

Configurare FMC con Anable per aggiornare l'indirizzo IP dell'interfaccia FTD

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Premesse](#)

[Configurazione](#)

[Esempio di rete](#)

[Configurazioni](#)

[Verifica](#)

[Risoluzione dei problemi](#)

[Informazioni correlate](#)

Introduzione

In questo documento viene descritto come automatizzare Firepower Management Center (FMC) per configurare l'interfaccia IP di Firepower Threat Defense (FTD) con Ansible.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Ansioso
- Server Ubuntu
- Virtual Cisco Firepower Management Center (FMC)
- Virtual Cisco Firepower Threat Defense (FTD)

Nel contesto di questa situazione di laboratorio, Ansible è schierato su Ubuntu.

È essenziale assicurarsi che Ansible sia installato correttamente su qualsiasi piattaforma supportata da Ansible per l'esecuzione dei comandi Ansible a cui si fa riferimento in questo articolo.

Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Ubuntu Server 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

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

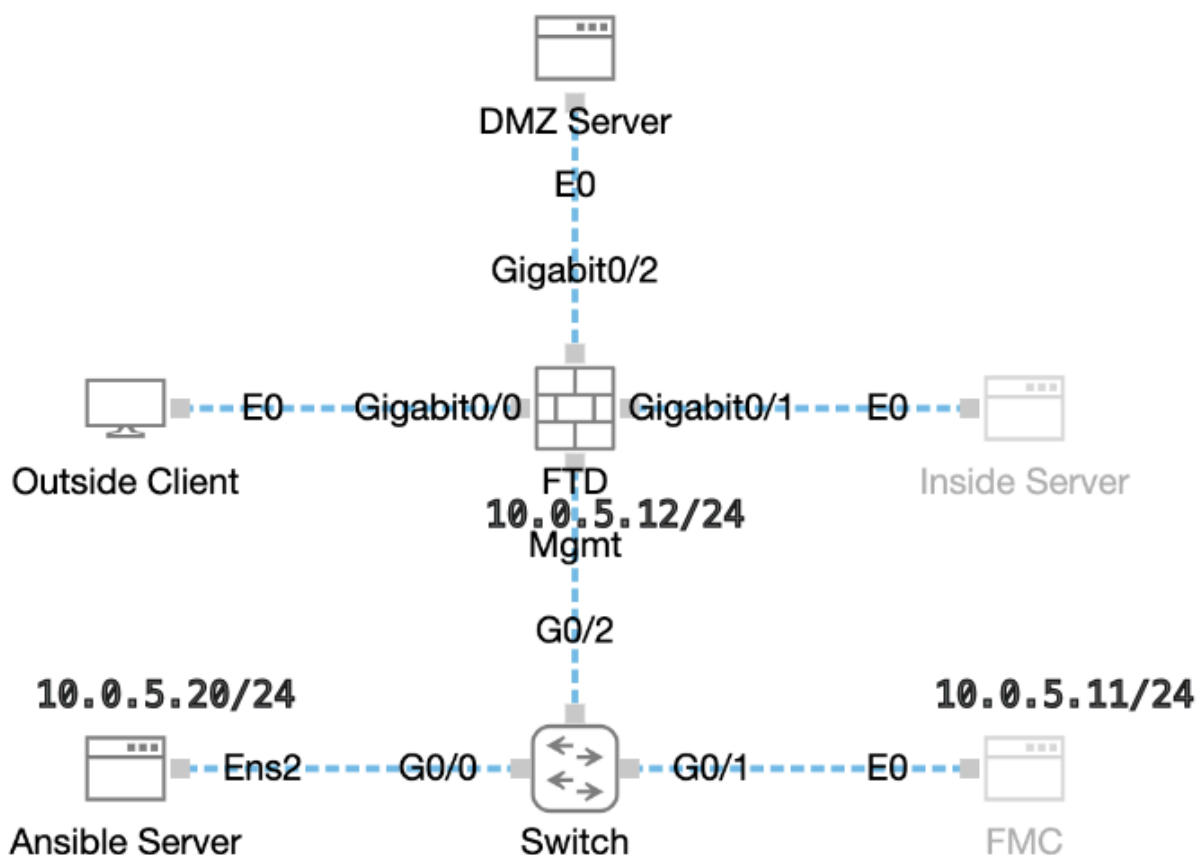
Premesse

Ansible è uno strumento estremamente versatile che dimostra una notevole efficacia nella gestione dei dispositivi di rete. Numerose metodologie possono essere impiegate per eseguire operazioni automatizzate con Ansible. Il metodo utilizzato in questo articolo serve da riferimento ai fini della prova.

