Configuration du cadre de stratégie modulaire Firewall Threat Defense

Table des matières

Introduction

Conditions préalables

- Exigences
- Composants utilisés

Informations générales

- Ingrédients MPF
- Directionnalité des fonctionnalités

Configurer

- Topologie
- Tâche 1 : désactivation globale de l'inspection SIP sur FTD
- Tâche 2 : désactivation de l'inspection SIP pour des hôtes spécifiques
- Tâche 3 : configuration du contournement d'état TCP pour des hôtes spécifiques
- Tâche 4. Modification des résultats de la commande traceroute
- Tâche 5. Définition des délais de connexion
- Tâche 6. Authentification BGP via FTD
- Tâche 7. Détection des connexions inactives (DCD)

Informations connexes

Introduction

Ce document décrit le cadre de politique modulaire (MPF) de la défense contre les menaces de pare-feu (FTD)

Conditions préalables

Exigences

Il n'y a pas de conditions spécifiques pour ce document.

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Cisco Secure Firewall 3130 Threat Defense Version 10.0.0 (Build 140)
- Firewall Management Center (FMC) version 10.0.0 (build 140)

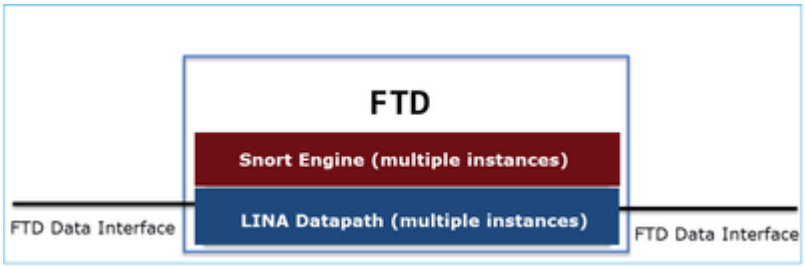
The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

Présentation du plan de données FTD

Cisco Firepower Threat Defense (FTD) est une image logicielle unifiée qui comprend deux moteurs principaux :

- Datapath (également appelé LINA)
- Moteur du renifleur



Le chemin de données LINA et le moteur Snort sont les principales parties du plan de données du FTD.

Ingrédients MPF

MPF utilise les composants suivants :

- class-map correspond au trafic intéressant.
- policy-map applique des actions au trafic intéressant correspondant à la class-map.
- service-policy applique le policy-map globalement (sur toutes les interfaces) ou sur une interface spécifique.

Directionnalité des fonctionnalités

En ce qui concerne la direction des fonctionnalités, consultez le guide de configuration ASA :

https://www.cisco.com/c/en/us/td/docs/security/asa/asa924/configuration/firewall/asa-924-firewall-config/inspect-service-policy.html

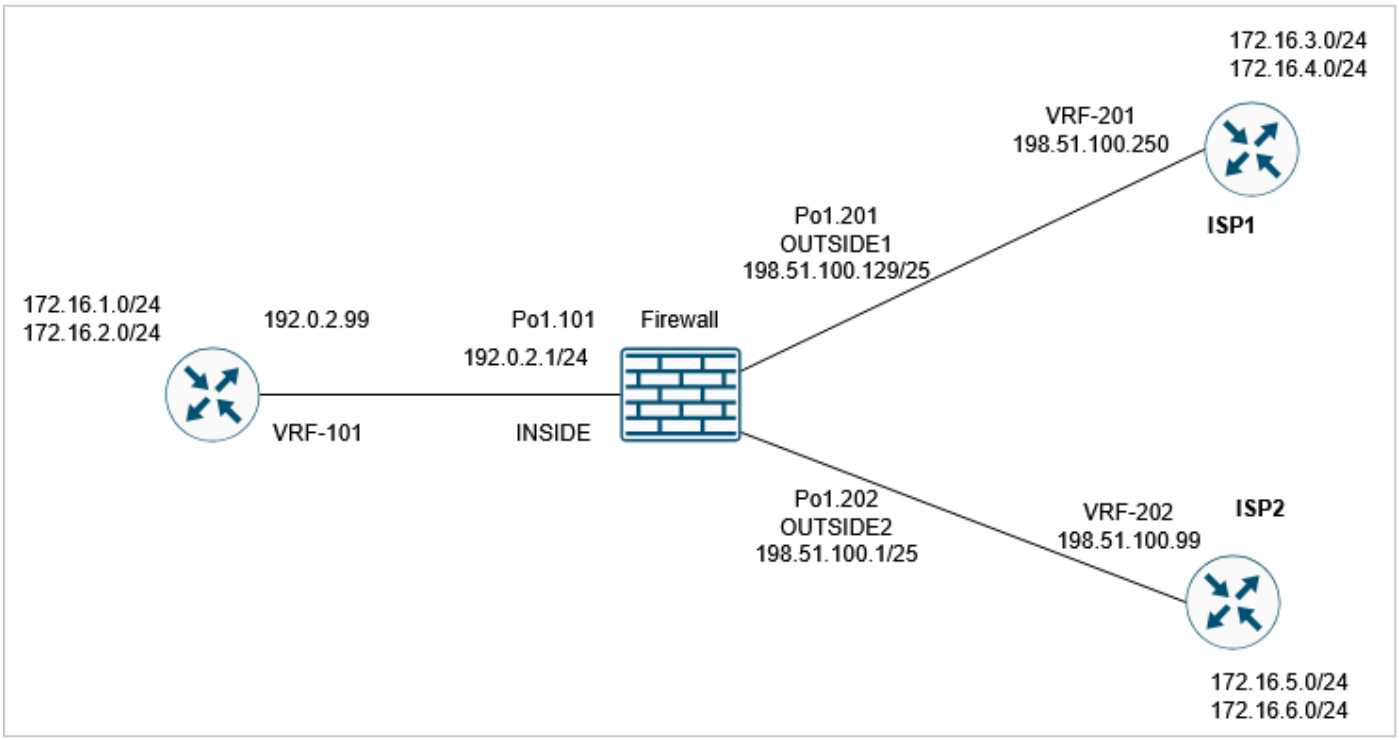
Les fonctions associées au FTD sont mises en surbrillance :

Table 2. Feature Directionality

Feature	Single Interface Direction	Global Direction
Application inspection (multiple types)	Bidirectional	Ingress
NetFlow Secure Event Logging filtering	N/A	Ingress
QoS input policing	Ingress	Ingress
QoS output policing	Egress	Egress
QoS standard priority queue	Egress	Egress
TCP and UDP connection limits and timeouts, and TCP sequence number randomization	Bidirectional	Ingress
TCP normalization	Bidirectional	Ingress
TCP state bypass	Bidirectional	Ingress
User statistics for Identity Firewall	Bidirectional	Ingress

Configurer

Topologie



La configuration MPF par défaut (10.0.0) :

<#root>

```
firewall#

show run policy-map

!
!
policy-map type inspect dns preset_dns_map
 parameters
  message-length maximum client auto
  message-length maximum 512
  no tcp-inspection
policy-map type inspect ip-options UM_STATIC_IP_OPTIONS_MAP
 parameters
  eool action allow
  nop action allow
  router-alert action allow
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
  inspect h323 ras
  inspect rsh
  inspect rtsp
  inspect sqlnet
  inspect skinny
  inspect sunrpc
  inspect sip
  inspect netbios
  inspect tftp
```

```
  inspect icmp
  inspect icmp error
  inspect ip-options UM_STATIC_IP_OPTIONS_MAP
 class class_snmp
  inspect snmp
 class class-default
  set connection advanced-options UM_STATIC_TCP_MAP

firewall#
```

show run class-map

```
!
class-map inspection_default
 match default-inspection-traffic
class-map class_snmp
 match port udp eq 4161
!
firewall#
```

show run service-policy

```
service-policy global_policy global
```

Tâche 1 : désactivation globale de l'inspection SIP sur FTD

Cette tâche nécessite la désactivation de l'inspection SIP dans le moteur FTD LINA. L'une des raisons peut être une exigence de politique ou un défaut logiciel lié au SIP qui affecte le trafic de transit.

Solution

Avant de désactiver l'inspection SIP, vérifiez qu'elle est appliquée au trafic de transit :

<#root>

```
firewall#
```

packet-tracer input INSIDE udp 172.16.1.1 5060 172.16.3.1 5060

```
...
Phase: 8
```

Type: INSPECT

Subtype: inspect-sip

Result: ALLOW

Elapsed time: 34788 ns
Config:

class-map inspection_default

match default-inspection-traffic

policy-map global_policy

class inspection_default

inspect sip

service-policy global_policy global

Additional Information:
...
Result:
input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
output-interface: OUTSIDE1(vrfid:0)
output-status: up
output-line-status: up

Action: allow

Time Taken: 326018 ns

Il existe 2 façons de désactiver globalement l'inspection SIP :

Solution 1 : Désactiver SIP de FTD CLISH CLI

<#root>

>

configure inspection sip disable

Building configuration...
Cryptochecksum: ef7528dc 7338986d 6714a3a2 4770528e

7818 bytes copied in 0.250 secs
[OK]

