

VMware vRealize Orchestrator 설치 및 구성

vRealize Orchestrator 7.5

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

VMware vRealize Orchestrator 설치 및 구성 6

1 VMware vRealize Orchestrator 소개 7

- Orchestrator 플랫폼의 주요 기능 7
- Orchestrator 사용자 유형과 관련 책임 9
- Orchestrator 아키텍처 10
- Orchestrator 플러그인 11

2 Orchestrator 시스템 요구 사항 12

- Orchestrator Appliance의 하드웨어 요구 사항 12
- Orchestrator에서 지원하는 브라우저 12
- Orchestrator 데이터베이스 요구 사항 13
- Orchestrator Appliance에 포함된 소프트웨어 13
- 국제화 지원 레벨 13
- Orchestrator 네트워크 포트 14

3 vRealize Orchestrator 구성 요소 설정 16

- vCenter Server 설정 16
- 인증 방법 16

4 vRealize Orchestrator 설치 17

- vRealize Orchestrator Appliance 다운로드 및 배포 17
 - vRealize Orchestrator Appliance의 전원을 켜고 홈 페이지 열기 18
 - 루트 암호 변경 19
 - vRealize Orchestrator Appliance에서 SSH 관리자 로그인을 사용 또는 사용 안 함 19
 - vRealize Orchestrator Appliance에 대한 네트워크 설정 구성 20

5 초기 구성 21

- 독립형 Orchestrator 서버 구성 21
 - vRealize Automation 인증을 사용하여 독립형 Orchestrator 서버 구성 21
 - vSphere 인증을 사용하여 독립형 Orchestrator 서버 구성 23
- Orchestrator 네트워크 포트 24
- Orchestrator 데이터베이스 연결 25
- 인증서 관리 25
 - Orchestrator 인증서 관리 25
- Orchestrator 플러그인 구성 28

- vRealize Orchestrator 플러그인 관리 28
- vRealize Orchestrator 플러그인 설치 또는 업데이트 28
 - 플러그인 제거 29
- Orchestrator 가용성 및 확장성 30
 - VAMI에서 vRealize Orchestrator 인스턴스의 클러스터 구성 30
 - Orchestrator 클러스터 모니터링 32
 - Orchestrator 클러스터에 대해 동기화 모드 사용하도록 설정 32
 - Orchestrator 복제 노드를 기본 노드로 승격 33
 - Orchestrator 클러스터 노드 삭제 33
- 고객 경험 향상 프로그램 구성 34
 - VMware에 수신되는 정보의 범주 34
 - CEIP(고객 환경 향상 프로그램) 참여 34
- 6 API 서비스 사용 35**
 - REST API를 통해 SSL 인증서 관리 35
 - REST API를 사용하여 SSL 인증서 삭제 35
 - REST API를 사용하여 SSL 인증서 가져오기 36
 - REST API를 사용하여 키 저장소 생성 37
 - REST API를 사용하여 키 저장소 삭제 38
 - REST API를 사용하여 키 추가 38
 - 제어 센터 REST API를 사용하여 Orchestrator 구성 자동화 38
- 7 추가 구성 옵션 40**
 - 인증 재구성 40
 - 인증 제공자 변경 40
 - 인증 매개 변수 변경 41
 - Orchestrator 구성 내보내기 42
 - Orchestrator 구성 가져오기 42
 - 워크플로 실행 속성 구성 43
 - Orchestrator 로그 파일 44
 - 로깅 지속성 44
 - Orchestrator 로그 구성 45
 - Orchestrator 로그 필터링 45
 - 원격 서버와 로깅 통합 구성 46
 - 네트워크 인터페이스 컨트롤러 추가 47
 - 정적 경로 구성 47
- 8 구성 사용 사례 및 문제 해결 49**
 - vSphere Web Client용 vRealize Orchestrator 플러그인 구성 49
 - Orchestrator 인증 등록 취소 50

- SSL 인증서 변경 50
 - 로컬 저장소에 인증서 추가 51
 - Orchestrator Appliance 관리 사이트의 인증서 변경 51
- 실행 중인 워크플로 취소 52
- Orchestrator 서버 디버깅 사용 52
- Orchestrator 구성 및 요소 백업 53
- vRealize Orchestrator 백업 및 복원 56
 - vRealize Orchestrator 백업 56
 - vRealize Orchestrator 인스턴스 복원 57
- Site Recovery Manager를 사용한 Orchestrator의 재해 복구 58
 - vSphere Replication에 대한 가상 시스템 구성 58
 - 보호 그룹 만들기 59
 - 복구 계획 만들기 60
 - 폴더의 복구 계획 구성 60
 - 복구 계획 편집 61

9 시스템 속성 설정 62

- Orchestrator 클라이언트에 대한 비관리자의 액세스 사용 안 함 62
- 워크플로 및 작업에 대한 서버 파일 시스템 액세스 설정 63
 - Orchestrator 시스템에 대한 쓰기 액세스 권한을 허용하는 js-io-rights.conf 파일의 규칙 63
 - 워크플로 및 작업에 대한 서버 파일 시스템 액세스 설정 64
- 워크플로 및 작업에 대한 운영 체제 명령의 액세스 설정 65
- Java 클래스에 JavaScript 액세스 설정 66
- 사용자 지정 시간 초과 속성 설정 67

10 이후 작업 정보 68

- Orchestrator Appliance 웹 콘솔에서 Orchestrator 클라이언트에 로그인 68

VMware vRealize Orchestrator 설치 및 구성

"설치 및 구성 VMware vRealize Orchestrator" 은 VMware® vRealize Orchestrator의 설치, 업그레이드 및 구성에 대한 정보와 지침을 제공합니다.

대상 사용자

이 정보는 가상 시스템 기술과 데이터 센터 작업에 익숙한 고급 vSphere 관리자 및 숙련된 시스템 관리자를 대상으로 제공됩니다.

VMware vRealize Orchestrator 소개

1

VMware vRealize Orchestrator는 VMware 제품은 물론 타사 기술을 관리할 자동화, 구성 가능 프로세스를 만들고 실행할 수 있는 확장 가능한 워크플로 라이브러리를 제공하는 개발 및 프로세스 자동화 플랫폼입니다.

vRealize Orchestrator는 VMware 및 타사 애플리케이션(서비스 데스크, 변경 관리 시스템, IT 자산 관리 시스템 등) 모두의 관리 및 운영 작업을 자동화합니다.

본 장은 다음 항목을 포함합니다.

- [Orchestrator 플랫폼의 주요 기능](#)
- [Orchestrator 사용자 유형과 관련 책임](#)
- [Orchestrator 아키텍처](#)
- [Orchestrator 플러그인](#)

Orchestrator 플랫폼의 주요 기능

Orchestrator는 오케스트레이션 도구에 필요한 공통 기능을 제공하는 오케스트레이션 플랫폼, 하위 시스템의 제어를 통합하는 플러그인 아키텍처 그리고 워크플로 라이브러리의 세 가지 다른 계층으로 구성됩니다. Orchestrator는 새 플러그인 및 라이브러리를 사용하여 확장할 수 있는 개방형 플랫폼이며 REST API를 통해 더 큰 아키텍처에 통합될 수 있습니다.

Orchestrator에는 워크플로 실행 및 관리에 도움이 되는 몇 가지 주요 기능이 포함되어 있습니다.

지속성

운영 등급 데이터베이스는 프로세스, 워크플로 상태 및 Orchestrator 구성과 같은 관련 정보를 저장하는 데 사용됩니다.

중앙 집중식 관리

Orchestrator는 중앙 집중식으로 프로세스를 관리할 방법을 제공합니다. 전체 버전 기록을 가진 애플리케이션 서버 기반 플랫폼은 같은 스토리지 위치에 스크립트와 프로세스 관련 기본 형식을 저장할 수 있습니다. 이러한 방식으로 버전 관리가 되지 않고 서버에 적절한 변경 제어가 없는 스크립트를 방지할 수 있습니다.

검사점 설정

워크플로의 각 단계는 데이터베이스에 저장되므로 서버를 다시 시작해야 할 경우 데이터 손실이 예방됩니다. 이 기능은 특히 장기 실행 프로세스에 유용합니다.

제어 센터

제어 센터는 런타임 작업, 워크플로 모니터링, 통합 로그 액세스 및 구성, 그리고 워크플로 실행 및 시스템 리소스 간 상관 관계에 대한 중앙 집중식 관리 인터페이스를 제공하여 vRealize Orchestrator 인스턴스의 관리 효율성을 높이는 웹 기반 포털입니다. Orchestrator 로깅 메커니즘은 Orchestrator 엔진 처리량에 대한 다양한 성능 메트릭을 수집하는 추가 로그 파일을 사용해 최적화됩니다.

버전 관리

모든 Orchestrator 플랫폼 개체에는 관련 버전 기록이 있습니다. 버전 기록은 프로젝트 단계 또는 위치에 프로세스를 배포할 때 기본 변경 관리에 유용합니다.

스크립팅 엔진

Mozilla Rhino JavaScript 엔진은 Orchestrator 플랫폼을 위한 빌딩 블록을 만드는 방법을 제공합니다. 스크립팅 엔진은 기본 버전 제어, 변수 유형 확인, 이름 공간 관리 및 예외 처리를 사용해 향상됩니다. 이 엔진은 다음 빌딩 블록에서 사용할 수 있습니다.

- 작업
- 워크플로
- 정책

워크플로 엔진

워크플로 엔진을 사용해 비즈니스 프로세스를 자동화할 수 있습니다. 워크플로 엔진은 워크플로에서 단계별 프로세스 자동화를 만들기 위해 다음 개체를 사용합니다.

- Orchestrator가 제공하는 워크플로 및 작업
- 고객이 만든 사용자 지정 빌딩 블록
- 플러그인이 Orchestrator에 추가한 개체

사용자, 기타 워크플로, 스케줄 또는 정책이 워크플로를 시작할 수 있습니다.

정책 엔진

정책 엔진을 사용해 Orchestrator 서버 또는 플러그인된 기술에서 변경되는 조건에 반응해 이벤트를 모니터링하고 생성할 수 있습니다. 정책은 플랫폼이나 플러그인에서 이벤트를 집계할 수 있으며 이는 통합된 기술에서 변경되는 조건을 처리하는 데 도움이 됩니다.

모니터링 클라이언트

웹 UI 모니터링 클라이언트를 통해 Orchestrator 프로세스를 모니터링합니다. 이 정보를 사용하여 Orchestrator 프로세스의 문제를 해결할 수 있습니다.

개발 및 리소스

vRealize Orchestrator에서 사용할 수 있는 고유한 플러그인을 개발하는 데 도움이 되도록 Orchestrator 방문 페이지에서는 리소스에 빠르게 액세스할 수 있습니다. Orchestrator REST API를 사용하여 Orchestrator 서버에 요청을 보내는 방법에 대한 정보도 찾을 수 있습니다.

보안

Orchestrator는 다음과 같은 고급 보안 기능을 제공합니다.

- 서버 간에 가져오고 내보낸 콘텐츠에 서명하고 암호화하는 공용 키 인프라(PKI).
- 내보낸 콘텐츠의 보기, 편집 및 재배포 방법을 제어하는 디지털 저작권 관리(DRM).
- 데스크톱 클라이언트와 서버 간 암호화된 통신과 웹 프론트 엔드에 HTTPS 액세스를 제공하는 SSL(Secure Sockets Layer).
- 고급 액세스 권한 관리를 통해 이러한 프로세스가 조작하는 프로세스 및 개체에 대한 액세스를 제어할 수 있습니다.

암호화

vRealize Orchestrator는 문자열 암호화를 위해 256비트 암호화 키를 사용하는 FIPS 준수 AES(Advanced Encryption Standard)를 사용합니다. 암호화 키는 임의로 생성되며 클러스터의 일부가 아닌 장치 전반에서 고유합니다. 클러스터의 모든 노드가 같은 암호 키를 공유합니다.

Orchestrator 사용자 유형과 관련 책임

Orchestrator는 전역 사용자 역할의 특정 책임을 기반으로 다양한 도구와 인터페이스를 제공합니다. Orchestrator에는 관리자 그룹에 포함된 모든 권한을 가진 사용자(관리자)와 관리자 그룹에 포함되지 않은 제한된 권한을 가진 사용자(최종 사용자)가 있습니다.

모든 권한을 가진 사용자

Orchestrator 관리자 및 개발자는 동등한 관리 권한을 가지지만 책임이라는 관점에서 구분됩니다.

관리자

이 역할은 Orchestrator 플랫폼의 모든 기능에 액세스할 수 있습니다. 기본 관리 책임은 다음 항목을 포함합니다.

- Orchestrator 설치 및 구성
- Orchestrator 및 애플리케이션의 액세스 권한 관리
- 패키지 가져오기 및 내보내기
- 워크플로 및 스케줄 지정 작업 실행
- 가져온 요소의 버전 제어 관리

- 새 워크플로 및 플러그인 생성

개발자

이 사용자 유형은 Orchestrator 플랫폼의 모든 기능에 액세스할 수 있습니다. 개발자에게는 Orchestrator 클라이언트 인터페이스의 액세스 권한이 부여되며 다음과 같은 책임이 있습니다.

- Orchestrator 플랫폼 기능을 확장하는 애플리케이션 작성
- 기존 워크플로를 사용자 지정하고 새 워크플로 및 플러그인을 생성해 프로세스 자동화

제한된 권한을 가진 사용자

최종 사용자

최종 사용자는 관리자나 개발자가 Orchestrator 클라이언트에서 사용할 수 있도록 만든 워크플로 및 정책을 실행 및 스케줄링할 수 있습니다.

Orchestrator 아키텍처

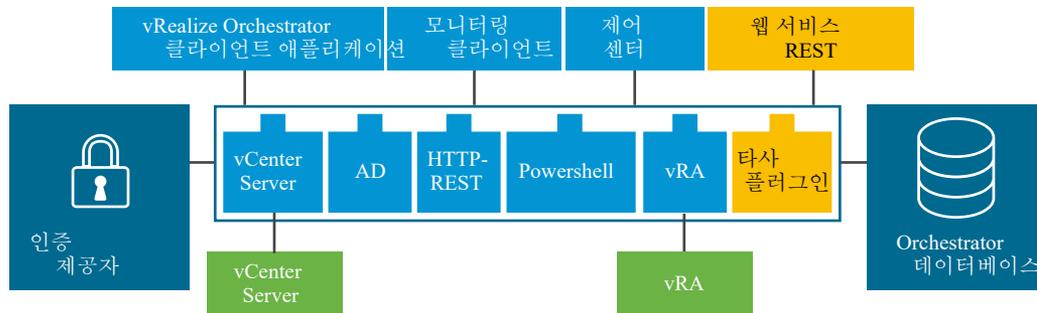
Orchestrator는 사용자가 오케스트레이션 프로세스를 자동화하는 워크플로를 생성하고 실행할 수 있는 워크플로 라이브러리와 워크플로 엔진을 포함하고 있습니다. 사용자는 Orchestrator가 일련의 플러그인을 통해 액세스하는 다양한 기술의 개체에서 워크플로를 실행합니다.

Orchestrator는 vCenter Server 및 vRealize Automation용 플러그인을 포함하는 표준 플러그인 집합을 제공해 사용자가 플러그인이 노출된 다양한 환경에서 작업을 오케스트레이션할 수 있습니다.

또한 Orchestrator는 외부 타사 애플리케이션을 오케스트레이션 플랫폼에 연결하기 위한 개방형 아키텍처를 제공합니다. 사용자는 직접 정의한 플러그인된 기술의 개체를 워크플로에서 실행할 수 있습니다.

Orchestrator는 사용자 계정을 관리하기 위해 인증 제공자에게 접속하며 실행하는 워크플로의 정보를 저장하기 위해 데이터베이스에 접속합니다. Orchestrator 클라이언트 인터페이스 또는 웹 서비스를 통해 Orchestrator, 노출되는 개체 및 Orchestrator 워크플로에 액세스할 수 있습니다. Orchestrator 워크플로 및 서비스의 모니터링과 구성은 모니터링 클라이언트 및 제어 센터를 통해 수행됩니다.

그림 1-1. VMware vRealize Orchestrator 아키텍처



Orchestrator 플러그인

플러그인을 사용하여 Orchestrator로 외부 기술 및 애플리케이션에 액세스하고 이를 제어할 수 있습니다. Orchestrator 플러그인에서 외부 기술을 노출시킴으로써 해당 외부 기술의 개체와 기능에 액세스하는 워크플로에서 개체와 기능을 통합할 수 있습니다.

플러그인을 사용하여 액세스할 수 있는 외부 기술에는 가상화 관리 도구, 이메일 시스템, 데이터베이스, 디렉토리 서비스, 원격 제어 인터페이스 등이 포함됩니다.

Orchestrator는 VMware vCenter Server API 및 이메일 기능과 같은 기술을 워크플로에 통합하는 데 사용할 수 있는 표준 플러그인 집합을 제공합니다. 플러그인을 사용하여 새로운 IT 서비스의 배포를 자동화하거나 기존 vRealize Automation 인프라 및 애플리케이션 서비스의 기능을 조정할 수 있습니다. 또한 Orchestrator 개방형 플러그인 아키텍처를 사용하여 다른 애플리케이션에 액세스하는 플러그인을 개발할 수 있습니다.

VMware가 개발하는 Orchestrator 플러그인은 `.vmoapp` 파일로 배포됩니다. VMware에서 개발하고 배포하는 Orchestrator 플러그인에 대한 자세한 내용은 [vRealize Orchestrator 외부 플러그인](#)을 참조하십시오. 타사 Orchestrator 플러그인에 대한 자세한 내용은 [VMware Solution Exchange](#)를 참조하십시오.

Orchestrator 시스템 요구 사항

2

시스템은 Orchestrator가 제대로 작동하기 위해 필요한 기술적 요구 사항을 충족해야 합니다.

지원되는 vCenter Server, vSphere Web Client, vRealize Automation 및 기타 VMware 솔루션과 호환 가능한 데이터베이스 버전 목록은 [VMware 제품 상호 운용성 매트릭스](#)를 참조하십시오.

본 장은 다음 항목을 포함합니다.

- [Orchestrator Appliance의 하드웨어 요구 사항](#)
- [Orchestrator에서 지원하는 브라우저](#)
- [Orchestrator 데이터베이스 요구 사항](#)
- [Orchestrator Appliance에 포함된 소프트웨어](#)
- [국제화 지원 레벨](#)
- [Orchestrator 네트워크 포트](#)

Orchestrator Appliance의 하드웨어 요구 사항

Orchestrator Appliance는 미리 구성된 Linux 기반 가상 시스템입니다. 해당 장치를 배포하기 전에 시스템이 최소 하드웨어 요구 사항을 충족하는지 확인합니다.

