

# FTD 인터페이스 IP를 업데이트할 수 있도록 FMC 구성

## 목차

---

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[구성](#)

[네트워크 다이어그램](#)

[설정](#)

[다음을 확인합니다.](#)

[문제 해결](#)

[관련 정보](#)

---

## 소개

이 문서에서는 FMC(Firepower Management Center)를 자동화하여 FTD(Firepower Threat Defense) 인터페이스 IP를 Ansible과 함께 구성하는 단계를 설명합니다.

## 사전 요구 사항

### 요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- 앤서블
- Ubuntu 서버
- Cisco FMC(Firepower 관리 센터) 가상
- Cisco FTD(Firepower Threat Defense) 가상

이러한 실험실 상황에서 Ansible은 Ubuntu에 구축됩니다.

이 문서에서 참조하는 Ansible 명령을 실행하기 위해 Ansible이 지원하는 모든 플랫폼에 Ansible이 성공적으로 설치되도록 하는 것이 중요합니다.

### 사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- Ubuntu Server 22.04

- Ansible 2.10.8
- 파이썬 3.10
- Cisco Firepower Threat Defense Virtual 7.4.1
- Cisco Firepower Management Center Virtual 7.4.1

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

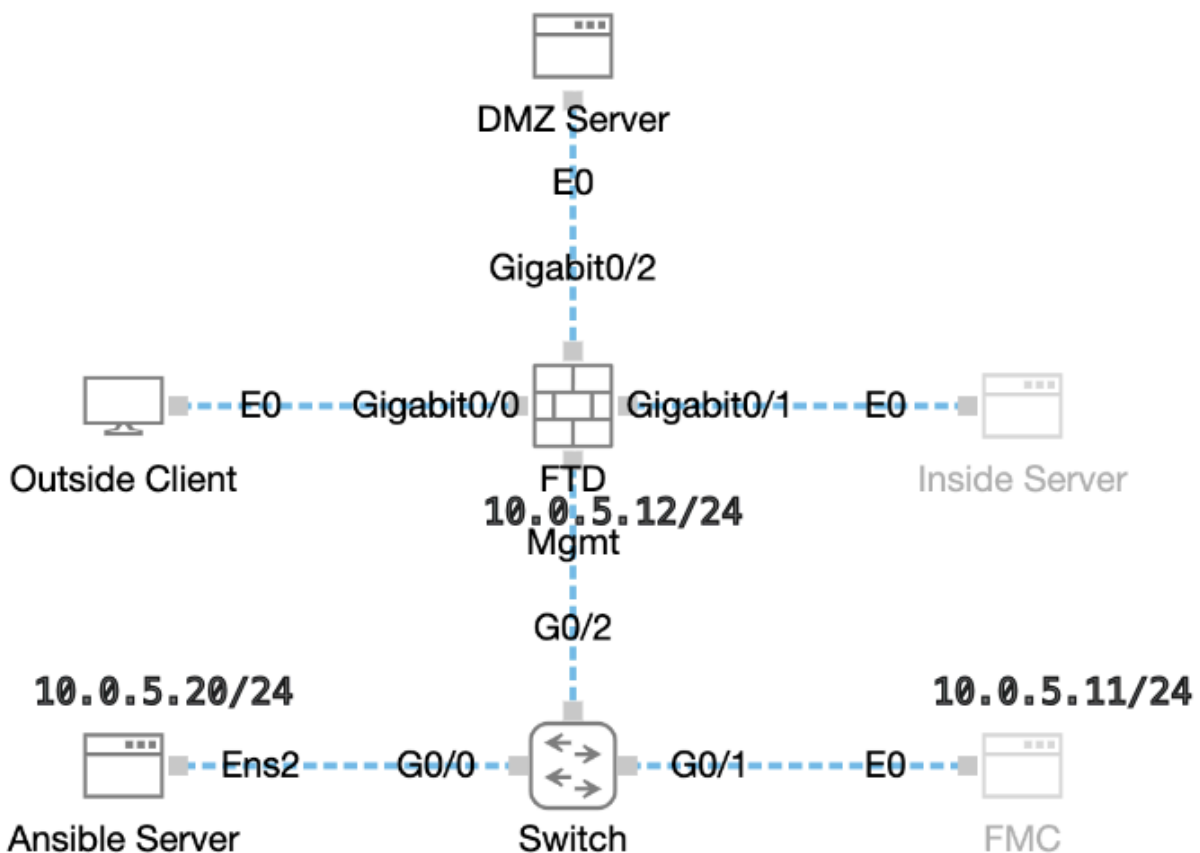
## 배경 정보

Ansible은 다양한 기능을 갖춘 툴로서 네트워크 디바이스 관리에 상당한 효율성을 입증했습니다. Ansible을 사용하여 자동화된 작업을 실행하기 위해 수많은 방법론이 사용될 수 있습니다. 이 글에 채용된 방법은 시험 목적의 참조로서 사용된다.

이 예에서는 플레이북 예제를 성공적으로 실행한 후 인터페이스 ip 주소, 마스크 및 인터페이스 이름이 FTD로 업데이트됩니다.

## 구성

### 네트워크 다이어그램



## 설정

Cisco는 예제 스크립트나 고객 작성 스크립트를 지원하지 않으므로, 요구 사항에 따라 테스트할 수 있는 몇 가지 예제가 있습니다.

사전 검증이 적법하게 마무리되도록 하는 것이 필수적이다.

- Ansible 서버는 인터넷 연결을 보유하고 있습니다.
- Ansible 서버는 FMC GUI 포트와 성공적으로 통신할 수 있습니다(FMC GUI의 기본 포트는 443).
- FTD가 FMC에 성공적으로 등록되었습니다.

1단계. SSH 또는 콘솔을 통해 Ansible 서버의 CLI에 연결합니다.

