vzAny with PBR, Release 4.3.1

Table of Contents

New and changed information

The following table provides an overview of the significant changes up to this current release. The table does not provide an exhaustive list of all changes or of the new features up to this release.

Release Version	Feature	Description
Nexus Dashboard 4.3.1	There were no major changes from the previous release.	

vzAny with PBR Overview

The following sections provide an overview, requirements and guidelines, and configuration steps for enabling vzAny contracts with Policy-Base Redirects (PBR) in your Multi-Fabric domain. For an overview of vzAny in general and basic vzAny use cases that do not include PBR, see vzAny Contracts.

Use Cases

The following basic vzAny use cases (without PBR) are supported with Multi-Fabric, all of which are described in the vzAny Contracts:

- Free communication between EPGs within the same VRF.
- Many-to-one communication allowing all EPGs within the same VRF to consume a shared service from a single EPG that is in the same or different VRF.

The following additional use cases for vzAny with PBR are supported for ACI fabrics running APIC release 6.0(4) or later, which allow redirecting traffic to a logical firewall service connected in each fabric in one-arm mode:

- Any intra-VRF communication (vzAny-to-vzAny) between two EPGs or External EPGs within the same VRF.
- Many-to-one communication between all the EPGs in a VRF (vzAny) and a specific EPG that is part of the same VRF.
- Many-to-one communication between all the EPGs in a VRF (vzAny) and a specific External EPG that is part of the same VRF.

General Workflow for Configuring vzAny with PBR

This section outlines how to configure vzAny with Policy-Based Redirect (PBR) use cases. It includes the steps needed to create essential building blocks (templates, EPGs, contracts) and workflows to assemble them for specific configurations.

1. Start by creating a Service Device Template and associate it with the relevant tenant and fabrics. Include any necessary IP SLA policies already defined in a Tenant Policy Template. Add service node devices to the template and provide a bridge domain (BD) that exists in an Application Template. Ensure the BD aligns with the guidelines in the vzAny with PBR Guidelines and Limitations section.
    - Providing fabric-level configurations for the service node device defined in the Service Device template and deploying it. Configure fabric-level settings for the service node device and deploy the template.

      There's no Service Graph object that must be explicitly created in Nexus Dashboard for PBR use cases. Nexus Dashboard implicitly creates the service graph and deploys it in the fabric's APIC.

2. Complete the configuration for the specific tenant associated to the Service Device template that you just created, which includes:

- Creating a Tenant Application template and assigning it to all fabrics where the configuration is required.
- Configuring vzAny VRF settings required to enable PBR and a contract.
- Configuring the consumer and provider EPGs.

  While the service BD must be stretched across fabrics, the BDs you use for the EPGs can be stretched or fabric-local.

3. Associate the service device you created in Step 1 with the vzAny contract you created in Step 2.

> Please refer *ACI Contract Guide* and *ACI PBR White Paper* to understand Cisco ACI contract and PBR terminologies.

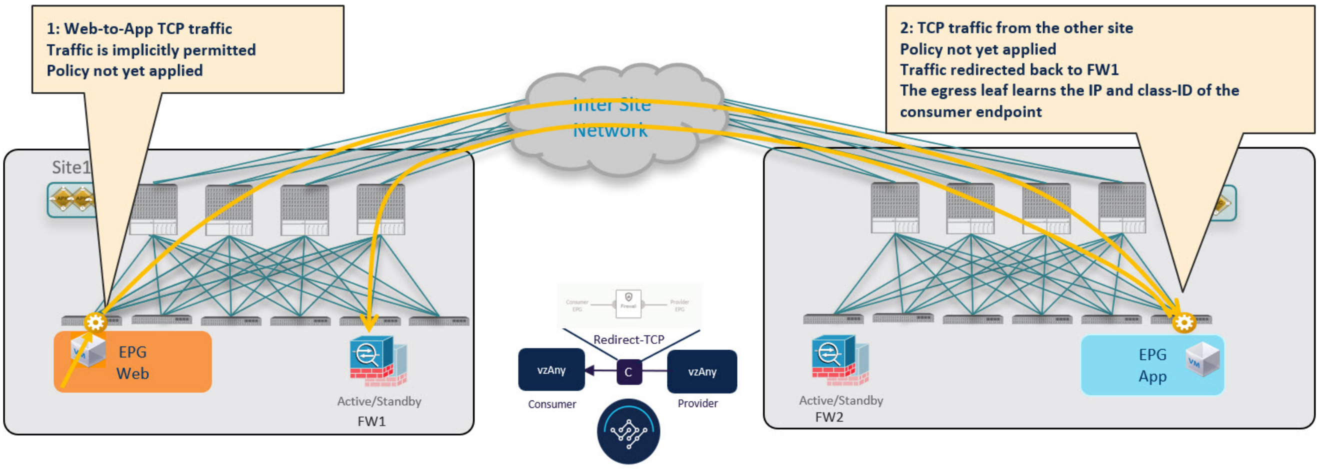
Traffic Flow: Intra-VRF vzAny-to-vzAny

This section summarizes the traffic flow between two EPGs that are part of the logical vzAny construct for a given VRF in different fabrics. In this use case, vzAny is both the provider and the consumer of a PBR contract.

> In this case, the traffic flow in both directions is redirected through both firewalls in order to avoid asymmetric traffic flows due to independent FW nodes deployed in the two fabrics.

