使用安全FTD和ASA配置安全访问，以通过NAT进行私有访问

目录

简介

本文档介绍如何使用网络地址转换配置安全访问网络隧道组。

先决条件

要求

Cisco 建议您了解以下主题：

- 思科防火墙管理中心
- 思科安全防火墙威胁防御
- 思科安全访问
- Cisco 自适应安全设备
- 网络地址转换

- 基于策略的路由
- 防火墙上的数据包捕获

使用的组件

本文档中的信息基于：

- 思科防火墙管理中心7.10
- 思科安全防火墙威胁防御7.10
- 思科安全访问
- 思科自适应安全设备9.22
- Cisco 路由器

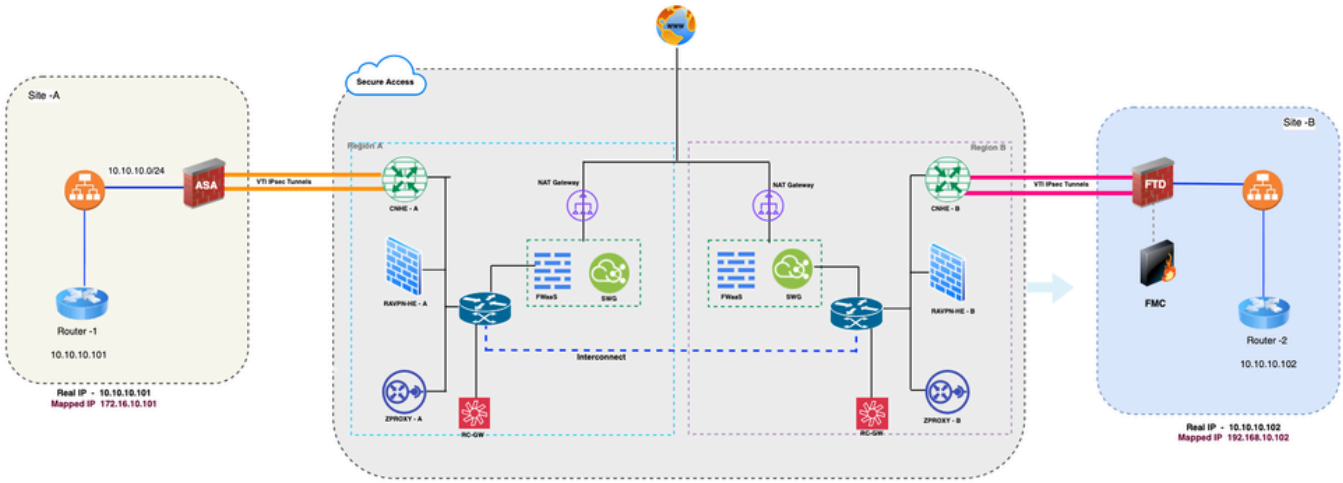
本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

思科安全访问提供云原生安全服务边缘功能，包括安全互联网访问(SIA)和安全私有访问(SPA)。 对于将私有应用访问扩展到远程用户而不回传通过数据中心的所有流量的组织，安全访问使用IPsec网络隧道组(NTG)在本地网络设备(例如思科防火墙威胁防御(FTD)、自适应安全设备(ASA))和思科安全访问云入网点(PoPs)之间创建加密隧道。

本文档详细介绍如何使用IKEv2在Cisco FTD、ASA和Cisco Secure Access之间建立基于路由的IPsec隧道，从而在FTD和ASA上启用安全访问上的网络地址转换(NAT)和基于策略的路由(PBR)以仅引导受专用访问限制的流量进入隧道。

网络图



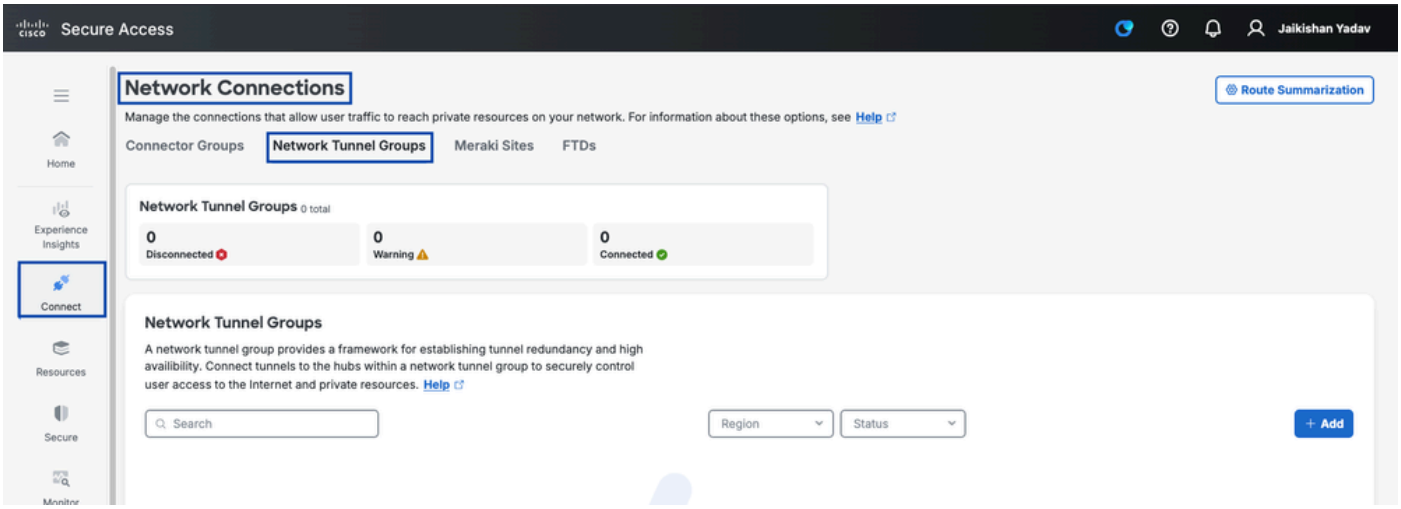
网络拓扑

配置

安全访问和安全防火墙线程防御 — 基于PBR的动态(BGP)路由VPN

在安全访问中配置网络隧道组

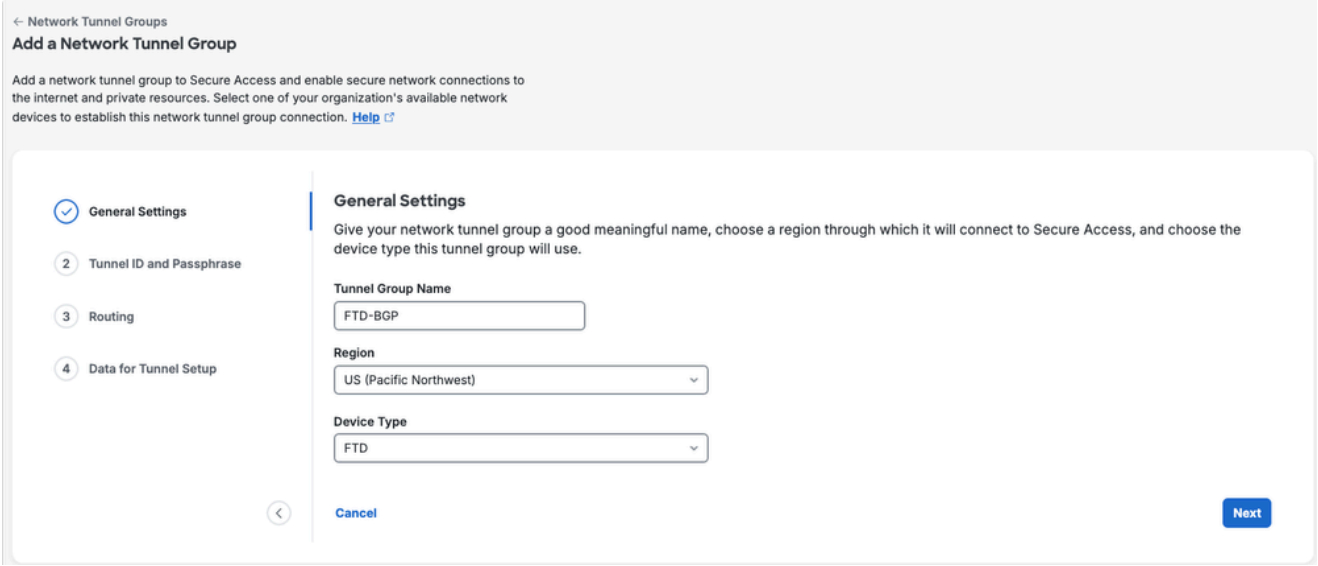
1.登录安全访问控制面板安全访问
2.导航到Connect > Network Connections > Network Tunnel Groups，然后单击+Add配置网络隧道组



安全访问 — 网络隧道组

3.配置网络隧道组 — 常规设置

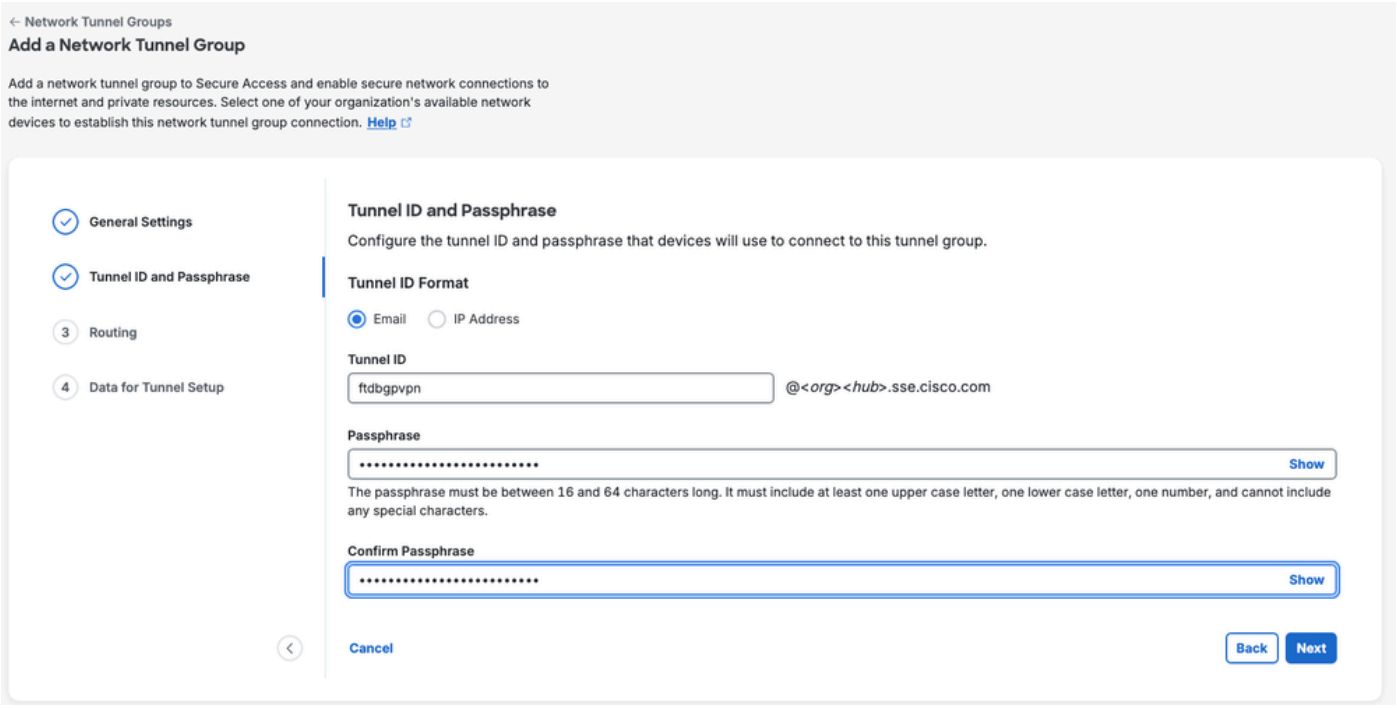
- 配置隧道组名称、区域和设备类型
- 单击“下一步”



安全访问 — 网络隧道组常规设置

4.配置隧道ID和口令。

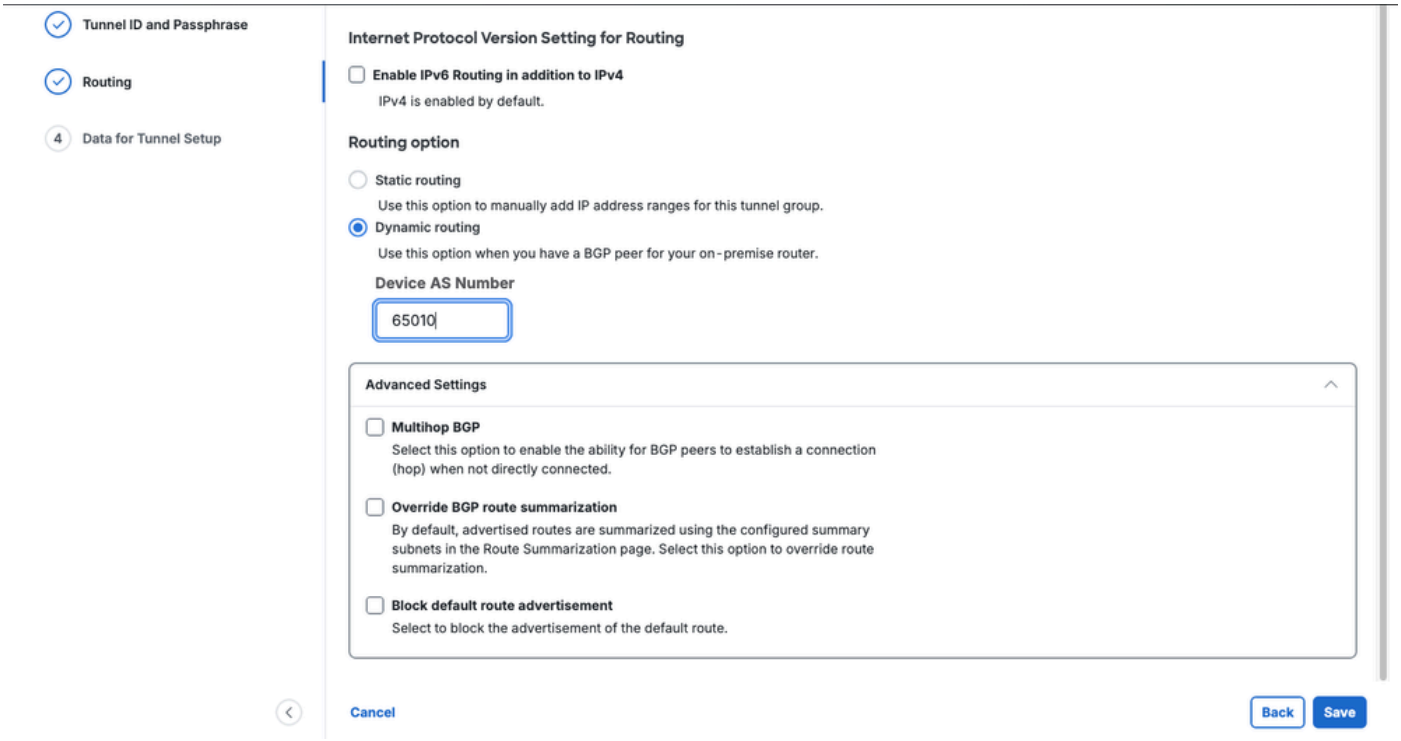
- 配置隧道ID和口令。
- 点击“下一步”



