

VMware vRealize Orchestrator 8.6.1 릴리스 정 보

vRealize Orchestrator 8.6.1

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

Copyright © 2023 VMware, Inc. All rights reserved. [저작권 및 상표 정보](#)

목차

- 1 소개 4
- 2 릴리스 버전 5
- 3 vRealize Orchestrator 8.6.1의 새로운 기능 6
- 4 vRealize Orchestrator Appliance 배포 7
- 5 vRealize Orchestrator 업그레이드 및 마이그레이션 8
- 6 vRealize Orchestrator 8.6.1과 함께 설치되는 플러그인 9
- 7 이전 릴리스의 vRealize Orchestrator 10
- 8 해결된 문제 11
- 9 알려진 문제 13
- 10 이전에 알려진 문제 14

소개

1

vRealize Orchestrator 8.6.1 | 2021년 11월 19일
이 릴리스 정보의 추가 사항 및 업데이트 사항을 확인하십시오.

릴리스 버전

2

vRealize Orchestrator 8.6.1 | 2021년 11월 19일

- vRealize Orchestrator 8.6.1 Appliance 빌드 18937479
- vRealize Orchestrator 8.6.1 업데이트 저장소 빌드 18937479

이 문서의 업데이트:

일자	네트워크 설명	유형
2021년 11월 19일	초기 게시.	
2022년 3월 2일	log4j 취약성과 관련된 업그레이드 실패를 해결하는 데 사용되는 KB 해결 방법 링크가 추가되었습니다.	

중요

KB 87120에서 단계를 수행한 후 업그레이드 실패

KB 87120에 설명된 CVE-2021-44228 및 CVE-2021-45046 log4j 취약성을 해결하는 데 사용되는 지침을 수행하면 vRealize Automation 및 vRealize Orchestrator 8.6.2 또는 이전 버전에서 업그레이드가 실패할 수 있습니다. 해결 방법은 [KB 87794](#)를 참조하십시오.

vRealize Orchestrator 8.6.1의 새로운 기능

3

새로 지원되는 Node.js 및 PowerCLI 버전.

vRealize Orchestrator는 이제 다음 버전의 Node.js 및 PowerCLI 런타임을 지원합니다.

- Node.js 버전 14
- PowerShell 7.1용 PowerCLI 버전 12.3.0

클러스터 상태 및 동기화 상태 모니터링에 대한 변경 사항.

vRealize Orchestrator 이전 릴리스에서는 제어 센터의 **Orchestrator 클러스터 관리** 및 **구성 검증** 페이지에서 클러스터의 상태 및 동기화 상태를 모니터링할 수 있었습니다. 더 이상 이러한 페이지를 사용하여 클러스터의 상태를 모니터링할 수 없습니다.

클러스터의 상태 및 동기화 상태를 모니터링하기 위한 새로운 권장 접근 방식은 vRealize Orchestrator 클라이언트에 로그인하고 대시보드의 **시스템** 탭으로 이동해야 합니다. 또는 vRealize Orchestrator 클라이언트를 사용할 수 없는 경우 vRealize Orchestrator Appliance 명령줄에서 `kubectl get pods -n prelude` 명령을 실행하여 클러스터의 상태를 확인할 수도 있습니다.

vRealize Orchestrator Appliance 배포

4

vRealize Orchestrator Appliance는 OVA 파일로 배포된 VMware Photon OS 기반 장치입니다. 내부 PostgreSQL 데이터베이스로 사전 구축 및 사전 구성되어 있으며 vCenter Server 6.0 이상과 함께 배포할 수 있습니다.

vRealize Orchestrator Appliance는 빠르고 사용법이 쉬우며 경제적인 방식으로 vRealize Automation, vCenter Server를 포함한 VMware Cloud 스택을 현재 사용 중인 IT 프로세스 및 환경에 통합할 수 있습니다.

