



# Red Hat Ansible Automation Platform 2.4

## Ansible 플레이북 시작하기

Ansible Playbook 시작하기



# Red Hat Ansible Automation Platform 2.4 Ansible 플레이북 시작하기

---

Ansible Playbook 시작하기

## 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 플레이북을 생성하고 사용하여 자동화 요구 사항을 해결하는 방법을 보여줍니다. 이 문서에는 업스트림 [docs.ansible.com](https://docs.ansible.com) 문서의 내용이 포함되어 있으며 이는 GNU GENERAL PUBLIC LICENSE v3.0에서 다룹니다.

## 차례

머리말 .....	3
RED HAT 문서에 관한 피드백 제공 .....	4
<b>1장. 소개</b> .....	<b>5</b>
1.1. ANSIBLE 플레이북은 어떻게 작동합니까?	5
1.2. ANSIBLE PLAYBOOK을 사용하려면 어떻게 해야 합니까?	6
1.3. ANSIBLE로 자동화 시작	7
1.4. 인벤토리 빌드	7
1.5. 변수 생성	10
1.6. 첫 번째 플레이북 생성	10
<b>2장. PLAYBOOK을 사용하여 관리 노드에 대한 연결 설정</b> .....	<b>12</b>
2.1. 네트워크 ANSIBLE 명령 실행	12
2.2. 네트워크 ANSIBLE PLAYBOOK 실행	12
2.3. 네트워크 장치에서 팩트 수집	16
<b>3장. ANSIBLE 플레이북의 실용적인 예</b> .....	<b>18</b>
3.1. 플레이북 실행	18
<b>4장. 오픈 소스 라이선스</b> .....	<b>20</b>
preamble	20
이용 약관	20
이 약관을 새 프로그램에 적용하는 방법	27



## 머리말

Red Hat Ansible Automation Platform에 관심을 가져 주셔서 감사합니다. Ansible Automation Platform은 Ansible 기반 환경에 제어, 지식, 위임을 추가하여 팀이 복잡한 다중 계층 배포를 관리하는 데 도움이 되는 상용 서비스입니다.

이 가이드에서는 Ansible 플레이북 사용에 대한 소개를 제공합니다.

## RED HAT 문서에 관한 피드백 제공

이 문서를 개선하기 위한 제안이 있거나 오류를 찾을 수 있는 경우 <https://access.redhat.com> 에서 기술 지원에 문의하여 **docs-product** 구성 요소를 사용하여 Ansible Automation Platform Jira 프로젝트에 문제를 생성하십시오.



# 1장. 소개

Ansible 플레이북은 자동화 작업을 위한 블루프린트로, 솔루션 인벤토리에서 제한된 수동 노력으로 실행되는 작업입니다. 플레이북은 Ansible에서 어떤 장치에서 수행할 작업을 지시합니다. IT 환경에서 수백 또는 수천 개의 유사한 기술에 동일한 작업을 수동으로 적용하는 대신 플레이북을 실행하면 지정된 유형의 인벤토리(예: 라우터)에 대해 동일한 작업이 자동으로 완료됩니다.

플레이북은 운영 체제 및 Kubernetes 플랫폼(네트워크, 보안 시스템, GitHub와 같은 코드 리포지토리)과 같은 IT 인프라를 자동화하는 데 정기적으로 사용됩니다. 플레이북을 사용하여 모든 항목을 처음부터 생성하는 수동 오버헤드 없이 애플리케이션, 서비스, 서버 노드 및 기타 장치를 프로그래밍할 수 있습니다. 플레이북과 그 내의 조건, 변수 및 작업을 무기한 저장, 공유 또는 재사용할 수 있습니다. 이를 통해 운영 지식을 조정하고 동일한 작업을 일관되게 수행할 수 있습니다.

## 1.1. ANSIBLE 플레이북은 어떻게 작동합니까?

Ansible 플레이북은 지정된 인벤토리 또는 호스트 그룹에 대해 자동으로 실행되는 작업 목록입니다. 하나 이상의 Ansible 작업을 결합하여 플레이(즉, 특정 호스트에 매핑된 정렬된 작업 그룹)를 만들 수 있습니다.

작업은 작성된 순서대로 실행됩니다.

플레이북에는 하나 이상의 플레이가 포함될 수 있습니다.

플레이북은 정렬된 목록에 있는 하나 이상의 **플레이**로 구성됩니다.

**플레이북**과 **플레이**라는 용어는 활동적인 유사성입니다.

각 플레이는 하나 이상의 작업을 실행하여 플레이북의 전체 목표의 일부를 실행합니다.

각 작업은 Ansible 모듈을 호출합니다.

### Playbook

전체 목표를 달성하기 위해 Ansible에서 작업을 수행하는 순서를 정의하는 플레이 목록입니다.

### play

인벤토리의 관리 노드에 매핑되는 정렬된 작업 목록입니다.

### Task

Ansible에서 수행하는 작업을 정의하는 단일 모듈에 대한 참조입니다.

### 역할

역할은 기능을 "조리"에 배치하여 필요에 따라 플레이북에서 코드를 사용할 수 있는 플레이북 코드를 재사용 가능하게 만드는 방법입니다.

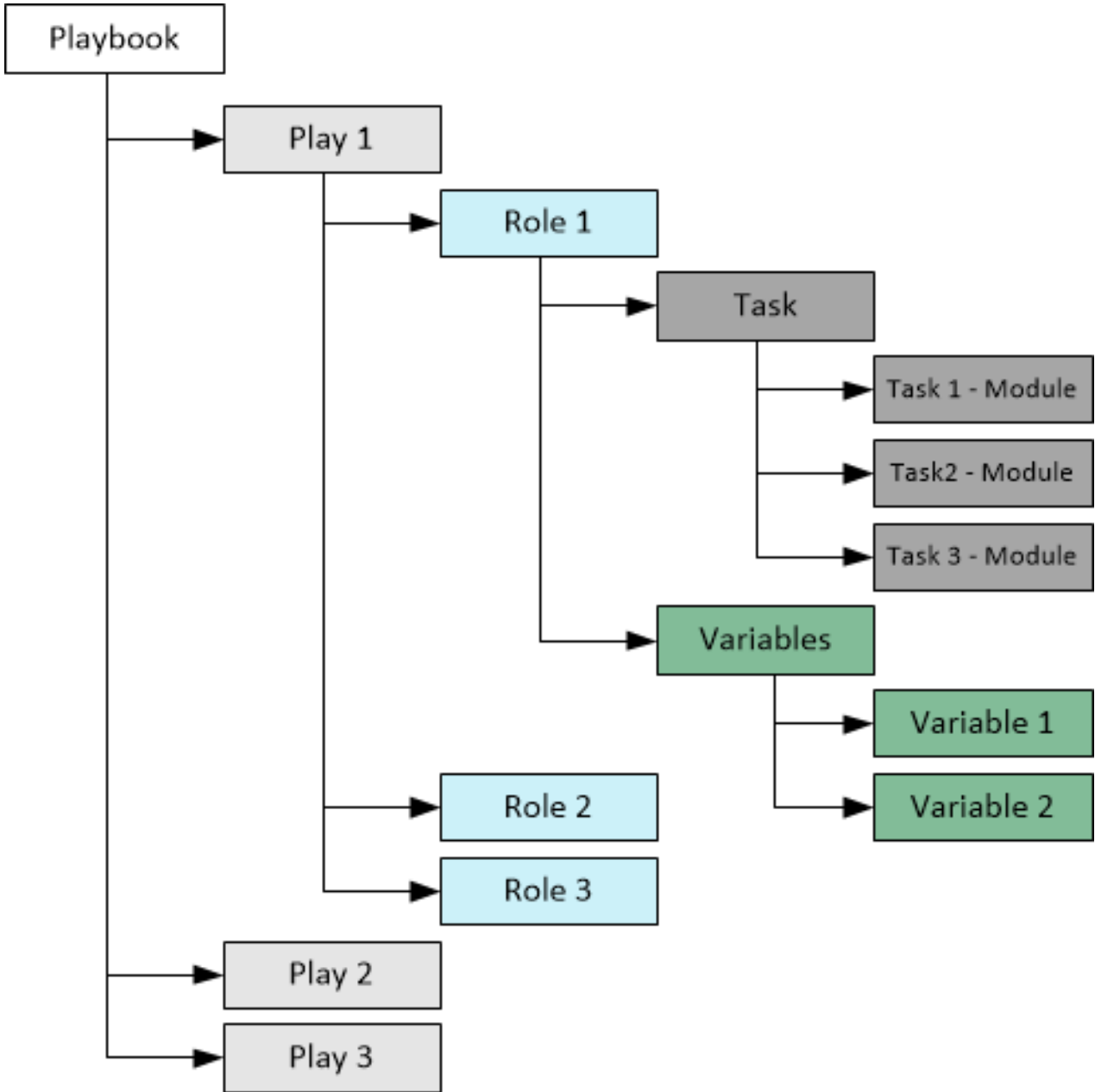
### module

Ansible이 관리 노드에서 실행되는 코드 또는 바이너리 단위입니다.

