업	요청	요청 본문	응답
어댑터 인스턴스 개체 생성	POST <API-URL>/suite-api/api/adapters	resourceIdentifiers 및 credential 값	adapterid
제공된 인증서를 승인하기 위해 어댑터 인스턴스 패치	PATCH <API-URL>/suite-api/api/adapters	POST <API-URL>/suite-api/api/adapters의 API 응답	adapter-certificates가 없는 POST <API-URL>/suite-api/api/adapters의 API 응답
어댑터 모니터링 시작	PUT <API-URL>/suite-api/api/adapters/{adapterid}/monitoringstate/start	없음	200 OK

솔루션 및 해당 어댑터 식별

vRealize Operations Manager 인스턴스에 몇 가지 솔루션이 설치되어 있을 수 있습니다. vSphere 솔루션 및 해당 어댑터 유형을 찾으려면 GET 요청을 수행하여 모든 솔루션의 목록을 검색합니다. 응답에는 솔루션이 포함된 모든 어댑터가 포함됩니다.

이 예제 사용 사례에서는 vRealize Operations Manager 인스턴스에 대한 API-URL이 `vrealize.example.com`입니다.

사전 요구 사항

vRealize Operations Manager 인스턴스에 대한 API URL에 로그인할 수 있는지 확인합니다. [인증 토큰 획득](#)을 참조하십시오.

절차

- 1 모든 솔루션을 나열하기 위해 GET 요청을 수행합니다.

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

- 2 응답을 검토하여 vSphere 솔루션 및 해당 어댑터 유형을 찾습니다.

[vSphere 솔루션의 어댑터 유형](#)의 응답 부분을 참조하십시오.

예제: vSphere 솔루션의 어댑터 유형

이 예에서는 설치된 모든 솔루션과 각 솔루션의 어댑터 유형을 나열합니다.

요청 헤더:

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

JSON의 응답:

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"
      ]
    }
  ]
}
```

응답은 설치된 세 가지 솔루션을 보여 줍니다.

- Management Pack for Log Insight 솔루션
- End Point Operations 솔루션
- vSphere 솔루션

vSphere 솔루션에는 두 가지 어댑터 유형이 있습니다.

- VMWARE
- PythonRemediationVcenterAdapter

vCenter 어댑터의 경우 어댑터 유형은 VMWARE입니다.

어댑터에 필요한 개체 유형 식별

vCenter 어댑터의 인스턴스를 생성하도록 결정한 후에는 해당 어댑터에 필요한 개체 유형을 식별해야 합니다. vCenter 어댑터의 모든 개체 유형 목록을 검색하도록 GET 요청을 수행합니다.

사전 요구 사항

vCenter 어댑터의 어댑터 유형을 파악하고 있는지 확인합니다.

절차

- 1 vCenter 어댑터의 모든 개체 유형을 나열하기 위해 GET 요청을 수행합니다.

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

- 2 응답을 살펴 필요한 개체 유형을 식별합니다.

vCenter 어댑터에 필요한 개체 유형의 응답 부분을 참조하십시오.

예제: vCenter 어댑터에 필요한 개체 유형

이 예에서는 vCenter 어댑터의 모든 개체 유형을 찾습니다.

요청 헤더:

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

JSON의 응답 코드 조각:

```
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
}
],
...
}

```

이 코드 조각은 "resourceKindType": "ADAPTER_INSTANCE" 특성이 포함된 리소스 종류를 표시합니다. 리소스 식별자 "isPartOfUniqueness": true 이(가) 있는 모든 개체 유형에는 어댑터 인스턴스를 생성하기 위한 API 요청과 함께 해당 개체 유형에 대한 값이 필요합니다.

vCenter 어댑터의 어댑터 인스턴스에는 VCURL의 값 또는 vCenter의 URL이 필요합니다.

어댑터 인스턴스 생성

어댑터에 필요한 개체 유형을 식별한 후에는 개체 유형에 대한 매개 변수 값을 제공하여 어댑터 인스턴스를 생성합니다. POST 요청에 필수 매개 변수가 포함된 요청 본문이 포함되어 있습니다.

어댑터 인스턴스를 생성하려면 VCURL 설정이 필수입니다.

사전 요구 사항

vCenter에 대한 IP 주소 및 자격 증명이 있는지 확인합니다.

절차

- 1 어댑터 인스턴스를 생성하려면 POST 요청을 수행합니다.

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

- 2 응답을 검토하여 vSphere 솔루션 및 해당 어댑터 유형의 이름을 찾습니다.

어댑터 인스턴스의 응답 부분을 참조하십시오.

예제: 어댑터 인스턴스

이 예에서는 다음 매개 변수를 사용하여 vCenter에 대한 어댑터 인스턴스를 생성합니다.

- 표시 이름: VC 어댑터 인스턴스
- 설명: VC 12.345.678.9용 vCenter 어댑터 인스턴스
- vCenter Server IP 주소: https://12.345.678.9
- 자격 증명 이름: VC-Credential-1
- 사용자 이름: administrator@vsphere.local
- 비밀번호: VC-dummy-passwd

AUTODISCOVERY 및 PROCESSCHANGEEVENTS는 선택 사항이지만 요청 본문과 응답에 리소스 식별자의 추가 예제를 표시하기 위해 포함됩니다.

요청 헤더:

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

JSON 형식의 요청 본문:

```
{
  "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"
    }
],
},
}

```

JSON의 응답 코드 조각:

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": ""
      }
    ]
  }
}

```

```

        "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"
}

```

API는 개체를 고유하게 식별하는 내부 생성 UUID를 사용하여 새 어댑터를 생성합니다. API 응답에는 vCenter 12.345.678.9가 제공하는 인증서가 포함됩니다. 어댑터 인스턴스 ID 값은 데이터 모니터링 및 수집을 시작하는 데 사용됩니다.

인증서 유효성 증명 제공

vRealize Operations Manager가 vCenter에 연결하고 데이터 수집을 시작할 수 있으려면 먼저 어댑터 인스턴스에서 검색된 데이터 소스에서 유효한 인증서를 제공하는지 확인해야 합니다. PATCH 요청은 어댑터를 생성하는 데 사용된 POST 요청의 응답인 요청 본문을 포함하여 인증서가 유효한지 증명을 제공합니다.

사전 요구 사항

어댑터를 생성하는 데 사용된 POST 요청의 응답이 있는지 확인합니다. [어댑터 인스턴스](#)의 응답을 참조하십시오.

절차

- ◆ 사용자가 vCenter에서 제공한 인증서를 수락했음을 시스템에 알려려면 PATCH 요청을 수행합니다.

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

예제: 인증서 검증

이 예에서는 PATCH 요청에 대한 요청 본문이 어댑터 인스턴스를 생성하는 데 사용된 POST 요청의 응답과 동일합니다.

요청 헤더:

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

JSON 형식의 요청 본문:

```
{
  "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"
}
```

JSON의 응답:

```
{
  "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"
}

```

응답은 어댑터-인증서 섹션이 없는 요청 본문과 동일합니다.

새 어댑터 인스턴스 모니터링 시작

어댑터 인스턴스를 생성하고 유효한 인증서를 인식하도록 vRealize Operations Manager를 구성하고 나면 데이터 모니터링 및 수집을 시작합니다. PUT 요청은 새 개체를 검색하는 데 사용되는 어댑터 인스턴스의 UUID를 제공합니다.

사전 요구 사항

새로 생성된 어댑터 인스턴스의 UUID가 있는지 확인합니다. [어댑터 인스턴스](#)의 응답을 참조하십시오.

절차

- ◆ 새 어댑터 인스턴스로 모니터링을 시작하기 위한 PUT 요청을 수행합니다.

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

예제: 개체 검색 및 데이터 수집

이 예에서는 어댑터 인스턴스를 생성한 PUT 요청의 어댑터 인스턴스 ID를 사용하여 어댑터 모니터링 프로세스를 시작합니다.

요청 헤더:

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