2단계. Ansible 서버 `ansible-galaxy collection install cisco.fmcansible`에 FMC의 Ansible 컬렉션을 설치하려면 명령을 실행합니다.

<#root>

```
cisco@inserthostname-here:~$
```

```
ansible-galaxy collection install cisco.fmcansible
```

3단계. 관련 파일 `mkdir /home/cisco/fmc_ansible`을 저장할 새 폴더를 만들려면 명령을 실행합니다. 이 예에서 홈 디렉토리는 `/home/cisco/`이고 새 폴더 이름은 `fmc_ansible`입니다.

<#root>

```
cisco@inserthostname-here:~$
```

```
mkdir /home/cisco/fmc_ansible
```

4단계. 폴더 `/home/cisco/fmc_ansible`로 이동하여 인벤토리 파일을 생성합니다. 이 예에서 인벤토리 파일 이름은 `inventory.ini`입니다.

<#root>

```
cisco@inserthostname-here:~$
```

```
cd /home/cisco/fmc_ansible/
```

```
ccisco@inserthostname-here:~/fmc_ansible$
```

```
ls
```

```
inventory.ini
```

이 내용을 복제한 다음 활용을 위해 붙여 넣어 강조 표시된 섹션을 정확한 매개변수로 변경할 수 있습니다.

```
<#root>
```

```
[fmc]
```

```
10.0.5.11
```

```
[fmc:vars]
```

```
ansible_user=
```

```
cisco
```

```
ansible_password=
```

```
cisco
```

```
ansible_httpapi_port=443
```

```
ansible_httpapi_use_ssl=True
```

```
ansible_httpapi_validate_certs=False
```

```
network_type=HOST
```

```
ansible_network_os=cisco.fmcansible.fmc
```

5단계. 폴더/홈/cisco/fmc\_ansible로 이동하여 변수 파일을 생성합니다. 이 예에서 변수 파일 이름은 fmc-configure-interface-vars.yml입니다.

```
<#root>
```

```
cisco@inserthostname-here:~$
```

```
cd /home/cisco/fmc_ansible/
```

```
ccisco@inserthostname-here:~/fmc_ansible$
```

```
ls
```

```
fmc-configure-interface-vars.yml
```

```
inventory.ini
```