Ansible 모듈은 각 모듈에 대해 *정규화된 컬렉션 이름* (FQCN)을 사용하여 컬렉션으로 그룹화됩니다. 작업은 모듈에서 실행되며, 각 작업은 플레이북에서 특정 작업을 수행합니다. 모듈에는 작업이 실행되는 시기와 위치를 결정하는 메타데이터와 이를 실행하는 사용자가 포함됩니다. 다음과 같은 모든 종류의 IT 작업을 수행하는 수천 개의 Ansible 모듈이 있습니다.

- 클라우드 관리
- 사용자 관리
- 네트워킹

- 보안
- 구성 관리
- 통신



## 1.2. ANSIBLE PLAYBOOK을 사용하려면 어떻게 해야 합니까?

Ansible은 YAML 구문을 사용합니다. YAML은 복잡한 코딩 언어를 배울 필요 없이 플레이북을 생성할 수 있는 사람이 읽을 수 있는 언어입니다.

YAML에 대한 자세한 내용은 YAML 구문을 참조하고 텍스트 편집기에 대한 애드온을 설치하는 것이 좋습니다. 기타 툴 및 programs을 참조하여 플레이북에 깔끔한 YAML 구문을 작성하는 데 도움이 됩니다.

Ansible 플레이북을 사용하는 방법은 다음 두 가지가 있습니다.

- CLI( 명령줄 인터페이스 )에서
- Red Hat Ansible Automation Platform의 푸시 버튼 배포 사용.

### 1.2.1. CLI에서

를 사용하여 오픈 소스 Ansible 프로젝트 또는 Red Hat Ansible Automation Platform을 설치한 후

```
$ sudo dnf install ansible
```

Red Hat Enterprise Linux CLI에서는 **ansible-playbook** 명령을 사용하여 Ansible 플레이북을 실행할 수 있습니다.

### 1.2.2. 플랫폼 내에서

Red Hat Ansible Automation Platform 사용자 인터페이스는 대규모 작업 또는 작업 템플릿의 일부로 사용할 수 있는 푸시 버튼 Ansible 플레이북 배포를 제공합니다. 이러한 배포에는 IT 자동화 최신 사용자 또는 CLI에서 많은 작업이 없는 사용자에게 도움이 되는 추가 보호 기능이 제공됩니다.

## 1.3. ANSIBLE로 자동화 시작

자동화 프로젝트를 생성하고, 인벤토리를 빌드하고, **Hello World** 플레이북을 생성하여 Ansible을 시작합니다.

#### 사전 요구 사항

- Ansible 패키지가 설치되어 있어야 합니다.

#### 프로세스

- 파일 시스템에 프로젝트 폴더를 생성합니다.

```
mkdir ansible_quickstart
cd ansible_quickstart
```

단일 디렉터리 구조를 사용하면 소스 제어에 쉽게 추가하고 자동화 콘텐츠를 재사용 및 공유할 수 있습니다.

## 1.4. 인벤토리 빌드

인벤토리는 Ansible에 시스템 정보 및 네트워크 위치를 제공하는 중앙 집중식 파일로 관리형 노드를 구성합니다. Ansible은 인벤토리 파일을 사용하여 단일 명령으로 많은 수의 호스트를 관리할 수 있습니다. 다음 단계를 완료하려면 하나 이상의 호스트 시스템의 IP 주소 또는 FQDN(정규화된 도메인 이름)이 필요합니다. 설명을 위해 호스트는 컨테이너 또는 가상 시스템에서 로컬로 실행될 수 있습니다.

또한 공용 SSH 키가 각 호스트의 **authorized\_keys** 파일에 추가되었는지 확인해야 합니다. 다음 절차에 따라 인벤토리를 빌드합니다.

#### 프로세스

생성한 **ansible\_quickstart** 디렉터리에 **inventory.ini** 라는 파일을 생성합니다. 새 **[myhosts]** 그룹을 **inventory.ini** 파일에 추가하고 각 호스트 시스템의 IP 주소 또는 FQDN(정규화된 도메인 이름)을 지정합니다.

```
[myhosts]
192.0.2.50
192.0.2.51
```

### 192.0.2.52

다음을 사용하여 인벤토리를 확인합니다.

```
ansible-inventory -i inventory.ini --list
```

다음을 사용하여 인벤토리의 **myhosts** 그룹을 ping합니다.

```
ansible myhosts -m ping -i inventory.ini
```

사용자 이름이 제어 노드와 관리 노드에서 다른 경우 Ansible 명령으로 **-u** 옵션을 전달합니다.

```

192.0.2.50 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
192.0.2.51 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
192.0.2.52 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}

```

인벤토리를 성공적으로 구축했습니다.

#### 1.4.1. INI 또는 YAML 형식의 인벤토리

INI 파일 또는 YAML에서 인벤토리를 생성할 수 있습니다. 위의 예와 같은 대부분의 경우 INI 파일은 적은 수의 관리형 노드에서 쉽게 읽을 수 있습니다. YAML 형식으로 인벤토리를 생성하면 관리 노드 수가 증가함에 따라 합리적인 옵션이 됩니다.

예를 들어 다음은 관리 노드의 고유 이름을 선언하고 **ansible\_host** 필드를 사용하는 **inventory.ini** 와 동일합니다.

```

myhosts:
  hosts:
    my_host_01:
      ansible_host: 192.0.2.50
    my_host_02:
      ansible_host: 192.0.2.51
    my_host_03:
      ansible_host: 192.0.2.52

```

### 1.4.2. 인벤토리 빌드 팁

- 그룹 이름이 의미 있고 고유해야 합니다.
- 그룹 이름도 대소문자를 구분합니다.
- 그룹 이름에 공백, 하이픈 또는 이전 숫자(19th\_floor가 아닌 floor\_19 사용)를 사용하지 마십시오.
- What, Where, When 에 따라 인벤토리의 호스트를 논리적으로 그룹화합니다.
  - what: 토폴로지에 따른 그룹 호스트(예: db, web, leaf, spine)입니다.
  - 여기서: 지리적 위치별 그룹 호스트(예: 데이터 센터, 지역, 현장, 빌드)입니다.
  - When: Group hosts by stage (예: development, test, staging, production)

### 1.4.3. 메타 그룹 사용

다음 구문을 사용하여 인벤토리에서 여러 그룹을 구성하는 메타 그룹을 생성합니다.

```
metagroupname:
  children:
```

다음 인벤토리는 데이터 센터의 기본 구조를 보여줍니다. 이 예제 인벤토리에는 모든 네트워크 장치와 네트워크 그룹 및 모든 Cryostat를 포함하는 데이터 센터 메타 그룹을 포함하는 네트워크 메타 그룹이 포함되어 있습니다.

```
leafs:
  hosts:
    leaf01:
      ansible_host: 192.0.2.100
    leaf02:
      ansible_host: 192.0.2.110

spines:
  hosts:
    spine01:
      ansible_host: 192.0.2.120
    spine02:
      ansible_host: 192.0.2.130

network:
  children:
    leafs:
    spines:

webservers:
  hosts:
    webserver01:
      ansible_host: 192.0.2.140
    webserver02:
      ansible_host: 192.0.2.150

datacenter:
```

```
children:
  network:
  webservers:
```

## 1.5. 변수 생성

변수는 IP 주소, FQDN, 운영 체제, SSH 사용자와 같은 관리 노드의 값을 설정하므로 Ansible 명령을 실행할 때 이를 전달할 필요가 없습니다.

변수는 특정 호스트에 적용할 수 있습니다.

```
webservers:
  hosts:
    webserver01:
      ansible_host: 192.0.2.140
      http_port: 80
    webserver02:
      ansible_host: 192.0.2.150
      http_port: 443
```

변수는 그룹의 모든 호스트에도 적용할 수 있습니다.

```
webservers:
  hosts:
    webserver01:
      ansible_host: 192.0.2.140
      http_port: 80
    webserver02:
      ansible_host: 192.0.2.150
      http_port: 443
  vars:
    ansible_user: my_server_user
```

인벤토리 및 Ansible 인벤토리 변수에 대한 자세한 내용은 [설치 프로그램 인벤토리 파일 및 인벤토리 파일 변수](#) 정보를 참조하십시오.

## 1.6. 첫 번째 플레이북 생성

다음 절차에 따라 호스트를 ping하고 "Hello world" 메시지를 출력하는 플레이북을 생성합니다.

