

Configurar o FMC com Ansible para atualizar o IP da interface do FTD

Contents

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Configurar](#)

[Diagrama de Rede](#)

[Configurações](#)

[Verificar](#)

[Troubleshooting](#)

[Informações Relacionadas](#)

Introdução

Este documento descreve as etapas para automatizar o Firepower Management Center (FMC) para configurar o IP de interface do Firepower Threat Defense (FTD) com o Ansible.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Ansible
- Servidor Ubuntu
- Cisco Firepower Management Center (FMC) Virtual
- Cisco Firepower Threat Defense (FTD) Virtual

No contexto desta situação de laboratório, Ansible é implantado no Ubuntu.

É essencial garantir que o Ansible seja instalado com êxito em qualquer plataforma suportada pelo Ansible para executar os comandos Ansible referenciados neste artigo.

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- Servidor Ubuntu 22.04

- Ansible 2 10 8
- Python 3. 10
- Cisco Firepower Threat Defense Virtual 7.4.1
- Cisco Firepower Management Center Virtual 7.4.1

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

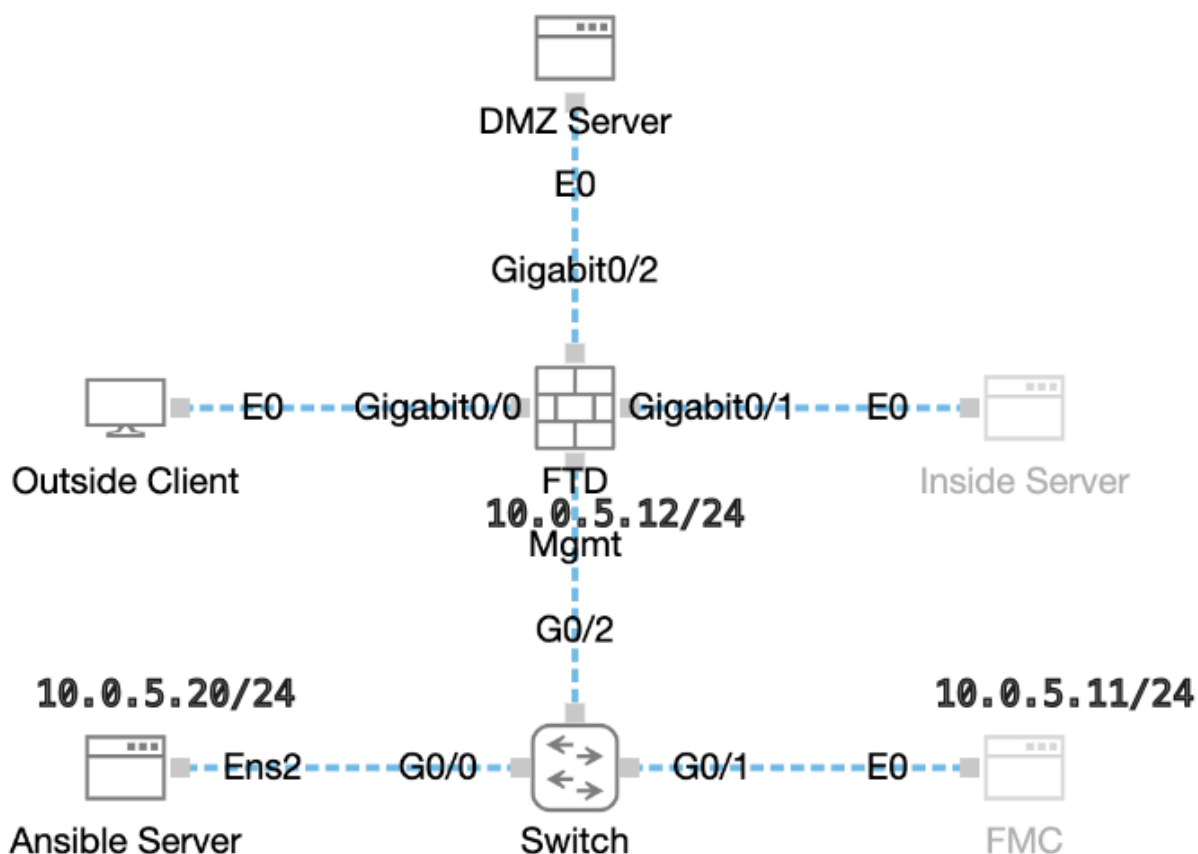
Informações de Apoio

O Ansible é uma ferramenta altamente versátil que demonstra uma eficiência significativa no gerenciamento de dispositivos de rede. Várias metodologias podem ser empregadas para executar tarefas automatizadas com a Ansible. O método utilizado neste artigo serve de referência para fins de teste.

Neste exemplo, o endereço ip da interface, a máscara e o nome da interface são atualizados para FTD após a execução bem-sucedida do exemplo do manual de atividades.

Configurar

Diagrama de Rede



Configurações

Como a Cisco não oferece suporte a scripts de exemplo ou scripts escritos por clientes, temos alguns exemplos que você pode testar de acordo com suas necessidades.

É essencial assegurar que a verificação preliminar foi devidamente concluída.

- Um servidor possível possui conectividade com a Internet.
- Um servidor Ansible pode se comunicar com êxito com a porta GUI do FMC (a porta padrão da GUI do FMC é 443).
- O DTF é registrado com êxito no CVP.

Etapa 1. Conecte-se ao CLI do servidor Ansible via SSH ou console.

