Ansible to Onboard FTD로 FMC 구성

목차

소개

이 문서에서는 Ansible을 사용하여 FMC(Firepower Management Center)에 대한 FTD(Firepower Threat Defense) 등록을 자동화하는 단계에 대해 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.

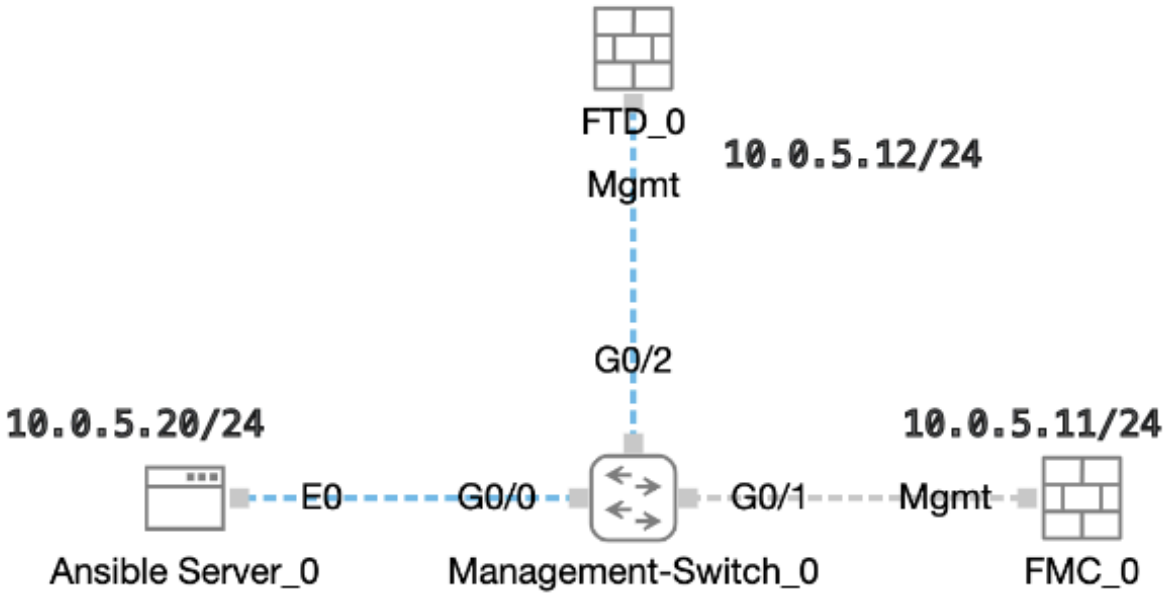
배경 정보

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 가상 FTD를 성공적으로 온보딩한 후 기본 라이센스, 라우팅 모드, 기능 계층 FTDv30, FMC로 보내는 로그를 활성화한 기본 허용 작업과 함께 액세스 제어 정책을 사용합니다.

구성

네트워크 다이어그램



토폴로지

설정

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

사전 검증이 적법하게 완료되었는지 확인하는 것이 필수적이다.

- Ansible 서버는 인터넷 연결을 보유하고 있습니다.
- Ansible 서버는 FMC GUI 포트와 성공적으로 통신할 수 있습니다(FMC GUI의 기본 포트는 443).
- FTD는 올바른 관리자 ip 주소, 레지스터 키 및 nat-id로 구성됩니다.
- FMC가 Smart License로 활성화되었습니다.

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

2단계. Ansible 서버ansible-galaxy collection install cisco.fmcansible에 FMC의 Ansible 컬렉션을 설치하려면 명령을 실행합니다.

<#root>

cisco@inserthostname-here:~$

ansible-galaxy collection install cisco.fmcansible

3단계. 관련 파일mkdir /home/cisco/fmc_ansible을 저장할 새 폴더를 만들려면 명령을 실행합니다. 이 예에서 홈 디렉토리는 /home/cisco/이고 새 폴더 이름은 fmc_ansible입니다.

<#root>

cisco@inserthostname-here:~$

mkdir /home/cisco/fmc_ansible

4단계. /home/cisco/fmc_ansible 폴더로 이동하여 인벤토리 파일을 생성합니다. 이 예에서 인벤토리 파일 이름은 inventory.ini입니다.