Vérification

<#root>

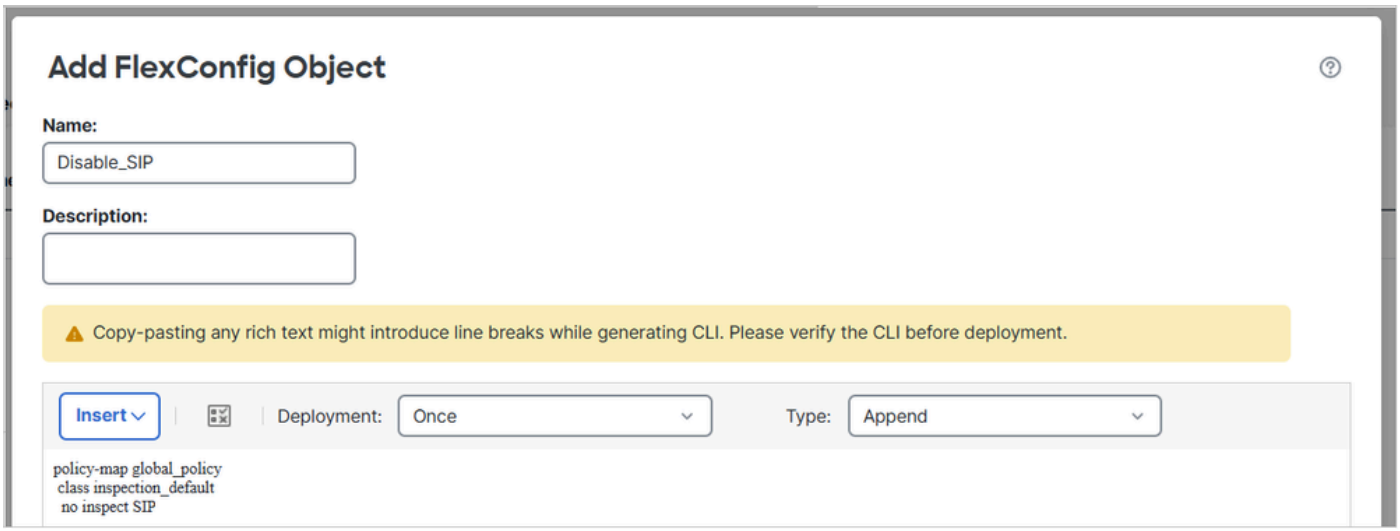
>

show running-config policy-map | include sip

>

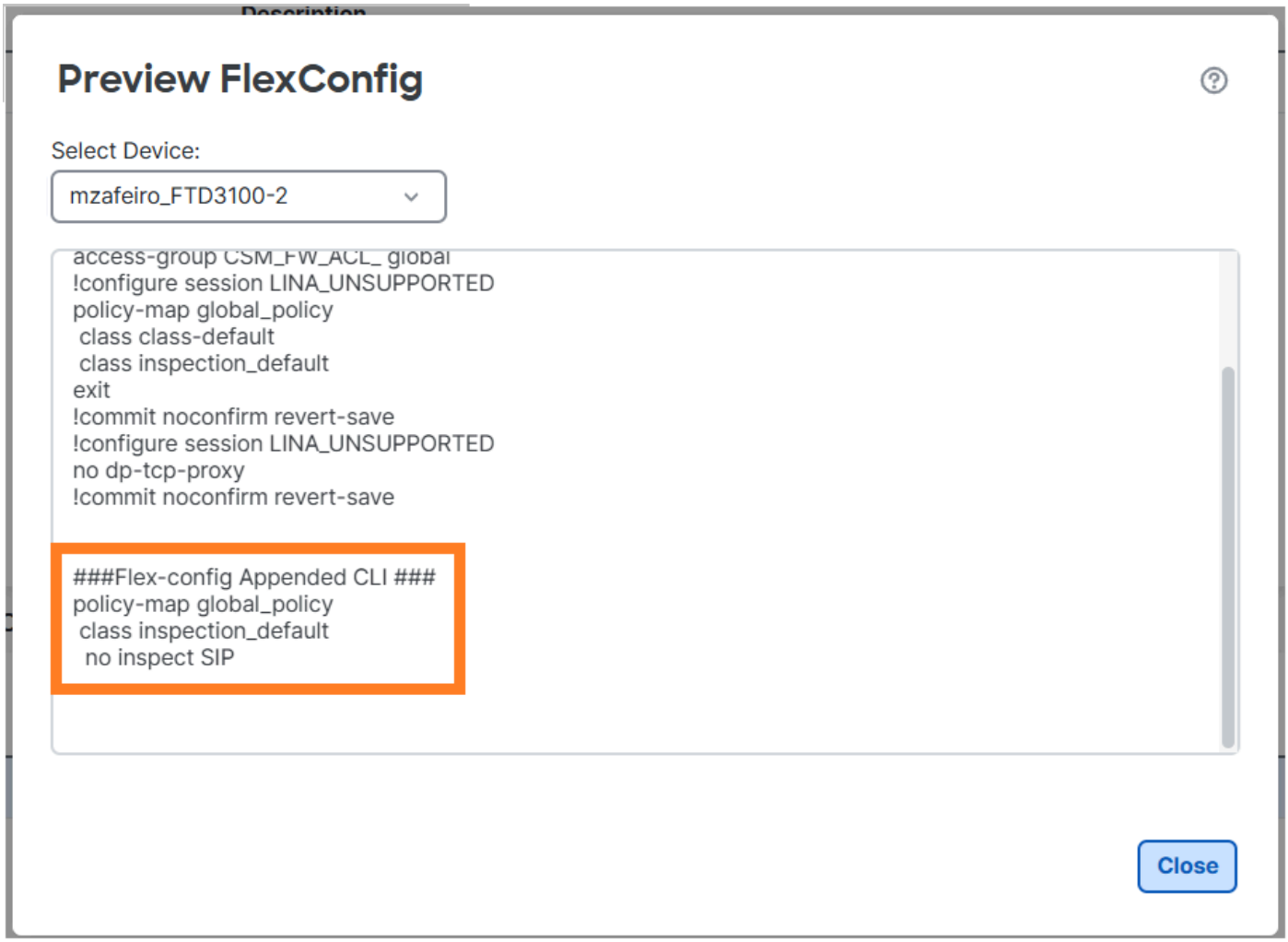
Solution 2 : Désactiver SIP avec FlexConfig

Sur FMC, accédez à Devices > FlexConfig et créez un objet FlexConfig :




```
policy-map global_policy
 class inspection_default
  no inspect sip
```

Appliquer Sélectionnez la stratégie FlexConfig et sélectionnez Preview Config afin de l'afficher :



Enfin, déployez la stratégie.

Vérification

<#root>

```
firewall#
```

show run policy-map | include sip

```
firewall#
```

Remarque - Vous devez effacer la connexion SIP existante de la table de connexion LINA afin que les connexions soient rétablies sans inspection SIP. Vous pouvez utiliser cette commande afin de vérifier les connexions SIP existantes :

<#root>

```
firewall#
```

```
show conn port 5060
```

Tâche 2 : désactivation de l’inspection SIP pour des hôtes spécifiques

Dans cette tâche, il est nécessaire de désactiver l'inspection SIP pour le trafic entre ces réseaux :

- SRC : 172.16.1.0/24
- DST : 172.16.3.0/24

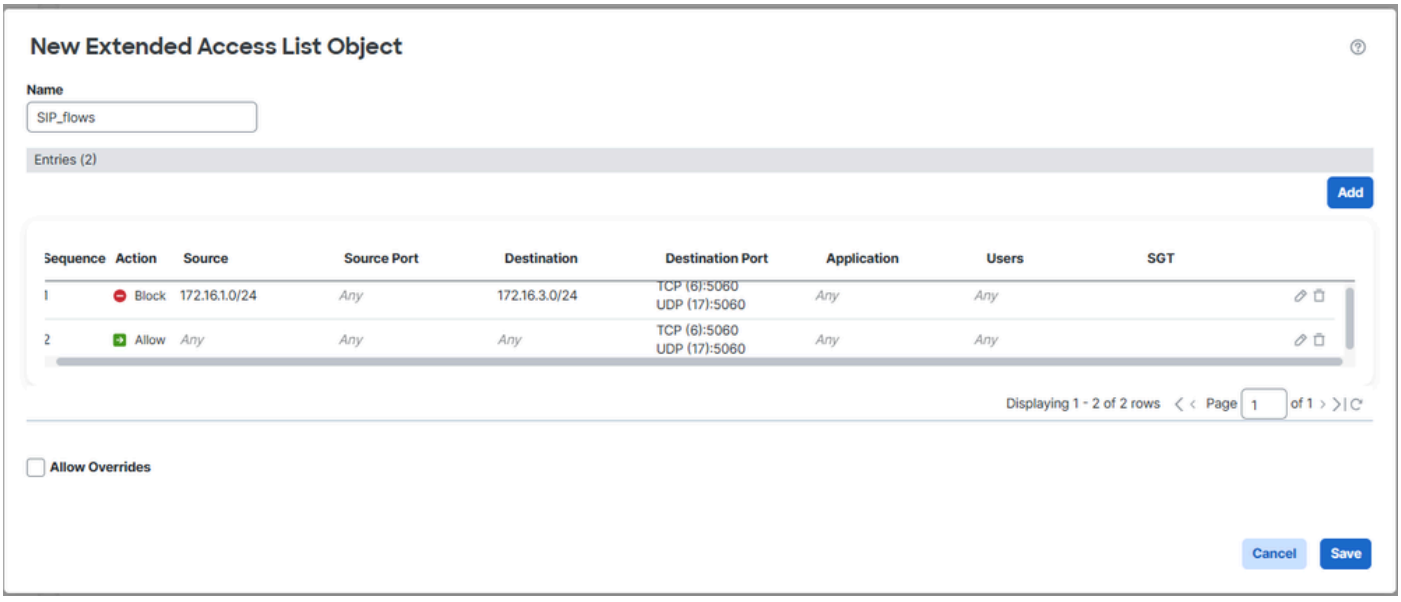
L'une des raisons de ce choix peut être un défaut logiciel lié au SIP qui affecte le trafic de transit

Solution

Utilisez FlexConfig.

Étape 1

Accédez à Objets > Liste d'accès > Étendue et créez une liste d'accès étendue qui correspond au trafic intéressant. Vous devez utiliser l'action Bloquer depuis l'objectif d'exclure le trafic spécifique. En outre, ajoutez une règle Allow pour correspondre au reste du trafic :



Remarque : Assurez-vous que la règle « Autoriser » ne spécifie que le trafic SIP (TCP/UDP

5060). Après la correction du bogue Cisco CSCvw35444 le nouveau comportement consiste à abandonner tout trafic non SIP :
Ancien comportement :

