



## Red Hat Satellite 6.15

### Ansible 통합을 사용하여 구성 관리

Satellite에서 Ansible 통합을 구성하고 Ansible 역할 및 플레이북을 사용하여 호스트를 구성합니다.



## Red Hat Satellite 6.15 Ansible 통합을 사용하여 구성 관리

---

Satellite에서 Ansible 통합을 구성하고 Ansible 역할 및 플레이북을 사용하여 호스트를 구성합니다.

Red Hat Satellite Documentation Team  
satellite-doc-list@redhat.com

## 법적 공지

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 초록

이 가이드에서는 Red Hat Satellite를 Ansible과 통합하는 방법과 Satellite에서 Ansible 역할 및 플레이 북을 사용하여 원격 실행을 수행하고 반복적인 작업을 자동화하는 방법을 설명합니다.

## 차례

RED HAT 문서에 관한 피드백 제공 .....	4
<b>1장. SATELLITE에서 ANSIBLE 시작하기 .....</b>	<b>5</b>
1.1. 지원되는 ANSIBLE 버전	5
1.2. ANSIBLE 역할을 실행하도록 SATELLITE 구성	5
1.3. SATELLITE와 ANSIBLE 통합 활성화	5
1.4. ANSIBLE 역할 및 변수 가져오기	6
1.5. SATELLITE에서 ANSIBLE 변수 덮어쓰기	6
1.6. RED HAT ENTERPRISE LINUX 시스템 역할 추가	8
1.7. ANSIBLE 컬렉션 동기화	9
1.8. ANSIBLE 구성 사용자 정의	10
1.9. SATELLITE에서 ANSIBLE VAULT 사용	11
<b>2장. ANSIBLE 역할을 사용하여 클라이언트에서 반복적인 작업 자동화 .....</b>	<b>12</b>
2.1. 기존 호스트에 ANSIBLE 역할 할당	12
2.2. 호스트에서 ANSIBLE 역할 제거	12
2.3. ANSIBLE 역할의 순서 변경	12
2.4. 호스트에서 ANSIBLE 역할 실행	13
2.5. 호스트 그룹에 ANSIBLE 역할 할당	13
2.6. 호스트 그룹에서 ANSIBLE 역할 실행	14
2.7. 검사 모드에서 ANSIBLE 역할 실행	14
<b>3장. SATELLITE에서 ANSIBLE 플레이북 실행 .....</b>	<b>15</b>
<b>4장. 원격 작업 구성 및 설정 .....</b>	<b>16</b>
4.1. RED HAT SATELLITE에서 원격 실행	16
4.2. 원격 실행 워크플로	16
4.3. 원격 실행에 대한 권한	17
4.4. 원격 실행을 위한 전송 모드	18
4.5. 풀 클라이언트를 사용하도록 호스트 구성	19
4.6. 작업 템플릿 생성	19
4.7. 이름으로 ANSIBLE 플레이북 가져오기	21
4.8. 사용 가능한 모든 ANSIBLE 플레이북 가져오기	21
4.9. SATELLITE에서 CAPSULE 원격 실행 설정에 대한 폴백 구성	22
4.10. SATELLITE에서 글로벌 CAPSULE 원격 실행 설정 구성	23
4.11. 대체 디렉토리를 사용하여 호스트에서 원격 작업을 실행하도록 SATELLITE 구성	23
4.12. 권한 상승 방법 변경	24
4.13. 원격 실행을 위한 SSH 키 배포	24
4.14. 원격 실행을 위해 SSH 키 배포	25
4.15. 원격 실행에 사용되는 SSH 키에 암호 추가	25
4.16. SATELLITE API를 사용하여 원격 실행을 위한 SSH 키 가져오기	25
4.17. 프로비저닝 중에 SSH 키를 배포하도록 KICKSTART 템플릿 구성	26
4.18. 티켓을 부여하는 KERBEROS 티켓의 키맵 구성	26
4.19. 원격 실행을 위한 KERBEROS 인증 구성	27
4.20. 작업 템플릿 설정	28
4.21. 원격 작업 실행	29
4.22. 작업 마법사의 고급 설정	31
4.23. 확장 CRON 라인 사용	33
4.24. 호스트에 대한 반복적인 ANSIBLE 작업 예약	34
4.25. 호스트 그룹에 대한 반복적인 ANSIBLE 작업 예약	34
4.26. 작업 모니터링	35
4.27. 패키지 및 에라타 작업에 ANSIBLE 공급자 사용	36

4.28. CAPSULE에 작업 속도 제한 설정	37
<b>5장. RED HAT SATELLITE 및 ANSIBLE AUTOMATION CONTROLLER 통합</b> .....	<b>38</b>
5.1. 동적 인벤토리 항목으로 ANSIBLE AUTOMATION CONTROLLER에 SATELLITE SERVER 추가	38
5.2. 호스트의 프로비저닝 콜백 구성	39
<b>6장. 작업 템플릿 예 및 확장</b> .....	<b>43</b>
6.1. 작업 템플릿 사용자 정의	43
6.2. 기본 작업 템플릿 카테고리	43
6.3. RESTORECON 템플릿 예	44
6.4. RESTORECON 템플릿 렌더링	45
6.5. 여러 호스트에서 RESTORECON 템플릿 실행	45
6.6. 템플릿에 전원 작업 포함	47



## RED HAT 문서에 관한 피드백 제공

문서 개선을 위한 의견에 감사드립니다. 어떻게 개선할 수 있는지 알려주십시오.

Red Hat Jira에서 **문제 생성** 양식을 사용하여 피드백을 제공합니다. Jira 문제는 Red Hat Satellite Jira 프로젝트에서 생성되어 진행 상황을 추적할 수 있습니다.

### 사전 요구 사항

- [Red Hat 계정](#)을 등록했는지 확인합니다.

### 프로세스

1. 다음 링크를 클릭합니다. **문제 만들기**. Jira가 로그인 오류를 표시하는 경우 양식으로 리디렉션된 후 로그인하고 계속 진행합니다.
2. **요약** 및 **설명** 필드를 작성합니다. **설명** 필드에 문서 URL, 장 또는 섹션 번호, 문제에 대한 자세한 설명을 포함합니다. 양식의 다른 필드를 수정하지 마십시오.
3. **생성**을 클릭합니다.



# 1장. SATELLITE에서 ANSIBLE 시작하기

이 가이드를 사용하여 원격 실행 시 Ansible을 사용하도록 Satellite를 구성합니다.

## 1.1. 지원되는 ANSIBLE 버전

Satellite는 Satellite Server의 기본 운영 체제에서 제공하는 Ansible 또는 원격 실행을 위해 모든 Capsule을 사용합니다. 따라서 지원되는 Ansible 버전은 기본 OS 구성에 따라 다릅니다.

## 1.2. ANSIBLE 역할을 실행하도록 SATELLITE 구성

Satellite에서는 일상적인 작업 자동화에 도움이 되도록 Ansible 역할을 가져올 수 있습니다. Ansible은 Satellite 서버에서 기본적으로 활성화되어 있습니다.

### Ansible 경로

Satellite는 다음 경로에서 Ansible 역할을 가져오고 실행합니다.

- `/etc/ansible/roles`
- `/usr/share/ansible/roles`
- `/etc/ansible/collections`
- `/usr/share/ansible/collections`

설치된 패키지의 역할 및 컬렉션은 `/usr/share/ansible`에 배치됩니다. 사용자 지정 역할 또는 컬렉션을 추가하려면 `/etc/ansible`에 배치합니다.

Red Hat은 Red Hat에서 얻은 Ansible 역할 및 컬렉션에 대해서만 지원합니다.

경로는 Satellite에 의해 구성됩니다. 자세한 내용은 [1.8절. "Ansible 구성 사용자 정의"](#)의 내용을 참조하십시오.

### 프로세스

1. Satellite Server의 [Ansible 경로](#) 및 역할을 사용하려는 위치에서 모든 Capsule Server의 디렉터리에 역할을 추가합니다. 사용자 지정 또는 타사 Ansible 역할을 사용하려면 Satellite Server와 Capsule Server 간에 역할을 동기화하도록 외부 버전 제어 시스템을 구성해야 합니다.
2. 호스트에서 Ansible 역할을 실행하는 데 사용할 모든 Capsule Server에서 Ansible 플러그인을 활성화합니다.

```
# satellite-installer --enable-foreman-proxy-plugin-ansible
```

3. SSH 키를 배포하여 SSH를 사용하여 Capsule이 호스트에 연결할 수 있도록 합니다. 자세한 내용은 [호스트 관리에서 원격 실행을 위한 SSH 키](#) 배포를 참조하십시오. Satellite는 원격 실행 작업을 실행하는 방식과 동일하게 Ansible 역할을 실행합니다.
4. Ansible 역할을 Satellite로 가져옵니다.
5. [2장. Ansible 역할을 사용하여 클라이언트에서 반복적인 작업 자동화](#) 로 이동합니다.

## 1.3. SATELLITE와 ANSIBLE 통합 활성화

다음 절차를 수행하여 Satellite 서버에서 Ansible 플러그인을 활성화합니다.

#### 프로세스

- Satellite 서버에서 Ansible 플러그인을 활성화합니다.

```
# satellite-installer \
--enable-foreman-plugin-ansible \
--enable-foreman-proxy-plugin-ansible
```

## 1.4. ANSIBLE 역할 및 변수 가져오기

Ansible 역할 및 변수를 Satellite Server의 [Ansible 경로](#) 또는 Ansible이 활성화된 Capsule Server에서 가져올 수 있습니다.

#### 사전 요구 사항

- 가져오는 역할 및 변수가 역할을 사용하려는 위치에서 모든 Capsule의 [Ansible 경로](#)에 있는지 확인합니다.

#### 프로세스

1. Satellite 웹 UI에서 **Configure > Ansible > Roles** 로 이동합니다.
2. **Import** 를 클릭하여 가져올 Capsule을 선택합니다.
3. 가져올 역할을 선택합니다.
4. **Submit** 을 클릭합니다.

## 1.5. SATELLITE에서 ANSIBLE 변수 덮어쓰기

Satellite에서 Ansible 역할을 실행하는 경우 Satellite를 사용하여 해당 역할의 Ansible 변수를 재정의할 수 있습니다.

다음 절차에서는 호스트 및 호스트 그룹을 나타냅니다. 자세한 내용은 [호스트 관리](#)를 참조하십시오.

#### 변수 덮어쓰기의 우선순위

Ansible 역할을 사용하여 영향을 받는 **사용자**가 아닌 사용자로 작업을 실행하는 경우 Ansible 변수를 재정의하기 위한 엄격한 우선 순위 순서가 있습니다. 변수가 우선순위의 올바른 순서를 따르도록 하려면 변수 우선 순위: **변수를 배치해야 하는 위치**를 참조하십시오.

#### 사전 요구 사항

- Satellite에 Ansible 변수가 있어야 합니다. 자세한 내용은 다음을 참조하십시오. [1.4절. "Ansible 역할 및 변수 가져오기"](#)
- 재정의된 Ansible 변수를 사용하려면 사용자에게 호스트와 일치하는 특성을 볼 수 있는 역할이 있어야 합니다.

#### 프로세스

1. Satellite 웹 UI에서 **Configure > Ansible > Variables** 로 이동합니다.

2. Satellite를 사용하여 재정의하고 관리할 Ansible 변수를 선택합니다.
3. **Default Behavior** 영역에서 **Override** 확인란을 선택합니다.
4. **Parameter Type** 필드에서 **string** 또는 **boolean** 과 같은 유효성 검사를 위한 값 유형을 선택합니다. 유형 **배열** 및 **해시** 에는 변수 일치 시 처리하기 위한 추가 옵션이 있습니다. 자세한 내용은 아래 **의 속성 순서 우선 순위 영역**을 참조하십시오.
5. **Default Value** 필드에 변수에 일치하는 항목이 없는 경우 사용할 기본값을 입력합니다.
6. 선택 사항: Satellite 웹 UI에서 변수 값을 일반 텍스트로 표시하지 않으려면 **Hidden Value** 확인란을 선택하여 변수의 내용을 별표로 표시합니다. 이는 암호 또는 시크릿 토큰과 같은 민감한 값에 유용합니다.
7. 선택 사항: **선택적 입력 유효성 검사기** 영역을 확장하고 변수의 관련 값을 확인하는 데 사용할 조건을 지정합니다.
  - 이 변수를 채우도록 사용자를 강제 적용하려면 **Required** 를 선택합니다.
  - **Validator Type** 필드에서 값의 유효성을 검사하는 방법을 선택합니다.
    - **list** - 허용되는 값의 열거에 대해 값의 유효성을 검사합니다. List - The value will be validated against an enumeration of allowed values.
    - **regex** - 정규식 패턴에 대해 값의 유효성을 검사합니다.
8. 선택 사항: **속성 순서** 지정 영역에서 호스트 속성별 재정의와 일치시킬 우선순위 순서를 지정합니다. 맨 위에 있는 순서는 우선 순위가 높습니다. 첫 번째 일치 항목이 우선합니다. 여러 특성을 쉼표를 AND 작업으로 사용하여 단일 일치 키에 결합할 수 있습니다. 예를 들어, hostgroup의 matcher 키인 **환경은 hostgroup = "web servers" AND environment = production** 과 같은 일치자를 예상합니다.
 