安全访问 — 网络隧道组ID和口令

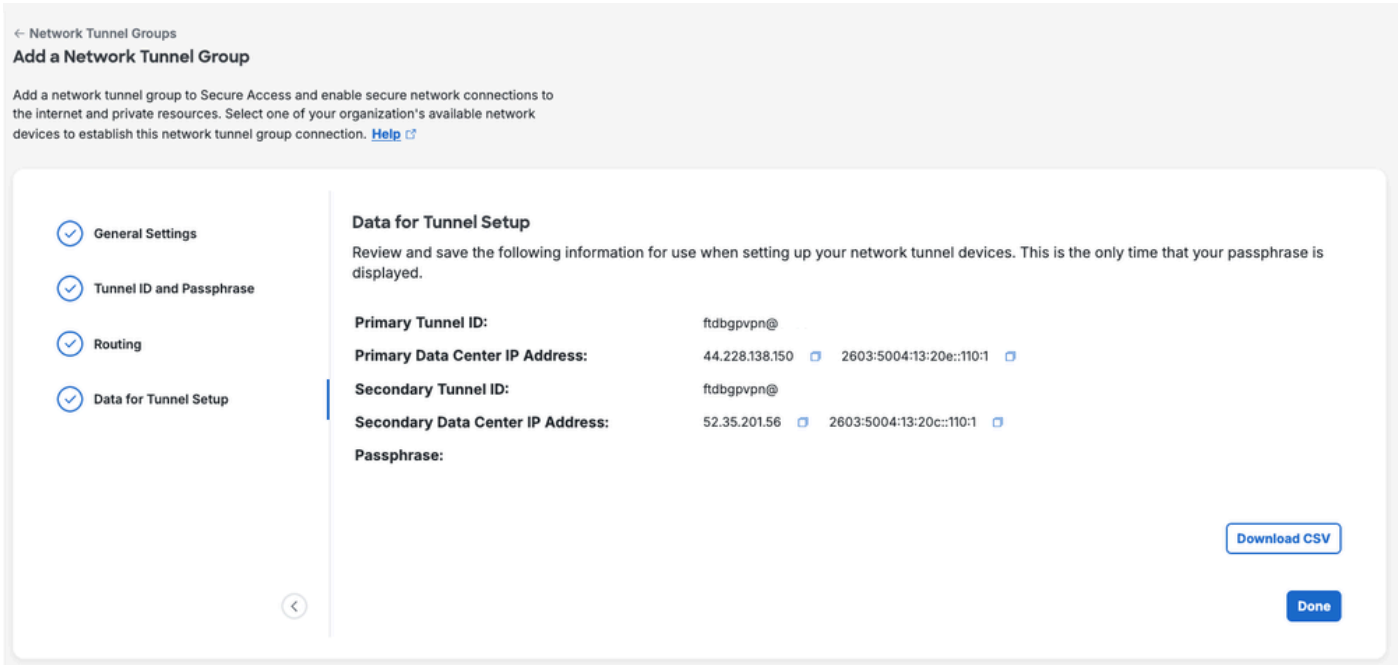
5.配置路由

- 配置设备AS编号
- 点击保存



安全访问 — 网络隧道组路由

6.保存网络隧道组配置

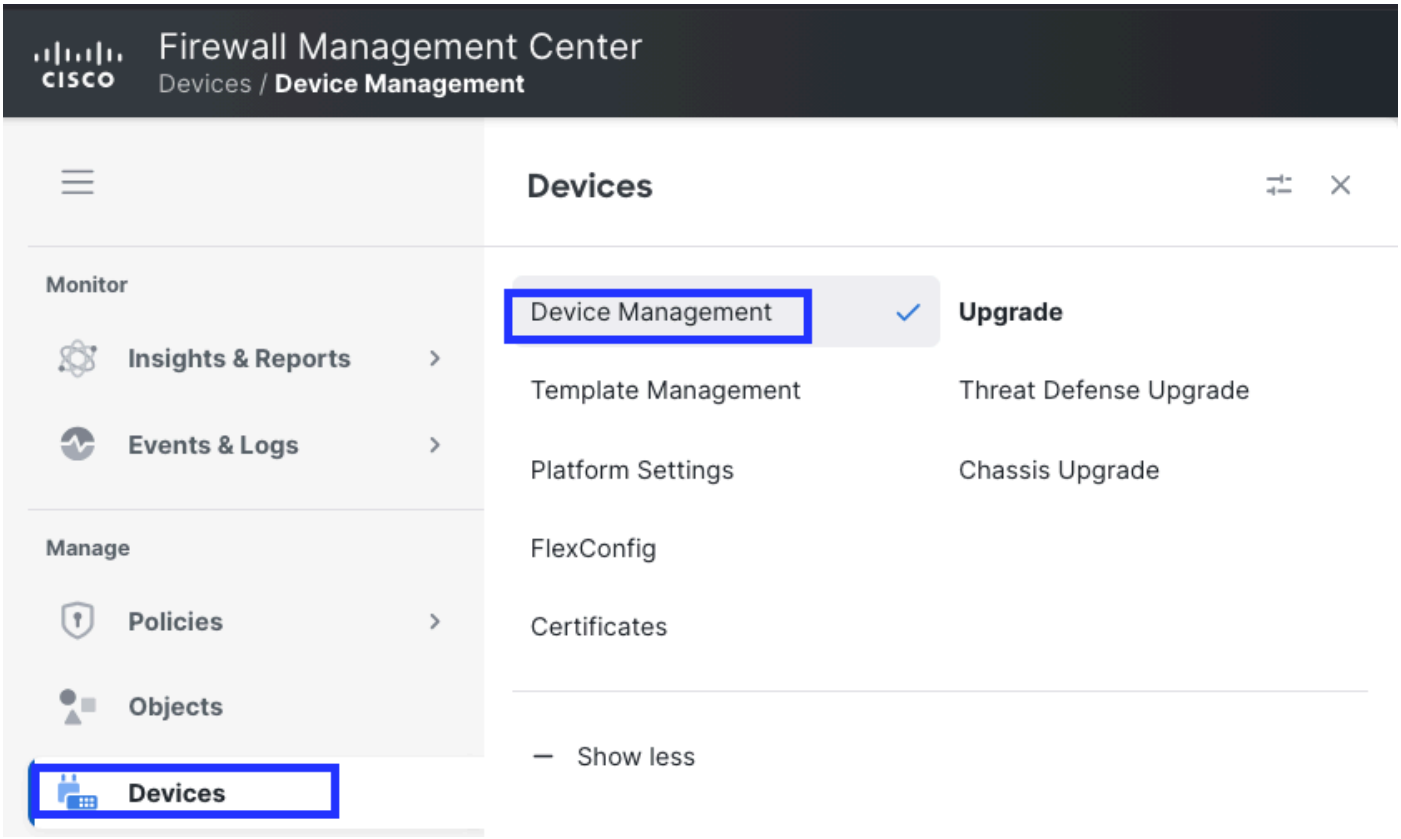


在具有PBR的安全防火墙威胁防御(FTD)上配置基于动态路由的IPsec VPN

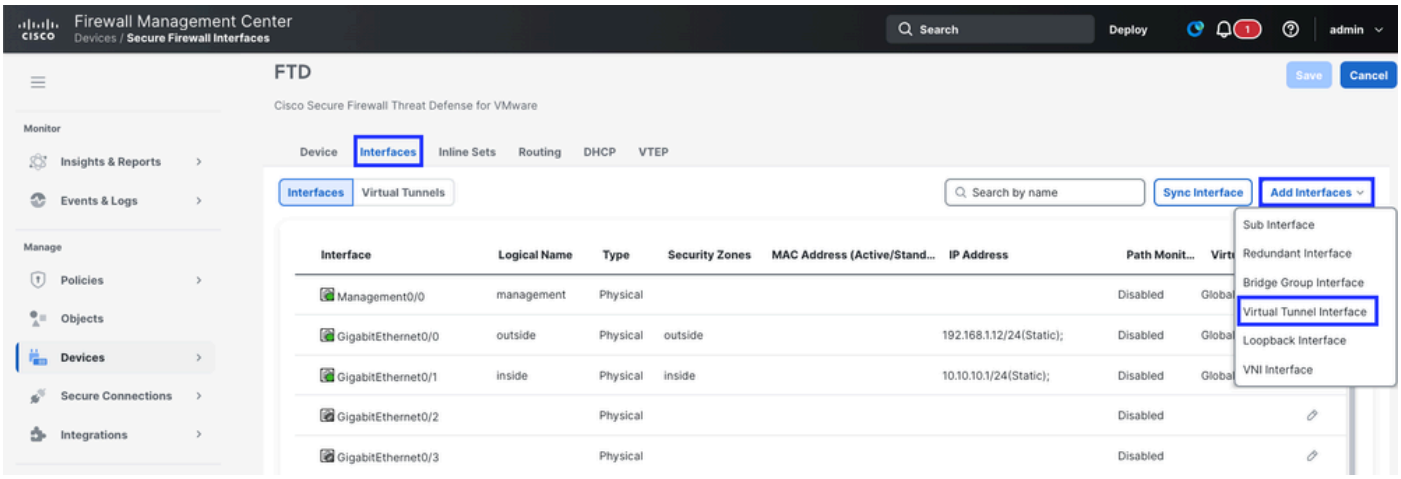
1.登录防火墙管理中心(FMC)

2.配置虚拟隧道接口VTI

- 导航到Devices > Device Management



- 点击设备旁边的铅笔图标，导航到Interface（接口）部分
- 单击Add interfaces并选择Virtual Tunnel Interface



安全防火墙管理 — FTD VTI配置

- **配置VTI**

Add Virtual Tunnel Interface

General | Path Monitoring

Tunnel Type

(•) Static ( ) Dynamic

Name:*

VTI-1

[x] Enabled

Description:

Virtual Tunnel interface for Primary VTI VPN

Security Zone:

vti

Priority:

0 *(0 - 65535)*

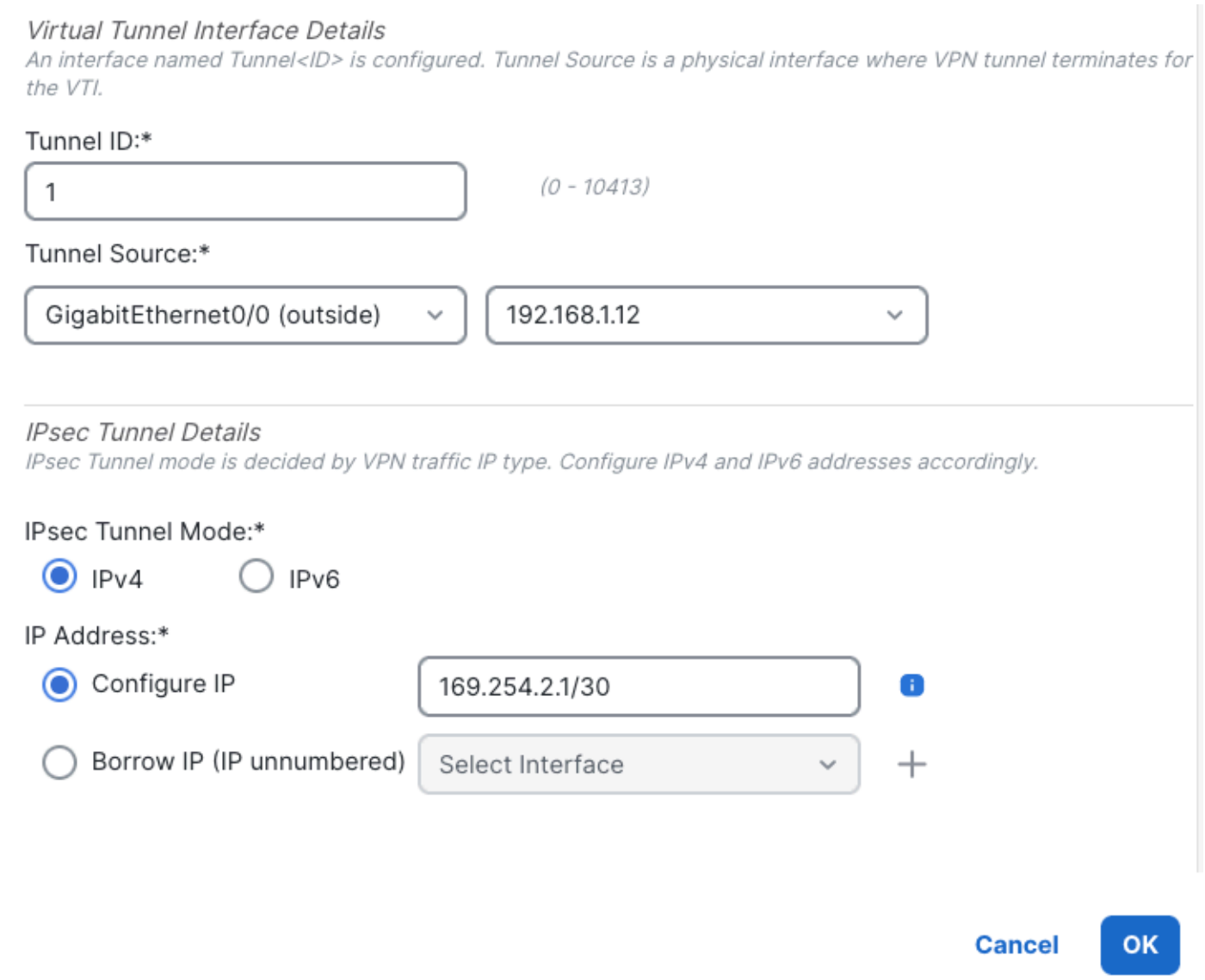
Propagate Security Group Tag: [x]

Virtual Tunnel Interface Details

An interface named Tunnel<ID> is configured. Tunnel Source is a physical interface where VPN tunnel terminates for the VTI.