Orchestrator Appliance의 하드웨어 요구 사항은 다음과 같습니다.

- CPU 2개
- 6GB 메모리
- 17GB 하드 디스크

Orchestrator 서버에는 2GB 이상의 여유 메모리가 필요하므로 기본 메모리 크기를 줄이지 마십시오.

Orchestrator에서 지원하는 브라우저

제어 센터에는 웹 브라우저가 필요합니다.

제어 센터에 연결하려면 다음 브라우저 중 하나를 사용해야 합니다.

- Microsoft Edge
- Mozilla Firefox
- Google Chrome

Orchestrator 데이터베이스 요구 사항

Orchestrator 서버는 운영 환경에 즉시 사용할 수 있는 미리 구성된 PostgreSQL 데이터베이스를 포함하고 있습니다.

vRealize Orchestrator 7.5부터 외부 데이터베이스 통합은 지원되지 않습니다. 미리 구성된 PostgreSQL 데이터베이스만 사용할 수 있습니다.

Orchestrator Appliance에 포함된 소프트웨어

Orchestrator Appliance는 Orchestrator를 실행하도록 최적화된 미리 구성된 가상 시스템입니다. 이 장치는 미리 설치된 소프트웨어와 함께 배포됩니다.

Orchestrator Appliance 패키지에는 다음과 같은 소프트웨어가 포함되어 있습니다.

- VMware용 SUSE Linux Enterprise Server 11 업데이트 3, 64비트 버전
- PostgreSQL
- Orchestrator

기본 Orchestrator Appliance 데이터베이스 구성은 즉시 사용할 수 있습니다.

참고 운영 환경에서 Orchestrator Appliance를 사용하려면 vRealize Automation 또는 vSphere를 통해 인증하도록 Orchestrator 서버를 구성해야 합니다. 인증 제공자 구성에 대한 자세한 내용은 [독립형 Orchestrator 서버 구성](#)을 참조하십시오.

국제화 지원 레벨

Orchestrator 제어 센터는 스페인어, 프랑스어, 독일어, 중국어(번체), 중국어(간체), 한국어, 일본어 로케일을 지원합니다. Orchestrator 클라이언트는 국제화 레벨 1을 지원합니다.

Orchestrator의 비 ASCII 문자 지원

Orchestrator 클라이언트는 현지화되지 않았지만 영어 이외 운영 체제에서 실행 가능하고 비 ASCII 문자도 지원합니다.

표 2-1. Orchestrator GUI의 비 ASCII 문자 지원

비 ASCII 문자에 대한 지원				
Orchestrator 항목	설명 필드	이름 필드	입력 및 출력 매개 변수	특성
작업	예	아니요	아니요	아니요
폴더	예	예	-	-
구성 요소	예	예	-	아니요
패키지	예	예	-	-
정책	예	예	-	-
정책 템플릿	예	예	-	-
리소스 요소	예	예	-	-
워크플로	예	예	아니요	아니요
워크플로 프레젠테이션 표시 그룹 및 입력 단계	예	예	-	-

Orchestrator 네트워크 포트

Orchestrator는 다른 시스템과 통신하기 위해 특정 포트를 사용합니다. 포트는 변경할 수 없는 기본값으로 설정되어 있습니다.

기본 구성 포트

Orchestrator 서비스를 제공하려면 기본 포트를 설정하고 방화벽이 수신되는 TCP 연결을 허용하도록 구성해야 합니다.

참고 기타 포트는 사용자 지정 플러그인을 사용하면 필요할 수 있습니다.

표 2-2. VMware vRealize Orchestrator 기본 구성 포트

포트	번호	프로토콜	소스	대상	설명
가상 장치 관리 인터페이스	5480	TCP			장치 시스템 설정 인터페이스에 대한 액세스 포트
HTTP 서버 포트	8280	TCP	최종 사용자 웹 브라우저	Orchestrator 서버	Orchestrator 기본 HTTP 웹 포트 8280으로 전송된 요청은 기본 HTTPS 웹 포트 8281로 리디렉션됩니다.
HTTPS 서버 포트	8281	TCP	최종 사용자 웹 브라우저	Orchestrator 서버	Web Orchestrator 홈 페이지의 액세스 포트.
웹 구성 HTTPS 액세스 포트	8283	TCP	최종 사용자 웹 브라우저	Orchestrator 구성	Orchestrator 구성의 웹 UI에 대한 SSL 액세스 포트.

외부 통신 포트

Orchestrator가 외부 서비스와 통신할 수 있도록 방화벽이 송신 연결을 허용하도록 구성해야 합니다.

표 2-3. VMware vRealize Orchestrator 외부 통신 포트

포트	번호	프로토콜	소스	대상	설명
PostgreSQL	5432	TCP	Orchestrator 서버	PostgreSQL 서버	Orchestrator 데이터베이스로 구성된 PostgreSQL Server와 통신하기 위해 사용되는 포트.
SMTP 서버 포트	25	TCP	Orchestrator 서버	SMTP 서버	이메일 알림을 위해 사용되는 포트.
vCenter Server API 포트	443	TCP	Orchestrator 서버	vCenter Server	오케스트레이션된 vCenter Server 인스턴스에서 가상 인프라 및 가상 시스템 정보를 얻기 위해 Orchestrator가 사용하는 vCenter Server API 통신 포트.

vRealize Orchestrator 구성 요소 설정

3

vRealize Orchestrator Appliance를 다운로드하고 배포하면 vRealize Orchestrator 서버가 미리 구성됩니다. 배포 후 서비스가 자동으로 시작됩니다.

vRealize Orchestrator 설정의 가용성 및 확장성을 강화하려면 다음 지침을 따르십시오.

- 인증 제공자를 설치 및 구성하고 vRealize Orchestrator가 이 제공자와 작동하도록 구성합니다.
- 클러스터링된 vRealize Orchestrator 환경의 경우, 로드 밸런싱 서버를 설치 및 구성하고 둘 이상의 vRealize Orchestrator 서버 간에 워크로드를 분산하도록 구성합니다.

본 장은 다음 항목을 포함합니다.

- [vCenter Server 설정](#)
- [인증 방법](#)

vCenter Server 설정

Orchestrator 설정에서 vCenter Server 인스턴스의 수를 늘리면 Orchestrator가 더 많은 세션을 관리하게 됩니다. 활성 세션이 너무 많으면 10개 이상의 vCenter Server 연결이 발생할 때 Orchestrator에서 시간 초과가 발생할 수 있습니다.

지원되는 vCenter Server 버전 목록은 [VMware 제품 상호 운용성 매트릭스](#)를 참조하십시오.

참고 네트워크에 대역폭과 지연 시간이 충분한 경우 Orchestrator 설정의 다양한 가상 시스템에서 여러 개의 vCenter Server 인스턴스를 실행할 수 있습니다. Orchestrator와 vCenter Server 간의 통신을 개선하기 위해 LAN을 사용 중인 경우 100Mb 회선을 반드시 사용해야 합니다.

인증 방법

사용자 권한을 인증 및 관리하려면 Orchestrator에 vRealize Automation 또는 vSphere 서버 인스턴스에 대한 연결이 필요합니다.

Orchestrator Appliance를 다운로드 및 배포할 때 vRealize Automation 또는 vSphere와의 연결을 설정해야 합니다.

vRealize Orchestrator 설치

4

vRealize Orchestrator는 서버 구성 요소와 클라이언트 구성 요소로 이루어집니다.

vRealize Orchestrator를 사용하려면 vRealize Orchestrator Appliance를 배포하고 vRealize Orchestrator 서버를 구성해야 합니다.

vRealize Orchestrator 제어 센터를 사용하여 기본 vRealize Orchestrator 구성 설정을 변경할 수 있습니다.

본 장은 다음 항목을 포함합니다.

- [vRealize Orchestrator Appliance 다운로드 및 배포](#)

vRealize Orchestrator Appliance 다운로드 및 배포

템플릿에서 vRealize Orchestrator Appliance를 배포하여 다운로드 및 설치합니다.

사전 요구 사항

- vCenter Server가 설치되고 실행 중인지 확인합니다.
- vRealize Orchestrator Appliance를 배포하는 호스트가 하드웨어 최소 요구 사항을 충족하는지 확인합니다. 자세한 내용은 [Orchestrator Appliance의 하드웨어 요구 사항](#) 항목을 참조하십시오.
- 시스템이 분리되어 있고 인터넷에 연결되지 않은 경우 VMware 웹 사이트에서 해당 장치에 대한 .ova 파일을 다운로드해야 합니다.

절차

- 1 vSphere Web Client에 관리자로 로그인합니다.
- 2 vSphere Web Client에서 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트와 같은 가상 시스템의 유효한 상위 개체인 인벤토리 개체를 선택합니다.
- 3 **작업 > OVF 템플릿 배포**를 선택합니다.
- 4 .ova 파일의 경로 또는 URL을 입력하고 **다음**을 클릭합니다.
- 5 배포된 vRealize Orchestrator Appliance의 이름 및 위치를 입력하고 **다음**을 클릭합니다.
- 6 호스트, 클러스터, 리소스 풀 또는 vApp을 장치가 실행되는 대상으로 선택하고 **다음**을 클릭합니다.

- 7 배포 세부 정보를 검토하고 **다음**을 클릭합니다.
- 8 라이선스 계약 조건에 동의하고 **다음**을 클릭합니다.
- 9 배포된 vRealize Orchestrator Appliance에 사용할 스토리지 형식을 선택합니다.

형식	설명
느리게 비워지는 썸 프로비저닝	기본 썸 형식의 가상 디스크를 생성합니다. 가상 디스크에 필요한 공간은 가상 디스크 생성 중에 할당됩니다. 물리적 디바이스에 남아 있는 데이터는 생성 동안에 지워지지 않지만 나중에 가상 시스템에서 처음으로 쓰는 경우, 해당 데이터는 요구대로 비워집니다.
빠르게 비워지는 썸 프로비저닝	Fault Tolerance와 같은 클러스터링 기능을 지원합니다. 가상 디스크에 필요한 공간은 가상 디스크 생성 중에 할당됩니다. 물리적 디바이스에 남아 있는 모든 데이터는 가상 디스크를 생성하는 동안 비워집니다. 다른 형식의 디스크를 생성하는 것보다 이 형식의 디스크를 생성하는 것이 더 오래 걸릴 수도 있습니다.
썸 프로비저닝된 형식	하드 디스크 공간을 절약합니다. 썸 디스크의 경우 선택하는 디스크 크기 값에 기반하여 디스크가 필요로 하는 만큼의 데이터스토어 공간을 프로비저닝합니다. 썸 디스크는 먼저 작은 크기부터 시작합니다. 초기 작업을 위해 이 디스크에 필요한 데이터스토어 공간 만큼의 크기만 사용합니다.

- 10 **다음**을 클릭합니다.
- 11 (선택 사항) 네트워크 설정을 구성하고 **다음**을 클릭합니다.

기본적으로 vRealize Orchestrator Appliance는 DHCP를 사용합니다. 장치 웹 콘솔에서 이 설정을 변경하고 고정 IP 주소를 할당할 수 있습니다.

- 12 사용하려는 옵션을 선택하고 루트 사용자 계정의 초기 암호를 설정합니다.
초기 암호는 8자 이상이어야 합니다.

중요 Orchestrator Appliance의 루트 계정에 대한 암호는 365일 후에 만료됩니다. Orchestrator Appliance에 root로 로그인하고 `passwd -x number_of_days name_of_account`를 실행하여 계정에 대한 만료 시간을 늘릴 수 있습니다. Orchestrator Appliance 루트 암호의 만료 시간을 무한대로 늘리려면 `passwd -x 99999 root`를 실행합니다.

- 13 **완료 준비** 페이지를 검토하고 **마침**을 클릭합니다.

결과

vRealize Orchestrator Appliance가 배포되었습니다.

vRealize Orchestrator Appliance의 전원을 켜고 홈 페이지 열기

vRealize Orchestrator Appliance를 사용하려면 먼저 전원을 켜고 가상 장치의 IP 주소를 가져와야 합니다.

절차

- 1 vSphere Web client에 관리자로 로그인합니다.

- 2 vRealize Orchestrator Appliance를 마우스 오른쪽 버튼으로 클릭하고 **전원 > 전원 켜기**를 선택합니다.
- 3 장치 전원이 켜진 후 **요약** 탭을 선택하여 vRealize Orchestrator Appliance IP 주소를 확인합니다.
- 4 웹 브라우저에서 vRealize Orchestrator Appliance 가상 시스템의 호스트 주소로 이동합니다.
https://your_orchestrator_hostname/vco.

루트 암호 변경

보안상의 이유로 vRealize Orchestrator Appliance의 루트 암호를 변경할 수 있습니다.

기본적으로, vRealize Orchestrator Appliance의 루트 계정에 대한 암호는 365일 후에 만료됩니다. SSH 클라이언트를 통해 vRealize Orchestrator Appliance에 로그인하고 `passwd -x number_of_days name_of_account`를 실행하여 루트 계정의 만료 기간을 늘릴 수 있습니다. vRealize Orchestrator Appliance 루트 암호의 만료 시간을 무한대로 늘리려면 `passwd -x 99999 root`를 실행합니다.

사전 요구 사항

- vRealize Orchestrator Appliance를 다운로드하고 배포합니다.
- vRealize Orchestrator Appliance가 가동되어 실행 중인지 확인합니다.

절차

- 1 vRealize Orchestrator VAMI에 **root**로 로그인합니다.
https://your_orchestrator_hostname:5480에서 VAMI에 액세스합니다.
- 2 **관리자** 탭을 선택합니다.
- 3 **현재 관리자 암호** 텍스트 상자에 현재 루트 암호를 입력합니다.
- 4 **새 관리자 암호 및 새 관리자 암호 다시 입력** 텍스트 상자에 새 암호를 입력합니다.
- 5 **설정 저장**을 클릭합니다.

결과

vRealize Orchestrator Appliance의 루트 Linux 사용자 암호가 변경되었습니다.

vRealize Orchestrator Appliance에서 SSH 관리자 로그인을 사용 또는 사용 안 함

vRealize Orchestrator Appliance에 대한 SSH 액세스를 사용하거나 사용하지 않도록 설정할 수 있습니다.

사전 요구 사항

- vRealize Orchestrator Appliance를 다운로드하고 배포합니다.
- vRealize Orchestrator Appliance가 가동되어 실행 중인지 확인합니다.

절차

- 1 vRealize Orchestrator VAMI에 **root**로 로그인합니다.
https://your_orchestrator_hostname:5480에서 VAMI에 액세스합니다.
- 2 **관리자** 탭에서 **SSH 서비스 사용**을 클릭하여 vRealize Orchestrator SSH 서비스를 사용하거나 사용하지 않도록 설정합니다.
- 3 (선택 사항) **관리자 SSH 로그인 사용**을 클릭하여 SSH를 사용한 vRealize Orchestrator Appliance에 대한 루트 액세스를 사용하거나 사용하지 않도록 설정합니다.
- 4 **설정 저장**을 클릭합니다.

결과

사용하도록 설정하면 **SSH 상태**가 **실행 중**으로 표시됩니다. 사용하지 않도록 설정하면 **SSH 상태**가 **중지됨**으로 표시됩니다.

vRealize Orchestrator Appliance에 대한 네트워크 설정 구성

정적 IP 주소를 할당하고 프록시 설정을 정의하도록 vRealize Orchestrator Appliance에 대한 네트워크 설정을 구성합니다.

사전 요구 사항

- vRealize Orchestrator Appliance를 다운로드하고 배포합니다.
- vRealize Orchestrator Appliance가 가동되어 실행 중인지 확인합니다.

절차

- 1 vRealize Orchestrator VAMI에 **root**로 로그인합니다.
https://your_orchestrator_hostname:5480에서 VAMI에 액세스합니다.
- 2 **네트워크** 탭에서 **주소**를 클릭합니다.
- 3 vRealize Orchestrator Appliance에서 IP 설정을 가져오는 방법을 선택합니다.

옵션	설명
DHCP	DHCP 서버에서 IP 설정을 가져옵니다. 기본 설정입니다.
정적	정적 IP 설정 사용. 이 옵션을 선택하면 IP 주소, 넷마스크(IPv4의 경우), 접두사 (IPv6의 경우) 및 게이트웨이 정보를 입력하라는 메시지가 표시됩니다.

네트워크 설정에 따라 IPv4 및 IPv6 주소 유형을 선택해야 할 수도 있습니다.

- 4 **설정 저장**을 클릭합니다.
- 5 (선택 사항) 프록시 서버를 구성하려면 **프록시** 탭을 선택합니다.
- 6 (선택 사항) 프록시 설정을 구성한 후 **설정 저장**을 클릭합니다.

Orchestrator로 작업 자동화와 시스템 및 애플리케이션 관리를 시작하기 전에, 외부 인증 제공자를 사용하도록 구성하고 다양한 사용자에게 역할을 할당해야 합니다. 또한 CA 서명 인증서를 가져오고, 플러그인을 설치하거나 기본 로그 구성을 변경할 수 있습니다.

본 장은 다음 항목을 포함합니다.

- 독립형 Orchestrator 서버 구성
- Orchestrator 네트워크 포트
- Orchestrator 데이터베이스 연결
- 인증서 관리
- Orchestrator 플러그인 구성
- Orchestrator 가용성 및 확장성
- 고객 경험 향상 프로그램 구성

독립형 Orchestrator 서버 구성

Orchestrator Appliance는 미리 구성되어 있는 Linux 기반 가상 시스템이지만 Orchestrator 제어 센터에 액세스하려면 먼저 구성 마법사에 따라 작업해야 합니다.

vRealize Automation 인증을 사용하여 독립형 Orchestrator 서버 구성

Orchestrator Appliance 사용을 준비하려면 호스트 설정과 인증 제공자를 구성해야 합니다. vRealize Automation 구성 요소 레지스트리를 통해 인증하도록 Orchestrator를 구성할 수 있습니다.

사전 요구 사항

- 최신 버전의 vRealize Orchestrator Appliance를 다운로드한 후 배포합니다. [vRealize Orchestrator Appliance 다운로드 및 배포](#) 항목을 참조하십시오.
- vRealize Automation을 설치 및 구성하고 vRealize Automation 서버가 실행되고 있는지 확인합니다. vRealize Automation 설명서를 참조하십시오.

클러스터를 생성하려는 경우 다음 작업을 수행하십시오.