### 프로세스

1. 다음 콘텐츠를 사용하여 **ansible\_quickstart** 디렉터리에 **playbook.yaml** 이라는 파일을 생성합니다.

```
- name: My first play
  hosts: myhosts
  tasks:
    - name: Ping my hosts
      ansible.builtin.ping:
    - name: Print message
      ansible.builtin.debug:
        msg: Hello world
```

- 2. 다음 명령을 사용하여 플레이북을 실행합니다.

```
ansible-playbook -i inventory.ini playbook.yaml
```

- 3. Ansible은 다음 출력을 반환합니다.

```
PLAY [My first play] *****
TASK [Gathering Facts] *****
ok: [192.0.2.50]
ok: [192.0.2.51]
ok: [192.0.2.52]

TASK [Ping my hosts] *****
ok: [192.0.2.50]
ok: [192.0.2.51]
ok: [192.0.2.52]

TASK [Print message] *****
ok: [192.0.2.50] => {
  "msg": "Hello world"
}
ok: [192.0.2.51] => {
  "msg": "Hello world"
}
ok: [192.0.2.52] => {
  "msg": "Hello world"
}

PLAY RECAP *****
192.0.2.50: ok=3  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
192.0.2.51: ok=3  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
192.0.2.52: ok=3  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

이 출력에서는 다음을 확인할 수 있습니다.

- 플레이 및 각 작업을 지정하는 이름입니다. 항상 플레이북을 쉽게 확인하고 문제를 해결할 수 있는 설명이 포함된 이름을 사용합니다.
- Gather Facts 작업은 암시적으로 실행됩니다. 기본적으로 Ansible은 플레이북에서 사용할 수 있는 인벤토리에 대한 정보를 수집합니다.
- 각 작업의 상태입니다. 각 작업의 상태가 **ok** 이므로 성공적으로 실행됨을 의미합니다.
- 호스트당 플레이북의 모든 작업의 결과를 요약하는 play recap. 이 예제에는 3개의 작업이 있으므로 **ok=3** 은 각 작업이 성공적으로 실행되었음을 나타냅니다.

## 2장. PLAYBOOK을 사용하여 관리 노드에 대한 연결 설정

인증 정보를 확인하려면 네트워크 장치에 수동으로 연결하고 구성을 검색할 수 있습니다. 샘플 사용자 및 장치 이름을 실제 인증 정보로 교체합니다.

예를 들어 VyOS 라우터의 경우 다음을 수행합니다.

```
ssh my_vyos_user@vyos.example.net
show config
exit
```

### 2.1. 네트워크 ANSIBLE 명령 실행

네트워크 장치에서 수동으로 명령을 연결하고 실행하는 대신 단일 Ansible 명령을 사용하여 구성을 검색할 수 있습니다.

```
ansible all -i vyos.example.net, -c ansible.netcommon.network_cli -u \
my_vyos_user -k -m vyos.vyos.vyos_facts -e \
ansible_network_os=vyos.vyos.vyos
```

이 명령의 플래그는 7개의 값을 설정합니다.

- 명령이 적용해야 하는 호스트 그룹(이 경우 모두)
- 인벤토리(-i, 대상 장치 또는 장치 - 후행 쉼표 -i 가 인벤토리 파일을 가리키지 않음)
- 연결 방법(-c, ansible 연결 및 실행을 위한 방법)
- 사용자(-u, SSH 연결의 사용자 이름)
- SSH 연결 방법(-k, 암호 요청)
- 정규화된 컬렉션 이름(FQCN)을 사용하여 실행할 모듈(-m)
- 추가 변수(-e, 이 경우 네트워크 OS 값 설정)



#### 참고

ssh 키와 함께 **ssh-agent** 를 사용하는 경우 Ansible은 자동으로 로드합니다. **-k** 플래그를 생략할 수 있습니다.

가상 환경에서 Ansible을 실행하는 경우 **ansible\_python\_interpreter=/path/to/venv/bin/python** 변수도 추가해야 합니다.

### 2.2. 네트워크 ANSIBLE PLAYBOOK 실행

매일 특정 명령을 실행하려면 ansible 대신 **ansible-playbook** 을 사용하여 플레이북에 저장하고 실행할 수 있습니다. 플레이북은 명령줄에 플래그와 함께 제공한 많은 매개변수를 저장할 수 있으므로 명령줄에 입력하는 것은 적습니다. 플레이북과 인벤토리 파일이라는 두 개의 파일이 필요합니다.

#### 사전 요구 사항

여기에서 [first\\_playbook.yml](#) 을 다운로드합니다.



플레이북은 다음과 같습니다.

```
---
- name: Network Getting Started First Playbook
  connection: ansible.netcommon.network_cli
  gather_facts: false
  hosts: all
  tasks:

  - name: Get config for VyOS devices
    vyos.vyos.vyos_facts:
      gather_subset: all

  - name: Display the config
    debug:
      msg: "The hostname is {{ ansible_net_hostname }} and the OS is {{ ansible_net_version }}"
```

레이블	설명
<b>gather_facts</b>	Ansible의 기본 팩트 수집( <b>ansible.builtin.setup</b> )은 이 네트워킹 컬렉션의 플랫폼별 모듈( <b>vyos.vyos.vyos_facts</b> )에서 제공하는 팩트를 사용하기 때문에 여기에서 비활성화됩니다.

플레이북은 위의 명령줄에서 7개의 값 중 세 가지를 설정합니다.

- 그룹(호스트: **all**)
- 연결 방법(**Connection: ansible.netcommon.network\_cli**) 및
- 모듈(각 작업에 있음).

플레이북에 해당 값을 설정하면 명령줄에서 해당 값을 생략할 수 있습니다. 플레이북은 구성 출력을 표시하는 두 번째 작업도 추가합니다.

**vyos.vyos\_facts** 또는 **ansible.builtin.setup** 과 같은 컬렉션 관련 팩트 모듈을 통해 시스템에서 팩트를 수집할 때 수집된 데이터는 콘솔에 기록되는 대신 향후 작업에서 사용하기 위해 메모리에 보관됩니다.

모듈이 플레이북에서 실행되면 콘솔에 쓰는 대신 향후 작업에서 사용할 수 있도록 출력이 메모리에 유지됩니다. 대부분의 다른 모듈에서는 모듈 또는 작업의 출력을 저장하고 재사용할 변수를 명시적으로 등록해야 합니다.

팩트에 대한 자세한 내용은 *Ansible Playbook 참조 가이드*의 [\[Ansible facts\]](#)를 참조하십시오.

다음 디버그 작업을 사용하면 셸에 결과를 볼 수 있습니다.

### 프로세스

1. 다음 명령을 사용하여 플레이북을 실행합니다.

```
ansible-playbook -i vyos.example.net, -u ansible -k -e ansible_network_os=vyos.vyos.vyos
first_playbook.yml
```

플레이북에는 두 개의 작업이 있는 하나의 플레이가 포함되어 있으며 다음과 같은 출력을 생성합니다.

```
$ ansible-playbook -i vyos.example.net, -u ansible -k -e ansible_network_os=vyos.vyos.vyos
first_playbook.yml

PLAY [Network Getting Started First Playbook]
*****
*****

TASK [Get config for VyOS devices]
*****
*****
ok: [vyos.example.net]

TASK [Display the config]
*****
*****
ok: [vyos.example.net] => {
  "msg": "The hostname is vyos and the OS is VyOS 1.1.8"
}
```

- 이제 장치 구성을 검색할 수 있으므로 Ansible로 업데이트를 시도할 수 있습니다.
- 첫 번째 플레이북의 확장 버전인 **first\_playbook\_ext.yml** 을 [여기에서](#) 다운로드합니다. 플레이북은 다음과 같습니다.

```
---

- name: Network Getting Started First Playbook Extended
  connection: ansible.netcommon.network_cli
  gather_facts: false
  hosts: all
  tasks:

    - name: Get config for VyOS devices
      vyos.vyos.vyos_facts:
        gather_subset: all

    - name: Display the config
      debug:
        msg: "The hostname is {{ ansible_net_hostname }} and the OS is {{ ansible_net_version
        }}"

    - name: Update the hostname
      vyos.vyos.vyos_config:
        backup: yes
        lines:
          - set system host-name vyos-changed

    - name: Get changed config for VyOS devices
      vyos.vyos.vyos_facts:
        gather_subset: all

    - name: Display the changed config
```

```

debug:
  msg: "The new hostname is {{ ansible_net_hostname }} and the OS is {{
ansible_net_version }}"

```

4. 확장 첫 번째 플레이북에는 단일 플레이에 5개의 작업이 있습니다.
5. 다음 명령을 사용하여 플레이북을 실행합니다.

```

$ ansible-playbook -i vyos.example.net, -u ansible -k -e ansible_network_os=vyos.vyos.vyos
first_playbook_ext.yml

```

6. 출력에는 구성에 대한 Ansible 변경 사항이 표시됩니다.