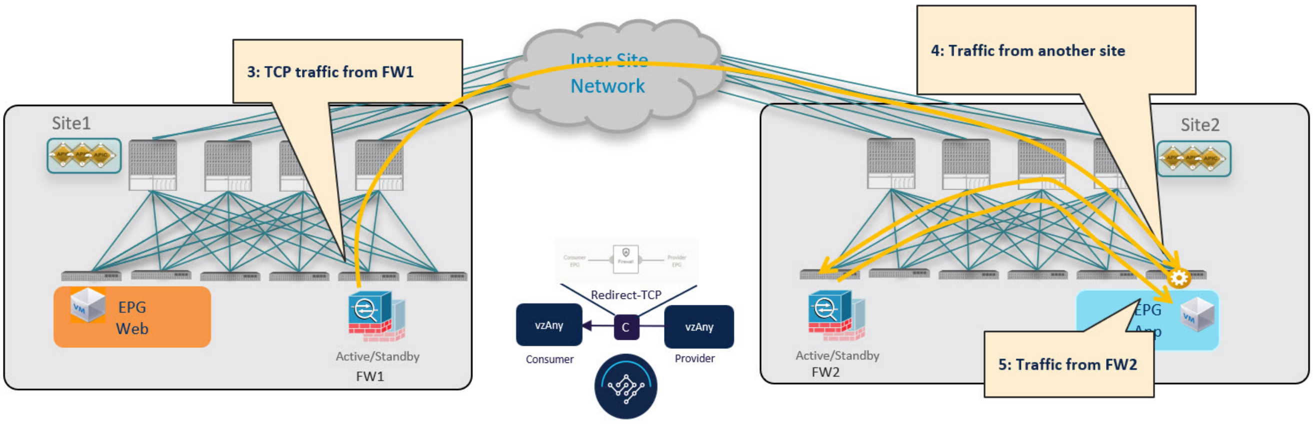
Initial Consumer-to-Provider Traffic Flow and Conversational Learning

The design principle for redirecting the traffic to the FW service nodes in both the local fabric and the remote fabric is that the PBR policy should always be applied on the ingress leaf switch for both directions of the traffic flow. For this to happen, the ingress leaf switch must be aware of the destination’s endpoint policy information (Class-ID). The figure below shows an example where communication is initiated from the consumer endpoint, and the ingress (consumer) leaf switch does not yet have the Class-ID information for the destination (provider) endpoint. So the traffic is simply forwarded toward the destination connected to the remote fabric. This release implements a new logic to support this use case, so that the provider leaf switch that receives the traffic can understand that the flow originated in Fabric 1 but it has not been sent through the firewall service node connected in that fabric. As a result, after learning the consumer endpoint information (Class-ID), the provider leaf in Fabric 2 bounces back the traffic toward the firewall in Fabric 1.



Conversational Learning

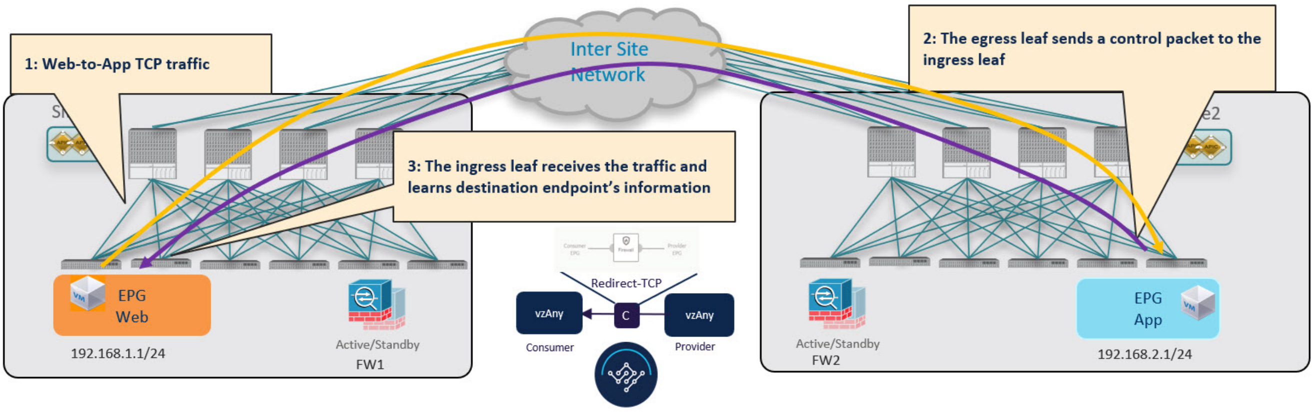
The firewall in Fabric 1 applies the security policy, then the traffic is forwarded again to the destination leaf switch in Fabric 2. This leaf is now able to understand that, while the traffic is still coming from Fabric 1, it now has been sent through the firewall deployed in that fabric. As a result, the destination leaf switch forwards the packet to its local firewall device for inspection and after that it is delivered to the destination endpoint as shown in the following figure.



Conversational Learning

In order to avoid the suboptimal bounce of traffic shown in !!!Dita2Adoc_MissingReference:!!!, the provider leaf switch generates a special control packet and sends it to the consumer leaf switch in Fabric 1, so that the consumer leaf can learn the provider endpoint’s Class-ID information.