- vRealize Orchestrator의 여러 인스턴스에서 트래픽을 분배하기 위해 로드 밸런서를 설정합니다. 자세한 내용은 vRealize Orchestrator 로드 밸런싱 설명서를 참조하십시오.

절차

- 1 제어 센터에 액세스하여 구성 마법사를 시작합니다.
 - a `https://your_orchestrator_server_IP_or_DNS_name:8283/vco-controlcenter`로 이동합니다.
 - b OVA 배포 동안 입력한 암호를 사용하여 **root**로 로그인합니다.

- 2 **변경**을 클릭하여 제어 센터에서 액세스 가능한 호스트 이름을 구성합니다.

참고 Orchestrator 클러스터를 구성할 계획인 경우, 로드 밸런서 가상 서버의 호스트 이름을 입력합니다.

- 3 인증 제공자를 구성합니다.
 - a **인증 제공자 구성** 페이지의 **인증 모드** 드롭다운 메뉴에서 **vRealize Automation**을 선택합니다.
 - b **호스트 주소** 텍스트 상자에서 vRealize Automation 호스트 주소를 입력하고 **연결**을 클릭합니다.
 - c **인증서 수락**을 클릭합니다.
 - d **사용자 이름** 및 **암호** 텍스트 상자에서 vRealize Automation에서의 SSO 연결을 위해 구성된 사용자 계정의 자격 증명을 입력합니다. **등록**을 클릭합니다.
기본적으로 SSO 계정은 **관리자**이며 기본 테넌트 이름은 **vsphere.local**입니다.
 - e **관리자 그룹** 텍스트 상자에서 관리자 그룹의 이름을 입력하고 **검색**을 클릭합니다.
예를 들어 **vsphere.local\vcadmins**와 같습니다.
 - f 그룹 목록에서 선택할 그룹의 이름을 두 번 클릭합니다.
 - g **변경 내용 저장**을 클릭합니다.
성공적으로 저장했다는 메시지가 표시되고 제어 센터의 기본 보기로 리디렉션됩니다.

결과

제어 센터 구성을 성공적으로 완료했습니다.

다음에 수행할 작업

- **VRA가 라이선싱** 페이지에서 구성된 라이선스 제공자인지 확인합니다.
- **구성 검증** 페이지에서 노드가 올바르게 구성되었는지 확인합니다.

참고 인증 제공자의 구성에 따라 2분 후에 Orchestrator 서버가 자동으로 다시 시작됩니다. 프로세스 완료 직후에 구성을 확인하면 잘못된 구성 상태가 반환될 수 있습니다.

vSphere 인증을 사용하여 독립형 Orchestrator 서버 구성

vSphere 인증 모드를 사용하여 vCenter Single Sign-On 서버로 Orchestrator 서버를 등록합니다. vCenter Single Sign-On 인증을 vCenter Server 6.0 이상과 함께 사용합니다.

사전 요구 사항

- 최신 버전의 vRealize Orchestrator Appliance를 다운로드한 후 배포합니다. [vRealize Orchestrator Appliance 다운로드 및 배포](#) 항목을 참조하십시오.
- vCenter Single Sign-On이 실행 중인 vCenter Server를 설치 및 구성합니다. 자세한 정보는 vSphere 설명서를 참조하십시오.

클러스터를 생성하려는 경우 다음 작업을 수행하십시오.

- vRealize Orchestrator의 여러 인스턴스에서 트래픽을 분배하기 위해 로드 밸런서를 설정합니다. 자세한 내용은 vRealize Orchestrator 로드 밸런싱 설명서를 참조하십시오.

절차

- 1 제어 센터에 액세스하여 구성 마법사를 시작합니다.
 - a https://your_orchestrator_server_IP_or_DNS_name:8283/vco-controlcenter로 이동합니다.
 - b OVA 배포 동안 입력한 암호를 사용하여 **root**로 로그인합니다.
- 2 **변경**을 클릭하여 제어 센터에서 액세스 가능한 호스트 이름을 구성합니다.

참고 Orchestrator 클러스터를 구성할 계획인 경우, 로드 밸런서 가상 서버의 호스트 이름을 입력합니다.

- 3 인증 제공자를 구성합니다.
 - a **인증 제공자 구성** 페이지에서 **vSphere(인증 모드 드롭다운 메뉴에서)**를 선택합니다.
 - b **호스트 주소** 텍스트 상자에 vCenter Single Sign-On을 포함하는 Platform Services Controller 인스턴스의 정규화된 도메인 이름 또는 IP 주소를 입력하고 **연결**을 클릭합니다.

참고 외부 Platform Services Controller 인스턴스 또는 로드 밸런서의 뒤에 있는 여러 Platform Services Controller 인스턴스를 사용하는 경우 동일한 vCenter Single Sign-On 도메인을 공유하는 모든 Platform Services Controller의 인증서를 Orchestrator에 수동으로 가져와야 합니다.

- c **인증서 수락**을 클릭합니다.
- d **사용자 이름 및 암호** 텍스트 상자에서 vCenter Single Sign-On 도메인의 로컬 관리자 계정의 자격 증명을 입력합니다. **등록**을 클릭합니다.

기본적으로 이 계정은 **administrator@vsphere.local**이며 기본 테넌트 이름은 **vsphere.local**입니다.

e **관리자 그룹** 텍스트 상자에서 관리자 그룹의 이름을 입력하고 **검색**을 클릭합니다.

예를 들어 **vsphere.local\vcadmins**와 같습니다.

f 그룹 목록에서 선택할 그룹의 이름을 두 번 클릭합니다.

g **변경 내용 저장**을 클릭합니다.

성공적으로 저장했다는 메시지가 표시되고 제어 센터의 기본 보기로 리디렉션됩니다.

결과

제어 센터 구성을 성공적으로 완료했습니다.

다음에 수행할 작업

- **CIS가 라이선싱** 페이지에서 구성된 라이선스 제공자인지 확인합니다.
- **구성 검증** 페이지에서 노드가 올바르게 구성되었는지 확인합니다.

참고 인증 제공자의 구성에 따라 2분 후에 Orchestrator 서버가 자동으로 다시 시작됩니다. 프로세스 완료 직후에 구성을 확인하면 잘못된 구성 상태가 반환될 수 있습니다.

Orchestrator 네트워크 포트

Orchestrator는 다른 시스템과 통신하기 위해 특정 포트를 사용합니다. 포트는 변경할 수 없는 기본값으로 설정되어 있습니다.

기본 구성 포트

Orchestrator 서비스를 제공하려면 기본 포트를 설정하고 방화벽이 수신되는 TCP 연결을 허용하도록 구성해야 합니다.

참고 기타 포트는 사용자 지정 플러그인을 사용하면 필요할 수 있습니다.

표 5-1. VMware vRealize Orchestrator 기본 구성 포트

포트	번호	프로토콜	소스	대상	설명
가상 장치 관리 인터페이스	5480	TCP			장치 시스템 설정 인터페이스에 대한 액세스 포트
HTTP 서버 포트	8280	TCP	최종 사용자 웹 브라우저	Orchestrator 서버	Orchestrator 기본 HTTP 웹 포트 8280으로 전송된 요청은 기본 HTTPS 웹 포트 8281로 리디렉션됩니다.
HTTPS 서버 포트	8281	TCP	최종 사용자 웹 브라우저	Orchestrator 서버	Web Orchestrator 홈 페이지의 액세스 포트.
웹 구성 HTTPS 액세스 포트	8283	TCP	최종 사용자 웹 브라우저	Orchestrator 구성	Orchestrator 구성의 웹 UI에 대한 SSL 액세스 포트.

외부 통신 포트

Orchestrator가 외부 서비스와 통신할 수 있도록 방화벽이 송신 연결을 허용하도록 구성해야 합니다.

표 5-2. VMware vRealize Orchestrator 외부 통신 포트

포트	번호	프로토콜	소스	대상	설명
PostgreSQL	5432	TCP	Orchestrator 서버	PostgreSQL 서버	Orchestrator 데이터베이스로 구성된 PostgreSQL Server와 통신하기 위해 사용되는 포트.
SMTP 서버 포트	25	TCP	Orchestrator 서버	SMTP 서버	이메일 알림을 위해 사용되는 포트.
vCenter Server API 포트	443	TCP	Orchestrator 서버	vCenter Server	오케스트레이션된 vCenter Server 인스턴스에서 가상 인프라 및 가상 시스템 정보를 얻기 위해 Orchestrator가 사용하는 vCenter Server API 통신 포트.

Orchestrator 데이터베이스 연결

Orchestrator 서버에는 데이터 저장을 위한 데이터베이스가 필요합니다.

Orchestrator Appliance를 다운로드하고 배포하면 Orchestrator 서버는 해당 장치에 사전 설치된 PostgreSQL 데이터베이스와 작동하도록 구성됩니다.

사전 구성된 Orchestrator PostgreSQL 데이터베이스는 즉시 사용 가능합니다. Orchestrator PostgreSQL의 모든 트랜잭션은 VAMI 인터페이스를 통해 자동으로 처리됩니다.

참고 vRealize Orchestrator 7.5부터 Oracle 및 Microsoft SQL 같은 외부 데이터베이스는 지원되지 않습니다.

인증서 관리

특정 서버용으로 발급되고 서버 공개 키에 대한 정보가 포함된 인증서를 사용하여 vRealize Orchestrator에서 생성된 모든 요소에 서명하고 신뢰성을 보장할 수 있습니다. 클라이언트가 서버에서 일반적으로 패키지인 요소를 수신하면 클라이언트는 사용자의 ID를 확인하고 서명의 신뢰 여부를 결정합니다.

■ Orchestrator 인증서 관리

제어 센터의 **인증서** 페이지에서 또는 Orchestrator 클라이언트를 통해 구성 워크플로 범주에서 SSL 신뢰 관리자 워크플로를 사용하여 Orchestrator 인증서를 관리할 수 있습니다.

Orchestrator 인증서 관리

제어 센터의 **인증서** 페이지에서 또는 Orchestrator 클라이언트를 통해 구성 워크플로 범주에서 SSL 신뢰 관리자 워크플로를 사용하여 Orchestrator 인증서를 관리할 수 있습니다.

인증서를 Orchestrator 신뢰 저장소에 가져오기

제어 센터는 vCenter Server, 관계형 데이터베이스 관리 시스템(RDBMS), LDAP, Single Sign-On 및 기타 서버와 통신하는 데 보안 연결을 사용합니다. URL 또는 PEM- 인코드 파일에서 필수 SSL 인증서를 가져올 수 있습니다. 서버 인스턴스에 대해 SSL 연결을 사용할 때마다 **인증서** 페이지의 **신뢰할 수 있는 인증서** 탭에서 해당 인증서를 가져오고 해당 SSL 인증서를 가져와야 합니다.

Orchestrator에서 URL 주소 또는 PEM 인코딩된 파일로부터 SSL 인증서를 로드할 수 있습니다.

옵션	설명
URL 또는 프록시 URL에서 가져오기	원격 서버의 URL입니다. https://your_server_IP_address 또는 your_server_IP_address:port
파일에서 가져오기	PEM 인코딩된 인증서 파일의 경로입니다. PEM- 인코드 인증서 파일 가져오기에 대한 자세한 정보는 제어 센터를 통해 신뢰할 수 있는 인증서 가져오기 를 참조하십시오.

자체 서명된 서버 인증서 생성

Orchestrator Appliance는 장치의 네트워크 설정을 기준으로 자동 생성되고 자체 서명된 SSL 인증서를 포함합니다. 장치의 네트워크 설정이 변경된 경우 수동으로 자체 서명된 새 인증서를 생성해야 합니다. 자체 서명된 인증서를 생성하여 암호화된 통신을 보장하고 패키지에 서명을 제공할 수 있습니다. 그러나, 수신자는 사용자로 할당된 타사 제공자가 아니라 사용자의 서버에 의해 실제로 발급되고 자체 서명된 패키지인지 확인할 수 없습니다. 서버 ID를 증명하려면 CA(인증 기관)에 의해 서명된 인증서를 사용합니다.

제어 센터 [인증서](#) 페이지의 **Orchestrator 서버 SSL 인증서** 탭에서 자체 서명된 인증서를 생성할 수 있습니다.

옵션	설명
서명 알고리즘	디지털 서명을 생성하는 암호화 알고리즘입니다.
일반 이름	Orchestrator 서버의 호스트 이름입니다.
조직	조직의 이름입니다. (예: VMware)
조직 구성 단위	조직 구성 단위의 이름입니다. (예: R&D)
국가 코드	국가 코드 약어입니다. (예: US)

Orchestrator는 사용 중인 환경에 대한 고유한 서버 인증서를 생성합니다. 인증서의 공개 키에 대한 세부 정보가 **Orchestrator 서버 SSL 인증서** 탭에 표시됩니다. 개인 키는 Orchestrator 데이터베이스의 vmo_keystore 테이블에 저장됩니다.

Orchestrator 서버 SSL 인증서 가져오기

vRealize Orchestrator는 SSL 인증서를 사용하여 보안 통신 중 클라이언트와 원격 서버에 대해 자체 시스템을 식별합니다. 기본적으로 Orchestrator는 장치의 네트워크 설정을 기준으로 자동 생성되고 자체 서명된 SSL 인증서를 포함합니다. CA(인증 기관)에 의해 서명된 SSL 인증서를 가져와서 인증서 신뢰 오류를 방지할 수 있습니다.

CA(인증 기관)에 의해 서명된 인증서를 공개 키 및 개인 키를 포함하는 PEM 인코딩된 파일로 가져와야 합니다.

참고 SSL 서버 인증서를 생성하거나 가져온 후 Orchestrator 구성자 서비스를 다시 시작합니다.

```
service vco-configurator restart
```

패키지 서명 인증서

Orchestrator 서버에서 내보낸 패키지는 디지털로 서명되어 있습니다. 패키지 서명에 사용될 새 인증서를 가져오고, 내보내고 또는 생성합니다. 패키지 서명 인증서는 암호화된 통신 및 Orchestrator 패키지 서명을 보장하는 데 사용되는 디지털 ID 양식입니다.

Orchestrator Appliance는 장치의 네트워크 설정을 기준으로 자동 생성된 패키지 서명 인증서를 포함합니다. 장치의 네트워크 설정이 변경된 경우 수동으로 새 패키지 서명 인증서를 생성해야 합니다.

참고 Orchestrator Appliance는 초기 Orchestrator 구성 중 자동 생성되고 자체 서명된 패키지 서명 인증서를 포함합니다. 패키지 서명 인증서를 변경할 수 있으며 변경 후에 내보내는 모든 패키지는 새 인증서로 서명됩니다.

제어 센터를 통해 신뢰할 수 있는 인증서 가져오기

기타 서버와 안전하게 통신하려면 Orchestrator 서버가 기타 서버의 ID를 확인할 수 있어야 합니다. 이를 위해 원격 엔티티의 SSL 인증서를 Orchestrator 신뢰 저장소로 가져와야 할 수 있습니다. 인증서를 신뢰하기 위해 특정 URL에 대한 연결을 설정하는 방법으로 인증서를 신뢰 저장소로 가져오거나 PEM- 인코드 파일 형태로 직접 가져올 수 있습니다.

사전 요구 사항

SSL을 통해 Orchestrator를 연결할 서버의 정규화된 도메인 이름을 찾습니다.

절차

- 1 SSH를 통해 Orchestrator Appliance에 **루트**로 로그인합니다.
- 2 다음 명령을 실행하여 원격 서버의 인증서를 검색합니다.

```
openssl s_client -connect host_or_dns_name:secure_port
```

- a 암호화되지 않은 포트를 사용하는 경우, **starttls** 및 필수 프로토콜을 **openssl** 명령과 함께 사용합니다.

```
openssl s_client -connect host_or_dns_name:port -starttls smtp
```

- 3 -----BEGIN CERTIFICATE-----부터 -----END CERTIFICATE----- 태그까지의 텍스트를 텍스트 편집기에 복사하고 파일로 저장합니다.
- 4 제어 센터에 **root**로 로그인합니다.
- 5 **인증서** 페이지로 이동합니다.
- 6 **신뢰할 수 있는 인증서** 탭에서 **가져오기**를 클릭하고 **PEM- 인코드 파일에서 가져오기** 옵션을 선택합니다.
- 7 인증서 파일을 찾아보고 **가져오기**를 클릭합니다.

결과

원격 서버 인증서를 Orchestrator 신뢰 저장소로 성공적으로 가져왔습니다.

Orchestrator 플러그인 구성

기본 Orchestrator 플러그인은 워크플로를 통해서만 구성됩니다.

기본 Orchestrator 플러그인을 구성하려면 Orchestrator 클라이언트에서 특정 워크플로를 사용해야 합니다.

vRealize Orchestrator 플러그인 관리

vRealize Orchestrator 제어 센터의 **플러그인 관리** 페이지에서 vRealize Orchestrator에 설치된 모든 플러그인 목록을 확인하고 기본 관리 작업을 수행할 수 있습니다.

플러그인 로깅 수준 변경

vRealize Orchestrator의 로깅 수준을 변경하는 대신 특정 플러그인에 대해서만 로깅 수준을 변경할 수 있습니다.

새 플러그인 설치 또는 업그레이드

vRealize Orchestrator 플러그인을 사용하여 vRealize Orchestrator 서버를 다른 소프트웨어 제품과 통합할 수 있습니다. vRealize Orchestrator Appliance에는 미리 설치된 플러그인 집합이 포함되어 있습니다. 사용자 지정 플러그인을 설치하여 vRealize Orchestrator 플랫폼의 기능을 한층 확장할 수도 있습니다.

vRealize Orchestrator의 **플러그인 관리** 페이지에서 플러그인을 설치하거나 업그레이드할 수 있습니다. 사용할 수 있는 파일 확장명은 `.vmoapp` 및 `.dar`입니다. `.vmoapp` 파일은 여러 `.dar` 파일 모음을 포함할 수 있으며, 애플리케이션으로 설치할 수 있습니다. `.dar` 파일은 하나의 플러그인과 연결된 모든 리소스를 포함합니다.

참고 vRealize Orchestrator 플러그인에 선호하는 파일 형식은 `.vmoapp`입니다.

vRealize Orchestrator 플러그인 설치 또는 업그레이드에 대한 자세한 내용은 [vRealize Orchestrator 플러그인 설치 또는 업데이트](#) 항목을 참조하십시오.

플러그인 사용 안 함

플러그인 이름 옆에 있는 **사용** 확인란을 선택 취소하여 플러그인을 사용하지 않도록 설정할 수 있습니다.