```

$ ansible-playbook -i vyos.example.net, -u ansible -k -e ansible_network_os=vyos.vyos.vyos
first_playbook_ext.yml

```

```

PLAY [Network Getting Started First Playbook Extended]

```

```

*****
*****

```

```

TASK [Get config for VyOS devices]

```

```

*****
*****

```

```

ok: [vyos.example.net]

```

```

TASK [Display the config]

```

```

*****
*****

```

```

ok: [vyos.example.net] => {
  "msg": "The hostname is vyos and the OS is VyOS 1.1.8"
}

```

```

TASK [Update the hostname]

```

```

*****
*****

```

```

changed: [vyos.example.net]

```

```

TASK [Get changed config for VyOS devices]

```

```

*****
*****

```

```

ok: [vyos.example.net]

```

```

TASK [Display the changed config]

```

```

*****
*****

```

```

ok: [vyos.example.net] => {
  "msg": "The new hostname is vyos-changed and the OS is VyOS 1.1.8"
}

```

```

PLAY RECAP

```

```

*****
*****

```

```

vyos.example.net      : ok=5  changed=1  unreachable=0  failed=0

```

## 2.3. 네트워크 장치에서 팩트 수집

**gather\_facts** 키워드는 표준화된 키/값 쌍으로 네트워크 장치 팩트 수집을 지원합니다. 이러한 네트워크 사실을 추가 작업에 제공하여 네트워크 장치를 관리할 수 있습니다. 아래 표시된 대로 네트워크 \*\_facts 모듈(예: **arista.eos.eos\_facts**)과 함께 **gather\_network\_resources** 매개변수를 사용하여 장치 구성의 하위 집합을 반환할 수도 있습니다.

```
- hosts: arista
gather_facts: True
gather_subset: interfaces
module_defaults:
  arista.eos.eos_facts:
    gather_network_resources: interfaces
```

플레이북은 다음 인터페이스 팩트를 반환합니다.

```
"network_resources": {
  "interfaces": [
    {
      "description": "test-interface",
      "enabled": true,
      "mtu": "512",
      "name": "Ethernet1"
    },
    {
      "enabled": true,
      "mtu": "3000",
      "name": "Ethernet2"
    },
    {
      "enabled": true,
      "name": "Ethernet3"
    },
    {
      "enabled": true,
      "name": "Ethernet4"
    },
    {
      "enabled": true,
      "name": "Ethernet5"
    },
    {
      "enabled": true,
      "name": "Ethernet6"
    }
  ]
}
```



### 참고

**gather\_network\_resources** 는 구성 데이터를 지원되는 모든 리소스 (**interfaces/bgp/ospf/etc**)에 대한 팩트로 렌더링하는 반면 **gather\_subset** 은 주로 운영 데이터를 가져오는 데 사용됩니다.

이러한 팩트를 저장하고 **eos\_interfaces** 리소스 모듈과 같은 다른 작업에 직접 사용할 수 있습니다.

## 3장. ANSIBLE 플레이북의 실용적인 예

Ansible은 클라우드 기반 REST API에서 Linux 및 Windows 시스템, 네트워킹 하드웨어 등에 이르기까지 다양한 장치 분류와 통신할 수 있습니다.

다음은 두 가지 유형의 서버를 자동으로 업데이트하는 두 가지 Ansible 모듈의 샘플입니다.

### 3.1. 플레이북 실행

플레이북은 맨 위에서 아래로 실행됩니다. 각 플레이 내에서 작업도 위에서 아래로 실행됩니다. 여러 '플레이'가 있는 플레이북은 다중 시스템 배포를 오케스트레이션하고, WebAssembly에서 하나의 플레이를 실행한 다음, 데이터베이스 서버에서 다른 플레이, 네트워크 인프라에서 세 번째 플레이 등을 오케스트레이션할 수 있습니다.

최소한 각 플레이는 다음 두 가지를 정의합니다.

- 패턴을 사용하여 대상으로 하는 관리형 노드
- 실행할 하나 이상의 작업



#### 참고

Ansible 2.10 이상에서는 플레이북에서 정규화된 컬렉션 이름을 사용하여 여러 컬렉션에 동일한 이름의 모듈을 포함할 수 있으므로(예: **사용자**) 올바른 모듈이 선택되었는지 확인합니다.

자세한 내용은 [플레이북에서 컬렉션 사용](#)을 참조하십시오.

이 예제에서 첫 번째 플레이는 웹 서버를 대상으로 합니다. 두 번째 플레이는 데이터베이스 서버를 대상으로 합니다.

```
---
- name: Update web servers
  hosts: webservers
  become: true

  tasks:
    - name: Ensure apache is at the latest version
      ansible.builtin.yum:
        name: httpd
        state: latest
    - name: Write the apache config file
      ansible.builtin.template:
        src: /srv/httpd.j2
        dest: /etc/httpd.conf
        mode: "0644"

- name: Update db servers
  hosts: databases
  become: true

  tasks:
    - name: Ensure postgresql is at the latest version
      ansible.builtin.yum:
```