<#root>

cisco@inserthostname-here:~$

cd /home/cisco/fmc_ansible/

ccisco@inserthostname-here:~/fmc_ansible$

ls

inventory.ini

다음 내용을 복제하여 활용을 위해 붙여 넣을 수 있으며, 강조 표시된 섹션을 정확한 **매개변수**로 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. /home/cisco/fmc_ansible 폴더로 이동하여 변수 파일을 생성합니다. 이 예에서 변수 파일 이름은 fmc-onboard-ftd-vars.yml입니다

.

<#root>

cisco@inserthostname-here:~$

cd /home/cisco/fmc_ansible/

ccisco@inserthostname-here:~/fmc_ansible$

ls

fmc-onboard-ftd-vars.yml

inventory.ini

다음 내용을 복제하여 활용을 위해 붙여 넣을 수 있으며, 강조 표시된 섹션을 정확한 **매개변수**로 변경할 수 있습니다.

<#root>

user:
domain: 'Global'
onboard:
acp_name: '

TEMPACP

'
device_name:
ftd1: '

FTDA

'
 ftd1_reg_key: '

cisco

'
 ftd1_nat_id: '

natcisco

'
mgmt:
 ftd1: '

10.0.5.12

'

6단계 /home/cisco/fmc_ansible 폴더로 이동하여 플레이북 파일을 만듭니다. 이 예에서 플레이북 파일 이름은 fmc-onboard-ftd-playbook.yaml입니다.

<#root>

cisco@inserthostname-here:~$

 cd /home/cisco/fmc_ansible/

ccisco@inserthostname-here:~/fmc_ansible$

ls

fmc-onboard-ftd-playbook.yaml

fmc-onboard-ftd-vars.yml inventory.ini

다음 내용을 복제하여 활용을 위해 붙여 넣을 수 있으며, 강조 표시된 섹션을 정확한 **매개변수**로 변경할 수 있습니다.

<#root>

- name: FMC Onboard FTD
 hosts: fmc
 connection: httpapi

 tasks:

 - name: Task01 - Get User Domain
 cisco.fmcansible.fmc_configuration:
 operation: getAllDomain
 filters:
 name: "{{

user.domain

```
 }}"
 register_as: domain

 - name: Task02 - Create ACP TEMP_ACP
 cisco.fmcansible.fmc_configuration:
 operation: "createAccessPolicy"
 data:
 type: "AccessPolicy"
 name: "{{accesspolicy_name | default(
```

onboard.acp_name

```
) }}"
 defaultAction: {
 'action': 'PERMIT',
 'logEnd': True,
 'logBegin': False,
 'sendEventsToFMC': True
 }
 path_params:
 domainUUID: "{{ domain[0].uuid }}"

 - name: Task03 - Get Access Policy
 cisco.fmcansible.fmc_configuration:
 operation: getAllAccessPolicy
 path_params:
 domainUUID: "{{ domain[0].uuid }}"
 filters:
 name: "{{
```

onboard.acp_name

```
 }}"
 register_as: access_policy

 - name: Task04 - Add New FTD1
 cisco.fmcansible.fmc_configuration:
 operation: createMultipleDevice
 data:
 hostName: "{{ ftd_ip | default(item.key) }}"
 license_caps:
 - 'BASE'
 ftdMode: 'ROUTED'
 type: Device
 regKey: "{{ reg_key | default(
```

device_name.ftd1_reg_key

```
) }}"
 performanceTier: "FTDv30"
 name: "{{ ftd_name | default(item.value) }}"
 accessPolicy:
 id: '{{ access_policy[0].id }}'
 type: 'AccessPolicy'
 natID: "{{ nat_id | default(
```

device_name.ftd1_nat_id

```
) }}"
 path_params:
 domainUUID: '{{ domain[0].uuid }}'
 loop: "{{ ftd_ip_name | dict2items }}"
 vars:
 ftd_ip_name:
```

```
 "{{
```