이 작업은 플러그인 파일을 제거하지 않습니다. Orchestrator에서 플러그인 설치 제거에 대한 자세한 내용은 [플러그인 제거](#)를 참조하십시오.

vRealize Orchestrator 플러그인 설치 또는 업데이트

vRealize Orchestrator 제어 센터를 사용하여 타사 플러그인을 설치하거나 업데이트할 수 있습니다.

사전 요구 사항

플러그인의 `.dar` 또는 `.vmoapp` 파일을 다운로드합니다.

참고 vRealize Orchestrator 플러그인에 선호하는 파일 형식은 `.vmoapp`입니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **플러그인 관리** 페이지를 선택합니다.
- 3 **찾아보기**를 클릭하고 설치하거나 업데이트할 플러그인의 **.dar** 또는 **.vmoapp** 파일을 선택합니다.
- 4 **업로드**를 클릭합니다.
- 5 플러그인 정보를 검토하고 최종 사용자 라이선스 계약에 동의한 후(해당하는 경우) **설치**를 클릭합니다.

플러그인이 설치 또는 업데이트되고 vRealize Orchestrator 서버 서비스가 다시 시작됩니다.

다음에 수행할 작업

플러그인 관리 페이지에 올바른 플러그인 정보가 나열되어 있는지 확인합니다.

플러그인 제거

제어 센터를 사용하여 플러그인을 삭제할 수 있지만, 이 작업을 수행해도 vRealize Orchestrator 환경에서 해당 콘텐츠가 모두 삭제되지는 않습니다. 제어 센터에서 플러그인을 삭제한 후에는 vRealize Orchestrator Client에서 연결된 플러그인 패키지 및 폴더를 삭제해야 합니다.

절차

- 1 Orchestrator 제어 센터에서 플러그인을 삭제합니다.
 - a 제어 센터에 **root**로 로그인합니다.
 - b **플러그인 관리**를 선택합니다.
 - c 삭제할 플러그인을 찾아서 삭제 아이콘을 클릭합니다.
 - d **삭제**를 클릭합니다.
- 2 vRealize Orchestrator Client에서 플러그인 패키지 및 폴더 삭제
 - a vRealize Orchestrator 클라이언트에 로그인합니다.
 - b 왼쪽 상단의 드롭다운 메뉴에서 **디자인**을 선택합니다.
 - c **패키지** 탭을 선택합니다.
 - d 삭제할 패키지를 마우스 오른쪽 버튼으로 클릭하고 **컨텐츠와 함께 요소 삭제**를 선택합니다.

참고 공유 사용자 지정 콘텐츠를 포함한 모든 플러그인 콘텐츠를 삭제하려면 **모두 삭제**를 선택합니다. 플러그인 패키지가 다른 vRealize Orchestrator 개체와 공유하는 사용자 지정 콘텐츠를 보존하려면 **공유 유지**를 선택합니다. 표준 라이브러리의 워크플로와 같이 읽기 전용 상태로 잠겨 있는 vRealize Orchestrator 내용은 선택된 옵션에 관계없이 삭제되지 않습니다.

- e **워크플로** 탭을 선택합니다.
- f 워크플로 라이브러리를 확장하고 제거하려는 플러그인의 폴더를 삭제합니다.

g **작업** 탭을 선택합니다.

h 제거할 플러그인의 작업 모듈을 삭제합니다.

3 vRealize Orchestrator 서비스를 다시 시작합니다.

```
service vco-configurator restart && service vco-server restart
```

결과

vRealize Orchestrator 환경에서 플러그인 및 해당 콘텐츠를 제거했습니다.

Orchestrator 가용성 및 확장성

Orchestrator 서비스의 가용성을 높이려면 클러스터의 여러 Orchestrator 서버 인스턴스를 공유 데이터베이스와 함께 시작합니다. vRealize Orchestrator는 클러스터의 일부로 작동하도록 구성되지 않는 한 단일 인스턴스로 작동합니다.

Orchestrator 클러스터

동일한 서버 및 플러그인 구성을 지닌 여러 Orchestrator 서버 인스턴스는 클러스터에서 함께 작동하며 단일 데이터베이스를 공유합니다.

모든 Orchestrator 서버 인스턴스는 하트비트를 교환하며 서로 간에 통신합니다. 각 하트비트는 노드가 특정 시간 간격으로 클러스터의 공유 데이터베이스에 기록하는 타임 스탬프입니다. 네트워크 문제, 무응답 데이터베이스 서버 또는 과부하로 Orchestrator 클러스터 노드의 응답이 중지될 수 있습니다. 활성 Orchestrator 서버 인스턴스가 페일오버 시간 초과 기간 내에 하트비트를 보내지 못하면 응답 없음으로 간주됩니다. 페일오버 시간 초과는 페일오버 하트비트 수와 하트비트 간격 값을 곱한 값과 동일합니다. 이는 신뢰할 수 없는 노드에 대한 정의로 사용되고 사용 가능한 리소스 및 운영 모드에 따라 사용자 지정할 수 있습니다.

데이터베이스와의 연결이 끊어지면 Orchestrator 노드는 대기 모드로 전환되고 데이터베이스 연결이 복구되기 전까지 해당 모드를 유지합니다. 클러스터의 다른 노드는 스크립팅 가능한 작업 또는 워크플로 호출과 같은 완료되지 않은 지난 항목에서 모든 중단된 워크플로를 재개하여 활성 작업을 제어합니다.

Orchestrator는 클러스터 상태 모니터링 및 페일오버 알림을 전송하는 내장된 도구를 제공하지 않습니다. 로드 밸런서와 같은 외부 구성 요소를 사용하여 클러스터 상태를 모니터링할 수 있습니다. 노드가 실행 중인지 확인하려면 https://your_orchestrator_server_IP_or_DNS_name:8281/vco/api/healthstatus에서 상태 REST API 서비스를 사용하여 노드의 상태를 확인하거나 https://your_orchestrator_server_IP_or_DNS_name:8283/vco-controlcenter/docs/에서 제어 센터의 상태를 모니터링할 수 있습니다.

VAMI에서 vRealize Orchestrator 인스턴스의 클러스터 구성

vRealize Orchestrator 7.5부터 Orchestrator 장치의 VAMI 인터페이스를 통해 모든 클러스터링 작업이 수행됩니다.

Orchestrator 클러스터는 하나의 데이터베이스를 공유하는 2개 이상의 Orchestrator 인스턴스로 구성됩니다. 새 Orchestrator 클러스터를 구성하거나 Orchestrator VAMI 인터페이스의 기존 클러스터에 새 노드를 추가합니다. Orchestrator 클러스터에는 세 가지 유형의 노드가 있습니다.

노드 유형	정의
기본 노드	각 Orchestrator 클러스터에는 기본 노드가 한 개 있습니다. 클러스터의 모든 노드는 기본 노드의 PostgreSQL 데이터베이스를 공유합니다. 기본 데이터베이스는 동기 모드 및 비동기 모드 모두에서 실행할 수 있습니다. 기본 노드는 클러스터가 작동하려면 정상적인 상태여야 합니다.
복제 노드	복제 노드는 기본 Orchestrator 노드에 조인하는 Orchestrator 인스턴스입니다.
동기화된 복제 노드	동기화 모드를 사용하도록 설정하면 복제 노드가 동기화된 복제 노드 상태로 승격됩니다. 동기화된 복제는 기본 노드의 자동 페일 오버를 사용하도록 설정합니다.

사전 요구 사항

- 독립형 서버 노드를 두 개 이상 구성합니다. 자세한 내용은 [독립형 Orchestrator 서버 구성](#) 항목을 참조하십시오.
- Orchestrator 인스턴스가 설치된 가상 시스템의 시계를 동기화합니다.
- 여러 Orchestrator 인스턴스에서 트래픽을 분배하기 위해 로드 밸런서를 설정합니다.

절차

- 1 대상 Orchestrator 환경의 VAMI 인터페이스에 **root**로 로그인합니다.
https://your_orchestrator_server_ip_or_DNS_name:5480에서 VAMI 인터페이스에 액세스합니다.
- 2 **클러스터** 탭을 선택하고 클러스터의 기본 노드가 될 Orchestrator 노드의 자격 증명을 입력합니다.
 기존의 클러스터링된 Orchestrator 환경의 경우, Orchestrator 클러스터의 기본 노드 자격 증명을 입력합니다.
- 3 **클러스터에 가입**을 클릭합니다.
- 4 노드의 인증서 정보를 검토하고 **확인**을 클릭합니다.
- 5 클러스터링 작업은 Orchestrator 노드의 콘텐츠를 동기화하고 복제 노드를 기본 노드의 PostgreSQL 데이터베이스에 조인시킵니다.

다음에 수행할 작업

Orchestrator 제어 센터의 **구성 검증** 페이지에서 클러스터가 올바르게 구성되었는지 확인합니다.

참고 클러스터 노드의 구성에 따라 2분 후에 Orchestrator 서버가 자동으로 다시 시작됩니다. 프로세스 완료 직후에 구성을 확인하면 잘못된 클러스터 상태가 반환될 수 있습니다.

Orchestrator 클러스터 모니터링

클러스터를 생성한 후에 클러스터 노드의 상태를 모니터링할 수 있습니다.

제어 센터의 **Orchestrator 클러스터 관리** 페이지에서 클러스터에 조인된 Orchestrator 인스턴스의 구성 동기화 상태를 모니터링할 수 있습니다.

구성 동기화 상태	설명
실행 중	Orchestrator 서비스를 사용할 수 있으며 요청을 수락할 수 있습니다.
대기	Orchestrator 서비스는 다음과 같은 이유로 요청을 처리할 수 없습니다. <ul style="list-style-type: none"> ■ 노드는 HA(고가용성) 클러스터의 일부이며 기본 노드에 장애가 발생할 때까지 대기 모드를 유지합니다. ■ 이 서비스는 데이터베이스, 인증 제공자 및 Orchestrator 인스턴스 라이선스에 대한 유효한 연결과 같은 구성 필수 요건을 확인할 수 없습니다.
서비스 상태 검색 실패	Orchestrator 서버 서비스가 중지되었거나 네트워크 문제가 있으므로 이 서비스에 연결할 수 없습니다.
다시 시작 대기 중	제어 센터는 구성 변경 사항을 감지하고 Orchestrator 서버는 자동으로 다시 시작됩니다.

Orchestrator 클러스터에 대해 동기화 모드 사용하도록 설정

Orchestrator 데이터베이스 클러스터를 동기화 모드에서 실행되도록 구성할 수 있습니다.

동기화 모드는 기본 Orchestrator 데이터베이스의 자동 페일오버를 사용하도록 설정합니다. 이 프로세스는 복제 노드 중 하나를 **동기화된 복제** 상태로 승격합니다. 현재 기본 노드가 실패할 경우, 동기화된 복제가 자동으로 기본 노드로 승격됩니다. 동기화된 복제가 기본 노드의 데이터베이스에서 완료된 트랜잭션을 모두 수신합니다.

사전 요구 사항

세 개 이상의 Orchestrator 노드로 구성된 Orchestrator 클러스터를 구성합니다.

절차

- 1 대상 Orchestrator 환경의 VAMI 인터페이스에 **root**로 로그인합니다.

`https://your_orchestrator_server_ip_or_DNS_name:5480`에서 VAMI 인터페이스에 액세스합니다.

- 2 **클러스터** 탭을 선택합니다.

- 3 **동기화 모드**를 클릭합니다.

- 4 클러스터의 노드 중 하나가 **동기화된 복제** 상태로 승격됩니다.

동기화 작업의 성공 여부를 확인하려면 **클러스터** 탭의 복제 모드 상태가 **데이터베이스가 동기 모드임**인지 확인합니다.

Orchestrator 복제 노드를 기본 노드로 승격

복제 노드를 기본 노드로 승격하여 Orchestrator 클러스터를 재구성할 수 있습니다.

Orchestrator 노드는 비동기 모드 및 동기화 모드 모두에서 승격될 수 있습니다.

참고 동기화 모드에서 Orchestrator 클러스터에는 자동 페일오버 기능이 있으므로 현재 기본 노드가 실패할 경우 동기화된 복제 노드는 자동으로 새 기본 노드가 됩니다.

사전 요구 사항

두 개 이상의 Orchestrator 인스턴스로 구성된 Orchestrator 클러스터를 구성합니다.

절차

- 1 대상 Orchestrator 환경의 VAMI 인터페이스에 **root**로 로그인합니다.
https://your_orchestrator_server_ip_or_DNS_name:5480에서 VAMI 인터페이스에 액세스합니다.
- 2 **클러스터** 탭을 선택합니다.
- 3 새 기본 노드의 상태로 승격할 복제 노드 옆에 있는 **승격**을 클릭합니다.
- 4 **새 마스터 노드로 승격됨** 메시지가 VAMI UI의 왼쪽 상단에 나타나고 노드의 상태가 **마스터**로 변경됩니다.

Orchestrator 클러스터 노드 삭제

Orchestrator 클러스터에서 Orchestrator 복제 노드를 삭제하여 복제 노드를 대체하거나 용량을 줄일 수 있습니다.

클러스터에서 복제 노드만 삭제할 수 있습니다. 기본 노드를 제거하려면 먼저 대체할 복제 노드를 승격시켜야 합니다. 자세한 내용은 [Orchestrator 복제 노드를 기본 노드로 승격](#) 항목을 참조하십시오.

절차

- 1 대상 Orchestrator 환경의 VAMI 인터페이스에 **root**로 로그인합니다.
https://your_orchestrator_server_ip_or_DNS_name:5480에서 VAMI 인터페이스에 액세스합니다.
- 2 **클러스터** 탭을 선택합니다.
- 3 복제 노드 옆에 있는 **삭제** 명령을 선택합니다.
- 4 클러스터에서 복제 노드를 삭제할지 확인한 후 **확인**을 클릭합니다.

참고 삭제된 복제 노드의 호스트 이름을 로드 밸런서 서버에서 제거해야 합니다.

- 5 Orchestrator 노드는 클러스터에서 삭제되며 **노드를 삭제했습니다**. 메시지가 UI의 왼쪽 상단에 나타납니다.

고객 경험 향상 프로그램 구성

고객 경험 향상 프로그램(CEIP) 참여를 선택하면 VMware에 VMware 제품 및 서비스의 품질, 안정성 및 기능을 개선할 수 있는 익명의 정보가 전달됩니다.

VMware에 수신되는 정보의 범주

CEIP(고객 환경 향상 프로그램)은 VMware가 해당 제품 및 서비스를 향상시키고 문제를 수정할 수 있도록 하는 정보를 VMware에 제공합니다.

CEIP를 통해 수집되는 데이터에 대한 세부 정보와 VMware에서 해당 정보를 사용하는 목적은 신뢰 및 보장 센터(<http://www.vmware.com/trustvmware/ceip.html>)에 명시되어 있습니다. 이 제품에 대한 CEIP에 가입하거나 탈퇴하려면 [CEIP\(고객 환경 향상 프로그램\) 참여](#)를 참조하십시오.

CEIP(고객 환경 향상 프로그램) 참여

제어 센터에서 고객 환경 향상 프로그램에 가입합니다.

절차

- 1 제어 센터에 **root**로 로그인하고 **고객 환경 향상 프로그램** 페이지를 엽니다.
- 2 **고객 환경 향상 프로그램 참여** 확인란을 선택하여 CEIP를 사용하도록 설정하거나 이 확인란의 선택을 취소하여 이 프로그램을 사용하지 않도록 설정한 다음 **저장**을 클릭합니다.
- 3 (선택 사항) 프록시 호스트를 수동으로 추가하려는 경우 **자동 프록시 검색** 확인란의 선택을 취소합니다.

제어 센터를 통한 Orchestrator 구성 외에도 장치에 저장된 Orchestrator REST API, 제어 센터 REST API 또는 명령줄 유틸리티를 사용하여 Orchestrator 서버 구성 설정을 수정할 수 있습니다.

구성 플러그인은 Orchestrator 패키지에 기본적으로 포함되어 있습니다. 구성 플러그인 워크플로는 Orchestrator 워크플로 라이브러리 또는 Orchestrator REST API를 통해 액세스할 수 있습니다. 이 워크플로를 통해 신뢰할 수 있는 인증서 및 Orchestrator 서버의 키 저장소 설정을 변경할 수 있습니다. 제공되는 모든 Orchestrator REST API 서비스 호출에 대한 자세한 내용은 https://orchestrator_서버_IP_또는_DNS_이름:8281/vco/api/docs에서 "Orchestrator REST API 참조" 설명서를 참조하십시오.

■ REST API를 사용하여 SSL 인증서 및 키 저장소 관리

제어 센터를 통한 SSL 인증서 관리 외에도 구성 플러그인에서 워크플로를 실행하거나 REST API를 사용하여 신뢰할 수 있는 인증서 및 키 저장소를 관리할 수도 있습니다.

■ 제어 센터 REST API를 사용하여 Orchestrator 구성 자동화

제어 센터 REST API는 Orchestrator 서버 구성을 위해 리소스에 대한 액세스를 제공합니다. 제어 센터 REST API를 타사 시스템과 사용하여 Orchestrator 구성을 자동화할 수 있습니다.

REST API를 사용하여 SSL 인증서 및 키 저장소 관리

제어 센터를 통한 SSL 인증서 관리 외에도 구성 플러그인에서 워크플로를 실행하거나 REST API를 사용하여 신뢰할 수 있는 인증서 및 키 저장소를 관리할 수도 있습니다.

구성 플러그인은 SSL 인증서 및 키 저장소 가져오기 및 삭제에 대한 워크플로를 포함합니다. 이러한 워크플로는 Orchestrator 클라이언트의 워크플로 보기에서 **라이브러리 > 구성 > SSL 신뢰 관리자 및 라이브러리 > 구성 > 키 저장소**로 이동하여 액세스할 수 있습니다. 또한 이러한 워크플로는 Orchestrator REST API를 사용하여 실행할 수도 있습니다.

REST API를 사용하여 SSL 인증서 삭제

구성 플러그인의 신뢰할 수 있는 인증서 삭제 워크플로를 실행하거나 REST API를 사용하여 SSL 인증서를 삭제할 수 있습니다.

절차

- 1 신뢰할 수 있는 인증서 삭제 워크플로의 워크플로 서비스 URL에서 GET 요청을 만듭니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows?conditions=name>Delete trusted certificate
```

- 2 정의의 URL에서 GET 요청을 만들어 신뢰할 수 있는 인증서 삭제 워크플로 정의를 검색합니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/8a70a326-ffd7-4fef-97e0-2002ac49f5bd
```

- 3 신뢰할 수 있는 인증서 삭제 워크플로의 실행 개체를 보유하는 URL에서 POST 요청을 만듭니다.

