Configuración de FMC con FTD de Ansible a Onboard

Contenido

Introducción

Este documento describe los pasos para automatizar el registro de Firepower Threat Defence (FTD) en Firepower Management Center (FMC) con Ansible.

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

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

En el contexto de esta situación de laboratorio, Ansible está desplegado en Ubuntu.

Es esencial asegurarse de que Ansible se instale correctamente en cualquier plataforma compatible con Ansible para ejecutar los comandos Ansible a los que se hace referencia en este artículo.

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y 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

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

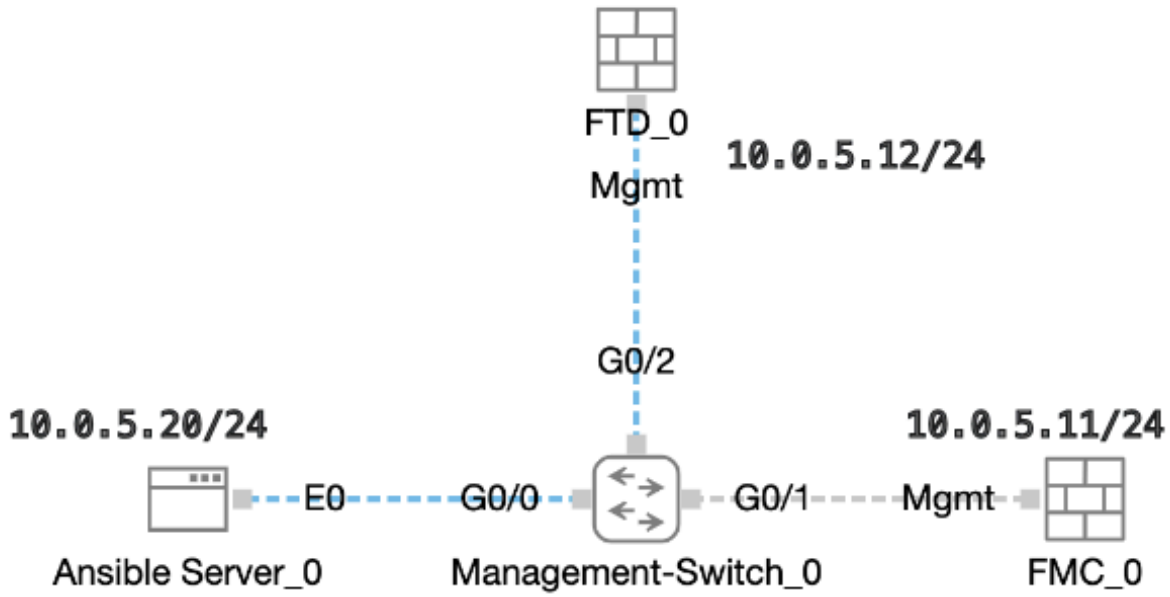
Antecedentes

Ansible es una herramienta muy versátil que demuestra una eficacia significativa en la gestión de dispositivos de red. Se pueden emplear numerosas metodologías para ejecutar tareas automatizadas con Ansible. El método empleado en este artículo sirve de referencia a efectos de ensayo.

En este ejemplo, después de incorporar correctamente el FTD virtual, se utiliza la licencia base, el modo enrutado, el nivel de función FTDv30 y la política de control de acceso, que es con la acción de permiso predeterminada con el envío a FMC habilitado para registro.

Configurar

Diagrama de la red



Topología

Configuraciones

Como Cisco no admite scripts de ejemplo ni scripts escritos por el cliente, tenemos algunos ejemplos que puede probar según sus necesidades.

Es esencial garantizar que la verificación preliminar se haya completado debidamente.

- El servidor Ansible posee conectividad a Internet.
- El servidor Ansible puede comunicarse correctamente con el puerto GUI de FMC (el puerto predeterminado para la GUI de FMC es 443).
- El FTD se configura con la dirección IP del administrador, la clave de registro y el nat-id correctos.
- El FMC se ha activado correctamente con la licencia inteligente.

Paso 1. Conéctese a la CLI del servidor Ansible mediante SSH o la consola.

Paso 2. Ejecute ansible-galaxy collection install cisco.fmcansible el comando para instalar la colección Ansible de FMC en su servidor Ansible.