In questo esempio, l'indirizzo IP dell'interfaccia, la maschera e il nome dell'interfaccia vengono aggiornati a FTD dopo la corretta esecuzione dell'esempio della playbook.

Configurazione

Esempio di rete



Configurazioni

Poiché Cisco non supporta gli script di esempio o quelli scritti dal cliente, sono disponibili alcuni esempi che è possibile verificare in base alle esigenze.

È essenziale garantire che la verifica preliminare sia stata debitamente completata.

- Il server ansible dispone di connettività Internet.
- Ansible Server è in grado di comunicare con la porta GUI FMC (la porta predefinita per l'interfaccia GUI FMC è 443).
- Registrazione dell'FTD nel CCP completata.

Passaggio 1. Connettersi alla CLI del server Ansible tramite SSH o console.

Passaggio 2. Eseguire il comando `ansible-galaxy collection install cisco.fmcansible` per installare la raccolta Ansible di FMC nel server Ansible.

```
<#root>
```

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

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

Passaggio 3. Eseguire il comando `mkdir /home/cisco/fmc_ansible` per creare una nuova cartella in cui archiviare i file correlati. In questo esempio, la home directory è `/home/cisco/`, il nome della nuova cartella è `fmc_ansible`.

```
<#root>
```

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

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

Passaggio 4. Passare alla cartella `/home/cisco/fmc_ansible`, creare il file di inventario. In questo esempio, il nome del file di inventario è `inventory.ini`.

```
<#root>
```

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

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

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

```
ls
```

```
inventory.ini
```

Potete duplicare questo contenuto e incollarlo per utilizzarlo, modificando le sezioni **evidenziate** con parametri accurati.

```
<#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
```

Passaggio 5. Passare alla cartella **/home/cisco/fmc_ansible**, crea file variabile. In questo esempio, il nome del file della variabile è **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
```

Potete duplicare questo contenuto e incollarlo per utilizzarlo, modificando le sezioni **evidenziate** con i parametri accurati.

```
<#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
```

```
'
```

Passaggio 6. Passare alla cartella **/home/cisco/fmc_ansible**, creare il file della playbook. In questo esempio, il nome del file della playbook è `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
```

Potete duplicare questo contenuto e incollarlo per utilizzarlo, modificando le sezioni **evidenziate** con parametri accurati.

```
<#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: '{{
```

Nota: i nomi evidenziati in questo esempio di playbook fungono da variabili. I valori corrispondenti per queste variabili vengono mantenuti nel file delle variabili.

Passaggio 7. Passare alla cartella `/home/cisco/fmc_ansible`, eseguire il comando `ansible-playbook -i <inventory_name>.ini <playbook_name>.yaml -e"<playbook_vars>.yaml"` per eseguire l'attività andabile.

Nell'esempio, il comando è `ansible-playbook -i inventory.ini fmc-configure-interface-playbook.yaml -e"fmcconfigureinterfacevars.yml"` .

`<#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
```

Verifica

Per verificare che la configurazione funzioni correttamente, consultare questa sezione.

Connettersi alla CLI del FTD tramite SSH o console ed eseguire i comandi show interface ip brief e show running-config interface GigabitEthernet 0/X.

Il nome dell'interfaccia, l'indirizzo IP e la maschera sono stati configurati correttamente.

```
<#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
```

Risoluzione dei problemi

In questa sezione vengono fornite informazioni utili per risolvere i problemi di configurazione.

Per visualizzare più registri di un playbook ansible, è possibile eseguire un playbook ansible con -vv

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

Informazioni correlate

[Cisco Devent FMC Ansible](#)

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l'accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).