이 내용을 복제한 다음 사용을 위해 붙여 넣어 강조 표시된 섹션을 정확한 매개변수로 변경할 수 있습니다.

```
<#root>
```

```
user: domain: 'Global' onboard: acp_name: 'TEMPACP' device_name: ftd1: 'FTDA' ftd_data: outside_name: '
```

```
Outside
```

```
' inside_name: '
```

Inside

```
' dmz_name: '
```

DMZ

```
' outside_ip: '
```

```
10.1.1.1
```

```
' inside_ip: '
```

```
10.1.2.1
```

```
' dmz_ip: '
```

```
10.1.3.1
```

```
' mask24: '
```

```
255.255.255.0
```

```
'
```

6단계. 폴더/home/cisco/fmc\_ansible로 이동하여 플레이북 파일을 생성합니다. 이 예에서 플레이북 파일 이름은 fmc-configure-interface-playbook.yaml입니다.

<#root>

```
cisco@inserthostname-here:~$
```

```
cd /home/cisco/fmc_ansible/
```

```
ccisco@inserthostname-here:~/fmc_ansible$
```

```
ls
```

```
fmc-configure-interface-playbook.yaml
```

```
fmc-configure-interface-vars.yml inventory.ini
```

이 콘텐츠를 복제한 후 활용을 위해 붙여 넣을 수 있으며, 강조 표시된 섹션을 정확한 매개변수로 변경할 수 있습니다.

<#root>

```
--- - name: Update FTD Interface IP Address hosts: fmc connection: httpapi tasks: - name: Task01 - Get User Domain cisco.fmcansible.fmc_configuration
```

```
user.domain
```

```
  }}" register_as: domain - name: Task02 - Get Devices cisco.fmcansible.fmc_configuration: operation: get
```

```
device_name.ftd1
```

```
  }}" register_as: device_list - name: Task03 - Get Physical Interfaces cisco.fmcansible.fmc_configuration
```

```
ftd_data.outside_name
```

```
    }}" ipv4: static: address: "{{ Outside_ip | default(
ftd_data.outside_ip
) }}" netmask: "{{ Outside_netmask | default(
ftd_data.mask24
) }}" MTU: 1500 enabled: True mode: NONE type: physicalinterface name:
GigabitEthernet0/0
  path_params: domainUUID: '{{ domain[0].uuid }}' containerUUID: '{{ device_list[0].id }}' objectId: '{{
ftd_data.inside_name
  }}" ipv4: static: address: "{{ Inside_ip | default(
ftd_data.inside_ip)
  }}" netmask: "{{ Inside_netmask | default(
ftd_data.mask24
) }}" MTU: 1500 enabled: True mode: NONE type: physicalinterface name:
GigabitEthernet0/1
  path_params: domainUUID: '{{ domain[0].uuid }}' containerUUID: '{{ device_list[0].id }}' objectId: '{{
ftd_data.dmz_name
  }}" ipv4: static: address: "{{ DMZ_ip | default(
ftd_data.dmz_ip
) }}" netmask: "{{ DMZ_netmask | default(
ftd_data.mask24
) }}" MTU: 1500 enabled: True mode: NONE type: physicalinterface name:
GigabitEthernet0/2
  path_params: domainUUID: '{{ domain[0].uuid }}' containerUUID: '{{ device_list[0].id }}' objectId: '{{
```

---

참고: 이 예제 플레이북에서 강조 표시된 이름은 변수로 사용됩니다. 이러한 변수에 대한 해당 값은 변수 파일 내에 보존됩니다.

---

7단계. 폴더/홈/cisco/fmc\_ansible로 이동하여 명령ansible-playbook -i <inventory\_name>.ini <playbook\_name>.yaml -e@"<playbook\_vars>.yaml"을 실행하여 ansible 작업을 재생합니다.

이 예에서 명령은 입니다ansible-playbook -i inventory.ini fmc-configure-interface-playbook.yaml -e@"fmc-configure-interface-vars.yaml".