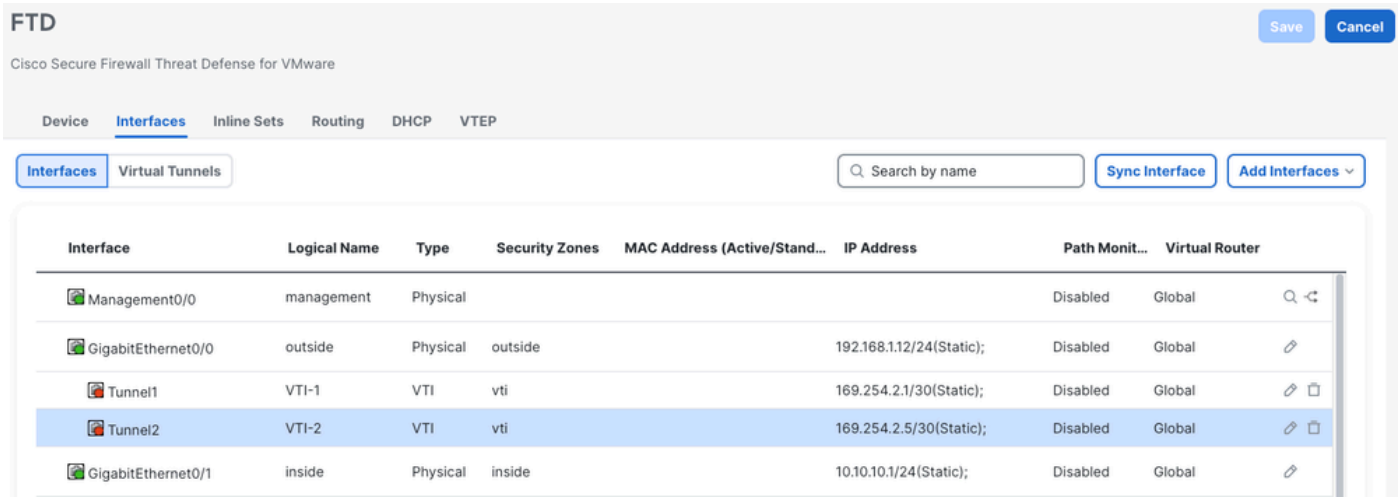
Tunnel ID:*

1 *(0 - 10413)*

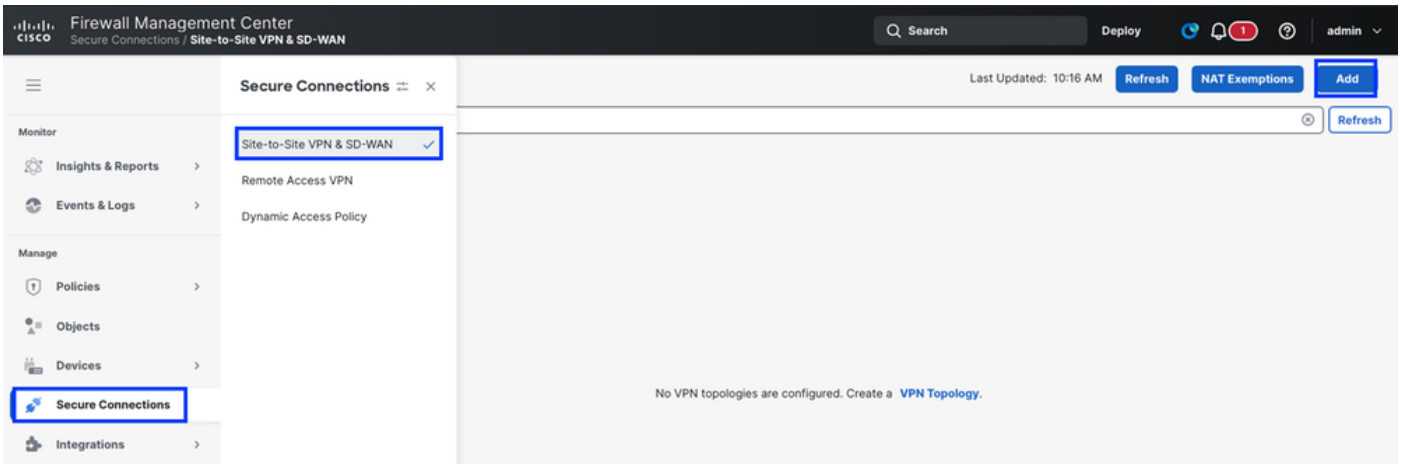


安全防火墙管理 — FTD VTI配置

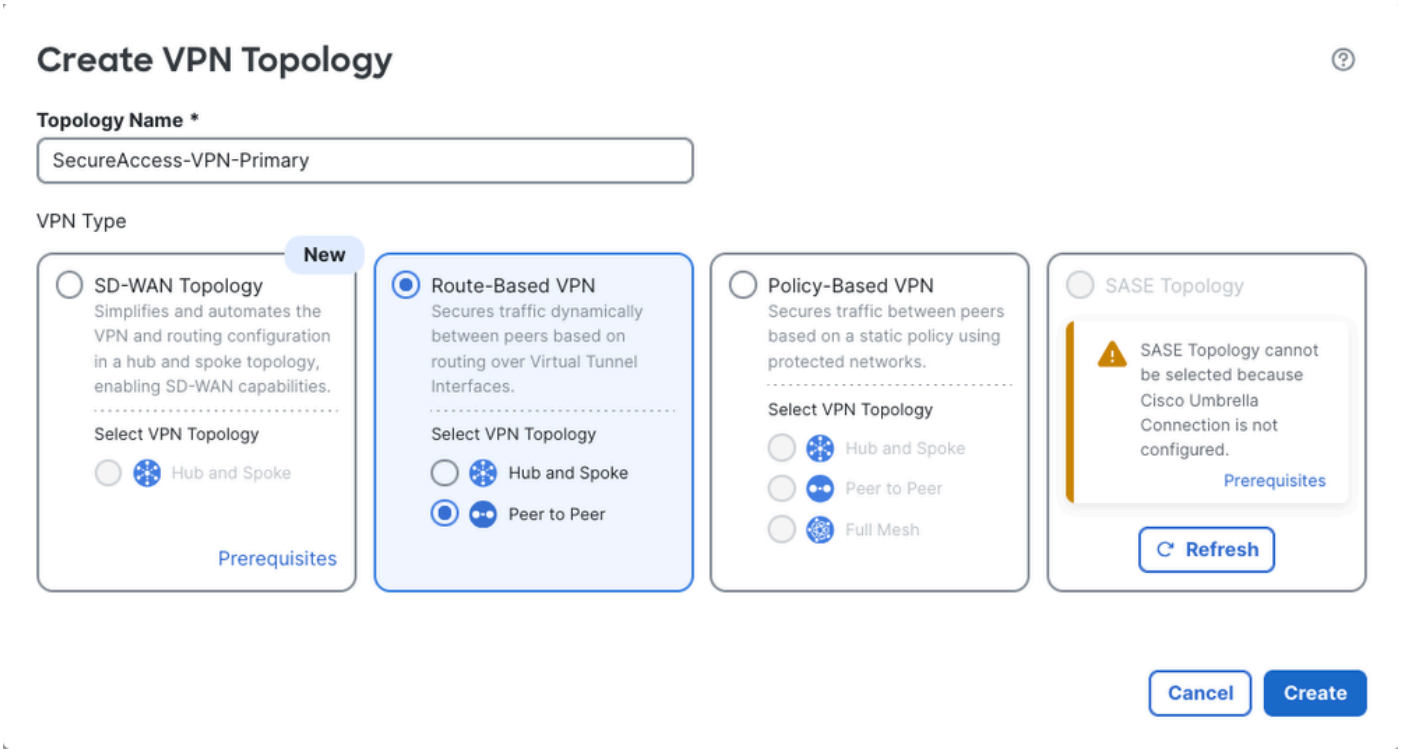
- 配置VTI-2以及具有安全访问的辅助IPsec VPN



3.导航到Secure Connections > Site-to-Site VPN & SD-WAN，然后单击Add以配置VPN

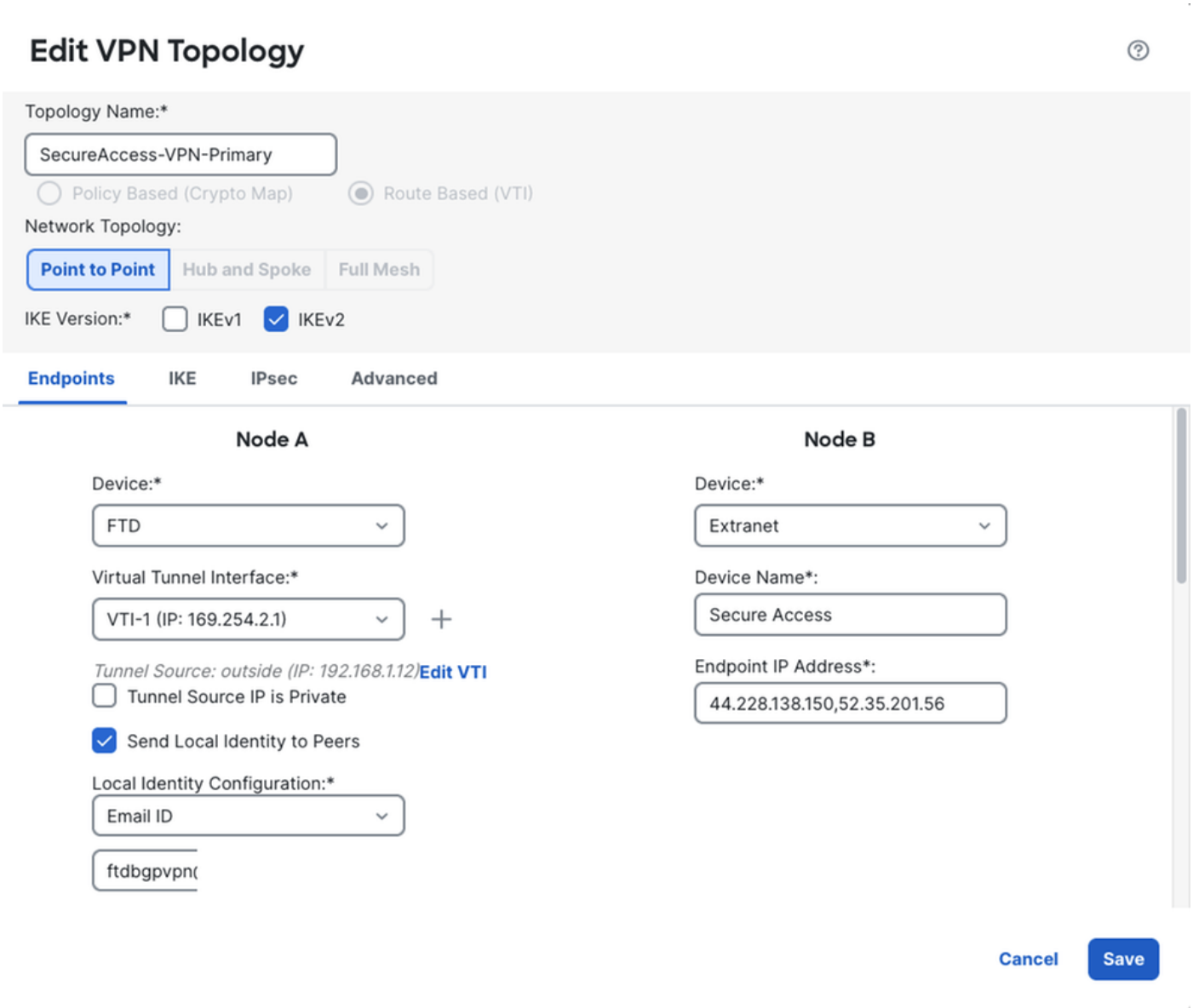


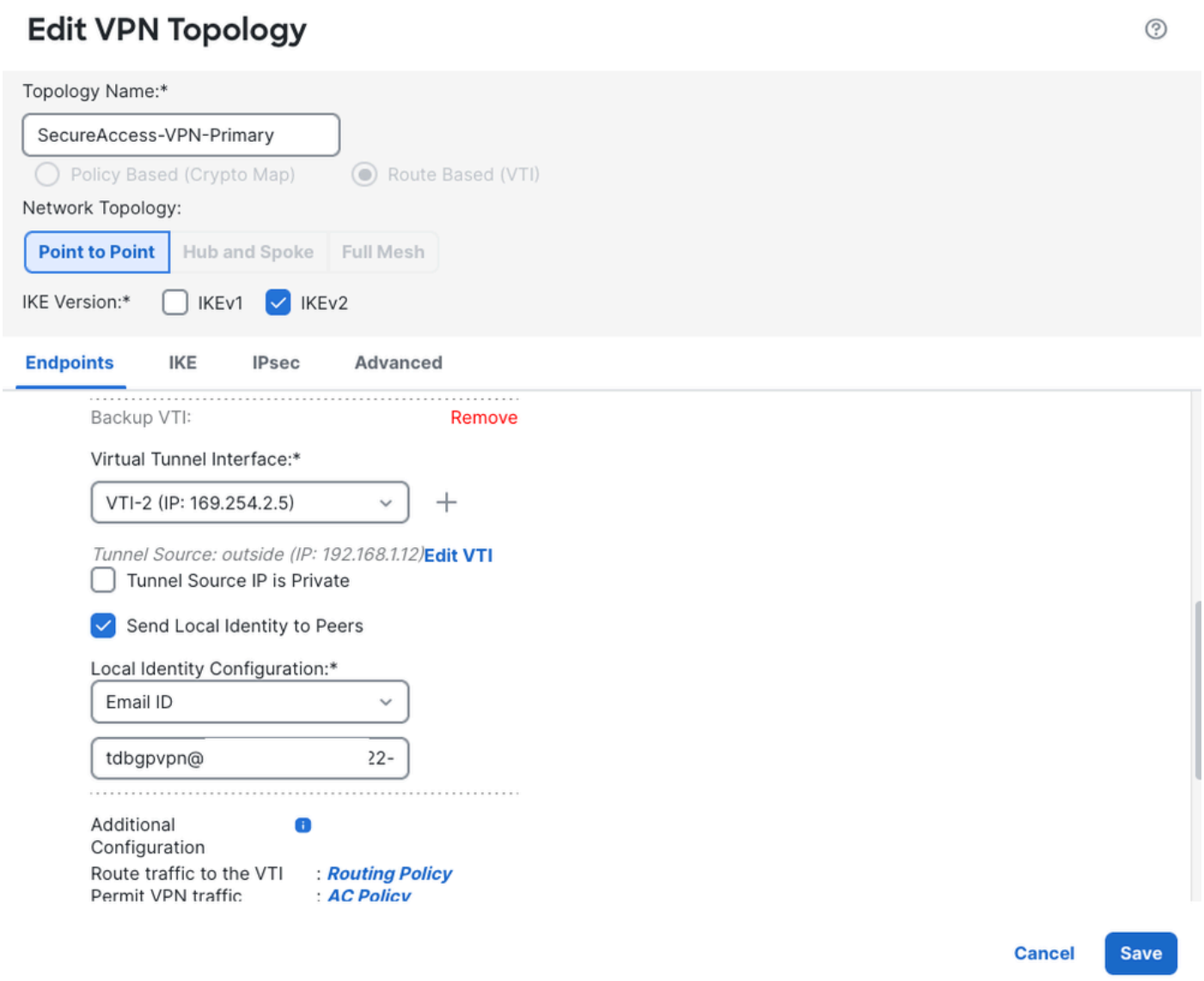
- 创建VPN拓扑



- 从Configure Network Tunnel Group on Secure Access的第6步获取主要和辅助数据中心的隧道ID和IP地址。

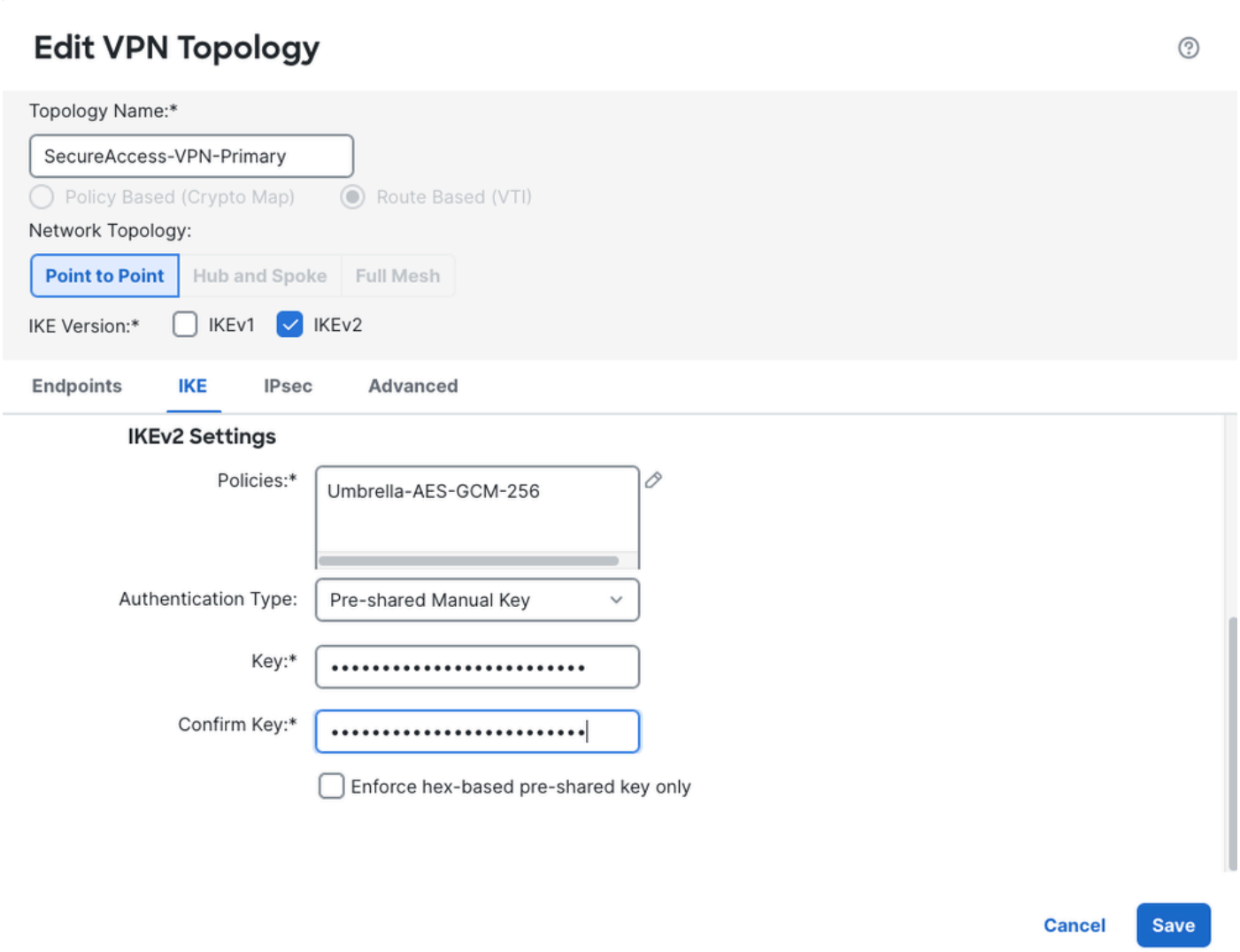
- 点击Endpoints
- 在节点A下，点击设备并选择您的FTD设备
- 点击Virtual Tunnel Interface并选择上一步中创建的第一个VTI接口
- 点击Send Local Identity to Peers选项并选择Email ID，输入主隧道ID（从Configure Network Tunnel Group on Secure Access下的“Save Network Tunnel Group Configuration”中）
- 在Node B下，点击Device并选择Extranet
- 点击Device Name并为其指定名称
- 点击Endpoint IP Addresses（终端IP地址），然后输入安全访问主和辅助IP地址，以逗号分隔(在Configure Network Tunnel Group on Secure Access下“Save Network Tunnel Group Configuration”（保存网络隧道组配置））
- 点击Add Backup VTI
- 点击Virtual Tunnel Interface并选择上一步中创建的第二个VTI接口
- 点击Send Local Identity to Peers选项并选择Email ID，输入辅助隧道ID（从Configure Network Tunnel Group on Secure Access下的“Save Network Tunnel Group Configuration”中）
- 点击Save