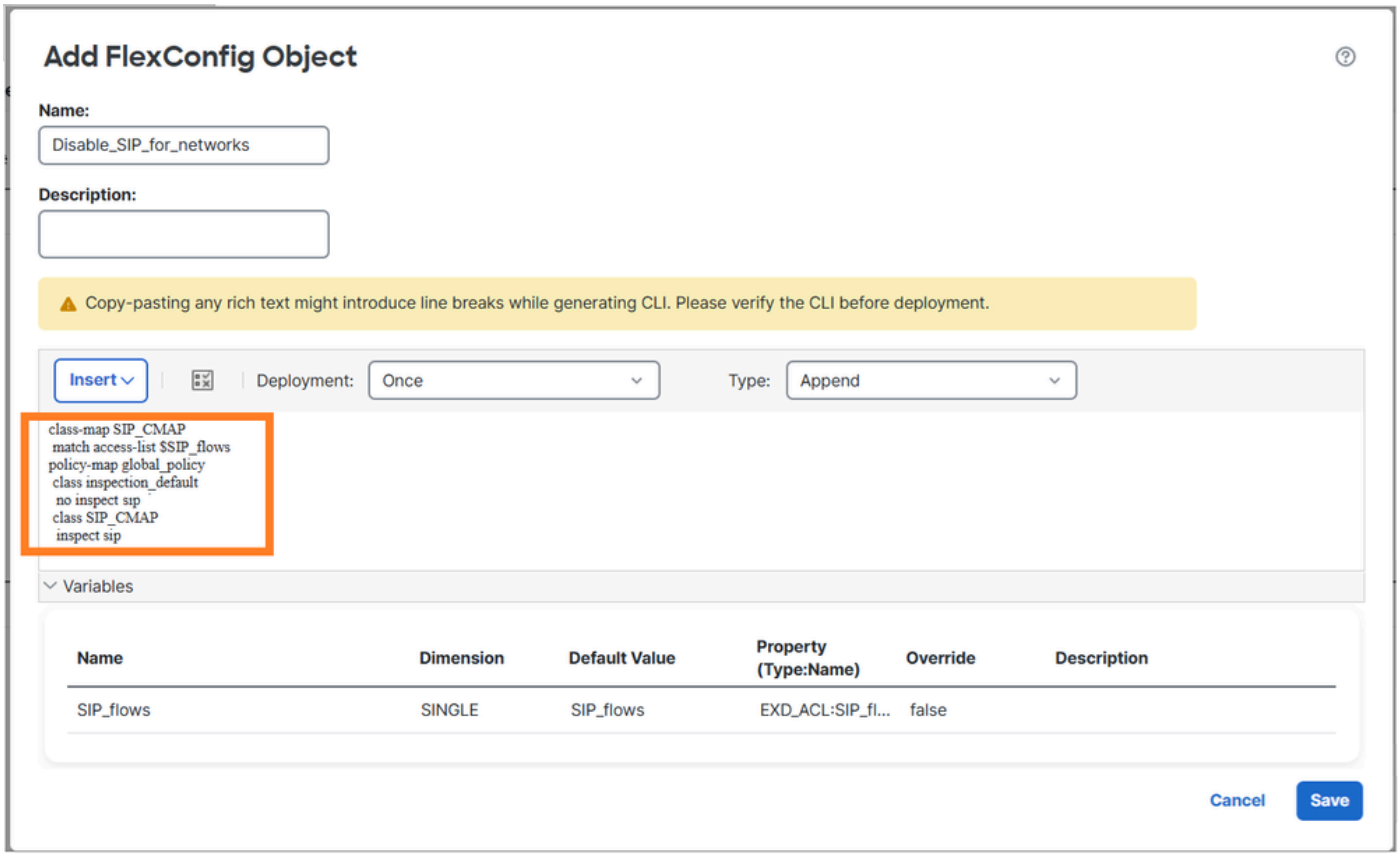
policy-map type inspect sip pmap_test
 paramètres
  traffic-non-sip ----> La valeur par défaut est autoriser.

Nouveau comportement : Par défaut, le trafic non SIP est abandonné :
policy-map type inspect sip pmap_test
  paramètres
    no traffic-non-sip ----> La valeur par défaut est « drop ».

ciscoasa
SIP::SIP, paramètre: Réponse SIP abandonnée INVALID de inside:192.0.2.1.230/1025 vers outside:192.0.2.230/5060 ; no traffic-non-sip : Trafic non SIP refusé

Étape 2

Créez un objet FlexConfig avec un class-map qui correspond à la liste de contrôle d'accès (ACL) SIP et appliquez-le à la politique globale :



L'objet FlexConfig configuré :

```
class-map SIP_CMAP
 match access-list $SIP_flows
policy-map global_policy
 class inspection_default
  no inspect sip
 class SIP_CMAP
  inspect sip
```

Remarque

Lors de la configuration de la liste de contrôle d'accès permit, essayez d'être aussi spécifique que possible (par exemple, mettre des ports de protocole) pour éviter tout impact potentiel sur le CPU. L'exemple de cette tâche ne spécifie pas de ports de protocole et peut être évité en production.

Vérification 1

<#root>

```
firewall#

show run policy-map | begin global


policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
  inspect h323 ras
  inspect rsh
  inspect rtsp
  inspect sqlnet
  inspect skinny
  inspect sunrpc
  inspect netbios
  inspect tftp
  inspect icmp
  inspect icmp error
  inspect ip-options UM_STATIC_IP_OPTIONS_MAP
 class class_snmp
  inspect snmp

 class SIP_CMAP



  inspect sip


 class class-default
```

```
  set connection advanced-options UM_STATIC_TCP_MAP

firewall#
```

show run class-map

```
!
```

class-map SIP_CMAP

```
 match access-list SIP_flows

class-map inspection_default
 match default-inspection-traffic
class-map class_snmp
 match port udp eq 4161

firepower#
```

show run access-list SIP_flows

```
access-list SIP_flows extended deny object-group ProxySG_ExtendedACL_150323855735 172.16.1.0 255.255.25
access-list SIP_flows extended permit object-group ProxySG_ExtendedACL_150323855739 any any

firepower#
```

show access-list SIP_flows

```
access-list SIP_flows; 4 elements; name hash: 0xb2b7de4d
access-list SIP_flows line 1 extended deny object-group ProxySG_ExtendedACL_150323855735 172.16.1.0 255
access-list SIP_flows line 1 extended deny tcp 172.16.1.0 255.255.255.0 172.16.3.0 255.255.255.0 eq sip
access-list SIP_flows line 1 extended deny udp 172.16.1.0 255.255.255.0 172.16.3.0 255.255.255.0 eq sip
access-list SIP_flows line 2 extended permit object-group ProxySG_ExtendedACL_150323855739 any any (hit
access-list SIP_flows line 2 extended permit tcp any any eq sip (hitcnt=0) 0x60e34fab
access-list SIP_flows line 2 extended permit udp any any eq sip (hitcnt=0) 0xdae62994
```

Vérification 2

Le trafic qui n'est pas inspecté par l'inspection SIP a deny=true :

<#root>

```
firewall#
```

packet-tracer input INSIDE udp 172.16.1.1 5060 172.16.3.1 5060 detail | begin INSPECT

Type: INSPECT

Subtype: inspect-sip

Result: ALLOW
Elapsed time: 37910 ns
Config:

class-map SIP_CMAP

match access-list SIP_flows

policy-map global_policy

class SIP_CMAP

inspect sip

service-policy global_policy global

Additional Information:
Forward Flow based lookup yields rule:
in id=0x14af42cfa810, priority=70, domain=inspect-sip,

deny=true

hits=1

, user_data=0x000014af4570bea0, cs_id=0x0, use_real_addr, flags=0x0, protocol=0

src ip/id=172.16.1.0, mask=255.255.255.0, port=0, tag=any

dst ip/id=172.16.3.0, mask=255.255.255.0, port=5060, tag=any,

dscp=0x0, input_ifc=INSIDE(vrfid:0), output_ifc=any
...

Le trafic qui est inspecté par l'inspection SIP a deny=false :

<#root>

firewall#

packet-tracer input INSIDE udp 172.16.2.1 5060 172.16.3.1 5060 detail | begin INSPECT

Type: INSPECT

Subtype: inspect-sip

Result: ALLOW
Elapsed time: 34788 ns
Config:
class-map SIP_CMAP
 match access-list SIP_flows
policy-map global_policy
 class SIP_CMAP
  inspect sip
service-policy global_policy global
Additional Information:
 Forward Flow based lookup yields rule:
 in id=0x14af459099d0, priority=70, domain=inspect-sip,

deny=false

hits=1, user_data=0x000014af4570bea0, cs_id=0x0, use_real_addr, flags=0x0, protocol=0

src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any

```
       dst ip/id=0.0.0.0, mask=0.0.0.0, port=5060, tag=any,


...
```

Vérification 3

Le compteur d'inspection « sip » augmente lorsqu'un paquet est inspecté par le pare-feu :

<#root>

```
firewall#

show service-policy inspect sip



Global policy:
  Service-policy: global_policy
    Class-map: inspection_default
    Class-map: class_snmp
    Class-map: SIP_CMAP
      Inspect: sip ,

packet 2

, lock fail 0, drop 0, reset-drop 0, 5-min-pkt-rate 0 pkts/sec, v6-fail-close 0 sctp-drop-override 0
               tcp-proxy: bytes in buffer 0, bytes dropped 0
...
firewall#

packet-tracer input INSIDE udp 172.16.2.1 5060 172.16.3.1 5060



firewall#

show service-policy inspect sip



Global policy:
  Service-policy: global_policy
    Class-map: inspection_default
    Class-map: class_snmp
    Class-map: SIP_CMAP
      Inspect: sip ,

packet 3

, lock fail 0, drop 0, reset-drop 0, 5-min-pkt-rate 0 pkts/sec, v6-fail-close 0 sctp-drop-override 0
               tcp-proxy: bytes in buffer 0, bytes dropped 0
```

...

Tâche 3 : configuration du contournement d’état TCP pour des hôtes spécifiques

Dans cette tâche, il est nécessaire d'activer le contournement d'état TCP pour le trafic entre ces réseaux :

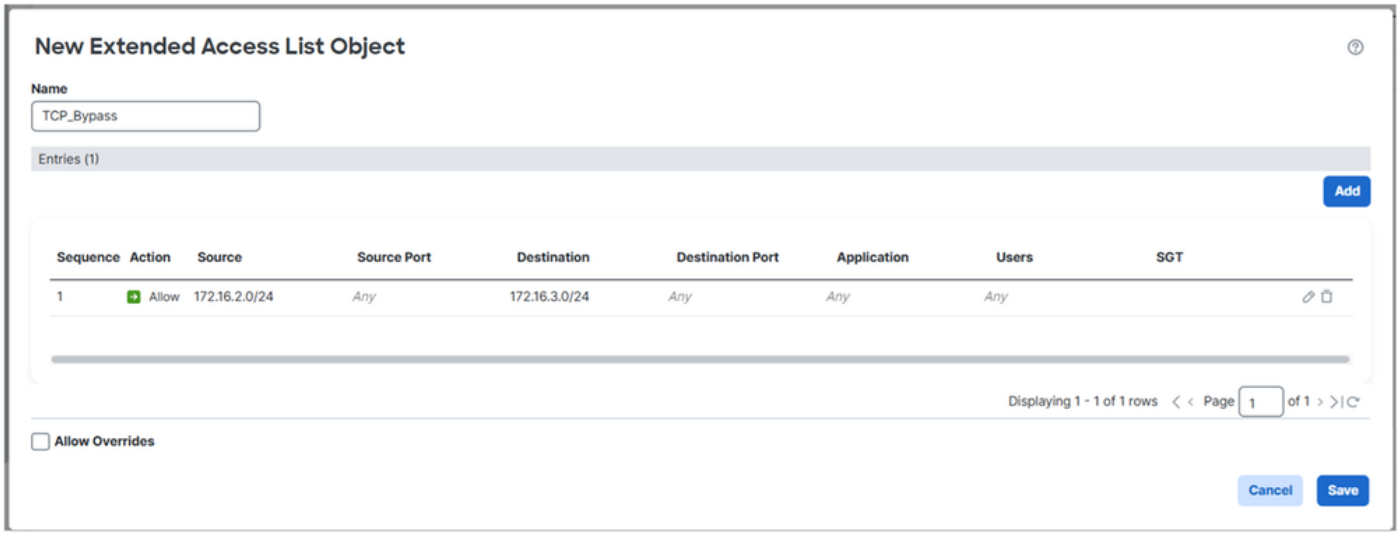
- SRC : 172.16.2.0/24
- DST : 172.16.3.0/24

En général, il n'est pas recommandé d'utiliser le contournement d'état TCP, mais il peut être utilisé comme solution de contournement temporaire pour gérer les flux asymétriques.