mgmt.ftd1

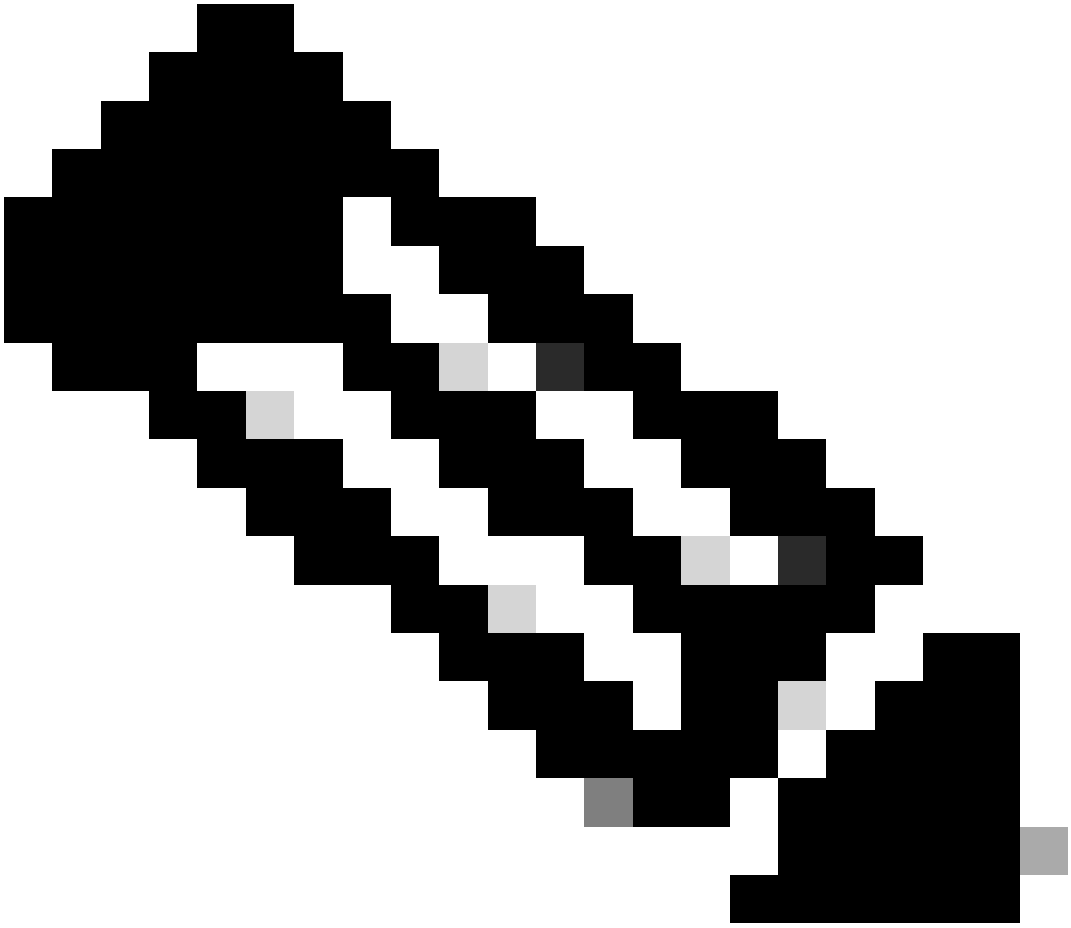
```
}}": "{{
```

device_name.ftd1

```
}}"

 - name: Task05 - Wait For FTD Registration Completion
 ansible.builtin.wait_for:
 timeout: 120
 delegate_to: localhost


 - name: Task06 - Confirm FTD Init Deploy Complete
 cisco.fmcansible.fmc_configuration:
 operation: getAllDevice
 path_params:
 domainUUID: '{{ domain[0].uuid }}'
 query_params:
 expanded: true
 filters:
 name: "{{
```

device_name.ftd1

```
 }}"
 register_as: device_list
 until: device_list[0].deploymentStatus is match("DEPLOYED")
 retries: 1000
 delay: 3
```



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

7단계. /home/cisco/fmc_ansible 폴더로 이동한 다음 명령을 실행하여 **ansible-playbook -i <inventory_name>.ini <playbook_name>.yaml -e@"<playbook_vars>.yml"** ansible 작업을 재생합니다. 이 예에서 명령은 입니다ansible-playbook -i inventory.ini fmc-onboard-ftd-playbook.yaml -e @"fmc-onboard-ftd-vars.yml" .