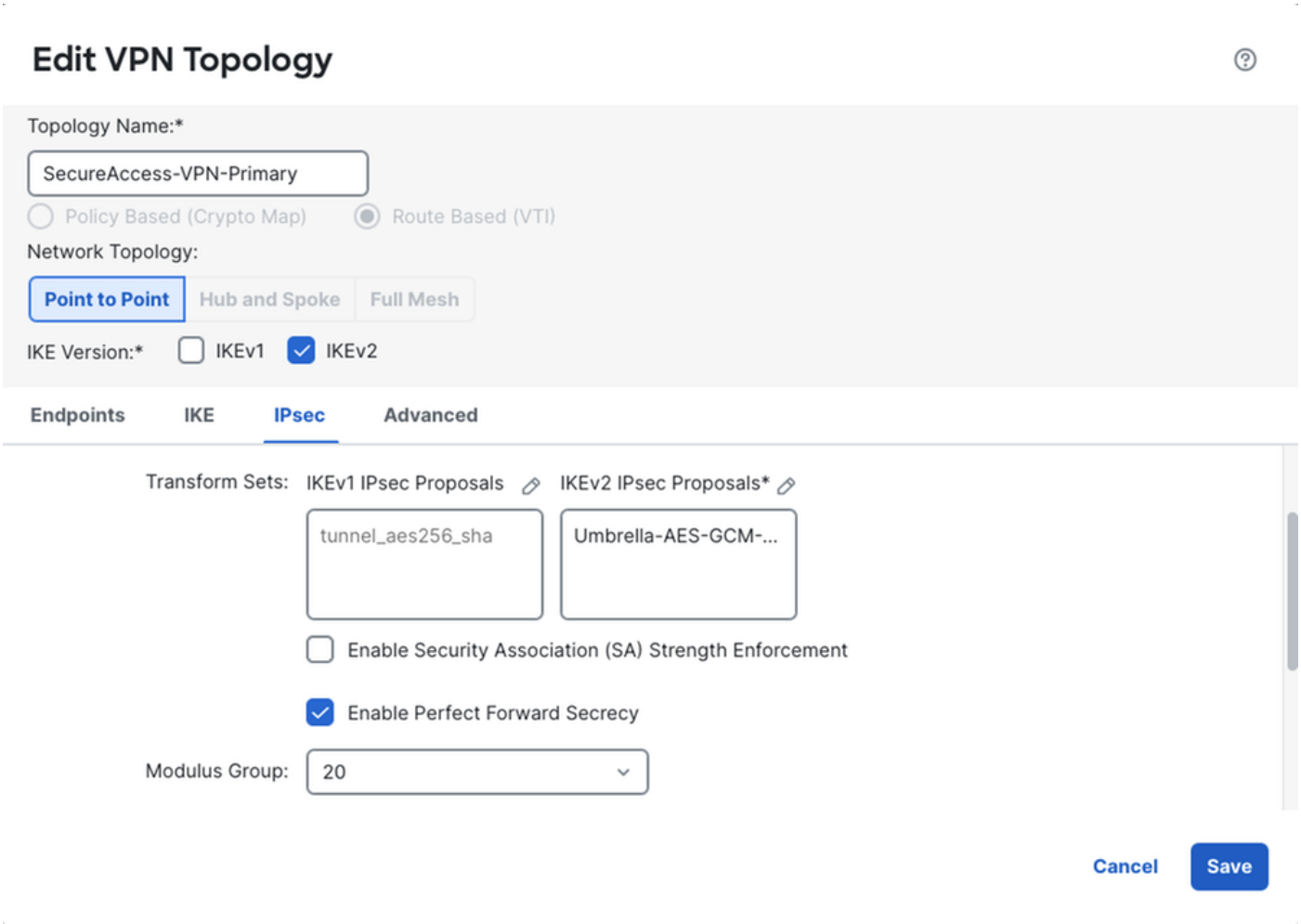
安全FTD - VPN终端配置

- 根据受支持的IPsec参数配置IKEv2设置
  - 选择Policies as Umbrella-AES-GCM-256
  - 选择Authentication Type作为Pre-Shared Manual Key



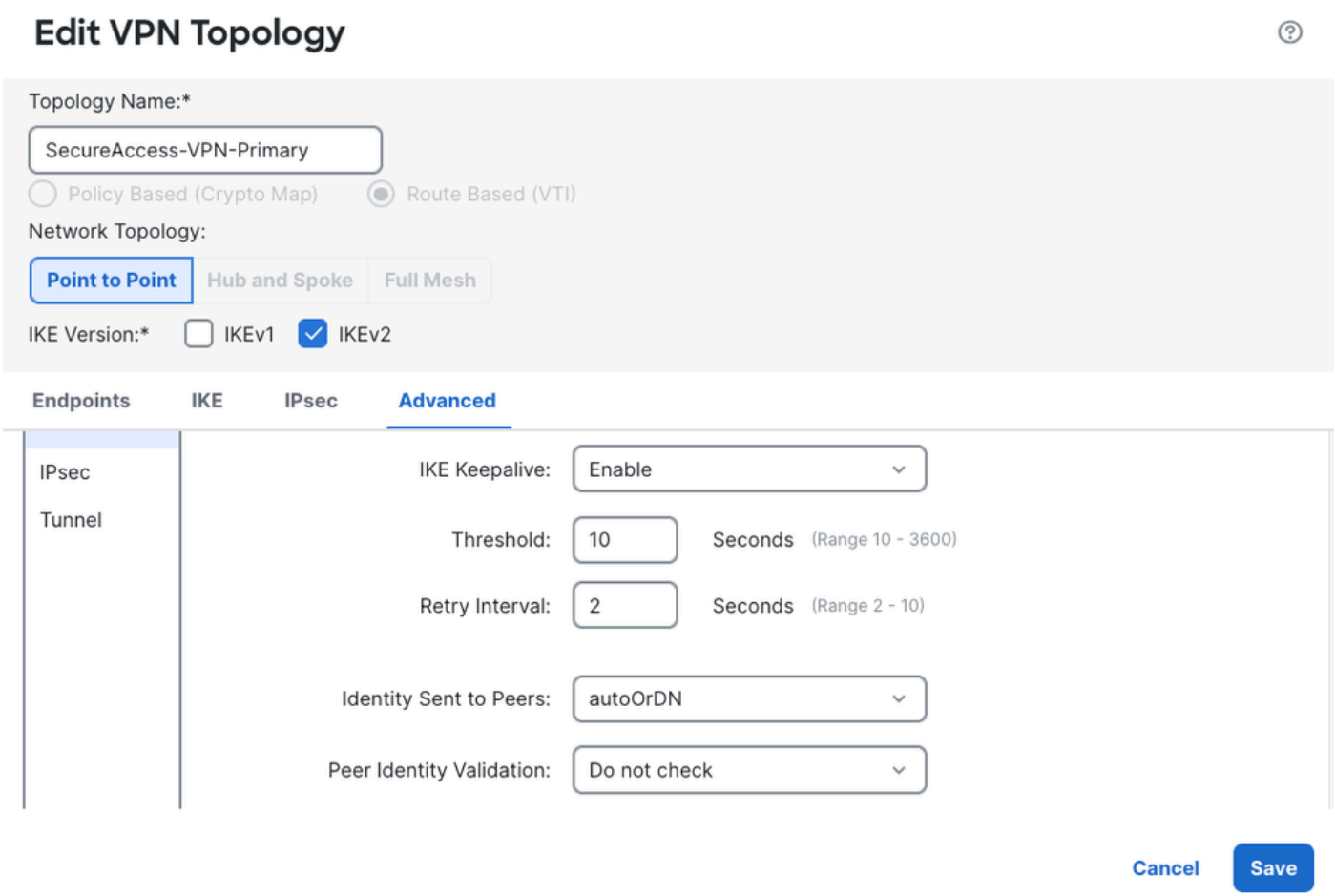
安全FTD - VPN IKE设置

- 配置IPsec设置
  - 选择IKEv2 IPsec Proposals作为Umbrella-AES-GCM-256
  - 启用PFS作为选择模数组作为20



安全FTD - VPN IPsec设置

- 高级设置
- 点击保存



安全FTD - VPN高级设置

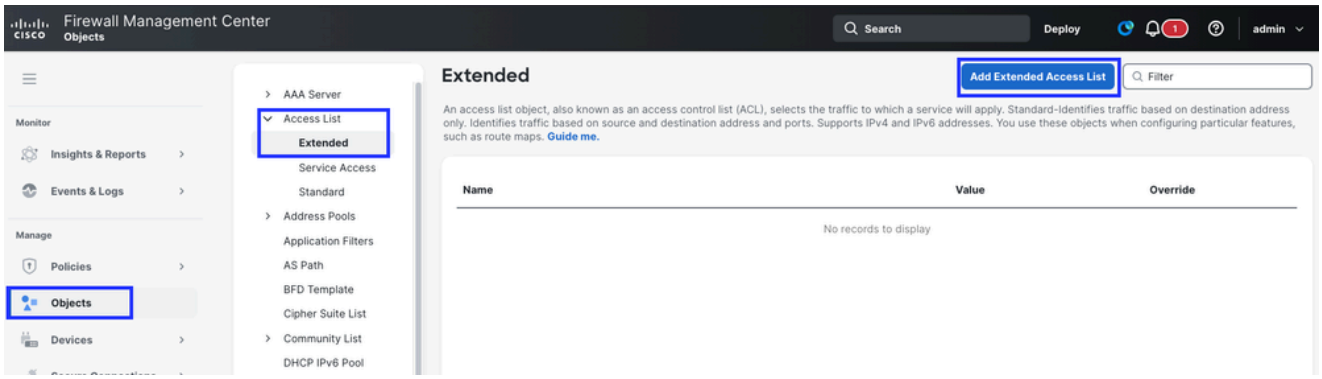
基于FTD策略的路由

在传统路由中，数据包根据目的IP地址进行路由。在基于目标的路由系统中更改特定流量的路由很困难。基于策略的路由(PBR)扩展和补充了路由协议提供的机制，让您能够更好地控制路由。

PBR允许您设置IP优先级。它还允许您为特定流量指定路径，例如高开销链路上的优先级流量。使用PBR，您可以根据目标网络以外的条件（例如源端口、目标地址、目标端口、协议、应用程序或这些对象的组合）定义路由。有关详细信息，请参阅基于策略的路由

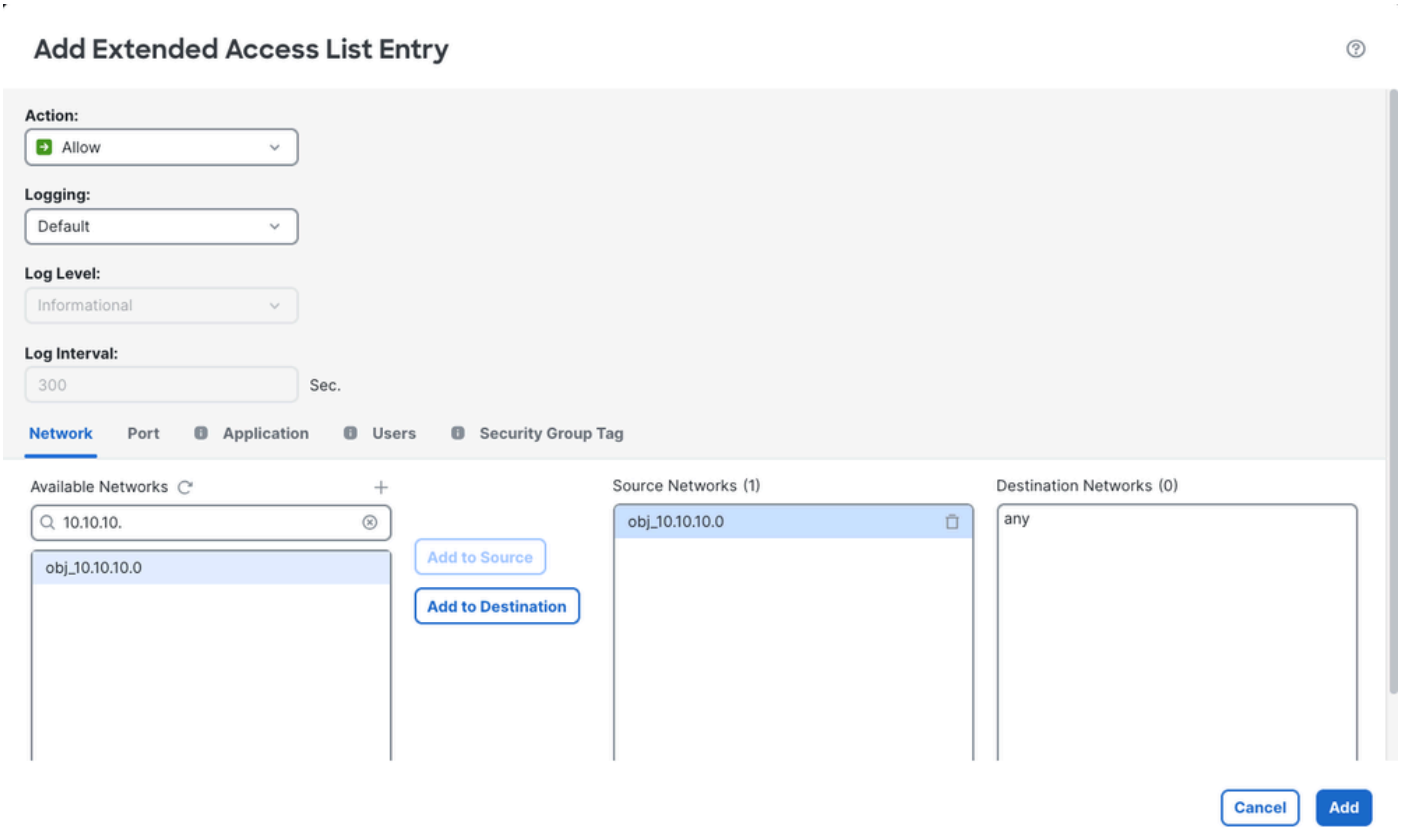
1.配置访问列表

- 导航到对象>访问列表>扩展
- 单击Add Extended Access List

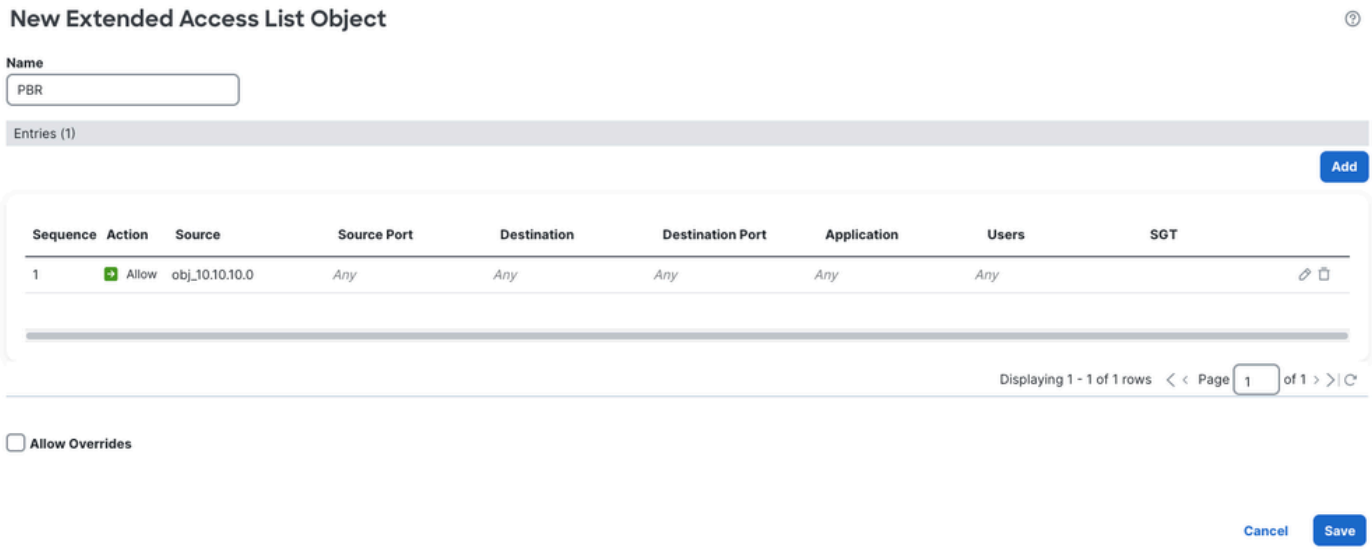


安全FTD — 扩展访问列表

- 提供访问列表名称
- 定义操作
  - 允许. — 定义的流量将发送到IPsec隧道
  - 阻止 — 流量将绕过来自IPsec隧道的流量
  - 将根据序列号匹配规则（从低到高）
  - 点击Add
  - 点击保存