```
name: postgresql
state: latest
- name: Ensure that postgresql is started
  ansible.builtin.service:
    name: postgresql
    state: started
```

Playbook에는 두 개의 플레이가 포함되어 있습니다.

- 첫 번째는 웹 서버 소프트웨어가 최신 상태인지 확인하고 필요한 경우 업데이트를 실행합니다.
- 두 번째는 데이터베이스 서버 소프트웨어가 최신 상태인지 확인하고 필요한 경우 업데이트를 실행합니다.

플레이북은 단순한 호스트 행 및 작업 이상을 포함할 수 있습니다.

예를 들어 이 플레이북은 각 플레이에 대해 `remote_user`를 설정합니다. SSH 연결의 사용자 계정입니다. 플레이북, 플레이 또는 작업 수준에서 다른 플레이북 키워드를 추가하여 Ansible 작동 방식에 영향을 미칠 수 있습니다. 플레이북 키워드는 연결 플러그인, 권한 에스컬레이션 사용 여부, 오류 처리 방법 등을 제어할 수 있습니다.

Ansible을 사용하면 다양한 환경을 지원하기 위해 이러한 매개변수를 Ansible 구성 또는 인벤토리에서 명령줄 플래그로 설정할 수 있습니다. 이러한 데이터 소스에 대한 우선 순위 규칙을 학습하면 Ansible 에코시스템을 확장하는 데 도움이 될 수 있습니다.

## 4장. 오픈 소스 라이선스

### GNU 일반 공개 라이선스

버전 3, 2007년 6월 29일

Copyright © 2007 Free Software Foundation, Inc. <<https://fsf.org/>>

모든 사람이 이 라이선스 문서의 대문자 사본을 복사하고 배포할 수 있지만 변경할 수는 없습니다.

#### preamble

GNU General Public License는 소프트웨어 및 기타 종류의 작업에 대한 무료 copyleft 라이선스입니다.

대부분의 소프트웨어 및 기타 실제 작업에 대한 라이선스는 작업을 공유하고 변경할 수 있는 자유를 확보하도록 고안되어 있습니다. 반대로, GNU General Public License는 프로그램의 모든 버전을 공유하고 변경할 수 있는 자유를 보장하기 위해 모든 사용자를 위한 자유 소프트웨어를 유지하기 위한 것입니다. 자유 소프트웨어 기반은 대부분의 소프트웨어에 대해 GNU General Public License를 사용합니다. 이 라이선스는 작성자가 이러한 방식으로 릴리스된 다른 작업에도 적용됩니다. 이를 귀하의 프로그램에도 적용할 수 있습니다.

우리가 자유 소프트웨어에 대해 말할 때, 우리는 값이 아닌 자유를 나타냅니다. 당사의 일반 공개 라이선스는 귀하가 자유 소프트웨어의 사본을 배포할 수 있도록 고안되어 있으며, 원하는 경우 소스 코드를 받거나 소스 코드를 받을 수 있는지, 원하는 경우 소프트웨어를 변경하거나 새로운 자유 프로그램에서 일부를 사용할 수 있으며, 이러한 작업을 수행할 수 있다는 것을 알고 있습니다.

귀하의 권리를 보호하기 위해, 당사는 다른 사람이 귀하에게 이러한 권리를 거부하거나 귀하가 권리를 속이도록 요청하는 것을 방지해야 합니다. 따라서 소프트웨어의 사본을 배포하거나 수정하려는 경우 특정 책임이 있습니다: 다른 사람의 자유를 존중해야 할 책임이 있습니다.

예를 들어, 만족스럽거나 유료로든 이러한 프로그램의 사본을 배포하는 경우 수신자에게 수령한 동일한 자유를 전달해야 합니다. 소스 코드를 수신하거나 가져올 수 있는지 확인해야 합니다. 그리고 이러한 조건을 제시하여 자신의 권리를 알아야 합니다.

GNU GPL을 사용하는 개발자는 두 단계로 귀하의 권리를 보호합니다: (1) 소프트웨어에 대한 저작권을 주장하고 (2) 귀하에게 복사, 배포 및/또는 수정할 수 있는 법적 권한을 부여합니다.

개발자 및 작성자의 보호를 위해 GPL은 이 무료 소프트웨어에 대한 보증이 없음을 명확히 설명합니다. 사용자와 작성자 모두 GPL은 수정 버전을 변경으로 표시해야 하므로 이전 버전의 작성자에게는 문제가 잘못 표시되지 않습니다.

일부 장치는 제조업체가 이러한 작업을 수행할 수 있지만 사용자가 해당 소프트웨어 내에서 수정된 버전의 소프트웨어를 설치하거나 실행하기 위한 액세스를 거부하도록 설계되었습니다. 이는 소프트웨어 변경을 위한 사용자의 자유를 보호하기 위한 목적에 근본적으로 호환되지 않습니다. 이러한 악용의 체계적인 패턴은 개인이 사용할 수 있는 제품의 영역에서 발생하며, 이는 가장 용인 할 수 없는 곳입니다. 따라서 이 버전의 GPL은 해당 제품에 대한 관행을 금지하도록 설계되었습니다. 이러한 문제가 다른 영역에서 크게 발생하는 경우, 당사는 GPL의 향후 버전에서 사용자의 자유를 보호하는 데 필요한 경우 이러한 도메인을 해당 도메인에 확장할 준비가 되어 있습니다.

마지막으로 모든 프로그램은 소프트웨어 공급업체에 의해 지속적으로 위협 받고 있습니다. 국가는 전사가 범용 컴퓨터에서 소프트웨어의 개발 및 사용을 제한하도록 허용하지 않아야 하지만, 그렇게 할 경우 무료 프로그램에 적용되는 특수한 위협을 피하고자합니다. 이를 방지하기 위해 GPL은 해당 프로그램을 무료로 렌더링하는 데 공급업체를 사용할 수 없음을 보장합니다.

복사, 배포 및 수정을 위한 정확한 용어 및 조건은 다음과 같습니다.

#### 이용 약관



## 0. 정의.

"이 라이선스"는 GNU General Public License의 버전 3을 나타냅니다.

또한 "Copyright"는 다른 종류의 작업에 적용되는 저작권과 유사한 법률(예: Cryostat 마스크)을 의미합니다.

"소프트웨어"는 이 라이선스에 따라 라이선스가 부여된 모든 저작권을 나타냅니다. 각 라이선스는 "귀하"로 처리됩니다. "라이선스"와 "보유자"는 개인이나 조직이 될 수 있습니다.

작업을 "수정"한다는 것은 정확한 사본을 만드는 것 이외의 저작권 권한이 필요한 방식으로 작업의 전체 또는 일부를 복사하거나 조정하는 것을 의미합니다. 결과 작업을 이전 작업의 "수정 버전" 또는 이전 작업 "기반"이라고 합니다.

"취소된 작업"은 수정되지 않은 프로그램 또는 프로그램에 기반을 둔 작업을 의미합니다.

작업을 "승인"한다는 것은 허가없이 컴퓨터에서 실행하거나 개인 사본을 수정하는 것을 제외하고는 해당 저작권법 위반에 대한 직접 또는 두 번째 책임이 있음을 의미합니다. 전파에는 복사, 배포(수정 포함 또는 수정 없이), 대중 및 일부 국가의 다른 활동도 포함됩니다.

작업을 "동의"한다는 것은 다른 당사자가 사본을 만들거나 받을 수 있도록 하는 모든 종류의 전파를 의미합니다. 복사를 전송하지 않고 컴퓨터 네트워크를 통해 사용자와의 상호 작용은 전달되지 않습니다.

대화형 사용자 인터페이스는 "적용한 법적 고지 사항"이 포함되어 있는 범위까지 (1) 적절한 저작권 공지를 표시하고 (2) 사용자에게 작업에 대한 보장이 없음을 알립니다(보증이 제공되는 범위를 제외하고), 라이선스는 이 라이선스의 사본을 보는 방법과 이 라이선스의 사본을 보는 방법을 사용자에게 알려줍니다. 인터페이스가 메뉴와 같은 사용자 명령 또는 옵션 목록을 표시하는 경우 목록의 눈에 띄는 항목이 이 기준을 충족합니다.

## 1. 소스 코드.

작업에 대한 "소스 코드"는 수정을 위해 권장되는 형식의 작업을 의미합니다. "오브젝트 코드"는 작업의 소스가 아닌 형식을 의미합니다.

"표준 인터페이스"는 인식된 표준 본문에서 정의한 공식 표준이거나 특정 프로그래밍 언어를 위해 지정된 인터페이스의 경우 해당 언어로 작업하는 개발자 간에 널리 사용되는 인터페이스를 의미합니다.

실행 가능한 작업의 "시스템 라이브러리"에는 전체 작업 이외의 모든 항목이 포함됩니다. (a) 주요 구성 요소의 패키징에 일반적인 형식으로 포함되지만 (b) 주요 구성 요소의 일부가 아니며 (b) 해당 주요 구성 요소와 함께 작업을 사용하거나 소스 코드 형식으로 구현할 수 있는 표준 인터페이스만 구현하는 역할을 합니다. 이 컨텍스트에서 "Major Component"는 실행 파일이 실행되는 특정 운영 체제(있는 경우)의 주요 필수 구성 요소(커널, 창 시스템 등) 또는 작업을 생성하는 데 사용되는 컴파일러 또는 이를 실행하는 데 사용되는 오브젝트 코드 인터프리터를 의미합니다.

오브젝트 코드 형식의 작업에 대한 "Corresponding Source"는 개체 코드를 생성, 설치 및 (실행 가능한 작업의 경우)하는 데 필요한 모든 소스 코드를 의미하고 해당 작업을 제어하기 위한 스크립트를 포함하여 작업을 수정합니다. 그러나 작업의 System Libraries 또는 일반적인 목적의 도구 또는 일반적으로 이러한 활동을 수행하는 데 수정되지 않고 작업의 일부가 아닌 무료 프로그램을 포함하지 않습니다. 예를 들어, Corresponding Source에는 작업을 위한 소스 파일과 관련된 인터페이스 정의 파일, 공유 라이브러리의 소스 코드 및 해당 하위 프로그램 및 작업의 기타 부분 간의 절충된 데이터 통신 또는 제어 흐름과 같이 작업이 특별히 요구하도록 설계된 동적으로 연결된 하위 프로그램 등이 포함됩니다.

Corresponding Source에는 사용자가 Corresponding Source의 다른 부분에서 자동으로 다시 생성할 수 있는 항목이 포함되어 있지 않습니다.

소스 코드 형식의 작업에 대한 Corresponding Source는 동일한 작업입니다.

## 2. 기본 권한.

이 라이선스에 따라 부여된 모든 권리는 프로그램에 대한 저작권에 대해 부여되며, 명시된 조건이 충족되면 되돌릴 수 없습니다. 이 라이선스는 수정되지 않은 프로그램을 실행할 수 있는 무제한 권한을 명시적으로 나타냅니다. 처리된 작업 실행의 출력은 출력이 제공된 경우에만 이 라이선스에서 적용됩니다. 내용이 적용되는 작업을 구성합니다. 이 라이선스는 저작권법에서 제공하는 공정 사용 또는 기타 동등한 권리에 동의합니다.

라이선스가 유효한 상태로 유지되는 한 조건 없이 전달하지 않는 작업을 실행, 실행 및 전파할 수 있습니다. 귀하는 귀하가 저작권을 통제하지 않는 모든 자료를 전달하여 본 사용 약관을 준수하고 있는 경우, 해당 작업을 수행하기 위한 유일한 목적을 위해 제공된 작업을 다른 사람에게 전달하거나, 귀하에게 해당 작업을 실행할 수 있는 기능을 제공할 수 있습니다. 따라서 귀하가 처리한 작업을 수행하거나 실행하는 것은 귀하의 지시와 제어 하에 전적으로 귀하의 지시에 따라 귀하와의 관계를 벗어나서 저작권이 있는 자료의 사본을 만들 수 없도록 하는 것입니다.

다른 모든 상황에서의 전달은 아래에 명시된 조건에 따라만 허용됩니다. 하위 라이선스는 허용되지 않습니다. 섹션 10은 필요하지 않습니다.

## 3. 비인증 범으로부터 사용자의 법적 권리 보호.

적용 대상 작업은 12월 20일, 이러한 조치의 금지 또는 금지하는 유사한 법률에 따라 채택된 11조의 제11조에 따른 해당 법률 이행에 따른 효과적인 기술적 조치의 일부로 간주되지 않습니다.

