

使用ACI多站点交换矩阵配置站点间L3out

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[站点间L3out配置支持的方案](#)

[配置](#)

[网络图](#)

[物理拓扑](#)

[逻辑拓扑](#)

[配置](#)

[配置架构配置1](#)

[配置交换矩阵策略](#)

[配置RTEP/ETEP](#)

[配置Stretch租户](#)

[配置方案](#)

[创建架构](#)

[创建站点A模板](#)

[配置模板](#)

[创建拉伸模板](#)

[附加模板](#)

[配置静态端口绑定](#)

[配置BD](#)

[配置主机A\(N9K\)](#)

[创建站点B模板](#)

[配置站点B L3out](#)

[创建外部EPG](#)

[配置外部N9K \(站点B \)](#)

[将站点B L3out连接到站点A EPG\(BD\)](#)

[配置合同](#)

[创建合同](#)

[验证](#)

[终端学习](#)

[ETEP/RTEP验证](#)

[ICMP可达性](#)

[路由验证](#)

[故障排除](#)

[站点2 枝叶1](#)

[站点2 主干](#)

[站点1 主干](#)

[了解路由区分器条目](#)

[站点1 枝叶1](#)

[检验ELAM\(Site1 Spine\)](#)

[站点1 主干检验路由映射](#)

简介

本文档介绍使用思科以应用为中心的基础设施(ACI)多站点交换矩阵进行站点间L3out配置的步骤。

先决条件

要求

Cisco 建议您了解以下主题：

- 功能性ACI多站点交换矩阵设置
- 外部路由器/连接

使用的组件

本文档中的信息基于：

- 多站点协调器(MSO)2.2(1)版或更高版本
- ACI版本4.2(1)或更高版本
- MSO节点
- ACI交换矩阵
- Nexus 9000系列交换机(N9K) (终端主机和L3out外部设备模拟)
- Nexus 9000系列交换机(N9K)(站点间网络(ISN))

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

站点间L3out配置支持的方案

架构配置1

- 租户在站点 (A和B) 之间延伸。
- 虚拟路由和转发(VRF)在站点 (A和B) 之间延伸。
- 一个站点(A)的本地终端组(EPG)/网桥域(BD)。
- L3out本地到另一个站点(B)。
- L3out本地到站点(B)的外部EPG。
- 从MSO创建和配置合同。

架构配置2

- 租户在站点 (A和B) 之间延伸。

- VRF在站点 (A和B) 之间延伸。
- EPG/BD在站点 (A和B) 之间延伸。
- L3out本地到一个站点(B)。
- L3out本地到站点(B)的外部EPG。
- 合同配置可以从MSO完成，或者每个站点都通过应用策略基础设施控制器(APIC)创建本地合同，并在延伸EPG和L3out外部EPG之间本地连接。在这种情况下，由于本地合同关系和策略实施需要影子External_EPG，因此它会出现在站点A。

方案配置3

- 租户在站点 (A和B) 之间延伸。
- VRF在站点 (A和B) 之间延伸。
- EPG/BD在站点 (A和B) 之间延伸。
- L3out本地到一个站点(B)。
- L3out的外部EPG在站点 (A和B) 之间延伸。
- 合同配置可以从MSO完成，或每个站点都从APIC创建本地合同，并在延伸EPG和延伸外部EPG之间本地连接。

方案配置4

- 租户在站点 (A和B) 之间延伸。
- VRF在站点 (A和B) 之间延伸。
- EPG/BD本地到一个站点(A)或EPG/BD本地到每个站点 (站点A中的EPG-A和站点B中的EPG-B)。
- L3out本地到一个站点(B)，或为了实现外部连接的冗余，您可以将L3out本地到每个站点 (本地到站点A，本地到站点B)。
- L3out的外部EPG在站点 (A和B) 之间延伸。
- 合同配置可以从MSO完成，或每个站点具有从APIC创建的本地合同，并在延伸EPG和延伸外部EPG之间本地连接。

方案配置5 (传输路由)

- 租户在站点 (A和B) 之间延伸。
- VRF在站点 (A和B) 之间延伸。
- L3out本地到每个站点 (本地到站点A和本地到站点B)。
- 每个站点 (A和B) 的本地外部EPG。
- 合同配置可以从MSO完成，或每个站点具有从APIC创建的本地合同，并在外部EPG本地和影子外部EPG本地之间本地附加。

方案配置5 (VRF间传输路由)

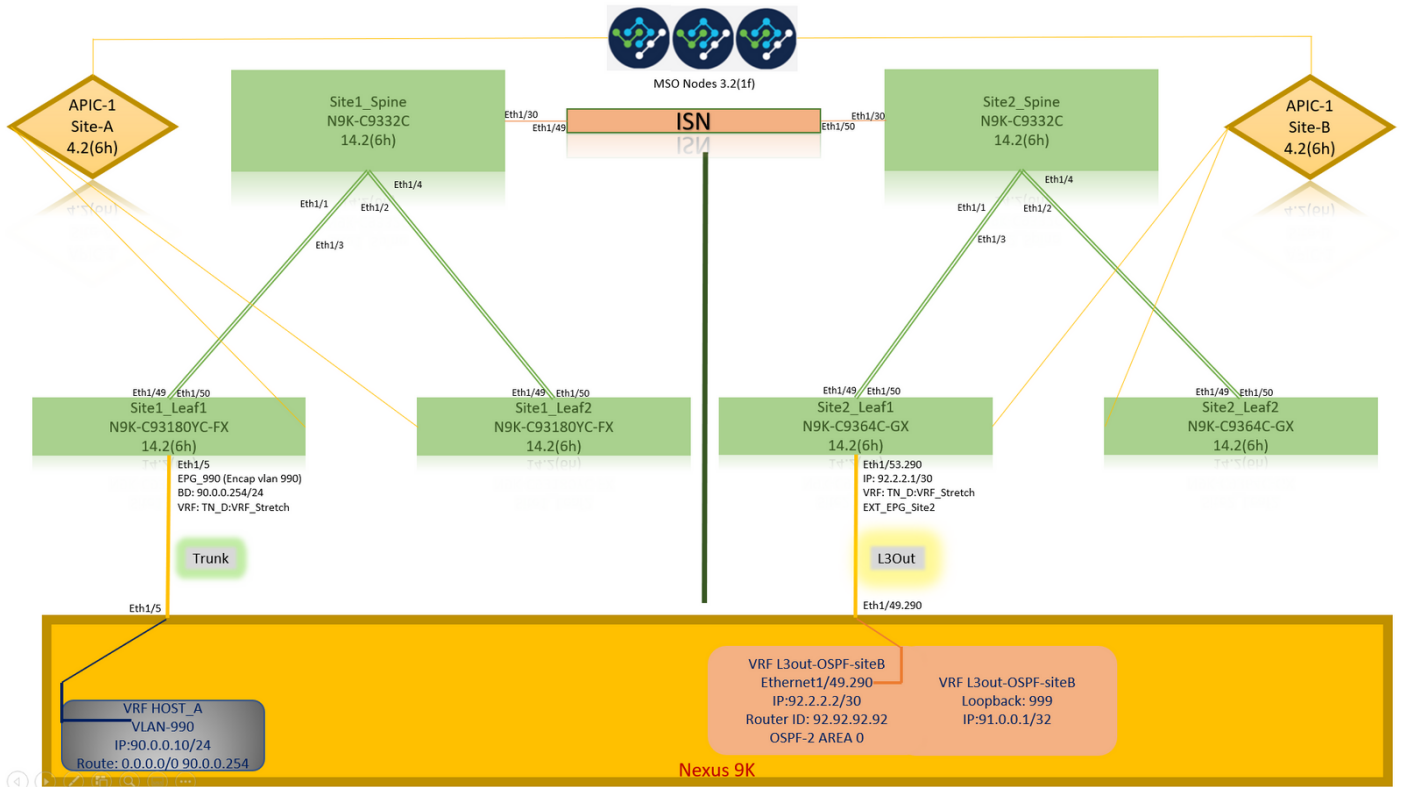
- 租户在站点 (A和B) 之间延伸。
- 本地到每个站点 (A和B) 的VRF。
- L3out本地到每个站点 (本地到站点A和本地到站点B)。
- 每个站点 (A和B) 的本地外部EPG。
- 合同配置可以从MSO完成，或每个站点具有从APIC创建的本地合同，并在外部EPG本地和影子外部EPG本地之间本地附加。

注意：本文档提供基本站点间L3out配置步骤和验证。在本例中，使用Schema-config1。

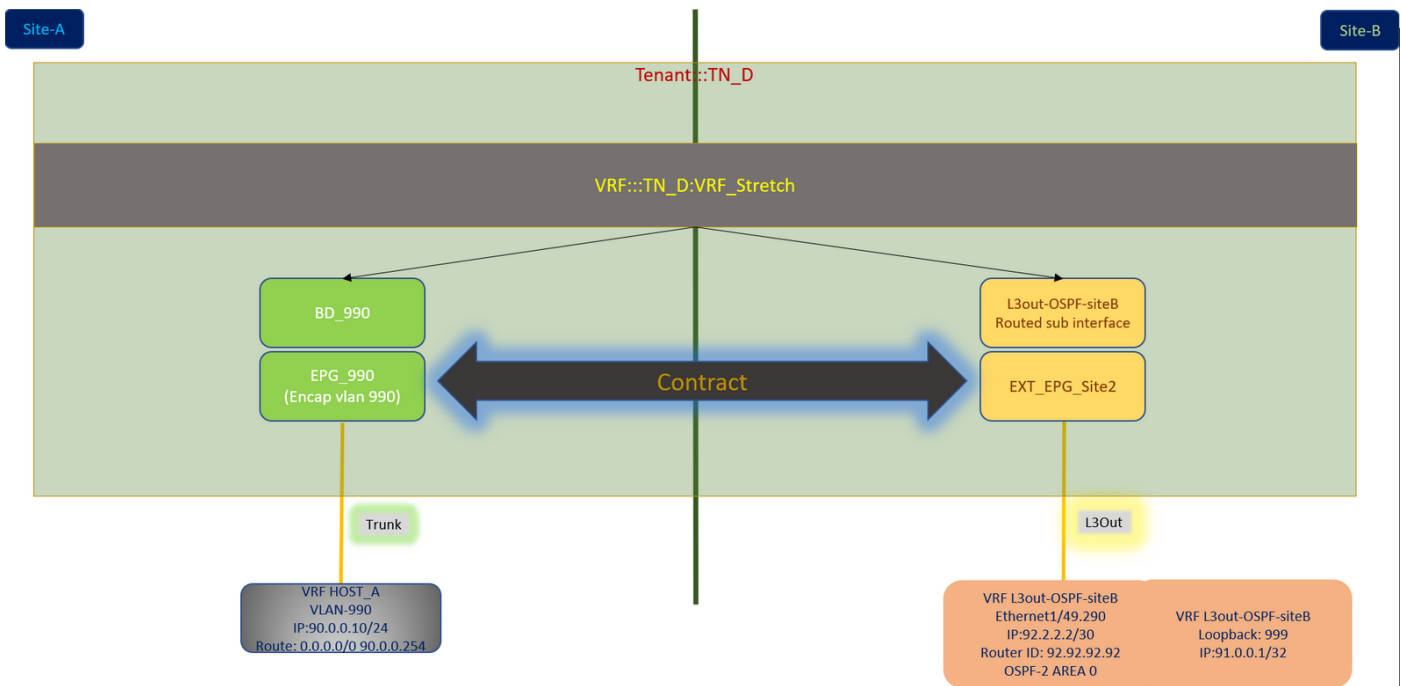
配置

网络图

物理拓扑



逻辑拓扑



配置

在本示例中，我们使用Schema-config1。但是，此配置可以以类似方式完成（根据合同关系进行微小更改）其他受支持的架构配置，但拉伸对象需要位于拉伸模板而不是特定站点模板中。

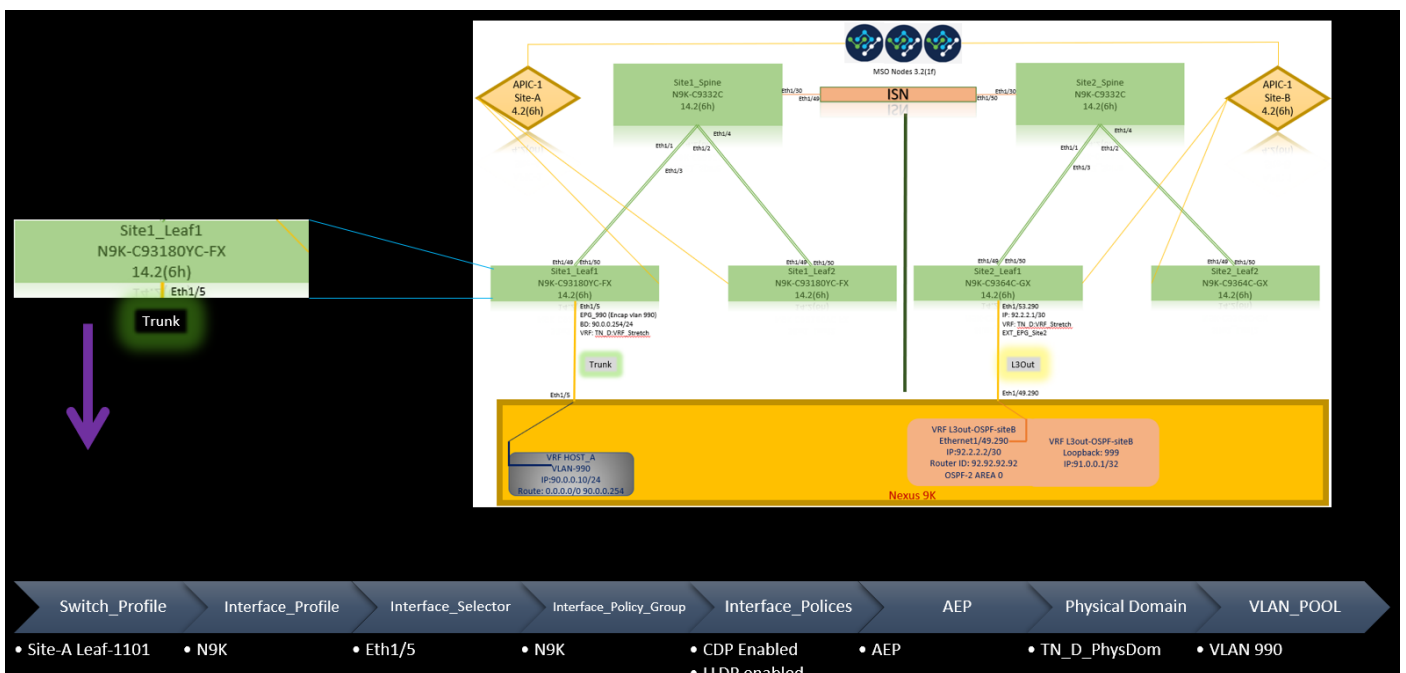
配置架构配置1

- 租户在站点（A和B）之间延伸。
- VRF在站点（A和B）之间延伸。
- EPG/BD本地到一个站点(A)。
- L3out本地到另一个站点(B)。
- L3out本地到站点(B)的外部EPG。
- 从MSO创建合同和配置。
查看[站点间L3Out准则和限制](#)。
- 站点间L3out的配置不受支持：站点中的组播接收器，通过另一站点L3out从外部源接收组播。站点中从外部源接收的组播从不发送到其他站点。当站点中的接收方从外部源接收组播时，必须在本地L3out上接收组播。内部组播源使用PIM-SM任意源组播(ASM)将组播发送到外部接收器。内部组播源必须能够从本地L3out到达外部交汇点(RP)。巨型OverLay交换矩阵(GOLF)。外部EPG的首选组。

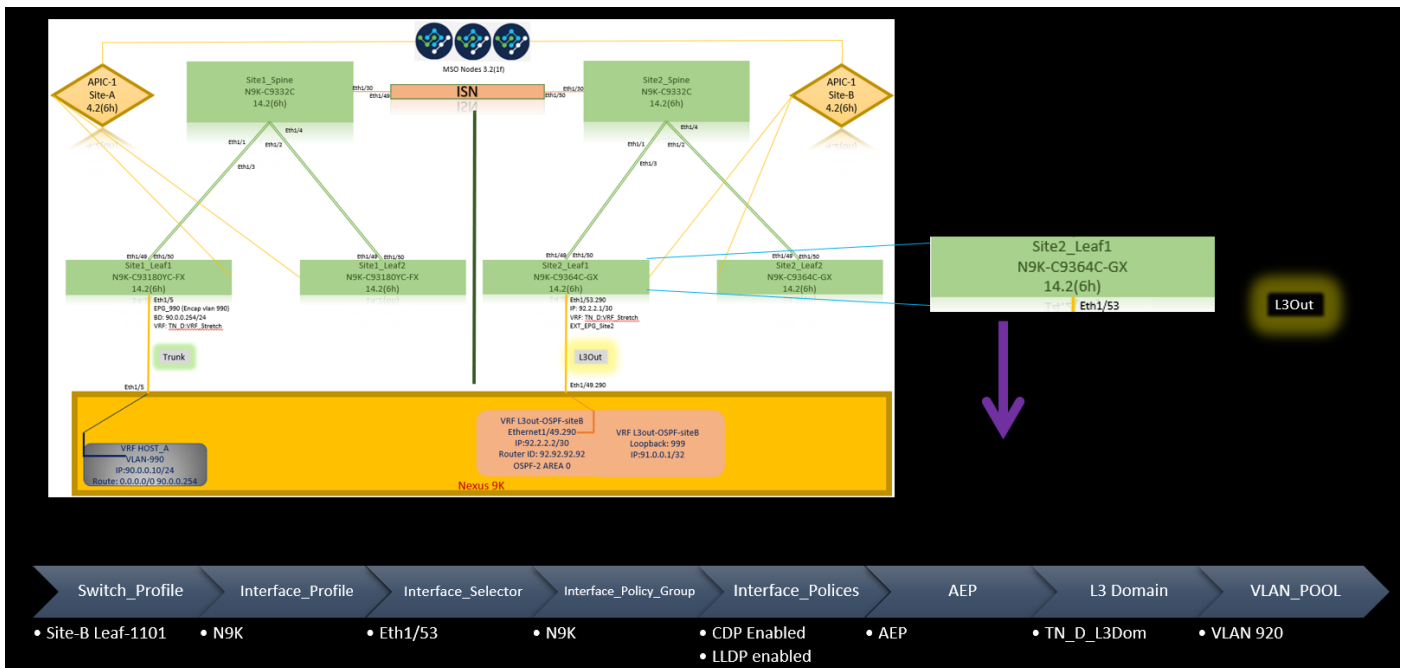
配置交换矩阵策略

每个站点的交换矩阵策略是基本配置，因为这些策略配置链接到特定租户/EPG/静态端口绑定或L3out物理连接。使用交换矩阵策略的任何错误配置都可能导致来自APIC或MSO的逻辑配置失败，从而导致实验室设置中使用的所提供的交换矩阵策略配置失败。它有助于了解在MSO或APIC中链接到哪个对象的对象。

Host_A站点A的连接交换矩阵策略



站点B的L3out连接交换矩阵策略



可选步骤

一旦为各个连接设置了交换矩阵策略，您就可以确保从各个APIC集群发现并可访问所有枝叶/主干。接下来，您可以验证从MSO可到达的两个站点（APIC集群），并且多站点设置可操作（和IPN连接）。

配置RTEP/ETEP

可路由隧道终端池(RTEP)或外部隧道终端池(ETEP)是站点间L3out的必需配置。旧版本的MSO显示“可路由TEP池”，而较新版本的MSO显示“外部TEP池”，但两者同义。这些TEP池通过VRF“Overlay-1”用于边界网关协议(BGP)以太网VPN(EVPN)。

来自L3out的外部路由通过BGP EVPN通告到另一个站点。此RTEP/ETEP也用于远程枝叶配置，因此，如果APIC中已存在ETEP/RTEP配置，则必须将其导入MSO。

以下是从MSO GUI配置ETEP的步骤。由于版本为3.X MSO，因此它显示ETEP。ETEP池在每个站点必须唯一，并且不得与每个站点的任何内部EPG/BD子网重叠。

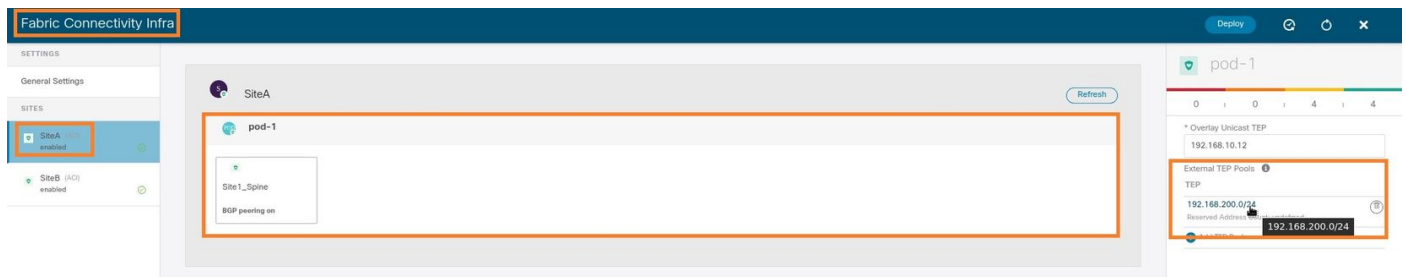
站点A

步骤1.在MSO GUI页面（在网页中打开多站点控制器）中，选择“基础设施”>“基础设施配置”。单击Configure Infra。

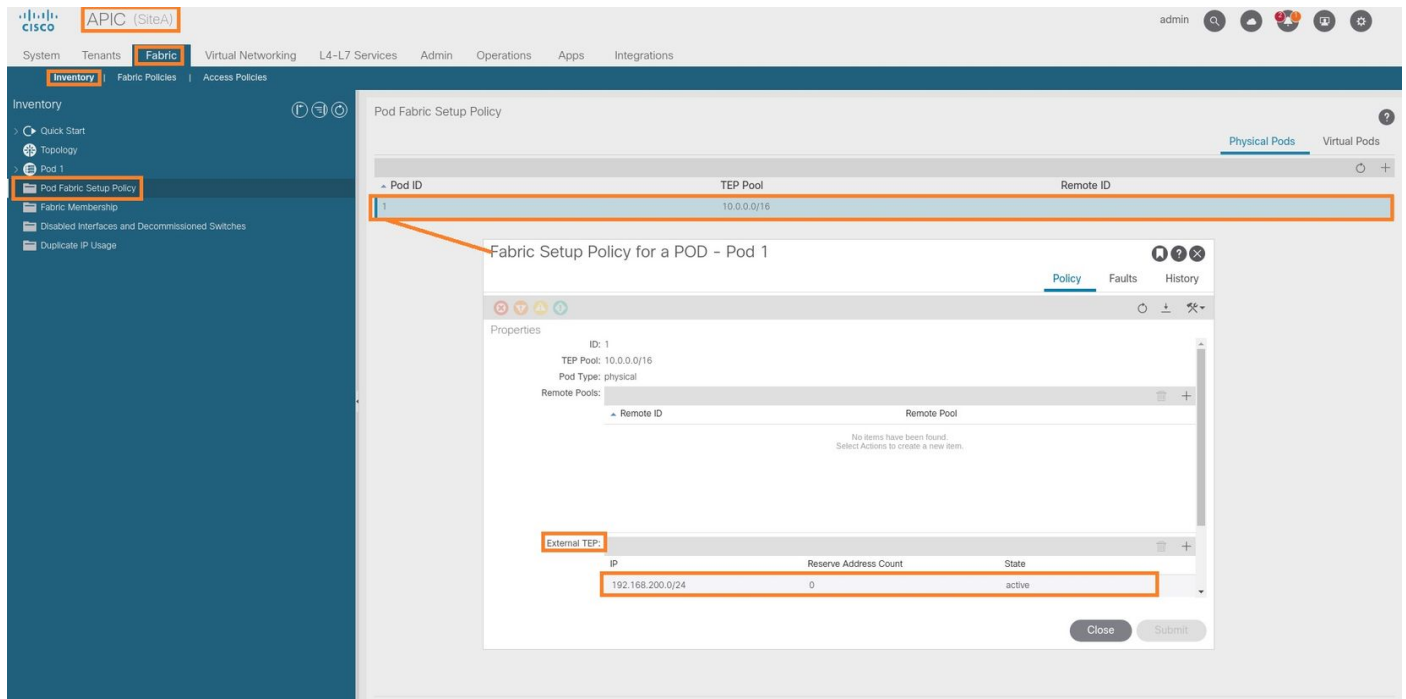


步骤2.在Configure Infra中，选择Site-A, Inside Site-A，选择pod-1。然后，在Pod-1内部，使用Site-A的外部TEP IP地址配置External TEP Pools。（在本例中为192.168.200.0/24）。如果站点A中有多

POD，请对其他Pod重复此步骤。



步骤3.要验证APIC GUI中ETEP池的配置，请选择Fabric > Inventory > Pod Fabric Setup Policy > Pod-ID(双击以打开[Fabric Setup Policy a POD-Pod-x])> External TEP。



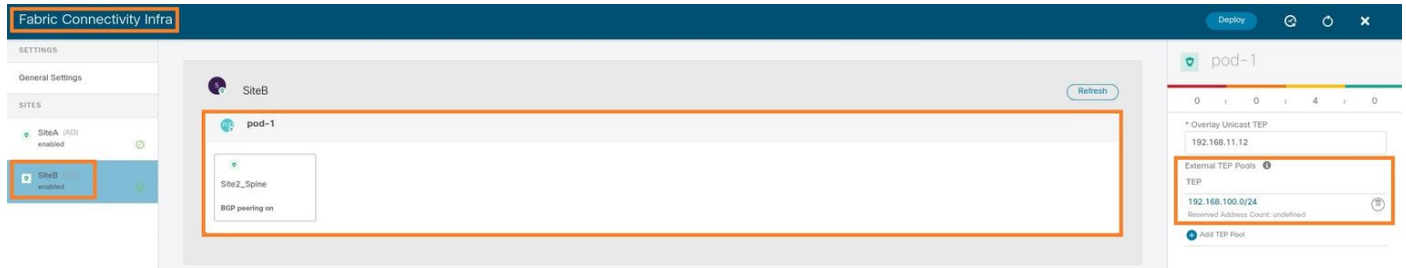
您还可以使用以下命令验证配置：

```
moquery -c fabricExtRoutablePodSubnet
moquery -c fabricExtRoutablePodSubnet -f 'fabric.ExtRoutablePodSubnet.pool=="192.168.200.0/24"'
```