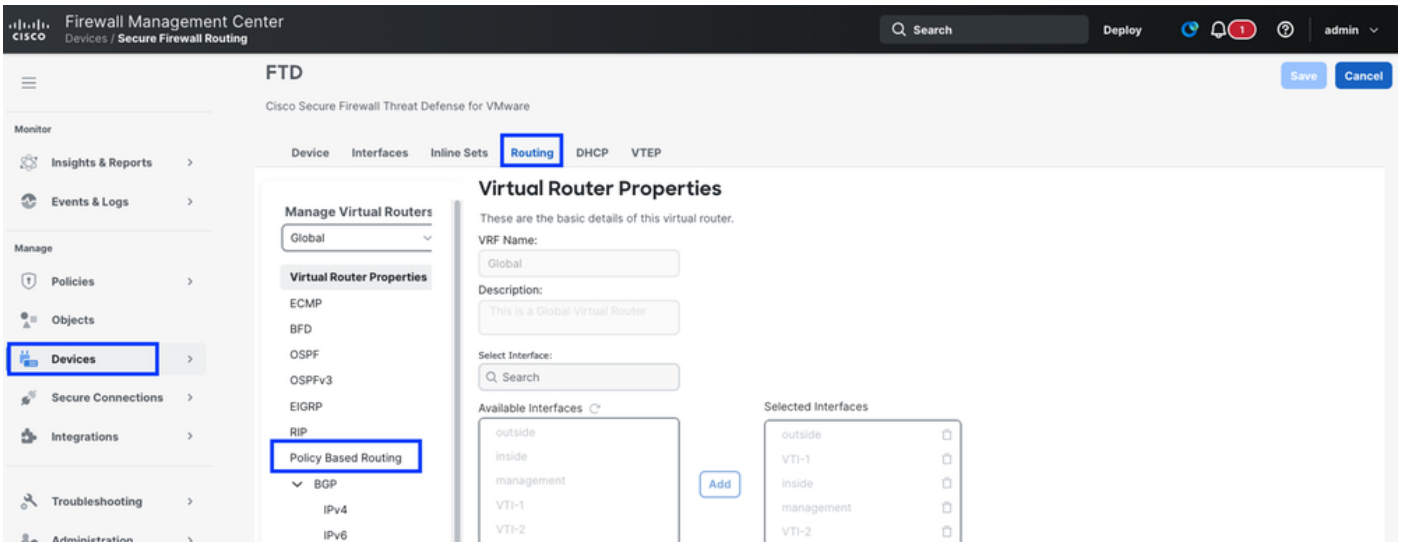
安全FTD — 扩展访问列表



安全FTD — 扩展访问列表

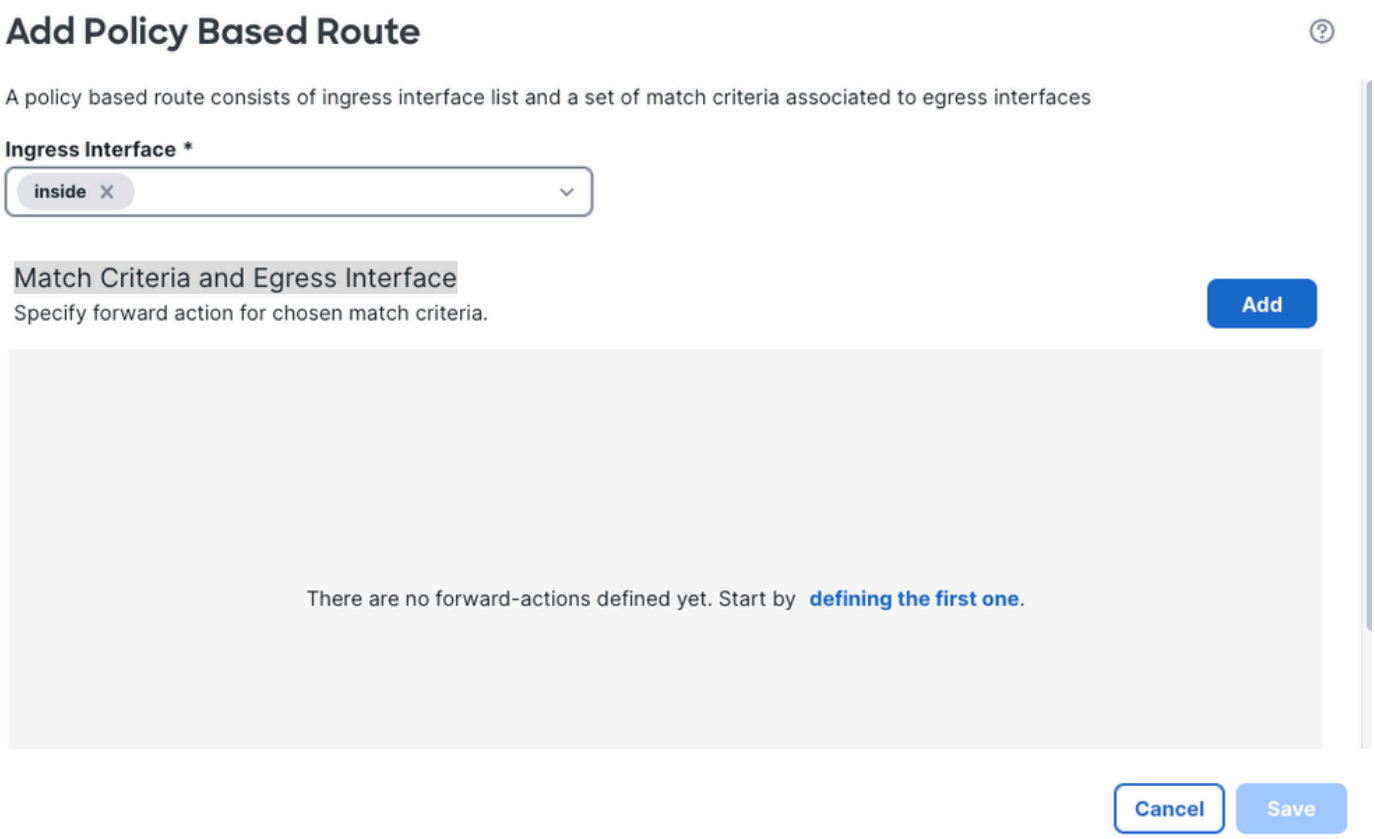
2.配置基于策略的路由

- 导航到设备>设备管理> FTD >路由



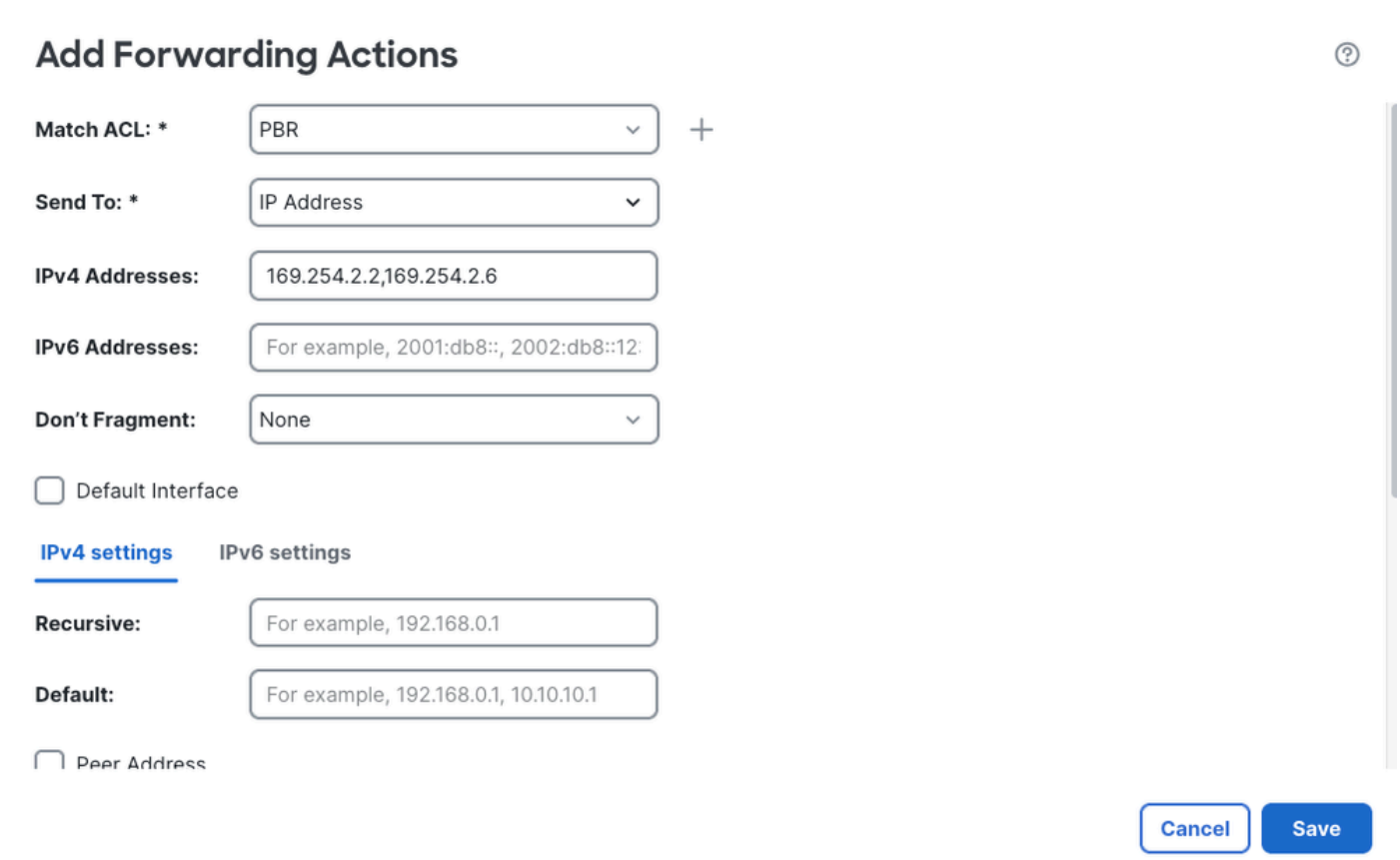
安全FTD — 基于策略的路由

- 点击Add
- 选择Ingress Interface - Ingress interface作为我们在访问列表中定义的子网

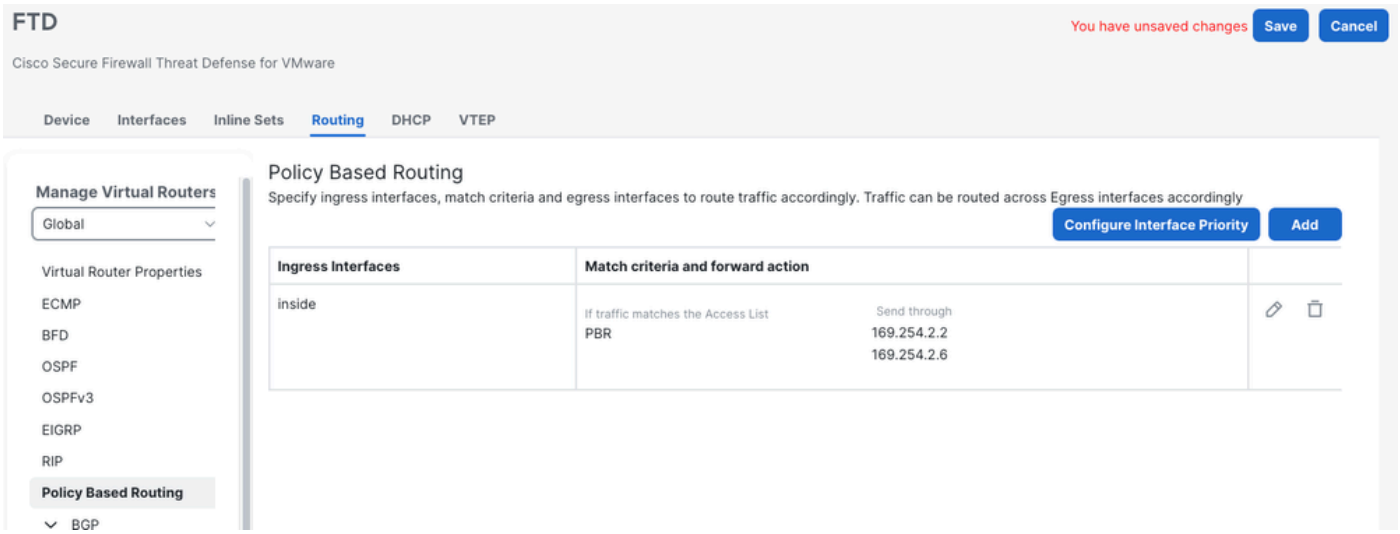


安全FTD — 基于策略的路由

- 定义匹配条件和出口接口
  - 点击Add
  - 选择Match ACL
  - 选择Send to as IP address
  - 添加下一跳IP地址。- VTI接口IP地址为。169.254.2.130和。169.254.2.5/30。因此，VTI -1和VTI-2的下一跳IP地址将是169.254.2.2和。169.254.2.6
  - 单击Save