매개변수 유형 **배열** 또는 **해시** 를 사용하는 경우 다음과 같이 추가로 설정할 수 있습니다.

  - **병합 덮어쓰기** - 전체 배열 또는 해시를 대체하는 대신 배열/hashes의 멤버입니다. 해시에 동일한 키가 포함된 경우 호스트 값으로 값을 덮어씁니다.
  - **merge Default** - 배열 또는 해시에 기본값을 추가합니다.
  - **중복 방지** - 배열 또는 해시의 값이 고유하도록 합니다. **Avoid Duplicates** - Ensures that the values in the array or hash are unique.
9. 선택 사항: **Matchers** 지정 영역을 확장하고 변수가 덮어쓸 호스트를 선택하기 위한 기준을 지정합니다.
10. 재정의 설정을 저장하려면 **Submit** 을 클릭합니다.

Ansible 변수를 사용하려면 변수를 호스트 또는 호스트 그룹에 매개 변수로 추가하거나 변수를 글로벌 매개 변수로 추가합니다.

### 호스트에 변수 추가

1. Satellite 웹 UI에서 **호스트 > 모든 호스트**로 이동하여 사용하려는 호스트를 선택합니다.
2. **Ansible** 탭을 클릭하고 **변수** 영역에서 연필 아이콘을 클릭하여 변수 값을 편집합니다.
3. 눈금 아이콘을 클릭하여 변경된 변수 또는 교차 아이콘의 값을 수락하여 변경 사항을 취소합니다.

## 호스트 그룹에 변수 추가

1. Satellite 웹 UI에서 **Configure > Host Groups** 로 이동하여 사용할 호스트 그룹을 선택합니다.
2. **매개 변수** 탭을 클릭하고 **호스트 그룹 매개 변수 영역**에서 **매개 변수 추가**를 클릭합니다.
3. **이름** 필드에 Ansible 변수 이름을 추가합니다.
4. **유형** 목록에서 검증을 위해 변수 유형을 선택합니다.
5. **값** 필드에 변수 값을 입력합니다.

## 전역 매개변수로 변수 추가

1. Satellite 웹 UI에서 **Configure > Global Parameters** 로 이동하여 **Create Parameter** 를 클릭합니다.
2. **이름** 필드에 Ansible 변수 이름을 추가합니다.
3. **유형** 목록에서 검증을 위해 변수 유형을 선택합니다.
4. **값** 필드에 변수 값을 입력합니다.
5. **선택 사항:** Ansible 변수를 일반 텍스트로 표시하지 않으려면 **Hidden Values** 확인란을 선택하여 변수의 내용을 Satellite 웹 UI에 별표로 표시합니다.

## 1.6. RED HAT ENTERPRISE LINUX 시스템 역할 추가

Red Hat Enterprise Linux 시스템 역할은 Red Hat Enterprise Linux 서버를 원격으로 관리하는 구성 인터페이스입니다. Red Hat Enterprise Linux 시스템 역할을 사용하여 Satellite에 Ansible 역할을 추가할 수 있습니다. Satellite에서 Ansible 역할을 사용하면 구성을 더 빠르고 쉽게 만들 수 있습니다.

일부 Red Hat Enterprise Linux 시스템 역할에 대한 지원 수준은 기술 프리뷰일 수 있습니다. 지원 수준 및 Red Hat Enterprise Linux 시스템 역할에 대한 일반적인 정보는 [Red Hat Enterprise Linux 시스템 역할](#)을 참조하십시오.

Extras 채널에 가입하기 전에 [Red Hat Enterprise Linux Extras 제품 라이프 사이클 문서](#)를 참조하십시오.

### 프로세스

1. 다음 리포지토리가 활성화되었는지 확인합니다.
  - Red Hat Enterprise Linux 8에서 **Appstream** 리포지토리가 활성화되어 있는지 확인합니다.

```
# subscription-manager repos --enable=rhel-8-for-x86_64-appstream-rpms
```

아키텍처에 지정된 Appstream 리포지토리를 활성화해야 합니다. 자세한 내용은 [RHEL 8 리포지토리](#)를 참조하십시오.

- Red Hat Enterprise Linux 7에서 **Extras** 리포지토리가 활성화되어 있는지 확인합니다.

```
# subscription-manager repos --enable=rhel-7-server-extras-rpms
```

2. **rhel-system-roles** 패키지를 설치합니다.

```
# satellite-maintain packages install rhel-system-roles
```

**rhel-system-roles** 패키지가 `/usr/share/ansible/roles/` 에 다운로드됩니다. 가져오기 전에 파일을 보고 수정할 수 있습니다.

3. Satellite 웹 UI에서 **Configure > Ansible > Roles** 로 이동합니다.
4. 가져올 역할이 포함된 Capsule을 클릭합니다.
5. Ansible 역할 목록에서 가져올 역할의 확인란을 선택한 다음 **업데이트**를 클릭합니다.

이제 호스트 또는 호스트 그룹에 Ansible 역할을 할당할 수 있습니다. 자세한 내용은 2.1절. "[기존 호스트에 Ansible 역할 할당](#)"의 내용을 참조하십시오.

Ansible 작업 템플릿에 추가하여 이러한 역할에 포함된 모듈을 Ansible 플레이북에 추가할 수도 있습니다. 작업 템플릿에 **hosts:all** 행을 포함해야 합니다. 자세한 내용은 [RHEL \(Red Hat Enterprise Linux\) 시스템 역할](#) 을 참조하십시오.

## 1.7. ANSIBLE 컬렉션 동기화

Satellite에서는 Private Automation Hub, [console.redhat.com](https://console.redhat.com) 및 기타 Satellite 인스턴스에서 Ansible 컬렉션을 동기화할 수 있습니다. Ansible 컬렉션은 동기화 후 **Content** 의 Satellite 웹 UI 메뉴에 Satellite를 새 리포지토리 유형으로 표시합니다.

### 절차

1. Satellite 웹 UI에서 **콘텐츠 > 제품**으로 이동합니다.
2. 필요한 제품 이름을 선택합니다.
3. **제품** 창에서 리포지토리를 생성할 제품의 이름을 선택합니다.
4. **Repositories** 탭을 클릭한 다음 **New Repository** 를 클릭합니다.
5. **이름** 필드에 리포지토리의 이름을 입력합니다.  
**레이블** 필드는 이름에 따라 자동으로 채워집니다.
6. **유형** 목록에서 **ansible 컬렉션**을 선택합니다.
7. **Upstream URL** 필드에 업스트림 컬렉션 리포지토리의 URL을 입력합니다.  
URL은 모든 Ansible Galaxy 엔드포인트가 될 수 있습니다. 예:  
**https://console.redhat.com/api/automation-hub/**
8. 선택 사항: **Requirements.yml** 필드에서 끝점에서 동기화할 컬렉션 목록과 해당 버전을 지정할 수 있습니다.  
컬렉션 목록을 지정하지 않으면 끝점의 모든 항목이 동기화됩니다.

```
---
collections:
- name: my_namespace.my_collection
  version: 1.2.3
```

자세한 내용은 [Galaxy 사용자 가이드](#) 의 [동일한 requirements.yml 파일에서 역할 및 컬렉션 설치](#) 를 참조하십시오.

9. 인증.

- a. **Private Automation Hub**에서 Satellite를 동기화하려면 **Auth Token** 필드에 토큰을 입력합니다.  
자세한 내용은 [Hub에 연결에서 프라이빗 자동화 허브 연결을 참조하십시오](#).
- b. **console.redhat.com**에서 Satellite를 동기화하려면 **Auth Token** 필드에 토큰을 입력하고 **Auth URL** 필드에 SSO URL을 입력합니다.  
자세한 내용은 [자동화 허브 시작하기를 참조하십시오](#).
- c. Satellite에서 Satellite를 동기화하려면 두 인증 필드를 모두 비워 둡니다.

10. **저장**을 클릭합니다.

- 11. Ansible Collections 리포지토리로 이동합니다.
- 12. **Select Action** 메뉴에서 **Sync now**를 선택합니다.

## 1.8. ANSIBLE 구성 사용자 정의

Satellite는 Ansible과 Satellite를 통합하는 데 필요한 필수 Ansible 구성을 관리합니다. 그러나 다른 Ansible 구성 옵션을 일반적으로 사용자 지정할 수 있습니다.

Satellite는 필수 Ansible 구성을 **/etc/foreman-proxy/ansible.env**에 환경 변수로 저장합니다. 이 파일은 **satellite-installer**에 의해 관리됩니다.

Ansible은 구성 파일 및 Capsule에서 제공하는 환경에서 구성을 읽습니다. Ansible 구성을 사용자 지정해야 하는 경우 시스템 전체 **/etc/ansible/ansible.cfg** 파일 또는 **foreman-proxy** 사용자의 홈 디렉터리에 있는 **/usr/share/foreman-proxy/.ansible.cfg** 파일에서 이를 수행할 수 있습니다. **/usr/share/foreman-proxy/.ansible.cfg**를 사용하는 경우 Satellite에서 생성한 Ansible은 **/etc/ansible/ansible.cfg**의 구성을 무시합니다.

환경 변수가 **ansible.cfg**의 값보다 우선하므로 Satellite에 필요한 필수 구성이 유지됩니다.

다음 표에는 Satellite에서 관리하는 필수 Ansible 구성 옵션이 나열되어 있습니다.

표 1.1. 필수 Ansible 구성

환경 변수	구성 키	설명
ANSIBLE_CALLBACKS_ENABLED	callbacks_enabled	버전 간 호환성을 위해 <b>callback_whitelist</b> 에 해당하는 콜백을 활성화합니다.
ANSIBLE_CALLBACK_WHITELIST	callback_whitelist	콜백을 Satellite로 활성화합니다. <b>callbacks_enabled</b> for cross-version compatibility와 동일합니다.
ANSIBLE_COLLECTIONS_PATHS	collections_paths	Ansible 컬렉션 경로 목록
ANSIBLE_HOST_KEY_CHECKING	host_key_checking	SSH 연결 중 호스트 키 확인 비활성화

환경 변수	구성 키	설명
ANSIBLE_LOCAL_TEMP	local_tmp	Capsule의 임시 디렉터리
ANSIBLE_ROLES_PATH	roles_path	Ansible 역할에 대한 경로 목록
ANSIBLE_SSH_ARGS	ssh_args	SSH 연결에 전달된 인수

## 추가 리소스

- [Ansible 구성 설정](#)

## 1.9. SATELLITE에서 ANSIBLE VAULT 사용

Ansible Vault 툴을 사용하여 중요한 Ansible 데이터 파일을 암호화하고 파일에 저장된 암호를 사용하여 암호화된 파일에 액세스하도록 Ansible을 구성할 수 있습니다.

### 프로세스

1. `/etc/ansible/ansible.cfg`를 사용자 지정한 경우 `/etc/ansible/ansible.cfg`에서 `/usr/share/foreman-proxy/.ansible.cfg`로 구성을 복사합니다.

2. `ansible-vault` 명령을 사용하여 중요한 파일을 암호화합니다.

```
# ansible-vault encrypt /etc/ansible/roles/Role_Name/vars/main.yml
```

`ansible-vault`는 파일 권한을 **600**으로 변경합니다.

3. 암호화된 파일의 그룹 및 권한을 변경하여 `foreman-proxy` 사용자가 읽을 수 있도록 합니다.

```
# chgrp foreman-proxy /etc/ansible/roles/Role_Name/vars/main.yml
# chmod 0640 /etc/ansible/roles/Role_Name/vars/main.yml
```

4. `/usr/share/foreman-proxy/.ansible_vault_password` 파일을 생성하고 Vault 암호를 입력합니다.

5. `foreman-proxy` 사용자만 읽을 수 있도록 `.ansible_vault_password` 파일의 사용자 및 권한을 변경합니다.

```
# chown foreman-proxy:foreman-proxy /usr/share/foreman-proxy/.ansible_vault_password
# chmod 0400 /usr/share/foreman-proxy/.ansible_vault_password
```

6. Vault 암호 파일의 경로를 `/usr/share/foreman-proxy/.ansible.cfg`의 `[defaults]` 섹션에 추가합니다.

```
[defaults]
vault_password_file = /usr/share/foreman-proxy/.ansible_vault_password
```

Vault 암호 파일의 경로는 절대적이어야 합니다.