<#root>

cisco@inserthostname-here:~$

ansible-galaxy collection install cisco.fmcansible

Paso 3. Ejecute mkdir /home/cisco/fmc_ansible el comando para crear una nueva carpeta para almacenar los archivos relacionados. En este ejemplo, el directorio principal es /home/cisco/, el nuevo nombre de carpeta es fmc_ansible.

<#root>

cisco@inserthostname-here:~$

mkdir /home/cisco/fmc_ansible

Paso 4. Vaya a la carpeta /home/cisco/fmc_ansible, crear archivo de inventario. En este ejemplo, el nombre del archivo de inventario es Inventory.ini.

<#root>

cisco@inserthostname-here:~$

cd /home/cisco/fmc_ansible/

ccisco@inserthostname-here:~/fmc_ansible$

ls

inventory.ini

Puede duplicar el siguiente contenido y pegarlo para su uso, alterando las secciones **resaltadas** con los parámetros 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

Paso 5. Vaya a la carpeta /home/cisco/fmc_ansible, crear archivo de variable. En este ejemplo, el nombre de archivo de la variable es fmc-onboard-ftd-vars.yml.

<#root>

cisco@inserthostname-here:~$

cd /home/cisco/fmc_ansible/

ccisco@inserthostname-here:~/fmc_ansible$

ls

fmc-onboard-ftd-vars.yml

inventory.ini

Puede duplicar el siguiente contenido y pegarlo para su uso, alterando las secciones **resaltadas** con los parámetros precisos.

<#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

'

Paso 6.Navegue hasta la carpeta /home/cisco/fmc_ansible, cree el archivo del cuaderno. En este ejemplo, el nombre del archivo del cuaderno es fmc-onboard-ftd-playbook.yaml.

<#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

Puede duplicar el siguiente contenido y pegarlo para su uso, alterando las secciones **resaltadas** con los parámetros precisos.