安全FTD — 基于策略的路由

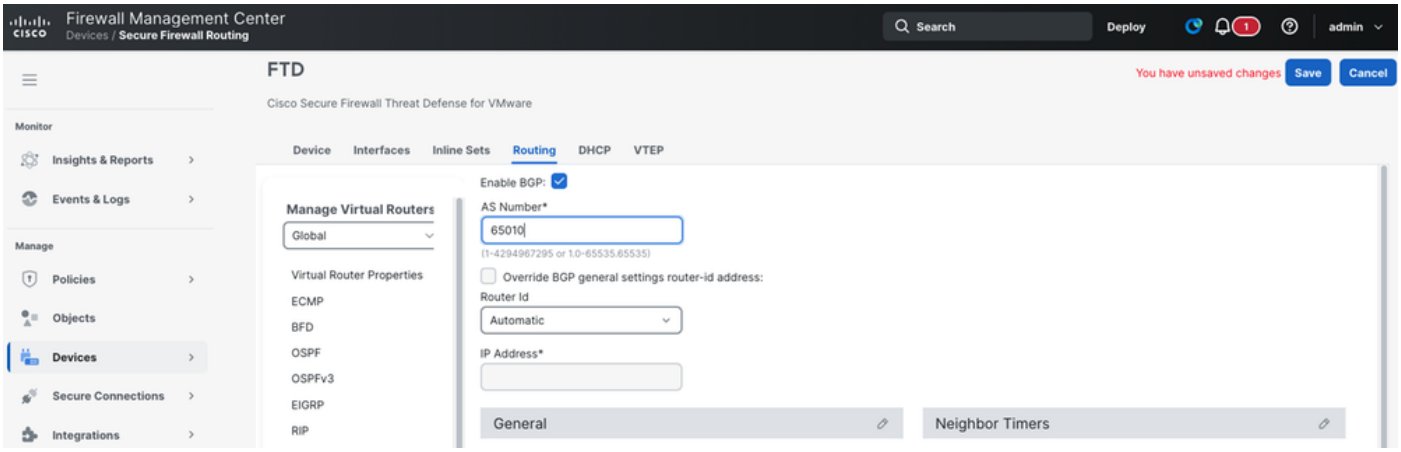


安全FTD — 基于策略的路由

在FTD上配置BGP

1.启用BGP

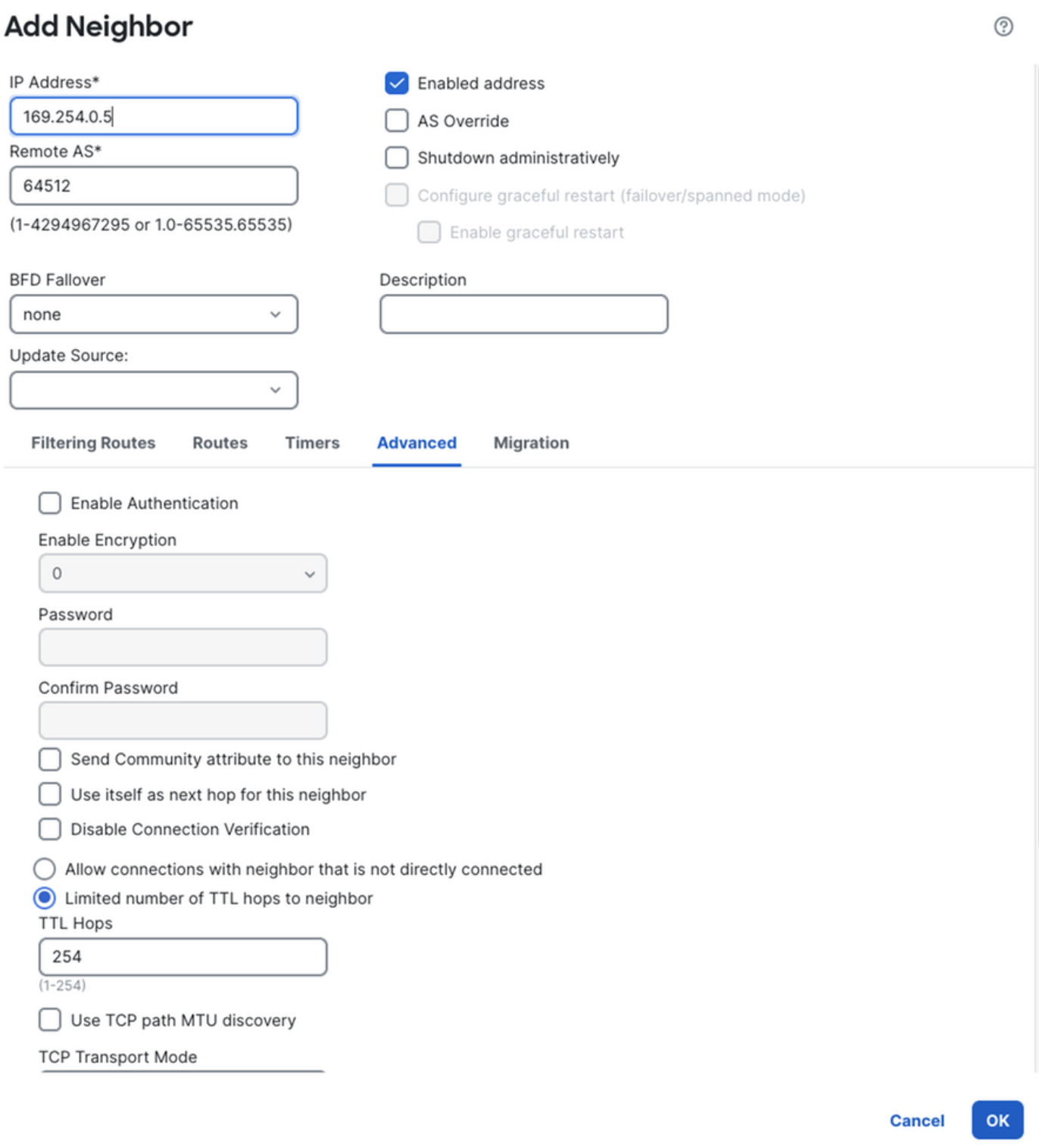
- 导航到设备>设备管理> FTD >路由>常规设置> BGP
- 启用BGP并添加AS编号（FTD本地AS编号）
- 点击保存



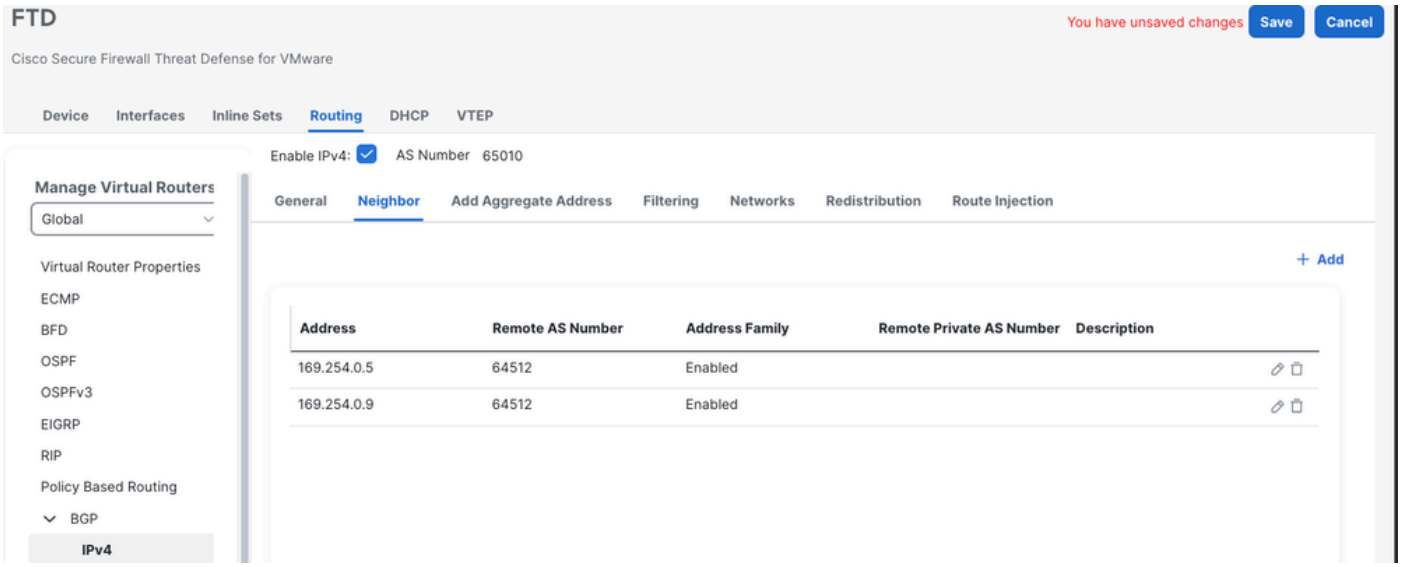
安全FTD - BGP路由

- 导航到BGP > IPv4

2.添加BGP Neighbor — 安全访问BGP对等体为169.254.0.5和。169.254.0.9


Add Neighbor
IP Address*
169.254.0.5
Remote AS*
64512
(1-4294967295 or 1.0-65535.65535)
Enabled address
AS Override
Shutdown administratively
Configure graceful restart (failover/spanned mode)
Enable graceful restart
BFD Fallover
none
Description
Update Source:
Filtering Routes
Routes
Timers
Advanced
Migration
Enable Authentication
Enable Encryption
0
Password
Confirm Password
Send Community attribute to this neighbor
Use itself as next hop for this neighbor
Disable Connection Verification
Allow connections with neighbor that is not directly connected
Limited number of TTL hops to neighbor
TTL Hops
254
(1-254)
Use TCP path MTU discovery
TCP Transport Mode
Cancel
OK