Solution 1

Étape 1

Créez une liste de contrôle d’accès étendue correspondant au trafic intéressant :

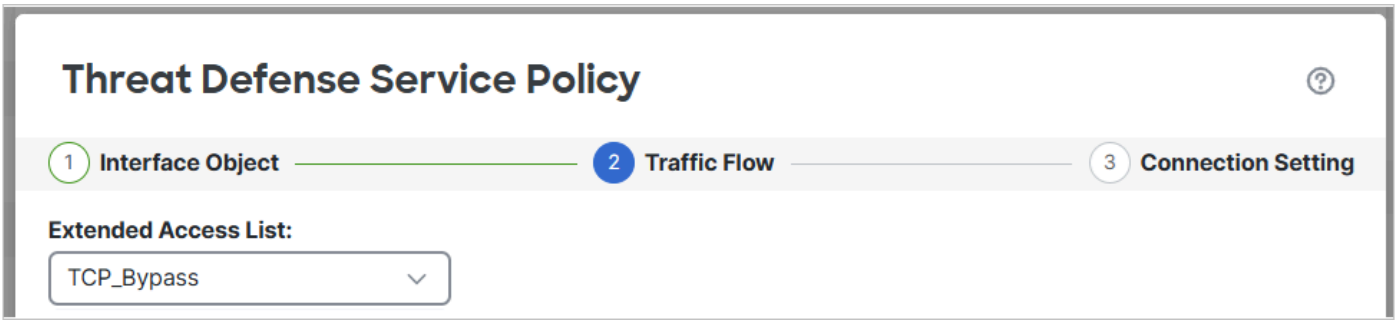


Étape 2

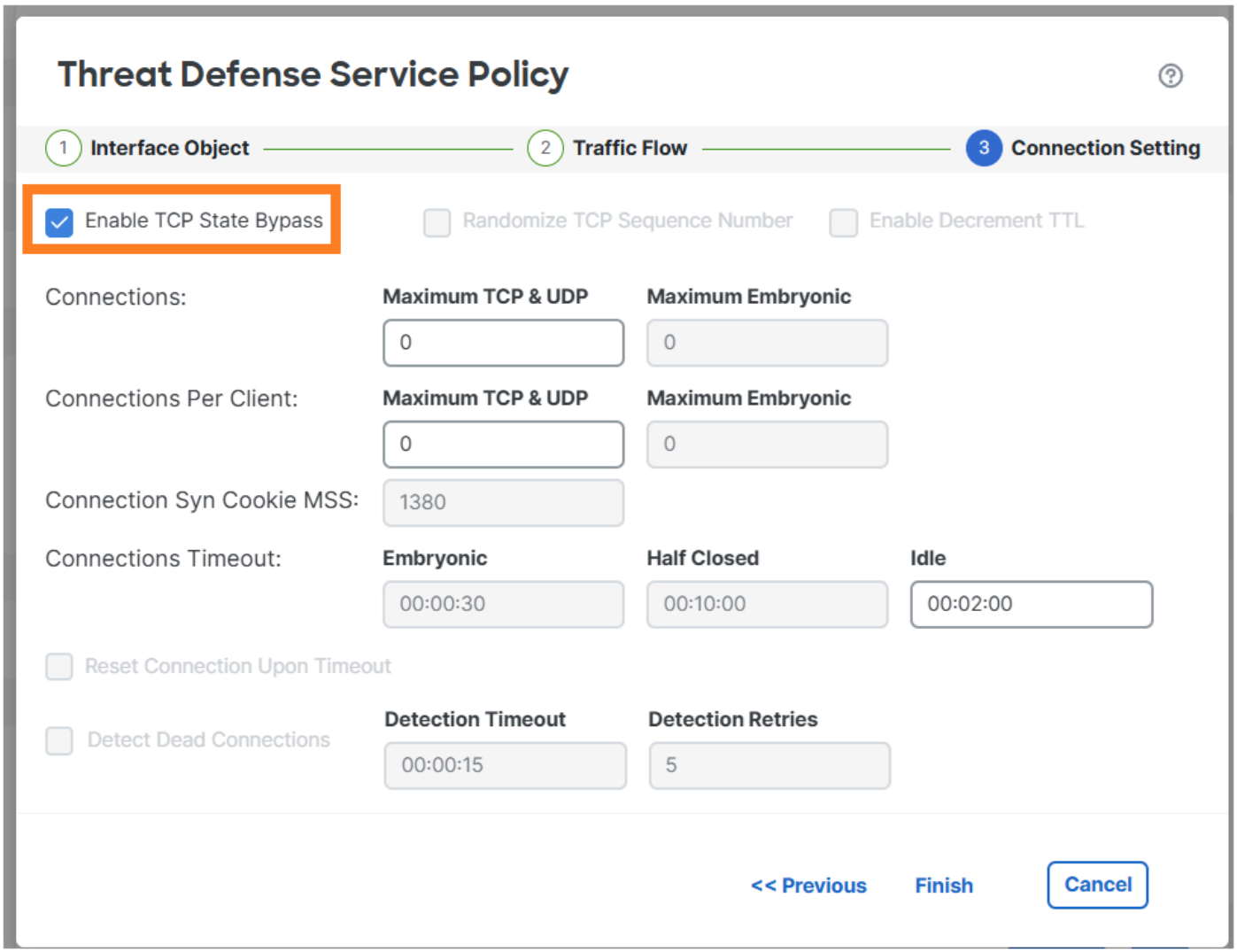
Modifiez la politique de contrôle d'accès (ACP) affectée au FTD, sélectionnez l'onglet Advanced Settings et modifiez la politique de service de défense contre les menaces. Sélectionnez Add Rule et Next.

Étape 3

Sélectionnez la liste de contrôle d’accès étendue :



Étape 4



Étape 5

Sélectionnez Terminer, OK, Enregistrer et Déployer.

Le résultat :

<#root>

```
firewall#

show run policy-map global_policy


!
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
  inspect h323 ras
  inspect rsh
  inspect rtsp
  inspect sqlnet
  inspect skinny
  inspect sunrpc
  inspect netbios
  inspect tftp
  inspect icmp
  inspect icmp error
  inspect ip-options UM_STATIC_IP_OPTIONS_MAP

 class class_map_TCP_Bypass



  set connection random-sequence-number disable

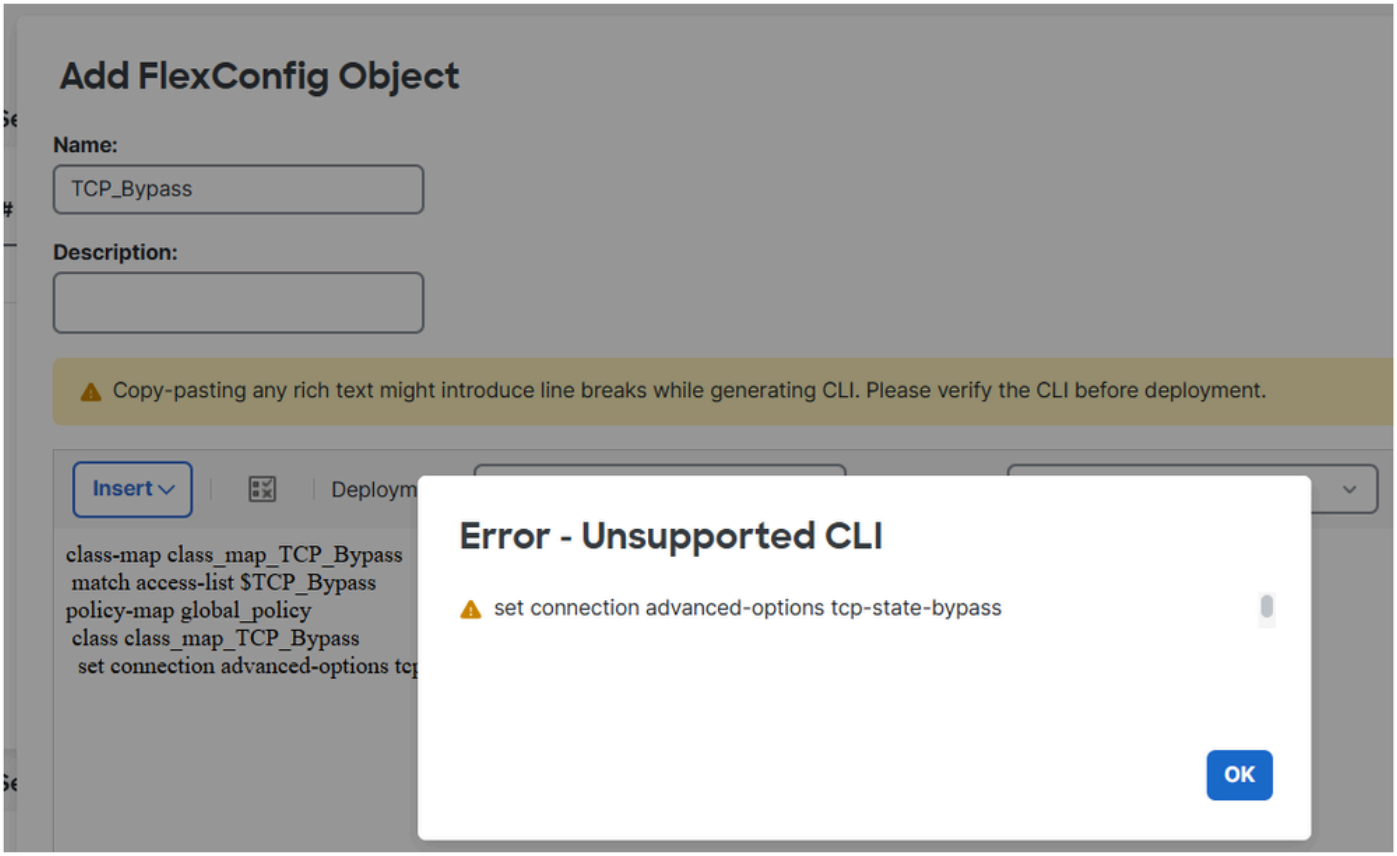


  set connection advanced-options tcp-state-bypass


 class class_snmp
  inspect snmp
 class class-default
  set connection advanced-options UM_STATIC_TCP_MAP
```