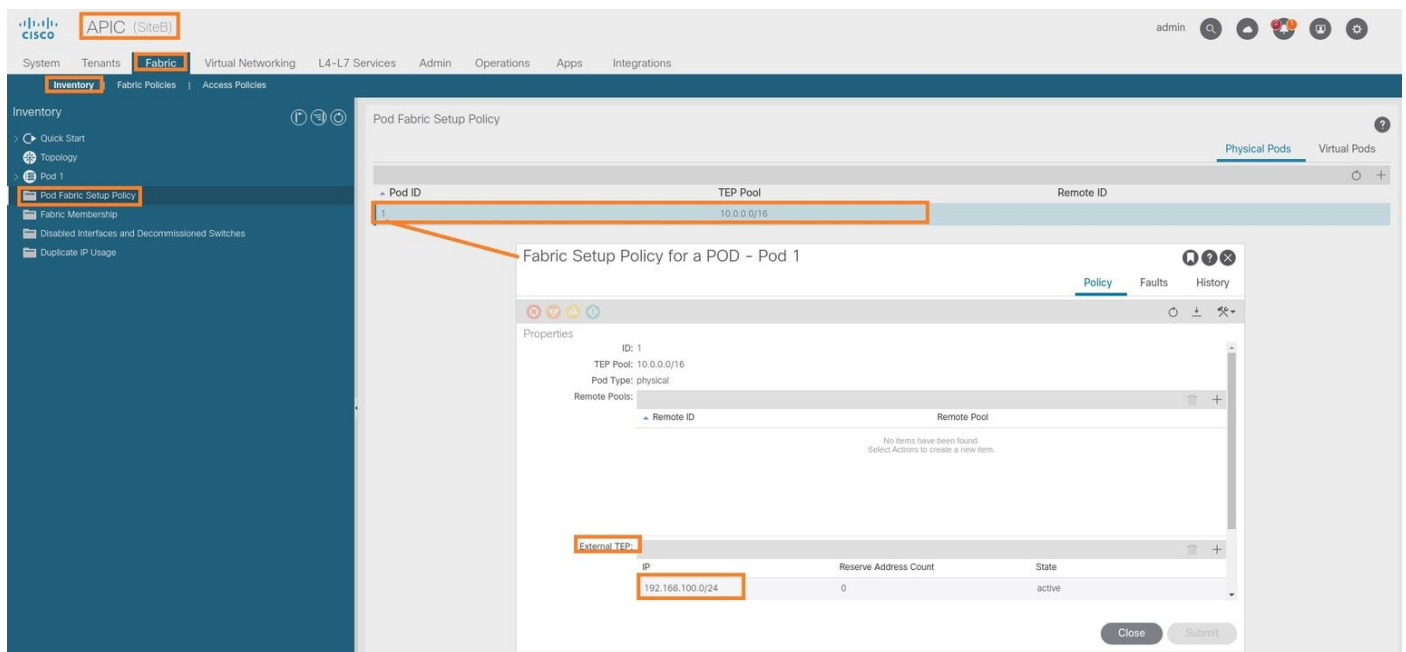
```
APIC1# moquery -c fabricExtRoutablePodSubnet
Total Objects shown: 1
# fabric.ExtRoutablePodSubnet
pool                : 192.168.200.0/24
annotation          : orchestrator:msc
childAction         :
descr               :
dn                  : uni/controller/setuppod1/setupp-1/extrtpodsubnet-[192.168.200.0/24]
extMngdBy           :
lcOwn                : local
modTs               : 2021-07-19T14:45:22.387+00:00
name                 :
nameAlias           :
reserveAddressCount : 0
rn                  : extrtpodsubnet-[192.168.200.0/24]
state                : active
status              :
uid                  : 0
```

站点B

步骤1.为站点B配置外部TEP池（与站点A的步骤相同）在MSO GUI页面（在网页中打开多站点控制器）中，选择**Infrastructure > Infra Configuration**。单击**配置基础设施**。在配置基础设施内，选择**站点B**。在Site-B内，选择**pod-1**。然后，在Pod-1内，使用站点B的外部TEP IP地址配置**External TEP Pools**。（在本例中为192.168.100.0/24）。如果您在站点B中有多POD，请对其他Pod重复此步骤。



步骤2.要验证APIC GUI中ETEP池的配置，请选择**Fabric > Inventory > Pod Fabric Setup Policy > Pod-ID**(双击以打开[Fabric Setup Policy a POD-Pod-x])> **External TEP**。



对于Site-B APIC，输入此命令以验证ETEP地址池。

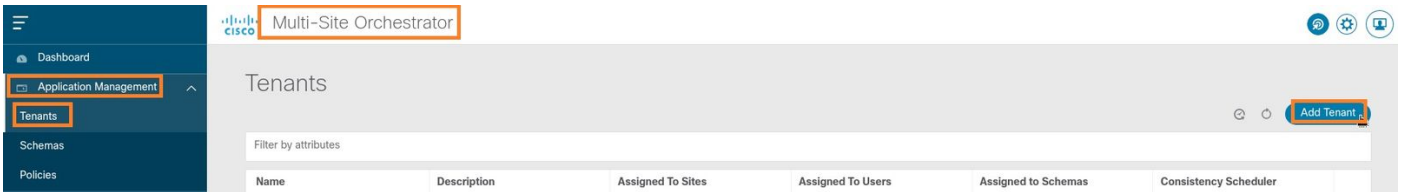
```
apic1# moquery -c fabricExtRoutablePodSubnet -f
'fabric.ExtRoutablePodSubnet.pool=="192.168.100.0/24"'
Total Objects shown: 1
# fabric.ExtRoutablePodSubnet
pool                : 192.168.100.0/24
annotation          : orchestrator:mso <<< This means, configuration pushed from MSO.
childAction         :
descr               :
dn                  : uni/controller/setuppod/setupp-1/extrtpodsubnet-[192.168.100.0/24]
extMngdBy           :
lcOwn               : local
modTs               : 2021-07-19T14:34:18.838+00:00
name                :
nameAlias           :
reserveAddressCount : 0
rn                  : extrtpodsubnet-[192.168.100.0/24]
```



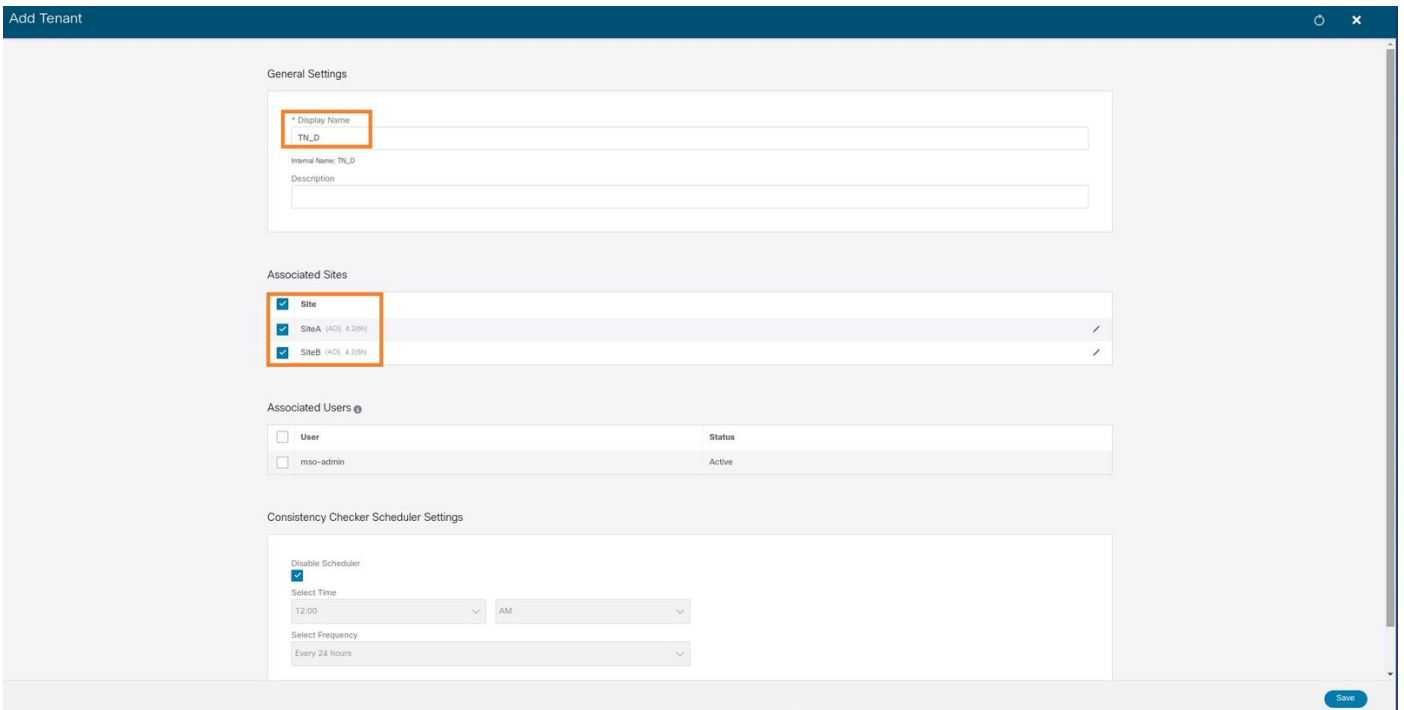
```
state           : active
status          :
uid             : 0
```

配置Stretch租户

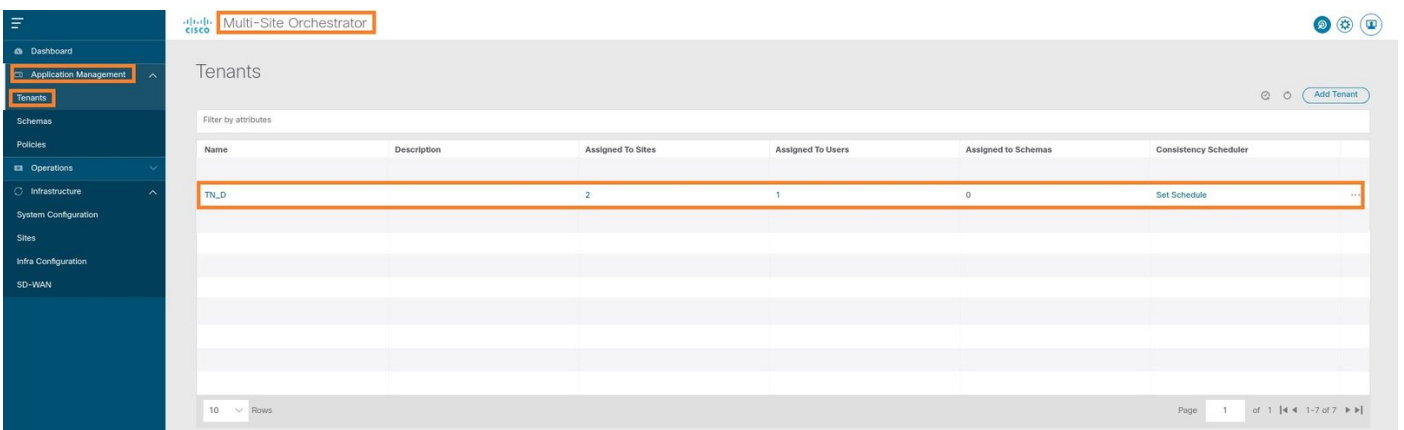
步骤1.在MSO GUI中，选择Application Management > Tenants。单击“添加租户”。在本例中，租户名称为“TN_D”。



步骤2.在“显示名称”字段中，输入租户的名称。在“关联的站点”部分，选中“站点A”和“站点B”复选框。



步骤3.检验新租户“Tn_D”是否已创建。



逻辑视图

从MSO创建租户时，它基本上会在站点A和站点B创建租户。它是一个延伸租户。此租户的逻辑视图如本例所示。此逻辑视图有助于了解租户TN_D是站点A和站点B之间的扩展租户。

The screenshot displays the Multi-Site Orchestrator interface. On the left is a navigation menu with options: Dashboard, Application Management, Tenants, Schemas, Policies, Operations, Infrastructure, and System Configuration. The main area is titled 'Tenants' and includes a search filter and an 'Add Tenant' button. Below this is a table with the following data:

Name	Description	Assigned To Sites	Assigned To Users	Assigned to Schemas	Consistency Scheduler
TN_D		2	1	0	Set Schedule

Below the screenshot, a large light green rectangular area is labeled 'Tenant:::TN_D' in red text at the top left corner.

您可以验证每个站点的APIC中的逻辑视图。您可以看到站点A和站点B都显示已创建“TN_D”租户。



This has been created from

TN_D

> Quick Start

∨ TN_D

> Application Profiles

> Networking

> Contracts

> Policies

> Services

站点B中也创建了相同的拉伸租户“TN_D”。



System

Tenants

Fabric

ALL TENANTS

| Add Tenant

| T



This has been created from

TN_D

> Quick Start

▼ TN_D

> Application Profiles

> Networking

> Contracts

> Policies

> Services

此命令显示从MSO推送的租户，您可以将其用于验证目的。您可以在两个站点的APIC中运行此命令。

```
APIC1# moquery -c fvTenant -f 'fv.Tenant.name=="TN_D"'
```

```
Total Objects shown: 1
```

```
# fv.Tenant
```

```
name          : TN_D
annotation    : orchestrator:misc
childAction   :
descr         :
dn            : uni/tn-TN_D
extMngdBy    : msc
lcOwn         : local
modTs        : 2021-09-17T21:42:52.218+00:00
monPolDn     : uni/tn-common/monepg-default
nameAlias     :
ownerKey      :
ownerTag      :
rn           : tn-TN_D
status       :
uid          : 0
```

```
apic1# moquery -c fvTenant -f 'fv.Tenant.name=="TN_D"'
```

```
Total Objects shown: 1
```

```
# fv.Tenant
```

```
name          : TN_D
```

```
annotation      : orchestrator:msc
childAction     :
descr           :
dn              : uni/tn-TN_D
extMngdBy       : msc
lcOwn           : local
modTs           : 2021-09-17T21:43:04.195+00:00
monPolDn        : uni/tn-common/monepg-default
nameAlias       :
ownerKey        :
ownerTag        :
rn              : tn-TN_D
status          :
uid             : 0
```

配置方案

接下来，创建一个共有三个模板的架构：

1. 站点A的模板：站点A的模板仅与站点A关联，因此该模板中的任何逻辑对象配置都只能推送到站点A的APIC。
2. 站点B的模板：站点B的模板仅与站点B关联，因此该模板中的任何逻辑对象配置都只能推送到站点B的APIC。
3. 拉伸模板：拉伸模板与两个站点关联，并且拉伸模板中的任何逻辑配置都可推送到APIC的两个站点。

创建架构

方案在MSO中具有本地意义，它不在APIC中创建任何对象。方案配置是每个配置的逻辑分离。您可以为同一租户使用多个架构，也可以在每个架构内使用多个模板。

例如，您可以为租户X的数据库服务器设置一个模式，而应用服务器为同一租户X使用不同的模式。这有助于分离每个与应用程序相关的特定配置，并且在您需要调试问题时非常容易。信息也很容易找到。

使用租户名称（例如TN_D_Schema）创建架构。但是，不需要将架构名称以租户名称开头，您可以创建具有任何名称的架构。

步骤1.选择应用程序管理>方案。单击“添加架构”。



步骤2.在“名称”字段中，输入方案的名称。在本例中，它是“TN_D_Schema”，但是，您可以保留适合您环境的任何名称。单击 **Add**。

General
✕

* Name

TN_D_Schema

Description

Schema for Tenant TN_D

Add

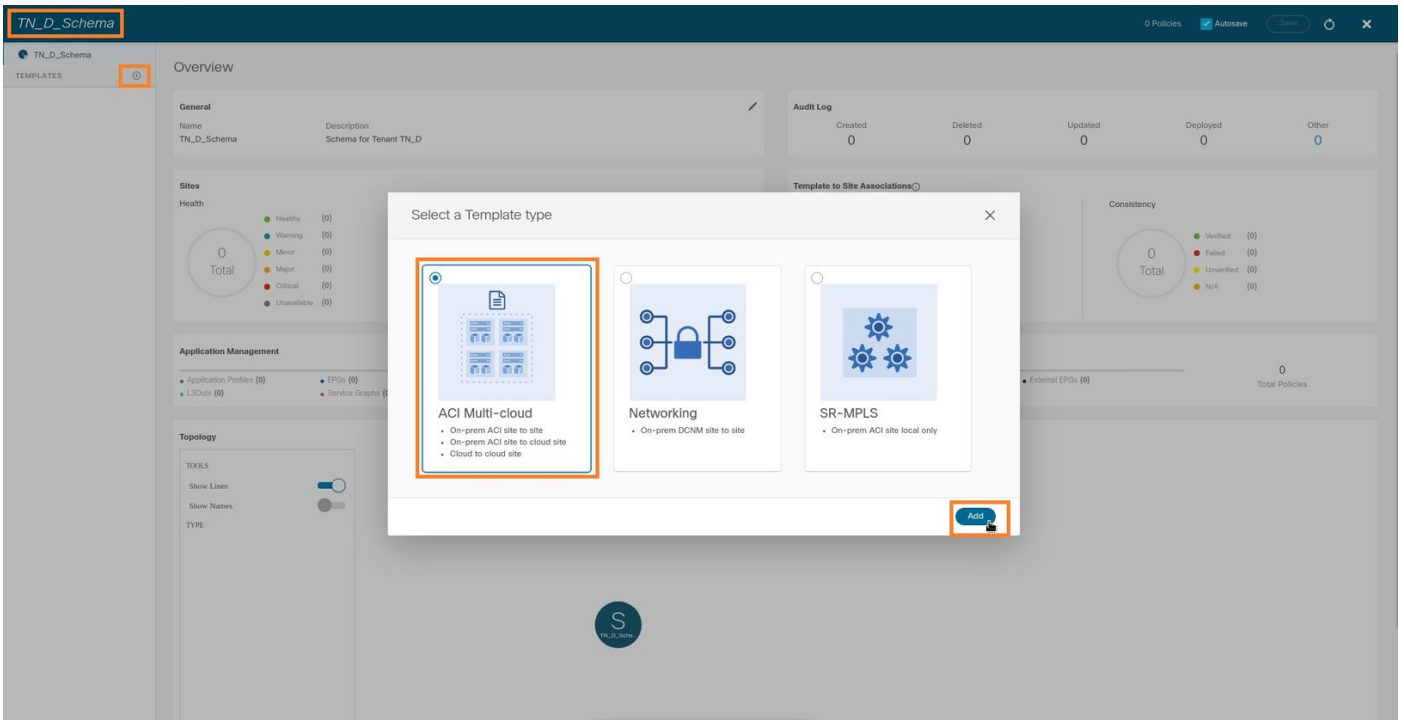
步骤3.检验架构“TN_D_Schema”是否已创建。

The screenshot displays the 'TN_D_Schema' configuration page. The 'General' tab is selected, showing the name 'TN_D_Schema' and the description 'Schema for Tenant TN_D'. The 'Overview' section provides a summary of the template's status, including health (0 total), type (0 total), and consistency (0 total). The 'Application Management' section shows 0 total policies. The interface includes a sidebar with 'TEMPLATES' and a top navigation bar with '0 Policies', 'Autosave', and 'Save' buttons.

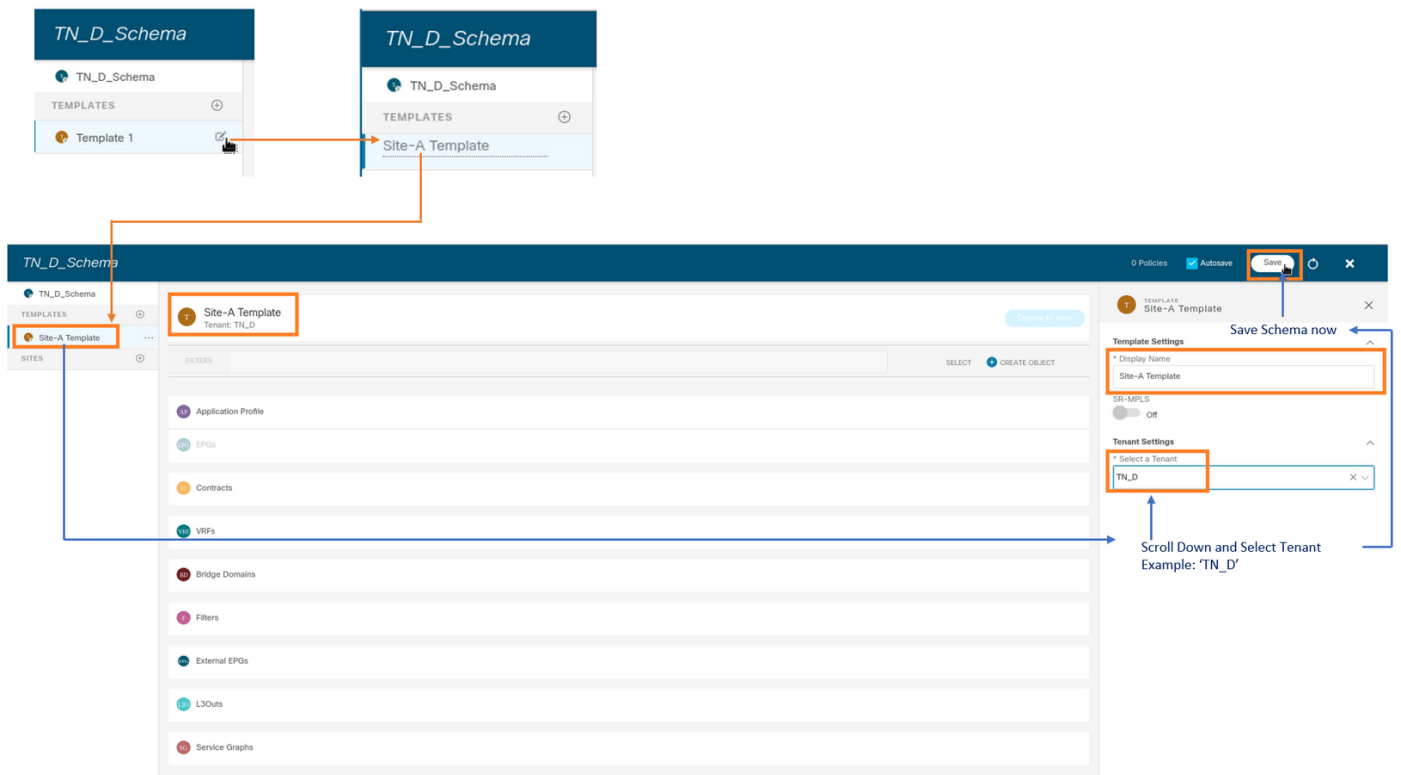
创建站点A模板

步骤1.在架构内添加模板。

1. 要创建模板，请单击已创建的架构下的模板。此时将显示“选择模板类型”对话框。
2. 选择ACI Multi-cloud。
3. 单击 Add。



步骤2.输入模板的名称。此模板特定于站点A，因此模板名称为“站点A模板”。创建模板后，可以将特定租户附加到模板。在本例中，租户“TN_D”已附加。



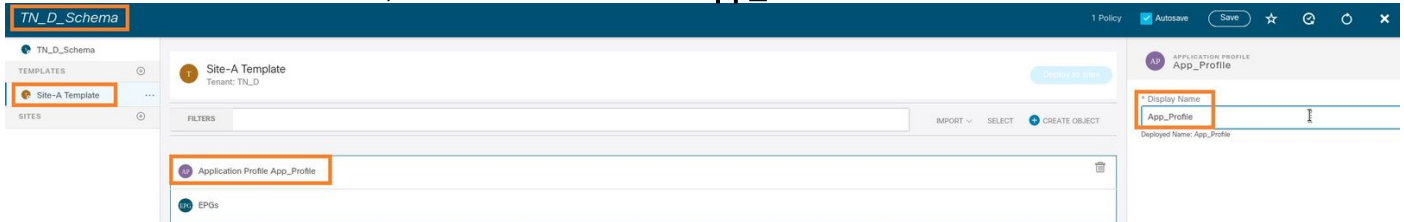
配置模板

应用配置文件配置

步骤1.从您创建的方案中，选择Site-A模板。单击“添加应用配置文件”。



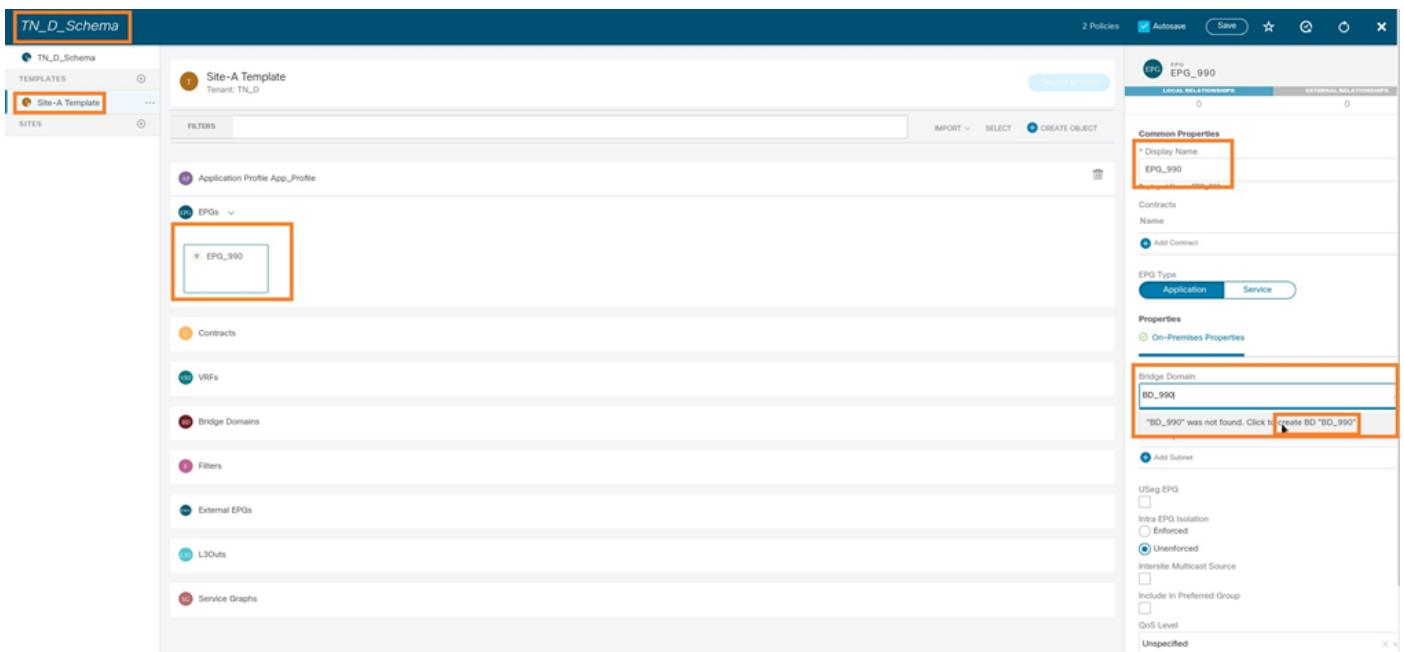
步骤2.在“显示名称”字段中，输入应用配置文件名App_Profile。



步骤3.下一步是创建EPG。要在应用配置文件下添加EPG，请点击Site-A模板下的Add EPG。您可以看到在EPG配置内创建了新EPG。



步骤4.为了将EPG与BD和VRF连接，您必须在EPG下添加BD和VRF。选择Site-A模板。在显示名称字段中，输入EPG的名称并附加新BD（您可以创建新BD或附加现有BD）。