vRealize Orchestrator Appliance 배포에 대한 지침은 [vRealize Orchestrator Appliance 다운로드 및 배포](#)를 참조하십시오.

vRealize Orchestrator Appliance 서버 구성에 관한 자세한 내용은 [독립형 vRealize Orchestrator 서버 구성](#)을 참조하십시오.

vRealize Orchestrator 업그레이드 및 마이그레이션

5

마운트된 ISO 이미지를 사용하여 독립형 또는 클러스터링된 vRealize Orchestrator 8.x 배포를 최신 제품 버전으로 업그레이드할 수 있습니다.

vRealize Orchestrator Appliance 업그레이드에 대한 자세한 내용은 [vRealize Orchestrator 업그레이드](#)를 참조하십시오.

vSphere 또는 vRealize Automation을 통해 인증된 독립형 vRealize Orchestrator 인스턴스를 vRealize Orchestrator 8.6.1로 마이그레이션할 수 있습니다. 마이그레이션이 지원되는 vRealize Orchestrator 7.x 제품 버전에는 버전 7.3~7.6이 포함됩니다. 클러스터링된 vRealize Orchestrator 7.x 배포 마이그레이션은 지원되지 않습니다.

vRealize Orchestrator Appliance 마이그레이션에 대한 자세한 내용은 [vRealize Orchestrator 마이그레이션](#)을 참조하십시오.

vRealize Orchestrator 8.6.1과 함께 설치되는 플러그인

6

vRealize Orchestrator 8.6.1은 기본적으로 다음 플러그인과 함께 설치됩니다.

- vRealize Orchestrator vCenter Server 플러그인 7.0.0
- vRealize Orchestrator 메일 플러그인 8.0.0
- vRealize Orchestrator SQL 플러그인 1.1.8
- vRealize Orchestrator SSH 플러그인 7.3.0
- vRealize Orchestrator SOAP 플러그인 2.0.6
- vRealize Orchestrator HTTP-REST 플러그인 2.4.1
- Microsoft Active Directory용 vRealize Orchestrator Plug-in 3.0.11
- vRealize Orchestrator AMQP 플러그인 1.0.6
- vRealize Orchestrator SNMP 플러그인 1.0.3
- vRealize Orchestrator PowerShell 플러그인 1.0.20
- vRealize Orchestrator Multi-Node 플러그인 8.6.1
- vRealize Orchestrator 동적 유형 1.3.6
- vRealize Orchestrator vCloud Suite API(vAPI) 플러그인 7.5.2

이전 릴리스의 vRealize Orchestrator

7

이전 릴리스의 vRealize Orchestrator 기능 및 문제가 각 릴리스에 해당하는 릴리스 정보에 설명되어 있습니다. 이전 릴리스의 vRealize Orchestrator 릴리스 정보를 검토하려면 다음 링크 중 하나를 클릭합니다.

- [vRealize Orchestrator 8.6](#)
- [vRealize Orchestrator 8.5.1](#)
- [vRealize Orchestrator 8.5](#)
- [vRealize Orchestrator 8.4.2](#)
- [vRealize Orchestrator 8.4.1](#)
- [vRealize Orchestrator 8.4](#)
- [vRealize Orchestrator 8.3](#)
- [vRealize Orchestrator 8.2 패치 1](#)
- [vRealize Orchestrator 8.2](#)
- [vRealize Orchestrator 8.1 패치 3](#)
- [vRealize Orchestrator 8.1 패치 2](#)
- [vRealize Orchestrator 8.1 패치 1](#)
- [vRealize Orchestrator 8.1](#)
- [vRealize Orchestrator 8.0.1](#)
- [vRealize Orchestrator 8.0](#)
- [vRealize Orchestrator 7.6.0](#)
- [vRealize Orchestrator 7.5.0](#)
- [vRealize Orchestrator 7.4.0](#)

해결된 문제

8

- 워크플로 회귀가 발생합니다.