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/8a70a326-ffd7-4fef-97e0-2002ac49f5bd/executions/
```

- 4 요청 본문의 실행 컨텍스트 요소에 신뢰할 수 있는 인증서 삭제 워크플로의 입력 매개 변수로 삭제하려는 인증서 이름을 제공합니다.

REST API를 사용하여 SSL 인증서 가져오기

구성 플러그인에서 워크플로를 실행하거나 REST API를 사용하여 SSL 인증서를 가져올 수 있습니다.

파일 또는 URL에서 신뢰할 수 있는 인증서를 가져올 수 있습니다. 제어 센터를 사용하여 Orchestrator에서 인증서 가져오기에 대한 자세한 내용은 [Orchestrator 인증서 관리](#)를 참조하십시오.

절차

- 1 워크플로 서비스의 URL에서 GET 요청을 만듭니다.

옵션	설명
파일에서 신뢰할 수 있는 인증서 가져오기	파일에서 신뢰할 수 있는 인증서를 가져옵니다.
URL에서 신뢰할 수 있는 인증서 가져오기	URL 주소에서 신뢰할 수 있는 인증서를 가져옵니다.
프록시 서버를 사용하여 URL에서 신뢰할 수 있는 인증서 가져오기	프록시 서버를 사용하여 URL 주소에서 신뢰할 수 있는 인증서를 가져옵니다.
인증서 별칭이 포함된 URL에서 신뢰할 수 있는 인증서 가져오기	URL 주소에서 인증서 별칭이 있는 신뢰할 수 있는 인증서를 가져옵니다.

파일에서 신뢰할 수 있는 인증서를 가져오려면 다음 GET 요청을 만듭니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows?conditions=name=Import trusted certificate from a file
```

- 2 정의의 URL에서 GET 요청을 만들어 워크플로 정의를 검색합니다.

파일 워크플로에서 신뢰할 수 있는 인증서 가져오기에 대한 정의를 검색하려면 다음 GET 요청을 만듭니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/93a7bb21-0255-4750-9293-2437abe9d2e5
```

- 3 워크플로의 실행 개체를 보유하는 URL에서 POST 요청을 만듭니다.

파일 워크플로에서 신뢰할 수 있는 인증서를 가져오려면 다음 POST 요청을 만듭니다.

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/93a7bb21-0255-4750-9293-2437abe9d2e5/
executions
```

- 4 요청 본문의 실행 컨텍스트 요소에 워크플로의 입력 매개 변수에 대한 값을 제공합니다.

매개 변수	설명
cer	SSL 인증서를 가져오려는 CER 파일입니다. 이 매개 변수는 파일 워크플로에서 신뢰할 수 있는 인증서 가져오기에 적용됩니다.
url	SSL 인증서를 가져오려는 URL입니다. HTTPS가 아닌 서비스의 경우 지원되는 형식은 <i>IP_address_or_DNS_name:port</i> 입니다. 이 매개 변수는 URL 워크플로에서 신뢰할 수 있는 인증서 가져오기에 적용됩니다.

REST API를 사용하여 키 저장소 생성

구성 플러그인의 키 저장소 생성 워크플로를 실행하거나 REST API를 사용하여 키 저장소를 생성할 수 있습니다.

절차

- 1 키 저장소 생성 워크플로의 워크플로 서비스 URL에서 GET 요청을 만듭니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows?conditions=name=Create a keystore
```

- 2 정의의 URL에서 GET 요청을 만들어 키 저장소 생성 워크플로 정의를 검색합니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/6c301bff-e8fe-4ae0-ad08-5318178594b3/
```

- 3 키 저장소 생성 워크플로의 실행 개체를 보유하는 URL에서 POST 요청을 만듭니다.

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/6c301bff-e8fe-4ae0-ad08-5318178594b3/
executions/
```

- 4 요청 본문의 실행 컨텍스트 요소에 키 저장소 생성 워크플로의 입력 매개 변수로 생성하려는 키 저장소 이름을 제공합니다.

REST API를 사용하여 키 저장소 삭제

구성 플러그인의 키 저장소 삭제 워크플로를 실행하거나 REST API를 사용하여 키 저장소를 삭제할 수 있습니다.

절차

- 1 키 저장소 삭제 워크플로의 워크플로 서비스 URL에서 GET 요청을 만듭니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows?conditions=name>Delete a keystore
```

- 2 정의의 URL에서 GET 요청을 만들어 키 저장소 삭제 워크플로 정의를 검색합니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/7a3389eb-1fab-4d77-860b-81b66bb45b86/
```

- 3 키 저장소 삭제 워크플로의 실행 개체를 보유하는 URL에서 POST 요청을 만듭니다.

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/7a3389eb-1fab-4d77-860b-81b66bb45b86/
executions/
```

- 4 요청 본문의 실행 컨텍스트 요소에 키 저장소 삭제 워크플로의 입력 매개 변수로 삭제하려는 키 저장소 이름을 제공합니다.

REST API를 사용하여 키 추가

구성 플러그인의 키 추가 워크플로를 실행하거나 REST API를 사용하여 키를 추가할 수 있습니다.

절차

- 1 키 추가 워크플로의 워크플로 서비스 URL에서 GET 요청을 만듭니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows?conditions=name>Add key
```

- 2 정의의 URL에서 GET 요청을 만들어 키 추가 워크플로 정의를 검색합니다.

```
GET https://{orchestrator_host}:{port}/vco/api/workflows/6c301bff-e8fe-4ae0-ad08-5318178594b3/
```

- 3 키 추가 워크플로의 실행 개체를 보유하는 URL에서 POST 요청을 만듭니다.

```
POST https://{orchestrator_host}:{port}/vco/api/workflows/6c301bff-e8fe-4ae0-ad08-5318178594b3/
executions/
```

- 4 요청 본문의 실행 컨텍스트 요소에서 키 저장소, 키 별칭, PEM-인코드 키, 인증서 체인 및 키 암호를 키 추가 워크플로의 입력 매개 변수로 제공합니다.

제어 센터 REST API를 사용하여 Orchestrator 구성 자동화

제어 센터 REST API는 Orchestrator 서버 구성을 위해 리소스에 대한 액세스를 제공합니다. 제어 센터 REST API를 타사 시스템과 사용하여 Orchestrator 구성을 자동화할 수 있습니다.

제어 센터 REST API의 루트 끝점은 `https://orchestrator_server_IP_or_DNS_name:8283/vco-controlcenter/api`입니다. 제어 센터 REST API를 통해 수행할 수 있는 모든 서비스 호출에 대한 자세한 내용은 `https://orchestrator_server_IP_or_DNS_name:8283/vco-controlcenter/docs`에서 "제어 센터 REST API 참조" 설명서를 참조하십시오.

명령줄 유틸리티

Orchestrator 명령줄 유틸리티를 사용하여 Orchestrator 구성을 자동화할 수 있습니다.

SSH를 통해 Orchestrator Appliance에 루트로 로그인하여 명령줄 유틸리티에 액세스합니다. 유틸리티는 `/var/lib/vco/tools/configuration-cli/bin`에 위치합니다. 사용 가능한 구성 옵션을 보려면 `./vro-configure.sh --help`를 실행합니다.

추가 구성 옵션

7

제어 센터를 사용하여 기본 Orchestrator 동작을 변경할 수 있습니다.

본 장은 다음 항목을 포함합니다.

- 인증 재구성
- Orchestrator 구성 내보내기
- Orchestrator 구성 가져오기
- 워크플로 실행 속성 구성
- Orchestrator 로그 파일
- 네트워크 인터페이스 컨트롤러 추가
- 정적 경로 구성

인증 재구성

제어 센터의 초기 구성 중에 인증 방법을 설정한 후에는 언제든지 인증 제공자 또는 구성한 매개 변수를 변경할 수 있습니다.

인증 제공자 변경

인증 모드 또는 인증 제공자 연결 설정을 변경하려면, 기존의 인증 제공자를 먼저 등록 취소해야 합니다.

사전 요구 사항

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **인증 제공자 구성** 페이지에서 호스트 주소 텍스트 상자 옆에 있는 **등록 취소** 버튼을 클릭하여 사용 중인 인증 제공자의 등록을 취소합니다.
- 3 **ID 서비스** 섹션에서 **등록 취소**를 클릭하여 서버 자격 증명을 삭제합니다.

결과

인증 제공자를 성공적으로 등록 취소했습니다.

다음에 수행할 작업

제어 센터에서 인증을 재구성합니다. 자세한 내용은 [vRealize Automation 인증을 사용하여 독립형 Orchestrator 서버 구성](#) 또는 [vSphere 인증을 사용하여 독립형 Orchestrator 서버 구성](#)을 참조하십시오.

인증 매개 변수 변경

제어 센터에서 인증 제공자로 vRealize Automation을 사용하는 경우, Orchestrator 관리자 그룹의 기본 테넌트를 변경할 수 있습니다. vSphere 인증을 사용하는 경우, 관리자 그룹을 변경할 수 있습니다.

사전 요구 사항

- 제어 센터에 **루트**로 로그인합니다.
- 인증 모드를 선택하고 인증 제공자의 연결 설정을 구성합니다.

절차

- 1 기본 테넌트를 변경합니다.

참고 vRealize Automation 인증 모드를 사용하는 경우에만 기본 테넌트를 변경할 수 있습니다.

- a 제어 센터의 **인증 제공자 구성** 페이지에서 **변경** 버튼(**기본 테넌트** 텍스트 상자 옆에 있음)을 클릭합니다.
- b 텍스트 상자에서 기존의 기본 테넌트 이름을 사용하고 싶은 다른 이름으로 교체합니다.
- c **변경** 버튼(**관리자 그룹** 텍스트 상자 옆에 있음)을 클릭합니다.

참고 관리자 그룹을 재구성하지 않으면 비어 있게 되며 더 이상 제어 센터에 액세스할 수 없습니다.

- d 관리자 그룹의 이름을 입력하고 **검색**을 클릭합니다.
- e 그룹 목록에서 선택할 그룹의 이름을 두 번 클릭합니다.
- f **변경 내용 저장**을 클릭합니다.

제어 센터에서 로그아웃되고 Single Sign-On 로그인 화면으로 리디렉션됩니다.

- 2 관리자 그룹을 변경합니다.

- a **변경** 버튼(**관리자 그룹** 텍스트 상자 옆에 있음)을 클릭합니다.
- b 관리자 그룹의 이름을 입력하고 **검색**을 클릭합니다.
- c 그룹 목록에서 선택할 그룹의 이름을 두 번 클릭합니다.
- d **변경 내용 저장**을 클릭합니다.

제어 센터에서 로그아웃되고 Single Sign-On 로그인 화면으로 리디렉션됩니다.

Orchestrator 구성 내보내기

제어 센터에서는 Orchestrator 구성 설정을 로컬 파일로 내보낼 수 있는 메커니즘을 제공합니다. 이 메커니즘을 사용하여 시스템 구성에 대한 원하는 시점을 스냅샷을 생성하고 이 구성을 새 Orchestrator 인스턴스로 가져올 수 있습니다.

정기적으로 구성 설정을 내보내고 저장해야 하며, 특히 수정 사항이 있거나 유지 보수 작업을 수행하거나, 시스템을 업그레이드할 때는 더욱 그래야 합니다.

중요 내보낸 구성이 있는 파일은 중요한 관리 정보를 포함하고 있으므로 안전하게 유지해야 합니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **구성 내보내기/가져오기**를 클릭합니다.
- 3 내보낼 파일의 유형을 선택합니다.

참고 플러그인 구성에 암호화된 속성이 포함된 경우 **플러그인 구성 내보내기**를 선택할 때는 가져올 때 데이터의 암호를 성공적으로 해독할 수 있도록 **서버 구성 내보내기**도 선택해야 합니다.

- 4 (선택 사항) 구성 파일을 보호할 암호를 입력합니다.

나중에 구성을 가져올 때 이 암호를 사용합니다.

- 5 **내보내기(Export)**를 클릭합니다.

결과

Orchestrator에서 `orchestrator-config-export-hostname-dateReference.zip` 파일이 생성되고 로컬 시스템에 다운로드됩니다. 이 파일을 사용하여 시스템을 복제하거나 복원할 수 있습니다.

Orchestrator 구성 가져오기

시스템 오류가 발생하는 경우 Orchestrator를 다시 설치한 후 이전에 내보낸 시스템 구성을 복원할 수 있습니다.

Orchestrator 구성을 복제하기 위해 가져오기 절차를 사용하는 경우 새 vCenter Server 플러그인 ID가 생성되므로 vCenter Server 플러그인 구성이 유효하지 않게 되며 작동하지 않습니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **구성 내보내기/가져오기**를 클릭하고 **구성 가져오기** 탭으로 이동합니다.
- 3 이전 설치에서 내보낸 **.zip** 파일을 찾아 선택합니다.

참고 내보낸 구성 파일에 대한 기본 구문은 `orchestrator-config-export-hostname-dateofexport_timeofexport.zip`입니다.

4 (선택 사항) 구성을 내보낼 때 사용한 암호를 입력합니다.

암호를 사용해 구성을 내보내지 않았다면 이 단계는 필요하지 않습니다.

5 가져오기 유형을 선택합니다.

옵션	설명
내장형	vRealize Automation에 내장된 Orchestrator 인스턴스로 마이그레이션합니다.
외부	외부 Orchestrator로 마이그레이션합니다.
복제본	동일한 Orchestrator 인스턴스를 복제합니다.

6 가져오기를 클릭합니다.

결과

새 시스템은 선택한 가져오기 유형을 기준으로 이전 구성을 복제합니다. Orchestrator 서버 서비스가 자동으로 다시 시작됩니다.

다음에 수행할 작업

제어 센터의 **구성 검증** 페이지에서 Orchestrator가 올바르게 구성되었는지 확인합니다.

워크플로 실행 속성 구성

기본적으로 노드별로 최대 300개의 워크플로를 실행할 수 있으며 현재 실행 중인 워크플로의 개수가 이에 도달하면 최대 10,000개의 워크플로를 대기열에 추가할 수 있습니다.

Orchestrator 노드가 동시에 300개 이상의 워크플로를 실행해야 하는 경우 보류 중인 워크플로는 대기열에 추가됩니다. 활성 워크플로 실행이 완료되면 대기열의 다음 워크플로가 실행되기 시작합니다. 대기 중인 워크플로가 최대 개수에 도달하면 보류 중인 워크플로 중 하나가 실행되기 전까지는 다음 워크플로 실행이 실패합니다.

제어 센터의 **고급 옵션** 페이지에서 워크플로 실행 속성을 구성할 수 있습니다.

옵션	설명
안전 모드 사용	안전 모드를 사용하면 실행 중인 모든 워크플로가 취소되고 다음 Orchestrator 노드 시작 시 재개되지 않습니다.
동시 실행 워크플로 개수	동시에 실행되는 Orchestrator 노드 워크플로의 최대 개수입니다.
대기열에서 실행 중인 워크플로의 최대 개수	사용할 수 없는 상태가 되기 전까지 Orchestrator 노드가 수락하는 워크플로 실행 요청의 개수입니다.
워크플로별로 유지되는 실행의 최대 개수	완료된 워크플로 실행의 최대 개수는 클러스터의 워크플로마다 각각 기록으로 유지됩니다. 개수를 초과하면 가장 오래된 워크플로 실행이 삭제됩니다.
로그 이벤트 만료 일수	로그 이벤트가 제거되기 전에 데이터베이스에서 유지되는 클러스터에 대한 로그 이벤트의 기간(일)입니다.

옵션	설명
모든 워크플로 실행 프로파일링	자동 워크플로 프로파일링을 사용하도록 또는 사용하지 않도록 설정합니다. 사용하도록 설정한 경우 워크플로 프로파일링은 모든 워크플로 실행에 대한 메트릭 데이터를 생성합니다.
워크플로 프로파일러 통계를 배분할 간격	프로파일러 통계가 사용자 환경의 모든 Orchestrator 인스턴스에 배분되는 간격입니다.

Orchestrator 로그 파일

VMware 기술 지원은 사용자의 지원 요청이 제출되면 정기적으로 진단 정보를 요청합니다. 해당 제품이 실행되는 호스트에서 진단된 이 정보에는 제품 관련 로그 및 구성 파일이 포함됩니다.

제어 센터의 **로그 내보내기** 메뉴에서 Orchestrator 구성 파일 및 로그 파일을 포함하는 ZIP 번들을 다운로드할 수 있습니다.

표 7-1. Orchestrator 로그 파일 목록

파일 이름	위치	설명
scripting.log	/var/log/vco/app-server	워크플로 및 작업의 스크립팅 로그 메시지를 제공합니다. <code>scripting.log</code> 파일을 사용하여 워크플로 실행 및 작업 실행을 일반 Orchestrator 작업에서 분리합니다. 이 정보는 <code>server.log</code> 파일에도 포함됩니다.
server.log	/var/log/vco/app-server	Orchestrator 서버의 모든 작업에 대한 정보를 제공합니다. Orchestrator 또는 Orchestrator에서 실행하는 모든 애플리케이션을 디버깅할 때 <code>server.log</code> 파일을 분석합니다.
metrics.log	/var/log/vco/app-server	서버에 대한 런타임 정보를 포함합니다. 해당 정보는 5분마다 로그 파일에 추가됩니다.
localhost_access_log.txt	/var/log/vco/app-server	서버의 HTTP 요청 로그입니다.
localhost_access_log.date.txt	/var/log/vco/configuration	제어 센터 서비스의 HTTP 요청 로그입니다.
controlcenter.log	/var/log/vco/configuration	제어 센터 서비스의 로그 파일입니다.

로깅 지속성

모든 종류의 Orchestrator 스크립트(예: 워크플로, 정책 또는 작업)에서 정보를 기록할 수 있습니다. 정보에는 유형 및 수준이 있습니다. 유형은 영구 또는 비영구로 구분할 수 있습니다. 수준은 `DEBUG`, `INFO`, `WARN`, `ERROR`, `TRACE` 및 `FATAL`로 구분됩니다.

표 7-2. 영구 및 비영구 로그 생성

로그 수준	영구 유형	비영구 유형
DEBUG	Server.debug("short text", "long text");	System.debug("text");
INFO	Server.log("short text", "long text");	System.log("text");
WARN	Server.warn("short text", "long text");	System.warn("text");
ERROR	Server.error("short text", "long text");	System.error("text");

영구 로그

영구 로그(서버 로그)는 이전 워크플로 실행 로그를 추적하고 Orchestrator 데이터베이스에 저장됩니다. 서버 로그를 보려면 워크플로, 완료된 워크플로 실행 또는 정책을 선택하고 Orchestrator 클라이언트의 **이벤트** 탭을 클릭해야 합니다.

비영구 로그

비영구 로그(시스템 로그)를 사용하여 스크립트를 생성하는 경우 Orchestrator 서버는 실행 중인 모든 Orchestrator 애플리케이션에 해당 로그에 대해 알리지만 이 정보는 데이터베이스에 저장되지 않습니다. 애플리케이션이 다시 시작되면 로그 정보는 손실됩니다. 비영구 로그는 실시간 정보 및 디버깅 용도로 사용됩니다. 시스템 로그를 보려면 Orchestrator 클라이언트의 완료된 워크플로 실행을 선택하고 **스키마** 탭에서 **로그**를 클릭해야 합니다.

Orchestrator 로그 구성

제어 센터의 **로그 구성** 페이지에서 필요한 서버 로그 및 스크립팅 로그의 수준을 설정할 수 있습니다. 하루에 두 로그 중 하나가 여러 번 생성되는 경우 문제가 발생하는 원인을 찾기 어렵습니다.

서버 로그 및 스크립팅 로그의 기본 로그 수준은 정보입니다. 로그 수준을 변경하면 서버가 로그에 입력되는 모든 새 메시지와 데이터베이스의 활성 연결 수에 영향을 줍니다. 로깅의 자세한 정도는 내림차순으로 감소합니다.

경고 로그 수준을 디버그 또는 모두로 설정해야만 문제를 디버그할 수 있습니다. 성능에 심각한 영향을 줄 수 있기 때문에 운영 환경에서는 이러한 설정을 사용하지 마십시오.

로그 순환 설정

서버 로그가 너무 커지는 것을 방지하려면 **최대 파일 개수** 및 **최대 파일 크기(MB)** 텍스트 상자의 값을 변경하여 서버 로그의 최대 파일 크기 및 개수를 설정할 수 있습니다.

Orchestrator 로그 파일 내보내기

제어 센터의 **로그 내보내기** 페이지에서 구성, 서버, 래퍼 및 설치 로그 파일을 포함하는 문제 해결 정보의 ZIP 아카이브를 생성할 수 있습니다.

로그 정보는 `vco-logs-date_hour.zip`으로 명명된 ZIP 아카이브에 저장됩니다.

참고 클러스터에 둘 이상의 Orchestrator 인스턴스를 보유한 경우, ZIP 아카이브는 클러스터에 있는 모든 Orchestrator 인스턴스의 로그를 포함합니다.

Orchestrator 로그 필터링

특정 워크플로 실행에 대한 Orchestrator 서버 로그를 필터링하고 워크플로 실행에 대한 진단 데이터를 수집할 수 있습니다.

Orchestrator 로그에는 실시간으로 모니터링할 수 있는 많은 유용한 정보가 들어 있습니다. 동일한 워크플로의 여러 인스턴스가 동시에 실행 중인 경우, Orchestrator 라이브 로그 스트림에서 각 실행에 대한 진단 데이터를 필터링하여 다양한 워크플로 실행을 추적할 수 있습니다.

참고 클러스터에 둘 이상의 Orchestrator 인스턴스가 있는 경우, 라이브 로그 스트림은 로컬 Orchestrator 노드의 로그만 표시합니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **라이브 로그 스트림**을 클릭합니다.
- 3 검색 창에서 검색 매개 변수를 입력합니다.
예를 들어, 사용자 이름, 워크플로 이름, 워크플로 ID 또는 토큰 ID별로 로그를 필터링할 수 있습니다.
- 4 (선택 사항) 검색 결과를 상세하게 필터링하려면 **대소문자 구분**과 **필터(grep)**를 선택합니다.
필터(grep)를 선택하면 라이브 스트림에서 검색 매개 변수와 일치하는 행만 표시합니다.

결과

Orchestrator 라이브 로그 스트림은 검색 매개 변수에 따라 필터링됩니다.

다음에 수행할 작업

제어 센터의 **라이브 로그 스트림** 페이지를 통해 액세스할 수 없는 이전 로그를 필터링하려면 타사 로그 분석 도구를 사용할 수 있습니다.

원격 서버와 로깅 통합 구성

vRealize Log Insight 또는 기타 Syslog 서버와 같은 원격 로깅 시스템에 로그를 보내도록 Orchestrator를 구성할 수 있습니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **로깅 통합** 메뉴로 이동합니다.
- 3 **원격 로그 서버로의 로그 기록 사용**을 설정합니다.
- 4 로깅 통합 옵션을 구성합니다.
 - a 로깅 시스템 유형을 선택합니다.
 - b 원격 로깅 서버의 호스트 이름 및 포트 값을 입력합니다.
 - c 원격 로깅 서버에 로그 이벤트를 전송하는 데 사용되는 프로토콜을 선택합니다.
- 5 원격 서버에 대한 로깅 통합 구성을 완료하려면 **저장**을 클릭합니다.

네트워크 인터페이스 컨트롤러 추가

vRealize Orchestrator는 여러 개의 NIC(네트워크 인터페이스 컨트롤러)를 지원합니다. 설치 후에 Orchestrator 장치에 NIC를 추가할 수 있습니다.

사전 요구 사항

vCenter Server 환경에 vRealize Orchestrator를 완전하게 설치합니다.

절차

- 1 vCenter Server에서 각 vRealize Orchestrator 장치에 NIC를 추가합니다.
 - a 장치를 마우스 오른쪽 버튼으로 클릭하고 **설정 편집**을 선택합니다.
 - b VMXNET3 NIC를 추가합니다.
 - c 전원이 켜져 있으면 장치를 다시 시작합니다.
- 2 vRealize Orchestrator 장치 관리 인터페이스에 root로 로그인합니다.
<https://orchestrator-appliance-IP:5480>
- 3 **네트워크**를 선택하고 여러 NIC를 사용할 수 있는지 확인합니다.
- 4 **주소**를 선택하고 NIC의 IP 주소를 구성합니다.

표 7-3. NIC 구성의 예

설정	값
IPv4 주소 유형	정적
IPv4 주소	172.22.0.2
넷마스크	255.255.255.0

- 5 **설정 저장**을 클릭합니다.

정적 경로 구성

vRealize Orchestrator 설치에 NIC를 추가할 때 정적 경로가 필요하다면 명령 프롬프트 세션을 열어서 구성합니다.

사전 요구 사항

vRealize Orchestrator 장치에 여러 개의 NIC를 추가합니다.

절차

- 1 vRealize Orchestrator 장치 명령줄에 root로 로그인합니다.
- 2 텍스트 편집기에서 경로 파일을 엽니다.