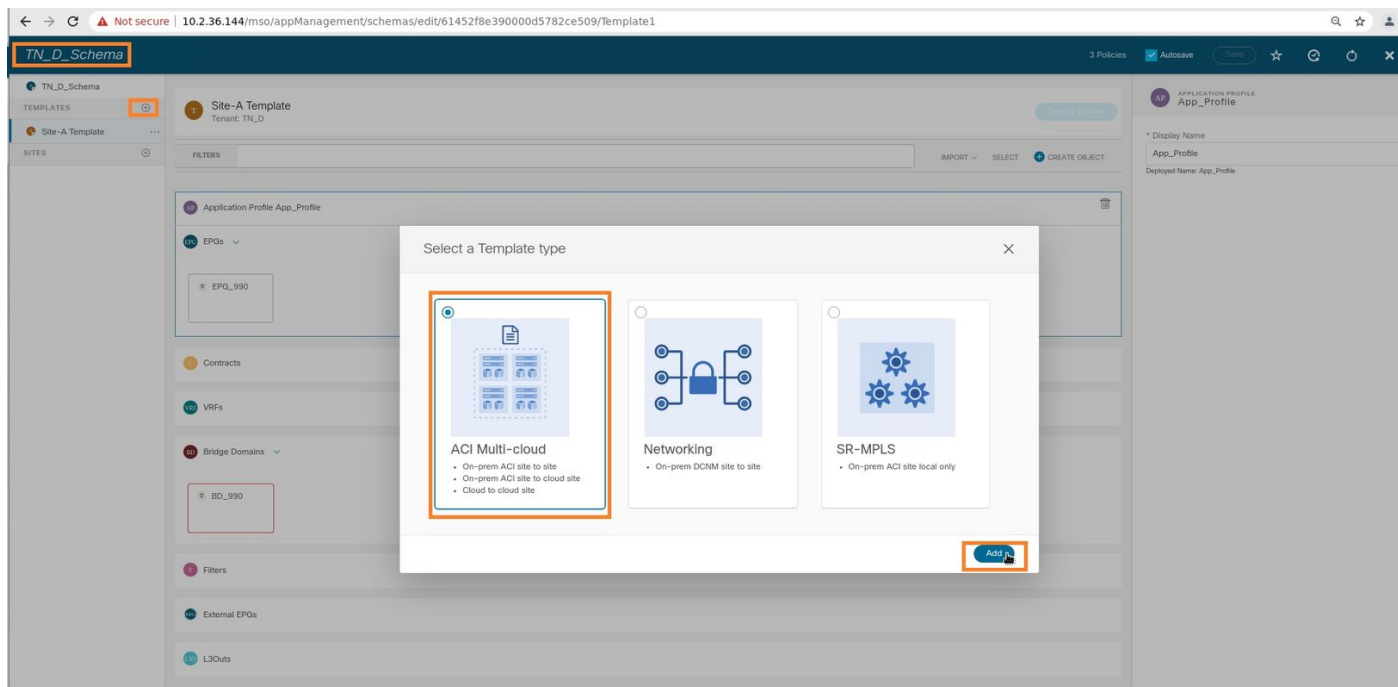


请注意，您必须将VRF连接到BD，但VRF在本例中是拉伸的。您可以使用拉伸VRF创建拉伸模板，然后将该VRF附加到站点特定模板下的BD(在我们的例子中为站点A模板)。

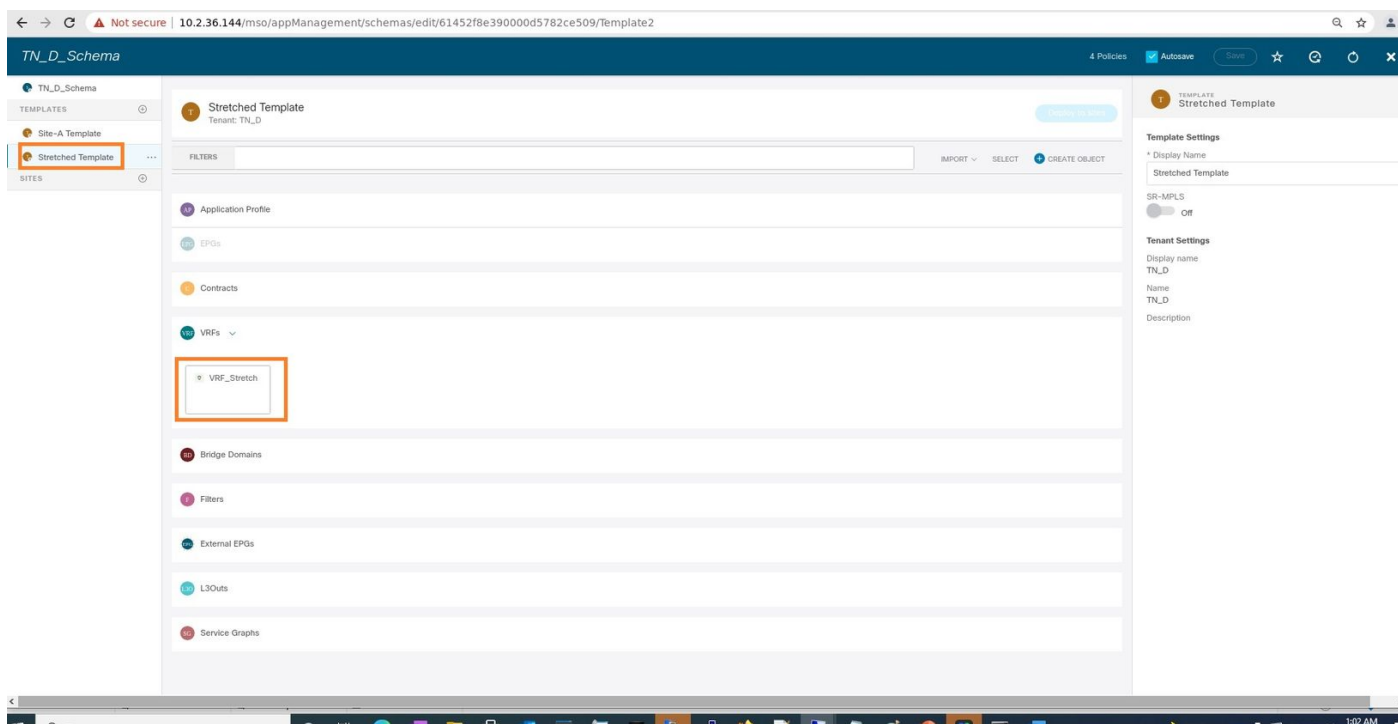
创建拉伸模板

步骤1.要创建拉伸模板，请在TN_D_Schema下单击Templates。此时将显示“选择模板类型”对话框

。选择ACI Multi-cloud。单击 Add。输入模板的名称“延伸模板”。（可以为拉伸模板输入任何名称。）

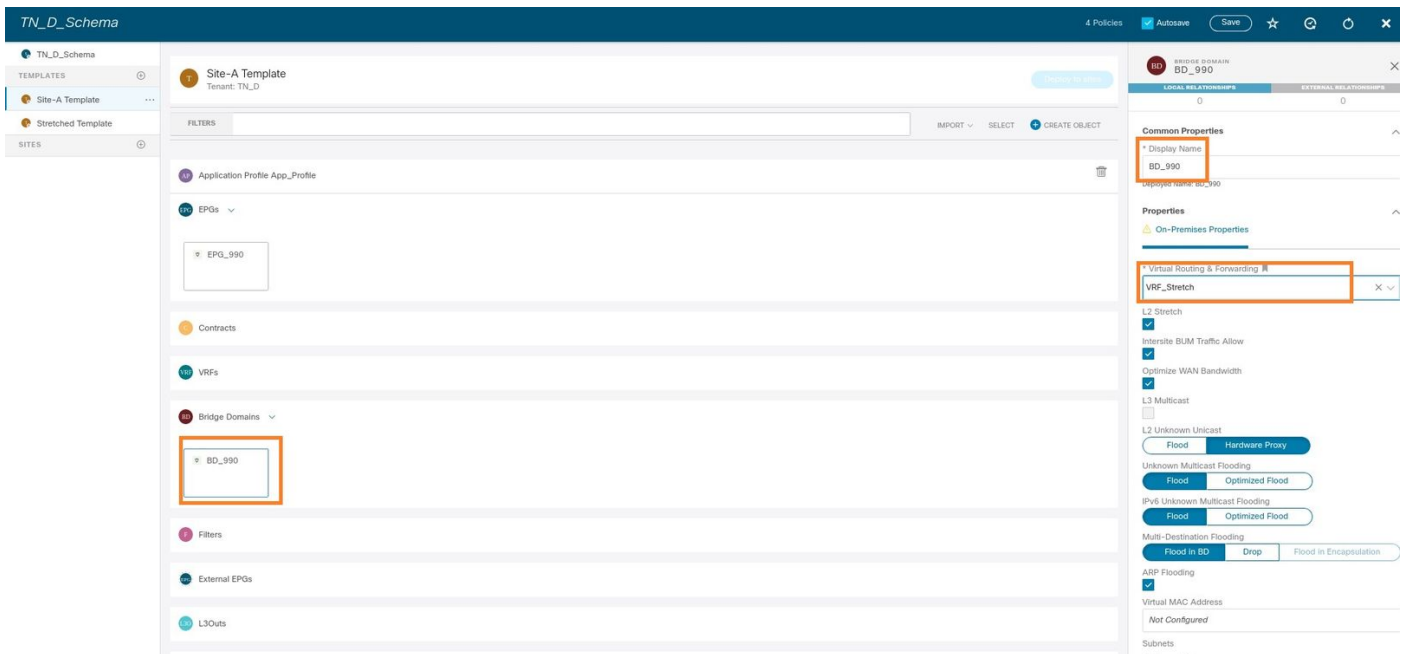


步骤2.选择“拉伸模板”并创建名为VRF_Stretch的VRF。（可以输入VRF的任何名称。）



BD是通过在Site-A模板下创建EPG而创建的，但没有附加VRF，因此您必须附加VRF，该VRF现在在延伸模板中创建。

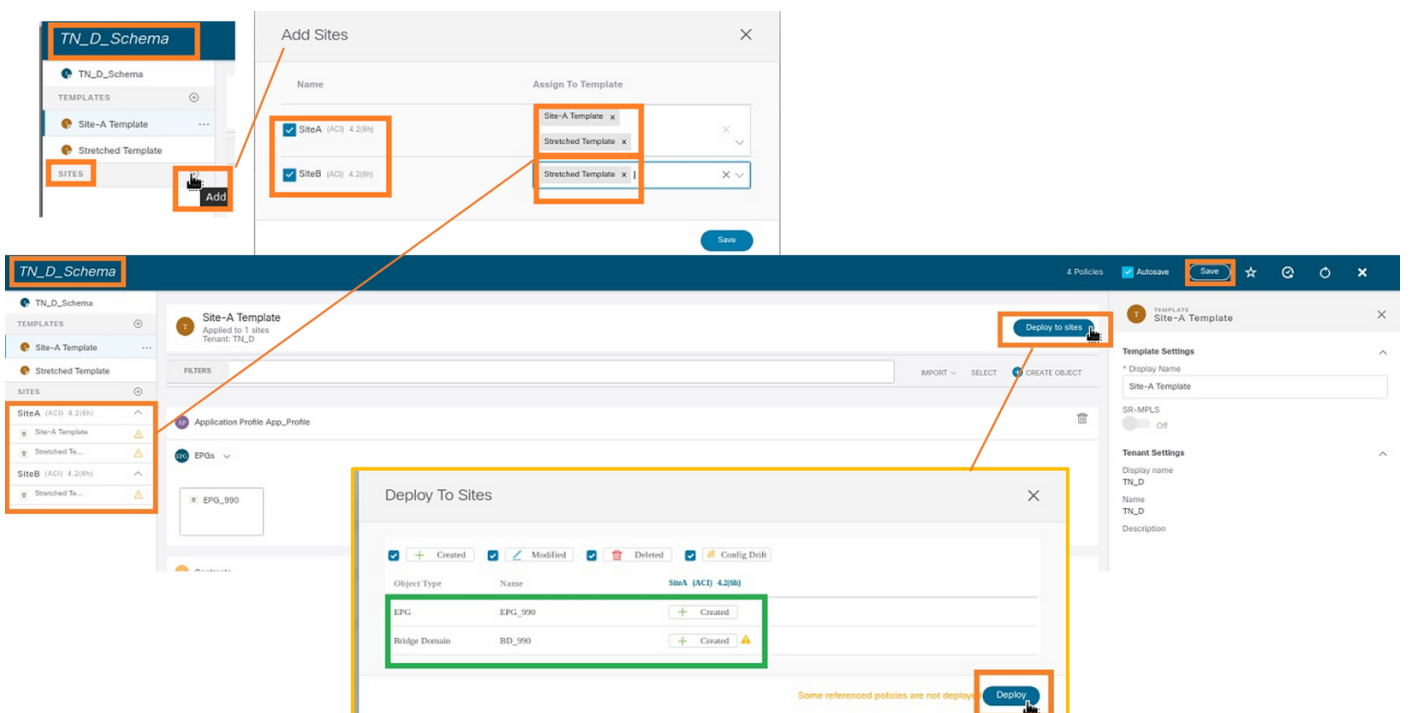
步骤3.选择Site-A Template > BD_990。在“虚拟路由和转发”下拉列表中，选择VRF_Stretch。（您在本节步骤2中创建的。）



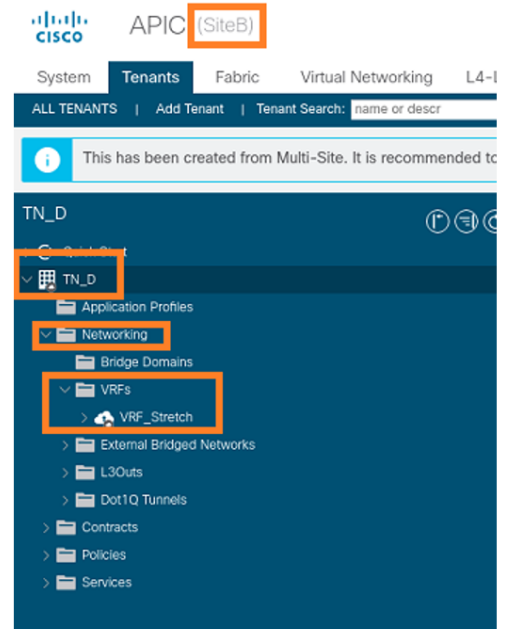
附加模板

下一步是仅将**Site-A模板**附加**Site-A**，并且拉伸模板需要附加到两个站点。单击**Deploy to site inside the schema**，以将模板部署到各个站点。

步骤1.单击TN_D_Schema > **SITES**下的**+**号将站点添加到模板。在**分配到模板**下拉列表中，选择相应站点的相应模板。



步骤2.您可以看到**站点A**现在已创建EPG和BD，但**站点B**没有创建相同的EPG/BD，因为这些配置仅适用于来自MSO的**站点A**。但是，您可以看到VRF是在拉伸模板中创建的，因此在两个站点中创建



步骤3.使用这些命令检验配置。

```

APIC1# moquery -c fvAEPg -f 'fv.AEPg.name=="EPG_990"'
Total Objects shown: 1
# fv.AEPg
name                : EPG_990
annotation          : orchestrator:msc
childAction         :
configIssues        :
configSt            : applied
descr               :
dn                  : uni/tn-TN_D/ap-App_Profile/epg-EPG_990
exceptionTag        :
extMngdBy           :
floodOnEncap        : disabled
fwdCtrl             :
hasMcastSource      : no
isAttrBasedEPg     : no
isSharedSrvMsiteEPg : no
lcOwn               : local
matchT              : AtleastOne
modTs               : 2021-09-18T08:26:49.906+00:00
monPolDn            : uni/tn-common/monepg-default
nameAlias           :
pcEnfPref           : unenforced
pctag               : 32770
prefGrMemb          : exclude
prio                : unspecified
rn                  : epg-EPG_990
scope               : 2850817
shutdown            : no
status              :
triggerSt           : triggerable
txId                : 1152921504609182523
uid                 : 0

```

```

APIC1# moquery -c fvBD -f 'fv.BD.name=="BD_990"'
Total Objects shown: 1
# fv.BD
name : BD_990
OptimizeWanBandwidth : yes
annotation : orchestrator:misc
arpFlood : yes
bcastP : 225.0.56.224
childAction :
configIssues :
descr :
dn : uni/tn-TN_D/BD-BD_990
epClear : no
epMoveDetectMode :
extMngdBy :
hostBasedRouting : no
intersiteBumTrafficAllow : yes
intersiteL2Stretch : yes
ipLearning : yes
ipv6McastAllow : no
lcOwn : local
limitIpLearnToSubnets : yes
llAddr : ::
mac : 00:22:BD:F8:19:FF
mcastAllow : no
modTs : 2021-09-18T08:26:49.906+00:00
monPolDn : uni/tn-common/monepg-default
mtu : inherit
multiDstPktAct : bd-flood
nameAlias :
ownerKey :
ownerTag :
pcTag : 16387
rn : BD-BD_990
scope : 2850817
seg : 16580488
status :
type : regular
uid : 0
unicastRoute : yes
unkMacUcastAct : proxy
unkMcastAct : flood
v6unkMcastAct : flood
vmac : not-applicable
: 0

```

```

APIC1# moquery -c fvCtx -f 'fv.Ctx.name=="VRF_Stretch"'
Total Objects shown: 1
# fv.Ctx
name : VRF_Stretch
annotation : orchestrator:misc
bdEnforcedEnable : no
childAction :
descr :
dn : uni/tn-TN_D/ctx-VRF_Stretch
extMngdBy :
ipDataPlaneLearning : enabled
knwMcastAct : permit
lcOwn : local
modTs : 2021-09-18T08:26:58.185+00:00

```

```

monPolDn      : uni/tn-common/monepg-default
nameAlias     :
ownerKey      :
ownerTag      :
pcEnfDir     : ingress
pcEnfDirUpdated : yes
pcEnfPref    : enforced
pcTag        : 16386
rn           : ctx-VRF_Stretch
scope        : 2850817
seg          : 2850817
status       :
uid          : 0

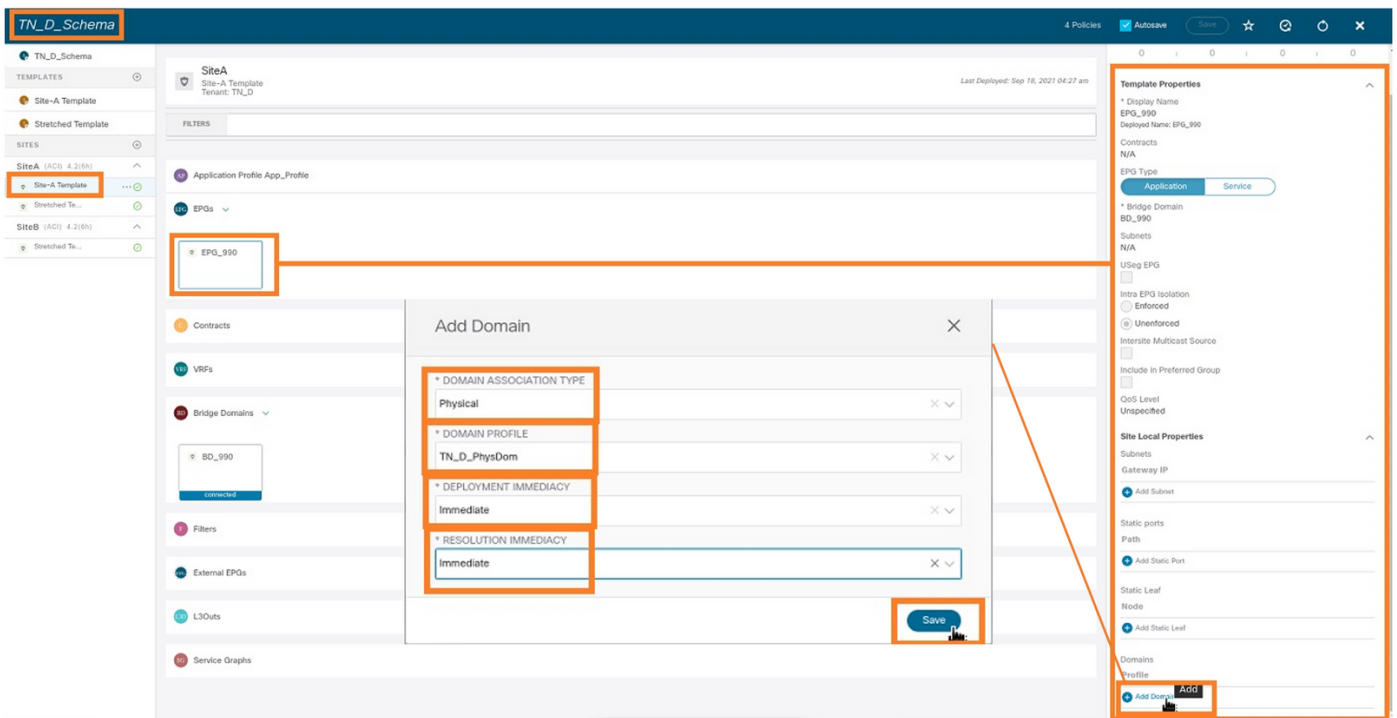
```

配置静态端口绑定

现在，您可以在EPG“EPG_990”下配置静态端口绑定，也可以使用VRF HOST_A配置N9K（基本上是模拟HOST_A）。ACI端静态端口绑定配置将首先完成。

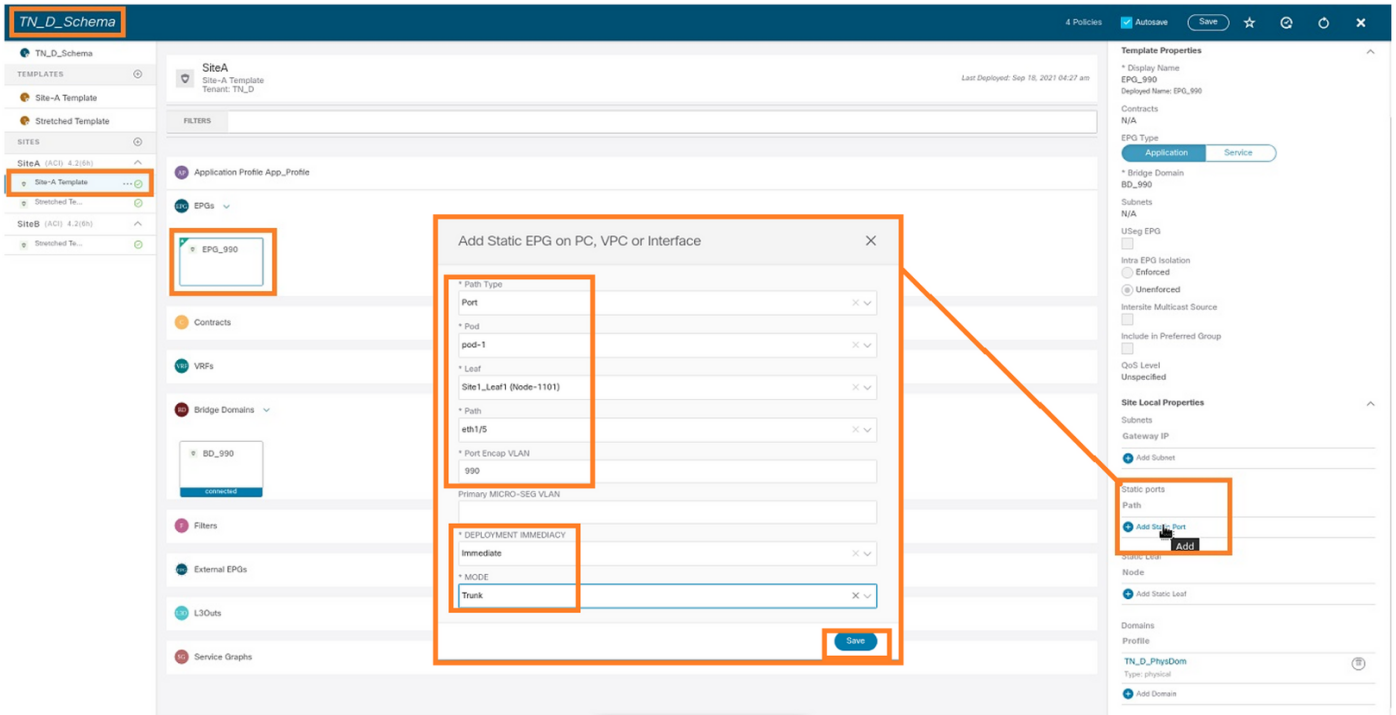
步骤1.在EPG_990下添加物理域。

1. 从您创建的架构中，选择**Site-A Template > EPG_990**。
2. 在“模板属性”框中，单击**添加域**。
3. 在“添加域”对话框中，从下拉列表中选择以下选项：域关联类型 — 物理域配置文件-TN_D_PhysDom部署即时性 — 即时解决方案即时性 — 即时
4. Click **Save**.