## 2장. ANSIBLE 역할을 사용하여 클라이언트에서 반복적인 작업 자동화

### 2.1. 기존 호스트에 ANSIBLE 역할 할당

Satellite 클라이언트의 원격 관리를 위해 Ansible 역할을 사용할 수 있습니다.

#### 사전 요구 사항

- Ansible 역할을 구성하고 가져왔는지 확인합니다.

#### 절차

1. Satellite 웹 UI에서 **호스트 > 모든 호스트로 이동합니다.**
2. 호스트를 선택하고 **편집** 을 클릭합니다.
3. **Ansible Roles** 탭의 **Available Ansible Roles** 목록에서 추가할 역할을 선택합니다.
4. + 아이콘을 클릭하여 호스트에 역할을 추가합니다. 두 개 이상의 역할을 추가할 수 있습니다.
5. **Submit** 을 클릭합니다.

호스트에 Ansible 역할을 할당한 후 Ansible을 사용하여 원격 실행을 수행할 수 있습니다. 자세한 내용은 [4.13절. "원격 실행을 위한 SSH 키 배포"](#) 의 내용을 참조하십시오.

#### 매개변수 변수 덮어쓰기

매개 변수 탭에서 매개 변수 추가를 클릭하여 런타임에 작업 템플릿에 전달할 매개변수변수를 추가합니다. 여기에는 호스트와 연결할 모든 Ansible 플레이북 매개변수 및 호스트 매개변수가 포함됩니다. Ansible 작업 템플릿과 함께 매개변수 변수를 사용하려면 호스트 매개 변수를 추가해야 합니다.

### 2.2. 호스트에서 ANSIBLE 역할 제거

다음 절차에 따라 호스트에서 Ansible 역할을 제거합니다.

#### 절차

1. Satellite 웹 UI에서 **호스트 > 모든 호스트로 이동합니다.**
2. 호스트를 선택하고 **편집** 을 클릭합니다.
3. **Ansible Roles** 탭을 선택합니다.
4. **Assigned Ansible Roles** 영역에서 - 아이콘을 클릭하여 호스트에서 역할을 제거합니다. 더 많은 역할을 제거하려면 반복합니다.
5. **Submit** 을 클릭합니다.

### 2.3. ANSIBLE 역할의 순서 변경

다음 절차에 따라 호스트에 적용된 Ansible 역할의 순서를 변경합니다.



### 절차

1. Satellite 웹 UI에서 **호스트 > 모든 호스트로 이동합니다.**
2. 호스트를 선택합니다.
3. **Ansible Roles** 탭을 선택합니다.
4. **Assigned Ansible Roles** 영역에서 역할을 기본 위치로 끌어서 놓음으로써 역할의 순서를 변경할 수 있습니다.
5. **Submit** 을 클릭하여 Ansible 역할의 순서를 저장합니다.

## 2.4. 호스트에서 ANSIBLE 역할 실행

Satellite 웹 UI를 통해 호스트에서 Ansible 역할을 실행할 수 있습니다.

### 사전 요구 사항

- Ansible 역할을 실행하도록 배포를 구성해야 합니다. 자세한 내용은 1.2절. [“Ansible 역할을 실행하도록 Satellite 구성”](#)의 내용을 참조하십시오.
- 호스트에 Ansible 역할을 할당해야 합니다.

### 절차

1. Satellite 웹 UI에서 **호스트 > 모든 호스트로 이동합니다.**
2. 실행할 Ansible 역할이 포함된 호스트의 확인란을 선택합니다.
3. **Select Action** 목록에서 **Run all Ansible roles** 를 선택합니다.

Ansible 작업의 상태는 **Run Ansible roles** 페이지에서 볼 수 있습니다. 작업을 다시 실행하려면 다시 실행을 클릭합니다.

## 2.5. 호스트 그룹에 ANSIBLE 역할 할당

Satellite 클라이언트의 원격 관리를 위해 Ansible 역할을 사용할 수 있습니다.

### 사전 요구 사항

- Ansible 역할을 실행하도록 배포를 구성해야 합니다. 자세한 내용은 1.2절. [“Ansible 역할을 실행하도록 Satellite 구성”](#)의 내용을 참조하십시오.

### 절차

1. Satellite 웹 UI에서 **Configure > Host Groups** 로 이동합니다.
2. Ansible 역할을 할당할 호스트 그룹 이름을 클릭합니다.
3. **Ansible Roles** 탭의 **Available Ansible Roles** 목록에서 추가할 역할을 선택합니다.
4. + 아이콘을 클릭하여 호스트 그룹에 역할을 추가합니다. 두 개 이상의 역할을 추가할 수 있습니다.
5. **Submit** 을 클릭합니다.

## 2.6. 호스트 그룹에서 ANSIBLE 역할 실행

Satellite 웹 UI를 통해 호스트 그룹에서 Ansible 역할을 실행할 수 있습니다.

### 사전 요구 사항

- Ansible 역할을 실행하도록 배포를 구성해야 합니다. 자세한 내용은 1.2절. "Ansible 역할을 실행하도록 Satellite 구성"의 내용을 참조하십시오.
- 호스트 그룹에 Ansible 역할을 할당해야 합니다.
- 호스트 그룹에 하나 이상의 호스트가 있어야 합니다.

### 절차

1. Satellite 웹 UI에서 **Configure > Host Groups** 로 이동합니다.
2. 호스트 그룹의 **Actions** (작업) 열의 목록에서 **Run all Ansible roles** 를 선택합니다.

Ansible 작업의 상태는 **Run Ansible roles** 페이지에서 볼 수 있습니다. **Rerun** 을 클릭하여 작업을 재실행합니다.

## 2.7. 검사 모드에서 ANSIBLE 역할 실행

Satellite 웹 UI를 통해 검사 모드에서 Ansible 역할을 실행할 수 있습니다.

### 사전 요구 사항

- Ansible 역할을 실행하도록 배포를 구성해야 합니다. 자세한 내용은 1.2절. "Ansible 역할을 실행하도록 Satellite 구성"의 내용을 참조하십시오.
- 호스트 그룹에 Ansible 역할을 할당해야 합니다.
- 호스트 그룹에 하나 이상의 호스트가 있어야 합니다.

### 절차

1. Satellite 웹 UI에서 호스트 > 모든 호스트로 이동합니다.
2. 점검 모드를 활성화할 호스트에 대해 **Edit** (편집)를 클릭합니다.
3. **Parameters** 탭에서 **부울** 유형이 **true** 로 설정된 **ansible\_roles\_check\_mode** 라는 매개 변수가 있는지 확인합니다.
4. **Submit** 을 클릭합니다.

## 3장. SATELLITE에서 ANSIBLE 플레이북 실행

Satellite에서 원격 작업을 실행하여 호스트 또는 호스트 그룹에서 Ansible 플레이북을 실행할 수 있습니다.

### Ansible 플레이북 작업 템플릿의 호스트 매개변수 제한

여러 호스트에서 Ansible 플레이북을 실행하는 경우 Satellite는 배치의 모든 호스트에 대해 플레이북을 렌더링하지만 첫 번째 호스트의 렌더링된 플레이북만 사용하여 배치의 모든 호스트에서 실행합니다. 따라서 템플릿 제어 흐름 구성에서 host 매개 변수를 사용하여 호스트당 플레이북의 동작을 수정할 수 없습니다. 호스트 매개 변수는 Ansible 변수로 변환되므로 이를 사용하여 기본 Ansible 구문의 동작을 제어할 수 있습니다. 자세한 내용은 [BZ#2282275](#) 를 참조하십시오.

### 사전 요구 사항

- Satellite에서 Ansible 플러그인이 활성화되어 있습니다.
- 원격 작업 실행이 구성되어 있습니다. 자세한 내용은 [4장. 원격 작업 구성 및 설정](#)의 내용을 참조하십시오.
- Ansible 플레이북을 사용할 준비가 되어 있습니다.

### 프로세스

1. Satellite 웹 UI에서 **Monitor > Jobs** 로 이동합니다.
2. **Run Job** 을 클릭합니다.
3. **Job category** 에서 **Ansible Playbook** 을 선택합니다.
4. 작업 템플릿에서 **Ansible - Run playbook** 을 선택합니다.
5. 다음을 클릭합니다.
6. Playbook을 실행할 호스트를 선택합니다.
7. 플레이북 필드에 Ansible 플레이북의 콘텐츠를 붙여넣습니다.
8. 마법사에 따라 원격 작업 설정을 완료합니다. 자세한 내용은 [4.21절. "원격 작업 실행"](#)의 내용을 참조하십시오.
9. **Submit** 을 클릭하여 호스트에서 Ansible 플레이북을 실행합니다.

### 추가 리소스

- 또는 Capsule Server에서 Ansible 플레이북을 가져올 수 있습니다. 자세한 내용은 다음 리소스를 참조하십시오.
  - [4.7절. "이름으로 Ansible 플레이북 가져오기"](#)
  - [4.8절. "사용 가능한 모든 Ansible 플레이북 가져오기"](#)

## 4장. 원격 작업 구성 및 설정

Red Hat Satellite는 호스트에서 원격 명령 실행을 지원합니다. 원격 실행을 사용하면 여러 호스트에서 동시에 다양한 작업을 수행할 수 있습니다.

### 4.1. RED HAT SATELLITE에서 원격 실행

원격 실행을 사용하면 셸 스크립트 또는 Ansible 작업 및 플레이북을 사용하여 Capsule에서 원격으로 호스트에서 작업을 실행할 수 있습니다.

Satellite에서 다음과 같은 이점을 위해 원격 실행을 사용하십시오.

- 한 번에 여러 호스트에서 작업을 실행합니다.
- 실행하는 작업을 보다 세밀하게 제어하려면 명령에서 변수를 사용합니다.
- 호스트 팩트 및 매개 변수를 사용하여 변수 값을 채웁니다.
- 명령을 실행할 때 템플릿의 사용자 지정 값을 지정합니다.

Capsule Server를 통해 원격 실행 통신이 이루어지므로 Satellite Server는 대상 호스트에 대한 직접 액세스 권한이 필요하지 않으며 많은 호스트를 관리하도록 확장할 수 있습니다. 자세한 내용은 [4.4절. "원격 실행을 위한 전송 모드"](#)의 내용을 참조하십시오.

원격 실행을 사용하려면 작업 템플릿을 정의해야 합니다. 작업 템플릿은 원격 호스트에 적용할 명령입니다. 작업 템플릿을 여러 번 실행할 수 있습니다.

Satellite는 ERB 구문 작업 템플릿을 사용합니다. 자세한 내용은 호스트 관리의 [템플릿 작성 참조](#)를 참조하십시오.

기본적으로 Satellite에는 셸 스크립트 및 Ansible을 위한 여러 작업 템플릿이 포함되어 있습니다. 자세한 내용은 [호스트 관리에서 작업 템플릿 설정](#)을 참조하십시오.

#### 추가 리소스

- [호스트 관리에서 원격 작업 실행](#)을 참조하십시오.

### 4.2. 원격 실행 워크플로

다운로드하는 사용자 지정 Ansible 역할 또는 다운로드한 역할의 경우 Capsule Server에 역할이 포함된 패키지를 설치해야 합니다. Ansible 역할을 사용하려면 먼저 해당 역할이 설치된 Capsule에서 역할을 Satellite로 가져와야 합니다.

호스트에서 원격 작업을 실행할 때 모든 호스트에 대해 Satellite는 다음 작업을 수행하여 사용할 원격 실행 Capsule을 찾습니다.

Satellite는 Ansible 기능이 활성화된 Capsule만 검색합니다.

1. Satellite는 **원격 실행** 확인란이 선택된 호스트의 인터페이스를 찾습니다.
2. Satellite는 이러한 인터페이스의 서브넷을 찾습니다.
3. Satellite는 이러한 서브넷에 할당된 원격 실행 Capsule을 찾습니다.

- 이 Capsule 세트에서 Satellite는 실행 중인 작업 수가 가장 적은 Capsule을 선택합니다. 이렇게 하면 Satellite를 통해 작업 부하가 원격 실행 Capsule 간에 균형을 유지합니다.

원격 실행을 위해 Capsule을 통해 등록된 Prefer를 활성화한 경우 Satellite는 호스트가 등록된 Capsule을 사용하여 REX 작업을 실행합니다.