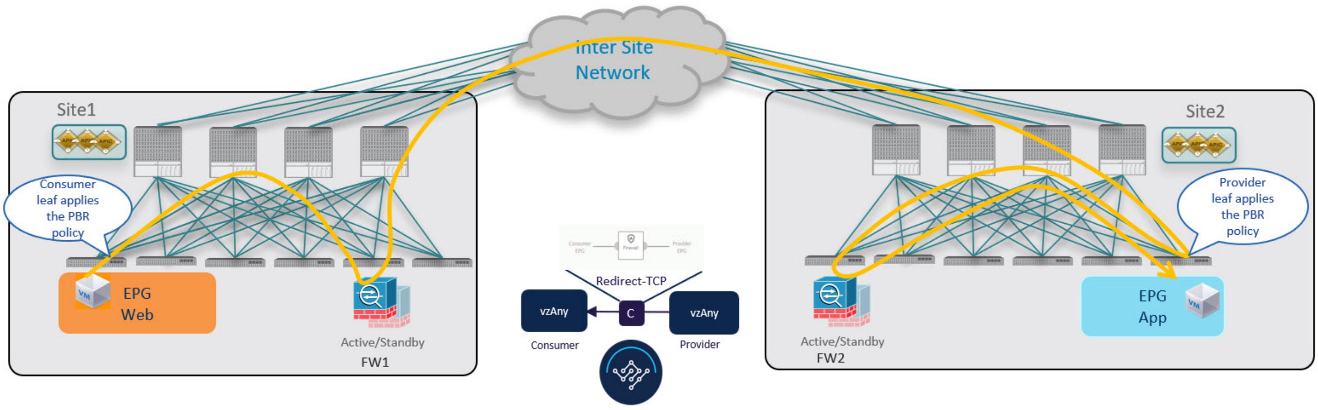
The same behavior described above for the consumer-to-provider traffic direction applies if the initial flow is established in the provider-to-consumer direction instead.



Conversational Learning

Consumer-to-Provider Traffic Flow (at Steady State)

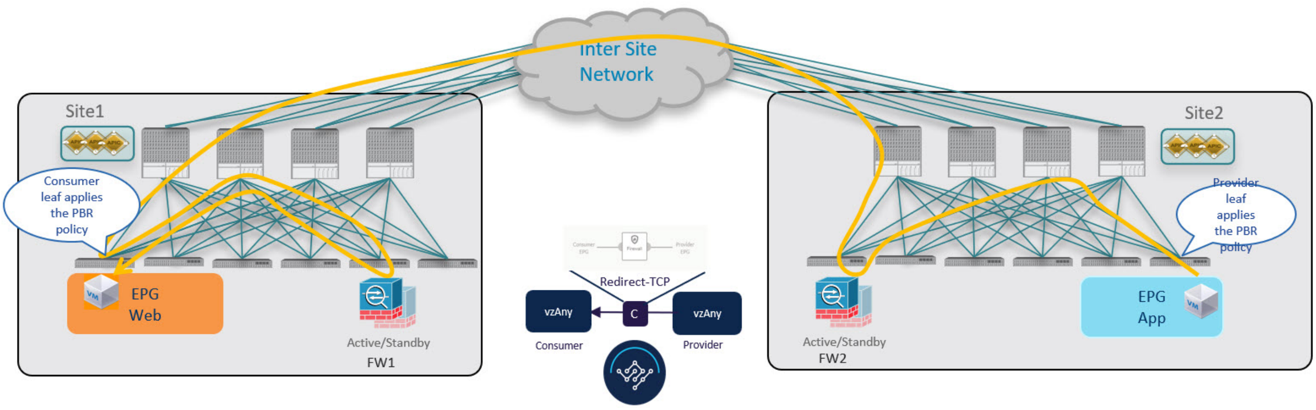
After the consumer leaf switch has learned the provider endpoint information from the conversational learning stage described above, it can apply policy and redirect traffic to its local firewall for all future traffic:



Consumer-to-Provider Traffic Flow

Provider-to-Consumer Traffic Flow (at Steady State)

After the provider leaf switch has learned the consumer endpoint information either from the direct packet shown in !!!Dita2Adoc_MissingReference:!!! or based on conversational learning, it can apply policy and redirect traffic to its local firewall for all future traffic:



Provider-to-Consumer Traffic Flow

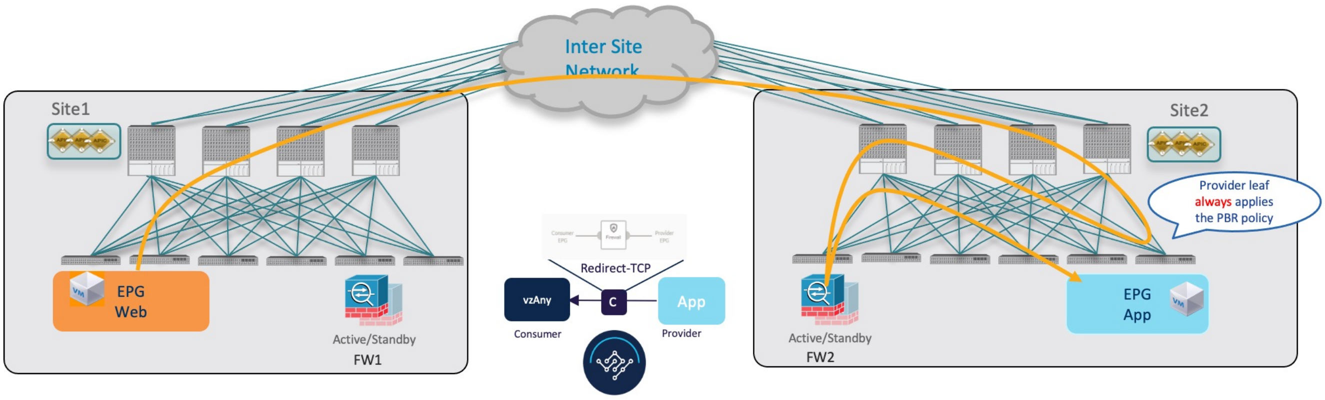
Traffic Flow: Intra-VRF vzAny-to-EPG

This section summarizes the traffic flow between a consumer EPG that is part of the logical vzAny construct for a given VRF and a provider EPG that is part of the same VRF. In this use case, vzAny is the consumer of the PBR contract, whereas a specific EPG is the provider.