귀하가 적용 대상 작업을 전달할 때, 귀하는 기술 조치를 금지할 수 있는 법적 권한을 포기할 때, 이러한 우회는 본 라이선스에 따른 행사 권리에 의해 영향을 받는 범위 내에서, 귀하는 해당 작업의 운영이나 수정을 금지하는 수단에 대해, 귀하의 또는 제3자의 기술조치에 대한 법적 권리를 포기해야 합니다.

## 4. Ver#159im 복사 전달.

귀하는 프로그램 소스 코드의 사본이 귀하가 수신한 대로, 모든 매체에서 각 사본에 따라 적절한 저작권 통지에 적절하고 적절하게 게시되는 경우, 본 라이선스 및 7항에 따라 추가 금지된 모든 약관을 그대로 유지하며, 어떠한 보증의 부재에 대한 모든 통지는 그대로 유지; 모든 수신자에게 보증이 없는 것에 대한 모든 통지를 제공합니다.

귀하가 전달하는 각 사본에 대해 가격이나 가격을 청구할 수 있으며, 귀하는 요금에 대한 지원 또는 보증 보호를 제공할 수 있습니다.

## 5. 소스 버전 전달.

귀하는 프로그램에 따라 작업을 전달할 수 있습니다, 또는 프로그램에서 생성하기 위한 수정, 섹션 4의 조건에 따라 소스 코드의 형태로, 다음의 모든 조건을 충족하는 경우:

- a) 작업은 사용자가 수정하고 관련 날짜를 제공하는 것을 알리는 눈에 띄는 통지를 해야 합니다.
- b) 이 작업은 이 라이선스 및 섹션 7에서 추가된 모든 조건에 따라 릴리스됨을 알리는 눈에 띄게 통지해야 합니다. 이 요구 사항은 섹션 4의 요구 사항을 "보 그대로 모든 알림 유지"로 수정합니다.
- c) 귀하는 전체적으로 사본을 소유하고 있는 모든 사람이 이 사용허락에 따라 전체 작업을 승인해야 합니다. 따라서 이 라이선스는 적용 가능한 모든 섹션 7 추가 조건, 작업의 전체 및 모든 해당 부분에 적용되는 방법에 관계없이 적용됩니다. 이 라이선스는 다른 방식으로 작업을 라이선스할 수 있는 권한을 부여하지 않지만 별도로 수령한 경우 이러한 권한이 무효화되지 않습니다.
- d) 작업에 대화형 사용자 인터페이스가 있는 경우 각각 Appropriate Legal Notices를 표시해야 합니다; 그러나 프로그램에 Appropriate 법적 고지를 표시하지 않는 대화형 인터페이스가 있는 경우, 귀하의 작업이 그렇게 할 필요가 없습니다. 대상 작업의 특성 확장에 속하지 않는 다른 별도의 독립적 작업으로 구성된 작업의 편집으로, 그리고 더 큰 프로그램을 형성하기 위해 결합되지 않는,

저장 또는 배포 매체의 볼륨과 결합되지 않는 경우 컴파일 및 그 결과 저작권은 편집 대상 사용자의 액세스 또는 법적 권리를 제한하지 않는 경우 "aggregate"라고 합니다. 집계에 적용되는 작업을 포함하면 이 라이선스가 집계의 다른 부분에 적용되지 않습니다.

## 6. 비소스 Cryostat 전달.

다음 방법 중 하나로 시스템에서 읽을 수 있는 Corresponding Source를 전달하면 섹션 4 및 5의 조건에 따라 대상 작업을 전달할 수 있습니다.

- a) 물리적 제품(물리적 배포 매체 포함)과 함께 오브젝트 코드를 통합하거나 소프트웨어 상호 교환에 사용되는 표준 물리적 매체에서 고정된 자료로 통합합니다.
- b) 개체 코드를 포함하거나 물리적 제품(물리적 배포 매체 포함)과 함께 최소 3년 동안 유효하고 해당 제품 모델에 대한 예비 부분 또는 고객 지원을 제공하는 동안 유효하며 (1)이 라이선스가 적용되는 모든 소프트웨어에 대한 수정 소스 복사본을 보유한 모든 사용자에게 제공합니다. 소프트웨어 상호 교환에 사용되는 physical medium customarily에서, 소스 전달을 물리적으로 수행하는 데 필요한 비용보다 더 적은 가격 또는 (2) 네트워크 서버에서 Corresponding Source를 무료로 복사하는 액세스입니다.
- c) Corresponding Source를 제공하기 위해 작성된 제안 사본과 함께 오브젝트 코드의 개별 복사본을 확인합니다. 이러한 대안은 때때로 또는 비상업적으로만 허용되며, 6b 섹션에 따라 이러한 제안을 통해 오브젝트 코드를 수령한 경우에만 허용됩니다.
- d) 지정된 위치(종료 또는 비용)로부터 액세스를 제공하고, 오브젝트 코드를 추가로 청구하지 않고 동일한 방식으로 Corresponding Source에 대한 동등한 액세스를 제공합니다. 수신자는 개체 코드와 함께 Corresponding 소스를 복사할 필요가 없습니다. 개체 코드를 복사하는 위치가 네트워크 서버인 경우, Corresponding Source는 동일한 복사 기능을 지원하는 다른 서버(사용자 또는 타사에서 작동함)에 있을 수 있으며, Corresponding Source를 찾을 위치를 나타내는 개체 코드 옆에 명확한 지침을 유지합니다. Corresponding Source를 호스팅하는 서버에 관계없이 이러한 요구 사항을 충족하는 데 필요한 기간 동안 해당 소스를 사용할 수 있는지 확인해야 합니다.
- e) 피어 투 피어 전송을 사용하여 오브젝트 코드를 확인하고, 다른 피어에게 오브젝트 코드와 수정 소스가 6d항에 따라 일반 대중에게 제공되는지 알려줍니다.

소스 코드가 시스템 라이브러리로 Corresponding 소스에서 제외되는 개체 코드의 분리 가능한 부분은 개체 코드 작업을 전달하는 데 포함될 필요가 없습니다.

"사용자 제품"은 (1) 개인, 가족 또는 가정 목적으로 일반적으로 사용되는 모든 유형의 개인 자산 또는 (2) 여행에 대한 통합을 위해 설계되거나 판매되는 모든 개인 재산입니다. 제품이 소비자 제품인지 여부를 결정할 때 적용 범위를 선호하여 의심스러운 사례를 해결합니다. 특정 사용자가 수신한 특정 제품의 경우 "일반적으로 사용되는"는 특정 사용자의 상태 또는 특정 사용자가 실제로 사용하거나 예상하는 방법의 상태에 관계없이 제품의 일반적인 또는 일반적인 사용을 나타냅니다. 제품은 제품이 실질적인 상업적, 산업 또는 비 소비자 사용 여부에 관계없이 소비자 제품으로, 이러한 사용이 제품의 중요한 사용 모드를 나타내지 않는 한 소비자 제품입니다.

사용자 제품에 대한 "설치 정보"는 수정된 버전의 Corresponding Source에서 해당 사용자 제품의 수정된 작업 버전을 설치 및 실행하는 데 필요한 방법, 절차, 권한 부여 키 또는 기타 정보를 의미합니다. 수정된 오브젝트 코드의 지속적인 기능이 어떠한 경우에도 수정이 이루어졌기 때문에 단독으로 방해되거나 간섭되지 않도록 하려면 정보가 충분해야 합니다.

사용자가 개체 코드 작업을 이 섹션에서 또는 구체적으로 사용자 제품에서 사용하기 위해, 그리고 특히 사용자 제품의 소유 및 사용 권한이 있는 트랜잭션의 일부로 발생하는 경우 또는 이 섹션에 기재된 수정 소스가 있는 경우, 또는 이 섹션에 전달된 수정 소스는 설치 정보를 제공해야 합니다. 그러나 이러한 요구 사항은 사용자 제품에 수정된 오브젝트 코드를 설치하는 기능을 보유하고 있지 않은 경우(예를 들어, work has been installed)는 적용되지 않습니다.

설치 정보를 제공해야 하는 요구 사항에는 수신자가 수정하거나 설치한 작업에 대한 지원 서비스, 보증 또는 업데이트를 계속 제공해야 합니다. 네트워크에 대한 액세스는 수정 자체가 실질적으로 네트워크 작동에 영향을 주거나 네트워크 간 통신을 위한 규칙 및 프로토콜을 위반할 때 거부될 수 있습니다.

이 섹션에 따라 전달된 해당 소스 및 설치 정보는 공개적으로 문서화된 형식이어야 하며(소스 코드 형식으로 공개되는 구현과 함께) 압축 해제, 읽기 또는 복사를 위해 특별한 암호나 키가 필요하지 않아야 합니다.

## 7. 추가 약관.

"추가 권한"은 하나 이상의 조건에서 예외를 만들어 이 라이선스 조건을 보완하는 용어입니다. 전체 프로그램에 적용되는 추가 권한은 관련 법률에 따라 유효한 범위 내에서 본 라이선스에 포함된 것처럼 취급됩니다. 추가 권한이 프로그램의 일부에만 적용되는 경우 해당 부분은 해당 권한에 따라 별도로 사용할 수 있지만 전체 프로그램은 추가 권한에 관계없이 이 라이선스의 적용을 받습니다.

대상 작업의 사본을 전달할 때 해당 사본에서 추가 권한을 제거하거나 해당 작업의 일부에서 추가 권한을 제거할 수 있습니다.(작업을 수정할 때 특정 경우에 자체 제거를 요구하도록 추가 권한을 작성할 수 있습니다.) 귀하가 귀하가 적절한 저작권 권한을 보유하거나 제공할 수 있는 대상 작업에 추가 권한을 부여한 자료에 대한 추가 권한을 배치할 수 있습니다.

이 라이선스의 다른 조항에도 불구하고, 귀하가 다루는 작업에 추가하는 자료를 위해, 귀하는 (해당 자료의 저작권 소유자가 승인한 경우) 본 사용 약관을 보완할 수 있습니다.

- a) 본 라이선스의 섹션 15 및 16항과 다른 방식으로 보증 또는 제한적 보증; 또는
- b) 해당 자료 또는 해당 자료 또는 이를 포함하는 작업에서 표시된 Appropriate 법적 고지에 대한 지정된 합리적인 통지 또는 작성자의 보존을 요청; 또는
- c) 해당 자료의 출처를 잘못 표현하거나 이러한 자료의 수정된 버전이 원본 버전과 다른 합리적인 방식으로 표시되도록 요구하는 경우; 또는
- d) 라이선스 제공자 또는 자료 작성자의 이름 공개 목적의 사용을 제한; 또는
- e) 일부 상표명, 상표 또는 서비스 마크 사용에 대한 상표 법률에 따라 권리를 부여할 수 있는 거부; 또는
- f) 이러한 계약 가정이 해당 라이선스와 작성자에 직접적으로 부과된다는 판단에 대해 계약상의 가정과 함께 자료를 전달하는 모든 사람이 해당 자료의 저작자 및 작성자를 추정해야 합니다.

다른 모든 허용되지 않는 추가 용어는 섹션 10의 의미 내에서 "유독한 제한"으로 간주됩니다. 귀하가 이를 수신한 프로그램이나 그 일부에 해당 프로그램이 추가 제한과 함께 이 라이선스에 의해 관리된다는 통지를 포함하는 경우, 귀하는 해당 용어를 제거할 수 있습니다. 라이선스 문서에 추가 제한이 포함되어 있지만 이 라이선스에 따라 라이선스가 부여되거나 이를 승인할 수 있는 경우, 추가 제한이 이러한 라이선스 또는 전달을 받지 않는 경우 해당 라이선스 문서의 조건에 따라 적용되는 적용 가능한 작업 자료를 추가할 수 있습니다.

이 섹션에 설명된 작업에 용어를 추가하는 경우 관련 소스 파일, 해당 파일에 적용되는 추가 용어의 설명 또는 해당 용어를 찾을 위치를 나타내는 공지를 배치해야 합니다.

추가 조건, 허용 또는 허용되지 않는 조건은 별도로 작성된 라이선스 형태로 명시되거나 예외로 명시될 수 있습니다. 위의 요구 사항이 두 가지 방법으로 적용됩니다.

## 8. 종료.

이 라이선스에 따라 명시적으로 제공된 경우를 제외하고 다루는 작업을 전파하거나 수정할 수 없습니다. 다른 방법으로 이를 전파하거나 수정하려는 시도는 무효화되며, 이 라이선스에 따라 귀하의 권리를 자동으로 종료할 것입니다(섹션 11 섹션의 제3조에 따라 부여된 모든 관련 라이선스 포함).

그러나 귀하가 본 라이선스의 모든 위반을 중단하는 경우, 특정 저작권 소유자로부터 귀하의 라이선스는 (a) 저작권 소유자가 최종적으로 귀하의 라이선스를 명시적으로 종료할 때까지, (b) 저작권 소유자가 중단 후 60일 전에 위반에 대해 통지하지 못하는 경우 영구적으로 복원됩니다.