기본적으로 원격 실행을 위해 Capsule을 통해 등록된 Prefer는 No로 설정됩니다. 이를 활성화하려면 Satellite 웹 UI에서 **Administer > Settings**로 이동하여 **Content** 탭에서 **원격 실행을 위해 Capsule을 통해 등록된 Prefer**를 **Yes**로 설정합니다. 이렇게 하면 Satellite가 등록된 Capsule에 의해 호스트에서 REX 작업을 수행합니다.

Satellite가 이 단계에서 원격 실행 Capsule을 찾지 못하고 모든 Capsule 설정에 대한 폴백이 활성화된 경우 Satellite는 다른 Capsule 세트를 추가하여 원격 실행 Capsule을 선택합니다. Satellite는 호스트에 할당된 다음 유형의 Capsule에서 가장 쉽게 로드된 Capsule을 선택합니다.

- 호스트의 서브넷에 할당된 DHCP, DNS 및 TFTP Capsule
- 호스트 도메인에 할당된 DNS Capsule
- 호스트 영역에 할당된 realm Capsule
- Puppet 서버 Capsule
- Puppet CA Capsule
- OpenSCAP Capsule

Satellite가 이 단계에서 원격 실행 Capsule을 찾지 못하고 **Enable Global Capsule** 설정이 활성화된 경우 Satellite는 원격 작업을 실행하기 위한 호스트 조직 및 위치 집합에서 가장 가벼우 로드된 원격 실행 Capsule을 선택합니다.

### 4.3. 원격 실행에 대한 권한

대상으로 할 수 있는 호스트를 포함하여 인프라 내에서 작업을 실행할 수 있는 역할을 제어할 수 있습니다. 원격 실행 기능은 다음 두 가지 기본 제공 역할을 제공합니다.

- **원격 실행 관리자:** 모든 원격 실행 기능 및 기능에 액세스할 수 있습니다.
- **원격 실행 사용자:** 작업을 실행할 수만 있습니다.

**Remote Execution User** 역할을 복제하고 필터를 사용자 지정하여 세분화를 늘릴 수 있습니다. 사용자 지정 역할에 대한 **view\_job\_templates** 권한을 사용하여 필터를 조정하는 경우 일치하는 작업 템플릿을 기반으로만 작업을 보고 트리거할 수 있습니다. **view\_hosts** 및 **view\_smart\_proxies** 권한을 사용하여 역할에 표시되는 호스트 또는 Capsule을 제한할 수 있습니다.

**execute\_template\_invocation** 권한은 작업 실행이 시작되기 직전에 확인되는 특수 권한입니다. 이 권한은 특정 호스트에서 실행할 수 있는 작업 템플릿을 정의합니다. 이렇게 하면 권한을 지정할 때 더 세분화할 수 있습니다.

**execute\_jobs\_on\_infrastructure\_hosts** 권한을 사용하여 Red Hat Satellite에 등록된 Capsule 및 Red Hat Satellite에 대해 원격 실행 작업을 실행할 수 있습니다. 표준 관리자 및 사이트 관리자 역할에는 기본적으로 이 권한이 있습니다. **Manager** 또는 **Site Manager** 역할을 사용하거나

**execute\_jobs\_on\_infrastructure\_hosts** 권한과 함께 사용자 지정 역할을 사용하는 경우 등록된 Red Hat Satellite 및 Capsule 호스트에 대해 원격 작업을 실행할 수 있습니다.

역할 및 권한 사용에 대한 자세한 내용은 *Red Hat Satellite 관리에서 역할 생성 및 관리*를 참조하십시오.

다음 예제에서는 `execute_template_invocation` 권한에 대한 필터를 보여줍니다.

```
name = Reboot and host.name = staging.example.com
name = Reboot and host.name ~ *.staging.example.com
name = "Restart service" and host_group.name = webserver
```

이 예제의 첫 번째 행을 사용하여 선택한 하나의 호스트에 **Reboot** 템플릿을 적용합니다. 두 번째 행을 사용하여 **.staging.example.com** 으로 끝나는 이름이 있는 호스트 풀을 정의합니다. 세 번째 행을 사용하여 템플릿을 호스트 그룹과 바인딩합니다.



**참고**

이러한 역할을 가진 사용자에게 할당된 권한은 시간이 지남에 따라 변경될 수 있습니다. 나중에 실행되도록 일부 작업을 이미 예약했으며 권한이 변경되면 작업이 실행되기 직전에 권한이 확인되므로 실행 오류가 발생할 수 있습니다.

### 4.4. 원격 실행을 위한 전송 모드

원격 작업 실행을 위해 두 가지 전송 모드를 사용하도록 Satellite를 구성할 수 있습니다. 하나의 모드 또는 다른 모드를 사용하도록 단일 Capsule을 구성할 수 있지만 둘 다 사용할 수는 없습니다.

**푸시 기반 전송**

**ssh** 모드의 Capsule에서 원격 실행에서는 SSH 서비스를 사용하여 작업 세부 정보를 전송합니다. 기본 전송 모드입니다. 대상 호스트에서 SSH 서비스를 활성화하고 활성화해야 합니다. 원격 실행 Capsule은 대상 호스트의 SSH 포트에 액세스할 수 있어야 합니다. 다른 설정이 없는 경우 표준 SSH 포트는 22입니다.

이 전송 모드는 스크립트 및 Ansible 공급자를 모두 지원합니다.

**풀 기반 전송**

**pull-mqtt** 모드의 Capsule에서 원격 실행은 MQTT(Message Queueing Telemetry Transport)를 사용하여 Satellite Server에서 수신하는 작업 실행을 시작합니다. 호스트는 **yggdrasil** 풀 클라이언트를 사용하여 작업 알림을 위해 Capsule의 MQTT 브로커를 구독합니다. 호스트에서 MQTT 브로커에서 알림을 수신하면 HTTPS를 통해 Capsule에서 작업 세부 정보를 가져오고 작업을 실행하고 결과를 Capsule에 다시 보고합니다.

이 전송 모드는 스크립트 공급자만 지원합니다.

**pull-mqtt** 모드를 사용하려면 Capsule Server에서 이를 활성화하고 호스트에서 가져오기 클라이언트를 구성해야 합니다.



**참고**

Capsule에서 이미 **pull-mqtt** 모드를 사용하고 **ssh** 모드로 다시 전환하려면 다음 **satellite-installer** 명령을 실행합니다.

```
# satellite-installer --foreman-proxy-plugin-remote-execution-script-mode=ssh
```

**추가 리소스**

- Capsule Server에서 가져오기 모드를 활성화하려면 Capsule Server 설치에서 [원격 실행을 위한 풀 기반 전송 구성](#) 을 참조하십시오.

- 등록된 호스트에서 가져오기 모드를 활성화하려면 [4.5절. "풀 클라이언트를 사용하도록 호스트 구성"](#) 을 계속 진행합니다.
- 새 호스트에서 가져오기 모드를 활성화하려면 호스트 관리에서 다음을 계속합니다.
  - [호스트 생성](#)
  - [호스트 등록](#)

## 4.5. 풀 클라이언트를 사용하도록 호스트 구성

**pull-mqtt** 모드를 사용하도록 구성된 Capsule의 경우 호스트는 원격 실행 가져오기 클라이언트를 사용하여 원격 작업을 구독할 수 있습니다. 호스트에는 Capsule Server의 SSH 연결이 필요하지 않습니다.

### 사전 요구 사항

- 호스트를 Satellite에 등록했습니다.
- 호스트가 등록된 Capsule은 **pull-mqtt** 모드를 사용하도록 구성되어 있습니다. 자세한 내용은 *Capsule Server* 설치에서 [원격 실행을 위한 풀 기반 전송 구성](#) 을 참조하십시오.
- 호스트의 운영 체제 버전에 대한 Red Hat Satellite Client 6 리포지토리는 Satellite Server에서 동기화되며, 콘텐츠 뷰 및 호스트의 라이프사이클 환경에서 사용할 수 있으며 호스트에 대해 활성화됩니다. 자세한 내용은 [콘텐츠 관리에서 Satellite에서 호스트의 리포지토리 세트 상태 변경을 참조하십시오.](#)
- 호스트는 포트 **1883** 을 사용하여 MQTT를 통해 Capsule과 통신할 수 있습니다.
- 호스트는 HTTPS를 통해 Capsule과 통신할 수 있습니다.

### 프로세스

- 호스트에 **katello-pull-transport-migrate** 패키지를 설치합니다.
  - Red Hat Enterprise Linux 9 및 Red Hat Enterprise Linux 8 호스트에서 다음을 수행합니다.

```
# dnf install katello-pull-transport-migrate
```

- Red Hat Enterprise Linux 7 호스트에서 다음을 수행합니다.

```
# yum install katello-pull-transport-migrate
```

이 패키지는 **foreman\_ygg\_worker** 및 **yggdrasil** 을 종속 항목으로 설치하고 **yggdrasil** 클라이언트를 구성하고 호스트에서 풀 클라이언트 작업자를 시작합니다.

### 검증

- **yggdrasild** 서비스의 상태를 확인합니다.

```
# systemctl status yggdrasild
```

## 4.6. 작업 템플릿 생성

작업 템플릿을 생성하려면 다음 절차를 사용하십시오. Satellite 웹 UI 대신 CLI를 사용하려면 [CLI 절차](#)를 참조하십시오.

## 프로세스

1. Satellite 웹 UI에서 **호스트 > 템플릿 > 작업 템플릿** 으로 이동합니다.
2. **새 작업 템플릿**을 클릭합니다.
3. **템플릿** 탭을 클릭하고 **이름** 필드에 작업 템플릿의 고유한 이름을 입력합니다.
4. **Default**를 선택하여 모든 조직 및 위치에 템플릿을 사용할 수 있도록 합니다.
5. 템플릿 편집기에서 템플릿을 직접 생성하거나 **가져오기**를 클릭하여 텍스트 파일에서 업로드합니다.
6. 선택 사항: **감사 주석** 필드에서 변경 사항에 대한 정보를 추가합니다.
7. **Job** 탭을 클릭하고 **Job category** 필드에 고유한 범주를 입력하거나 **호스트 관리의 기본 작업 템플릿 Categories에 나열된 기본** 카테고리에서 선택합니다.
8. 선택 사항: **설명 형식** 필드에 설명 템플릿을 입력합니다. 예를 들어 **Install package % {package\_name}**. 템플릿에 **%{template\_name}** 및 **%{job\_category}**를 사용할 수도 있습니다.
9. **공급자 유형** 목록에서 셸 스크립트의 경우 **SSH**를 선택하고 **Ansible** 작업 또는 플레이북에 대해 **Ansible**을 선택합니다.
10. 선택 사항: **시간 초과** 필드에 시간 초과 값을 입력하여 작업이 완료되지 않은 경우 작업을 종료합니다.
11. 선택 사항: **입력 추가**를 클릭하여 입력 매개변수를 정의합니다. 작업을 실행할 때 매개 변수가 요청되며 템플릿에 정의할 필요가 없습니다. 예를 들어 **도움말** 탭을 참조하십시오.
12. 선택 사항: 이 작업에 다른 템플릿을 포함하도록 **외래 입력 세트**를 클릭합니다.
13. 선택 사항: **효과적인 사용자** 영역에서 명령이 기본 **remote\_execution\_effective\_user** 설정을 사용할 수 없는 경우 사용자를 구성합니다.
14. 선택 사항: 이 템플릿이 다른 템플릿에 포함할 스니펫인 경우 **유형** 탭을 클릭하고 **Snippet**을 선택합니다.
15. 선택 사항: Ansible 공급자를 사용하는 경우 **Ansible** 탭을 클릭합니다. 작업이 완료된 후 구성 보고서를 생성하는 데 사용되는 팩트를 호스트가 보낼 수 있도록 하려면 **Ansible 콜백 사용**을 선택합니다.
16. **위치** 탭을 클릭하고 템플릿을 사용할 위치를 추가합니다.
17. **조직** 탭을 클릭하고 템플릿을 사용할 조직을 추가합니다.
18. **Submit**을 클릭하여 변경 사항을 저장합니다.

템플릿 구문에 다른 템플릿을 포함하여 작업 템플릿을 확장하고 사용자 지정할 수 있습니다. 자세한 내용은 [템플릿 작성 참조](#) 및 [작업 템플릿 예 및 호스트 관리의 확장](#)을 참조하십시오.

## CLI 절차

- `template-definition` 파일을 사용하여 작업 템플릿을 생성하려면 다음 명령을 입력합니다.



```
# hammer job-template create \
--file "Path_to_My_Template_File" \
--job-category "My_Category_Name" \
--name "My_Template_Name" \
--provider-type SSH
```

## 4.7. 이름으로 ANSIBLE 플레이북 가져오기

Capsule에 설치된 컬렉션에서 이름별로 Ansible 플레이북을 Satellite로 가져올 수 있습니다. Satellite는 가져온 플레이북에서 작업 템플릿을 생성하고 템플릿을 **Ansible Playbook - Imported** 작업 카테고리에 배치합니다.