Remarque : Dans les versions précédentes de FMC comme 6.x, vous pouvez utiliser FlexConfig

pour configurer le contournement d'état TCP. Dans les versions plus récentes, ceci n'est pas pris en charge :



Vérification

<#root>

```
firewall#
```

packet-tracer input INSIDE tcp 172.16.2.1 1111 172.16.3.1 80 detail | begin CONN

```
Type: CONN-SETTINGS
Subtype:
Result: ALLOW
Elapsed time: 334 ns
Config:
```

class-map class_map_TCP_Bypass

match access-list TCP_Bypass

```
policy-map global_policy


 class class_map_TCP_Bypass


  set connection conn-max 0 embryonic-conn-max 0 random-sequence-number disable syn-cookie-mss 1380


  set connection advanced-options tcp-state-bypass


service-policy global_policy global
Additional Information:
 Forward Flow based lookup yields rule:
 in id=0x14af45906b70, priority=7, domain=conn-set, deny=false


hits=1

, user_data=0x000014af45906df0, cs_id=0x0, use_real_addr, flags=0x0, protocol=0

       src ip/id=172.16.2.0, mask=255.255.255.0, port=0, tag=any



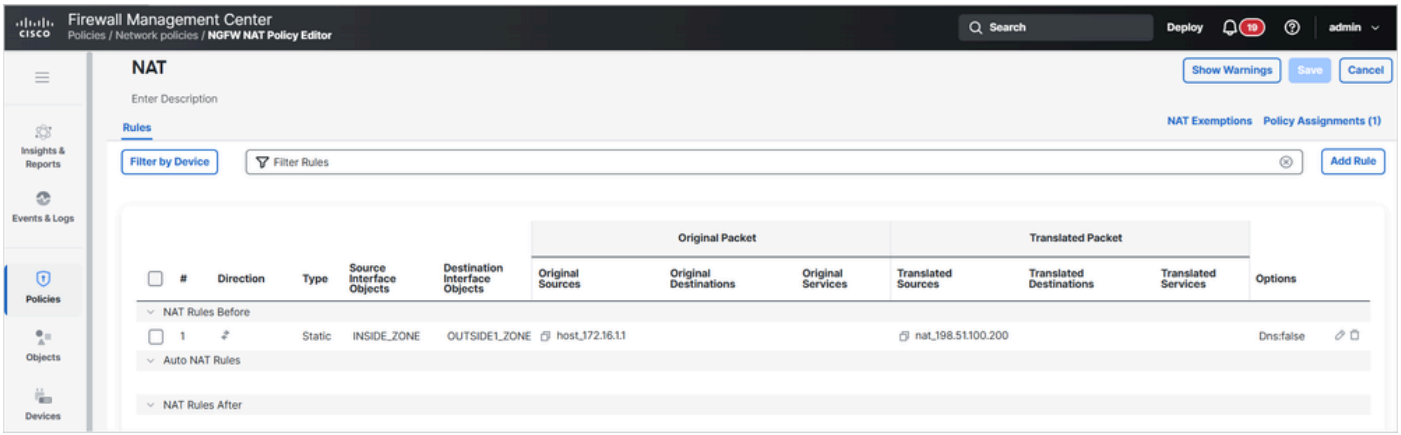
       dst ip/id=172.16.3.0, mask=255.255.255.0, port=0, tag=any,


       dscp=0x0, input_ifc=INSIDE(vrfid:0), output_ifc=any
...
```

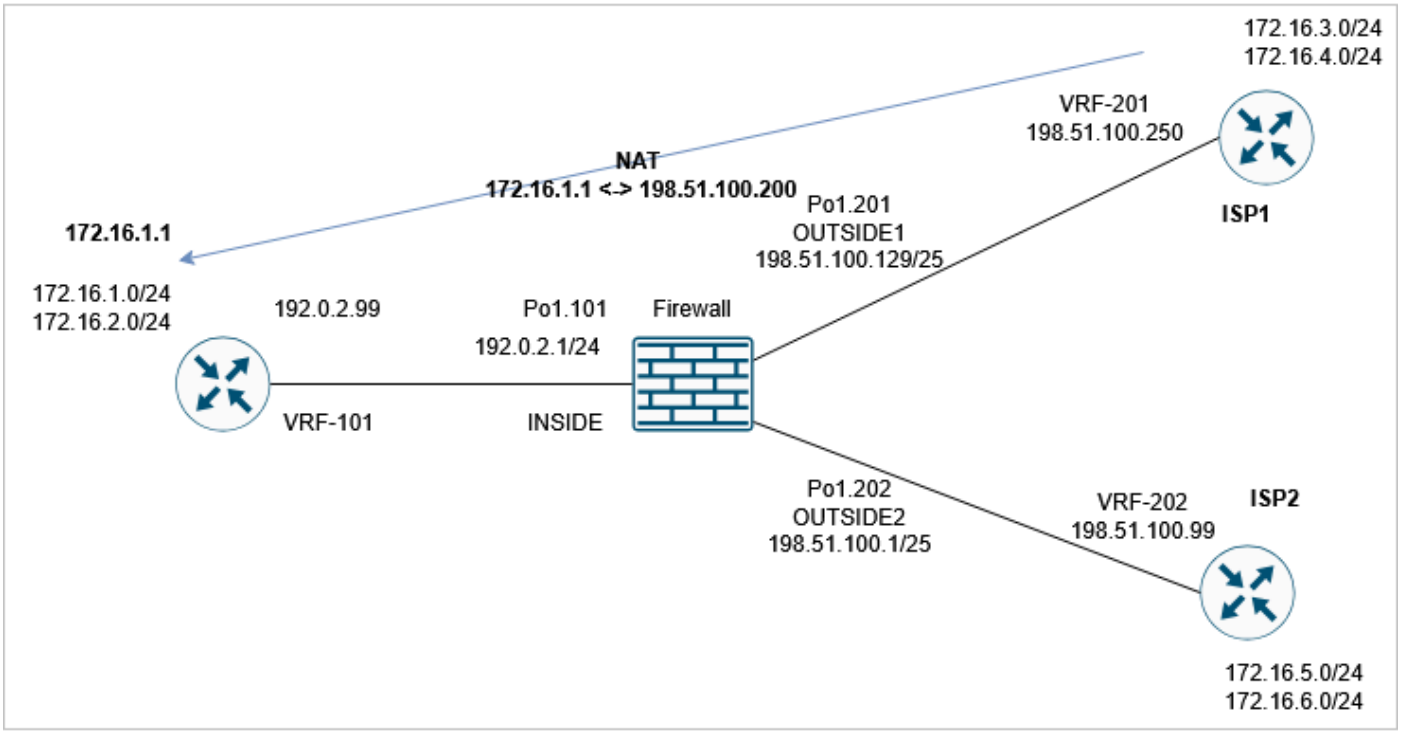
Tâche 4. Modification des résultats de la commande traceroute

Prérequis

Configurez la fonction NAT statique sur FTD de sorte que l’adresse IP 172.16.1.1 située derrière l’interface INSIDE apparaisse comme 198.51.100.200 sur les hôtes OUTSIDE1 :



Exécutez ensuite une commande traceroute depuis ISP1 vers 198.51.100.200 (hôte 172.16.1.1) :



<#root>

```
router1#
```

traceroute vrf VRF-201 198.51.100.200

```
Type escape sequence to abort.
Tracing the route to 198.51.100.200
VRF info: (vrf in name/id, vrf out name/id)
```

1 192.0.2.99 1 msec 1 msec *

Exigence

Modifiez la configuration FTD de sorte que la commande traceroute corresponde à cette sortie :

<#root>

```
router1#

traceroute vrf VRF-201 198.51.100.200


Type escape sequence to abort.
Tracing the route to 198.51.100.200
VRF info: (vrf in name/id, vrf out name/id)

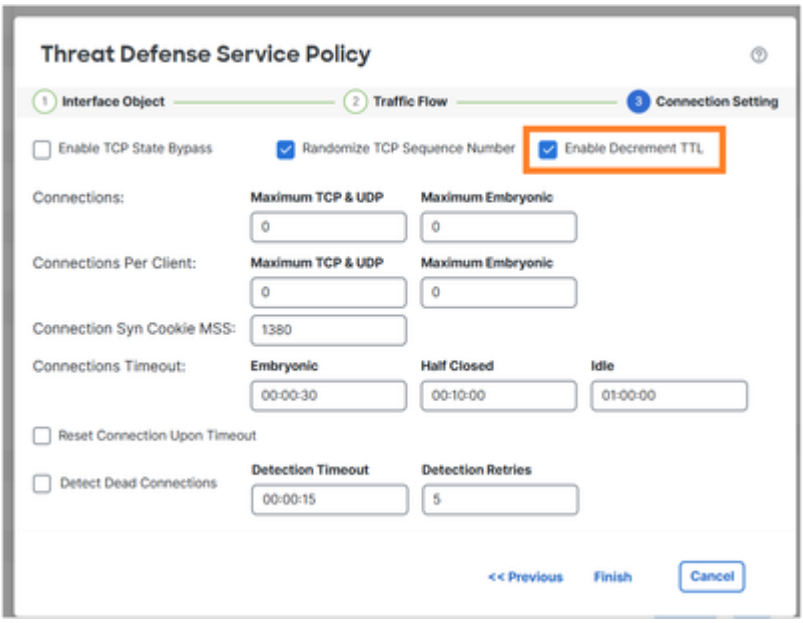
  1 198.51.100.129 1 msec 1 msec *



  2 198.51.100.200 1 msec 2 msec *
```

Solution

La solution comprend deux étapes de configuration :
1. Décrémenter la durée de vie :