<#root>

cisco@inserthostname-here:~\$

cd /home/cisco/fmc\_ansible/

cisco@inserthostname-here:~/fmc\_ansible\$

ls

fmc-configure-interface-playbook.yaml fmc-configure-interface-vars.yml inventory.ini

cisco@inserthostname-here:~/fmc\_ansible\$

ansible-playbook -i inventory.ini fmc-configure-interface-playbook.yaml -e@"fmc-configure-interface-vars"

PLAY [Update FTD Interface IP Address] \*\*\*\*\*

TASK [Gathering Facts] \*\*\*\*\*  
ok: [10.0.5.11]

TASK [Task01 - Get User Domain] \*\*\*\*\*  
ok: [10.0.5.11]

TASK [Task02 - Get Devices] \*\*\*\*\*  
ok: [10.0.5.11]

TASK [Task03 - Get Physical Interfaces] \*\*\*\*\*  
ok: [10.0.5.11]

TASK [Task04 - Setup Outside Interface with static IP] \*\*\*\*\*  
changed: [10.0.5.11]

TASK [Task05 - Setup Inside Interface with static IP] \*\*\*\*\*  
changed: [10.0.5.11]

TASK [Task06 - Setup DMZ Interface with static] \*\*\*\*\*  
changed: [10.0.5.11]

TASK [Task07 - Get Deployable Devices] \*\*\*\*\*  
ok: [10.0.5.11]

TASK [Task08 - Start Deployment] \*\*\*\*\*  
changed: [10.0.5.11]

TASK [Wait for Deployment Complete] \*\*\*\*\*  
ok: [10.0.5.11]

TASK [Task09 - Poll Deployment Status Until Deployment Successful] \*\*\*\*\*  
ok: [10.0.5.11]

TASK [Task10 - Stop The Playbook If The Deployment Failed] \*\*\*\*\*  
skipping: [10.0.5.11]

PLAY RECAP \*\*\*\*\*  
10.0.5.11 : ok=11 changed=4 unreachable=0 failed=0 skipped=1 rescued=0 ignored=0

다음을 확인합니다.



설정이 올바르게 작동하는지 확인하려면 이 섹션을 활용하십시오.

SSH 또는 콘솔을 통해 FTD의 CLI에 연결하고 명령 및 `show interface ip briefshow running-config interface GigabitEthernet 0/X` 실행합니다.

인터페이스 이름, ip 주소 및 마스크가 성공적으로 구성되었습니다.

**<#root>**

```
> show interface ip brief
```

```
Interface IP-Address OK? Method Status Protocol
```

```
GigabitEthernet0/0 10.1.1.1
```

```
YES manual
```

```
up up
```

```
GigabitEthernet0/1 10.1.2.1
```

```
YES manual
```

```
up up
```

```
GigabitEthernet0/2 10.1.3.1
```

```
YES manual
```

```
up up
```

```
>
```

```
show running-config interface GigabitEthernet 0/0
```

```
!  
interface GigabitEthernet0/0  
nameif
```

```
Outside
```

```
cts manual  
propagate sgt preserve-untag  
policy static sgt disabled trusted  
security-level 0
```

```
ip address 10.1.1.1 255.255.255.0
```

```
>
```

```
show running-config interface GigabitEthernet 0/1
```

```
!
```

```
interface GigabitEthernet0/1
nameif

Inside

cts manual
propagate sgt preserve-untag
policy static sgt disabled trusted
security-level 0

ip address 10.1.2.1 255.255.255.0
```

>

```
show running-config interface GigabitEthernet 0/2
```

```
!
interface GigabitEthernet0/2
nameif

DMZ

cts manual
propagate sgt preserve-untag
policy static sgt disabled trusted
security-level 0
ip address 10.1.3.1 255.255.255.0
```

## 문제 해결

이 섹션에서는 설정 문제 해결에 사용할 수 있는 정보를 제공합니다.

Ansible 플레이북의 로그를 더 많이 보려면 `-vv`를 사용하여 ansible 플레이북을 실행할 수 있습니다.

```
cisco@inserthostname-here:~/fmc_ansible$ ansible-playbook -i inventory.ini fmc-configure-interface-playbook.yaml -e@"fmc-configure-interface-vars.yml"
```

## 관련 정보

[Cisco Devnet FMC Ansible](#)

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.