사용자 지정 컬렉션이 있는 경우

**/etc/ansible/collections/ansible\_collections/My\_Namespace/My\_Collection**에 배치합니다.

### 사전 요구 사항

- Ansible 플러그인이 활성화되어 있습니다.
- Satellite 계정에는 **import\_ansible\_playbooks** 권한을 부여하는 역할이 있습니다.

### 프로세스

1. 다음 API 요청을 사용하여 사용 가능한 Ansible 플레이북을 가져옵니다.

```
# curl -X GET -H 'Content-Type: application/json'
https://satellite.example.com/ansible/api/v2/ansible_playbooks/fetch?
proxy_id=My_capsule_ID
```

2. 가져올 Ansible 플레이북을 선택하고 해당 이름을 확인합니다.
3. 이름으로 Ansible 플레이북을 가져옵니다.

```
# curl -X PUT -H 'Content-Type: application/json' -d '{"playbook_names":
["My_Playbook_Name"]}'
https://satellite.example.com/ansible/api/v2/ansible_playbooks/sync?
proxy_id=My_capsule_ID
```

가져오기가 완료된 후 Satellite 웹 UI에 알림이 표시됩니다.

### 다음 단계

- 생성된 작업 템플릿에서 원격 작업을 실행하여 플레이북을 실행할 수 있습니다. 자세한 내용은 [4.21절. "원격 작업 실행"](#)의 내용을 참조하십시오.

## 4.8. 사용 가능한 모든 ANSIBLE 플레이북 가져오기

Capsule에 설치된 컬렉션에서 사용 가능한 모든 Ansible 플레이북을 Satellite로 가져올 수 있습니다. Satellite는 가져온 플레이북에서 작업 템플릿을 생성하고 템플릿을 **Ansible Playbook - Imported** 작업 카테고리에 배치합니다.

사용자 지정 컬렉션이 있는 경우

**/etc/ansible/collections/ansible\_collections/My\_Namespace/My\_Collection**에 배치합니다.

### 사전 요구 사항

- Ansible 플러그인이 활성화되어 있습니다.
- Satellite 계정에는 **import\_ansible\_playbooks** 권한을 부여하는 역할이 있습니다.

### 프로세스

- 다음 API 요청을 사용하여 Ansible 플레이북을 가져옵니다.

```
# curl -X PUT -H 'Content-Type: application/json'
https://satellite.example.com/ansible/api/v2/ansible_playbooks/sync?
proxy_id=My_capsule_ID
```

가져오기가 완료된 후 Satellite 웹 UI에 알림이 표시됩니다.

### 다음 단계

- 생성된 작업 템플릿에서 원격 작업을 실행하여 플레이북을 실행할 수 있습니다. 자세한 내용은 [4.21절. "원격 작업 실행"](#)의 내용을 참조하십시오.

## 4.9. SATELLITE에서 CAPSULE 원격 실행 설정에 대한 폴백 구성

**Fallback**을 모든 **Capsule** 설정으로 활성화하여 호스트에 할당된 Capsule 목록에서 원격 실행 Capsule을 검색하도록 Satellite를 구성할 수 있습니다. 이 기능은 서브넷이 구성되지 않은 호스트에서 원격 작업을 실행해야 하거나 원격 실행 기능이 활성화된 Capsule에 호스트의 서브넷이 할당된 경우 유용할 수 있습니다.

**Fallback to Any Capsule** 설정이 활성화된 경우 Satellite는 다른 Capsule 세트를 추가하여 원격 실행 Capsule을 선택합니다. 또한 Satellite는 다음과 같이 호스트에 할당된 모든 Capsule 세트에서 가장 간단한 로드된 Capsule을 선택합니다.

- 호스트의 서브넷에 할당된 DHCP, DNS 및 TFTP Capsule
- 호스트 도메인에 할당된 DNS Capsule
- 호스트 영역에 할당된 realm Capsule
- Puppet 서버 Capsule
- Puppet CA Capsule
- OpenSCAP Capsule

### 절차

1. Satellite 웹 UI에서 **Administer > Settings** 로 이동합니다.
2. **원격 실행**을 클릭합니다.
3. **Fallback**을 모든 **Capsule** 설정으로 구성합니다.

### CLI 절차

- Satellite에 **hammer settings set** 명령을 입력하여 **Fallback**을 **Any Capsule** 설정으로 구성합니다. 값을 **true** 로 설정하려면 다음 명령을 입력합니다.

```
# hammer settings set \
--name=remote_execution_fallback_proxy \
--value=true
```

## 4.10. SATELLITE에서 글로벌 CAPSULE 원격 실행 설정 구성

기본적으로 Satellite는 Capsule이 호스트의 서브넷에 할당되었는지 여부에 관계없이 호스트의 조직 및 위치에서 원격 실행 Capsule을 검색합니다. 호스트 서브넷에 할당된 **Capsule**로 검색을 제한하려면 **글로벌 Capsule 활성화** 설정을 비활성화할 수 있습니다.

**Enable Global Capsule** 설정이 활성화된 경우 Satellite는 다른 Capsule 세트를 추가하여 원격 실행 Capsule을 선택합니다. Satellite는 또한 호스트의 조직 및 원격 작업을 실행할 위치에 있는 모든 Capsule 세트에서 가장 쉽게 로드된 원격 실행 Capsule을 선택합니다.

### 절차

1. Satellite 웹 UI에서 **Administer > Settings** 로 이동합니다.
2. **원격 실행**을 클릭합니다.
3. **글로벌 Capsule 활성화** 설정을 구성합니다.

### CLI 절차

- Satellite에 **hammer settings set** 명령을 입력하여 **글로벌 Capsule 활성화** 설정을 구성합니다. 값을 **true** 로 설정하려면 다음 명령을 입력합니다.

```
# hammer settings set \
--name=remote_execution_global_proxy \
--value=true
```

## 4.11. 대체 디렉토리를 사용하여 호스트에서 원격 작업을 실행하도록 SATELLITE 구성

Ansible은 서버 측에 필요한 자체 파일을 **/tmp** 디렉토리에 배치합니다. 필요한 경우 다른 디렉토리를 설정할 수 있는 옵션이 있습니다.

### 프로세스

1. Satellite Server 또는 Capsule Server에서 새 디렉토리를 생성합니다.

```
# mkdir /My_Remote_Working_Directory
```

2. 기본 **/tmp** 디렉토리에서 SELinux 컨텍스트를 복사합니다.

```
# chcon --reference=/tmp /My_Remote_Working_Directory
```

3. 새 디렉토리를 사용하도록 Satellite Server 또는 Capsule Server를 구성합니다.

```
# satellite-installer \
--foreman-proxy-plugin-ansible-working-dir /My_Remote_Working_Directory
```

## 4.12. 권한 상승 방법 변경

기본적으로 푸시 기반 원격 실행에서는 **sudo** 를 사용하여 SSH 사용자에서 호스트에서 스크립트를 실행하는 유효한 사용자로 전환합니다. 경우에 따라 **su** 또는 **dzdo** 와 같은 다른 방법을 사용해야 할 수도 있습니다. Satellite 설정에서 대체 방법을 전역적으로 구성할 수 있습니다.

### 사전 요구 사항

- 사용자 계정에는 **view\_settings** 및 **edit\_settings** 권한을 부여하는 역할이 할당됩니다.
- Ansible 작업에 **dzdo** 를 사용하려면 필요한 **dzdo become** 플러그인이 포함된 **community.general** Ansible 컬렉션이 설치되어 있는지 확인합니다. 자세한 내용은 *Ansible 문서의 컬렉션 설치*를 참조하십시오.

### 프로세스

1. **Administer > Settings** 로 이동합니다.
2. **원격 실행** 탭을 선택합니다.
3. **Effective User Method** 설정 값을 클릭합니다.
4. 새 값을 선택합니다.
5. **Submit** 을 클릭합니다.

## 4.13. 원격 실행을 위한 SSH 키 배포

**ssh** 모드의 Capsule의 경우 SSH를 사용하여 원격 실행 연결이 인증됩니다. Capsule의 공개 SSH 키는 관리하려는 연결된 호스트에 배포해야 합니다.

SSH 서비스가 활성화되어 호스트에서 실행 중인지 확인합니다. 포트 22에 액세스할 수 있도록 모든 네트워크 또는 호스트 기반 방화벽을 구성합니다.

다음 방법 중 하나를 사용하여 공개 SSH 키를 Capsule에서 대상 호스트로 배포합니다.

1. 4.14절. "원격 실행을 위해 SSH 키 배포".
2. 4.16절. "Satellite API를 사용하여 원격 실행을 위한 SSH 키 가져오기".
3. 4.17절. "프로비저닝 중에 SSH 키를 배포하도록 Kickstart 템플릿 구성".
4. 새 Satellite 호스트의 경우 글로벌 등록 템플릿을 사용하여 등록하는 동안 SSH 키를 Satellite 호스트에 배포할 수 있습니다. 자세한 내용은 *호스트 관리의 글로벌 등록 템플릿을 사용하여 Red Hat Satellite* 에 호스트 등록을 참조하십시오.

Satellite는 원격 실행 기능의 SSH 키를 기본적으로 Satellite에서 프로비저닝된 호스트에 배포합니다.

호스트가 Amazon Web Services에서 실행 중인 경우 암호 인증을 활성화합니다. 자세한 내용은 *새 사용자 계정을 참조*하십시오.

## 4.14. 원격 실행을 위해 SSH 키 배포

SSH 키를 수동으로 배포하려면 다음 단계를 완료합니다.

### 절차

- SSH pub 키를 Capsule에서 대상 호스트로 복사합니다.

```
# ssh-copy-id -i ~foreman-proxy/.ssh/id_rsa_foreman_proxy.pub root@client.example.com
```

관리하려는 각 대상 호스트에 대해 이 단계를 반복합니다.

### 검증

- 키가 대상 호스트에 성공적으로 복사되었는지 확인하려면 Capsule에 다음 명령을 입력합니다.

```
# ssh -i ~foreman-proxy/.ssh/id_rsa_foreman_proxy root@client.example.com
```

## 4.15. 원격 실행에 사용되는 SSH 키에 암호 추가

기본적으로 Capsule은 보호되지 않은 SSH 키를 사용하여 호스트에서 원격 작업을 실행합니다. 다음 절차에 따라 암호를 사용하여 SSH 키를 보호할 수 있습니다.

### 프로세스

- Satellite Server 또는 Capsule Server에서 **ssh-keygen** 을 사용하여 SSH 키에 암호를 추가합니다.

```
# ssh-keygen -p -f ~foreman-proxy/.ssh/id_rsa_foreman_proxy
```

### 다음 단계

- 이제 호스트에서 원격 실행 작업을 실행할 때 암호를 사용해야 합니다.

## 4.16. SATELLITE API를 사용하여 원격 실행을 위한 SSH 키 가져오기

Satellite API를 사용하여 Capsule에서 공개 키를 다운로드하려면 각 대상 호스트에서 이 절차를 완료합니다.

### 절차

1. 대상 호스트에서 **~/.ssh** 디렉토리를 생성하여 SSH 키를 저장합니다.

```
# mkdir ~/.ssh
```

2. Capsule에서 SSH 키를 다운로드합니다.

```
# curl https://capsule.example.com:9090/ssh/pubkey >> ~/.ssh/authorized_keys
```

3. **~/.ssh** 디렉토리에 대한 권한을 구성합니다.

```
# chmod 700 ~/.ssh
```

4. **authorized\_keys** 파일에 대한 권한을 구성합니다.

```
# chmod 600 ~/.ssh/authorized_keys
```

## 4.17. 프로비저닝 중에 SSH 키를 배포하도록 KICKSTART 템플릿 구성

사용자 지정 Kickstart 템플릿에 **remote\_execution\_ssh\_keys** 스니펫을 추가하여 프로비저닝 중에 SSH 키를 호스트에 배포할 수 있습니다. Satellite에서 제공하는 Kickstart 템플릿에는 기본적으로 이 스니펫이 포함되어 있습니다. Satellite는 원격 실행을 위한 SSH 키를 프로비저닝 중에 시스템에 복사합니다.