Unlike the vzAny-to-vzAny and vzAny-to-L3Out use cases where traffic always flows through the firewall devices in both fabrics, vzAny-to-EPG always uses only the device in the provider's fabric.

Consumer-to-Provider Traffic Flow

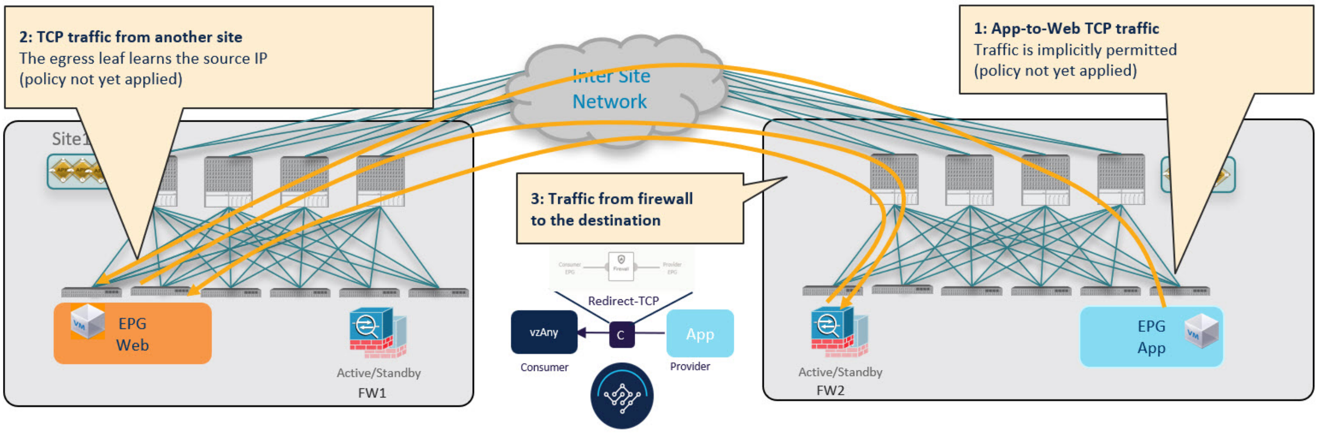
For the vzAny-to-EPG use case, policy is applied on the provider leaf switch only regardless of the traffic direction. So for consumer-to-provider traffic, the consumer EPG sends traffic directly to the provider EPG's leaf switch, which learns the consumer endpoint information (Class-ID) and redirects the traffic to its local firewall for inspection:



vzAny-to-EPG Consumer-to-Provider Traffic Flow

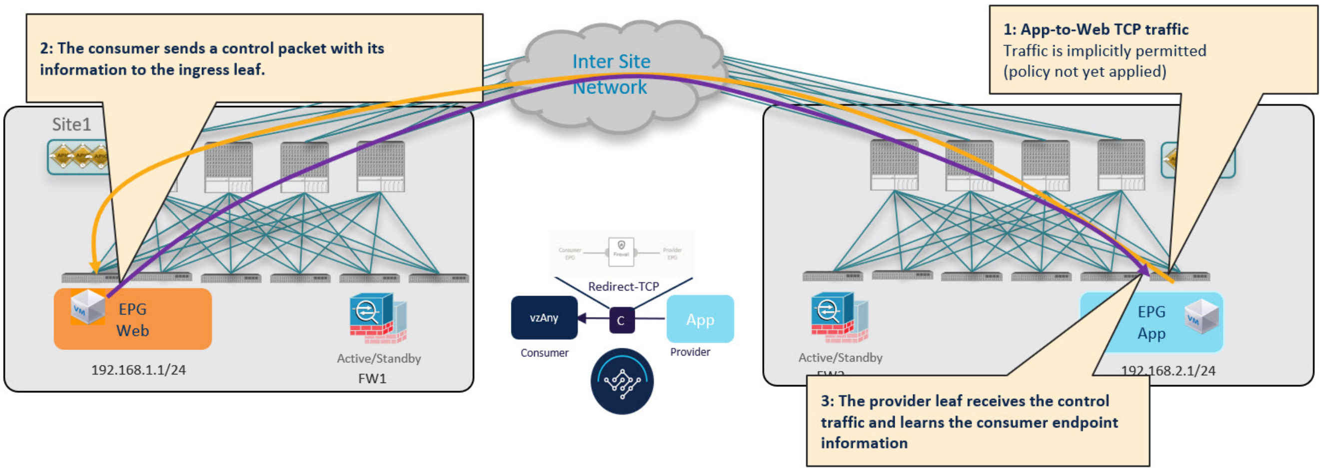
Provider-to-Consumer Traffic Flow (Initial Traffic and Conversational Learning)

If the communication is initiated by the provider endpoint before the provider leaf switch can learn the consumer endpoint information (Class-ID), it cannot apply the policy to redirect traffic to its local firewall, so the traffic is sent across fabrics to the consumer leaf switch. Because the policy was not applied (indicated by a control bit in the packet), the consumer leaf switch redirects the traffic back to the provider fabric's firewall for inspection, which finally bounces the traffic back to the consumer endpoint.