<#root>

cisco@inserthostname-here:~$

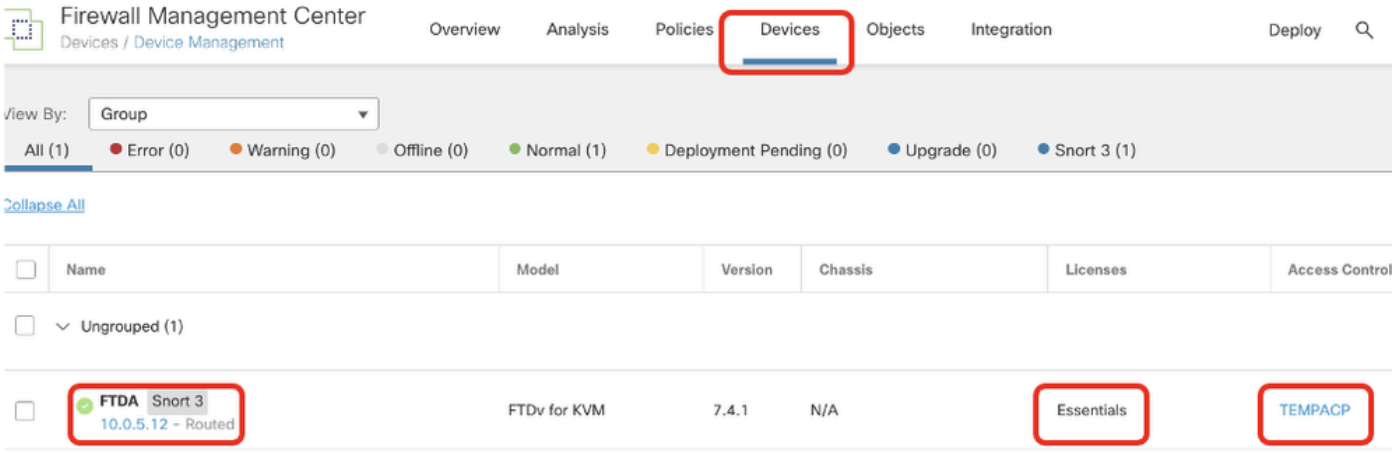
cd /home/cisco/fmc_ansible/

```
cisco@inserthostname-here:~/fmc_ansible$
ls

fmc-onboard-ftd-playbook.yaml fmc-onboard-ftd-vars.yml inventory.ini

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


PLAY [FMC Onboard FTD] *********************************************************************************

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

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

TASK [Task02 - Create ACP TEMP_ACP] ********************************************************************
changed: [10.0.5.11]

TASK [Task03 - Get Access Policy] **********************************************************************
ok: [10.0.5.11]

TASK [Task04 - Add New FTD1] ***************************************************************************
changed: [10.0.5.11] => (item={'key': '10.0.5.12', 'value': 'FTDA'})

TASK [Task05 - Wait For FTD Registration Completion] ***************************************************
ok: [10.0.5.11]

TASK [Task06 - Confirm FTD Init Deploy Complete] *******************************************************
FAILED - RETRYING: Task06 - Confirm FTD Init Deploy Complete (1000 retries left).
FAILED - RETRYING: Task06 - Confirm FTD Init Deploy Complete (999 retries left).
FAILED - RETRYING: Task06 - Confirm FTD Init Deploy Complete (998 retries left).
FAILED - RETRYING: Task06 - Confirm FTD Init Deploy Complete (997 retries left).
FAILED - RETRYING: Task06 - Confirm FTD Init Deploy Complete (996 retries left).
ok: [10.0.5.11]

PLAY RECAP *********************************************************************************************
10.0.5.11 : ok=7 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

다음을 확인합니다.

설정이 올바르게 작동하는지 확인하려면 이 섹션을 활용하십시오.

FMC GUI에 로그인합니다. Devices(**디바이스) > Device Management(디바이스 관리)**로 이동합니다. 이 경우 FTD는 구성된 액세스 제어 정책으로 FMC에 성공적으로 등록됩니다.



Device Management 페이지

문제 해결

이 섹션에서는 설정 문제 해결에 사용할 수 있는 정보를 제공합니다.

Ansible 플레이북의 로그를 더 보려면 -vv로 ansible 플레이북을 실행할 수 있습니다.

<#root>

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

-vvv

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.