### 절차

- 공개 키를 새로 프로비저닝된 호스트에 포함하려면 사용하는 Kickstart 템플릿에 다음 스니펫을 추가합니다.

```
<%= snippet 'remote_execution_ssh_keys' %>
```

## 4.18. 티켓을 부여하는 KERBEROS 티켓의 키탭 구성

키탭을 사용하여 티켓을 부여하는 Kerberos 티켓을 가져오도록 Satellite를 구성하려면 다음 절차를 사용하십시오. 키탭을 설정하지 않으면 수동으로 티켓을 검색해야 합니다.

### 절차

1. **foreman-proxy** 사용자의 ID를 찾습니다.

```
# id -u foreman-proxy
```

2. 새 파일에 권한 **600**이 있도록 **Cryostat** 값을 수정합니다.

```
# umask 077
```

3. 키탭의 디렉토리를 만듭니다.

```
# mkdir -p "/var/kerberos/krb5/user/My_User_ID"
```

4. 키탭을 만들거나 기존 키탭을 디렉토리에 복사합니다.

```
# cp My_Client.keytab /var/kerberos/krb5/user/My_User_ID/client.keytab
```

5. 디렉토리 소유자를 **foreman-proxy** 사용자로 변경합니다.

```
# chown -R foreman-proxy:foreman-proxy "/var/kerberos/krb5/user/My_User_ID"
```

6. 키탭 파일이 읽기 전용인지 확인합니다.

```
# chmod -wx "/var/kerberos/krb5/user/My_User_ID/client.keytab"
```

7. **SELinux** 컨텍스트를 복원합니다.

```
# restorecon -RvF /var/kerberos/krb5
```

#### 4.19. 원격 실행을 위한 **KERBEROS** 인증 구성

**Kerberos** 인증을 사용하여 **Satellite** 호스트에서 원격 실행에 대한 **SSH** 연결을 설정할 수 있습니다.

##### 사전 요구 사항

- **Kerberos** 서버에 **Satellite Server** 등록
- **Kerberos** 서버에 **Satellite** 대상 호스트 등록
- 원격 실행을 위해 **Kerberos** 사용자 계정 구성 및 초기화
- **Satellite**의 **foreman-proxy** 사용자에게 유효한 **Kerberos** 티켓 부여 티켓이 있는지 확인합니다.

##### 절차

1. 원격 실행을 위해 **Kerberos** 인증을 설치하고 활성화하려면 다음 명령을 입력합니다.

```
# satellite-installer --foreman-proxy-plugin-remote-execution-script-ssh-kerberos-auth true
```

2. 원격 실행을 위해 기본 사용자를 편집하려면 **Satellite** 웹 UI에서 **Administer > Settings** 로 이동하여 원격 실행 탭을 클릭합니다. **SSH** 사용자 행에서 두 번째 열을 편집하고 **Kerberos** 계정의 사용자 이름을 추가합니다.

3. `remote_execution_effective_user` 로 이동하고 두 번째 열을 편집하여 **Kerberos** 계정의 사용자 이름을 추가합니다.

검증

- **Kerberos** 인증을 사용할 준비가 되었는지 확인하려면 호스트에서 원격 작업을 실행합니다. 자세한 내용은 **Ansible 통합을 사용하여 구성 관리에서 원격 작업 실행**을 참조하십시오.

### 4.20. 작업 템플릿 설정

**Satellite**는 작업 실행에 사용할 수 있는 기본 작업 템플릿을 제공합니다. 작업 템플릿 목록을 보려면 호스트 > 템플릿 > 작업 템플릿 으로 이동합니다. 변경하지 않고 템플릿을 사용하려면 **관리 호스트에서 원격 작업 실행**으로 이동합니다.

기본 템플릿을 자체 개발을 위한 기반으로 사용할 수 있습니다. 편집할 기본 작업 템플릿이 잠겼습니다. 템플릿을 복제하고 복제본을 편집합니다.

절차

1. 템플릿을 복제하려면 **Actions** 열에서 **Clone** 을 선택합니다.
2. 복제본의 고유 이름을 입력하고 **Submit** 을 클릭하여 변경 사항을 저장합니다.

작업 템플릿은 **ERP(ERB)** 구문을 사용합니다. 템플릿 작성에 대한 자세한 내용은 **호스트 관리의 템플릿 작성 참조**를 참조하십시오.

### Ansible 고려 사항

**Ansible** 작업 템플릿을 생성하려면 다음 절차를 사용하고 **ERB** 구문 대신 **YAML** 구문을 사용합니다. `--` 로 템플릿을 시작합니다. **Ansible** 플레이북 **YAML** 파일을 작업 템플릿 본문에 포함할 수 있습니다. **ERB** 구문을 추가하여 **YAML Ansible** 템플릿을 사용자 지정할 수도 있습니다. **Satellite**에서 **Ansible** 플레이북을 가져올 수도 있습니다. 자세한 내용은 **호스트 관리의 리포지토리 템플릿 동기화**를 참조하십시오.

### 매개변수 변수

런타임에 작업 템플릿은 호스트에 대해 정의한 매개변수 변수를 허용할 수 있습니다. 호스트의 편집 페이지의 **Parameters** 탭에 표시되는 매개변수만 작업 템플릿의 입력 매개 변수로 사용할 수 있습니다.



## 4.21. 원격 작업 실행

하나 이상의 호스트에 대해 작업 템플릿을 기반으로 하는 작업을 실행할 수 있습니다.

**Satellite 웹 UI** 대신 **CLI**를 사용하려면 [CLI 절차](#) 를 참조하십시오.

### 프로세스

1. **Satellite 웹 UI**에서 **Monitor > Jobs** 로 이동하여 **Run job** 을 클릭합니다.
2. 작업 범주와 사용할 작업 템플릿을 선택한 다음 다음을 클릭합니다.
3. 작업을 실행할 호스트를 선택합니다. 호스트를 선택하지 않으면 현재 컨텍스트에서 볼 수 있는 모든 호스트에서 작업이 실행됩니다.



### 참고

호스트 그룹과 모든 하위 그룹을 선택하려면 하위 그룹의 호스트가 아닌 해당 그룹에서 직접 호스트에서 실행되므로 호스트 그룹을 선택하는 것만으로는 충분하지 않습니다. 대신 호스트 그룹 및 모든 하위 그룹을 선택하거나 다음 검색 쿼리를 사용해야 합니다.

```
hostgroup_fullname ~ "My_Host_Group"
```

**My\_Host\_Group** 을 최상위 호스트 그룹의 이름으로 교체합니다.

4. 필요한 경우 작업 템플릿에 대한 입력을 제공합니다. 템플릿마다 서로 다른 입력이 있고 일부 템플릿에는 입력이 없습니다. 필요한 모든 입력을 입력한 후 다음을 클릭합니다.
5. 선택 사항: 작업에 대한 고급 설정을 구성하려면 고급 필드를 작성합니다. 고급 설정에 대한 자세한 내용은 [4.22절. “작업 마법사의 고급 설정”](#) 을 참조하십시오.
6. 다음을 클릭합니다.

7.

작업의 시간을 예약합니다.

- 작업을 즉시 실행하려면 사전 선택된 **Immediate** 실행을 유지합니다.
- 나중에 작업을 실행하려면 **향후 실행**을 선택합니다.
- 작업을 정기적으로 실행하려면 **반복 실행**을 선택합니다.

8.

선택 사항: 향후 또는 반복 실행을 선택한 경우 쿼리 유형을 선택하고, 그렇지 않으면 다음을 클릭합니다.

- 정적 쿼리는 사용자가 제공한 정확한 호스트 목록에서 작업이 실행됨을 의미합니다.
- 동적 쿼리는 작업이 실행되기 직전 호스트 목록이 평가됨을 의미합니다. 일부 필터를 기반으로 호스트 목록을 입력한 경우 해당 필터를 처음 사용한 경우 결과가 다를 수 있습니다.

쿼리 유형을 선택한 후 다음을 클릭합니다.

9.

선택 사항: 향후 또는 반복 실행을 선택한 경우 추가 세부 정보를 제공합니다.

- 향후 실행의 경우 날짜와 시간에 시작 정보를 입력합니다. 또한 날짜 및 시간 이전 시작을 선택할 수 있습니다. 해당 시간 이전에 작업을 시작할 수 없는 경우 취소됩니다.
- 반복 실행의 경우 시작 날짜 및 시간, 빈도 및 반복 실행 종료 조건을 선택합니다. 반복은 종료하거나 특정 시간에 끝나거나 지정된 반복 횟수 후에 종료할 수 있습니다. 작업 추적을 위한 용도 - 특수 레이블을 추가할 수도 있습니다. 한 번에 하나의 활성 작업만 있을 수 있습니다.

필요한 정보를 입력한 후 다음을 클릭합니다.

10.

작업 세부 정보를 검토합니다. 작업 마법사로 돌아가서 정보를 편집할 수 있는 옵션이 있습니다.

11. **Submit** (제출)을 클릭하여 실행 작업을 예약합니다.

## CLI 절차

1. **Satellite**에 다음 명령을 입력합니다.

```
# hammer settings set \
--name=remote_execution_global_proxy \
--value=false
```

2. 사용하려는 작업 템플릿의 **ID**를 찾습니다.

```
# hammer job-template list
```

3. 템플릿에 필요한 매개변수를 확인하려면 템플릿 세부 정보를 표시합니다.

```
# hammer job-template info --id My_Template_ID
```

4. 사용자 지정 매개변수를 사용하여 원격 작업을 실행합니다.

```
# hammer job-invocation create \
--inputs My_Key_1="My_Value_1",My_Key_2="My_Value_2",... \
--job-template "My_Template_Name" \
--search-query "My_Search_Query"
```

***My\_Search\_Query*** 를 호스트를 정의하는 필터 표현식으로 바꿉니다(예: "**name ~ *My\_Pattern***"). **hammer**를 사용하여 원격 명령을 실행하는 방법에 대한 자세한 내용은 **hammer job-template --help** 및 **hammer job-invocation --help** 를 입력합니다.

### 4.22. 작업 마법사의 고급 설정

일부 작업 템플릿에는 고급 설정을 입력해야 합니다. 일부 고급 설정은 특정 작업 템플릿에만 표시됩니다. 다음은 일반 고급 설정 목록입니다.

#### SSH 사용자

**SSH**를 통해 호스트에 연결하는 데 사용할 사용자입니다.

유효한 사용자

작업 실행에 사용할 사용자입니다. 기본적으로 **SSH** 사용자입니다. 설정에 따라 **SSH** 사용자 **su** 또는 **sudo**와 다른 경우 계정을 전환하는 데 사용됩니다.

- 고급 설정에서 유효 사용자를 설정하는 경우 **Ansible**은 **ansible\_become\_user** 를 입력 값으로 설정하고 **ansible\_become** 을 **true** 로 설정합니다. 즉, 매개변수가 **true**이고 **become: user: My\_User** 를 플레이북 내에서 사용하면 **Satellite**에서 덮어씁니다.
- **SSH** 사용자와 유효 사용자가 동일한 경우 **Satellite**는 **become\_user** 를 덮어쓰지 않습니다. 따라서 **Ansible** 플레이북에서 사용자 지정 **become\_user** 를 설정할 수 있습니다.

### 설명

작업에 대한 설명 템플릿입니다.

### 종료할 시간 초과

작업이 시작된 후 작업이 아직 완료되지 않은 경우 종료되어야 하는 시간(초)입니다.

### pickup 시간

클라이언트에서 선택하지 않으면 작업이 취소되는 시간(초)입니다. 이 설정은 **pull-mqtt** 전송을 사용하는 호스트에만 적용됩니다.

### 암호

**SSH** 인증 방법이 **SSH** 키 대신 암호인 경우 사용됩니다.

### 개인 키 암호

**SSH** 키가 암호로 보호되는 경우 사용됩니다.

### 유효한 사용자 암호

유효 사용자가 **ssh** 사용자와 다른 경우 사용됩니다.

### 동시성 수준

한 번에 실행되는 최대 작업 수를 정의합니다. 이렇게 하면 많은 호스트에서 작업을 실행하는 경우 시스템 리소스의 과부하를 방지할 수 있습니다.

### 실행 순서

호스트에서 작업이 실행되는 순서를 결정합니다. 알파벳순 또는 임의화할 수 있습니다.

### 4.23. 확장 CRON 라인 사용

원격 실행을 통해 **cron** 작업을 예약할 때 확장 **cron** 라인을 사용하여 작업 주기를 지정할 수 있습니다. 표준 **cron** 행에는 분, 시간, 월, 요일, 요일을 지정하는 5개의 필드가 포함되어 있습니다. 예를 들어 **0 5 \*\***는 오전 5시에 매일을 나타냅니다.

확장된 **cron** 라인은 다음과 같은 기능을 제공합니다.

**#** 을 사용하여 한 달 동안 구체적인 요일을 지정할 수 있습니다.

예를 들면 다음과 같습니다.