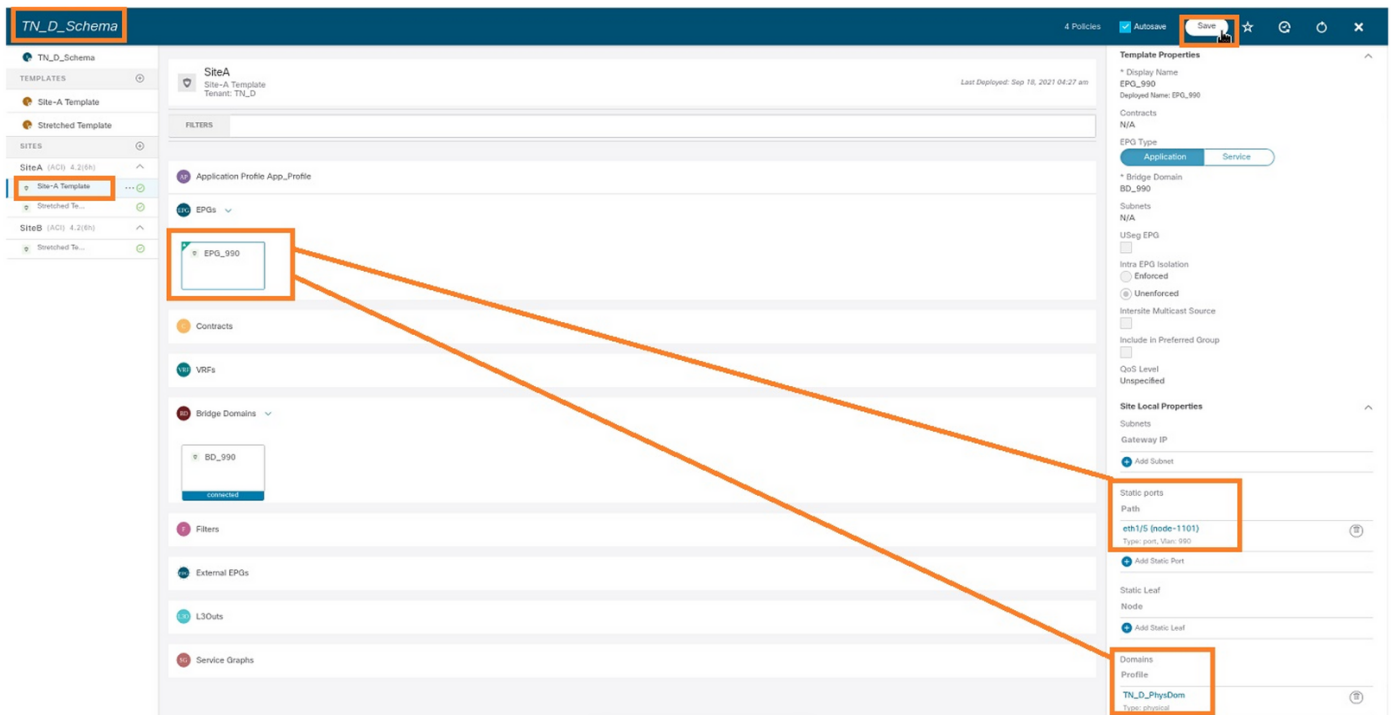


步骤2.添加静态端口(Site1_Leaf1 eth1/5)。

1. 从您创建的架构中，选择**Site-A Template > EPG_990**。
2. 在“模板属性”框中，单击**添加静态端口**。
3. 在“在PC、VPC或接口上添加静态EPG”对话框中，选择**Node-101 eth1/5**并分配**VLAN 990**。



步骤3.确保在EPG_990下添加静态端口和物理域。



使用以下命令验证静态路径绑定：

```

APIC1# moquery -c fvStPathAtt -f 'fv.StPathAtt.pathName=="eth1/5"' | grep EPG_990 -A 10 -B 5
# fv.StPathAtt
pathName      : eth1/5
childAction   :
descr         :
dn            : uni/epp/fv-[uni/tn-TN_D/ap-App_Profile/epg-EPG_990]/node-1101/stpathatt-[eth1/5]
lcOwn         : local
modTs        : 2021-09-19T06:16:46.226+00:00
monPolDn     : uni/tn-common/monepg-default
name         :

```

```

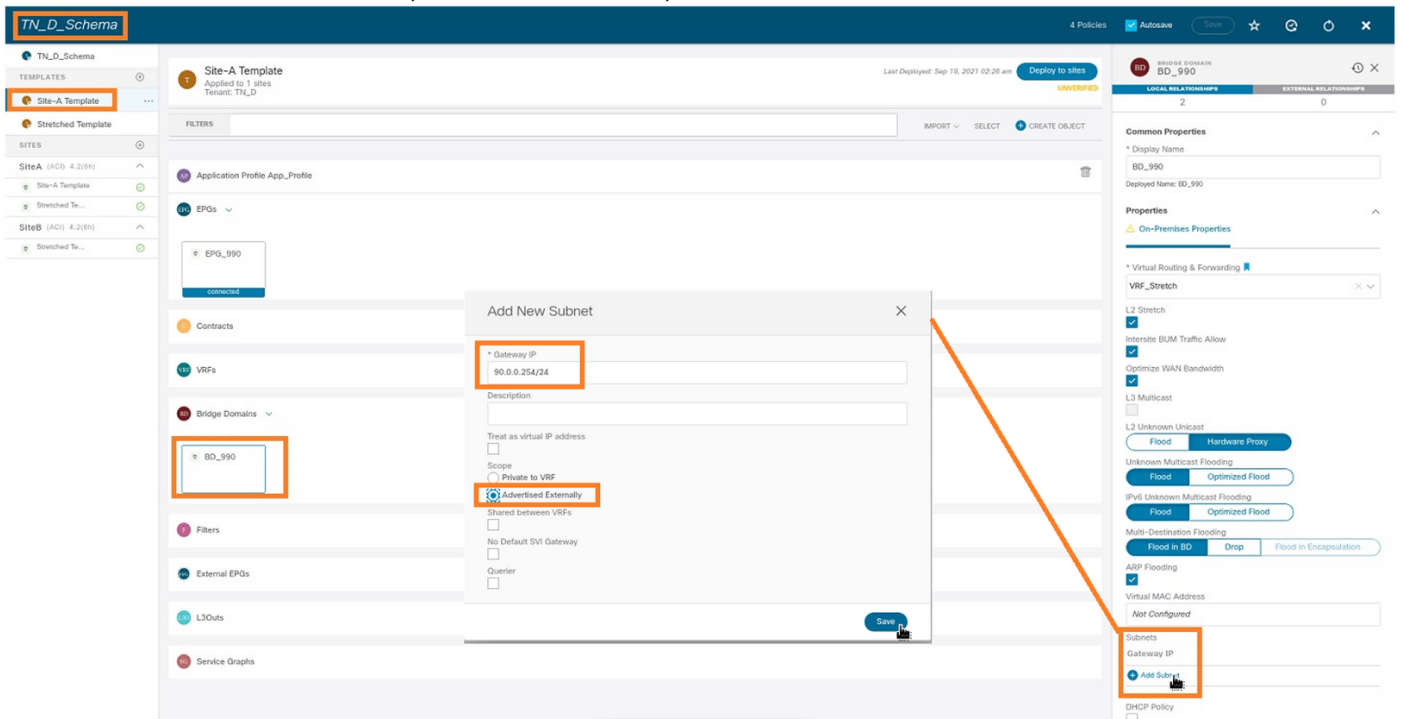
nameAlias      :
ownerKey       :
ownerTag       :
rn             : stpathatt-[eth1/5]
status        :

```

配置BD

步骤1.在BD下添加子网/IP (HOST_A使用BD IP作为网关)。

1. 从您创建的架构中，选择Site-A Template > BD_990。
2. 单击Add Subnet。
3. 在添加新子网对话框中，输入网关IP地址，然后单击外部通告单选按钮。



步骤2.使用此命令检验子网是否已添加到APIC1 Site-A中。

```

APIC1# moquery -c fvSubnet -f 'fv.Subnet.ip=="90.0.0.254/24"'
Total Objects shown: 1

```

```

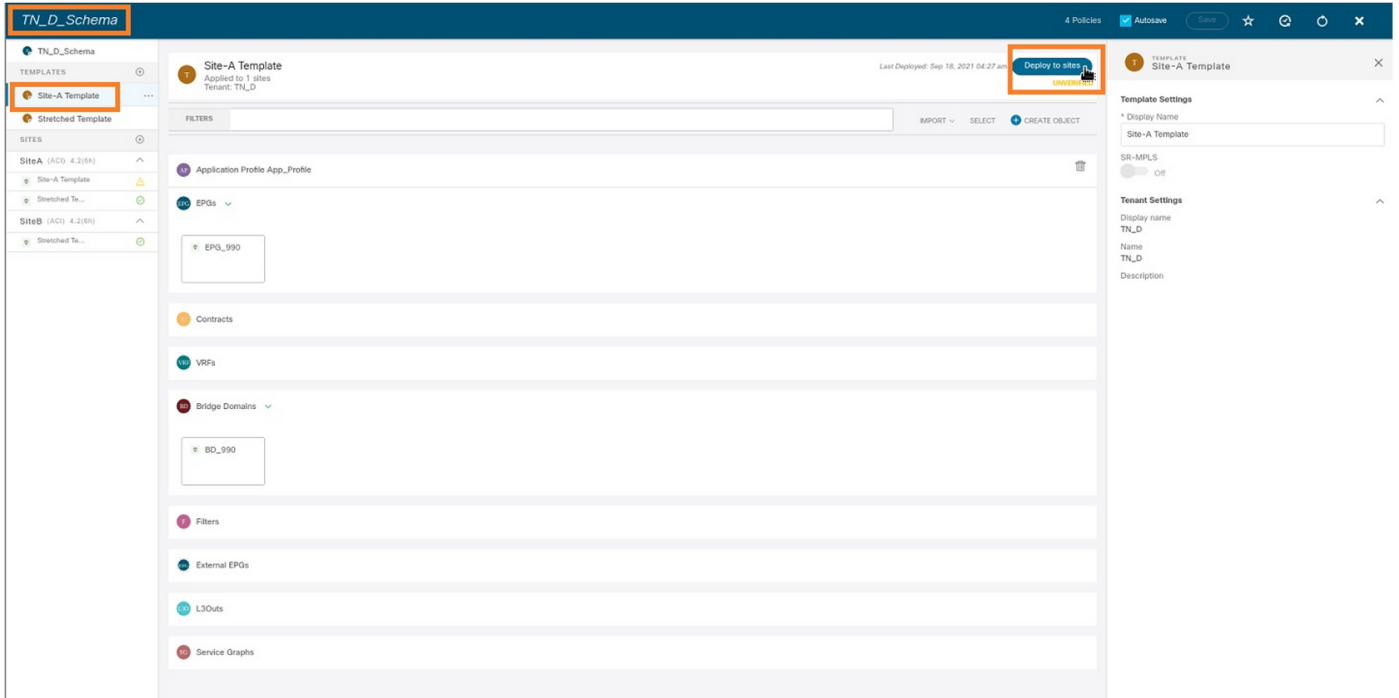
# fv.Subnet
ip           : 90.0.0.254/24
annotation   : orchestrator:misc
childAction  :
ctrl         : nd
descr       :
dn          : uni/tn-TN_D/BD-BD_990/subnet-[90.0.0.254/24]
extMngdBy   :
lcOwn       : local
modTs       : 2021-09-19T06:33:19.943+00:00
monPolDn    : uni/tn-common/monepg-default
name        :
nameAlias   :
preferred   : no
rn          : subnet-[90.0.0.254/24]
scope       : public
status      :

```

uid : 0
virtual : no

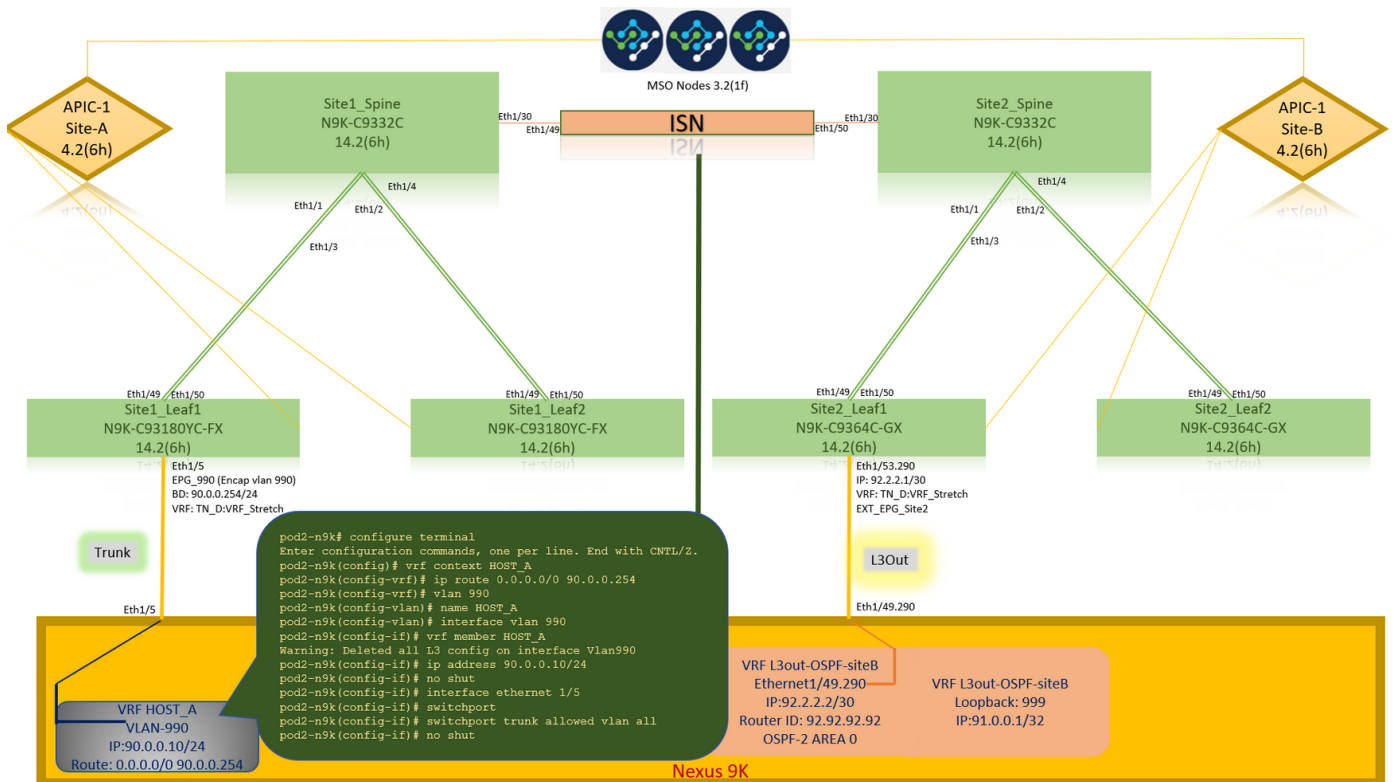
步骤3. 部署Site-A模板。

1. 从您创建的架构中，选择Site-A Template。
2. 单击“部署到站点”。



配置主机A(N9K)

使用VRF HOST_A配置N9K设备。完成N9K配置后，您可以看到ACI枝叶BD任播地址 (HOST_A的网关) 现在可通过ICMP(ping)到达。



在ACI操作选项卡中，您可以看到90.0.0.10 (HOST_A IP地址) 已获知。

The screenshot shows the Cisco APIC interface for a tenant named TN_D. The left sidebar shows the navigation tree with 'EPG_990' selected. The main area displays the configuration for 'EPG - EPG_990' under the 'Operational' tab. A table lists the learned end-points:

End Point	MAC	IP	Learning Source	Hosting Server	Reporting Controller Name	Interface	Multicast Address	Encap
EP-C0:14.FE.5E:1...	C0:14.FE.5E:14:07	90.0.0.10	learned	---	---	Pod-1/Node-1101/eth1/5 (learned)	---	vlan-990

Below the table, a terminal window shows the following output:

```
pod2-n9k# ping 90.0.0.254 vrf HOST_A
PING 90.0.0.254 (90.0.0.254): 56 data bytes
36 bytes from 90.0.0.10: Destination Host Unreachable
Request 0 timed out
64 bytes from 90.0.0.254: icmp_seq=1 ttl=63 time=0.902 ms
64 bytes from 90.0.0.254: icmp_seq=2 ttl=63 time=0.576 ms
64 bytes from 90.0.0.254: icmp_seq=3 ttl=63 time=0.708 ms
64 bytes from 90.0.0.254: icmp_seq=4 ttl=63 time=0.659 ms

--- 90.0.0.254 ping statistics ---
5 packets transmitted, 4 packets received, 20.00% packet loss
round-trip min/avg/max = 0.576/0.711/0.902 ms
pod2-n9k#
```

Additional details shown include the interface configuration for Eth1/5 (VLAN-990) and the VRF configuration for VRF HOST_A (VLAN-990, IP: 90.0.0.10/24).

创建站点B模板

步骤1.从您创建的方案中，选择TEMPLATES。单击+号，然后创建名为Site-B Template的模板。

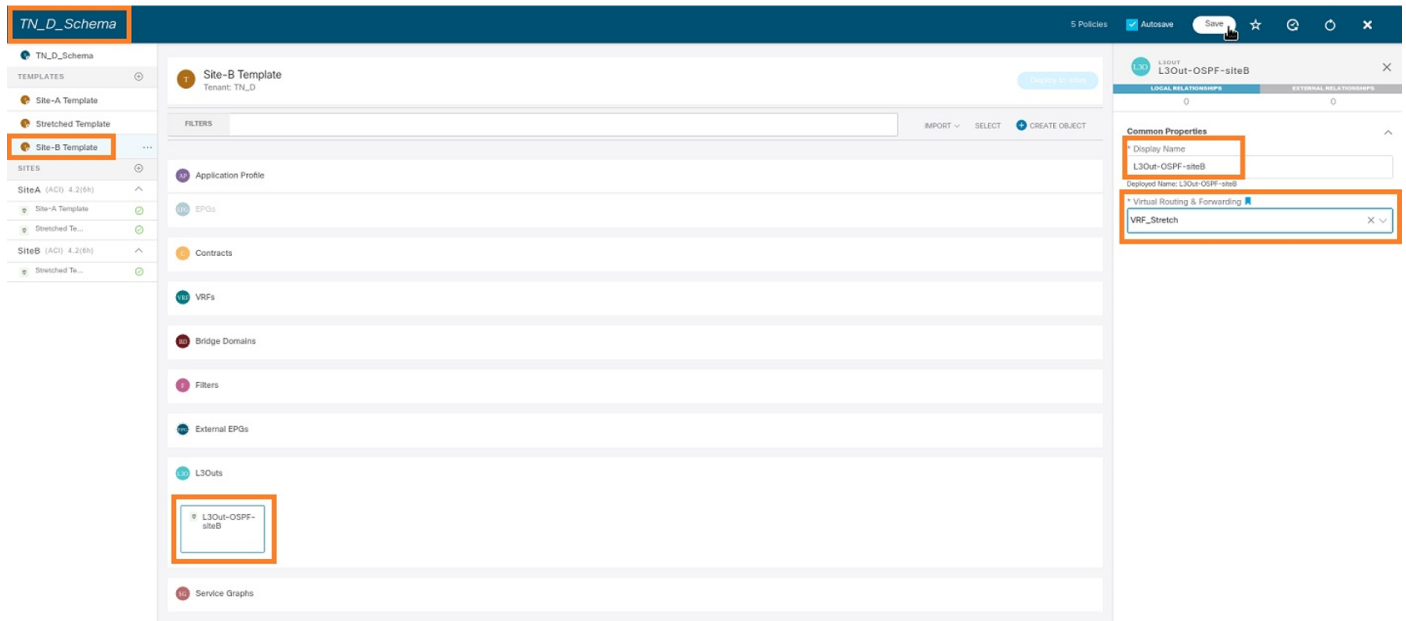


This screenshot shows the configuration page for the 'Site-B Template' in the APIC interface. The left sidebar shows the 'TEMPLATES' section with 'Site-B Template' selected. The main area displays the configuration for the template, including the 'Tenant Settings' section where the 'Display name' is set to 'TN_D' and the 'Name' is set to 'TN_D'.

配置站点B L3out

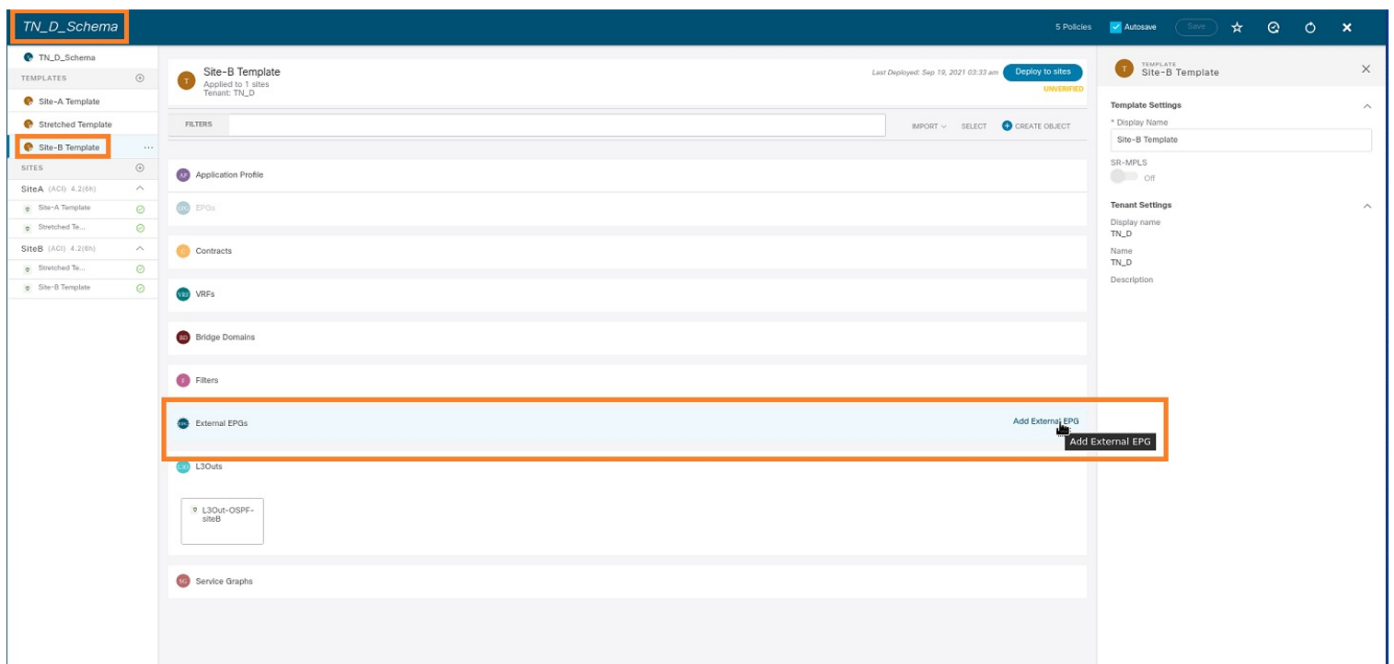
创建L3out并连接VRF_Stretch。您必须从MSO创建L3out对象，其余L3out配置需要从APIC完成（因为L3out参数在MSO中不可用）。此外，从MSO创建外部EPG（仅在站点B模板中，因为外部EPG不延伸）。

步骤1.从您创建的方案中，选择Site-B Template。在“显示名称”字段中，输入L3out_OSPF_siteB。在虚拟路由和转发下拉列表中，选择VRF_Stretch。



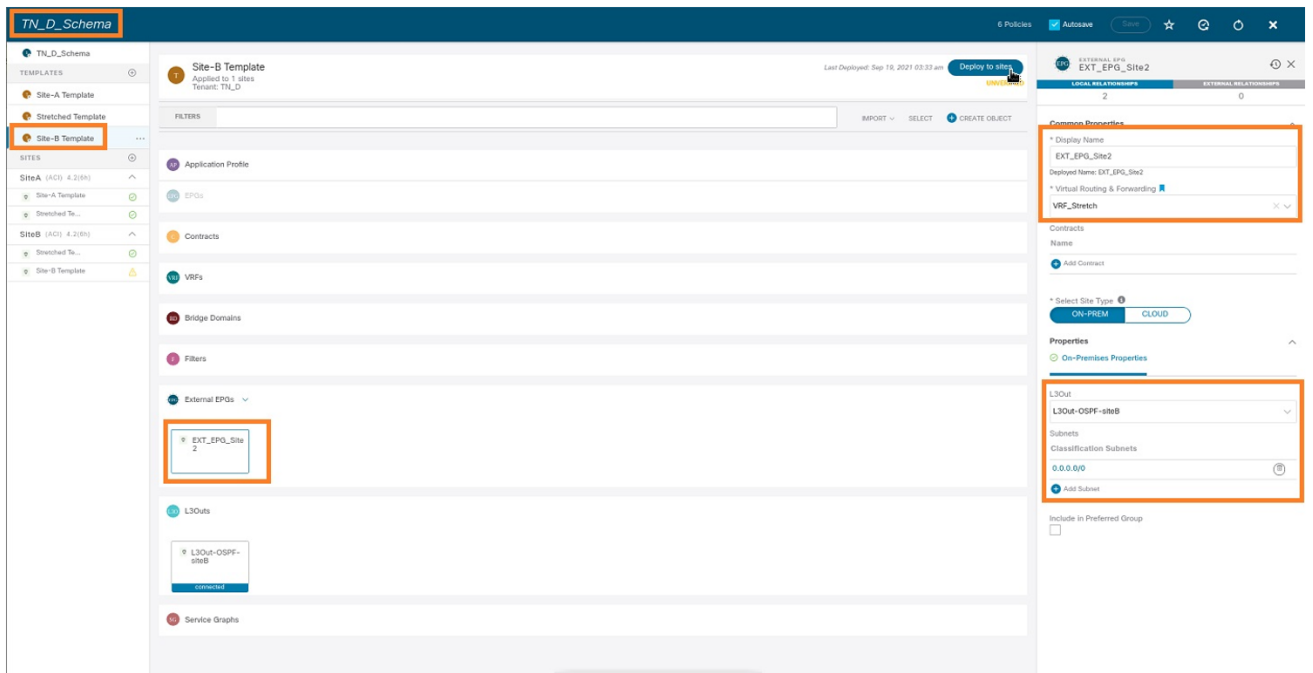
创建外部EPG

步骤1.从您创建的方案中，选择Site-B Template。单击Add External EPG。



步骤2.将L3out与外部EPG连接。

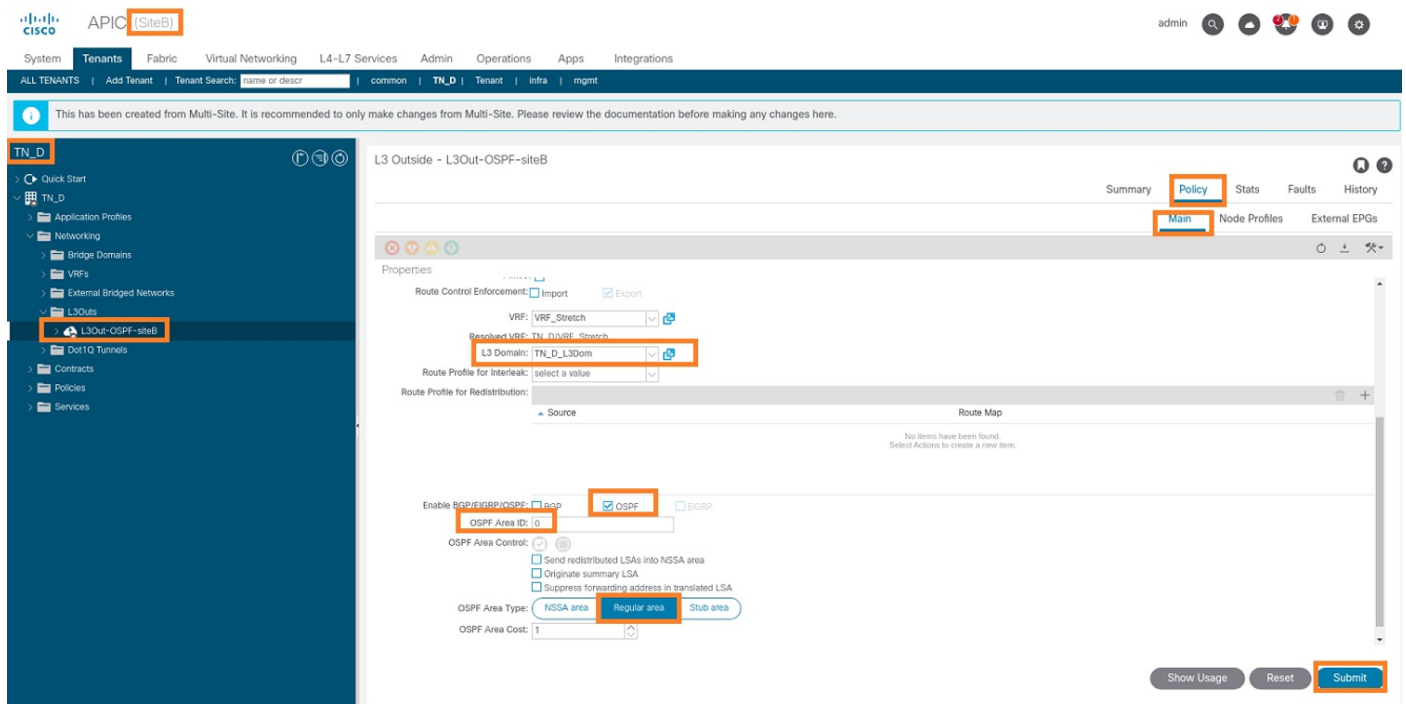
1. 从您创建的架构中，选择Site-B Template。
2. 在“显示名称”字段中，输入EXT_EPG_Site2。
3. 在分类子网字段中，输入0.0.0.0/0作为外部EPG的外部子网。



其余L3out配置从APIC (站点B) 完成。

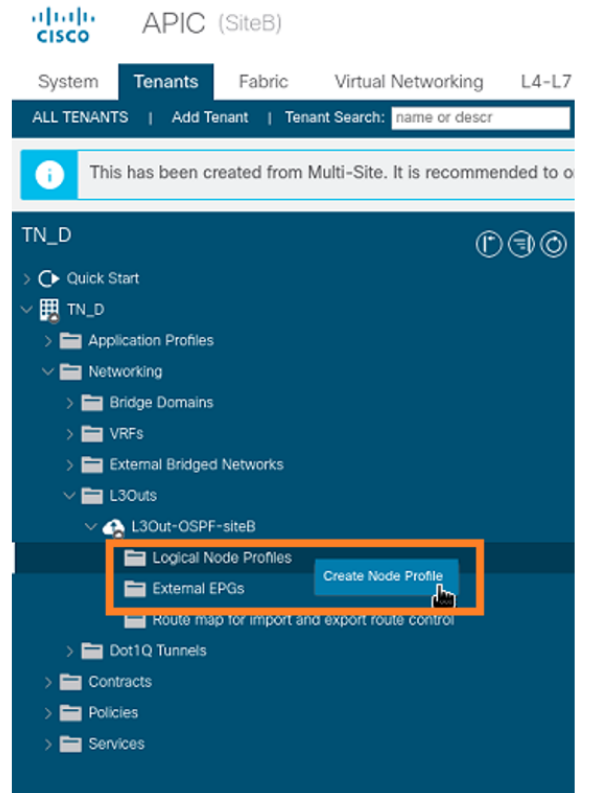
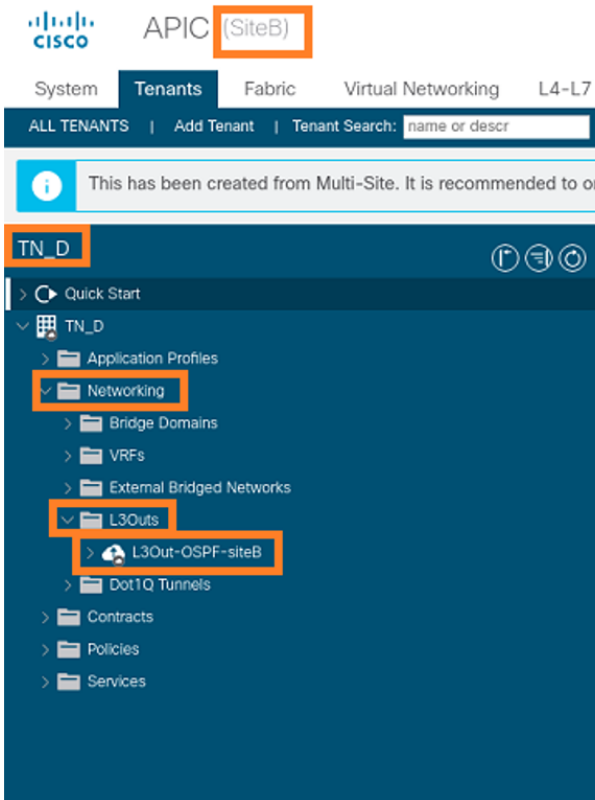
步骤3.添加L3域，启用OSPF协议，并使用常规区域0配置OSPF。