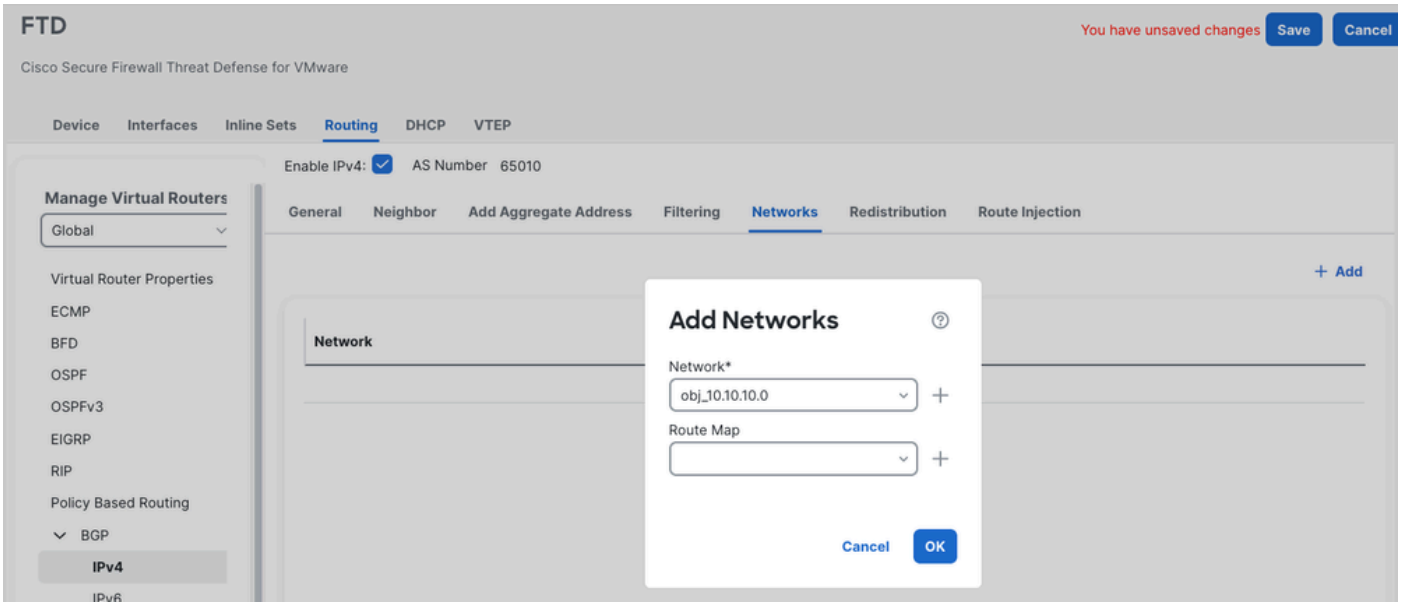
安全FTD - BGP路由



安全FTD - BGP路由

3.添加网络 — 需要通告到安全访问的网络

- 单击 OK

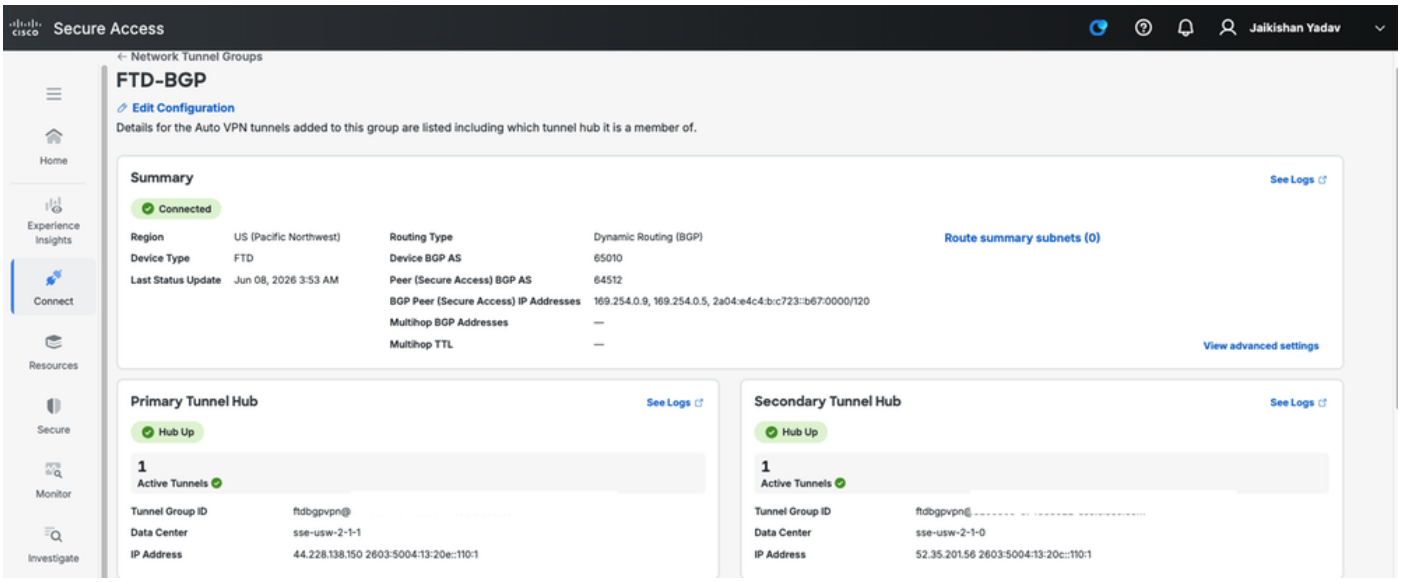


安全FTD - BGP路由

- 保存并部署更改

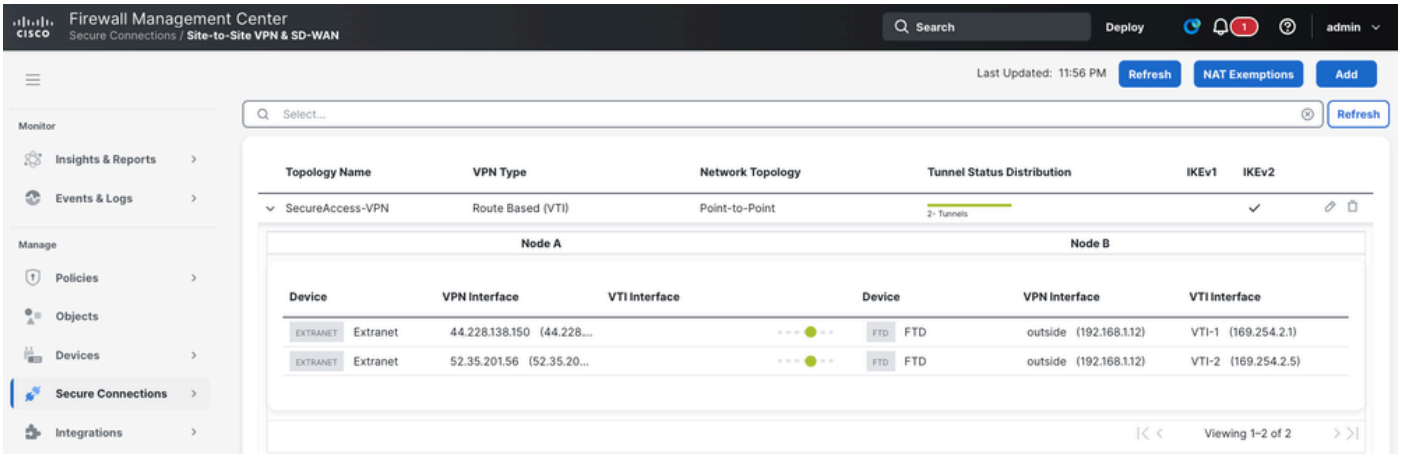
检验隧道状态

论安全访问



安全FTD - IPsec隧道状态

在FMC上

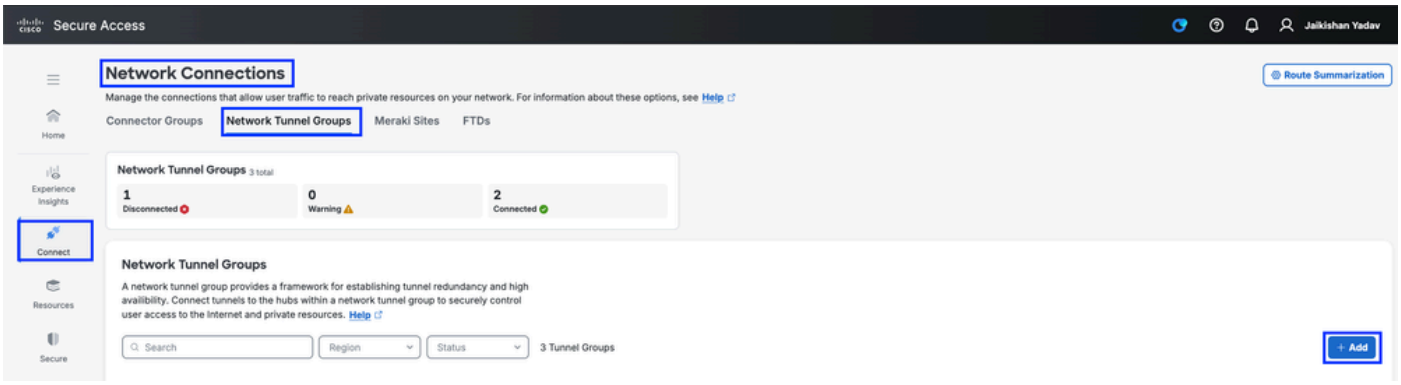


安全FMC - IPsec隧道状态

在带有PBR的自适应安全设备(ASA)上配置基于静态路由的IPsec VPN

安全访问上的VPN配置

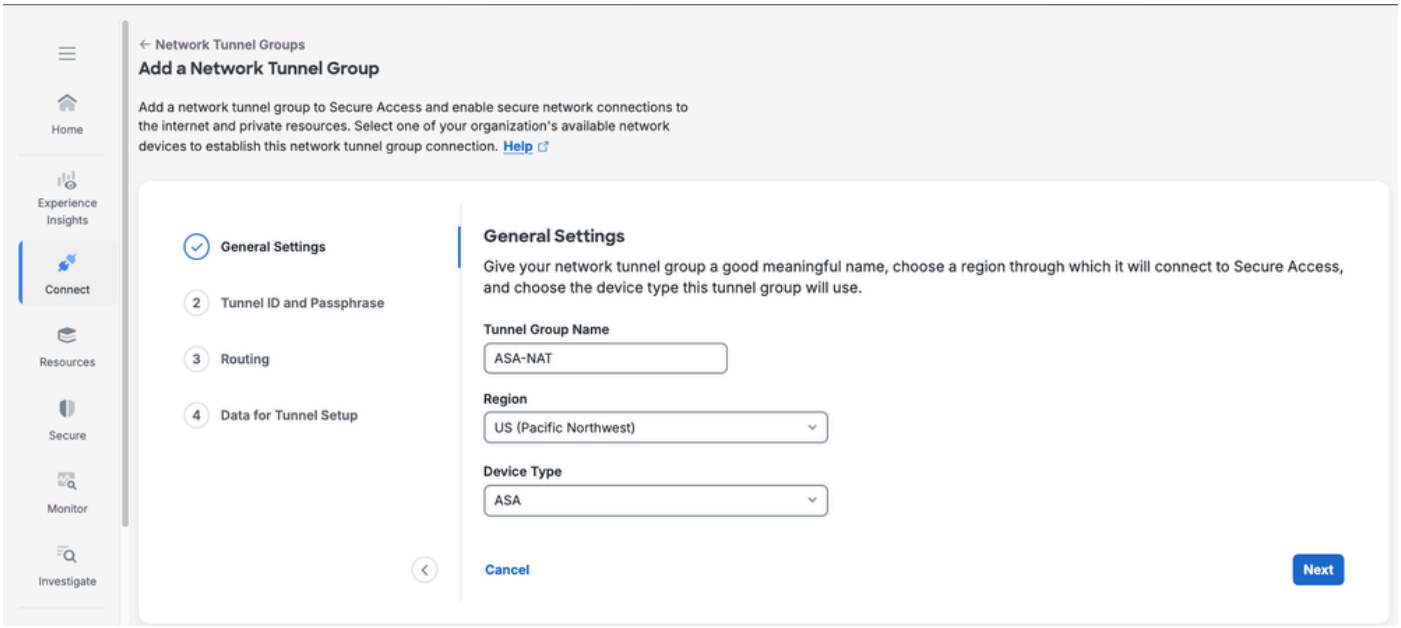
1.登录安全访问控制面板安全访问
2.导航到Connect > Network Connections > Network Tunnel Groups，然后单击+Add配置网络隧道组



安全访问 — 网络隧道组

3.配置网络隧道组 — 常规设置

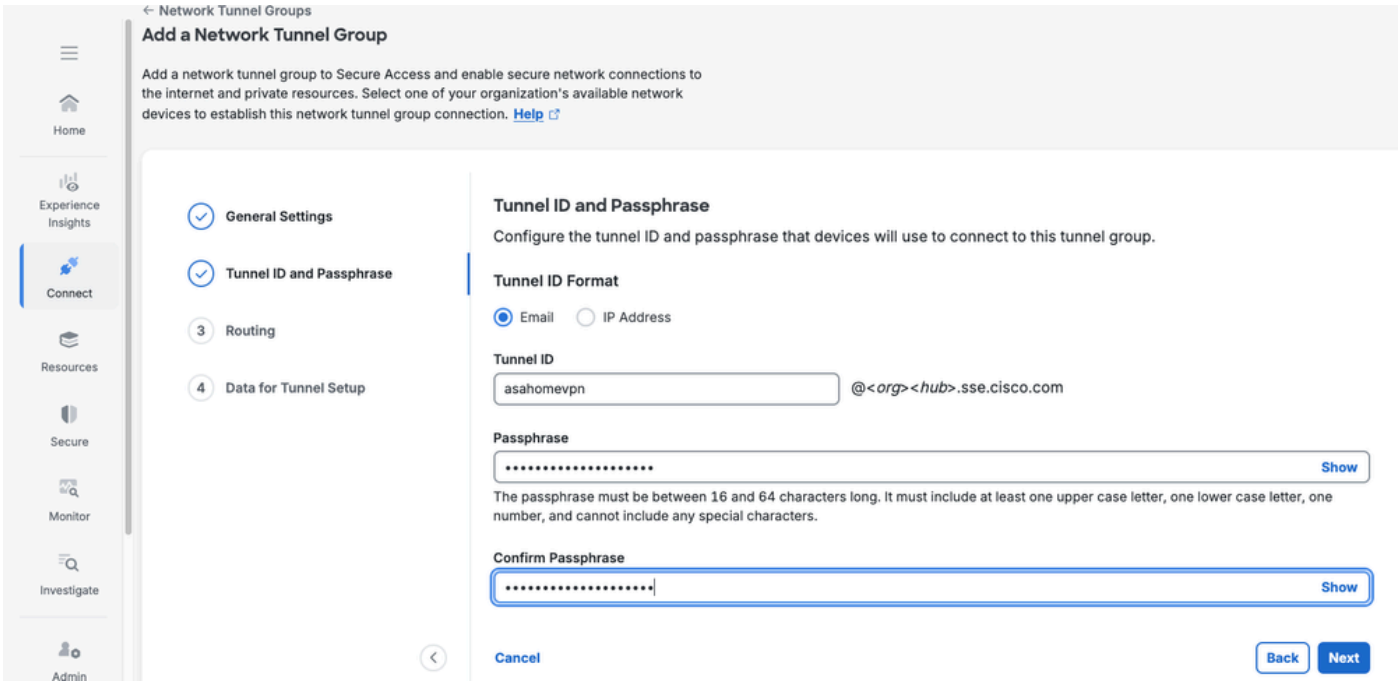
- 配置隧道组名称、区域和设备类型
- 单击“下一步”



安全访问 — 网络隧道组常规设置

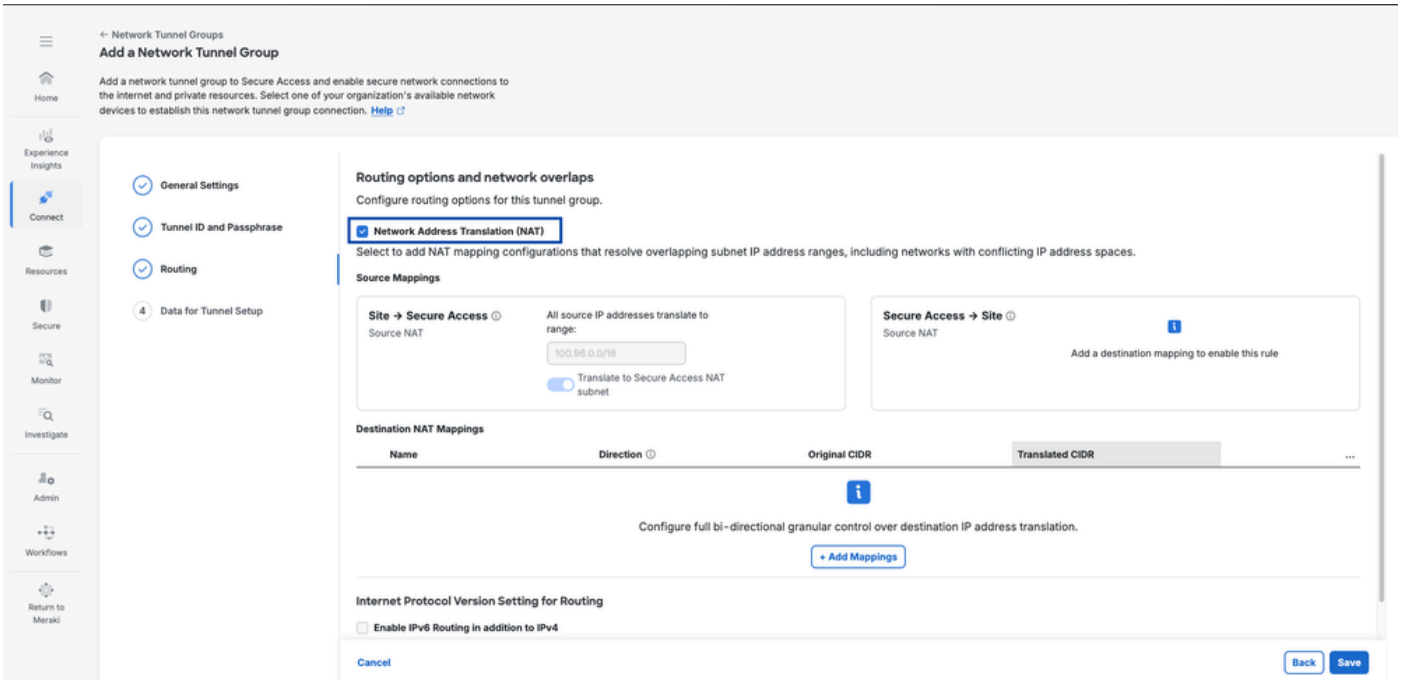
4.配置隧道ID和口令。

- 配置隧道ID和口令。
- 点击“下一步”



安全访问 — 网络隧道组

5.启用网络地址转换(NAT)

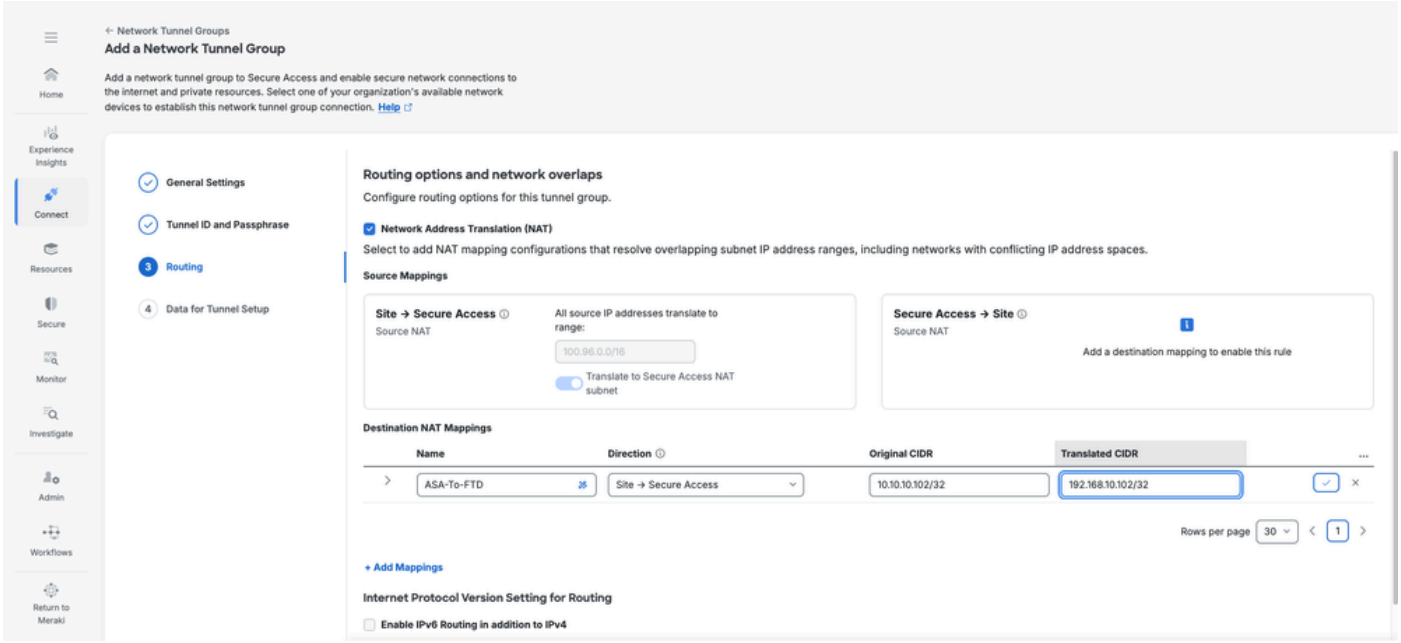


安全访问 — 网络隧道组NAT设置

6.添加目标NAT映射

流程1 — 路由器1到路由器2（站点到安全访问）

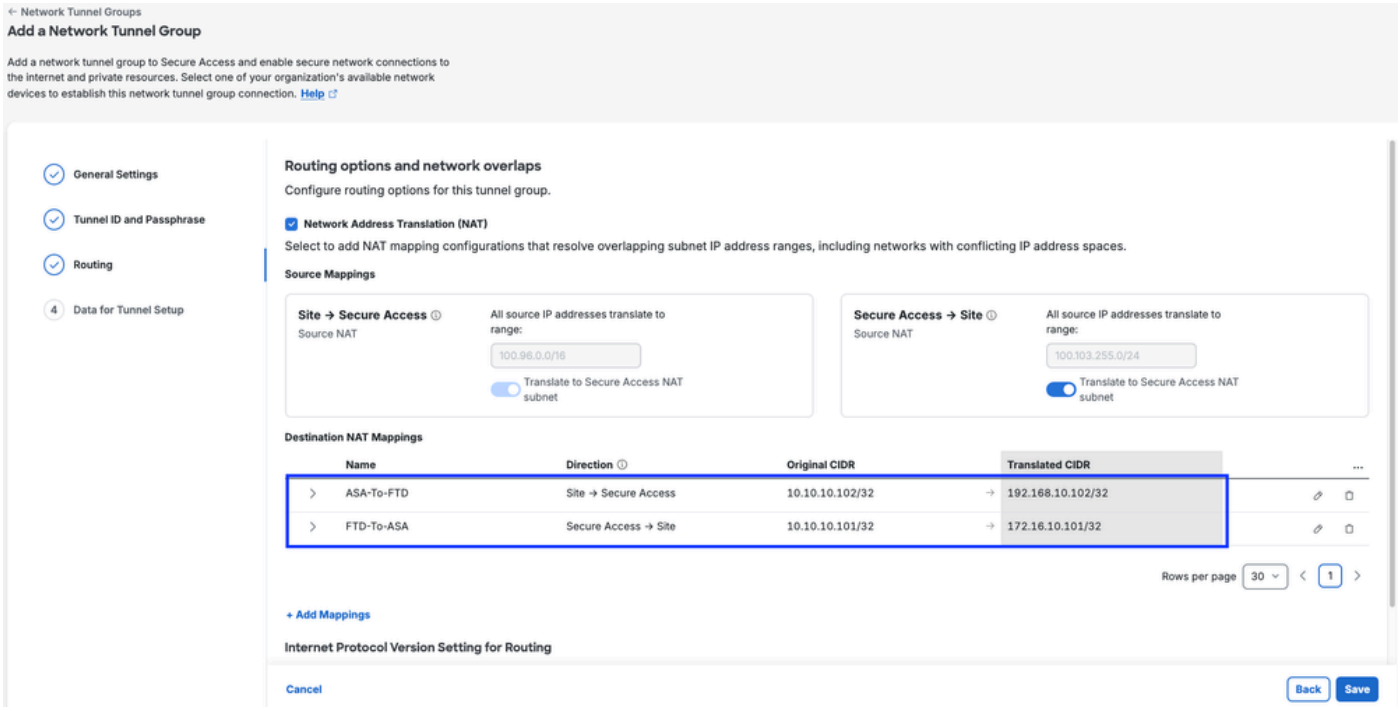
流程2 — 路由器2到路由器1（安全访问站点）



安全访问 — 网络隧道组NAT设置

⚠ 警告：启用NAT时，BGP不可用。此外，在客户边缘设备上配置静态VPN

🔍 提示：请始终记住ping已转换的IP 。
转换后的IP进入转换后的CIDR，而实际IP进入原始CIDR。



安全访问 — 网络隧道组NAT设置

点击 保存

提示：对于从“站点A”到“站点B”的流量，从CNHE-1的角度方向是“站点 — >安全访问”
对于从“站点B”到“站点A”的流量，从CNHE-1的观点方向是“SecureAccess -> Site”