- **0 0 \*\* Mon#1** 은 월의 첫 번째 월요일을 지정합니다.
- **0 0 \*\* Fri#3, fri#4** 는 월의 3 번째 및 4 번째 금요일을 지정합니다.
- **0 7 \*\* Fri#-1** 은 07:00에 달의 마지막 금요일을 지정합니다.
- **0 7 \*\* Fri#L** 은 07:00에 달의 마지막 금요일을 지정합니다.
- **0 23 \*\* Mon#2 tue** 는 월의 2 번째 월요일과 매주 화요일을 23:00에 지정합니다.

**%** 를 사용하여 월의 **n-th** 날마다 지정할 수 있습니다.

예를 들면 다음과 같습니다.

- **9 0 \*\* sun%2** 는 다른 모든 일요일을 00:09로 지정합니다.
- **0 0 \* sun%2+1** 은 모든 홀수 일요일을 지정합니다.
- **9 0 \*\* Sun%2, tue%3** 은 매주 다른 일요일과 3 번째 화요일을 지정합니다.

요일이 요일과 일치하도록 지정하려면 **& amp;**를 사용할 수 있습니다.

예를 들면 다음과 같습니다.

- `0 0 30 * 1& amp;`;은 월 30일을 지정하지만 월요일인 경우에만 적용됩니다.

#### 4.24. 호스트에 대한 반복적인 ANSIBLE 작업 예약

반복 작업을 예약하여 호스트에서 **Ansible** 역할을 실행할 수 있습니다.

사전 요구 사항

- `view_foreman_tasks`, `view_job_invocations`, `view_recurring_logics` 권한이 있는지 확인합니다.

절차

1. **Satellite** 웹 UI에서 호스트 > 모든 호스트로 이동하여 원격 작업을 실행할 대상 호스트를 선택합니다.
2. **Ansible** 탭에서 작업을 선택합니다.
3. 스케줄링 반복 작업을 클릭합니다.
4. **Create New Recurring Ansible Run** (새 반복 실행 생성) 창에서 첫 번째 실행 반복 빈도, 시작 시간 및 날짜를 정의합니다.
5. **Submit** 을 클릭합니다.
6. 선택 사항: 호스트 개요에서 또는 **Ansible** > 작업으로 이동하여 예약된 **Ansible** 작업을 확인합니다.

#### 4.25. 호스트 그룹에 대한 반복적인 ANSIBLE 작업 예약

반복 작업을 예약하여 호스트 그룹에서 **Ansible** 역할을 실행할 수 있습니다.

#### 절차

1. **Satellite** 웹 UI에서 **Configure > Host Groups** 로 이동합니다.
2. **Actions** 열에서 **Ansible** 역할 실행을 예약할 호스트 그룹에 대한 **Ansible** 작업 구성 을 선택합니다.
3. 스케줄링 반복 작업을 클릭합니다.
4. **Create New Recurring Ansible Run** (새 반복 실행 생성) 창에서 첫 번째 실행 반복 빈도, 시작 시간 및 날짜를 정의합니다.
5. **Submit** 을 클릭합니다.

#### 4.26. 작업 모니터링

실행 중인 작업의 진행 상황을 모니터링할 수 있습니다. 이는 필요할 수 있는 모든 문제 해결에 도움이 될 수 있습니다.

**Ansible** 작업은 100개의 호스트 배치에서 실행되므로 특정 호스트에서 실행되는 작업을 취소할 수 없습니다. **Ansible** 플레이북이 배치의 모든 호스트에서 실행된 후에만 작업이 완료됩니다.

#### 절차

1. **Satellite** 웹 UI에서 **Monitor > Jobs** 로 이동합니다. 이 페이지는 **Execute now** 설정으로 작업을 트리거하면 자동으로 표시됩니다. 예약된 작업을 모니터링하려면 **Monitor > Jobs** 로 이동하여 검사할 작업 실행을 선택합니다.
2. 작업 페이지에서 호스트 탭을 클릭합니다. 그러면 작업이 실행 중인 호스트 목록이 표시됩니다.
3. **Host** 열에서 검사할 호스트의 이름을 클릭합니다. 그러면 작업 실행을 실시간으로 모니터링할 수 있는 명령 세부 정보 페이지가 표시됩니다.

- 언제든지 작업으로 돌아가기를 클릭하여 작업 세부 정보 페이지로 돌아갑니다.

#### CLI 절차

- 작업의 ID를 찾습니다.

```
# hammer job-invocation list
```

- 작업 출력을 모니터링합니다.

```
# hammer job-invocation output \
--host My_Host_Name \
--id My_Job_ID
```

- 선택 사항: 작업을 취소하려면 다음 명령을 입력합니다.

```
# hammer job-invocation cancel \
--id My_Job_ID
```

#### 4.27. 패키지 및 에라타 작업에 ANSIBLE 공급자 사용

기본적으로 **Satellite**는 원격 실행 작업에 스크립트 공급자 템플릿을 사용하도록 구성됩니다. 원격 작업에 **Ansible** 작업 템플릿을 사용하려면 해당 템플릿과 연결된 원격 실행 기능에 기본적으로 **Satellite**를 구성할 수 있습니다.



#### 참고

**Ansible** 작업 템플릿은 **ssh** 모드로 원격 실행이 구성된 경우에만 작동합니다.

#### 프로세스

- Satellite** 웹 UI에서 **Administer > Remote Execution Features** 로 이동합니다.
- 이름에 **by\_search** 가 포함된 각 기능을 찾습니다.



3. 이러한 기능의 작업 템플릿을 **Katello Script Default** 에서 **Katello Ansible Default** 로 변경합니다.
4. **Submit** 을 클릭합니다.

**Satellite**는 이제 패키지 및 에라타 작업을 수행할 수 있는 원격 실행 작업에 **Ansible** 공급자 템플릿을 사용합니다. 이는 **Satellite** 웹 UI의 작업 호출과 변경된 원격 실행 기능과 함께 **hammer job-invocation create** 를 사용하여 적용됩니다.

#### 4.28. CAPSULE에 작업 속도 제한 설정

성능 급증을 방지하기 위해 한 번에 **Capsule**에서 최대 활성 작업 수를 제한할 수 있습니다. 이 작업은 **Capsule**이 호스트에서 작업이 완료될 때까지 작업에 대해 먼저 호스트에 알리려고 할 때부터 활성화됩니다.

작업 속도 제한은 **mqtt** 기반 작업에만 적용됩니다.



#### 참고

최적의 활성 작업 수는 **Capsule Server**의 컴퓨팅 리소스에 따라 다릅니다. 기본적으로 활성 작업의 최대 수는 무제한입니다.

#### 프로세스

- **satellite-installer** 를 사용하여 최대 활성 작업 수를 설정합니다.

```
# satellite-installer \
--foreman-proxy-plugin-remote-execution-script-mqtt-rate-limit MAX_JOBS_NUMBER
```

예를 들면 다음과 같습니다.

```
# satellite-installer \
--foreman-proxy-plugin-remote-execution-script-mqtt-rate-limit 200
```

## 5장. RED HAT SATELLITE 및 ANSIBLE AUTOMATION CONTROLLER 통합

Red Hat Satellite 및 Ansible Automation Controller를 통합하여 Satellite 서버를 Ansible Automation Controller의 동적 인벤토리 소스로 사용할 수 있습니다. Ansible Automation Controller는 Red Hat Ansible Automation Platform의 구성 요소입니다.

프로비저닝 콜백 기능을 사용하여 호스트 또는 Ansible Automation Controller에서 Satellite에서 관리하는 호스트에서 플레이북을 실행할 수도 있습니다. Satellite Server에서 새 호스트를 프로비저닝할 때 프로비저닝 콜백 기능을 사용하여 Ansible Automation Controller에서 플레이북 실행을 트리거할 수 있습니다. 플레이북은 프로비저닝 프로세스 후 호스트를 구성합니다.

### 5.1. 동적 인벤토리 항목으로 ANSIBLE AUTOMATION CONTROLLER에 SATELLITE SERVER 추가

Ansible Automation Controller에 Satellite Server를 동적 인벤토리 항목으로 추가하려면 Ansible Automation Controller에서 Satellite Server 사용자에게 대한 인증 정보를 생성하고, Ansible Automation Controller 사용자를 인증 정보에 추가한 다음 인벤토리 소스를 구성해야 합니다.

#### 사전 요구 사항

- 예를 들어, 수만 개의 호스트를 관리하는 것과 같이 Satellite 배포가 대규모인 경우 관리자가 아닌 사용자를 사용하면 권한 부여 확인 중에 누적되는 시간에 영향을 미치기 때문에 성능에 부정적인 영향을 미칠 수 있습니다. 대규모 배포의 경우 admin 사용자를 사용하는 것이 좋습니다.
- 관리자가 아닌 사용자의 경우 Ansible Tower 인벤토리 리더 역할을 Satellite Server 사용자에게 할당해야 합니다. 사용자, 역할 및 권한 필터 관리에 대한 자세한 내용은 [Red Hat Satellite 관리에서 역할 생성 및 관리](#)를 참조하십시오.
- 동일한 네트워크 또는 서브넷에서 Satellite Server 및 Ansible Automation Controller를 호스팅해야 합니다.

#### 프로세스

1. Ansible Automation Controller 웹 UI에서 Satellite에 대한 인증 정보를 생성합니다. 인증 정보 생성에 대한 자세한 내용은 [자동화 컨트롤러 사용자 가이드에서 새 인증 정보 및 Red Hat Satellite 인증 정보 추가](#)를 참조하십시오.

표 5.1. Satellite 인증 정보

인증 정보 유형:	Red Hat Satellite 6
Satellite URL:	https://satellite.example.com
사용자 이름:	Satellite 사용자의 사용자 이름과 통합 역할입니다.
암호:	Satellite 사용자의 암호입니다.

2.

**Ansible Automation Controller** 사용자를 새 인증 정보에 추가합니다. 인증 정보에 사용자를 추가하는 방법에 대한 자세한 내용은 *자동화 컨트롤러 사용자 가이드*의 [인증 정보 시작하기](#)를 참조하십시오.

3.

새 인벤토리를 추가합니다. 자세한 내용은 *자동화 컨트롤러 사용자 가이드*에서 [새 인벤토리 추가](#)를 참조하십시오.

4.

새 인벤토리에서 다음 인벤토리 소스 옵션을 지정하여 **Satellite Server**를 인벤토리 소스로 추가합니다. 자세한 내용은 *자동화 컨트롤러 사용자 가이드*의 [소스 추가](#)를 참조하십시오.

표 5.2. 인벤토리 소스 옵션

소스	Red Hat Satellite 6
인증 정보	Satellite Server에 대해 생성하는 인증 정보입니다.
덮어쓰기	선택 사항
변수 덮어쓰기	선택 사항
시작 시 업데이트	선택 사항
캐시 제한 시간	90

5.

추가한 소스를 동기화해야 합니다.

## 5.2. 호스트의 프로비저닝 콜백 구성

**Satellite**에서 호스트를 생성할 때 **Ansible Automation Controller**를 사용하여 플레이북을 실행하여 새로 생성된 호스트를 구성할 수 있습니다. 이를 **Ansible Automation Controller**에서 **프로비저닝 콜백**이라고 합니다.

프로비저닝 콜백 기능은 **Ansible Automation Controller**에서 프로비저닝 프로세스의 일부로 실행되는 플레이북을 트리거합니다. 플레이북은 프로비저닝 프로세스 후 호스트를 구성합니다.

프로비저닝 콜백에 대한 자세한 내용은 *자동화 컨트롤러 사용자 가이드*의 [콜백 프로비저닝](#)을 참조하십시오.

**Satellite Server**에서 **Kickstart** 기본값 및 **Kickstart** 기본 완료 템플릿에는 다음 세 가지 스니펫이 포함되어 있습니다.

1. `ansible_provisioning_callback`
2. `ansible_tower_callback_script`
3. `ansible_tower_callback_service`

호스트 또는 호스트 그룹에 매개 변수를 추가하여 이러한 스니펫에서 새로 생성된 호스트에서 **Ansible** 플레이북을 실행하는 데 사용할 수 있는 자격 증명을 제공할 수 있습니다.

#### 사전 요구 사항

프로비저닝 콜백을 구성하려면 먼저 **Satellite**를 **Ansible Automation Controller**에서 동적 인벤토리로 추가해야 합니다. 자세한 내용은 [Satellite 및 Ansible Automation Controller 통합](#)에서 참조하십시오.