1. 从Site-B的APIC-1，选择TN_D > Networking > L3out-OSPF-siteB > Policy > Main。
2. 在L3域下拉列表中，选择TN_D_L3Dom。
3. 选中启用BGP/EIGRP/OSPF的OSPF复选框。
4. 在OSPF区域ID字段中，输入0。
5. 在OSPF区域类型中，选择常规区域。
6. 单击“Submit”。



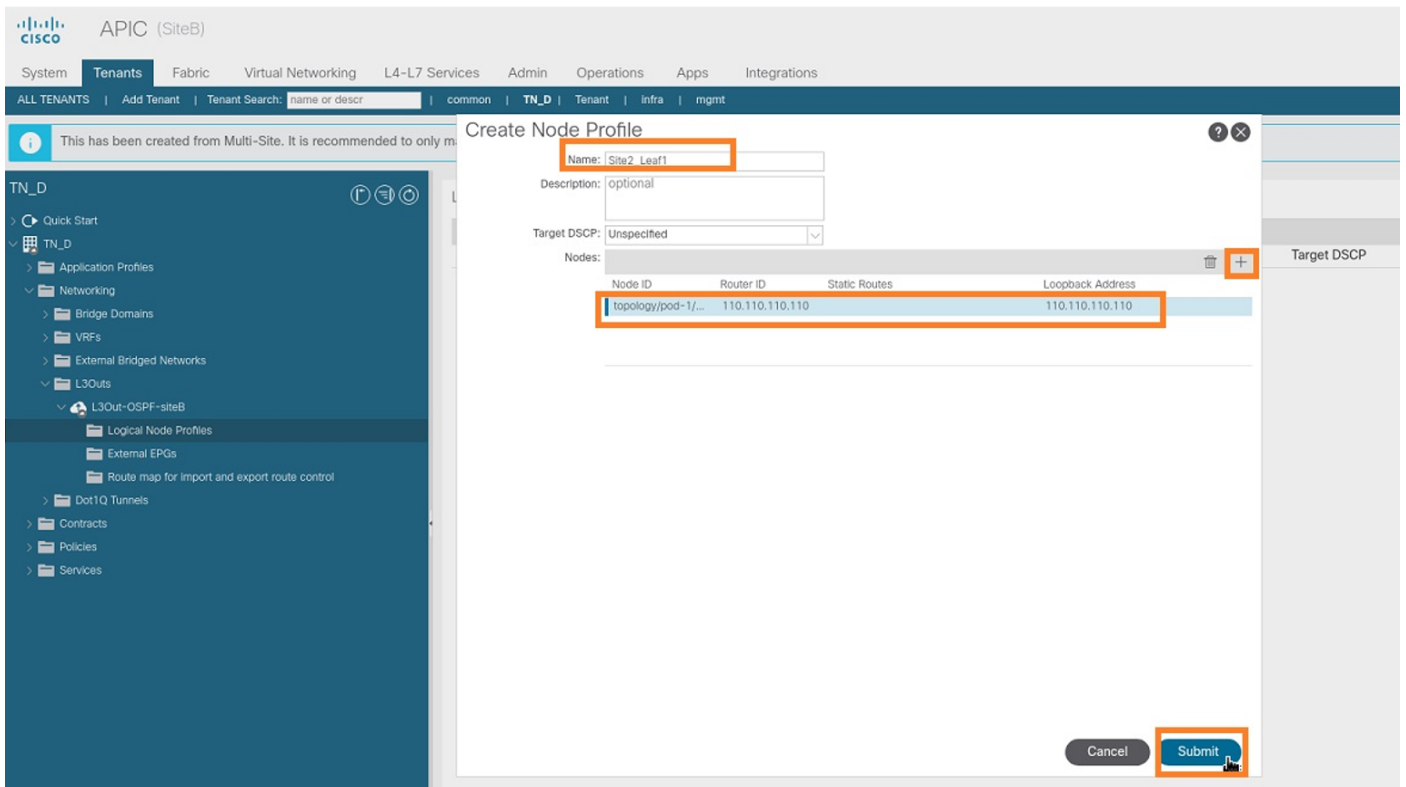
步骤4.创建节点配置文件。

1. 从Site-B的APIC-1，选择TN_D > Networking > L3Outs > L3Out-OSPF-siteB > Logical Node Profiles。
2. 单击“创建节点配置文件”。



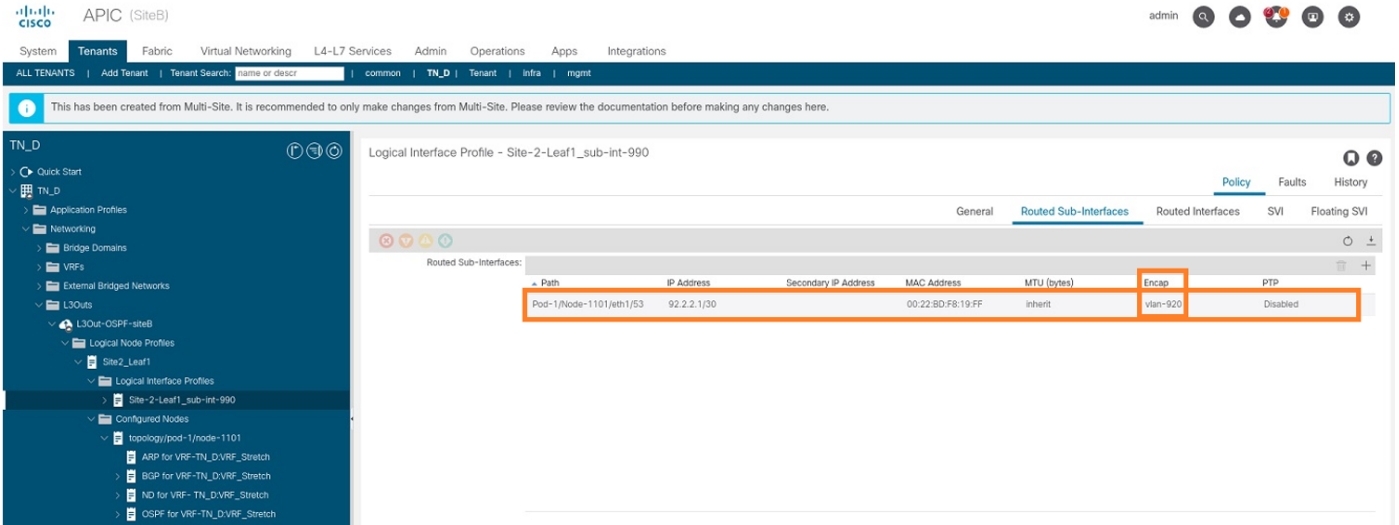
步骤5.选择交换机Site2_Leaf1作为站点B的节点。

1. 从Site-B的APIC-1，选择TN_D > Networking > L3Outs > L3Out-OSPF-siteB > Logical Node Profiles > Create Node Profile。
2. 在“名称”字段中，输入Site2_Leaf1。
3. 单击+号添加节点。
4. 使用路由器ID IP地址添加Pod-2节点101。



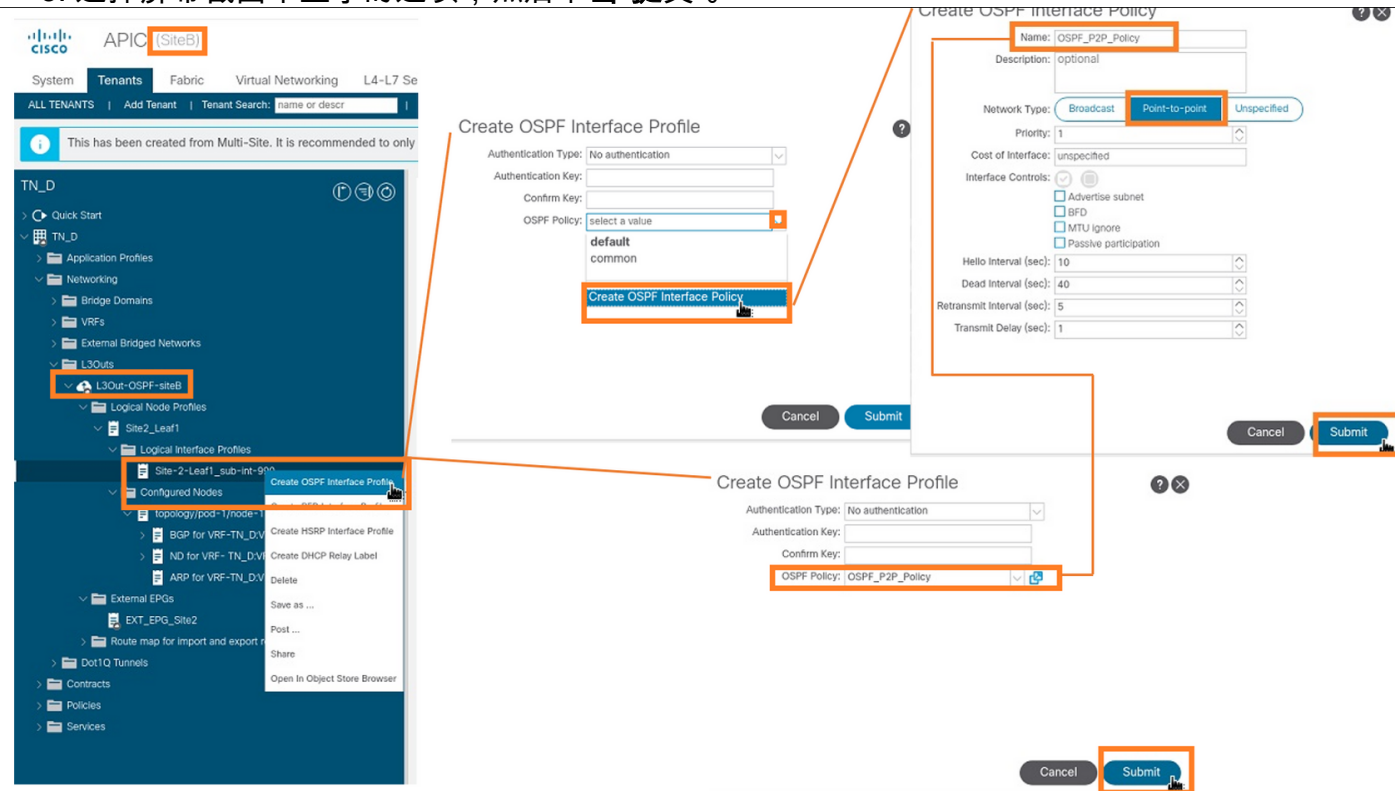
步骤6.添加接口配置文件(外部VLAN为920 (SVI创建))。

1. 从Site-B的APIC-1，选择TN_D > Networking > L3Outs > L3out-OSPF-SiteB > Logical Interface Profiles。
2. 右键单击并添加接口配置文件。
3. 选择Routed Sub-Interfaces。
4. 配置IP地址、MTU和VLAN-920。

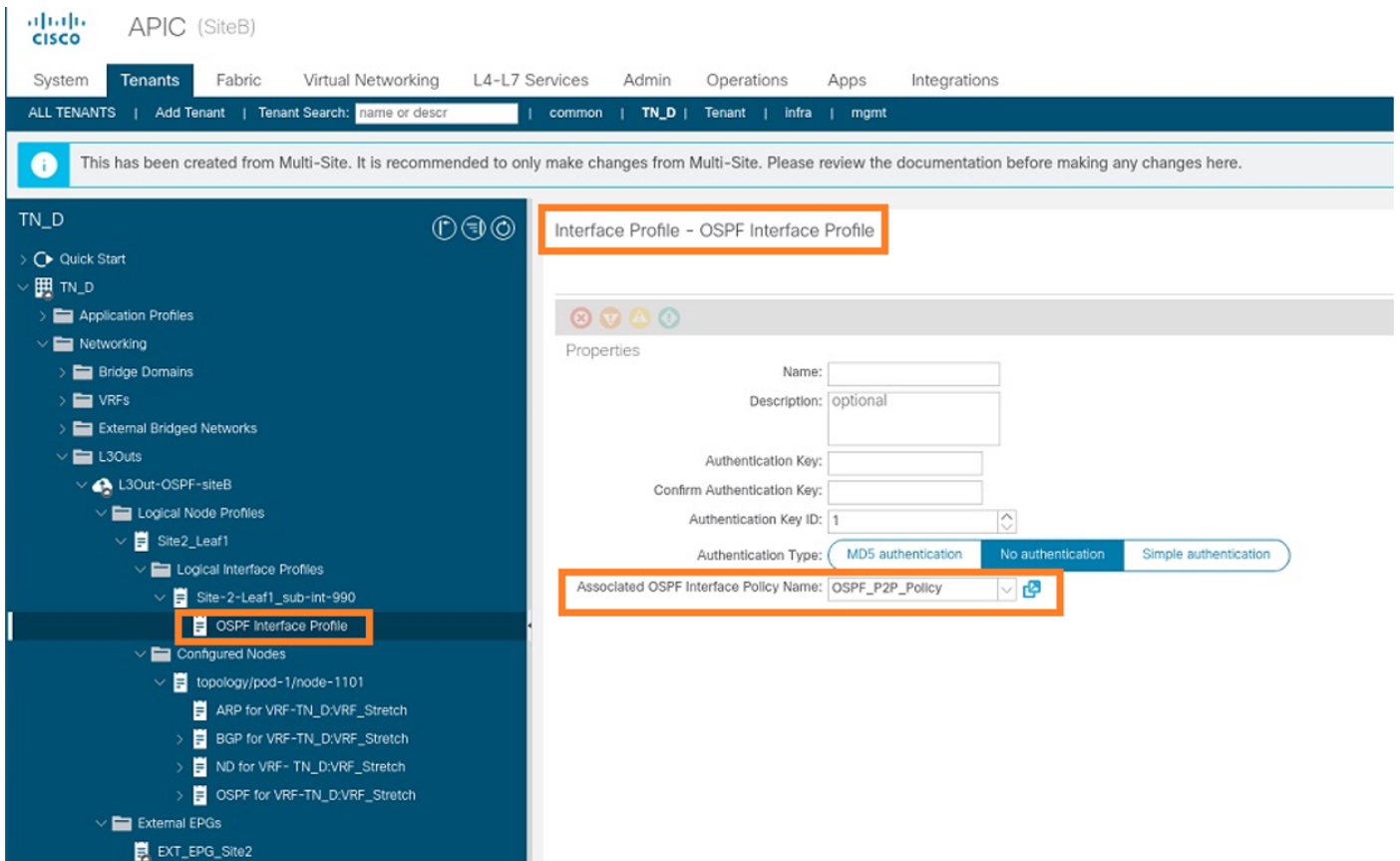


步骤7.创建OSPF策略（点对点网络）。

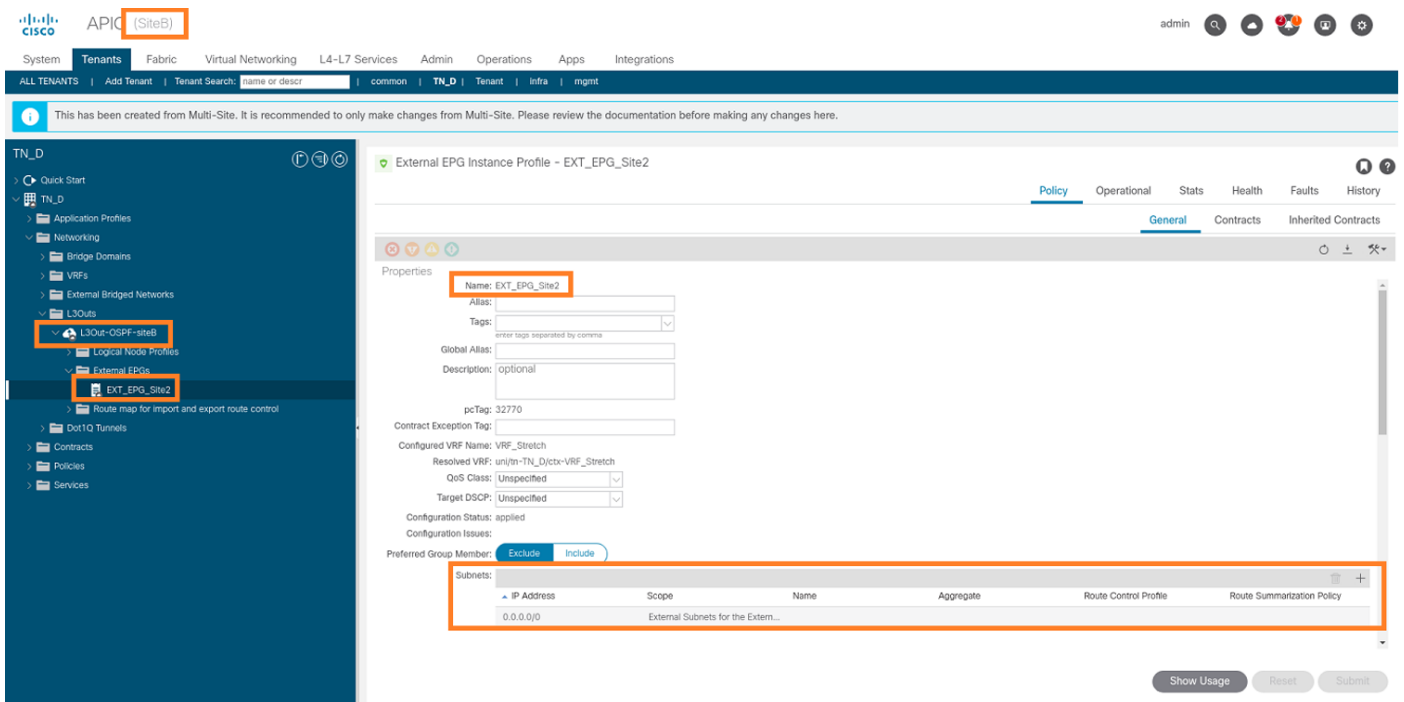
1. 从Site-B的APIC-1中，选择TN_D > Networking > L3Outs > L3Out-OSPF-siteB > Logical Interface Profiles。
2. 右键单击并选择Create OSPF Interface Profile。
3. 选择屏幕截图中显示的选项，然后单击“提交”。



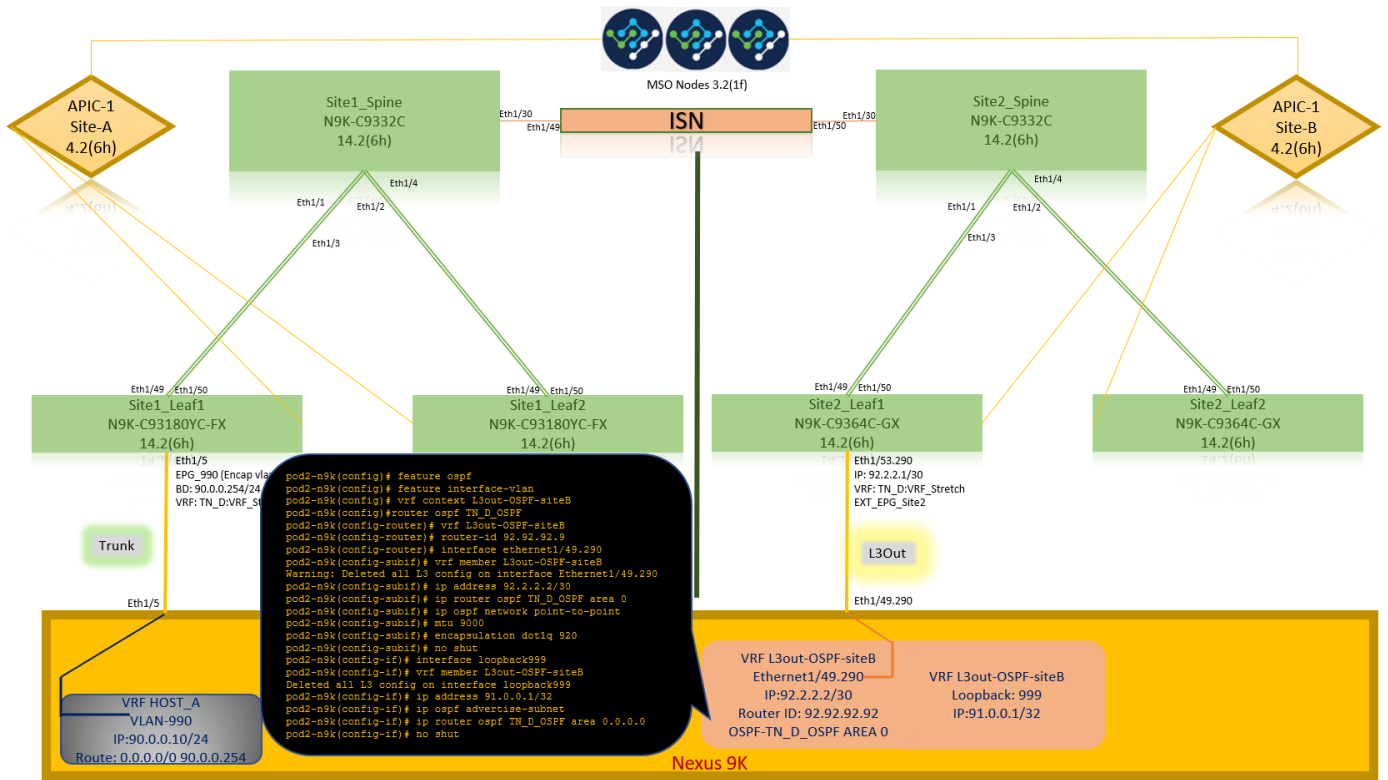
步骤8.检验在TN_D > Networking > L3Outs > L3Out-OSPF-siteB > Logical Interface Profiles > (接口配置文件) > OSPF Interface Profile下附加的OSPF接口配置文件策略。



步骤9.验证外部EPG“EXT_EPG_Site2”是否由MSO创建。从Site-B的APIC-1，选择TN_D > L3Outs > L3Out-OSPF-siteB > External EPGs > EXT_EPG_Site2。



配置外部N9K (站点B)



在N9K配置(VRF L3out-OSPF-siteB)后，我们可以看到N9K与ACI枝叶（位于站点B）之间已建立 OSPF邻居关系。

验证OSPF邻居关系已建立且UP（完全状态）。

从站点B的APIC-1，选择TN_D > Networking > L3Outs > L3Out-OSPF-siteB > Logical Node Profiles > Logical Interface Profiles > Configured Nodes > topology/pod01/node-1101 > OSPF for VRF_DVRF_Switch > Neighbor ID state > Full。

Neighbor Id	State	Peer Ip	Interface
92.92.92.92	Full	92.2.2.2	eth1/53.25

您还可以在N9K中检查OSPF邻居关系。此外，您还能ping通ACI枝叶IP（站点B）。

```

pod2-n9k(config-if)# ping 92.2.2.1 vrf L3out-OSPF-siteB
PING 92.2.2.1 (92.2.2.1): 56 data bytes
64 bytes from 92.2.2.1: icmp_seq=0 ttl=63 time=0.734 ms
64 bytes from 92.2.2.1: icmp_seq=1 ttl=63 time=0.591 ms
64 bytes from 92.2.2.1: icmp_seq=2 ttl=63 time=0.631 ms
64 bytes from 92.2.2.1: icmp_seq=3 ttl=63 time=0.588 ms
64 bytes from 92.2.2.1: icmp_seq=4 ttl=63 time=0.654 ms

--- 92.2.2.1 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 0.588/0.639/0.734 ms

```

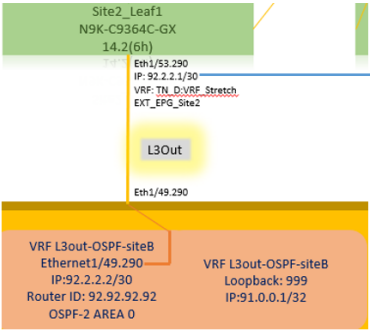
```

pod2-n9k(config-if)# show ip ospf neighbors vrf L3out-OSPF-siteB
OSPF Process ID TN_D_OSPF VRF L3out-OSPF-siteB
Total number of neighbors: 1
Neighbor ID Pri State Up Time Address Interface
110.110.110.110 1 FULL/ - 00:06:47 92.2.2.1 Eth1/49.290

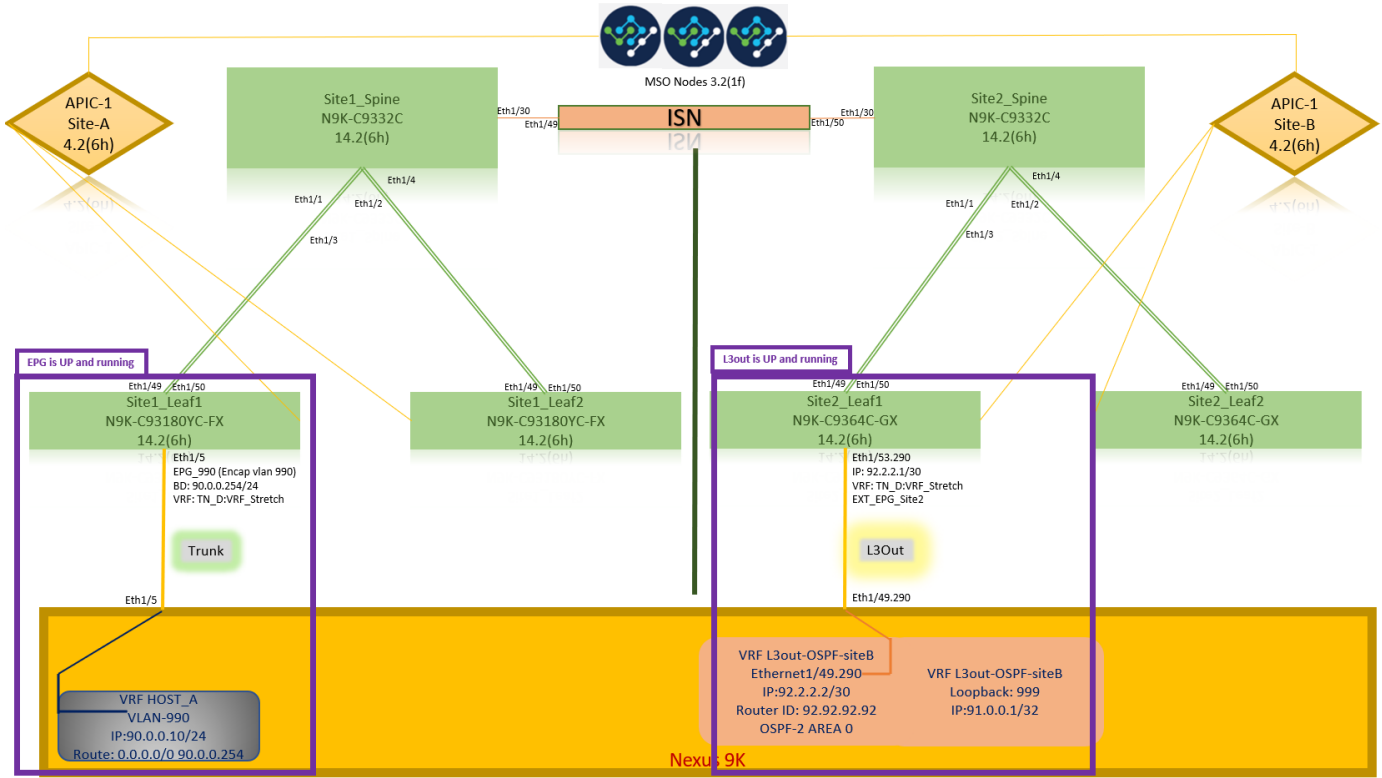
pod2-n9k(config-if)# show ip route vrf L3out-OSPF-siteB
IP Route Table for VRF "L3out-OSPF-siteB"
'*' denotes best ucast next-hop
'***' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

92.2.2.0/30, ubest/mbest: 1/0, attached
 *via 92.2.2.2, Eth1/49.290, [0/0], 00:19:38, direct
92.2.2.2/32, ubest/mbest: 1/0, attached
 *via 92.2.2.2, Eth1/49.290, [0/0], 00:19:38, local
110.110.110.110/32, ubest/mbest: 1/0
 *via 92.2.2.1, Eth1/49.290, [110/2], 00:06:48, ospf-TN_D_OSPF, intra