```
/etc/sysconfig/network/routes
```

- 3 기본 게이트웨이에 대한 **default** 줄을 찾고 수정하지는 않습니다.

참고 기본 게이트웨이를 변경해야 하는 경우에는 vRealize Orchestrator 관리 인터페이스를 대신 사용합니다.

- 4 **default** 줄 아래에 정적 경로에 대한 새 줄을 추가합니다. 예:

```
default 10.10.10.1 --  
172.30.30.0 192.168.100.1 255.255.255.0 eth0  
192.168.210.0 192.168.230.1 255.255.255.0 eth2
```

- 5 경로 파일을 저장하고 닫습니다.
- 6 장치를 다시 시작합니다.
- 7 HA 클러스터에서 각 장치에 대해 이 과정을 반복합니다.

구성 사용 사례 및 문제 해결

8

Orchestrator 서버가 vCenter Server 장치와 함께 작동하도록 구성할 수 있습니다. 또한 Orchestrator에서 플러그인을 제거하거나 자체 서명된 인증서를 변경할 수도 있습니다.

구성 사용 사례는 해결 방법이 있는 경우 문제를 이해하고 해결할 수 있는 문제 해결 주제뿐 아니라 Orchestrator 서버에 대한 특정 구성 요구 사항을 충족하도록 수행할 수 있는 작업 흐름을 제공합니다.

본 장은 다음 항목을 포함합니다.

- [vSphere Web Client용 vRealize Orchestrator 플러그인 구성](#)
- [Orchestrator 인증 등록 취소](#)
- [SSL 인증서 변경](#)
- [실행 중인 워크플로 취소](#)
- [Orchestrator 서버 디버깅 사용](#)
- [Orchestrator 구성 및 요소 백업](#)
- [vRealize Orchestrator 백업 및 복원](#)
- [Site Recovery Manager를 사용한 Orchestrator의 재해 복구](#)

vSphere Web Client용 vRealize Orchestrator 플러그인 구성

vSphere Web Client용 vRealize Orchestrator 플러그인을 사용하려면 vRealize Orchestrator를 vCenter Server의 확장으로 등록해야 합니다.

vRealize Orchestrator 서버를 vCenter Single Sign-On에 등록하고 vCenter Server와 작동하도록 구성한 후, vRealize Orchestrator를 vCenter Server의 확장으로 등록해야 합니다.

사전 요구 사항

관리되는 vCenter Server가 인증에 사용하는 것과 동일한 Platform Services Controller에 vSphere 인증을 사용하여 vRealize Orchestrator를 등록해야 합니다.

절차

- 1 vRealize Orchestrator 클라이언트에 로그인합니다.

- 2 **라이브러리 > 워크플로**로 이동합니다.
- 3 **vCenter Orchestrator**를 **vCenter Server 확장으로 등록** 워크플로를 검색하여 **실행**을 클릭합니다.
- 4 vRealize Orchestrator를 등록할 vCenter Server 인스턴스를 선택합니다.
- 5 (선택 사항) `https://your_orchestrator_hostname:8281` 또는 요청을 vRealize Orchestrator 서버 노드로 리디렉션하는 로드 밸런서의 서비스 URL을 입력합니다.
- 6 **실행**을 클릭합니다.

Orchestrator 인증 등록 취소

제어 센터의 인증 제공자 구성 페이지에서 Single Sign-On 솔루션으로 Orchestrator를 등록 취소합니다.

Orchestrator vCenter Single Sign-On 또는 vRealize Automation 인증을 다시 구성하려면 먼저 Orchestrator 인증을 등록 취소해야 합니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **인증 제공자 구성**을 클릭합니다.
- 3 **등록 취소**를 클릭합니다.
- 4 (선택 사항) ID 서버에서 등록 데이터를 삭제하려면 자격 증명을 입력합니다.
- 5 **ID 서비스** 섹션에서 **등록 취소**를 클릭합니다.

결과

Orchestrator 서버 인스턴스를 성공적으로 등록 취소했습니다.

SSL 인증서 변경

기본적으로 Orchestrator 서버는 자체 서명된 SSL 인증서를 사용하여 Orchestrator 클라이언트와 원격으로 통신합니다. 회사 보안 정책에 따라 회사의 SSL 인증서를 사용해야 하는 경우 SSL 인증서를 변경할 수 있습니다.

신뢰할 수 있는 SSL Internet 연결을 통해 Orchestrator를 사용하려고 시도하고 웹 브라우저에서 제어 센터를 열면 Mozilla Firefox를 사용하는 경우 연결을 신뢰할 수 없다는 경고가 표시되거나 Internet Explorer를 사용하는 경우 웹 사이트의 보안 인증서에 문제가 있다는 경고가 표시됩니다.

이 웹 사이트를 계속 탐색합니다(권장하지 않음)을 클릭하면 신뢰할 수 있는 저장소의 SSL 인증서를 가져왔다 해도 웹 브라우저의 주소 표시줄에 적색의 인증서 오류 알림이 지속적으로 표시됩니다. 웹 브라우저에서 Orchestrator를 사용할 수 있으나 타사 시스템이 HTTPS를 통해 API에 액세스하려는 경우 올바르게 작동하지 않을 수 있습니다.

또한 Orchestrator 클라이언트를 시작하고 SSL 연결을 통해 Orchestrator 서버에 연결하려고 시도하면 인증서 경고가 표시될 수 있습니다.

이 문제는 상용 인증 기관에서 서명한 인증서를 설치하여 해결할 수 있습니다. Orchestrator 클라이언트에서 인증서 경고가 표시되지 않도록 하려면 루트 CA 인증서를 Orchestrator 클라이언트가 설치되어 있는 시스템의 Orchestrator 키 저장소에 추가합니다.

로컬 저장소에 인증서 추가

CA에서 인증서를 받은 후 이를 로컬 저장소에 추가해야 인증서 경고 또는 오류 메시지가 없이 제어 센터가 작동됩니다.

이 워크플로는 Internet Explorer를 사용하여 인증서를 로컬 저장소에 추가하는 절차를 설명합니다.

- 1 Internet Explorer를 열고 `https://orchestrator_server_IP_or_DNS_name:8283/`로 이동합니다.
- 2 메시지가 표시되면 **이 웹 사이트를 계속 탐색합니다(권장하지 않음)**를 클릭합니다.
인증서 오류가 Internet Explorer 주소 표시줄의 오른쪽에 표시됩니다.
- 3 인증서 오류를 클릭하고 **인증서 보기**를 선택합니다.
- 4 **인증서 설치**를 클릭합니다.
- 5 **인증서 가져오기 마법사**의 시작 페이지에서 **다음**을 클릭합니다.
- 6 **인증서 저장소** 창에서 **모든 인증서를 다음 저장소에 배치**를 선택합니다.
- 7 **신뢰할 수 있는 루트 인증 기관**을 찾아 선택합니다.
- 8 마법사를 완료하고 Internet Explorer를 다시 시작합니다.
- 9 SSL 연결을 통해 Orchestrator 서버로 이동합니다.

더 이상 주의가 나타나지 않으며 주소 표시줄에 인증서 오류 메시지가 표시되지 않습니다.

VMware Service Manager와 같은 기타 애플리케이션 및 시스템은 SSL 연결을 통해 Orchestrator REST API에 액세스할 수 있어야 합니다.

Orchestrator Appliance 관리 사이트의 인증서 변경

Orchestrator Appliance에서는 Light HTTPd를 사용하여 고유한 관리 사이트를 실행합니다. 회사 보안 정책에 따라 회사의 SSL 인증서를 사용해야 하는 경우 Orchestrator Appliance 관리 사이트의 SSL 인증서를 변경할 수 있습니다.

사전 요구 사항

기본적으로 Orchestrator Appliance SSL 인증서와 개인 키는 `/opt/vmware/etc/lighttpd/server.pem`에 있는 PEM 파일에 저장됩니다. 새 인증서를 설치하려면 새 SSL 인증서 및 개인 키를 Java 키 저장소에서 PEM 파일로 내보내십시오.

절차

- 1 Orchestrator Appliance Linux 콘솔에 root로 로그인합니다.
- 2 `/opt/vmware/etc/lighttpd/lighttpd.conf` 파일을 찾아 편집기에서 엽니다.

3 다음 줄을 찾습니다.

```
#### SSL engine
ssl.engine = "enable"
ssl.pemfile = "/opt/vmware/etc/lighttpd/server.pem"
```

4 ssl.pemfile 특성이 새 SSL 인증서 및 개인 키를 포함하는 PEM 파일을 지정하도록 변경합니다.

5 lighttpd.conf 파일을 저장합니다.

6 다음 명령을 실행하여 light-httpd 서버를 다시 시작합니다.