**Ansible Automation Controller** 웹 UI에서 다음 작업을 완료해야 합니다.

1. 새 호스트에 대한 시스템 자격 증명을 생성합니다. **Satellite**에서 생성한 호스트에 할당할 자격 증명에 동일한 암호를 입력해야 합니다. 자세한 내용은 *자동화 컨트롤러 사용자 가이드*에서 [새 인증 정보 추가](#)를 참조하십시오.
2. 프로젝트를 생성합니다. 자세한 내용은 *Ansible Automation Controller 사용자 가이드*의 [프로젝트를 참조하십시오](#).
3. 프로젝트에 작업 템플릿을 추가합니다. 자세한 내용은 *자동화 컨트롤러 사용자 가이드*의 [작업](#)

템플릿을 참조하십시오.

4. 작업 템플릿에서 프로비저닝 콜백을 활성화하고, 호스트 구성 키를 생성하고, 작업 템플릿의 **template\_ID**를 기록해야 합니다. 작업 템플릿에 대한 자세한 내용은 *자동화 컨트롤러 사용자 가이드*의 **작업** 템플릿을 참조하십시오.

#### 프로세스

1. **Satellite** 웹 UI에서 **Configure > Host Group** 으로 이동합니다.
2. 호스트 그룹을 생성하거나 기존 호스트 그룹을 편집합니다.
3. **Host Group** 창에서 **Parameters** 탭을 클릭합니다.
4. 매개 변수 추가를 클릭합니다.
5. 각 새 매개변수에 대해 다음 정보를 입력합니다.

표 5.3. 호스트 매개변수

이름	현재의	설명
<b>ansible_tower_provisioning</b>	true	프로비저닝 콜백을 활성화합니다.
<b>ansible_tower_fqdn</b>	<i>controller.example.com</i>	Ansible Automation Controller의 FQDN(정규화된 도메인 이름)입니다. Satellite에 의해 추가되므로 <b>https</b> 를 추가하지 마십시오.
<b>ansible_job_template_id</b>	<i>template_ID</i>	템플릿의 URL에서 찾을 수 있는 프로비저닝 템플릿의 ID: <b>/templates/job_template/5</b> .
<b>ansible_host_config_key</b>	<i>config_KEY</i>	작업 템플릿이 Ansible Automation Controller에서 생성하는 호스트 구성 키입니다.

6. **Submit** 을 클릭합니다.

7. 호스트 그룹을 사용하여 호스트를 생성합니다.

8. 새 호스트에서 다음 명령을 입력하여 **ansible-callback** 서비스를 시작합니다.

```
# systemctl start ansible-callback
```

9. 새 호스트에서 다음 명령을 입력하여 **ansible-callback** 서비스의 상태를 출력합니다.

```
# systemctl status ansible-callback
```

명령에서 다음 출력을 반환하는 경우 프로비저닝 콜백이 올바르게 구성됩니다.

```
SAT_host systemd[1]: Started Provisioning callback to Ansible Automation Controller...
```

#### 수동 프로비저닝 콜백

- 프로비저닝 콜백 **URL** 및 호스트의 호스트 구성 키를 사용하여 **Ansible Automation Controller**를 호출할 수 있습니다.

```
# curl -k -s --data curl --insecure --data host_config_key=my_config_key\  
https://controller.example.com/api/v2/job_templates/8/callback/
```

프로비저닝 콜백 **URL**을 입력할 때 **https** 를 사용해야 합니다.

이렇게 하면 호스트에 대해 템플릿에 지정된 플레이북이 트리거됩니다.

## 6장. 작업 템플릿 예 및 확장

이 섹션을 참조하여 요구 사항에 맞게 작업 템플릿을 수정, 사용자 정의 및 확장할 수 있습니다.

### 6.1. 작업 템플릿 사용자 정의

작업 템플릿을 생성할 때 템플릿 편집기 필드에 기존 템플릿을 포함할 수 있습니다. 이렇게 하면 템플릿을 결합하거나 일반 템플릿에서 더 구체적인 템플릿을 생성할 수 있습니다.

다음 템플릿은 기본 템플릿을 결합하여 클라이언트에서 **nginx** 서비스를 설치하고 시작합니다.

```
<%= render_template 'Package Action - SSH Default', :action => 'install', :package => 'nginx' %>
<%= render_template 'Service Action - SSH Default', :action => 'start', :service_name => 'nginx' %>
```

위의 템플릿은 렌더링된 템플릿의 매개변수 값을 직접 지정합니다. 또한 **input()** 메서드를 사용하여 사용자가 작업 실행 시 렌더링된 템플릿에 대한 입력을 정의할 수 있습니다. 예를 들어 다음 구문을 사용할 수 있습니다.

```
<%= render_template 'Package Action - SSH Default', :action => 'install', :package => input("package") %>
```

위의 템플릿을 사용하면 렌더링된 템플릿에서 매개변수 정의를 가져와야 합니다. 이렇게 하려면 작업 탭으로 이동하여 **Add Foreign Input Set** 을 클릭하고 **Target** 템플릿 목록에서 렌더링된 템플릿을 선택합니다. 모든 매개변수를 가져오거나 쉼표로 구분된 목록을 지정할 수 있습니다.

### 6.2. 기본 작업 템플릿 카테고리

작업 템플릿 카테고리	설명
패키지	패키지 관련 작업을 수행하기 위한 템플릿입니다. 설치, 업데이트 및 제거 작업이 기본적으로 포함됩니다.
Puppet	대상 호스트에서 Puppet 실행을 위한 템플릿입니다.
power	전원 관련 작업을 수행하기 위한 템플릿입니다. 재시작 및 종료 작업은 기본적으로 포함됩니다.
명령	원격 호스트에서 사용자 지정 명령을 실행하기 위한 템플릿입니다.

작업 템플릿 카테고리	설명
서비스	서비스 관련 작업을 수행하기 위한 템플릿입니다. 기본적으로 start, stop, restart, status 작업이 포함됩니다.
katello	콘텐츠 관련 작업을 수행하기 위한 템플릿입니다. 이러한 템플릿은 주로 Satellite 웹 UI의 여러 부분에서 사용되지만(예: 콘텐츠 호스트에 대한 대량 작업 UI) 에라타 설치와 같은 작업을 수행하는 데 별도로 사용할 수 있습니다.

### 6.3. RESTORECON 템플릿 예

이 예제에서는 대상 호스트에서 선택한 디렉터리에 있는 모든 파일에 대한 기본 SELinux 컨텍스트를 복원하는 **Run Command - restorecon** 이라는 템플릿을 생성하는 방법을 보여줍니다.

#### 프로세스

1. **Satellite 웹 UI**에서 **호스트 > 템플릿 > 작업 템플릿** 으로 이동합니다.
2. 새 작업 템플릿을 클릭합니다.
3. 이름 필드에 **Run Command - restorecon** 을 입력합니다. **Default** 를 선택하여 모든 조직에서 템플릿을 사용할 수 있도록 합니다. 템플릿 편집기에 다음 텍스트를 추가합니다.

```
restorecon -RvF <%= input("directory") %>
```

& lt;%= input("directory") % > 문자열은 작업을 호출하는 동안 사용자 정의 디렉터리로 교체됩니다.

4. **Job** 탭에서 **Job category** 를 **Commands** 로 설정합니다.
5. **Add Input (입력 추가)**을 클릭하여 작업 사용자 지정 을 허용합니다. 이름 필드에 디렉토리를 입력합니다. 입력 이름은 템플릿 편집기에 지정된 값과 일치해야 합니다.
6. 사용자가 지정한 매개 변수 없이 명령을 실행할 수 없도록 **Required** 를 클릭합니다.
- 7.



입력 유형 목록에서 사용자 입력을 선택합니다. 작업 호출 중에 표시할 설명을 입력합니다(예: `restorecon`의 대상 디렉터리).

8.

**Submit** 을 클릭합니다. 자세한 내용은 [호스트 관리의 여러 호스트에서 `restorecon` 템플릿 실행을 참조하십시오.](#)

#### 6.4. RESTORECON 템플릿 렌더링

이 예제에서는 예제 `restorecon` 템플릿에서 생성된 `Run command - restorecon` 템플릿에서 파생된 템플릿을 생성하는 방법을 보여줍니다.

[https://docs.redhat.com/en/documentation/red\\_hat\\_satellite/6.15/html-single/managing\\_hosts/index#Example\\_restorecon\\_Template\\_managing-hosts](https://docs.redhat.com/en/documentation/red_hat_satellite/6.15/html-single/managing_hosts/index#Example_restorecon_Template_managing-hosts) 이 템플릿은 작업 실행에 대한 사용자 입력이 필요하지 않으며 대상 호스트의 `/home/` 디렉터리에 있는 모든 파일에서 SELinux 컨텍스트를 복원합니다.

작업 템플릿 설정에 설명된 대로 새 템플릿을 생성하고 템플릿 편집기에서 다음 문자열을 지정합니다.

```
<%= render_template("Run Command - restorecon", :directory => "/home") %>
```

#### 6.5. 여러 호스트에서 RESTORECON 템플릿 실행

이 예에서는 여러 호스트의 [Example restorecon](#) 템플릿에서 생성된 템플릿을 기반으로 작업을 실행하는 방법을 보여줍니다. 이 작업은 `/home/` 디렉터리의 모든 파일에서 SELinux 컨텍스트를 복원합니다.

##### 프로세스

1.

Satellite 웹 UI에서 **Monitor > Jobs** 로 이동하여 **Run job** 을 클릭합니다.

2.

**Job category** (작업 범주)로 **Commands** 를 선택하고 **Run Command - restorecon as Job template** 을 선택하고 **Next** 를 클릭합니다.

3.

작업을 실행할 호스트를 선택합니다. 호스트를 선택하지 않으면 현재 컨텍스트에서 볼 수 있는 모든 호스트에서 작업이 실행됩니다.

4.

디렉터리 필드에 `/home` 과 같은 디렉토리를 제공하고 **Next** 를 클릭합니다.

5.

선택 사항: 작업에 대한 고급 설정을 구성하려면 고급 필드를 작성합니다. 고급 설정에 대한 자세한 내용은 [4.22절](#). “[작업 마법사의 고급 설정](#)” 을 참조하십시오. 고급 설정 입력을 완료했거나 필수가 아닌 경우 **Next** 를 클릭합니다.

6.

작업의 시간을 예약합니다.

- 작업을 즉시 실행하려면 사전 선택된 **Immediate** 실행을 유지합니다.
- 나중에 작업을 실행하려면 향후 실행을 선택합니다.
- 작업을 정기적으로 실행하려면 반복 실행을 선택합니다.

7.

선택 사항: 향후 또는 반복 실행을 선택한 경우 쿼리 유형을 선택하고, 그렇지 않으면 다음을 클릭합니다.

- 정적 쿼리는 사용자가 제공한 정확한 호스트 목록에서 작업이 실행됨을 의미합니다.
- 동적 쿼리는 작업이 실행되기 직전 호스트 목록이 평가됨을 의미합니다. 일부 필터를 기반으로 호스트 목록을 입력한 경우 해당 필터를 처음 사용한 경우 결과가 다를 수 있습니다.

쿼리 유형을 선택한 후 다음을 클릭합니다.

8.

선택 사항: 향후 또는 반복 실행을 선택한 경우 추가 세부 정보를 제공합니다.

- 향후 실행의 경우 날짜와 시간에 시작 정보를 입력합니다. 또한 날짜 및 시간 이전 시작을 선택할 수 있습니다. 해당 시간 이전에 작업을 시작할 수 없는 경우 취소됩니다.
- 반복 실행의 경우 반복 실행의 시작 날짜 및 시간, 빈도 및 조건을 선택합니다. 반복은 종료하거나 특정 시간에 끝나거나 지정된 반복 횟수 후에 종료할 수 있습니다. 작업 추적을 위한 용도 - 특수 레이블을 추가할 수도 있습니다. 한 번에 하나의 활성 작업만 있을 수 있습니다.

필요한 정보를 입력한 후 다음을 클릭합니다.

9.

작업 세부 정보를 검토합니다. 작업 마법사로 돌아가서 정보를 편집할 수 있는 옵션이 있습니다.

10.

**Submit (제출)**을 클릭하여 실행 작업을 예약합니다.

## 6.6. 템플릿에 전원 작업 포함

이 예에서는 재부팅과 같은 전원 작업을 수행하기 위한 작업 템플릿을 설정하는 방법을 보여줍니다. 이 절차에서는 재부팅 시 오류가 발생하면 **Satellite**에서 연결 해제된 예외를 해석하지 않으므로 작업의 원격 실행이 올바르게 작동합니다.

작업 템플릿 설정에 설명된 대로 새 템플릿을 생성하고 템플릿 편집기에서 다음 문자열을 지정합니다.

```
<%= render_template("Power Action - SSH Default", :action => "restart") %>
```