```



此时，站点A的Host_A配置和站点B的L3out配置已完成。

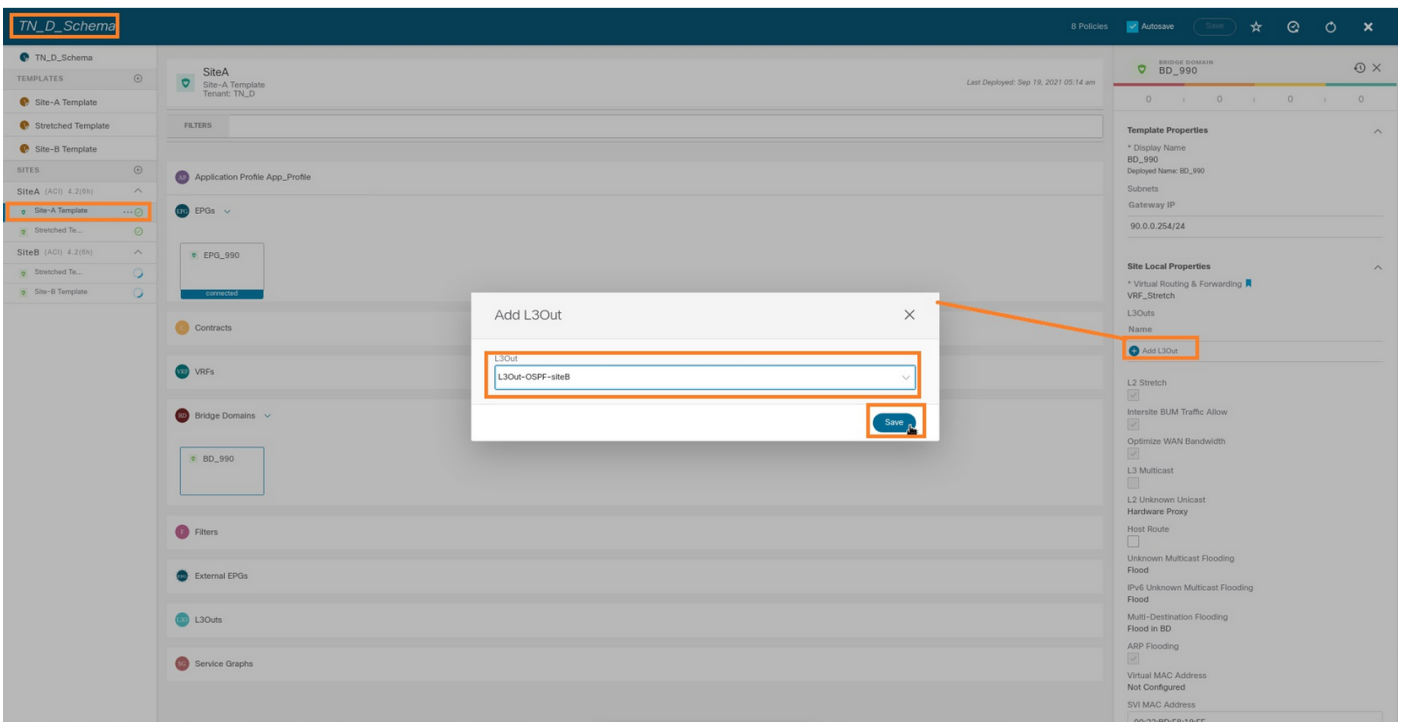
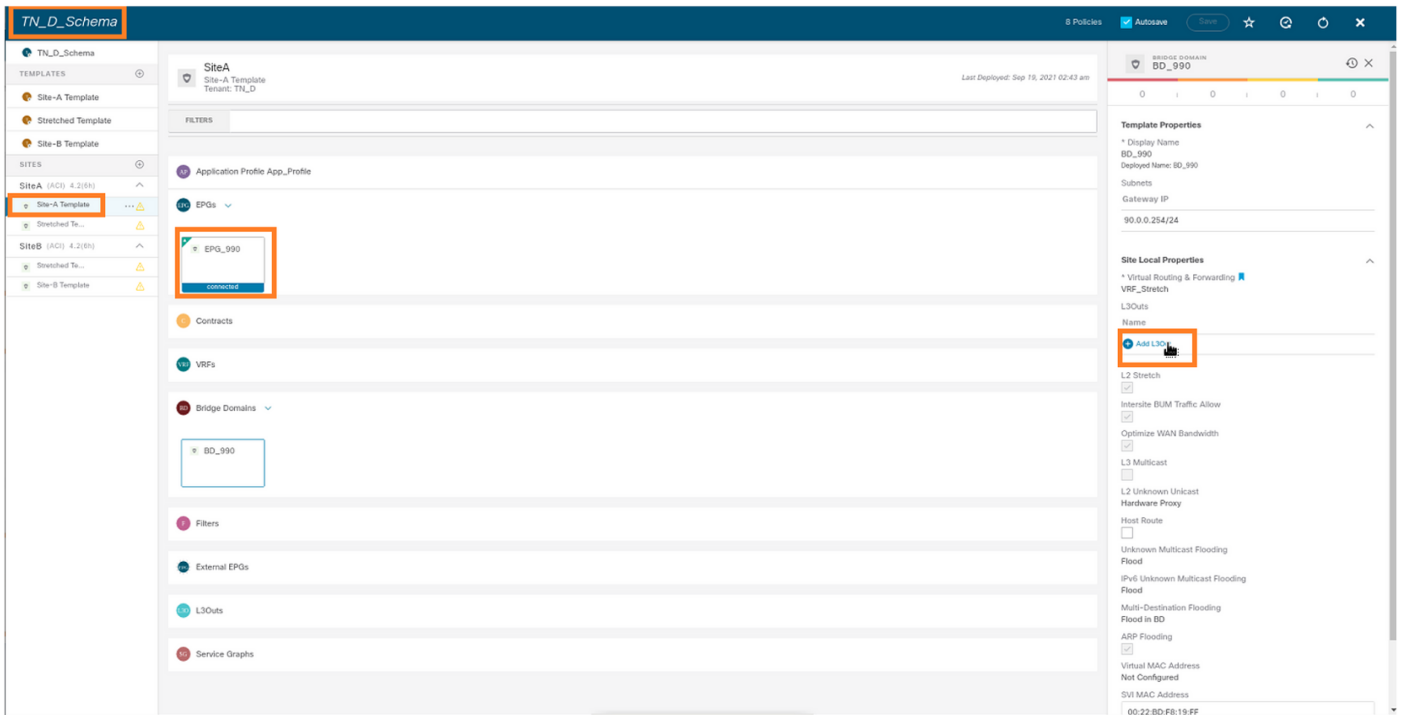


将站点B L3out连接到站点A EPG(BD)

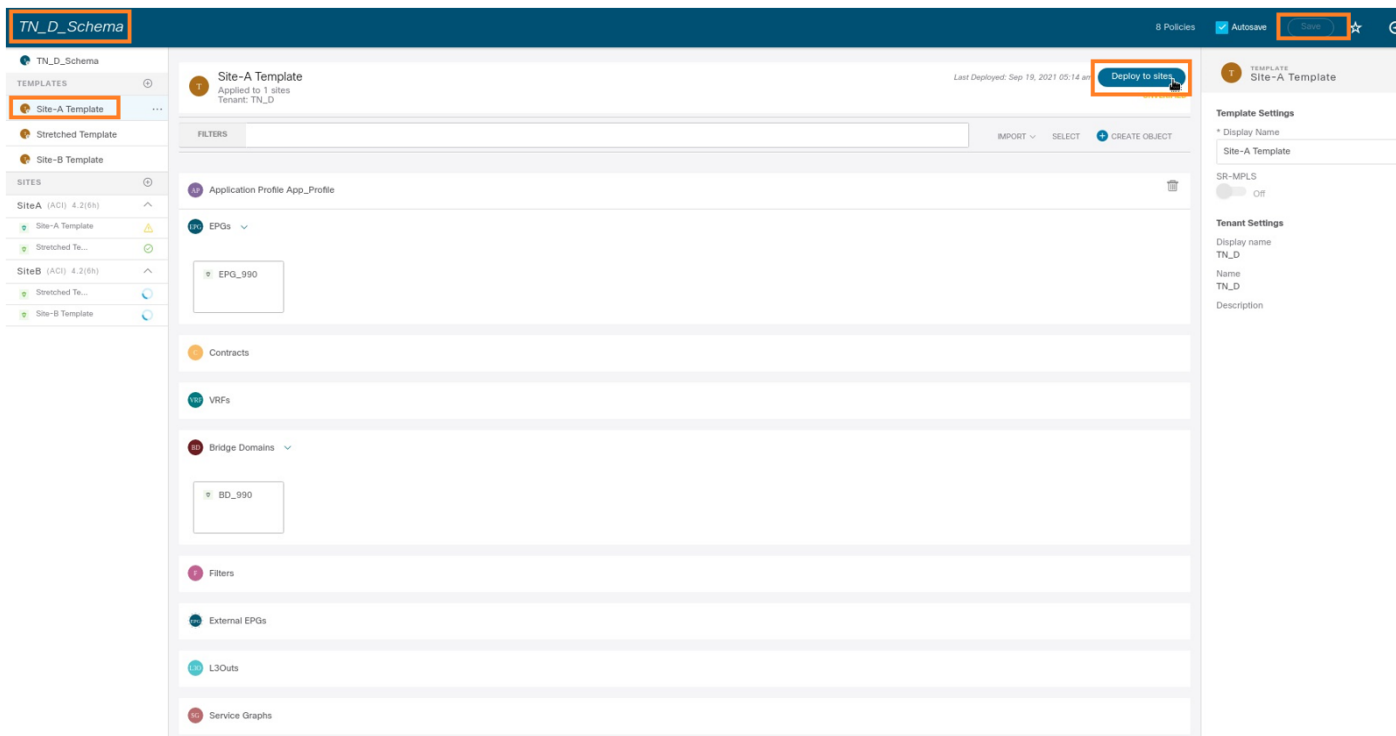
接下来，您可以从MSO将Site-B L3out连接到Site-A BD-990。请注意，左侧列有两个部分：1)模板和2)站点。

步骤1.在“站点”的第二部分，您可以看到每个站点附加的模板。将L3out附加到“Site-A Template”时，您基本上是从“Sites”部分内已附加的模板附加。

但是，在部署模板时，请从“模板”>“站点 — A模板”部分进行部署，然后选择保存/部署到站点。



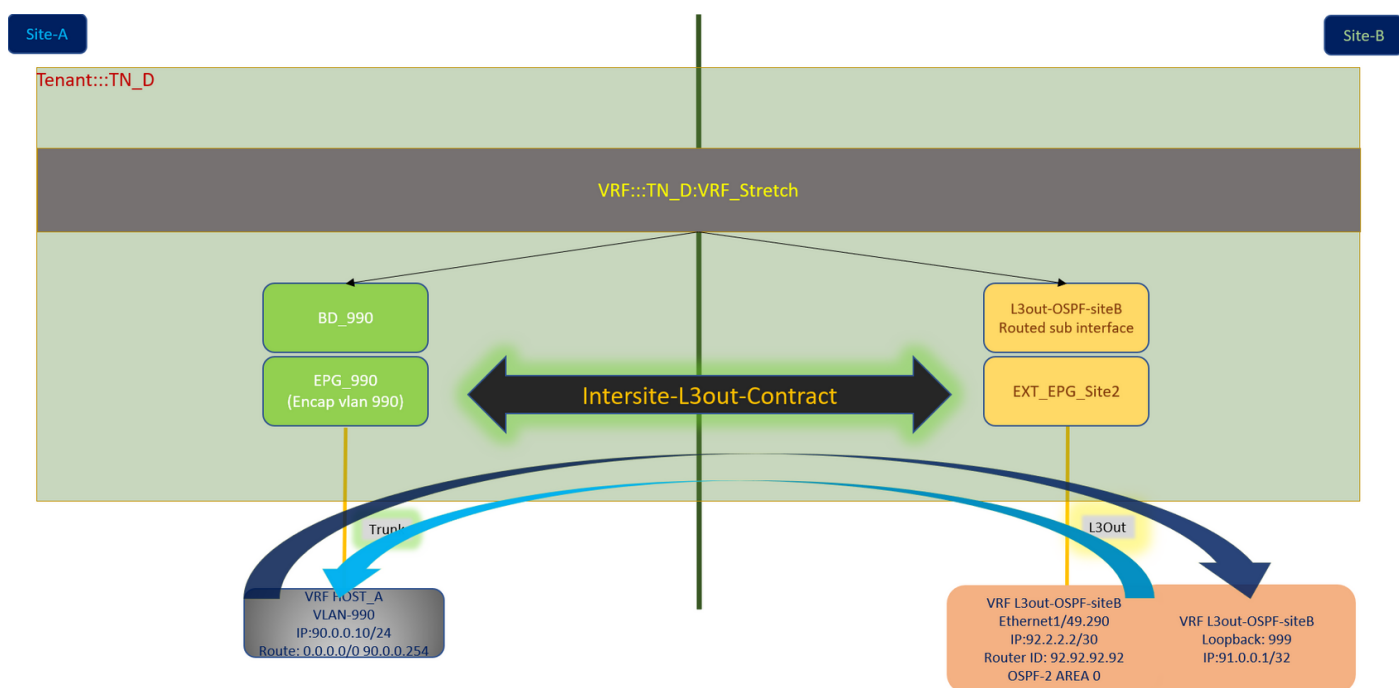
步骤2.从主模板“Site-A Template”（站点A模板）的第一部分“Templates”（模板）部署。



配置合同

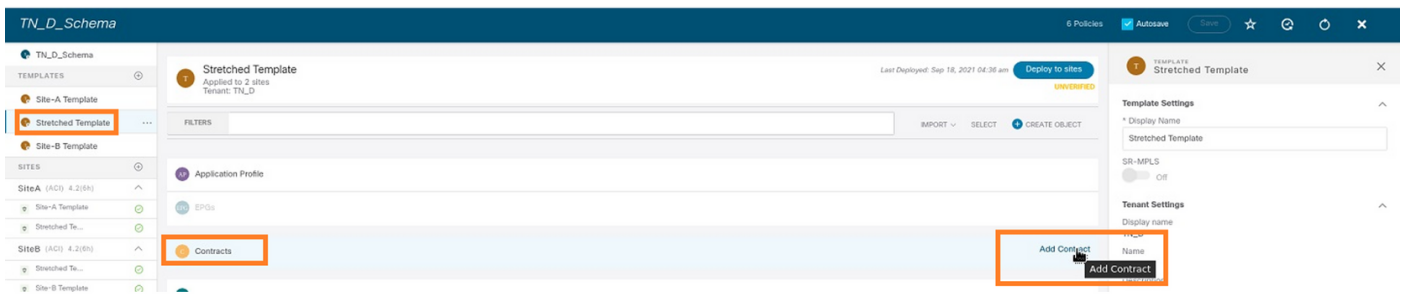
您需要在站点B的外部EPG和站点A的内部EPG_990之间签订合同。因此，您可以先从MSO创建合同，然后将其附加到两个EPG。

[思科以应用为中心的基础设施 — 思科ACI合同指南](#)可帮助理解合同。通常，内部EPG配置为提供商，而外部EPG配置为消费者。



创建合同

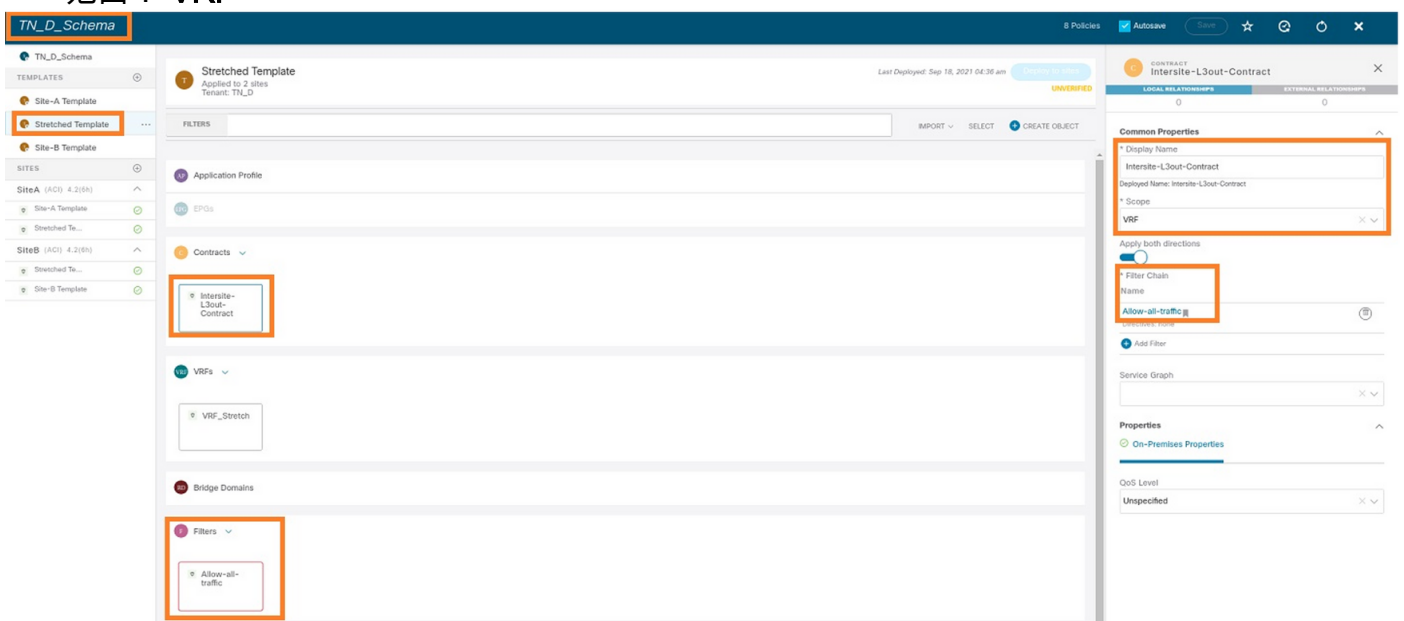
步骤1.从TN_D_Schema中，选择“拉伸模板”>“合同”。单击 添加合同。



步骤2.添加过滤器以允许所有流量。

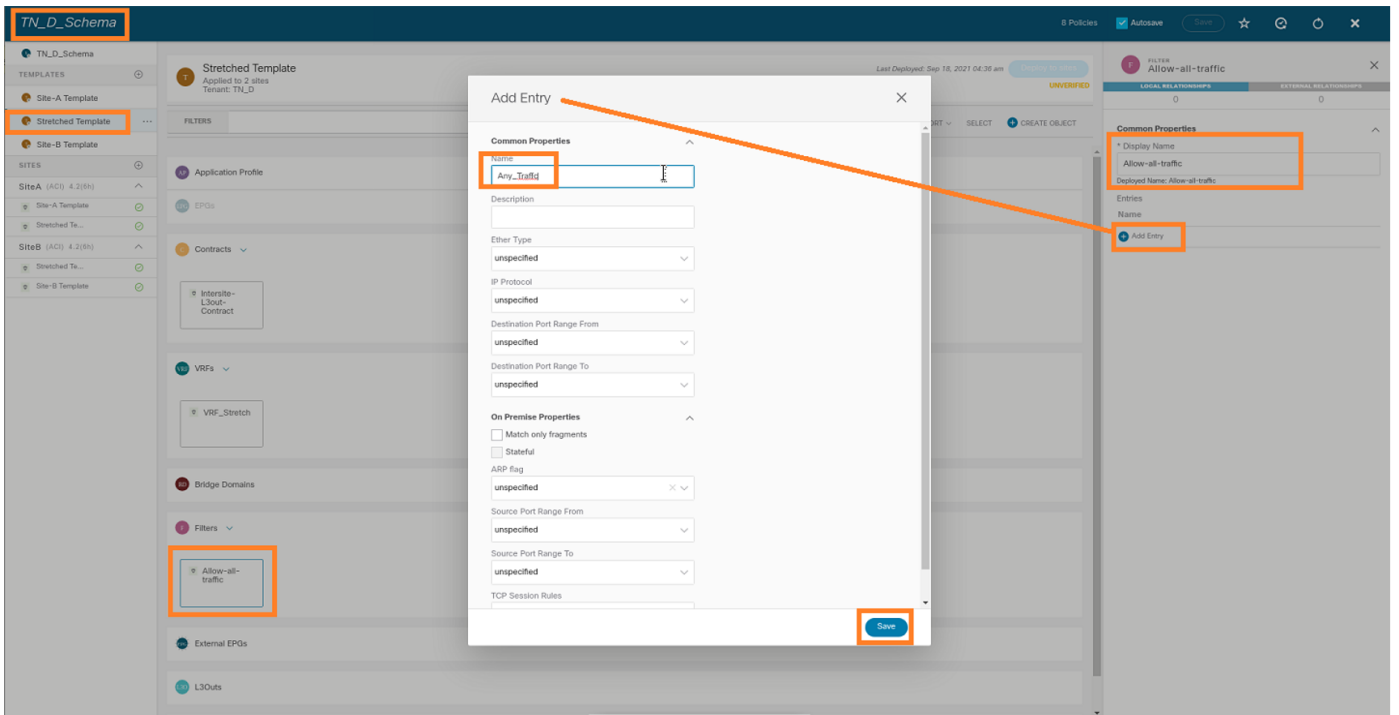
1. 从TN_D_Schema中，选择“延伸模板”>“合同”。
2. 添加合同时：应：

- 显示姓名：站点间L3输出合同
- 范围：VRF



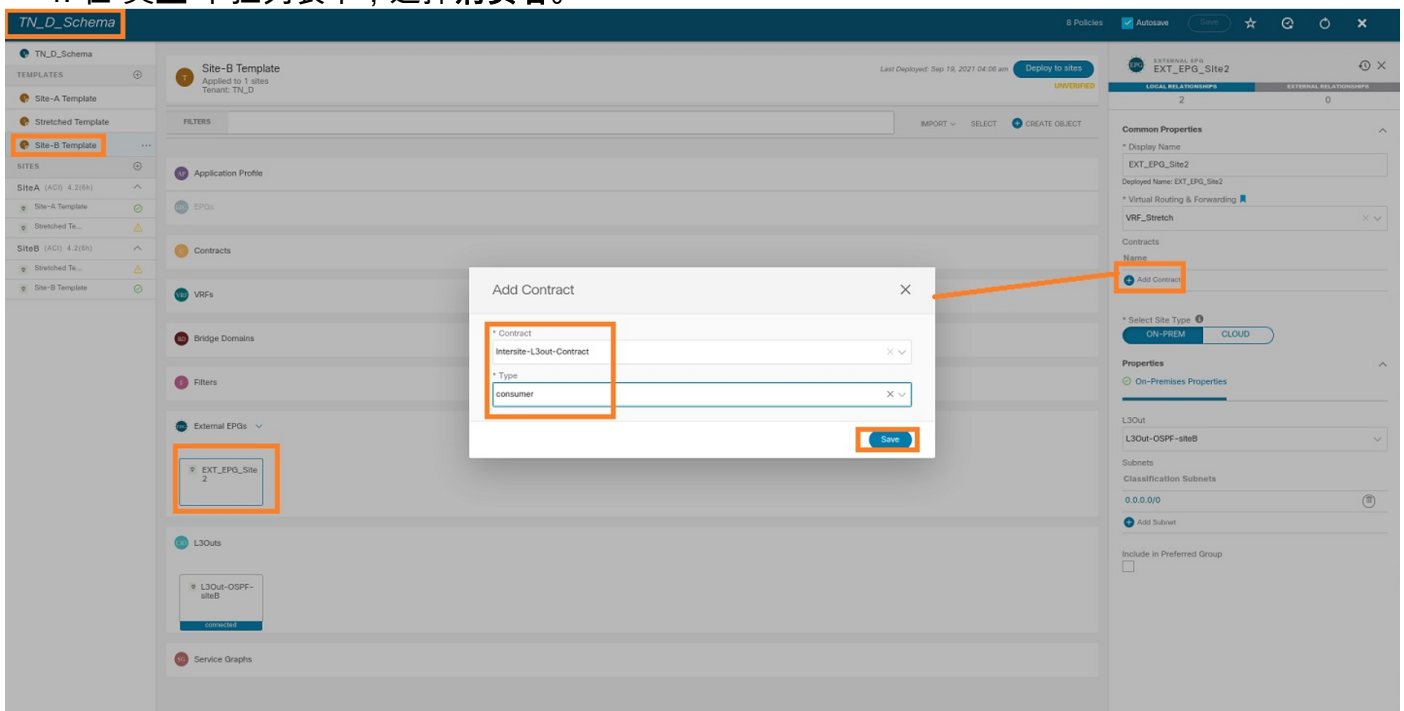
第三步：

1. 从TN_D_Schema中，选择“延伸模板”>“过滤器”。
2. 在“显示名称”字段中，输入“允许所有流量”。
3. 单击Add Entry。系统随即会显示“添加条目”对话框。
4. 在“名称”字段中，输入Any_Traffic。
5. 在Ether Type下拉列表中，选择未指定以允许所有流量。
6. Click Save.