<#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

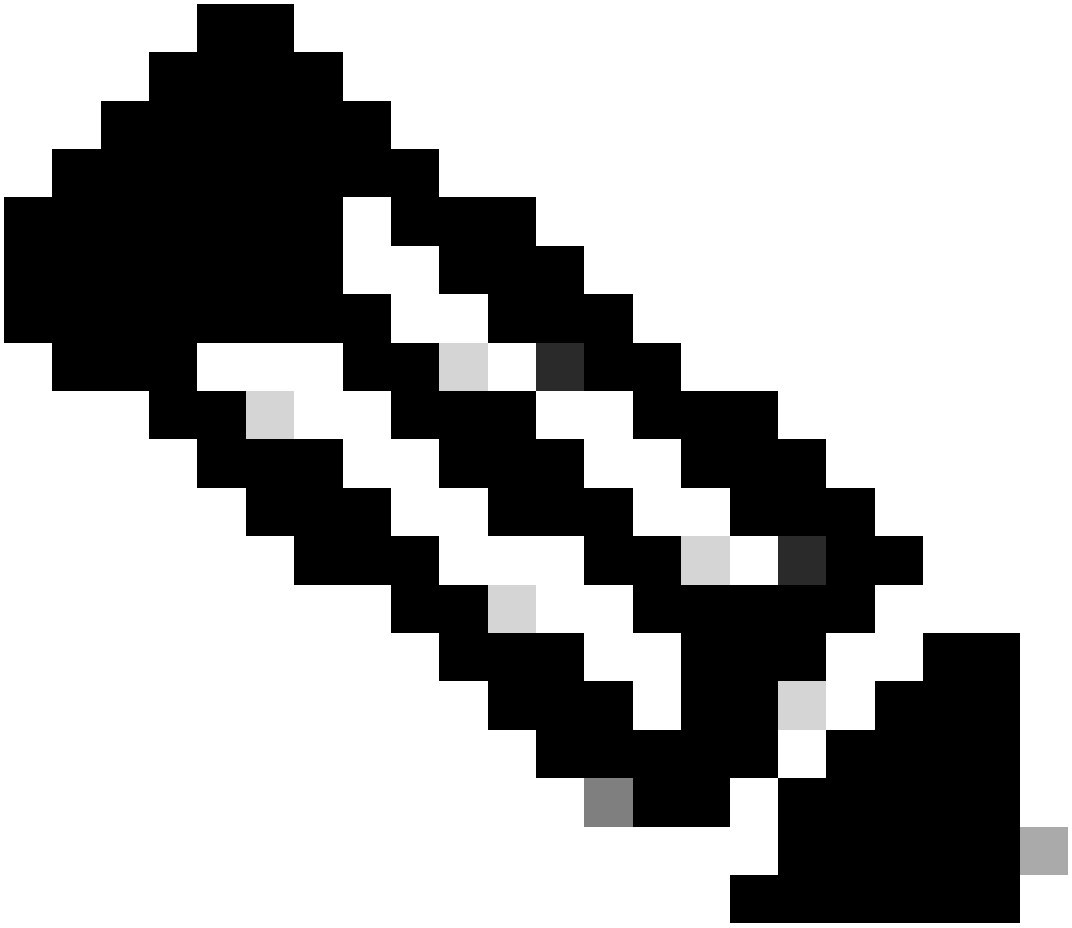
```
}}": "{{
```

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
```



Nota: Los nombres resaltados en este cuaderno de campaña de ejemplo sirven como variables. Los valores correspondientes de estas variables se conservan en el archivo de variables.

Paso 7. Navegue hasta la carpeta /home/cisco/fmc_ansible, ejecute **ansible-playbook -i <inventory_name>.ini <playbook_name>.yaml -e@"<playbook_vars>.yml"** el comando para reproducir la tarea ansible. En este ejemplo, el comando es ansible-playbook -i inventory.ini fmc-onboard-ftd-playbook.yaml -e @"fmc-onboard-ftd-vars.yml" .

<#root>

cisco@inserthostname-here:~$

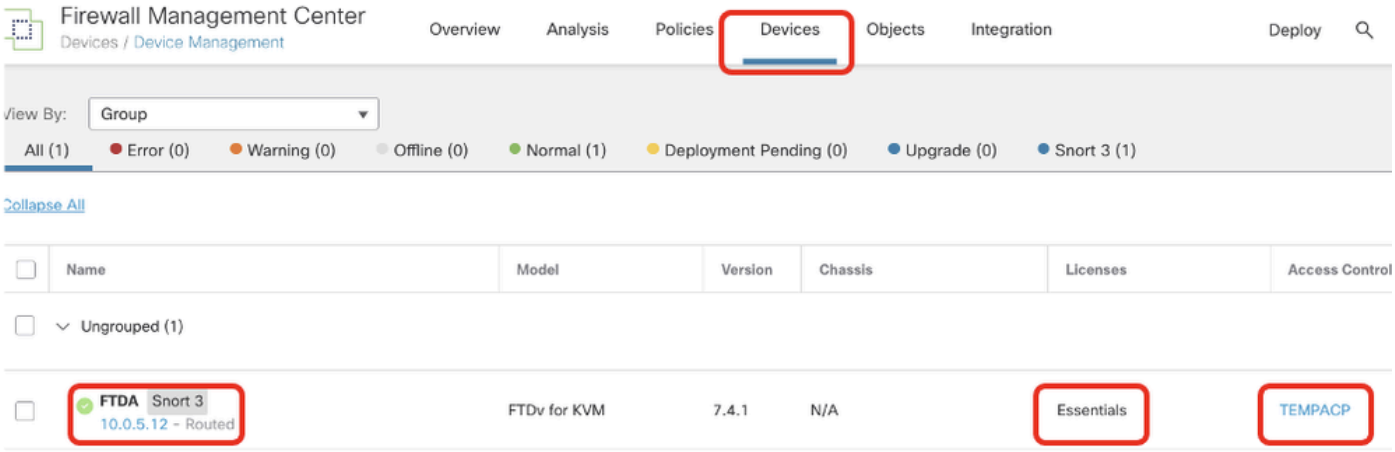
cd /home/cisco/fmc_ansible/

```
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
```

Verificación

Use esta sección para confirmar que su configuración funciona correctamente.

Inicie sesión en FMC GUI. Vaya a **Devices > Device Management,** el FTD se registró correctamente en el FMC con la política de control de acceso configurada.



Página Gestión de Dispositivos

Troubleshoot

En esta sección encontrará información que puede utilizar para solucionar problemas de configuración.

Para ver más registros del cuaderno de campaña de Ansible, puede ejecutar el cuaderno de campaña de Ansible con -vv.

<#root>

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

-vvv

Información Relacionada

[Cisco Devnet FMC Ansible]

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).