Après cette modification, la commande traceroute révèle le saut du pare-feu :

```
<#root>

router1#

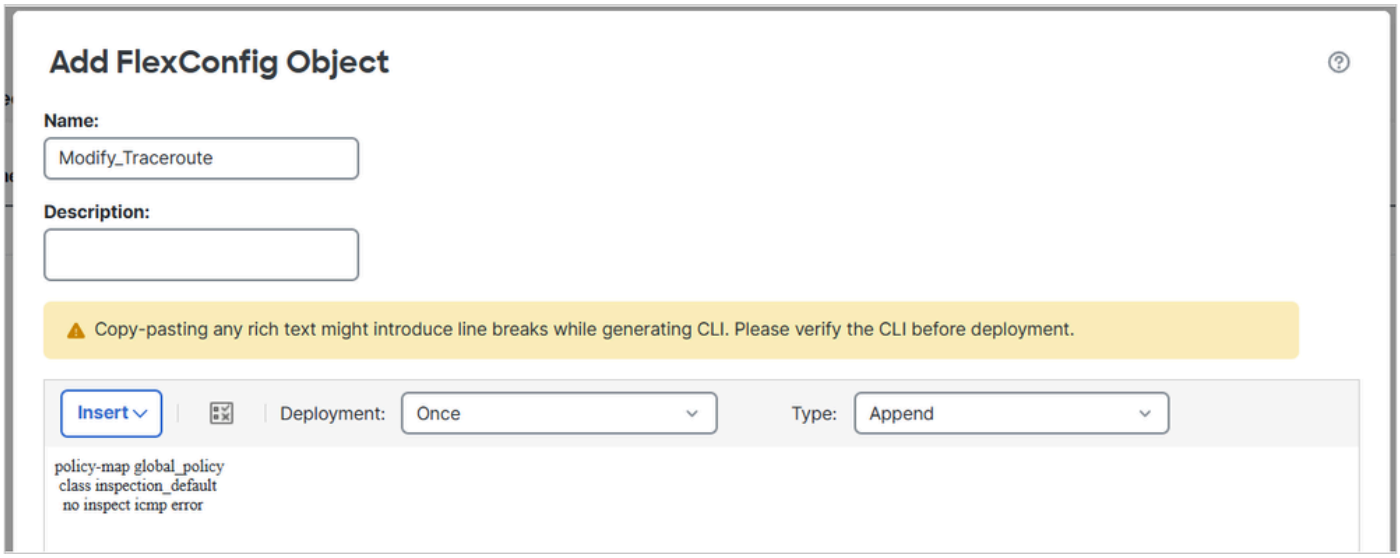
traceroute vrf VRF-201 198.51.100.200


Type escape sequence to abort.
Tracing the route to 198.51.100.200
VRF info: (vrf in name/id, vrf out name/id)

 1 198.51.100.129 1 msec 1 msec *


  2 192.0.2.99 1 msec 1 msec *
```

2. Désactivez l'inspection d'erreur ICMP :




```
policy-map global_policy
 class inspection_default
   no inspect icmp error
```

Vérification

La commande traceroute affiche l’adresse IP NAT traduite de l’hôte distant et l’adresse IP de l’interface FTD :

<#root>

```
router1#

traceroute vrf VRF-201 198.51.100.200


Type escape sequence to abort.
Tracing the route to 198.51.100.200
VRF info: (vrf in name/id, vrf out name/id)

  1 198.51.100.129 1 msec 1 msec *



  2 198.51.100.200 1 msec 2 msec *
```

Tâche 5. Définition des délais de connexion

Exigence

Remplacez le délai d'expiration par 1 semaine pour ce flux :

- Protocole : TCP
- SRC : 172.16.1.1
- DST : 172.16.5.1

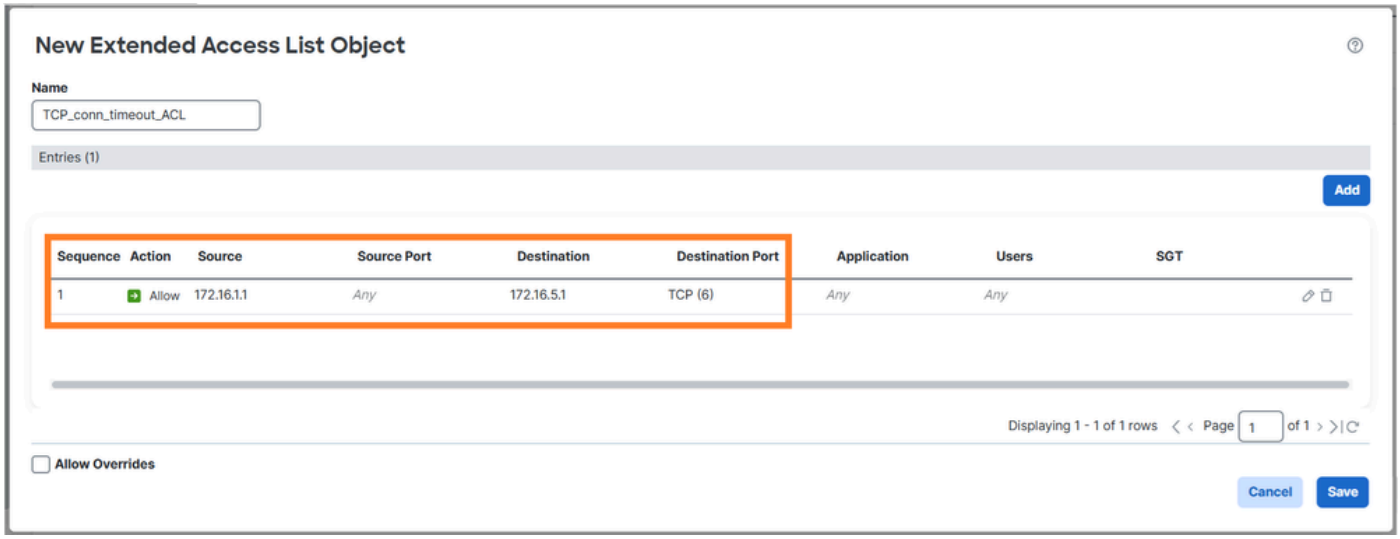
Solution

Pour définir le délai d'attente par flux, vous devez utiliser la stratégie de service.

Étape 1

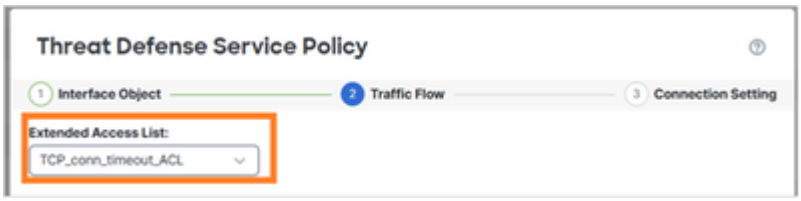
Accédez à Objets > Liste d'accès et créez une liste de contrôle d'accès étendue qui correspond au

trafic intéressant :

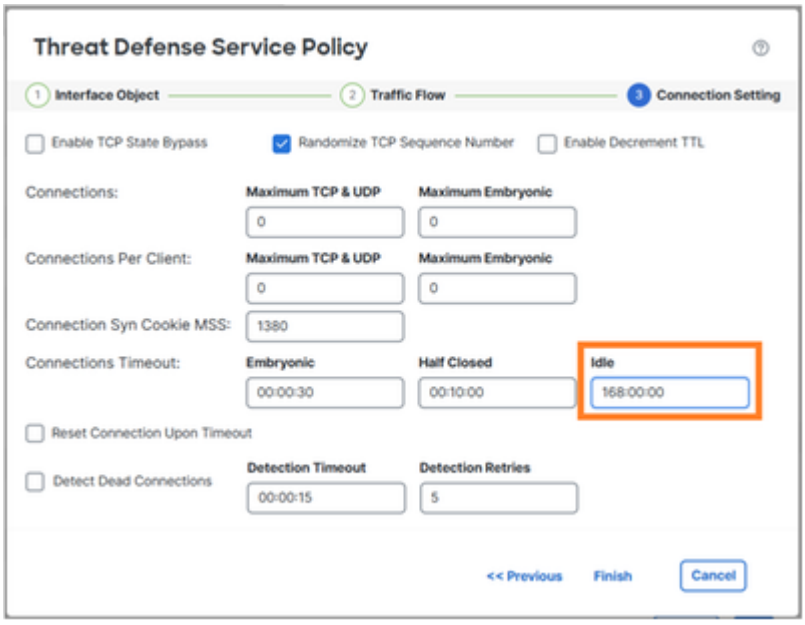


Étape 2

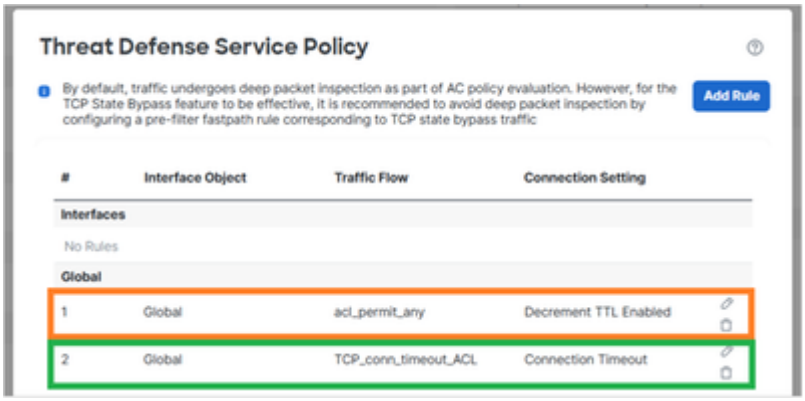
Configurez une stratégie MPF qui utilise la liste de contrôle d'accès créée à l'étape 1 :



Définissez le délai d'inactivité de la connexion :



Supprimez la règle de la tâche précédente car elle chevauche la nouvelle exigence :



Vérification

La configuration de carte de stratégie déployée :

<#root>

```
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
  inspect h323 ras
  inspect rsh
  inspect rtsp
  inspect sqlnet
  inspect skinny
  inspect sunrpc
  inspect netbios
  inspect tftp
  inspect icmp
  inspect ip-options UM_STATIC_IP_OPTIONS_MAP
  inspect sip

 class class_map_TCP_conn_timeout_ACL



  set connection timeout idle 168:00:00


 class class_snmp
  inspect snmp
 class class-default
  set connection advanced-options UM_STATIC_TCP_MAP
```