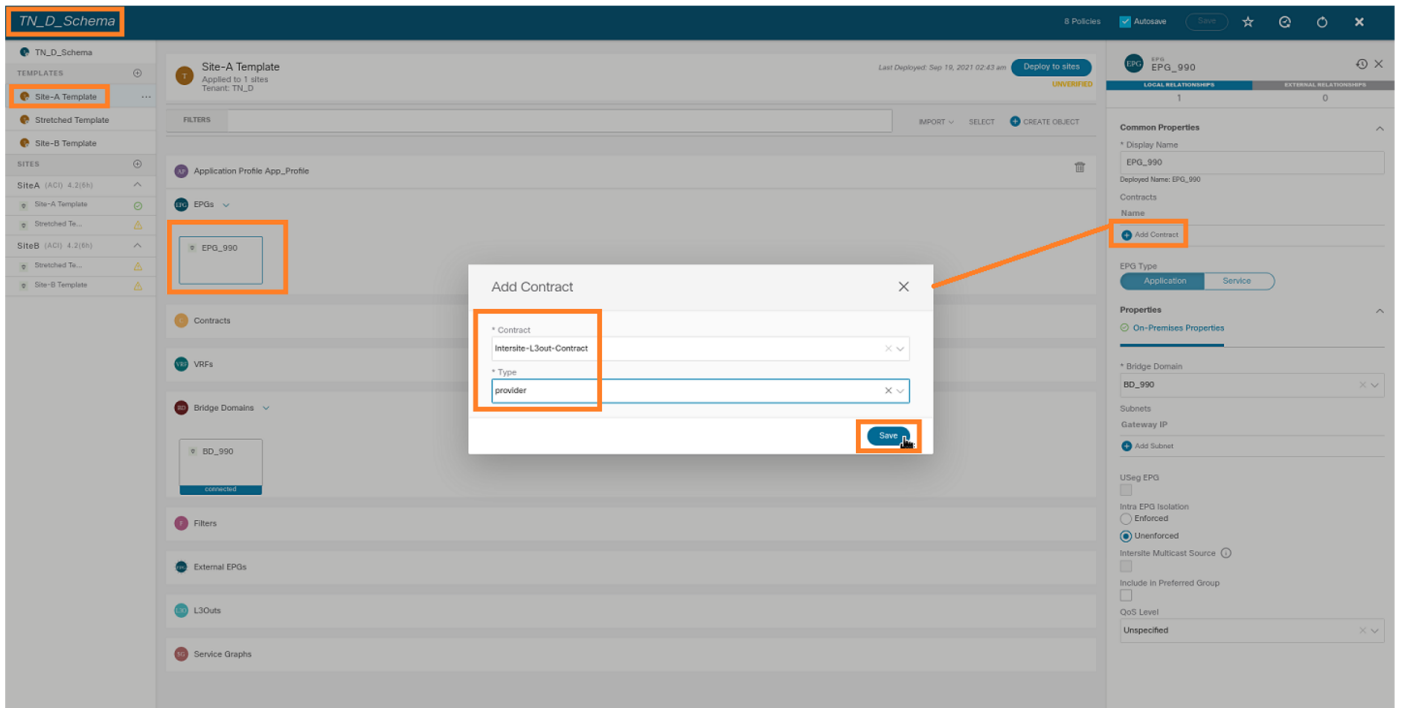
步骤4.将合同作为“消费者”（在站点B模板中）添加到外部EPG（部署到站点）。

1. 从TN_D_Schema中，选择Site-B Template > EXT_EPG_Site2。
2. 单击“添加合同”。系统随即会显示“添加合同”对话框。
3. 在“合同”字段中，输入Intersite-L3out-Contract。
4. 在“类型”下拉列表中，选择消费者。



步骤5.将合同作为“提供商”（在Site-A模板中）添加到内部EPG“EPG_990”（部署到站点）。

1. 从TN_D_Schema中，选择Site-A Template > EPG_990。
2. 单击“添加合同”。系统随即会显示“添加合同”对话框。
3. 在“Contract”(合同)字段中，输入Intersite-L3out-Contract。
4. 在“类型”下拉列表中，选择“提供程序”。



一旦添加合同，您就会看到在站点A创建的“影子L3out /外部EPG”。



APIC (SiteA)

System

Tenants

Fabric

Virtual Networking

L4-L7

ALL TENANTS

Add Tenant

Tenant Search:

name or descr



This has been created from Multi-Site. It is recommended to or

TN_D



Quick Start

TN_D

Application Profiles

Networking

Bridge Domains

VRFs

External Bridged Networks

L3Outs

L3Out-OSPF-siteB

Shadow L3out site-B

Logical Node Profiles

External EPGs

EXT_EPG_Site2

Shadow Ext EPG

Route map for import and export route control

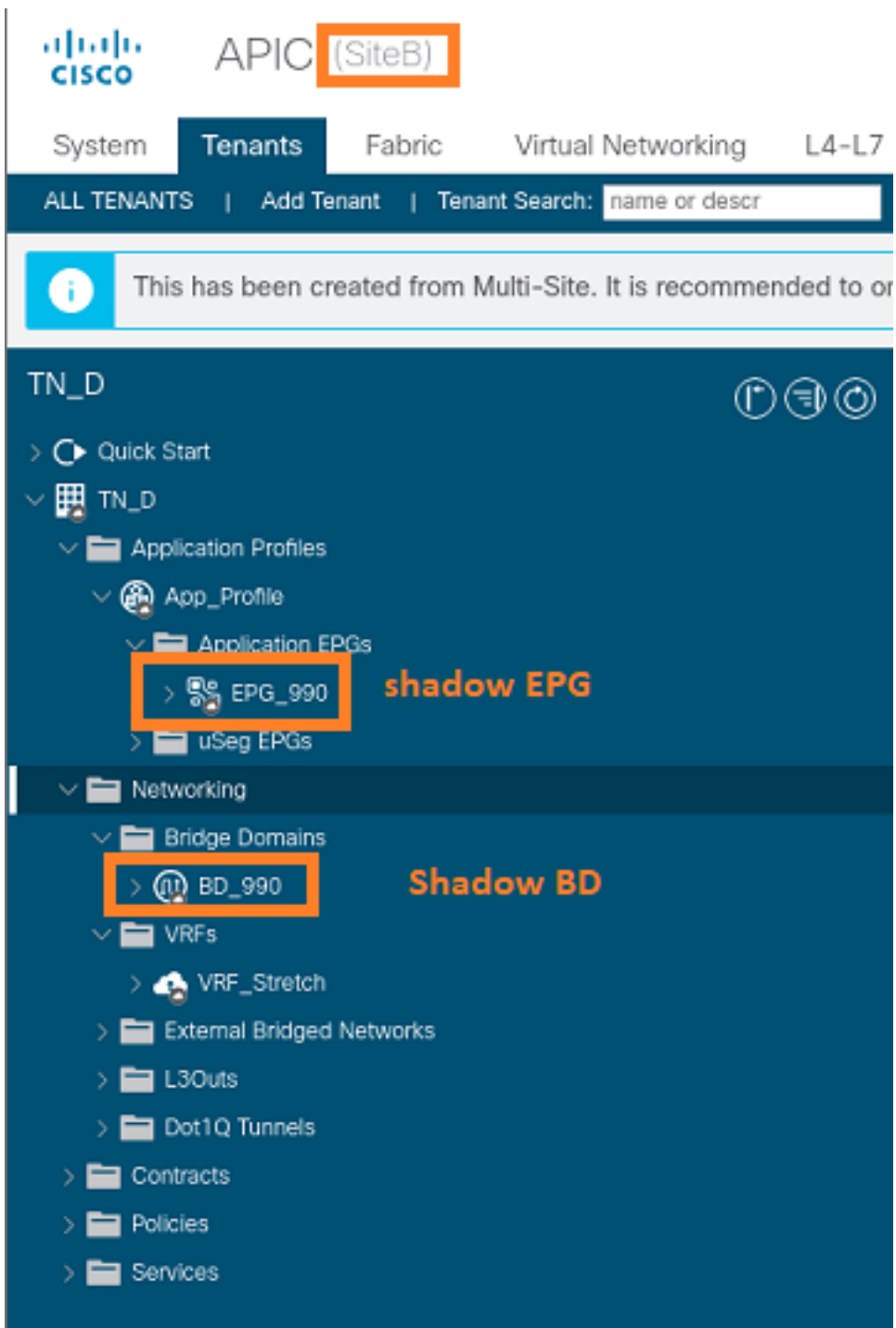
Dot1Q Tunnels

Contracts

Policies

Services

您还可以看到，“Shadow EPG_990和BD_990”也是在站点B上创建的。



步骤6.输入这些命令以检验Site-B APIC。

```
apic1# moquery -c fvAEPg -f 'fv.AEPg.name=="EPG_990"'
Total Objects shown: 1
# fv.AEPg
name                : EPG_990
annotation          : orchestrator:misc
childAction         :
configIssues        :
configSt            : applied
descr               :
dn                  : uni/tn-TN_D/ap-App_Profile/epg-EPG_990
exceptionTag        :
extMngdBy           :
floodOnEncap        : disabled
fwdCtrl             :
```

```

hasMcastSource      : no
isAttrBasedEPg     : no
isSharedSrvMsiteEPg : no
lcOwn               : local
matchT              : AtleastOne
modTs               : 2021-09-19T18:47:53.374+00:00
monPolDn            : uni/tn-common/monepg-default
nameAlias           :
pcEnfPref           : unenforced
pcTag              : 49153          <<< Note that pcTag is different for shadow EPG.
prefGrMemb          : exclude
prio                : unspecified
rn                  : epg-EPG_990
scope               : 2686978
shutdown            : no
status              :
triggerSt           : triggerable
txId                : 1152921504609244629
uid                 : 0

```

```

apic1# moquery -c fvBD -f 'fv.BD.name=="BD_990\"'

```

```

Total Objects shown: 1

```

```

# fv.BD

```

```

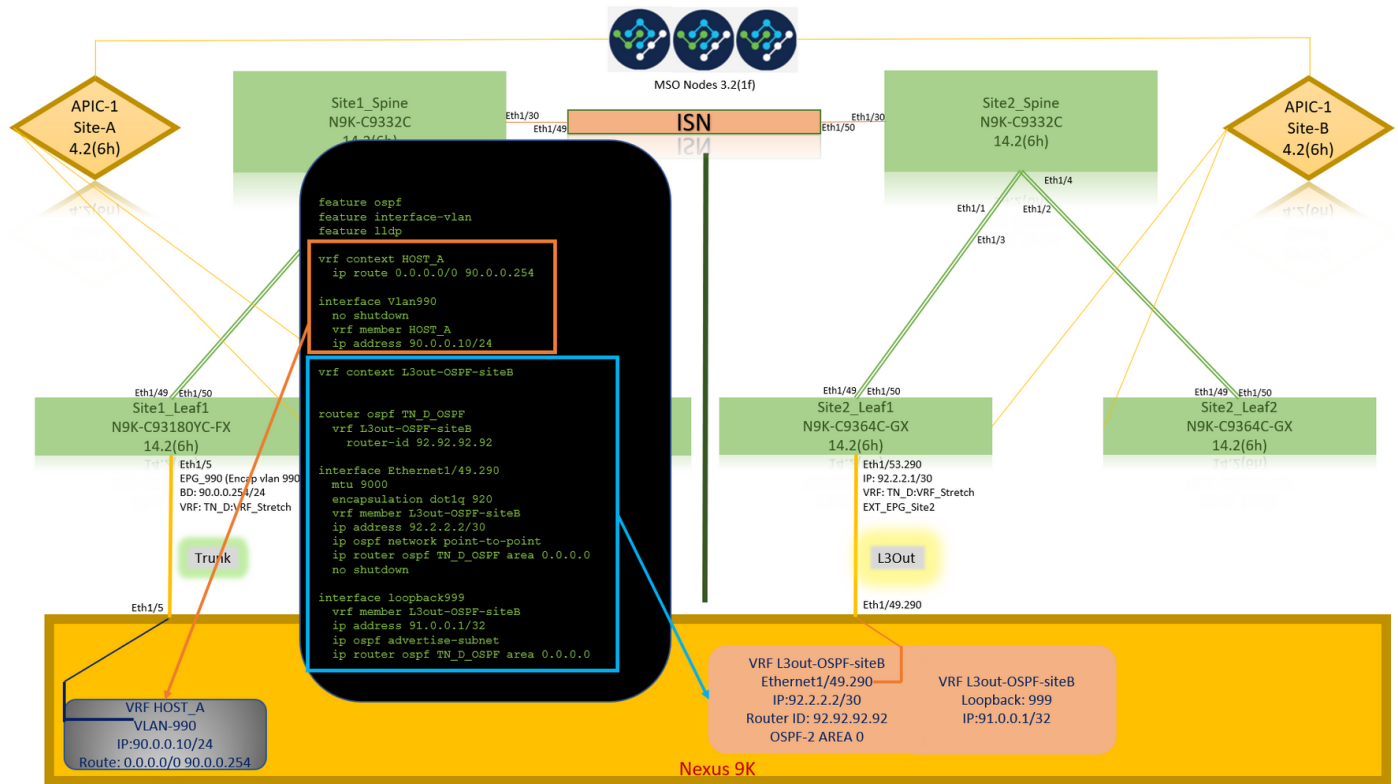
name              : BD_990
OptimizeWanBandwidth : yes
annotation            : orchestrator:misc
arpFlood              : yes
bcastP                : 225.0.181.192
childAction           :
configIssues          :
descr                 :
dn                : uni/tn-TN_D/BD-BD_990
epClear               : no
epMoveDetectMode      :
extMngdBy             :
hostBasedRouting      : no
intersiteBumTrafficAllow : yes
intersiteL2Stretch    : yes
ipLearning             : yes
ipv6McastAllow        : no
lcOwn                 : local
limitIpLearnToSubnets : yes
llAddr                : ::
mac                   : 00:22:BD:F8:19:FF
mcastAllow            : no
modTs                 : 2021-09-19T18:47:53.374+00:00
monPolDn              : uni/tn-common/monepg-default
mtu                   : inherit
multiDstPktAct        : bd-flood
nameAlias             :
ownerKey              :
ownerTag              :
pcTag                 : 32771
rn                    : BD-BD_990
scope                 : 2686978
seg                   : 15957972
status                :
type                  : regular
uid                   : 0
unicastRoute          : yes
unkMacUcastAct    : proxy
unkMcastAct      : flood

```



```
v6unkMcastAct      : flood
vmac                : not-applicable
```

步骤7.检查并检验外部设备N9K配置。



验证

使用本部分可确认配置能否正常运行。

终端学习

验证Site-A终端已作为终端在Site1_Leaf1中获知。

```
Site1_Leaf1# show endpoint interface ethernet 1/5
```

Legend:

```
s - arp          H - vtep          V - vpc-attached    p - peer-aged
R - peer-attached-rl B - bounce      S - static          M - span
D - bounce-to-proxy O - peer-attached  a - local-aged     m - svc-mgr
L - local        E - shared-service
```

```
-----+-----+-----+-----+
----+
      VLAN/
Interface Domain          Encap      VLAN      MAC Address      MAC Info/
      IP Address      IP Info
-----+-----+-----+-----+
----+
18                vlan-990      c014.fe5e.1407 L
eth1/5
TN_D:VRF_Stretch  vlan-990      90.0.0.10 L                eth1/5
```

ETEP/RTEP验证

站点_A枝叶。

Site1_Leaf1# show ip interface brief vrf overlay-1

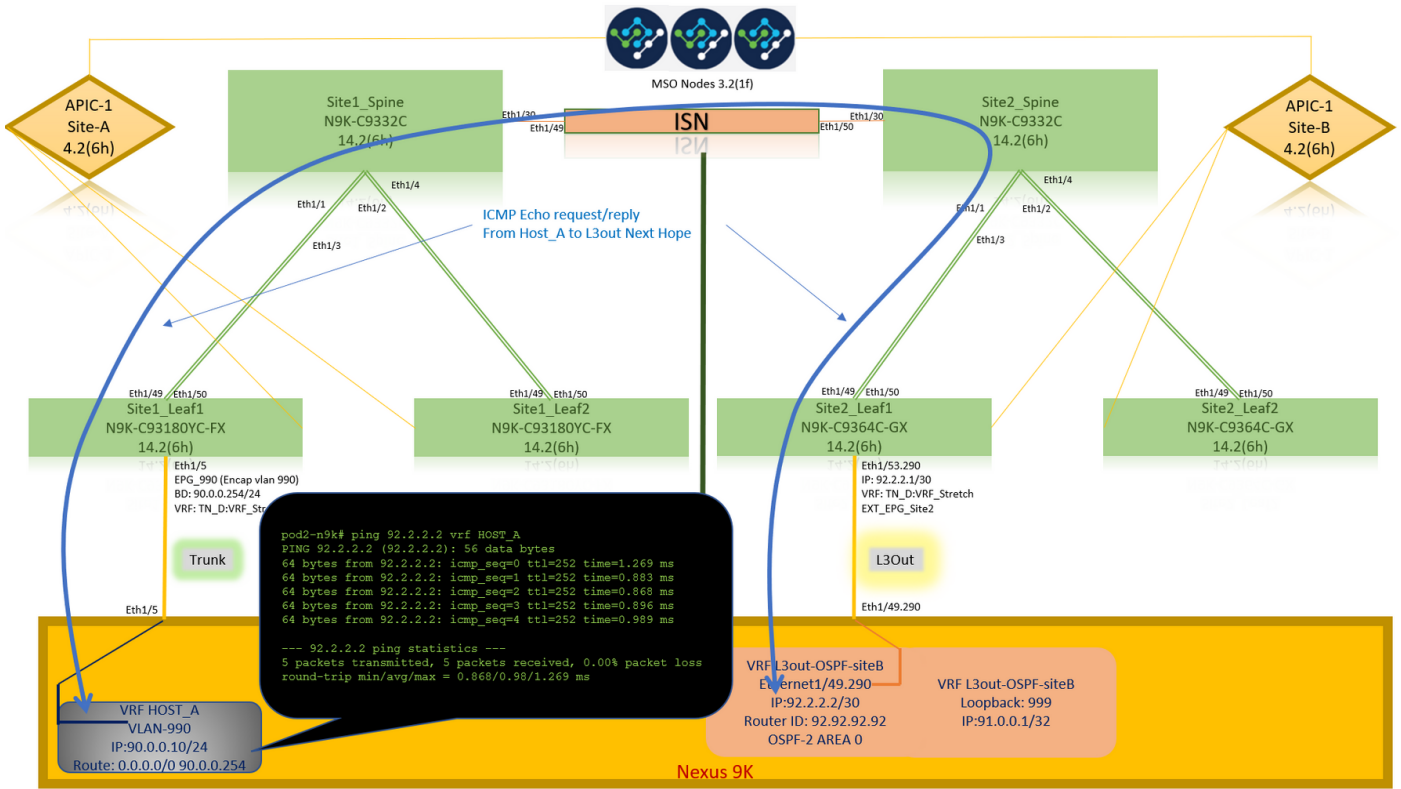
```
IP Interface Status for VRF "overlay-1" (4)
Interface                Address                Interface Status
eth1/49                  unassigned             protocol-up/link-up/admin-up
eth1/49.7                unnumbered             protocol-up/link-up/admin-up
                        (lo0)
eth1/50                  unassigned             protocol-up/link-up/admin-up
eth1/50.8                unnumbered             protocol-up/link-up/admin-up
                        (lo0)
eth1/51                  unassigned             protocol-down/link-down/admin-up
eth1/52                  unassigned             protocol-down/link-down/admin-up
eth1/53                  unassigned             protocol-down/link-down/admin-up
eth1/54                  unassigned             protocol-down/link-down/admin-up
vlan9                    10.0.0.30/27          protocol-up/link-up/admin-up
lo0                      10.0.80.64/32         protocol-up/link-up/admin-up
lo1                      10.0.8.67/32          protocol-up/link-up/admin-up
lo8                    192.168.200.225/32  protocol-up/link-up/admin-up <<<<< IP from ETEP site-
A
lo1023                  10.0.0.32/32          protocol-up/link-up/admin-up
```

Site2_Leaf1# show ip interface brief vrf overlay-1

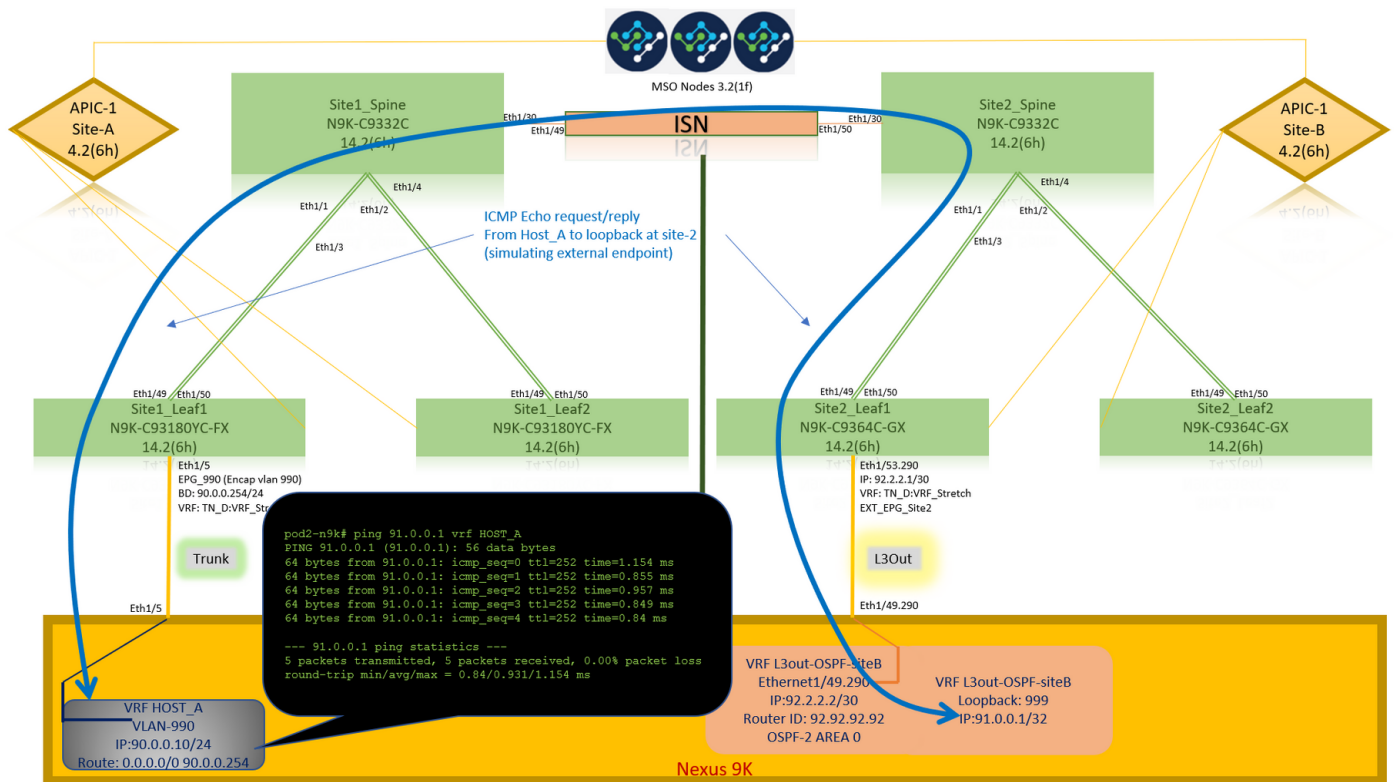
```
IP Interface Status for VRF "overlay-1" (4)
Interface                Address                Interface Status
eth1/49                  unassigned             protocol-up/link-up/admin-up
eth1/49.16              unnumbered             protocol-up/link-up/admin-up
                        (lo0)
eth1/50                  unassigned             protocol-up/link-up/admin-up
eth1/50.17              unnumbered             protocol-up/link-up/admin-up
                        (lo0)
eth1/51                  unassigned             protocol-down/link-down/admin-up
eth1/52                  unassigned             protocol-down/link-down/admin-up
eth1/54                  unassigned             protocol-down/link-down/admin-up
eth1/55                  unassigned             protocol-down/link-down/admin-up
eth1/56                  unassigned             protocol-down/link-down/admin-up
eth1/57                  unassigned             protocol-down/link-down/admin-up
eth1/58                  unassigned             protocol-down/link-down/admin-up
eth1/59                  unassigned             protocol-down/link-down/admin-up
eth1/60                  unassigned             protocol-down/link-down/admin-up
eth1/61                  unassigned             protocol-down/link-down/admin-up
eth1/62                  unassigned             protocol-down/link-down/admin-up
eth1/63                  unassigned             protocol-down/link-down/admin-up
eth1/64                  unassigned             protocol-down/link-down/admin-up
vlan18                  10.0.0.30/27          protocol-up/link-up/admin-up
lo0                      10.0.72.64/32         protocol-up/link-up/admin-up
lo1                      10.0.80.67/32          protocol-up/link-up/admin-up
lo6                    192.168.100.225/32  protocol-up/link-up/admin-up <<<<< IP from ETEP site-B
lo1023                  10.0.0.32/32          protocol-up/link-up/admin-up
```

ICMP可达性

从HOST_A对外部设备的WAN IP地址执行ping操作。

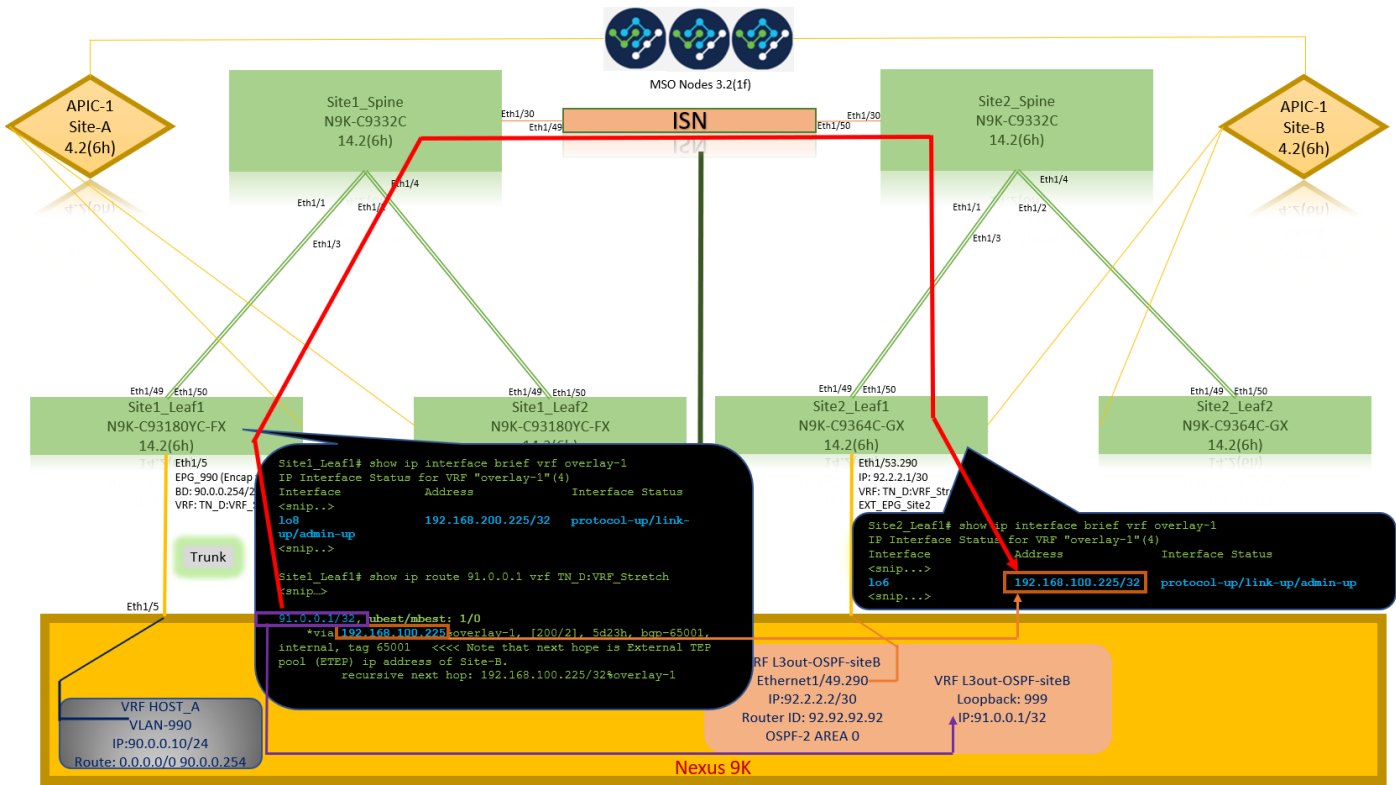


Ping外部设备环回地址。



路由验证

检验外部设备的WAN IP地址或环回子网路由是否存在于路由表中。当您在“Site1_Leaf1”中检查外部设备子网的下一跳时，它是枝叶“Site2-Leaf1”的外部TEP IP。



```

Site1_Leaf1# show ip route 92.2.2.2 vrf TN_D:VRF_Stretch
IP Route Table for VRF "TN_D:VRF_Stretch"
'*' denotes best ucast next-hop
*** denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
 '%' in via output denotes VRF
92.2.2.0/30, ubest/mbest: 1/0
  *via 192.168.100.225%overlay-1, [200/0], 5d23h, bgp-65001, internal, tag 65001 <<<< Note
that next hope is External TEP pool (ETEP) ip address of Site-B.
    recursive next hop: 192.168.100.225/32%overlay-1
Site1_Leaf1# show ip route 91.0.0.1 vrf TN_D:VRF_Stretch
IP Route Table for VRF "TN_D:VRF_Stretch"
'*' denotes best ucast next-hop
*** denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
 '%' in via output denotes VRF
91.0.0.1/32, ubest/mbest: 1/0
  *via 192.168.100.225%overlay-1, [200/2], 5d23h, bgp-65001, internal, tag 65001 <<<< Note
that next hope is External TEP pool (ETEP) ip address of Site-B.
    recursive next hop: 192.168.100.225/32%overlay-1
  