또한, 특정 저작권자의 라이선스는 저작권 소유자가 귀하에게 합리적인 방법으로 위반을 통지하는 경우 영구적으로 복원되며, 이는 귀하가 저작권 소유자로부터 본 라이선스 위반에 대한 통지를 받은 첫 번째이며, 귀하는 통지 수령 후 30일 전에 위반을 제기합니다.

이 섹션에 따른 귀하의 권리 종료는 이 라이선스에 따라 귀하가 사본을 수령한 당사자의 라이선스를 종료하지 않습니다. 귀하의 권리가 종료되고 영구적으로 복원되지 않은 경우, 섹션 10에서 동일한 자료에 대한 새로운 라이선스를 받을 수 없습니다.

## 9. 수락이 복사를 위해 필요하지 않습니다.

프로그램의 사본을 수신하거나 실행하기 위해 이 라이선스를 수락할 필요는 없습니다. 피어-투-피어 전송을 사용하여 복사본을 수신한 결과만 발생하는 처리 대상 작업의 제공은 수락을 필요로 하지 않습니다. 그러나 이 라이선스 이외의 경우, 적용되는 모든 작업을 전파하거나 수정할 수 있는 권한을 부여하는 것은 없습니다. 귀하가 이 라이선스를 수락하지 않는 경우 이러한 행위는 저작권을 대상으로 합니다. 따라서, 적용 대상 작업을 수정하거나 전파함으로써, 귀하는 귀하가 이를 수행할 수 있도록 이 라이선스를 수락했음을 나타냅니다.

## 10. 다운스트림 수신자 자동 라이선스.

대상 작업을 전달할 때마다 수신자는 이 라이선스에 따라 원래 라이선스로부터 라이선스를 자동으로 수신하여 해당 작업을 실행, 수정 및 전파합니다. 귀하는 이 라이선스를 통해 제3자의 준수를 이행할 책임이 없습니다.

"entity transaction"은 조직의 트랜잭션 제어 또는 한 조직의 실질적으로 모든 자산을 전송하거나 조직을 병합하는 것입니다. 법인 거래에서 처리된 작업 결과를 전파하는 경우, 해당 거래의 각 당사자는 관심의 사본을 수신하거나 이전 단락에 따라 제공할 수 있는 작업에 대한 라이선스를 수신하고, 이전 단락에 따라 제공할 수 있는 권리, 그리고 목제가 관심의 매수에서 작업의 출처를 소유할 수 있는 권리, 또는 합리적인 노력을 얻을 수 있습니다.

이 라이선스에 따라 부여되거나 확인되는 권리의 행사에 더 이상의 제한을 부과할 수 없습니다. 예를 들어, 본 라이선스에 따라 부여된 권리의 이행에 대한 라이선스 비용, 또는 기타 책임을 부과할 수 없으며, 귀하는 (복합에서 교차 청구 또는 거부를 포함하여) 프로그램을 작성, 사용, 판매, 판매, 제공 또는 프로그램 또는 일부를 임포트함으로써 제기되는 권리를 개시할 수 없습니다.

## 11. 오픈 소스입니다.

"contributor"는 프로그램의 이 라이선스에 따라 또는 프로그램이 기반을 둔 작업 하에 사용할 수 있는 저작권 소유자입니다. 따라서 라이선스가 부여된 작업을 기여자의 "프로덕터 버전"이라고 합니다.

기여자의 "필수 자격증 청구"는 기여자가 이미 인수했거나 인수한 후, 이 라이선스에 의해 허용되거나, 기여자 버전을 생성, 사용 또는 판매하거나 판매할 수 있는 주장을 소유하거나 제어하지만, 기여자 버전에 대한 추가 수정의 결과로서만 관련 청구를 포함하지는 않습니다. 이러한 정의의 목적을 위해, "통제"에는 이 라이선스의 요구 사항과 일치하는 방식으로 저작권 하위 라이선스를 부여할 수 있는 권한이 포함됩니다.

각 기여자는 기여자의 필수 자립 클레임에 따라 전 세계적으로 포괄적이지 않고 무독점 라이선스를 부여하여 기여자 버전의 기여자 버전 내용을 작성, 사용, 판매, 제안, 가져오기 및 다른 방식으로 실행, 수정 및 전파할 수 있습니다.

다음 세 단락에서, "병합 라이선스"는 어떠한 명시적 계약이나 약정이지만, 저작권을 행사할 수 있는 명시적 승인(예를 들어, 부정확성 또는 민주를 이행하지 않을 수 있는 명시적인 승인)입니다. 당사자에게 이러한 지식허가를 "승인"한다는 것은 당사자에 대한 권리를 행사하지 않을 것을 의미합니다.

귀하가 적용 대상 작업을 전달하거나, 지식허용 라이선스에 의존하는 것과, 작업의 Corresponding 소스는 공개적으로 사용 가능한 네트워크 서버 또는 기타 쉽게 액세스할 수 있는 수단을 통해 이 라이선스의 조건 하에서 복사하거나, 무료로 복사하거나, 또는 다른 쉽게 액세스할 수 있는 수단을 통해 복사용 소스를 사용할 수 없는 경우, (1) 사용 가능한 Corresponding Source to be so available, 또는 (2) 특정 작업에 대한 라이선스의 이점을 위임해야 합니다. 또는 (3) 본 라이선스의 요구 사항과 일치하는 방식으로, 이 라이선스에 따라 다운스트림 수신자로 저작권허가를 연장할 수 있습니다. "알당히 의존하는"란 실제 지식을 가지고 있지만, 지식의 경우, 지식의 경우, 귀하가 해당 국가에서 다루는 업무를 전달하는 경우, 또는 귀하의 수신자가 귀하가 신뢰할 수 있는 이유가 있는 국가 내에서 하나 이상의 식별 가능한 기여를 훼손할 수 있음을 의미합니다.

단일 거래 또는 배열에 따라, 귀하는 전달하거나, 커버된 작업의 전달을 통해 전파 또는 전파하고, 적용 대상 작업의 특정 사본을 사용, 전파, 수정, 전달 또는 전달할 것을 승인한 당사자의 일부 당사자에게 이식허가를 부여한 경우, 귀하가 부여한 인수는 자동으로 적용 대상의 모든 수신자에게 확장됩니다.

본 라이선스는 해당 적용 범위 내에 포함되지 않거나, 이 라이선스에 따라 특별히 부여된 권리 중 하나 이상의 금지사항이 적용되지 않는 경우, "주요"입니다. 귀하가 소프트웨어 배포, 작업 전달의 정도에 따라 제3자에게 제3자에게 결제를 하는 제3자와의 계약에 당사자인 경우, 그리고 제3자가 귀하로부터 보장된 작업을 수령한 당사자, 귀하로부터 업무를 수령한 당사자, 제3자로부터의 사본과 관련하여 제3자가 제공한 제3자와의 계약인 경우, (a) 귀하가 전달한 사본과 관련하여 해당 문서를 수령할 수 없습니다. 또는 (b) 주로 2007년 3월 28일 이전에 해당 배열에 입력했거나, 저작권 라이선스가 부여되지 않는 한, 적용 대상 작업이 포함된 특정 제품 또는 컴파일에 대해 및 관련하여 제공됩니다.

본 라이선스의 어떤 내용도 해당 권리법에서 귀하에게 제공될 수 있는 저지한 라이선스 또는 기타 방어를 제외하거나 제한한 것으로 해석되지 않습니다.

## 12. 다른 사람의 자유에 대한 의심은 없습니다.

이 라이선스 조건을 상충하는 명령, 계약 또는 기타 조건에 따라 귀하에게 적용되는 조건이 있는 경우, 이 라이선스의 조건에서는 귀하에게 불명확하지 않습니다. 이 라이선스 및 기타 관련 의무에 따라 동시에 귀하의 의무를 이행하기 위해 보장된 작업을 전달할 수 없는 경우, 결과적으로 이를 전혀 전달하지 못할 수 있습니다. 예를 들어, 귀하가 귀하가 프로그램을 전달하는 자로부터 더 많은 전달을 할 것을 의무하는 조건에 동의할 경우, 해당 약관을 모두 충족할 수 있는 유일한 방법은 해당 약관을 모두 충족할 수 있는 유일한 방법은 프로그램 전달을 완전히 거부하는 것입니다.

## 13. GNU Affero General Public License와 함께 사용합니다.

이 라이선스의 다른 조항에도 불구하고, 귀하는 GNU Affero General Public License의 버전 3에 따라 라이선스가 부여된 작업과 링크하거나 결합할 수 있으며 결과적인 작업을 전달할 수 있는 권한이 있습니다. 이 라이선스의 조건은 적용되는 부분에도 계속 적용되지만 네트워크를 통한 상호 작용에 관한 GNU Affero General Public License 섹션 13의 특수 요구 사항은 다음과 같은 조합에 적용됩니다.

## 14. 이 라이선스의 버전을 수정합니다.

Free Software Foundation은 때때로 GNU General Public License 개정 및/또는 새로운 버전을 게시할 수 있습니다. 이러한 새로운 버전은 현재 버전과 관련이 있지만 새로운 문제 또는 우려 사항을 해결하기 위해 자세히 다룰 수 있습니다.

각 버전에는 고유 버전 번호가 제공됩니다. 프로그램이 GNU General Public License "또는 이후 버전"이 적용되도록 지정하는 경우, 귀하는 그 번호가 매겨진 버전 또는 Free Software Foundation이 게시한 이후 버전의 약관을 따르는 옵션을 사용할 수 있습니다. 프로그램이 GNU General Public License의 버전 번호를 지정하지 않으면 Free Software Foundation에서 게시한 버전을 선택할 수 있습니다.

If the program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the program.

이후 라이선스 버전에서는 추가 또는 다른 권한을 제공할 수 있습니다. 그러나 이후 버전을 따르기로 선택한 결과 작성자 또는 저작권 소유자에 대한 추가 의무는 적용되지 않습니다.

### 15. 보장에 대한 면책 조항.

해당 법률에서 허용하는 범위 내에서 프로그램에 대한 보장은 없습니다. 저작권 소유자 및/또는 기타 당사자를 서면으로 달리 명시된 경우를 제외하고, 다음을 포함하여 어떤 종류의 보증도 하지 않고 "있는 그대로" 프로그램을 제공합니다. 그러나 이에 국한되지는 않으며 특정 목적을 위해 MERCHANTABILITY 및 적합성을 보장합니다. 프로그램의 품질과 성능에 대한 전체 위험은 귀하와 함께 있습니다. 프로그램이 결함이 있음을 증명하면 필요한 모든 서비스, 수리 또는 수정 비용을 가정합니다.

### 16. 가능성에 대한 제한입니다.

관련 법령에 의해 요구되거나 서면으로 동의한 경우를 제외하고, 어떠한 경우에도 저작권 소유자, 또는 위에서 허용되는 대로 프로그램을 수정 및/또는 전달하는 기타 당사자, 프로그램을 사용 또는 사용할 수 없는 일반적인, 특별한, 부수적 또는 결과적 손상을 포함하여 귀하가 해당 프로그램을 사용할 수 없도록 할 책임이 있습니다. (단, 포함) 귀하 또는 제3자에 의해 유지되거나 다른 프로그램과 함께 작동하는 프로그램의 실패에 의해 발생하는 부정확하거나 손실되는 데이터 또는 데이터의 손실에 국한되지 않음) 이러한 소유자 또는 다른 당사자가 이러한 손상의 가능성을 권고한 경우에도 마찬가지입니다.

### 17. 섹션 15 및 16의 해석입니다.

상기에 기재된 보증 및 제한에 대한 보증 및 제한 사항이 해당 조건에 따라 현지 법적 영향을 미칠 수 없는 경우, 검토관은 해당 프로그램과 관련하여 모든 민사성의 절대적인 포기를 가장 근접하게 추정하는 현지 법률이 적용됩니다.

이용 약관 종료

#### 이 약관을 새 프로그램에 적용하는 방법

새로운 프로그램을 개발하고 대중에게 최대한 많이 사용하기를 원한다면, 이를 달성하는 가장 좋은 방법은 모든 사람이 이러한 조건에 따라 재배포하고 변경할 수 있는 무료 소프트웨어를 만드는 것입니다.

이렇게 하려면 다음 알림을 프로그램에 첨부합니다. 각 소스 파일의 시작에 가장 효과적으로 보증의 제의를 명시하기 위해 이를 첨부하는 것이 가장 안전합니다. 각 파일에는 최소한 "복사" 라인과 전체 알림이 있는 위치에 대한 포인터가 있어야 합니다.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.
```