리팩터링 후 사용되는 기본 **RequestConfig** 개체가 의도치 않게 변경되었으며 이로 인해 압축이 기본적으로 도입되었습니다(예: Apache HTTP 클라이언트가 응답에서 **Content-Length** 헤더 제거). 이러한 차이로 인해 워크플로에 회귀가 발생할 수 있습니다.

- vRealize Orchestrator 8.4.0에서 vRealize Orchestrator 8.5.1로 업그레이드한 후 기본이 아닌 테넌트로 로그인한 경우 vRealize Orchestrator 클라이언트를 로드할 수 없지만 기본 테넌트로 로그인할 수 있습니다.

클라이언트가 특정 vRealize Automation 테넌트로 인해 통합으로 등록되어도 하위 테넌트는 vRealize Automation에 내장된 vRealize Orchestrator Client를 사용할 수 없습니다. vRealize Orchestrator 클라이언트에 로그인을 시도하면 하위 테넌트가 다음 오류 메시지를 수신합니다. **O, 알 수 없는 오류.**

- vRealize Orchestrator 7.6에서 vRealize Orchestrator 8.6으로 마이그레이션한 후 SQL 스크립트 오류로 인해 **vco-app** 포드가 시작되지 않습니다.

install_rpms init 컨테이너가 0이 아닌 상태로 종료되므로 **vco-app** 포드가 시작되지 않습니다. SQL 스크립트 **merge_duplicates**가 다음 오류로 인해 실패합니다.

```
[ERROR: column "createdat" does not exist Where: PL/pgSQL function
merge_duplicates(text,text,text) line 105 at EXECUTESQL statement "SELECT
merge_duplicates('vmo_workflowcategory', 'vmo_workflow', 'workflowcategoryid')"PL/pgSQL
function inline_code_block line 4 at PERFORM]
```

- HTTP-REST 플러그인은 대/소문자를 구분하지 않고 HTTP 헤더 이름을 처리해야 합니다.

[RFC 2616](#)에 따라 헤더 필드 이름은 대/소문자를 구분하지 않아야 합니다. HTTP-REST 플러그인의 이전 버전은 이름이 동일하더라도 대/소문자가 다른 헤더를 구분하므로 원치 않는 결과가 발생할 수 있습니다.

vRealize Orchestrator 8.6.1 이상에 패키징된 HTTP-REST 플러그인에는 다음 방법이 리팩터링되어 대/소문자와 관계없이 작동할 수 있습니다.

- **RESTRequest#setHeader**
- **RESTResponse#getHeaderValues**
- **RESTOperation#addMandatoryHeaderParameter**
- **RESTOperation#isHeaderParamMandatory**

RESTResponse#getAllHeaders 방법은 여전히 헤더의 대/소문자를 보장하지 않는 일반 사전을 반환합니다. 대/소문자를 구분하지 않는 특정 헤더를 가져오려면

RESTResponse#getHeaderValues(headerName)를 대신 사용합니다. HTTP-REST 플러그인의 이전 버전으로 생성된 예전 호스트 구성은 특히 REST 작업의 필수 헤더 매개 변수에 여전히 대/소문자를 구분하여 동작할 수 있습니다.

- **다시 실행** 기능을 사용하여 새 워크플로 실행을 시작하면 워크플로 입력 양식 위에 다음 오류 메시지가 표시됩니다. **일부 데이터를 검색할 수 없습니다. 문제가 지속되면 시스템 관리자에게 문의하십시오.**

단순한 유형의 비 개체 어레이(예: **어레이/숫자** 또는 **어레이/문자열**)로 구성된 입력 필드의 값이 잘못 평가되었으며 다른 입력 양식 필드의 값을 채울 때 사용하는 외부 소스 작업으로 전달됩니다.

알려진 문제

9

- **보호된 Git 분기로 커밋을 푸시할 수 없습니다.**

구성된 Git 분기가 보호되면 푸시 작업이 계속 실패해도 나타나는 메시지에는 푸시가 성공한 것으로 표시됩니다.