Lancez une nouvelle connexion TCP de 172.16.1.1 à 172.16.5.1 et vérifiez la table de connexion

du FTD :

<#root>

```
firewall#
```

show conn long address 172.16.5.1

```
...
TCP OUTSIDE2: 172.16.5.1/23 (172.16.5.1/23) INSIDE: 172.16.1.1/29389 (172.16.1.1/29389), flags UIoN1N7,
```

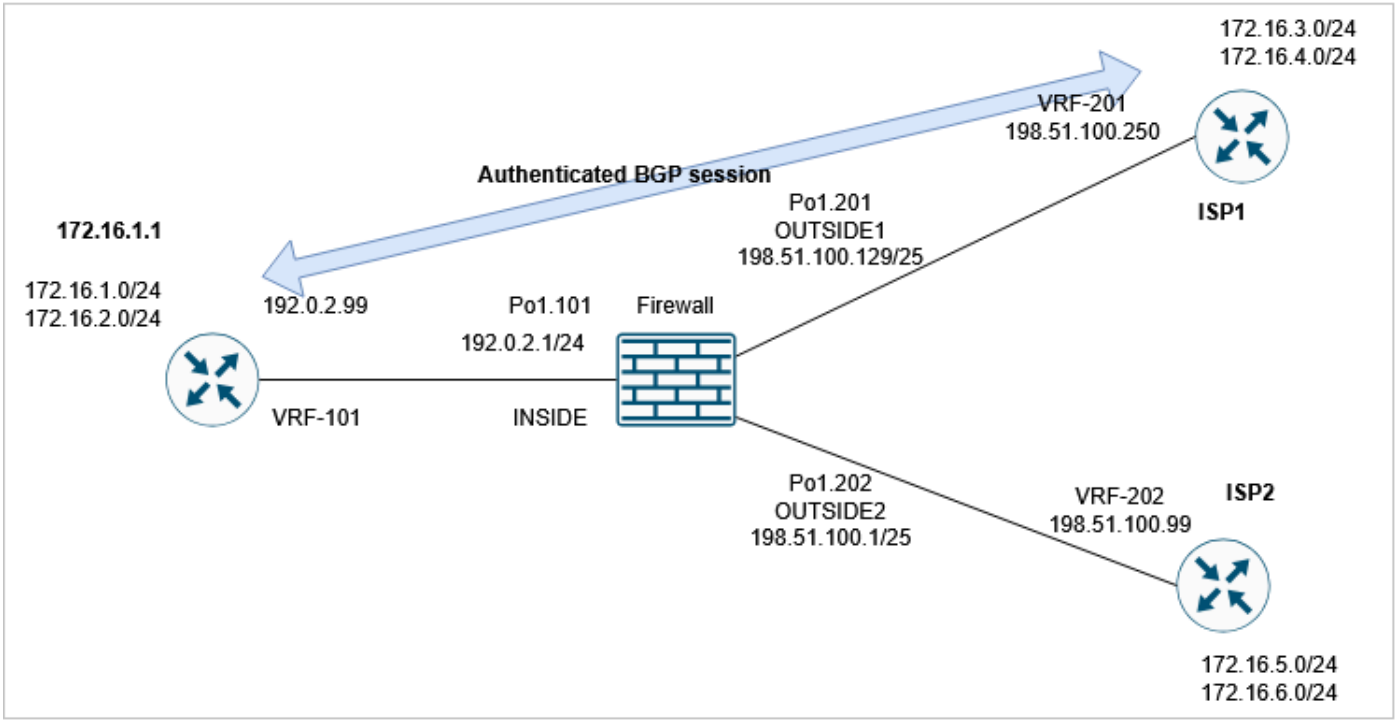
timeout 7D0h

```
, bytes 349, flow id 72, Snort id 6, rule id 268439559, Rx-RingNum 27, Internal-Data0/1
Initiator: 172.16.1.1, Responder: 172.16.5.1
Connection lookup keyid: 890
```

Tâche 6. Authentification BGP via FTD

Prérequis

Configurez une session BGP via le FTD. La session BGP doit utiliser l'authentification.



Vérification

Avec la configuration FTD par défaut, la session BGP n'est pas établie. Sur le routeur, vous pouvez voir :

```
<#root>

router1#
*May 21 07:51:23.595:

%TCP-6-BADAUTH: Invalid MD5 digest

 from 192.0.2.99(24591) to 198.51.100.250(179) tableid - 3
*May 21 07:51:25.595: %TCP-6-BADAUTH: Invalid MD5 digest from 192.0.2.99(24591) to 198.51.100.250(179)
*May 21 07:51:29.595: %TCP-6-BADAUTH: Invalid MD5 digest from 192.0.2.99(24591) to 198.51.100.250(179)
```

Sur le FTD, vous voyez que les deux côtés ne parviennent pas à établir la connexion TCP BGP (les indicateurs de connexion indiquent que seuls les paquets TCP SYN sont reçus) :

```
<#root>

firewall#

show conn port 179


3 in use, 16 most used
Inspect Snort:
        preserve-connection: 2 enabled, 0 in effect, 15 most enabled, 0 most in effect

TCP OUTSIDE1 198.51.100.250:41090 INSIDE 192.0.2.99:179, idle 0:00:00, bytes 0,

flags aA N1


TCP OUTSIDE1 198.51.100.250:179 INSIDE 192.0.2.99:53629, idle 0:00:02, bytes 0,

flags aA N1
```

Solution

Pour autoriser une session BGP authentifiée via le FTD, ces 2 conditions doivent être remplies :

1. TCP MD5 (option 19) doit être autorisé via le FTD.
2. La randomisation des numéros de séquence TCP doit être désactivée.

L'option TCP MD5 est autorisée par défaut :

9.6(2)	Default handling of the named options was changed to allow a packet if it contains a single option of a given type, and drop the packet if there are more than one option of that type. Also, the **md5**, **mss**, **allow multiple**, and **mss maximum** keywords were added. The default for the MD5 option was changed from clear to allow.

<#root>

```
firewall#
```

```
show run all tcp-map
```

```
!
tcp-map UM_STATIC_TCP_MAP
  no check-retransmission
  no checksum-verification
  exceed-mss allow
  queue-limit 0 timeout 4
  reserved-bits allow
  syn-data allow
  synack-data drop
  invalid-ack drop
  seq-past-window drop
  tcp-options range 6 7 allow
  tcp-options range 9 18 allow
  tcp-options range 20 255 allow
  tcp-options selective-ack allow
  tcp-options timestamp allow
  tcp-options window-scale allow
  tcp-options mss allow
  tcp-options md5 allow
  ttl-evasion-protection
  urgent-flag allow
  window-variation allow-connection
```

Désactivez globalement la randomisation TCP ISN (Initial Sequence Number) :

```
<#root>

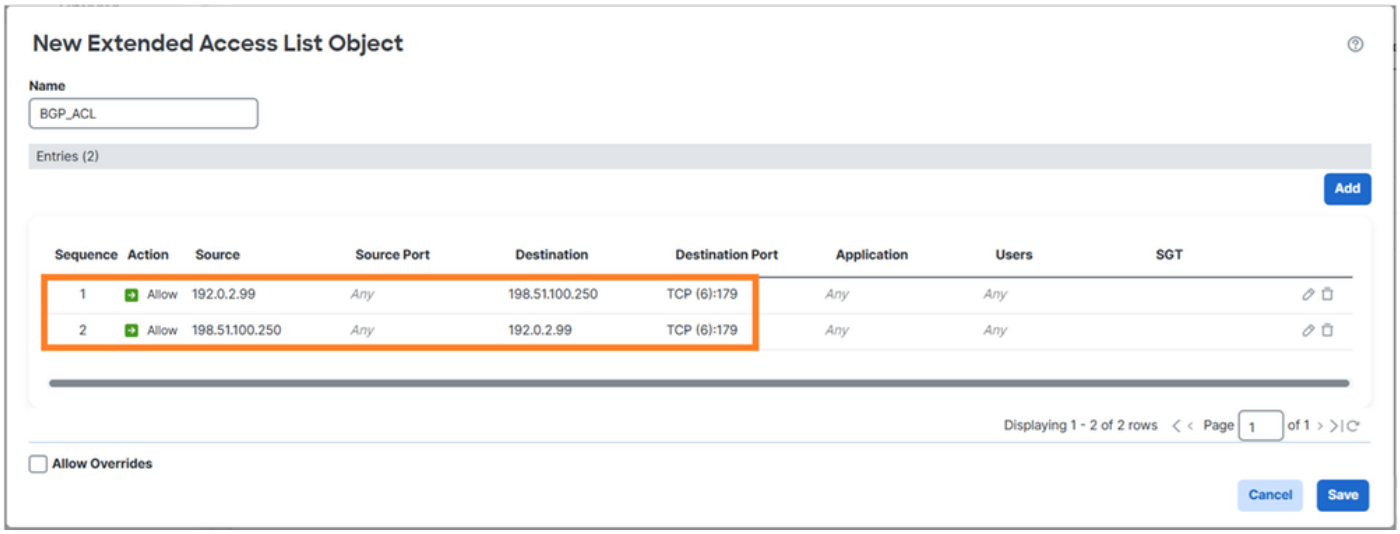
>

configure tcp-randomization disable

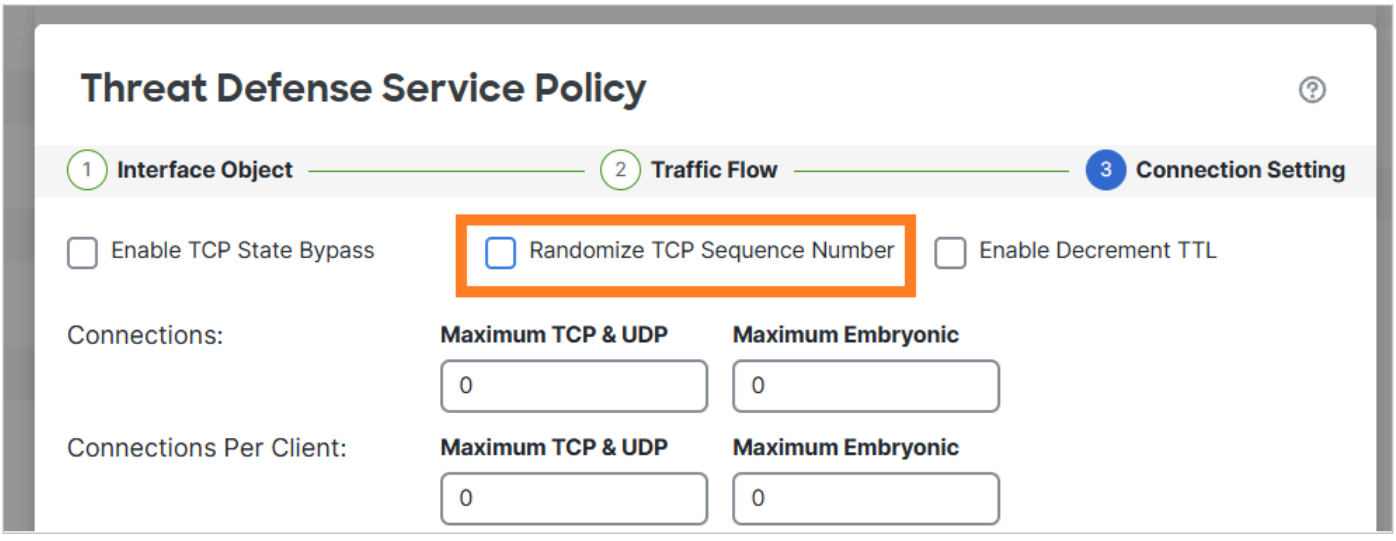

Building configuration...
Cryptochecksum: f8ac5587 7ccc635e bff886a1 bcab820c

8284 bytes copied in 0.260 secs
[OK]
>
```

ou (la méthode préférée) vous créez une liste d'accès étendue qui correspond à la connexion BGP :



et désactivez la randomisation des numéros de séquence TCP à l'aide de la politique de service de défense contre les menaces :