vzAny-to-EPG Provider-to-Consumer Traffic Flow (Initial Traffic and Conversational Learning)

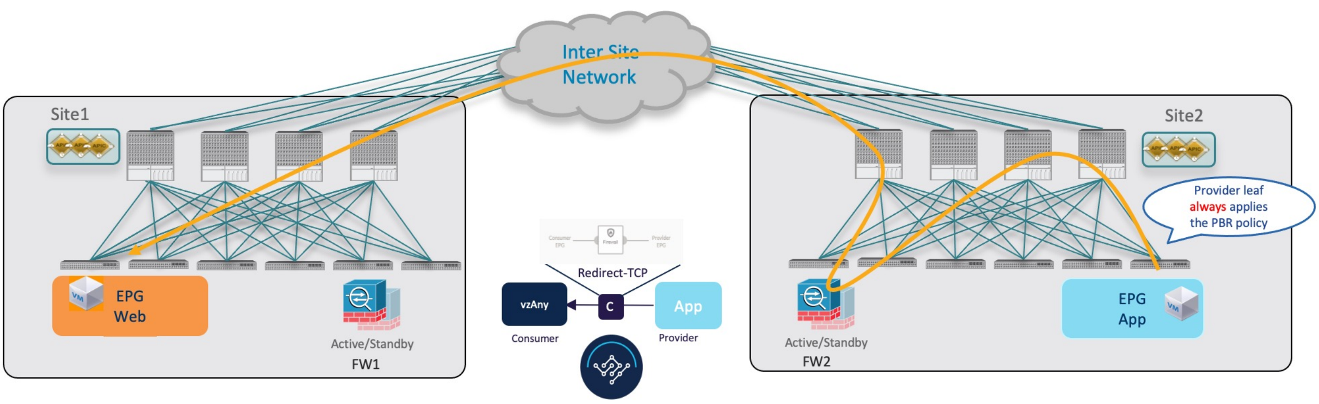
While this suboptimal traffic flow can continue indefinitely, the consumer EPG's leaf switch also sends a separate control packet to the provider leaf switch with consumer endpoint information in order to optimizes future traffic and prevent it from bouncing between both fabrics:



Conversational Learning

Provider-to-Consumer Traffic Flow (at Steady State)

After the provider leaf switch has learned the consumer endpoint information either from the direct packet originated from the consumer endpoint or based on conversational learning, it can apply policy and redirect traffic to its local firewall for all future traffic:



vzAny-to-EPG Provider-to-Consumer Traffic Flow

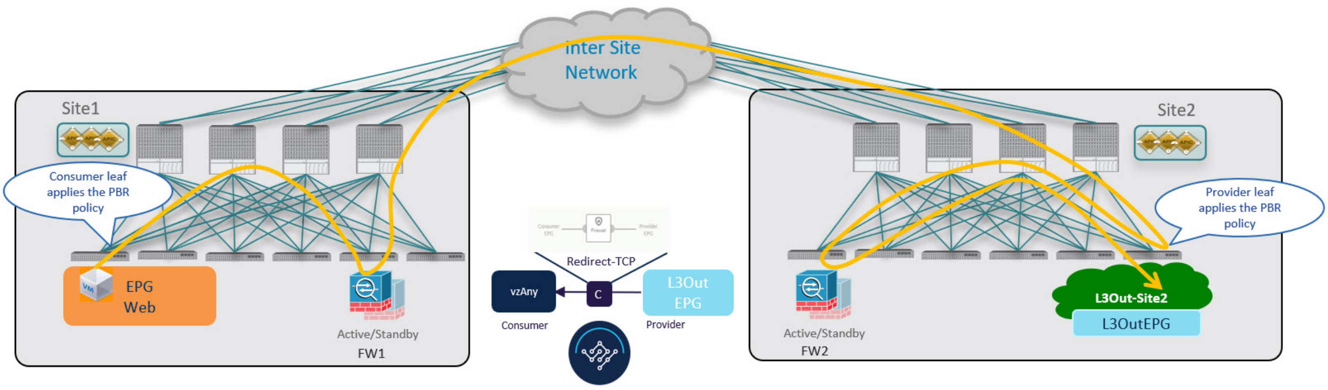
Traffic Flow: Intra-VRF vzAny-to-External-EPG (L3Out EPG)

This section summarizes the traffic flow between an EPG that is part of the logical vzAny construct for a given VRF and an external EPG (L3Out EPG) that is part of the same VRF in another fabric. In this use case, vzAny is the consumer of a vzAny contract, while an External EPG associated to the L3Out is the provider.

In this use case, the traffic is always redirected through firewall devices in both fabrics.