해결 방법: vRealize Orchestrator의 현재 또는 향후 릴리스에서 이 문제를 수정하지 않기로 했습니다. 이 알려진 문제 항목은 다음 릴리스의 릴리스 정보에서 삭제될 예정입니다.

- **워크플로를 복제 및 삭제한 후에는 로컬 변경 내용을 사용할 수 없습니다.**

워크플로를 복제한 후 삭제합니다. **Git 기록** 페이지에는 삭제된 워크플로의 로컬 변경 내용이 없습니다.

해결 방법은 없습니다.

- 변수 중 하나가 **속성** 유형인 경우 실패한 워크플로를 재개하는 상호 작용 양식이 정의된 모든 변수의 필드를 표시하지는 않습니다. **속성** 유형 변수 다음에 오는 변수의 입력은 숨겨집니다. **속성** 유형 변수의 입력 필드는 채워지지 않습니다.

변수 유형이 **속성** 유형으로 해석되지 않아 입력 양식 렌더링 중에 문제가 발생했습니다. 입력 양식을 생성하는 동안 이제 해당 유형이 사용됩니다.

해결 방법은 없습니다.

이전에 알려진 문제

10

이 섹션에는 이전에 알려진 문제(여전히 제품에 남아있는 이전 vRealize Orchestrator 릴리스의 알려진 문제)가 포함되어 있습니다.

- **vRealize Orchestrator 제어 센터에 플러그인을 설치하는 동안 오류 메시지가 표시됩니다.**

제어 센터의 **플러그인 관리** 페이지에서 플러그인을 설치하면 다음 오류 메시지가 나타납니다.

'name_of_the_plug-in' 플러그인(plug-in_file_name)이 현재 플랫폼 버전과 호환되지 않습니다. 지원되는 플랫폼 버전은 'names_of_the_supported_versions'입니다. 그래도 [설치] 버튼을 클릭하면 설치가 진행됩니다.

해결 방법: 이 오류 메시지는 무시하고 플러그인 설치를 계속 진행하면 됩니다.

- **인증 제공자 인증서가 변경되거나 다시 생성되면 vRealize Orchestrator 인증 구성이 무효화될 있습니다.**

제어 센터에서 인증 제공자로 구성된 vRealize Automation 또는 vSphere 인스턴스의 SSL 인증서가 변경되거나 다시 생성되면 vRealize Orchestrator 인증 구성이 무효화되고 vRealize Orchestrator 서버를 시작할 수 없습니다.

해결 방법: 새 인증 제공자 인증서를 가져옵니다.

- a 제어 센터에 **root**로 로그인합니다.
- b **인증서**를 클릭합니다.
- c **신뢰할 수 있는 인증서에서 가져오기** 탭을 클릭합니다.
- d URL 또는 파일에서 SSL 인증서를 로드합니다.
- e **가져오기**를 클릭합니다.

- **SOAP 플러그인이 인증된 프록시 서버를 통해 연결되지 않습니다.**

SOAP 호스트 추가 워크플로를 실행하려고 하면 프록시 서버 인증에 문제가 발생합니다.

해결 방법: 워크플로를 실행할 때 인증이 필요하지 않은 프록시 서버를 사용합니다.

- **SOAP 또는 REST 호스트에 연결하거나 인증서를 가져올 때 문제가 발생하는 경우 특정 버전의 SSL 또는 TLS를 사용해야 할 수 있습니다.**

이 문제와 관련된 내용은 [JSSE\(Java Secure Socket Extension\) 참조 가이드](#)를 참조하십시오.

해결 방법: 송신 HTTPS 연결이 SSLv3 및 TLSv1을 명시적으로 사용하도록 설정하는 방법에 관한 내용은 [vRO 6.0.x에서 송신 HTTPS 연결이 TLSv1.1 및 TLSv1.2를 사용하도록 설정하는 방법\(KB 2144316\)](#)을 참조하십시오.