ASA上的VPN配置

1.登录ASA CLI，进入启用模式并验证通向Internet的基本IP连接

ASA# sh ip
系统IP地址：
接口名称IP地址子网掩码方法
GigabitEthernet0/0 outside 192.168.1.40 255.255.255.0 manual
GigabitEthernet0/1 inside 10.10.1 255.255.255.0手册

当前IP地址：
接口名称IP地址子网掩码方法
GigabitEthernet0/0 outside 192.168.1.40 255.255.255.0 manual
GigabitEthernet0/1 inside 10.10.1 255.255.255.0手册

```
ASA# ping 8.8.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/28/30 ms
ASA#
```

2.配置IKEv2策略并在外部接口上启用IKEv2

```
crypto ikev2 policy 10
encryption aes-gcm-256
完整性空
第 19 组
lifetime seconds 86400
crypto ikev2 enable outside
```

3.配置IPSec提议

```
crypto ipsec ikev2 ipsec-proposal Ipsec-Proposal-primary
protocol esp encryption aes-gcm-256
```

4.配置IPsec配置文件

```
crypto ipsec profile csa-profile-primary
set ikev2 ipsec-proposal Ipsec-Proposal-primary
set ikev2 local-identity email-id <primary tunnel-id>
```

5.配置组策略，并在属性下启用IKEv2协议。

```
group-policy csa-policy-primary internal
组策略csa策略主要属性
vpn-tunnel-protocol ikev2
```

6.配置隧道组。

```
tunnel-group 44.228.138.150 type ipsec-l2l
tunnel-group 44.228.138.150 general-attributes
default-group-policy csa-policy-primary
tunnel-group 44.228.138.150 ipsec-attributes
ikev2 remote-authentication pre-shared-key
ikev2 local-authentication pre-shared-key
```

7.配置虚拟隧道接口(VTI)

- Nameif可以是（如果您选择）
- 分配给VTI的IP地址不能与ASA上的任何其他接口重叠
- 隧道sourcemustbe防火墙的外部接口
- 隧道目标必须为数据中心IP地址

interface Tunnel1
nameif VTI-1
ip address 169.254.2.1 255.255.255.252
隧道源接口外部
隧道目标44.228.138.150
隧道模式IPSec IPv4
tunnel protection ipsec profile csa-profile-primary

8.配置路由，使映射IP地址或子网的流量在VTI接口内路由。下一跳需要通过VTI范围内的IP到达。

route VTI-1 0.0.0.0 0.0.0.0 169.254.2.2 5.（适用于基于互联网的RAVPN池或CGNAT流量）

或

route VTI-1 172.16.10.101 255.255.255.255 169.254.2.2 5

基于策略的路由

1.访问列表

access-list PBR extended permit ip 10.10.10.0 255.255.255.0 any

2.路由映射

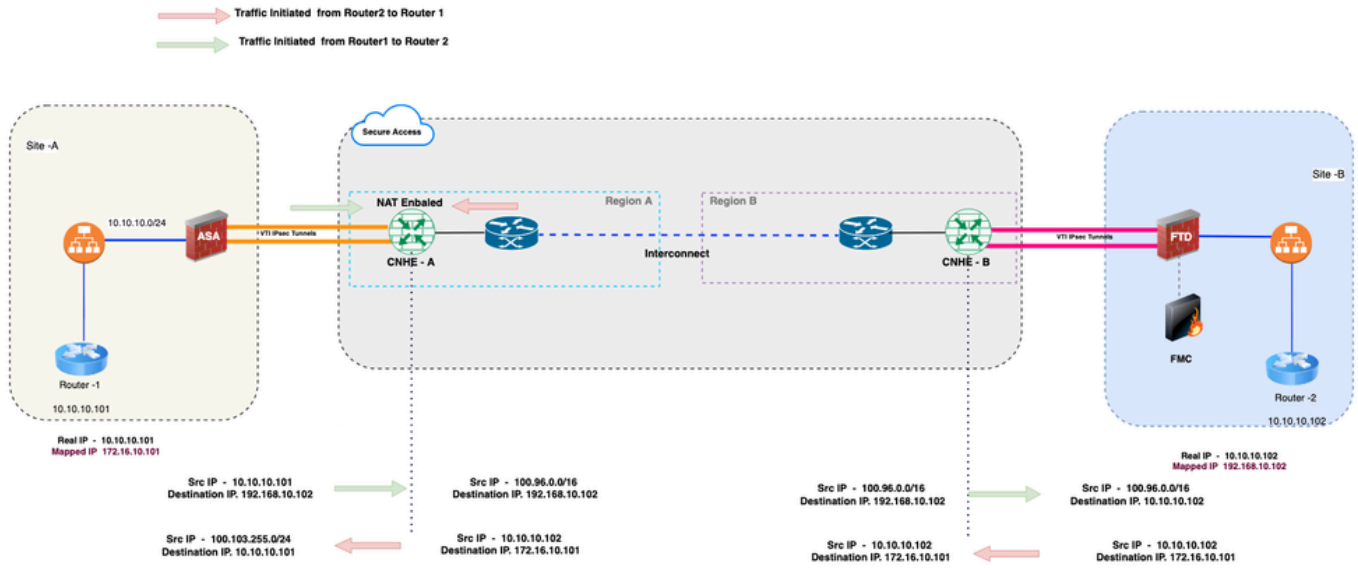
route-map RM-CSA permit 10
match ip address PBR
set ip next-hop 169.254.2.2 169.254.2.6

3.在入口接口上应用路由映射

interface GigabitEthernet0/1
nameif inside

安全级别100
ip address 10.10.10.1 255.255.255.0
policy-route route-map RM-CSA

验证



路由器接口和GW连通性状态

```
Router1#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       192.168.1.101   YES NVRAM  down                  down
GigabitEthernet2       10.10.10.101    YES NVRAM  up                    up
GigabitEthernet3       unassigned      YES NVRAM  administratively down down
GigabitEthernet9       unassigned      YES unset  administratively down down
Router1#ping  10.10.10.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

测试1- Router2 —> Router1。-Ping 测试

```
Router1#ping 192.168.10.102
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.102, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 207/211/223 ms
Router1#
```

ASA上的数据包捕获（本地FW）

```
ASA# sh capture
capture capin type raw-data trace interface inside [Capturing - 1300 bytes]
  match icmp any any
capture tunnel type raw-data trace interface vti-1 [Capturing - 1300 bytes]
  match icmp any any
capture asp type asp-drop all [Capturing - 0 bytes]
  match icmp any any
ASA#
ASA#
ASA# ter pag 20
ASA# sh cap capin

10 packets captured

   1: 10:56:45.024244       10.10.10.101 > 192.168.10.102 icmp: echo request
   2: 10:56:45.231143       192.168.10.102 > 10.10.10.101 icmp: echo reply
   3: 10:56:45.232470       10.10.10.101 > 192.168.10.102 icmp: echo request
   4: 10:56:45.441856       192.168.10.102 > 10.10.10.101 icmp: echo reply
   5: 10:56:45.443122       10.10.10.101 > 192.168.10.102 icmp: echo request
   6: 10:56:45.652477       192.168.10.102 > 10.10.10.101 icmp: echo reply
   7: 10:56:45.653423       10.10.10.101 > 192.168.10.102 icmp: echo request
   8: 10:56:45.875397       192.168.10.102 > 10.10.10.101 icmp: echo reply
   9: 10:56:45.876450       10.10.10.101 > 192.168.10.102 icmp: echo request
  10: 10:56:46.082301       192.168.10.102 > 10.10.10.101 icmp: echo reply
10 packets shown
ASA# sh cap tunnel

10 packets captured

   1: 10:56:45.024458       10.10.10.101 > 192.168.10.102 icmp: echo request
   2: 10:56:45.230914       192.168.10.102 > 10.10.10.101 icmp: echo reply
   3: 10:56:45.232516       10.10.10.101 > 192.168.10.102 icmp: echo request
   4: 10:56:45.441795       192.168.10.102 > 10.10.10.101 icmp: echo reply
   5: 10:56:45.443153       10.10.10.101 > 192.168.10.102 icmp: echo request
   6: 10:56:45.652432       192.168.10.102 > 10.10.10.101 icmp: echo reply
   7: 10:56:45.653454       10.10.10.101 > 192.168.10.102 icmp: echo request
   8: 10:56:45.875351       192.168.10.102 > 10.10.10.101 icmp: echo reply
   9: 10:56:45.876480       10.10.10.101 > 192.168.10.102 icmp: echo request
  10: 10:56:46.082240       192.168.10.102 > 10.10.10.101 icmp: echo reply
10 packets shown
```

FTD（远程防火墙）上的数据包捕获

```
ftd# sh capture
capture vti type raw-data trace interface VTI-1 include-decrypted [Capturing - 1300 bytes]
  match icmp any any
capture capin type raw-data interface inside [Capturing - 1300 bytes]
  match icmp any any
capture asp type asp-drop all [Capturing - 0 bytes]
  match icmp any any
ftd# sh cap vti

10 packets captured

   1: 16:06:41.122292       100.96.2.0 > 10.10.10.102 icmp: echo request
   2: 16:06:41.124856       10.10.10.102 > 100.96.2.0 icmp: echo reply
   3: 16:06:41.329557       100.96.2.0 > 10.10.10.102 icmp: echo request
   4: 16:06:41.330442       10.10.10.102 > 100.96.2.0 icmp: echo reply
   5: 16:06:41.546327       100.96.2.0 > 10.10.10.102 icmp: echo request
   6: 16:06:41.547106       10.10.10.102 > 100.96.2.0 icmp: echo reply
   7: 16:06:41.752036       100.96.2.0 > 10.10.10.102 icmp: echo request
   8: 16:06:41.752814       10.10.10.102 > 100.96.2.0 icmp: echo reply
   9: 16:06:41.967891       100.96.2.0 > 10.10.10.102 icmp: echo request
  10: 16:06:41.968501       10.10.10.102 > 100.96.2.0 icmp: echo reply
10 packets shown
ftd# sh cap capin

10 packets captured

   1: 16:06:41.123299       100.96.2.0 > 10.10.10.102 icmp: echo request
   2: 16:06:41.124825       10.10.10.102 > 100.96.2.0 icmp: echo reply
   3: 16:06:41.330000       100.96.2.0 > 10.10.10.102 icmp: echo request
   4: 16:06:41.330396       10.10.10.102 > 100.96.2.0 icmp: echo reply
   5: 16:06:41.546800       100.96.2.0 > 10.10.10.102 icmp: echo request
   6: 16:06:41.547090       10.10.10.102 > 100.96.2.0 icmp: echo reply
   7: 16:06:41.752448       100.96.2.0 > 10.10.10.102 icmp: echo request
   8: 16:06:41.752783       10.10.10.102 > 100.96.2.0 icmp: echo reply
   9: 16:06:41.968242       100.96.2.0 > 10.10.10.102 icmp: echo request
  10: 16:06:41.968486       10.10.10.102 > 100.96.2.0 icmp: echo reply
10 packets shown
```

测试2. - Router 2 —> Router 1 ping测试

```
Router2#sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       unassigned      YES NVRAM  administratively down down
GigabitEthernet2       10.10.10.102    YES NVRAM  up                    up
GigabitEthernet3       unassigned      YES NVRAM  administratively down down
Router2#ping 10.10.10.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

FTD数据包捕获（本地防火墙）

```
ftd# sh capture
capture vti type raw-data trace interface VTI-1 include-decrypted [Capturing - 1300 bytes]
  match icmp any any
capture capin type raw-data interface inside [Capturing - 1300 bytes]
  match icmp any any
capture asp type asp-drop all [Capturing - 0 bytes]
  match icmp any any
ftd#
ftd#
ftd#
ftd# sh cap vti

10 packets captured

   1: 16:11:45.622450       10.10.10.102 > 172.16.10.101 icmp: echo request
   2: 16:11:45.823825       172.16.10.101 > 10.10.10.102 icmp: echo reply
   3: 16:11:45.825762       10.10.10.102 > 172.16.10.101 icmp: echo request
   4: 16:11:46.045667       172.16.10.101 > 10.10.10.102 icmp: echo reply
   5: 16:11:46.046857       10.10.10.102 > 172.16.10.101 icmp: echo request
   6: 16:11:46.252321       172.16.10.101 > 10.10.10.102 icmp: echo reply
   7: 16:11:46.253572       10.10.10.102 > 172.16.10.101 icmp: echo request
   8: 16:11:46.453803       172.16.10.101 > 10.10.10.102 icmp: echo reply
   9: 16:11:46.454764       10.10.10.102 > 172.16.10.101 icmp: echo request
  10: 16:11:46.664119       172.16.10.101 > 10.10.10.102 icmp: echo reply
10 packets shown
ftd# sh cap capin

10 packets captured

   1: 16:11:45.622083       10.10.10.102 > 172.16.10.101 icmp: echo request
   2: 16:11:45.823886       172.16.10.101 > 10.10.10.102 icmp: echo reply
   3: 16:11:45.825442       10.10.10.102 > 172.16.10.101 icmp: echo request
   4: 16:11:46.045697       172.16.10.101 > 10.10.10.102 icmp: echo reply
   5: 16:11:46.046582       10.10.10.102 > 172.16.10.101 icmp: echo request
   6: 16:11:46.252352       172.16.10.101 > 10.10.10.102 icmp: echo reply
   7: 16:11:46.253313       10.10.10.102 > 172.16.10.101 icmp: echo request
   8: 16:11:46.453849       172.16.10.101 > 10.10.10.102 icmp: echo reply
   9: 16:11:46.454596       10.10.10.102 > 172.16.10.101 icmp: echo request
  10: 16:11:46.664150       172.16.10.101 > 10.10.10.102 icmp: echo reply
10 packets shown
```

ASA数据包捕获（远程FW）

```
ASA# sh capture
capture capin type raw-data trace interface inside [Capturing - 1300 bytes]
  match icmp any any
capture tunnel type raw-data trace interface vti-1 [Capturing - 1300 bytes]
  match icmp any any
capture asp type asp-drop all [Capturing - 0 bytes]
  match icmp any any
ASA# sh cap capin

10 packets captured

   1: 11:08:30.288391       100.103.255.195 > 10.10.10.101 icmp: echo request
   2: 11:08:30.289123       10.10.10.101 > 100.103.255.195 icmp: echo reply
   3: 11:08:30.504566       100.103.255.195 > 10.10.10.101 icmp: echo request
   4: 11:08:30.504963       10.10.10.101 > 100.103.255.195 icmp: echo reply
   5: 11:08:30.710992       100.103.255.195 > 10.10.10.101 icmp: echo request
   6: 11:08:30.711404       10.10.10.101 > 100.103.255.195 icmp: echo reply
   7: 11:08:30.917112       100.103.255.195 > 10.10.10.101 icmp: echo request
   8: 11:08:30.917402       10.10.10.101 > 100.103.255.195 icmp: echo reply
   9: 11:08:31.128243       100.103.255.195 > 10.10.10.101 icmp: echo request
  10: 11:08:31.128640       10.10.10.101 > 100.103.255.195 icmp: echo reply
10 packets shown
ASA# sh cap tunnel

10 packets captured

   1: 11:08:30.287964       100.103.255.195 > 10.10.10.101 icmp: echo request
   2: 11:08:30.289322       10.10.10.101 > 100.103.255.195 icmp: echo reply
   3: 11:08:30.504505       100.103.255.195 > 10.10.10.101 icmp: echo request
   4: 11:08:30.505009       10.10.10.101 > 100.103.255.195 icmp: echo reply
   5: 11:08:30.710961       100.103.255.195 > 10.10.10.101 icmp: echo request
   6: 11:08:30.711434       10.10.10.101 > 100.103.255.195 icmp: echo reply
   7: 11:08:30.917066       100.103.255.195 > 10.10.10.101 icmp: echo request
   8: 11:08:30.917417       10.10.10.101 > 100.103.255.195 icmp: echo reply
   9: 11:08:31.128197       100.103.255.195 > 10.10.10.101 icmp: echo request
  10: 11:08:31.128685       10.10.10.101 > 100.103.255.195 icmp: echo reply
10 packets shown
```

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。