```
service vami-lighttpd restart
```

결과

Orchestrator Appliance 관리 사이트의 인증서가 변경되었습니다.

실행 중인 워크플로 취소

제어 센터를 사용하여 제대로 완료되지 않은 워크플로를 취소할 수 있습니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **문제 해결**을 클릭합니다.
- 3 실행 중인 워크플로를 취소합니다.

옵션	설명
모든 워크플로 실행 취소	해당 워크플로에 대한 모든 토큰을 취소하려면 워크플로 ID를 입력합니다.
ID별로 워크플로 실행 취소	취소하려는 모든 토큰 ID를 입력합니다. 각 ID는 쉼표로 구분합니다.
실행 중인 모든 워크플로 취소	서버에서 실행 중인 모든 워크플로를 취소합니다.

참고 실행 스레드를 즉시 취소하는 신뢰할 만한 방법이 없으므로 ID별로 워크플로를 취소하는 작업은 실패할 수 있습니다.

결과

다음 서버 시작 시, 워크플로는 취소된 상태로 설정됩니다.

다음에 수행할 작업

제어 센터의 **워크플로 검사** 페이지에서 워크플로가 취소되었는지 확인합니다.

Orchestrator 서버 디버깅 사용

Orchestrator 서버를 디버그 모드에서 시작하여 플러그인 개발 시 문제를 디버깅할 수 있습니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **Orchestrator 디버깅**을 클릭합니다.
- 3 **디버그 사용**을 클릭합니다.
- 4 (선택 사항) 기본 포트와 다른 포트를 입력합니다.
- 5 (선택 사항) **일시 중단**을 클릭합니다.

이 옵션을 선택하면 Orchestrator 서버를 시작하기 전에 디버거를 연결해야 합니다.

- 6 **저장**을 클릭합니다.
- 7 제어 센터에서 시작 옵션 페이지를 열고 **다시 시작**을 클릭합니다.

결과

Orchestrator 서버는 시작 시 지정된 포트에 원격 Java 디버거를 연결하기 전까지 일시 중단됩니다.

Orchestrator 구성 및 요소 백업

사용자 지정 Orchestrator 서버 구성 및 워크플로 요소를 다른 Orchestrator 인스턴스에서 재사용할 수 있도록 백업합니다.

표준 워크플로, 작업, 정책 또는 구성 요소를 편집한 다음, 이후 Orchestrator 버전 번호의 동일한 요소를 포함하는 패키지를 가져오는 경우, 요소에 대한 변경 내용이 손실됩니다. Orchestrator 인스턴스를 마이그레이션하기 전에 사용자 지정된 워크플로 및 기타 요소를 내보내 손실되지 않도록 방지할 수 있습니다.

각 Orchestrator 서버 인스턴스에는 고유한 인증서가 있으며 각 vCenter Server 플러그인 인스턴스에는 고유한 ID가 있습니다. 이 인증서 및 고유한 ID는 Orchestrator 서버와 vCenter Server 플러그인의 ID를 정의합니다. Orchestrator 요소를 백업하지 않거나 백업 목적으로 Orchestrator 구성을 내보내지 않는 경우, 이 식별자를 변경했는지 확인하십시오.

사전 요구 사항

새로운 Orchestrator 서버 인스턴스를 배포하고 구성합니다. [독립형 Orchestrator 서버 구성](#)을 참조하십시오.

절차

- 1 Orchestrator 구성을 내보냅니다.
 - a 제어 센터에 **root**로 로그인합니다.
 - b **구성 내보내기/가져오기**를 클릭합니다.
 - c 내보낼 파일의 유형을 선택합니다.

- d (선택 사항) 암호를 입력하여 구성 파일을 보호합니다.
구성을 가져올 때 동일한 암호를 사용합니다.
 - e **내보내기**를 클릭합니다.
- 2** Orchestrator 클라이언트 애플리케이션에 로그인합니다.
- 3** 생성했거나 편집한 모든 Orchestrator 요소를 포함하는 패키지를 생성합니다.
- a **관리** 보기 아래에서 **패키지** 탭을 클릭합니다.
 - b 패키지 목록의 제목 표시줄에서 메뉴 버튼을 클릭하고 **패키지 추가**를 선택합니다.
 - c 새 패키지 이름을 입력하고 **확인**을 클릭합니다.
패키지 이름에 대한 구문은 *domain.your_company.folder.package_name*입니다.
예: *com.vmware.myfolder.mypackage*.
 - d 패키지를 마우스 오른쪽 버튼으로 클릭하고 **편집**을 선택합니다.
 - e **일반** 탭에서 패키지에 대한 설명을 추가합니다.
 - f **워크플로** 탭에서 패키지에 워크플로를 추가합니다.
 - g (선택 사항) 정책 템플릿, 작업, 구성 요소, 리소스 요소, 액세스 권한 및 플러그인을 패키지에 추가합니다.
 - h **저장 후 닫기**를 클릭합니다.
- 4** 패키지를 내보냅니다.
- a 내보내고 싶은 패키지를 마우스 오른쪽 버튼으로 클릭하고 **패키지 내보내기**를 선택합니다.
 - b 패키지를 저장하고 싶은 위치를 검색하여 선택합니다.
 - c (선택 사항) 해당하는 인증서를 사용하여 패키지에 서명합니다.
 - d (선택 사항) 내보낸 패키지에 제한 사항을 적용합니다.
 - e (선택 사항) 내보낸 패키지의 콘텐츠에 대해 제한 사항을 적용하려면 필요에 따라 옵션을 선택 취소합니다.

옵션	설명
버전 기록 내보내기	패키지의 버전 기록을 내보낼 수 없습니다.
구성 설정 값 내보내기	패키지 구성 요소의 특성 값을 내보낼 수 없습니다.
글로벌 태그 내보내기	패키지의 글로벌 태그를 내보낼 수 없습니다.

참고 구성 **SecureString** 설정 값 내보내기 옵션은 기본적으로 선택 취소되어 있습니다. 이러한 구성 설정을 내보내면 보안 문제가 야기될 수 있습니다. 따라서 주의해서 사용하십시오.

- f **저장**을 클릭합니다.

- 5 이전에 내보낸 Orchestrator 구성을 새 Orchestrator 서버 인스턴스로 가져옵니다.
 - a 새 Orchestrator 인스턴스의 제어 센터에 **root**로 로그인합니다.
 - b **구성 내보내기/가져오기**를 클릭하고 **구성 가져오기** 탭으로 이동합니다.
 - c 이전 설치에서 내보낸 **.zip** 파일을 찾아 선택합니다.
 - d 구성을 내보낼 때 사용한 암호를 입력합니다.
암호를 지정하지 않았다면 이 단계는 필요하지 않습니다.
 - e 가져오기 유형을 선택합니다.
 - f **가져오기**를 클릭합니다.

- 6 새로운 Orchestrator 인스턴스에 내보낸 패키지를 가져옵니다.
 - a 새로운 Orchestrator 인스턴스의 Orchestrator 클라이언트 애플리케이션에 로그인합니다.
 - b Orchestrator 클라이언트의 드롭다운 메뉴에서 **관리**를 선택합니다.
 - c **패키지** 탭을 클릭합니다.
 - d 패키지 목록의 제목 표시줄에서 메뉴 버튼을 클릭하고 **패키지 가져오기**를 선택합니다.
 - e 가져오고 싶은 패키지를 찾고 선택한 다음 **열기**를 클릭합니다.
내보내기에 대한 인증서 정보가 표시됩니다.
 - f 패키지 가져오기 세부 정보를 검토하고 **가져오기** 또는 **가져오기 및 제공자 신뢰**를 선택합니다.
패키지 가져오기 보기가 표시됩니다. 가져온 패키지 요소의 버전이 서버에 있는 버전보다 상위 버전인 경우 시스템이 가져올 요소를 자동으로 선택합니다.
 - g 가져올 요소를 선택합니다.

참고 최신 버전이 존재하는 사용자 지정 요소는 선택 해제합니다.

 - h (선택 사항) 패키지의 구성 요소 특성 값을 가져오지 않으려는 경우 **구성 설정 값 가져오기** 확인란의 선택을 해제합니다.
 - i 드롭다운 메뉴에서 패키지의 태그를 가져올지 여부를 선택합니다.

옵션	설명
태그 가져오기 및 기존 값 유지	기존 태그 값을 덮어쓰지 않고 패키지에서 태그를 가져옵니다.
태그 가져오기 및 기존 값 덮어쓰기	패키지에서 태그를 가져오고 값을 덮어씁니다.
태그 가져오지 않음	패키지에서 태그를 가져오지 않습니다.

- j **선택한 요소 가져오기**를 클릭합니다.

결과

Orchestrator 구성 및 요소가 백업되었습니다.

vRealize Orchestrator 백업 및 복원

vSphere Data Protection을 사용하여 vRealize Orchestrator 인스턴스를 포함하는 가상 시스템(VM)을 백업 및 복원할 수 있습니다.

vSphere Data Protection은 vSphere 환경에서 사용할 수 있는 VMware 디스크 기반 백업 및 복원 솔루션입니다. vSphere Data Protection은 vCenter Server와 완전히 통합됩니다. vSphere Data Protection을 통해 백업 작업을 관리하고 중복 제거된 대상 저장 위치에 백업을 저장할 수 있습니다.

vSphere Data Protection을 배포 및 구성한 후 vSphere Web Client 인터페이스를 통해 vSphere Data Protection에 액세스하여 백업 및 가상 시스템의 복구를 선택, 스케줄링, 구성 및 관리할 수 있습니다.

vSphere Data Protection는 백업을 수행할 때 가상 시스템에 대한 중지 상태의 스냅샷을 생성합니다. 모든 백업 작업에 중복 제거도 자동으로 수행됩니다.

vSphere Data Protection 배포 및 구성 방법에 대한 자세한 내용은 "vSphere Data Protection 관리" 설명서를 참조하십시오.

vRealize Orchestrator 백업

vRealize Orchestrator 인스턴스를 가상 시스템으로 백업할 수 있습니다.

단일 제품의 모든 VM 구성 요소가 함께 백업되도록 하려면 vRealize Orchestrator 환경의 VM을 단일 vCenter Server 폴더에 저장하고 해당 폴더에 대한 백업 정책 작업을 생성합니다.

사전 요구 사항

- vSphere Data Protection 장치가 배포되고 구성되었는지 확인합니다. vSphere Data Protection 배포 및 구성 방법에 대한 자세한 내용은 "vSphere Data Protection 관리" 설명서를 참조하십시오.
- vSphere Web Client를 사용하여 환경을 관리하는 vCenter Server 인스턴스에 로그인합니다. vSphere Data Protection 구성 중 사용된 관리자 권한이 있는 사용자로 로그인합니다.

절차

- 1 vSphere Web Client 홈 페이지에서 **vSphere Data Protection**을 클릭합니다.
- 2 **VDP 장치** 드롭다운 메뉴에서 vSphere Data Protection 장치를 선택하고 **연결**을 클릭합니다.
- 3 **시작** 탭에서 **백업 작성 작업**을 클릭합니다.
- 4 **게스트 이미지**를 클릭하여 vRealize Orchestrator 인스턴스를 백업하고 **다음**을 클릭합니다.
- 5 **전체 이미지**를 선택하여 전체 가상 시스템을 백업하고 **다음**을 클릭합니다.
- 6 **가상 시스템** 트리를 확장하고 vRealize Orchestrator VM의 확인란을 선택합니다.
- 7 프롬프트 메시지에 따라 백업 스케줄, 보존 정책 및 백업 작업의 이름을 설정합니다.

가상 시스템의 백업 및 복원 방법에 대한 자세한 내용은 "vSphere Data Protection 관리" 설명서를 참조하십시오.

백업 작업이 **백업** 탭의 백업 작업 목록에 표시됩니다.

- 8 (선택 사항) **백업** 탭을 열고 해당 백업 작업을 선택한 다음 **지금 백업**을 클릭하여 vRealize Orchestrator를 백업할 수 있습니다.

참고 또는, 설정한 스케줄에 따라 자동으로 백업이 시작되도록 대기할 수 있습니다.

백업 프로세스가 **최근 작업** 페이지에 표시됩니다.

결과

VM 이미지가 **복원** 탭의 백업 목록에 표시됩니다.

다음에 수행할 작업

복원 탭을 열고 VM 이미지가 백업 목록에 있는지 확인합니다.

vRealize Orchestrator 인스턴스 복원

원래 위치 또는 동일한 vCenter Server의 다른 위치에서 vRealize Orchestrator 인스턴스를 복원할 수 있습니다.

사전 요구 사항

- vSphere Data Protection 장치가 배포되고 구성되었는지 확인합니다. vSphere Data Protection 배포 및 구성 방법에 대한 자세한 내용은 "vSphere Data Protection 관리" 설명서를 참조하십시오.
- vRealize Orchestrator 인스턴스를 백업합니다. [vRealize Orchestrator 백업](#) 항목을 참조하십시오.
- vSphere Web Client를 사용하여 환경을 관리하는 vCenter Server 인스턴스에 로그인합니다. vSphere Data Protection 구성 중 사용된 관리자 권한이 있는 사용자로 로그인합니다.

절차

- 1 vSphere Web Client 홈 페이지에서 **vSphere Data Protection**을 클릭합니다.
- 2 **VDP 장치** 드롭다운 메뉴에서 vSphere Data Protection 장치를 선택하고 **연결**을 클릭합니다.
- 3 **복원** 탭을 엽니다.
- 4 백업 작업 목록에서 복원하려는 vRealize Orchestrator 백업을 선택합니다.

참고 VM이 여러 개 있는 경우 동시에 복원하여 동기화해야 합니다.

- 5 동일한 vCenter Server에서 vRealize Orchestrator 인스턴스를 복원하려면 **복원** 아이콘을 클릭하고 vRealize Orchestrator를 복원하려는 vCenter Server의 위치를 설정하라는 메시지를 따릅니다.

전원 켜기를 선택 취소하여 해당 장치가 전원이 켜지는 마지막 구성 요소가 되도록 해야 합니다. 가상 장치의 백업 및 복원 방법에 대한 자세한 내용은 "vSphere Data Protection 관리" 설명서를 참조하십시오.

복원이 성공적으로 초기화되었다는 메시지가 표시됩니다.

- 6 (선택 사항) 데이터베이스 호스트가 외부에 있고 로드 밸런서 구성을 복원하려는 경우 데이터베이스 호스트를 켭니다.

7 vRealize Orchestrator 장치의 전원을 켭니다.

결과

복원된 vRealize Orchestrator VM이 vCenter Server 인벤토리에 표시됩니다.

다음에 수행할 작업

제어 센터의 **구성 검증** 페이지를 열어 vRealize Orchestrator가 올바르게 구성되었는지 확인합니다.

Site Recovery Manager를 사용한 Orchestrator의 재해 복구

vRealize Orchestrator를 보호하려면 Site Recovery Manager를 구성해야 합니다. Site Recovery Manager에 대한 일반 구성 작업을 완료하여 해당 보호를 구현합니다.

환경 준비

Site Recovery Manager 구성을 시작하기 전에 다음 사전 요구 사항을 충족하는지 확인하십시오.

- 보호된 복구 사이트에 vSphere 5.5가 설치되어 있는지 확인합니다.
- Site Recovery Manager 5.8을 사용하고 있는지 확인합니다.
- vRealize Orchestrator가 구성되어 있는지 확인합니다.

vSphere Replication에 대한 가상 시스템 구성

Site Recovery Manager를 사용하려면 vSphere Replication에 대한 가상 시스템 또는 어레이 기반 복제를 구성해야 합니다.

다음 단계에 따라 필요한 가상 시스템에서 vSphere Replication를 활성화합니다.

절차

- 1 vSphere Web Client에서 vSphere Replication이 활성화되어야 하는 가상 시스템을 선택하고 **작업 > 모든 vSphere Replication 작업 > 복제 구성**을 클릭합니다.
- 2 **복제 유형** 창에서 **vCenter Server로 복제**를 선택하고 **다음**을 클릭합니다.
- 3 **대상 사이트** 창에서 복구 사이트용 vCenter를 선택하고 **다음**을 클릭합니다.
- 4 **복제 서버** 창에서 vSphere Replication서버를 선택하고 **다음**을 클릭합니다.
- 5 **대상 위치** 창에서 **편집**을 클릭하고 복제된 파일이 저장될 대상 데이터스토어를 선택한 후 **다음**을 클릭합니다.
- 6 **복제 옵션** 창의 기본 설정을 유지하고 **다음**을 클릭합니다.
- 7 **복구 설정** 창에서 **복구 지점 목표(RPO)** 및 **특정 시점 인스턴스**에 대한 시간을 입력하고 **다음**을 클릭합니다.
- 8 **완료 준비** 창에서 설정 내용을 확인하고 **완료**를 클릭합니다.
- 9 이 단계를 vSphere Replication이 활성화되어야 하는 모든 가상 시스템에서 반복합니다.

보호 그룹 만들기

보호 그룹을 만들어 Site Recovery Manager를 활성화하여 가상 시스템을 보호할 수 있습니다.

보호 그룹을 만드는 경우 작업이 예상대로 완료될 때까지 기다립니다. Site Recovery Manager가 보호 그룹을 만들고 그룹에서 가상 시스템의 보호가 성공적으로 수행되는지 확인합니다.

사전 요구 사항

다음 작업 중 하나를 수행했는지 확인하십시오.

- 어레이 기반 복제가 구성된 가상 시스템을 데이터스토어에 포함
- 가상 시스템에 vSphere Replication 구성
- 위 사항 전부 또는 일부를 조합하여 수행

절차

- 1 vSphere Web Client에서 **사이트 복구 > 보호 그룹**을 선택합니다.
- 2 **개체** 탭에서 아이콘을 클릭하여 보호 그룹을 만듭니다.
- 3 보호 그룹 유형 페이지에서 보호된 사이트를 선택하고 복제 유형을 선택한 후 **다음**을 클릭합니다.

옵션	작업
어레이 기반 복제 그룹	어레이 기반 복제(ABR)를 선택하고 어레이 쌍을 선택합니다.
vSphere Replication 보호 그룹	vSphere Replication을 선택합니다.

- 4 데이터스토어 그룹 또는 가상 시스템을 선택하여 보호 그룹에 추가합니다.

옵션	작업
어레이 기반 복제 보호 그룹	데이터스토어 그룹을 선택하고 다음 을 클릭합니다.
vSphere Replication 보호 그룹	목록에서 가상 시스템을 선택하고 다음 을 클릭합니다.

vSphere Replication 보호 그룹을 생성하면 vSphere Replication에 대해 구성되거나 보호 그룹에 아직 없는 가상 시스템만이 목록에 표시됩니다.

- 5 설정을 검토하고 **마침**을 클릭합니다.

보호 그룹 아래 **개체** 탭에서 보호 그룹의 생성 진행 상태를 모니터링할 수 있습니다.

결과

- Site Recovery Manager가 인벤토리 매핑을 보호된 가상 시스템에 올바르게 적용했다면 보호 그룹의 보호 상태는 정상입니다.
- Site Recovery Manager가 스토리지 정책과 연관된 모든 가상 시스템을 올바르게 보호했다면 보호 그룹의 보호 상태는 정상입니다.

복구 계획 만들기

복구 계획을 만들어 Site Recovery Manager의 가상 시스템 복구 방식을 설정합니다.

절차

- 1 vSphere Web Client에서 **사이트 복구 > 복구 계획**을 선택합니다.
- 2 **개체** 탭에서 아이콘을 클릭하여 복구 계획을 만듭니다.
- 3 계획의 이름 및 설명을 입력하고 폴더를 선택한 후 **다음**을 클릭합니다.
- 4 복구 사이트를 선택하고 **다음**을 클릭합니다.
- 5 메뉴에서 그룹 유형을 선택합니다.

옵션	설명
VM 보호 그룹	이 옵션을 선택하여 어레이 기반 복제 및 vSphere Replication 보호 그룹을 포함하는 복구 계획을 작성합니다.
스토리지 정책 보호 그룹	이 옵션을 선택하여 스토리지 정책 보호 그룹을 포함하는 복구 계획을 작성합니다.

기본값은 **VM 보호 그룹**입니다.

참고 확장된 스토리지를 사용하는 경우 그룹 유형에 대한 **스토리지 정책 보호 그룹**을 선택합니다.

- 6 복구할 계획에 대한 하나 이상의 보호 그룹을 선택하고 **다음**을 클릭합니다.
- 7 **네트워크 테스트** 값을 클릭하고 복구 테스트 동안 사용할 네트워크를 선택한 후 **다음**을 클릭합니다.
기본 옵션은 분리된 네트워크 자동 생성입니다.
- 8 요약 정보를 검토하고 **마침**을 클릭하여 복구 계획을 만듭니다.

폴더의 복구 계획 구성

복구 계획을 구성할 폴더를 생성할 수 있습니다.

복구 계획을 폴더에 구성하면 복구 계획이 여러 개인 경우 유용합니다. 복구 계획을 폴더에 배치하고 다른 사용자 또는 그룹에 대한 각 폴더에 다른 권한을 부여하여 복구 계획에 대한 액세스를 제한할 수 있습니다.

절차

- 1 vSphere Web Client의 홈 보기에서 **사이트 복구**를 클릭합니다.
- 2 **인벤토리 트리**를 확장하고 **복구 계획**을 클릭합니다.
- 3 **관련 개체** 탭을 선택하고 **폴더**를 클릭합니다.
- 4 **폴더 생성** 아이콘을 클릭하고 생성하려는 폴더의 이름을 입력한 후 **확인**을 클릭합니다.

- 5 새 복구 계획 또는 기존 복구 계획을 폴더에 추가합니다.

옵션	설명
새 복구 계획 생성	폴더를 마우스 오른쪽 버튼으로 클릭하고 복구 계획 생성 을 선택합니다.
기존 복구 계획 추가	인벤토리 트리에서 복구 계획을 폴더로 끌어 놓습니다.

- 6 (선택 사항) 폴더 이름을 바꾸거나 폴더를 삭제하려면 폴더를 마우스 오른쪽 버튼으로 클릭하고 **폴더 이름 변경** 또는 **폴더 삭제**를 선택합니다.

폴더는 비어 있는 경우에만 삭제할 수 있습니다.

복구 계획 편집

복구 계획을 편집하여 해당 복구 계획을 만들 때 지정한 속성을 변경할 수 있습니다. 보호된 사이트 또는 복구 사이트에서 복구 계획을 편집할 수 있습니다.

절차

- 1 vSphere Web Client에서 **사이트 복구 > 복구 계획**을 선택합니다.
- 2 복구 계획을 마우스 오른쪽 버튼으로 클릭하고 **계획 편집**을 선택합니다.
모니터 탭의 **복구 단계** 보기에서 **복구 계획 편집** 아이콘을 클릭하여 복구 계획을 편집할 수도 있습니다.
- 3 (선택 사항) **복구 계획 이름** 텍스트 상자에서 계획의 이름 또는 설명을 변경하고 **다음**을 클릭합니다.
- 4 복구 사이트 페이지에서 **다음**을 클릭합니다.
 복구 사이트는 변경할 수 없습니다.
- 5 (선택 사항) 복구 계획에 추가할 하나 이상의 보호 그룹을 선택 또는 선택 해제하거나 계획에서 보호 그룹을 제거하고 **다음**을 클릭합니다.
- 6 (선택 사항) 테스트 네트워크를 클릭하여 복구 사이트의 다른 테스트 네트워크를 선택하고 **다음**을 클릭합니다.
- 7 요약 정보를 검토하고 **마침**을 클릭하여 복구 계획에 대해 지정된 변경 내용을 수행합니다.
 최근 작업 보기에서 계획 업데이트를 모니터링할 수 있습니다.

시스템 속성 설정

9

시스템 속성을 설정하여 기본 Orchestrator 동작을 변경할 수 있습니다.

본 장은 다음 항목을 포함합니다.

- Orchestrator 클라이언트에 대한 비관리자의 액세스 사용 안 함
- 워크플로 및 작업에 대한 서버 파일 시스템 액세스 설정
- 워크플로 및 작업에 대한 운영 체제 명령의 액세스 설정
- Java 클래스에 JavaScript 액세스 설정
- 사용자 지정 시간 초과 속성 설정

Orchestrator 클라이언트에 대한 비관리자의 액세스 사용 안 함

Orchestrator 관리자 그룹의 구성원이 아닌 모든 사용자의 Orchestrator 클라이언트에 대한 액세스를 거부하도록 Orchestrator 서버를 구성할 수 있습니다.

기본적으로 실행 권한을 부여 받은 모든 사용자는 Orchestrator 클라이언트에 연결할 수 있습니다. 그러나 Orchestrator 구성 시스템 속성을 설정하여 Orchestrator 클라이언트에 대한 액세스를 Orchestrator 관리자로 제한할 수 있습니다.

중요 해당 설정이 구성되지 않았거나 잘못 설정된 경우 Orchestrator는 Orchestrator 클라이언트에 대한 모든 사용자의 액세스를 허용합니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **시스템 속성**을 클릭합니다.
- 3 **추가** 아이콘(+)을 클릭합니다.
- 4 키 텍스트 상자에 **com.vmware.o11n.smart-client-disabled**를 입력합니다.
- 5 **값** 텍스트 상자에 **true**를 입력합니다.
- 6 (선택 사항) **설명** 텍스트 상자에 **Orchestrator 클라이언트 연결 사용 안 함**을 입력합니다.

- 7 추가를 클릭합니다.
- 8 팝업 메뉴에서 **변경 내용 저장**을 클릭합니다.
성공적으로 저장했다는 메시지가 표시됩니다.
- 9 Orchestrator 서버를 다시 시작합니다.

결과

Orchestrator 관리자 그룹의 구성원이 아닌 모든 사용자의 Orchestrator 클라이언트에 대한 액세스를 비활성화했습니다.

워크플로 및 작업에 대한 서버 파일 시스템 액세스 설정

Orchestrator에서 워크플로 및 작업은 특정 파일 시스템 디렉토리에 대해 액세스가 제한되어 있습니다. `js-io-rights.conf` Orchestrator 구성 파일을 수정하여 서버 파일 시스템의 다른 부분으로 액세스를 확장할 수 있습니다.

Orchestrator 시스템에 대한 쓰기 액세스 권한을 허용하는 `js-io-rights.conf` 파일의 규칙

`js-io-rights.conf` 파일은 서버 파일 시스템의 정의된 디렉토리에 대한 쓰기 액세스 권한을 허용하는 규칙을 포함합니다.

중요 `js-io-rights.conf` 파일을 수정하기 전에 vRealize Orchestrator 제어 센터 서비스를 중지해야 합니다. 그렇지 않으면 `js-io-rights.conf` 파일이 기본 구성으로 되돌려집니다. [워크플로 및 작업에 대한 서버 파일 시스템 액세스 설정](#)의 내용을 참조하십시오.

`js-io-rights.conf` 파일의 필수 콘텐츠

`js-io-rights.conf` 파일의 각 줄은 다음 정보를 포함해야 합니다.

- 권한이 허용되는지 거부되는지를 나타내는 더하기(+) 또는 빼기(-) 기호
- 읽기(r), 쓰기(w) 및 실행(x) 권한 수준
- 권한을 적용할 경로

`js-io-rights.conf` 파일의 기본 콘텐츠

Orchestrator Appliance에 있는 `js-io-rights.conf` 구성 파일의 기본 콘텐츠는 다음과 같습니다.