Consumer-to-Provider Traffic Flow

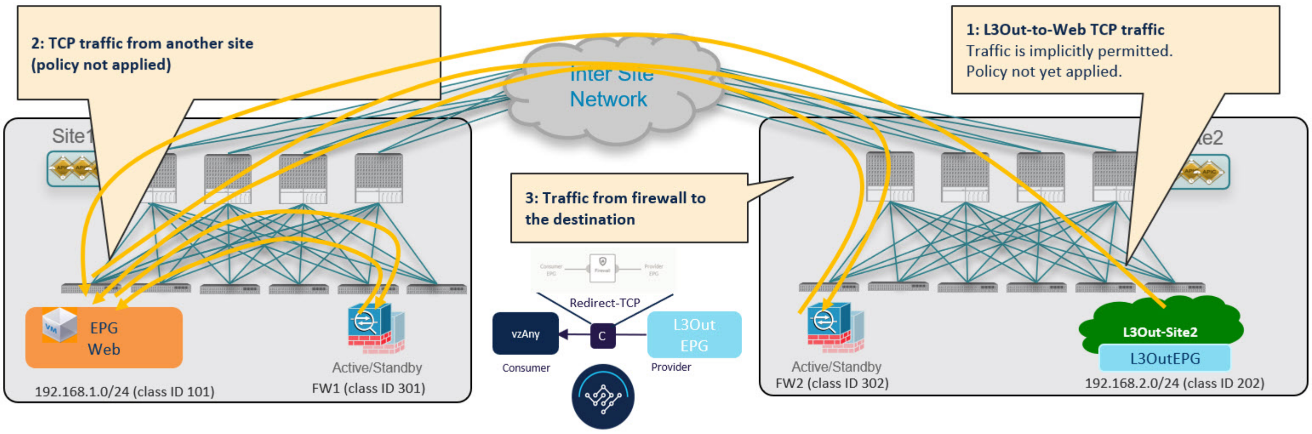
The ingress leaf switch can always resolve the class ID of the destination external EPG and applies the PBR policy redirecting the traffic to the local FW, so no conversational learning is necessary for traffic in this direction. Because the traffic is received by the provider leaf switch after going through the firewall node in Fabric 1, it is not possible for the provider leaf switch to learn the consumer endpoint information (Class-ID) from this data-plane communication.



vzAny-to-External EPG Consumer-to-Provider Traffic Flow

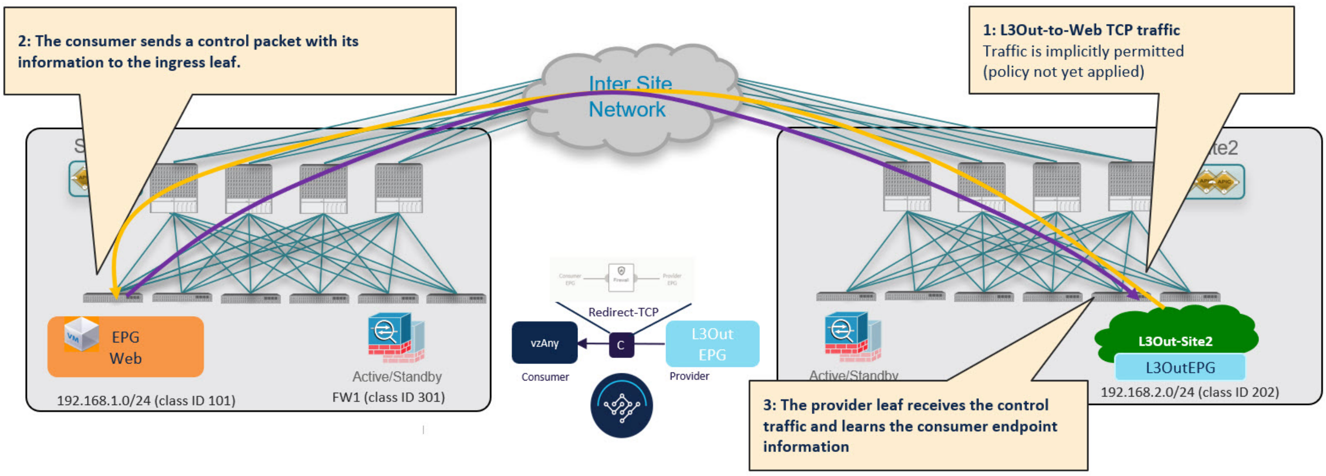
Provider-to-Consumer Traffic Flow (Initial Traffic and Conversational Learning)

Before the provider leaf switch learns the consumer endpoint information, it cannot apply the policy to redirect traffic to its local firewall, so the traffic is sent across fabrics to the consumer leaf switch. Because the policy was not applied (indicated by a control bit in the packet), the consumer leaf switch redirects the traffic back to the provider fabric’s firewall for inspection, which finally forwards the traffic back to the consumer endpoint.



vzAny-to-L3Out Provider-to-Consumer Traffic Flow (Initial Traffic and Conversational Learning)

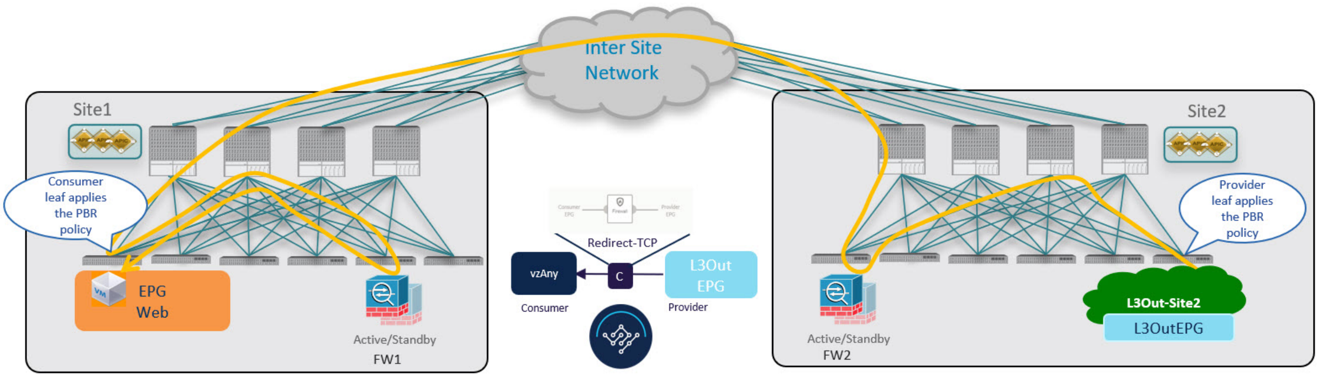
While this traffic flow can continue indefinitely, the consumer leaf switch also sends a separate control packet to the provider leaf switch with consumer endpoint information in order to optimizes future traffic and prevent it from bouncing between both fabrics:



vzAny-to-L3Out Provider-to-Consumer Traffic Flow (Initial Traffic and Conversational Learning)

Provider-to-Consumer Traffic Flow (at Steady State)

After the provider leaf switch has learned the consumer endpoint information, it applies the PBR policy to redirect traffic to its local firewall device first, which then sends traffic across fabrics to the consumer leaf switch, which redirects traffic to the firewall device in its fabric and then finally to the consumer endpoint.



vzAny-to-L3Out Provider-to-Consumer Traffic Flow

vzAny with PBR Guidelines and Limitations

The following guidelines and limitations apply when using vzAny with PBR in Multi-Fabric deployments:

> The following sections apply to the vzAny with PBR use cases only. For information about basic vzAny concepts and use cases, see vzAny Contracts.

- The ACI fabrics must be running Cisco APIC release 6.0(4) or later.
- This release supports redirecting vzAny traffic to a single-node firewall or to a single-node load-balancer with a single interface attached to the service bridge domain. Single-Node firewall is supported in vzAny-to-vzAny and vzAny-to-L3Out, vzAny-to-EPG, and L3Out-to-L3Out use cases. Single-Node load-balancer is supported in vzAny-to-EPG use case.

  This includes the following three use cases for one-arm mode firewall service graphs:

  - Any intra-VRF communication (vzAny-to-vzAny) across fabrics.
  - Many-to-one communication between all the EPGs in a VRF (vzAny) and a specific EPG that is part of the same VRF.
  - Many-to-one communication between all the EPGs in a VRF (vzAny) and a specific External EPG that is part of the same VRF.

  In all of the above cases, conversational endpoint learning is enabled only when vzAny with PBR is configured and is used when IP prefixes under the EPG are not configured. A mix of EPGs with IP prefixes and EPGs without IP prefixes is also supported.

- While you can use your existing Service Graph objects defined in Application templates for these use cases, we recommend using the service chaining workflows and implicitly creating new service graphs by defining the policies in Service Device templates and associating them to contracts.

  The steps described in the following sections use the new Service Device templates to enable the supported use cases but will call out the specific differences when applicable.

  > Configuration of Service Graph objects in Application templates will be deprecated in a future release.

- The vzAny VRF must be stretched across the fabrics.

  Note that the "Fabric-aware policy enforcement" and "L3 Multicast" options must be enabled for the vzAny VRF to enable the vzAny PBR use cases discussed in this chapter.

  The following sections assume that you already have a VRF for which you have or will enable vzAny and which you will use for these use cases.

  If you do not already have a VRF, you can create one in an Application template as you typically would. VRF configuration is described in detail in Configuring VRFs.

- The service BD to which you want to attach the service device interface must be L2 stretched (BUM forwarding is instead optional and should be disabled).

If you do not already have a service BD, you can create one in an Application template as you typically would. BD configuration is described in detail in Configuring Bridge Domains.

- The consumer, provider, and the service BDs must be configured in Hardware proxy-mode.
- The vzAny PBR destination must be connected to a stretched service BD, not to an L3Out.
- Only threshold down deny action and sip-dip-protocol hash is supported .
- The PBR destination node must be in either the consumer or provider VRF instance. For example see *Cisco Application Centric Infrastructure Policy-Based Redirect Service Graph Design White Paper*.

The following is not supported for vzAny with PBR use case:

- Specific Remote Leaf configurations.
  - Specific considerations apply for Multi-Fabric deployments leveraging Remote Leaf nodes. Inter-fabric transit routing with PBR is not supported on vzAny PBR and L3Out-to-L3Out for communication between endpoints (consumer or provider) deployed on remote leaf nodes that belongs to different fabrics.
- Each VRF is limited to utilizing only one device in a one-arm configuration for vzAny-to-vzAny, L3OutEPG-to-L3OutEPG, and vzAny-To-L3OutEPG Policy-Based Routing (PBR). This restriction is enforced due to special ACL in APIC.
- We must use different firewall VLAN interfaces for redirection for vzAny-to-vzAny/L3OutEPG-to-L3OutEPG, and other use cases such as vzAny-to-EPG, EPG-to-EPG and EPG-to-L3OutEPG if they are in the same VRF.
- When vzAny with PBR is applied to north-south communication for any of the newly supported use cases (vzAny-to-vzAny, vzAny-to-L3OutEPG, L3OutEPG-to-L3OutEPG), ingress traffic optimization needs to be enabled for stretched subnets.
- Only one node service chain with L3 PBR destination is supported.
- Contract Permit logging is not supported on the VRF that has Fabric-aware Policy Enforcement Mode is enabled, which is required for vzAny PBR and L3OutEPG-to-L3OutEPG PBR.
- Pod-aware vzAny with PBR is not supported.