또한 전자 및 서류 우편으로 귀하에게 연락하는 방법에 대한 정보를 추가하십시오.

프로그램이 터미널 상호 작용을 수행하는 경우 대화형 모드에서 시작될 때 다음과 같이 짧은 알림을 출력합니다.

```
<program> Copyright (C) <year> <name of author>  
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.  
This is free software, and you are welcome to redistribute it  
under certain conditions; type `show c' for details.
```

'w' 및 'show c'라는 예상 명령은 일반 공개 라이선스의 적절한 부분을 표시해야 합니다. 물론 프로그램의 명령이 다를 수 있습니다. GUI 인터페이스의 경우 "정보 상자"를 사용합니다.

또한 필요한 경우 고용주를 받아야 합니다 (프로그래머로 일한 경우, 필요한 경우 프로그램에 대한 "복지권 면책 조항"에 서명할 수 있습니다. 이에 대한 자세한 내용과 GNU GPL을 적용하고 따르는 방법은 <<https://www.gnu.org/licenses/>> 을 참조하십시오.

GNU General Public License는 귀하의 프로그램을 독점 프로그램에 통합하는 것을 허용하지 않습니다. 귀하의 프로그램이 서브루틴 라이브러리인 경우, 독점적인 애플리케이션을 라이브러리와 연결할 수 있도록 허용하는 것이 더 유용할 수 있습니다. 이 경우 이 라이선스 대신 GNU Lesser General Public License를 사용하십시오. 그러나 먼저 <<https://www.gnu.org/licenses/why-not-lgpl.html>>를 읽어 주십시오.