```
-rwx /
+rxw /var/run/vco
-rwx /etc/vco/app-server/security/
+rx /etc/vco
+rx /var/log/vco/
```

기본 `js-io-rights.conf` 구성 파일에서 처음 두 줄은 다음 액세스 권한을 허용합니다.

```
-rwx /
```

파일 시스템에 대한 모든 액세스가 거부됩니다.

```
+rwx /var/run/vco
```

`/var/run/vco` 디렉토리에서 읽기, 쓰기 및 실행 액세스 권한이 허용됩니다.

js-io-rights.conf 파일의 규칙

Orchestrator는 `js-io-rights.conf` 파일에 표시되는 순서대로 액세스 권한을 확인합니다. 각 줄은 이전 줄을 재정의할 수 있습니다.

중요 `js-io-rights.conf` 파일에서 `+rwx /`를 설정하여 파일 시스템의 모든 부분에 대한 액세스 권한을 허용할 수 있습니다. 단, 그렇게 하면 보안 위험이 높습니다.

워크플로 및 작업에 대한 서버 파일 시스템 액세스 설정

워크플로 및 vRealize Orchestrator API가 액세스할 수 있는 서버 파일 시스템의 부분을 변경하려면 `js-io-rights.conf` 구성 파일을 수정합니다. `js-io-rights.conf` 파일은 워크플로가 vRealize Orchestrator 서버 파일 시스템에 액세스하려고 시도하면 생성됩니다.

절차

- 1 루트로 vRealize Orchestrator Appliance Linux 콘솔에 로그인합니다.
- 2 vRealize Orchestrator 제어 센터 서비스를 중지합니다.

```
service vco-configurator stop
```

- 3 `/etc/vco/app-server`로 이동합니다.
- 4 텍스트 편집기에서 `js-io-rights.conf` 구성 파일을 엽니다.
- 5 `js-io-rights.conf` 파일에 필요한 줄을 추가합니다.

예를 들어 다음 줄은 `/path_to_folder/noexec` 디렉토리에서 실행 권한을 거부합니다.

```
-x /path_to_folder/noexec
```

`/path_to_folder/noexec`는 실행 권한을 유지하지만 `/path_to_folder/noexec/bar`는 실행 권한을 유지하지 않습니다. 두 디렉토리 모두 읽기 및 쓰기가 가능합니다.

- 6 변경 내용을 적용하려면 다음 명령을 실행합니다.

```
/var/lib/vco/tools/configuration-cli/bin/vro-configure.sh sync-local
```

- 7 vRealize Orchestrator 제어 센터 서비스를 시작합니다.

```
service vco-configurator start
```

결과

워크플로 및 vRealize Orchestrator API에 대해 파일 시스템의 액세스 권한을 수정했습니다.

워크플로 및 작업에 대한 운영 체제 명령의 액세스 설정

Orchestrator API는 Orchestrator 서버 호스트 운영 체제에서 명령을 실행하는 스크립팅 클래스인 **Command**를 제공합니다. Orchestrator 서버 호스트에 대한 무단 액세스를 방지하기 위해 Orchestrator 애플리케이션에는 기본적으로 **Command** 클래스를 실행할 권한이 없습니다. Orchestrator 애플리케이션이 호스트 운영 체제에서 명령을 실행하는 데 권한이 필요한 경우 **Command** 스크립팅 클래스를 활성화할 수 있습니다.

Orchestrator 구성 시스템 속성을 설정하여 **Command** 클래스에 대한 사용 권한을 부여할 수 있습니다.

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **시스템 속성**을 클릭합니다.
- 3 **추가** 아이콘(+)을 클릭합니다.
- 4 키 텍스트 상자에 **com.vmware.js.allow-local-process**를 입력합니다.
- 5 값 텍스트 상자에 **true**를 입력합니다.
- 6 **설명** 텍스트 상자에 시스템 속성에 대한 설명을 입력합니다.
- 7 **추가**를 클릭합니다.
- 8 팝업 메뉴에서 **변경 내용 저장**을 클릭합니다.
성공적으로 저장했다는 메시지가 표시됩니다.
- 9 Orchestrator 서버를 다시 시작합니다.

결과

Orchestrator 서버 호스트 운영 시스템에서 로컬 명령을 실행할 수 있는 권한이 Orchestrator 애플리케이션에 부여되었습니다.

참고 `com.vmware.js.allow-local-process` 시스템 속성을 `true`로 설정하면 **Command** 스크립팅 클래스가 파일 시스템의 모든 위치에서 쓰기가 허용됩니다. 이 속성은 `js-io-rights.conf` 파일에서 **Command** 스크립팅 클래스 전용으로 설정한 모든 파일 시스템 액세스 권한을 재정의합니다. `js-io-rights.conf` 파일에서 설정한 파일 시스템 액세스 권한은 여전히 **Command** 외 모든 스크립팅 클래스에 적용됩니다.

Java 클래스에 JavaScript 액세스 설정

기본적으로 Orchestrator에서는 제한된 Java 클래스 집합으로 JavaScript 액세스가 제한됩니다. 보다 광범위한 Java 클래스에 대한 JavaScript 액세스가 필요한 경우 이 액세스를 허용하도록 Orchestrator 시스템 속성을 설정해야 합니다.

JavaScript 엔진에 Java 가상 시스템(JVM)에 대한 전체 액세스 권한을 허용하면 잠재적 보안 문제가 나타납니다. 잘못된 형식 또는 악성 스크립트가 Orchestrator 서버를 실행하는 사용자가 액세스하는 모든 시스템 구성 요소에 액세스할 수 있습니다. 따라서 Orchestrator JavaScript 엔진은 기본적으로 `java.util.*` 패키지의 클래스에만 액세스할 수 있습니다.

JavaScript가 `java.util.*` 패키지 외부 클래스에 액세스하게 해야 한다면 JavaScript 액세스를 허용할 Java 패키지를 구성 파일에 나열하십시오. 그런 다음 이 파일을 가리키도록 `com.vmware.scripting.rhino-class-shutter-file` 시스템 속성을 설정하십시오.

절차

- 1 JavaScript 액세스를 허용할 Java 패키지 목록을 저장하려면 텍스트 구성 파일을 작성합니다.

예를 들어 JavaScript가 `java.net` 패키지의 모든 클래스와 `java.lang.Object` 클래스에 액세스하게 하려면 다음 내용을 파일에 추가합니다.

```
java.net.*
java.lang.Object
```

- 2 구성 파일에 적절한 이름을 지정해 적절한 위치에 저장합니다.
- 3 제어 센터에 **root**로 로그인합니다.
- 4 **시스템 속성**을 클릭합니다.
- 5 **추가** 아이콘(+)을 클릭합니다.
- 6 키 텍스트 상자에 `com.vmware.scripting.rhino-class-shutter-file`을 입력합니다.
- 7 값 텍스트 상자에 구성 파일의 경로를 입력합니다.
- 8 **설명** 텍스트 상자에 시스템 속성에 대한 설명을 입력합니다.
- 9 **추가**를 클릭합니다.
- 10 팝업 메뉴에서 **변경 내용 저장**을 클릭합니다.
성공적으로 저장했다는 메시지가 표시됩니다.
- 11 Orchestrator 서버를 다시 시작합니다.

결과

JavaScript 엔진이 지정한 Java 클래스에 액세스할 수 있습니다.

사용자 지정 시간 초과 속성 설정

vCenter Server가 오버로드되면 Orchestrator 서버로 응답을 반환하는 시간이 기본으로 지정된 20000 밀리초보다 오래 걸리게 됩니다. 이를 방지하려면 Orchestrator 구성 파일을 수정하여 기본 시간 초과 기간을 늘려야 합니다.

기본 시간 초과 기간이 특정 작업 완료 전에 만료되면 Orchestrator 서버 로그에 오류가 포함됩니다.

```
Operation 'getPropertyContent' total time : '5742228' for 1823 calls, mean time : '3149.0', min time : '0', max time : '32313' Timeout, unable to get property 'info' com.vmware.vmo.plugin.vi4.model.TimeoutException
```

절차

- 1 제어 센터에 **root**로 로그인합니다.
- 2 **시스템 속성**을 클릭합니다.
- 3 **추가** 아이콘()을 클릭합니다.
- 4 **키** 텍스트 상자에 **com.vmware.vmo.plugin.vi4.waitUpdatesTimeout**을 입력합니다.
- 5 **값** 텍스트 상자에 새 시간 초과 기간을 밀리초 단위로 입력합니다.
- 6 (선택 사항) **설명** 텍스트 상자에 시스템 속성에 대한 설명을 입력합니다.
- 7 **추가**를 클릭합니다.
- 8 팝업 메뉴에서 **변경 내용 저장**을 클릭합니다.
성공적으로 저장했다는 메시지가 표시됩니다.
- 9 Orchestrator 서버를 다시 시작합니다.

결과

설정된 값이 기본 시간 초과로 설정된 값인 20000밀리초를 재정의합니다.

vRealize Orchestrator를 설치 및 구성하면 Orchestrator를 사용하여 가상 환경의 관리와 연관되어 자주 반복되는 프로세스를 자동화할 수 있습니다.

- Orchestrator 클라이언트에 로그인하여 vCenter Server 인벤토리 개체 또는 Orchestrator가 플러그인을 통해 액세스하는 기타 개체에서 워크플로를 실행 및 스케줄링합니다. VMware vRealize Orchestrator 클라이언트 "사용" 을 참조하십시오.
- 표준 Orchestrator 워크플로를 복제 및 수정하고 자체적으로 작업 및 워크플로를 작성하여 vCenter Server에서 작업을 자동화합니다.
- 플러그인 및 웹 서비스를 개발하여 Orchestrator 플랫폼을 확장합니다.
- vSphere Web Client를 사용하여 vSphere 인벤토리 개체에서 워크플로를 실행합니다.

본 장은 다음 항목을 포함합니다.

- [Orchestrator Appliance 웹 콘솔에서 Orchestrator 클라이언트에 로그인](#)

Orchestrator Appliance 웹 콘솔에서 Orchestrator 클라이언트에 로그인

일반적인 관리 작업을 수행하거나 워크플로를 편집하고 생성하려면 Orchestrator 클라이언트 인터페이스에 로그인해야 합니다.

Orchestrator 클라이언트 인터페이스는 관리 권한을 가지고 있으며 워크플로, 작업 및 기타 사용자 지정 요소를 개발하고자 하는 개발자를 위해 설계되었습니다.

중요 Orchestrator Appliance 서버의 시계와 Orchestrator 클라이언트 시스템의 시계를 동기화해야 합니다.

사전 요구 사항

- Orchestrator Appliance를 다운로드 및 배포합니다.
- 애플리케이션이 가동되어 실행 중인지 확인합니다.

- Orchestrator 클라이언트를 실행할 워크스테이션에 64비트 Java를 설치합니다.

참고 32비트 Java는 지원되지 않습니다.

절차

- 1 웹 브라우저에서 Orchestrator Appliance 가상 시스템의 IP 주소로 이동합니다.

`http://orchestrator_appliance_ip`

- 2 **Orchestrator 클라이언트 시작**을 클릭합니다.
- 3 **호스트 이름** 텍스트 상자에 Orchestrator Appliance의 IP 또는 도메인 이름을 입력합니다.

Orchestrator Appliance의 IP 주소가 기본적으로 표시됩니다.

- 4 Orchestrator 클라이언트 사용자 이름과 암호를 사용하여 로그인합니다.

vRealize Automation 또는 vSphere를 인증 제공자로 사용하는지 여부에 따라 해당하는 자격 증명을 입력하여 Orchestrator 클라이언트에 로그인합니다.

Orchestrator 환경에서 다중 테넌트를 사용하는 경우 해당하는 시스템 관리자 또는 테넌트 관리자 사용자 이름, 암호 및 테넌트 ID를 입력합니다.

- 5 **보안 경고** 창에서 인증서 경고를 처리할 옵션을 선택합니다.

Orchestrator 클라이언트는 SSL 인증서를 사용하여 Orchestrator 서버와 통신합니다. 신뢰할 수 있는 CA는 설치하는 동안에는 인증서에 서명하지 않습니다. Orchestrator 서버에 연결할 때마다 인증서 경고가 수신됩니다.

옵션	설명
무시	현재 SSL 인증서를 계속 사용합니다. 같은 Orchestrator 서버에 다시 연결하거나, 원격 Orchestrator 서버와 워크플로를 동기화하려고 시도하면 경고 메시지가 다시 나타납니다.
취소	창을 닫고 로그인 프로세스를 중지합니다.
이 인증서를 설치하고 이와 관련된 보안 경고를 더 이상 표시 안 함	인증서를 설치하고 보안 경고를 더 이상 표시하지 않으려면 이 확인란을 선택하고 무시 를 클릭합니다.

기본 SSL 인증서를 CA가 서명한 인증서로 변경할 수 있습니다. SSL 인증서를 변경하는 방법에 대한 자세한 내용은 "VMware vRealize Orchestrator 설치 및 구성" 을 참조하십시오.

다음에 수행할 작업

패키지를 가져오거나, 워크플로를 시작하거나, 시스템에 대한 루트 액세스 권한을 설정할 수 있습니다.