```

故障排除

本部分提供的信息可用于对配置进行故障排除。

站点2_枝叶1

BGP地址系列路由在TN_D:VRF_stretch和Overlay-1之间导入/导出。

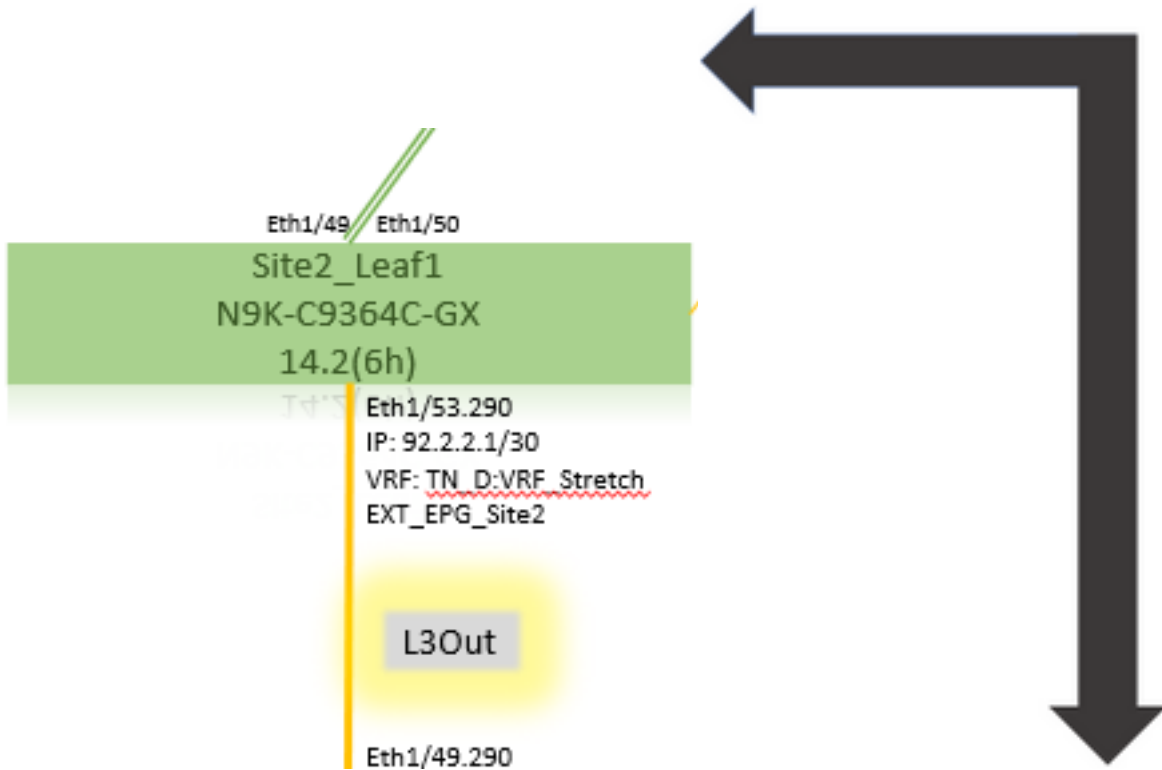
```

Site2_Leaf1# show system internal epn vrf TN_D:VRF_Stretch
+-----+-----+-----+-----+-----+-----+
  
```

VRF	Type	VRF vnid	Context ID	Status	Endpoint Count
TN_D:VRF_Stretch	Tenant	2686978	46	Up	1

Site2_Leaf1# show vrf TN_D:VRF_Stretch detail

VRF-Name: TN_D:VRF_Stretch, VRF-ID: 46, State: Up
 VPNID: unknown
RD: 1101:2686978
 Max Routes: 0 Mid-Threshold: 0
 Table-ID: 0x8000002e, AF: IPv6, Fwd-ID: 0x8000002e, State: Up
 Table-ID: 0x0000002e, AF: IPv4, Fwd-ID: 0x0000002e, State: Up



Site2_Leaf1# vsh

Site2_Leaf1# show bgp vpnv4 unicast 91.0.0.1 vrf TN_D:VRF_Stretch

BGP routing table information for VRF overlay-1, address family VPNv4 Unicast
Route Distinguisher: 1101:2686978 (VRF TN_D:VRF_Stretch)
BGP routing table entry for 91.0.0.1/32, version 12 dest ptr 0xae6da350
 Paths: (1 available, best #1)
 Flags: (0x80c0002 00000000) on xmit-list, is not in urib, exported
 vpn: version 346, (0x100002) on xmit-list
 Multipath: eBGP iBGP

Advertised path-id 1, VPN AF advertised path-id 1
 Path type: redistrib 0x408 0x1 ref 0 adv path ref 2, path is valid, is best path
 AS-Path: NONE, path locally originated
0.0.0.0 (metric 0) from 0.0.0.0 (10.0.72.64)
 Origin incomplete, MED 2, localpref 100, weight 32768
 Extcommunity:
RT:65001:2686978
VNID:2686978
 COST:pre-bestpath:162:110
 VRF advertise information:
 Path-id 1 not advertised to any peer
 VPN AF advertise information:
 Path-id 1 advertised to peers:
10.0.72.65

```

apic1# acidiag fnvread ID Pod ID Name Serial Number IP Address Role State LastUpdMsgId -----
-----
----- 101 1

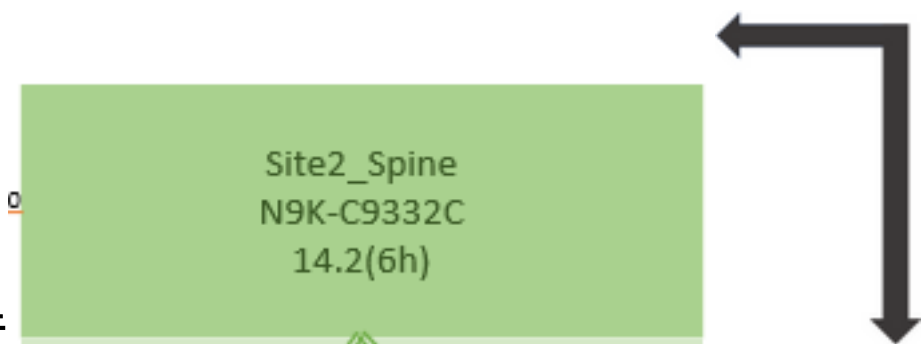
```

Site2_Spine FDO243207JH

```

10.0.72.65/32 spine active 0 102 1 Site2_Leaf2 FDO24260FCH 10.0.72.66/32 leaf active 0 1101
1 Site2_Leaf1 FDO24260ECW 10.0.72.64/32 leaf active 0

```

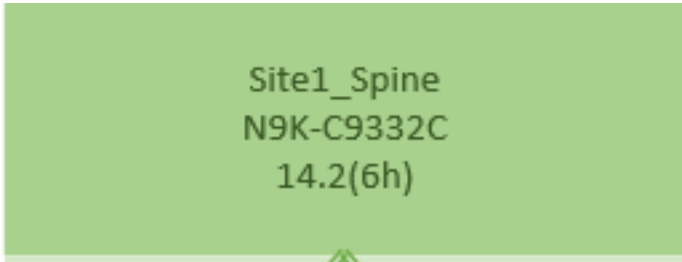


站点2_主干

```

Site2_Spine# vsh
Site2_Spine# show bgp vpnv4 unicast 91.0.0.1 vrf overlay-1
BGP routing table information for VRF overlay-1, address family VPNv4 Unicast
<-----26bits----->
Route Distinguisher: 1101:2686978 <<<<<2686978 <--
Binary--> 000010100100000000000000010
BGP routing table entry for 91.0.0.1/32, version 717 dest ptr 0xae643d0c
Paths: (1 available, best #1)
Flags: (0x000002 00000000) on xmit-list, is not in urib, is not in HW
Multipath: eBGP iBGP
  Advertised path-id 1
  Path type: internal 0x40000018 0x800040 ref 0 adv path ref 1, path is valid, is best path
  AS-Path: NONE, path sourced internal to AS
  10.0.72.64 (metric 2) from 10.0.72.64 (10.0.72.64) <<< Site2_leaf1 IP
  Origin incomplete, MED 2, localpref 100, weight 0
  Received label 0
  Received path-id 1
  Extcommunity:
    RT:65001:2686978
    COST:pre-bestpath:168:3221225472
    VNID:2686978
    COST:pre-bestpath:162:110
  Path-id 1 advertised to peers:
    192.168.10.13 <<<< Site1_Spine mscp-etest IP.
Site1_Spine# show ip interface vrf overlay-1
<snip...>
lo12, Interface status: protocol-up/link-up/admin-up, iod: 89, mode: mscp-etest
  IP address: 192.168.10.13, IP subnet: 192.168.10.13/32 <<

```



站点1_主干

```

Site1_Spine# vsh
Site1_Spine# show bgp vpnv4 unicast 91.0.0.1 vrf overlay-1
BGP routing table information for VRF overlay-1, address family VPNv4 Unicast
<-----26Bits----->
Route Distinguisher: 1101:36241410
<<<<<36241410<<<<--binary-->100010100100000000000000010
BGP routing table entry for 91.0.0.1/32, version 533 dest ptr 0xae643dd4
Paths: (1 available, best #1)
Flags: (0x000002 00000000) on xmit-list, is not in urib, is not in HW
Multipath: eBGP iBGP
  Advertised path-id 1
    Path type: internal 0x40000018 0x880000 ref 0 adv path ref 1, path is valid, is best path,
remote site path
  AS-Path: NONE, path sourced internal to AS
    192.168.100.225 (metric 20) from 192.168.11.13 (192.168.11.13) <<< Site2_Leaf1 ETEP IP
learn via Site2_Spine mscsp-etest address.
  Origin incomplete, MED 2, localpref 100, weight 0
  Received label 0
  Extcommunity:
    RT:65001:36241410
    SOO:65001:50331631
    COST:pre-bestpath:166:2684354560
    COST:pre-bestpath:168:3221225472
    VNID:2686978
    COST:pre-bestpath:162:110
  Originator: 10.0.72.64 Cluster list: 192.168.11.13 <<< Originator Site2_Leaf1 and
Site2_Spine ips are listed here...
  Path-id 1 advertised to peers:
    10.0.80.64 <<<< Site1_Leaf1 ip

```

```

Site2_Spine# show ip interface vrf overlay-1
<snip..>
lo13, Interface status: protocol-up/link-up/admin-up, iod: 92, mode: mscsp-etest IP address:
192.168.11.13, IP subnet: 192.168.11.13/32
  IP broadcast address: 255.255.255.255
  IP primary address route-preference: 0, tag: 0
<snip..>

```

```

Site-B apic1# acidiag fvnread

```

ID	Pod ID	Name	Serial Number	IP Address	Role	State
101	1	Site2_Spine	FDO243207JH	10.0.72.65/32	spine	active 0
102	1	Site2_Leaf2	FDO24260FCH	10.0.72.66/32	leaf	active 0
1101	1	Site2_Leaf1	FDO24260ECW	10.0.72.64/32	leaf	active 0

验证站点间标志。

```

Site1_Spine# moquery -c bgpPeer -f 'bgp.Peer.addr*"192.168.11.13"'

```

```

Total Objects shown: 1
# bgp.Peer
addr           : 192.168.11.13/32
activePfxPeers : 0
adminSt        : enabled
asn            : 65001
bgpCfgFailedBmp :
bgpCfgFailedTs : 00:00:00:00.000
bgpCfgState    : 0
childAction    :
ctrl           :
curPfxPeers    : 0
dn             : sys/bgp/inst/dom-overlay-1/peer-[192.168.11.13/32]
lcOwn          : local
maxCurPeers   : 0
maxPfxPeers    : 0
modTs          : 2021-09-13T11:58:26.395+00:00
monPolDn       :
name           :
passwdSet      : disabled
password       :
peerRole       : msite-speaker
privateASctrl  :
rn             : peer-[192.168.11.13/32] <<

```

<<

了解路由区分器条目设置站点间标志后，本地站点主干可以在从第25位开始的路由目标中设置本地站点ID。当站点1在RT中设置此位时获取BGP路径，它知道这是远程站点路径。

```

Site2_Leaf1# vsh
Site2_Leaf1# show bgp vpnv4 unicast 91.0.0.1 vrf TN_D:VRF_Stretch
BGP routing table information for VRF overlay-1, address family VPNv4 Unicast
<-----26Bits----->
Route Distinguisher: 1101:2686978      (VRF TN_D:VRF_Stretch)          <<<<<2686978
<--Binary--> 00001010010000000000000010
BGP routing table entry for 91.0.0.1/32, version 12 dest ptr 0xae6da350

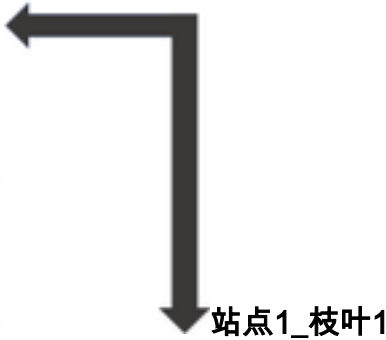
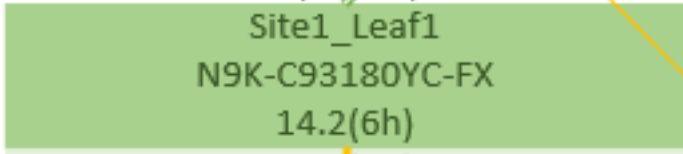
Site1_Spine# vsh
Site1_Spine# show bgp vpnv4 unicast 91.0.0.1 vrf overlay-1

<-----26Bits----->
Route Distinguisher: 1101:36241410
<<<<<36241410<--binary-->10001010010000000000000010

^^---26th bit set to 1 and with 25th bit value it become 10.

```

请注意，除第26位设置为1外，Site1的RT二进制值完全相同。它有十进制值（标记为蓝色）。1101:36241410是您预期在Site1中看到的内容，以及必须导入Site1的内部枝叶。



```
Site1_Leaf1# show vrf TN_D:VRF_Stretch detail
VRF-Name: TN_D:VRF_Stretch, VRF-ID: 46, State: Up
VPNID: unknown
RD: 1101:2850817
Max Routes: 0 Mid-Threshold: 0
Table-ID: 0x8000002e, AF: IPv6, Fwd-ID: 0x8000002e, State: Up
Table-ID: 0x0000002e, AF: IPv4, Fwd-ID: 0x0000002e, State: Up
```

```
Site1_Leaf1# show bgp vpnv4 unicast 91.0.0.1 vrf overlay-1
BGP routing table information for VRF overlay-1, address family VPNv4 Unicast
Route Distinguisher: 1101:2850817 (VRF TN_D:VRF_Stretch)
BGP routing table entry for 91.0.0.1/32, version 17 dest ptr 0xadeda550
Paths: (1 available, best #1)
Flags: (0x08001a 00000000) on xmit-list, is in urib, is best urib route, is in HW
      vpn: version 357, (0x100002) on xmit-list
Multipath: eBGP iBGP
  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: internal 0xc0000018 0x80040 ref 56506 adv path ref 2, path is valid, is best path,
remote site path
      Imported from 1101:36241410:91.0.0.1/32
AS-Path: NONE, path sourced internal to AS
  192.168.100.225 (metric 64) from 10.0.80.65 (192.168.10.13)
  Origin incomplete, MED 2, localpref 100, weight 0
  Received label 0
  Received path-id 1
  Extcommunity:
    RT:65001:36241410
    SOO:65001:50331631
    COST:pre-bestpath:166:2684354560
    COST:pre-bestpath:168:3221225472
    VNID:2686978
    COST:pre-bestpath:162:110
  Originator: 10.0.72.64 Cluster list: 192.168.10.13192.168.11.13 <<<<
'10.0.72.64'='Site2_Leaf1' , '192.168.10.13'='Site1_Spine' , '192.168.11.13'='Site2_Spine'
  VRF advertise information:
  Path-id 1 not advertised to any peer
  VPN AF advertise information:
  Path-id 1 not advertised to any peer
```

```
<snip..>
Site1_Leaf1# show bgp vpnv4 unicast 91.0.0.1 vrf TN_D:VRF_Stretch
BGP routing table information for VRF overlay-1, address family VPNv4 Unicast
Route Distinguisher: 1101:2850817 (VRF TN_D:VRF_Stretch)
BGP routing table entry for 91.0.0.1/32, version 17 dest ptr 0xadeda550
Paths: (1 available, best #1)
Flags: (0x08001a 00000000) on xmit-list, is in urib, is best urib route, is in HW
      vpn: version 357, (0x100002) on xmit-listMultipath: eBGP iBGP
  Advertised path-id 1, VPN AF advertised path-id 1
  Path type: internal 0xc0000018 0x80040 ref 56506 adv path ref 2, path is valid, is best path,
remote site path
      Imported from 1101:36241410:91.0.0.1/32
AS-Path: NONE, path sourced internal to AS
  192.168.100.225 (metric 64) from 10.0.80.65 (192.168.10.13)
  Origin incomplete, MED 2, localpref 100, weight 0
```

```

Received label 0
Received path-id 1
Extcommunity:
  RT:65001:36241410
  SOO:65001:50331631
  COST:pre-bestpath:166:2684354560
  COST:pre-bestpath:168:3221225472
  VNID:2686978
  COST:pre-bestpath:162:110
Originator: 10.0.72.64 Cluster list: 192.168.10.13 192.168.11.13
VRF advertise information:
Path-id 1 not advertised to any peer
VPN AF advertise information:
Path-id 1 not advertised to any peer

```

因此，“Site1_Leaf1”具有子网91.0.0.1/32的路由条目，下一跳为“Site2_Leaf1”ETEP地址192.168.100.225。

```

Site1_Leaf1# show ip route 91.0.0.1 vrf TN_D:VRF_Stretch
IP Route Table for VRF "TN_D:VRF_Stretch"
 '*' denotes best ucast next-hop
 '**' denotes best mcast next-hop
 '[x/y]' denotes [preference/metric]
 '%' in via output denotes VRF
91.0.0.1/32, ubest/mbest: 1/0
  *via 192.168.100.225%overlay-1, [200/2], 5d23h, bgp-65001, internal, tag 65001 <<<< Note
that next hope is External TEP pool (ETEP) ip address of Site-B.
  recursive next hop: 192.168.100.225/32%overlay-1

```

Site-A主干向“Site2_Spine” mcsp-ETEP的BGP邻居IP地址添加路由映射。因此，如果考虑流量，当站点A终端与外部IP地址通信时，数据包可以将源地址封装为“Site1_Leaf1” TEP地址，目标地址为“Site2_Leaf” IP地址192.168.100.225的ETEP地址。检验ELAM(Site1_Spine)

```

Site1_Spine# vsh_lc
module-1# debug platform internal roc elam ASIC 0
module-1(DBG-elam)# trigger reset
module-1(DBG-elam)# trigger init in-select 14 out-select 1
module-1(DBG-elam-insel14)# set inner ipv4 src_ip 90.0.0.10 dst_ip 91.0.0.1 next-protocol 1
module-1(DBG-elam-insel14)# start
module-1(DBG-elam-insel14)# status
  ELAM STATUS
  =====
  ASIC 0 Slice 0 Status Armed
  ASIC 0 Slice 1 Status Armed
  ASIC 0 Slice 2 Status Armed
  ASIC 0 Slice 3 Status Armed

```

```

pod2-n9k# ping 91.0.0.1 vrf HOST_A source 90.0.0.10
PING 91.0.0.1 (91.0.0.1) from 90.0.0.10: 56 data bytes
64 bytes from 91.0.0.1: icmp_seq=0 ttl=252 time=1.015 ms
64 bytes from 91.0.0.1: icmp_seq=1 ttl=252 time=0.852 ms
64 bytes from 91.0.0.1: icmp_seq=2 ttl=252 time=0.859 ms
64 bytes from 91.0.0.1: icmp_seq=3 ttl=252 time=0.818 ms
64 bytes from 91.0.0.1: icmp_seq=4 ttl=252 time=0.778 ms
--- 91.0.0.1 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 0.778/0.864/1.015 ms

```

Site1_Spine ELAM被触发。Ereport确认数据包使用Site-A枝叶TEP IP地址和目的地的TEP地址封装到Site2_Leaf1 ETEP地址。

```

module-1(DBG-elam-insel14)# status
  ELAM STATUS
  =====

```

```
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Armed
Asic 0 Slice 2 Status Triggered
Asic 0 Slice 3 Status Armed
module-1(DBG-elam-insell14)# ereport
Python available. Continue ELAM decode with LC Pkg
ELAM REPORT
```

```
-----
-----
Outer L3 Header
-----
-----
L3 Type           : IPv4
DSCP              : 0
Don't Fragment Bit : 0x0
TTL              : 32
IP Protocol Number : UDP
Destination IP    : 192.168.100.225    <<<'Site2_Leaf1' ETEP address
Source IP         : 10.0.80.64        <<<'Site1_Leaf1' TEP address
-----
-----
```

```
-----
-----
Inner L3 Header
-----
-----
L3 Type           : IPv4
DSCP              : 0
Don't Fragment Bit : 0x0
TTL              : 254
IP Protocol Number : ICMP
Destination IP    : 91.0.0.1
Source IP         : 90.0.0.10
-----
-----
```

站点1_主干检验路由映射当站点A主干收到数据包时，它可以重定向到“Site2_Leaf1” ETEP地址，而不是查看coop或路由条目。（当您在站点B上有站点间L3out时，站点A主干会创建名为“infra-intersite-l3out”的路由映射，以将流量重定向到站点2_Leaf1的ETEP并从L3out退出。）

```
Site1_Spine# show bgp vpnv4 unicast neighbors 192.168.11.13 vrf overlay-1
BGP neighbor is 192.168.11.13, remote AS 65001, ibgp link, Peer index 4
  BGP version 4, remote router ID 192.168.11.13
  BGP state = Established, up for 10w4d
  Using loopback12 as update source for this peer
  Last read 00:00:03, hold time = 180, keepalive interval is 60 seconds
  Last written 00:00:03, keepalive timer expiry due 00:00:56
  Received 109631 messages, 0 notifications, 0 bytes in queue
  Sent 109278 messages, 0 notifications, 0 bytes in queue
  Connections established 1, dropped 0
  Last reset by us never, due to No error
  Last reset by peer never, due to No error
  Neighbor capabilities:
    Dynamic capability: advertised (mp, refresh, gr) received (mp, refresh, gr)
    Dynamic capability (old): advertised received
    Route refresh capability (new): advertised received
    Route refresh capability (old): advertised received
    4-Byte AS capability: advertised received
    Address family VPNv4 Unicast: advertised received
    Address family VPNv6 Unicast: advertised received
    Address family L2VPN EVPN: advertised received
    Graceful Restart capability: advertised (GR helper) received (GR helper)
    Graceful Restart Parameters:
    Address families advertised to peer:
    Address families received from peer:
    Forwarding state preserved by peer for:
```

Restart time advertised by peer: 0 seconds
 Additional Paths capability: advertised received
 Additional Paths Capability Parameters:
 Send capability advertised to Peer for AF:
 L2VPN EVPN
 Receive capability advertised to Peer for AF:
 L2VPN EVPN
 Send capability received from Peer for AF:
 L2VPN EVPN
 Receive capability received from Peer for AF:
 L2VPN EVPN
 Additional Paths Capability Parameters for next session:
 [E] - Enable [D] - Disable
 Send Capability state for AF:
 VPNv4 Unicast[E] VPNv6 Unicast[E]
 Receive Capability state for AF:
 VPNv4 Unicast[E] VPNv6 Unicast[E]
 Extended Next Hop Encoding Capability: advertised received
 Receive IPv6 next hop encoding Capability for AF:
 IPv4 Unicast
 Message statistics:

	Sent	Rcvd
Opens:	1	1
Notifications:	0	0
Updates:	1960	2317
Keepalives:	107108	107088
Route Refresh:	105	123
Capability:	104	102
Total:	109278	109631
Total bytes:	2230365	2260031
Bytes in queue:	0	0

For address family: VPNv4 Unicast
 BGP table version 533, neighbor version 533
 3 accepted paths consume 360 bytes of memory
 3 sent paths
 0 denied paths

Community attribute sent to this neighbor
 Extended community attribute sent to this neighbor
 Third-party Nexthop will not be computed.

Outbound route-map configured is infra-intersite-l3out, handle obtained <<<< route-map to redirect traffic from Site-A to Site-B 'Site2_Leaf1' L3out

For address family: VPNv6 Unicast
 BGP table version 241, neighbor version 241
 0 accepted paths consume 0 bytes of memory
 0 sent paths
 0 denied paths

Community attribute sent to this neighbor
 Extended community attribute sent to this neighbor
 Third-party Nexthop will not be computed.

Outbound route-map configured is infra-intersite-l3out, handle obtained

<snip...> Site1_Spine# show route-map infra-intersite-l3out
 route-map infra-intersite-l3out, permit, sequence 1

Match clauses:
 ip next-hop prefix-lists: IPv4-Node-entry-102
 ipv6 next-hop prefix-lists: IPv6-Node-entry-102

Set clauses:
 ip next-hop 192.168.200.226

route-map infra-intersite-l3out, permit, sequence 2 <<<< This route-map match if destination IP of packet 'Site1_Spine' TEP address then send to 'Site2_Leaf1' ETEP address.

Match clauses:
 ip next-hop prefix-lists: IPv4-Node-entry-1101
 ipv6 next-hop prefix-lists: IPv6-Node-entry-1101

Set clauses:
 ip next-hop 192.168.200.225

```
route-map infra-intersite-l3out, deny, sequence 999
  Match clauses:
    ip next-hop prefix-lists: infra_prefix_local_pteps_inexact
  Set clauses:
route-map infra-intersite-l3out, permit, sequence 1000
  Match clauses:
  Set clauses:
    ip next-hop unchanged
```

```
Site1_Spine# show ip prefix-list IPv4-Node-entry-1101
ip prefix-list IPv4-Node-entry-1101: 1 entries
  seq 1 permit 10.0.80.64/32 <<
```