Etapa 2. Execute o comando `ansible-galaxy collection install cisco.fmcansible` para instalar a coleção Ansible do FMC em seu servidor Ansible.

```
<#root>
```

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

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

Etapa 3. Execute o comando `mkdir /home/cisco/fmc_ansible` para criar uma nova pasta para armazenar os arquivos relacionados. Neste exemplo, o diretório inicial é `/home/cisco/`, o nome da nova pasta é `fmc_ansible`.

```
<#root>
```

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

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

Etapa 4. Navegue para a pasta `/home/cisco/fmc_ansible`, crie o arquivo de inventário. Neste exemplo, o nome do arquivo de inventário é `inventory.ini`.

```
<#root>
```

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

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

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

```
ls
```

```
inventory.ini
```

Você pode duplicar esse conteúdo e colá-lo para utilização, alterando as **seções destacadas** com os parâmetros precisos.

```
<#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
```

Etapa 5. Navegue até a pasta **/home/cisco/fmc_ansible**, criar arquivo de variável. Neste exemplo, o nome do arquivo de variável é **fmc-configure-interface-vars.yml**.

```
<#root>
```

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

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

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

```
ls
```

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

```
inventory.ini
```

Você pode duplicar esse conteúdo e colá-lo para utilização, alterando as **seções destacadas** com os parâmetros precisos.

```
<#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
```

```
'
```

Etapa 6. Navegue até a pasta **/home/cisco/fmc_ansible**, crie o arquivo de manual de atividades. Neste exemplo, o nome do arquivo de playbook é **fmc-configure-interface-playbook.yaml**.

```
<#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
```

Você pode duplicar esse conteúdo e colá-lo para utilização, alterando as seções **destacadas** com os parâmetros precisos.

```
<#root>
```