Vérification

La configuration de carte de stratégie déployée :

<#root>

```
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
  inspect h323 ras
  inspect rsh
  inspect rtsp
  inspect sqlnet
  inspect skinny
  inspect sunrpc
  inspect netbios
  inspect tftp
  inspect icmp
  inspect ip-options UM_STATIC_IP_OPTIONS_MAP
  inspect sip

 class class_map_BGP_ACL



  set connection random-sequence-number disable


 class class_snmp
  inspect snmp
 class class-default
  set connection advanced-options UM_STATIC_TCP_MAP
```

La session BGP est établie via FTD :

<#root>

```
firewall#

show conn long port 179


...

TCP OUTSIDE1: 198.51.100.250/49863 (198.51.100.250/49863) INSIDE: 192.0.2.99/179 (192.0.2.99/179), flags
```

```
, idle 44s, uptime 1m40s, timeout 1h0m, bytes 274, flow id 111, Snort id 3, rule id 268439559, Rx-RingN
```

Initiator: 198.51.100.250, Responder: 192.0.2.99

```
  Connection lookup keyid: 83487134
```

Conseil : Vous pouvez configurer une règle de préfiltre fastpath pour le trafic BGP afin d'éviter l'inspection Snort.

Tâche 7. Détection des connexions inactives (DCD)

Exigence

Configurez DCD sur FTD pour le trafic TCP destiné à l'hôte 172.16.3.1.

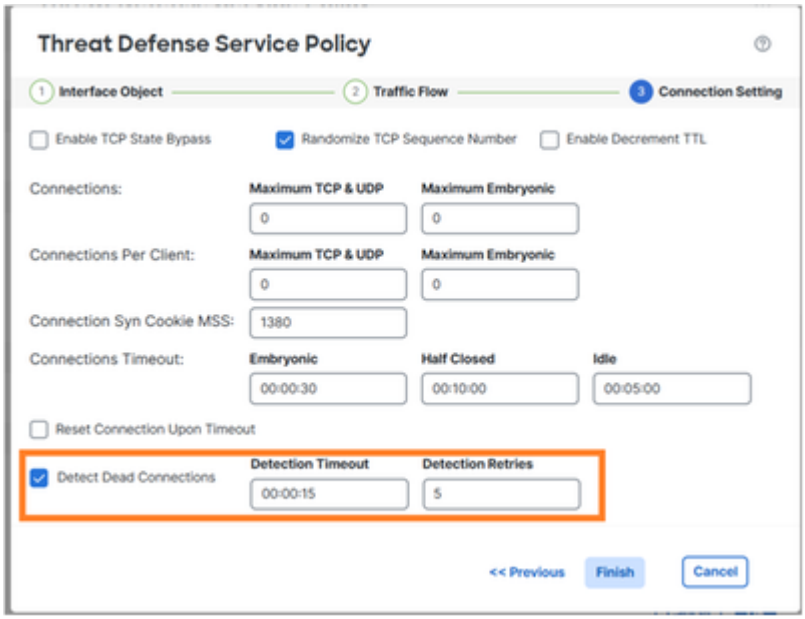
Solution

DCD est documenté à l'adresse :

https://www.cisco.com/c/en/us/td/docs/security/secure-firewall/management-center/device-config/100/management-center-device-config-10-0/advanced-access-service-policies.html#id_71048

1. Accédez à Objets > Liste d'accès et créez une liste d'accès qui correspond au trafic intéressant.

2. Modifiez l'ACP affecté à votre pare-feu, accédez aux options avancées et sélectionnez Threat Defense Service Policy pour activer DCD :



La configuration déployée :

```
access-list DCD_ACL extended permit object-group ProxySG_ExtendedACL_81604390279 any host 172.16.3.1
!
class-map class_map_DCD_ACL
 match access-list DCD_ACL
policy-map global_policy
 class class_map_DCD_ACL
  set connection timeout dcd
```

Comment ça fonctionne

Configurez les captures FTD pour voir le fonctionnement du back-end :

<#root>

```
firewall#
```

capture CAPI interface INSIDE match tcp host 172.16.3.1 any

```
firewall#
```

capture CAPO interface OUTSIDE1 match tcp host 172.16.3.1 any

Établissez une connexion TCP via le pare-feu :

```
<#root>
firewall#
show conn long address 172.16.3.1 | begin 172.16.3.1

TCP OUTSIDE1: 172.16.3.1/23 (172.16.3.1/23) INSIDE: 192.0.2.99/23241 (192.0.2.99/23241), flags UIO N1N7
idle 1m18s
, uptime 1m22s,
timeout 5m0s
, bytes 129, flow id 127, Snort id 4, rule id 268439559, Rx-RingNum 13, Internal-Data0/1
  Initiator: 192.0.2.99, Responder: 172.16.3.1

  DCD probes sent: Initiator 0, Responder 0 Connection lookup keyid: 76292550
```

Initialement, les captures de pare-feu ne contiennent aucun paquet DCD :

```
<#root>
firewall#
show capture

capture CAPI type raw-data interface INSIDE [
Capturing - 0 bytes
]
  match tcp host 172.16.3.1 any
capture CAPO type raw-data interface OUTSIDE1 [
Capturing - 0 bytes
]
  match tcp host 172.16.3.1 any
```

Lorsqu'une connexion inactive atteint le délai d'inactivité, le FTD envoie des messages ACK TCP usurpés à la source et à la destination :

<#root>

```
firewall#

show conn long address 172.16.3.1 | begin 172.16.3.1


TCP OUTSIDE1: 172.16.3.1/23 (172.16.3.1/23) INSIDE: 192.0.2.99/23241 (192.0.2.99/23241), flags UIO N1N7

idle 4m59s

, uptime 5m3s, timeout 5m0s, bytes 129, flow id 127, Snort id 4, rule id 268439559, Rx-RingNum 13, Inte
  Initiator: 192.0.2.99, Responder: 172.16.3.1
  DCD probes sent: Initiator 0, Responder 0 Connection lookup keyid: 76292550

firewall#

show conn long address 172.16.3.1 | begin 172.16.3.1


TCP OUTSIDE1: 172.16.3.1/23 (172.16.3.1/23) INSIDE: 192.0.2.99/23241 (192.0.2.99/23241), flags UIO N1N7

idle 0s

, uptime 5m3s, timeout 15s, bytes 129, flow id 127, Snort id 4, rule id 268439559, Rx-RingNum 13, Inter
  Initiator: 192.0.2.99, Responder: 172.16.3.1


DCD probes sent: Initiator 1

, Responder 0 Connection lookup keyid: 76292550

firewall#

 show conn long address 172.16.3.1 | begin 172.16.3.1


TCP OUTSIDE1: 172.16.3.1/23 (172.16.3.1/23) INSIDE: 192.0.2.99/23241 (192.0.2.99/23241), flags UIO N1N7
  Initiator: 192.0.2.99, Responder: 172.16.3.1


DCD probes sent: Initiator 1, Responder 1

 Connection lookup keyid: 76292550
```

Si les deux réponses, le compteur d'inactivité est réinitialisé :

```
<#root>
firewall#

show capture CAPI



3 packets captured

1: 09:01:30.433952 802.1Q vlan#101 P0 172.16.3.1.23 > 192.0.2.99.23241: . ack 3271882019 win 32757
2: 09:01:30.434334 802.1Q vlan#101 P0

192.0.2.99.23241 > 172.16.3.1.23: . ack 1746306341 win 32746



3: 09:01:30.955654 802.1Q vlan#101 P0 172.16.3.1.23 > 192.0.2.99.23241: . ack 3271882019 win 32757
3 packets shown
firewall#

 show capture CAPO



3 packets captured

1: 09:01:30.434364 802.1Q vlan#201 P0 192.0.2.99.23241 > 172.16.3.1.23: . ack 111661490 win 32746
2: 09:01:30.955288 802.1Q vlan#201 P0 192.0.2.99.23241 > 172.16.3.1.23: . ack 111661490 win 32746
3: 09:01:30.955639 802.1Q vlan#201 P0

172.16.3.1.23 > 192.0.2.99.23241: . ack 3875469573 win 32757



3 packets shown

firewall#

show conn long address 172.16.3.1 | begin 172.16.3.1



TCP OUTSIDE1: 172.16.3.1/23 (172.16.3.1/23) INSIDE: 192.0.2.99/23241 (192.0.2.99/23241), flags UIO N1N7

idle 1m29s


, uptime 6m33s, timeout 5m0s, bytes 129, flow id 127, Snort id 4, rule id 268439559, Rx-RingNum 13, Int
  Initiator: 192.0.2.99, Responder: 172.16.3.1
  DCD probes sent: Initiator 1, Responder 1 Connection lookup keyid: 76292550
```



Remarque : DCD ne fonctionne pas sur les connexions déchargées (indicateur « o »).

Informations connexes

https://www.cisco.com/c/en/us/td/docs/security/secure-firewall/management-center/device-config/100/management-center-device-config-10-0/advanced-access-service-policies.html#id_71048

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.