■ **SSH 플러그인에 방화벽 연결 문제가 발생합니다.**

SSH 플러그인을 Cisco Adaptive Security Appliance(ASA) 방화벽에 연결할 수 없습니다.

해결 방법: vRealize Orchestrator 7.1용 SSH 플러그인은 Cisco Adaptive Security Appliance(ASA) 방화벽에 대한 연결을 지원하지 않습니다.

■ **특정 컨텍스트에서 ASCII가 아닌 문자를 처리할 때 문제 발생.**

입력 매개 변수에서 ASCII가 아닌 문자를 사용하면 다음 상황에서 잘못된 동작이 발생합니다.

- 이름에 ASCII가 아닌 문자를 포함하는 파일이 있는 SSH 폴더에서 **SCP put** 또는 **SCP get** 워크플로를 실행하면 워크플로는 실행되지만 대상 시스템에서 결과 파일의 이름을 판독할 수 없습니다.
- ASCII가 아닌 문자를 변수 이름에 삽입하면 문자가 표시되지 않습니다. 이 문제는 워크플로 변수 및 작업 변수로 인해 발생합니다.

해결 방법은 없습니다.

■ **vCenter Server 플러그인의 스토리지 vSAN 워크플로가 SSD(Solid-State Drive) 디스크를 ESXi 호스트에 추가하는 것을 지원하지 않습니다.**

Add disks to disk group 및 **Remove disks from disk groups** 워크플로가 SSD 디스크를 용량 디스크로 ESXi 호스트에 추가하는 것을 지원하지 않습니다.

해결 방법은 없습니다.

■ **REST 호스트 인스턴스가 Swagger 사양을 사용하여 생성되면 RESTOperation ID가 올바르게 초기화되지 않습니다.**

HTTP-REST 플러그인에서 REST 호스트 인스턴스가 Swagger 사양으로 인해 생성되면 **RESTOperation ID**가 올바르게 초기화되지 않고 **RESTHost** 개체의 **getOperation** 메서드가 작동하지 않습니다.

해결 방법은 없습니다.

■ **어레이 유형의 vCenter Server 데이터 개체 속성에 값을 추가할 수 없습니다.**

vRealize Orchestrator가 스크립트를 실행하면 vCenter Server 플러그인이 JavaScript 어레이를 고정된 크기의 Java 어레이로 변환합니다. 따라서 어레이를 속성 값으로 갖는 vCenter Server 데이터 개체에 새 값을 추가할 수 없습니다. 미리 채워진 어레이로 전달하여 개체를 인스턴스화하면 어레이를 속성으로 갖는 개체를 생성할 수 있습니다. 하지만 개체를 인스턴스화한 후에는 값을 어레이에 추가할 수 없습니다.

예를 들어, 다음 코드는 작동하지 않습니다.

```
var spec = new VcVirtualMachineConfigSpec();
spec.deviceChange = [];
spec.deviceChange[0] = new VcVirtualDeviceConfigSpec();
System.log(spec.deviceChange[0]);
```

위 코드에서 vRealize Orchestrator는 빈 **spec.deviceChange** JavaScript 어레이를 고정 크기의 Java 어레이 **VirtualDeviceConfigSpec[]**로 **setDeviceChange()**를 호출하기 전에 변환합니다.

spec.deviceChange[0] = new VcVirtualDeviceConfigSpec()을 호출하면 vRealize Orchestrator는 **getDeviceChange()**를 호출하고 어레이는 고정된 비어 있는 Java 어레이 상태로 유지됩니다.

spec.deviceChange.add()를 호출하면 같은 동작이 발생합니다.

해결 방법: 어레이를 지역 변수로 선언:

```
var spec = new VcVirtualMachineConfigSpec();
var deviceSpec = [];
deviceSpec[0] = new VcVirtualDeviceConfigSpec();
spec.deviceChange = deviceSpec;
System.log(spec.deviceChange[0]);
```