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

```
user.domain
```

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

device_name.ftd1

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

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

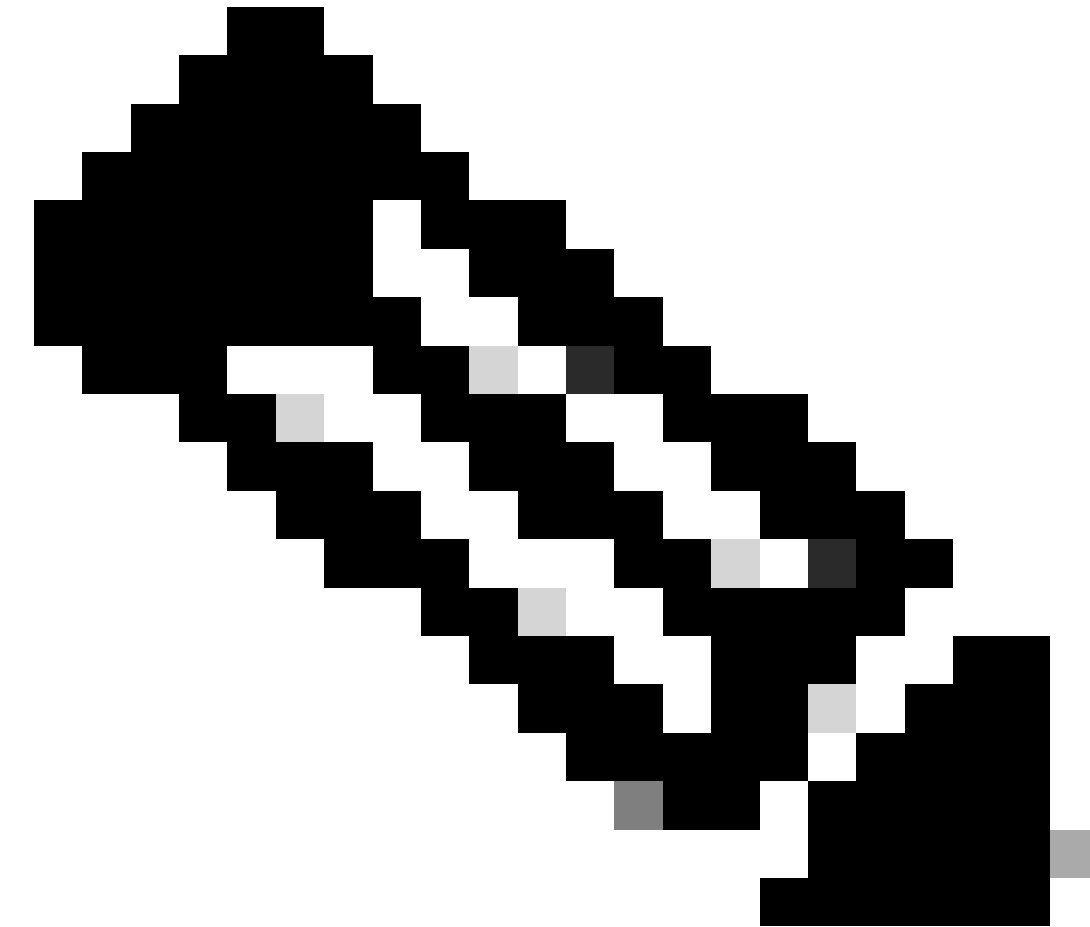
```
) }}" netmask: "{{ DMZ_netmask | default(
```

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: '{{
```



Observação: os nomes destacados neste manual de atividades de exemplo servem como variáveis. Os valores correspondentes para essas variáveis são preservados no arquivo de variáveis.

Passo 7. Navegue para a pasta **/home/cisco/fmc_ansible**, execute o comando `ansible-playbook -i <inventory_name>.ini <playbook_name>.yaml -e"<playbook_vars>.yaml"` para reproduzir a tarefa ansible.

Neste exemplo, o comando é `ansible-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
```

Verificar

Use esta seção para confirmar se a sua configuração funciona corretamente.

Conecte-se ao CLI do FTD via SSH ou console e execute os comandos `show interface ip brief` e `show running-config interface GigabitEthernet 0/X`.

O nome da interface, o endereço IP e a máscara são configurados com êxito.

```
<#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
```

Troubleshooting

Esta seção fornece informações que podem ser usadas para o troubleshooting da sua configuração.

Para ver mais registros de um manual de atividades possível, você pode executar um manual de atividades com o -vvv

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

Informações Relacionadas

[Cisco Devnet FMC Ansible](#)

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.