Create Service Device Template

Before you begin:

- Ensure that you have read and completed the requirements described in vzAny with PBR Guidelines and Limitations.
- You must have created a stretched service bridge domain (BD) to use with the service nodes you will define in this section.

  If you do not already have a BD, you can create one in an Application template as you typically would. BD configuration is described in detail in Configuring Bridge Domains.

The following steps describe how to create a Service Device template with a service node and its settings which you will use for the vzAny with PBR use cases.

1. Navigate to the **Orchestration** page.

   Manage > Orchestration

2. Click **Tenant Templates**.
3. (Optional) Create a Tenant Policies template and an IP SLA monitoring policy.

   We recommend that you configure an IP SLA policy for traffic redirection as it simplifies the configuration of the PBR policy described in Step 7 below. If you have an IP SLA policy already defined, you can skip this step, otherwise:

   a. Click **Tenant Policies**.
   b. On the **Tenant Policies** page, click **Create Tenant Policy Template**.
   c. In the **Tenant Policies** page's right properties sidebar, provide the **Name** for the template and **Select a Tenant**.
   d. In the **Template Properties** page, choose **Actions > Add/Remove Fabrics** and associate the template with both fabrics.
   e. In the main pane, choose **Create Object > IP SLA Monitoring Policy**.
   f. Provide the **Name** for the policy, and define its settings.
   g. Click **Save** to save the template.
   h. Click **Deploy Template** to deploy it.
4. Create a Service Device template and associate it with a tenant and with the fabrics.
   a. From **Tenant Templates**, click **Service Device**.
   b. Click **Create Service Device Template**.
   c. In the template properties sidebar that opens, provide the **Name** for the template and **Select a Tenant**.
   d. In the **Template Properties** page, choose **Actions > Add/Remove Fabrics** and associate the template with both fabrics.
   e. Click **Save** to save the template.
5. Create and configure the device cluster.

a. In the **Template Properties** page (template-level configuration), choose **Create Object > Service Device Cluster**.

   The device cluster defines the service to which you want to redirect traffic. This release supports redirection to a firewall service node that can be deployed with three different redundancy models: active/standby, active/active, or a cluster of multiple independent nodes. The provisioning for those different options is covered in Step 7 below. Note that you can choose the firewall deployment model at the fabric level and different options can be deployed across different fabrics that are part of the same Multi-Fabric domain.

b. In the **<cluster-name>** sidebar, provide the **Name** for the cluster.

   The **Device Location** and **Device Mode** are pre-populated based on the currently supported use case. **Device Location** should be pre-configured as ACI On-Prem and **Device Mode** as L3.

c. For **Device Type**, choose Firewall.

   This release supports only firewall devices for the vzAny with PBR use cases.

d. For **Device Mode**, choose L3.

e. For **Connectivity Mode**, choose One Arm.

   This release supports only one-arm device for the vzAny with PBR use cases.

   When changing the device connectivity mode between one arm, two arm and advanced mode, the name of the device interface might change in the process. A warning message will alert the user, and any attempt to modify the interface will be restricted if the interface is currently in use by a contract. If the user wishes to preserve the previously used interface name and avoid disrupting the deployed configuration, they may choose to override the name change during the modification process.

   Validations are conducted only for one-arm and two-arm modes. In Advanced mode, no validations are performed, and it is assumed that the user is an expert when choosing this mode.

f. Provide the **Interface Name**.

g. For the **Interface Type**, choose BD.

   For vzAny with PBR use cases, this release supports attaching the service device to a bridge domain only.

h. Click **Select BD >** to choose the service bridge domain to which you want to attach this device.

   This is the stretched service BD you created as part of the vzAny with PBR Guidelines and Limitations, for example FW-external.

i. For the **Redirect** option, choose Yes.

   You must choose to enable redirect for the PBR use case. After choosing Yes, the **IP SLA**

Monitoring Policy option becomes available.

j. (Optional) Click **Select IP SLA Monitoring Policy** and choose the IP SLA policy you have created in a previous step.

k. (Optional) In the **Advanced Settings** area, choose **Enable** if you want to provide additional settings for the service cluster.

   You can configure the following advanced settings:

   - **QoS Policy**—Allows you assign a specific QoS level within the ACI fabrics for the redirected traffic.
   - **Preferred Group** - specifies whether or not this service device interface is part of the preferred group.

     Leave this option disabled when configuring a vzAny use case.

   - **Load Balancing Hashing**—Allows you to specify the hashing algorithm for PBR load balancing.

     You must keep the default value for the vzAny-to-vzAny, vzAny-to-ExtEPG, and ExtEPG-to-ExtEPG use cases as they support only the default configuration. You can change the load balancing hashing for other use cases: EPG-to-EPG, ExtEPG-to-EPG and vzAny-to-EPG.

     For additional information, see *ACI Policy-Based Redirect Service Graph Design*.

   - **Pod Aware Redirection**—Can be configured in Multi-Pod configuration if you want to specify the preferred PBR node. When you enable Pod-aware redirection, you can specify the Pod ID and redirection is programmed only in the leaf switches located in the specified Pod.
   - **Rewrite Source MAC** - updates the source MAC address if the PBR node uses "source MAC based forwarding" instead of IP based forwarding.

     For additional information, see *ACI Policy-Based Redirect Service Graph Design*.

   - **Advanced Tracking Options**—Allows you to configure a number of advanced settings for the service node tracking. For additional information, see *Policy-Based Redirect and Threshold Settings for Tracking Service Nodes*

l. Click **Ok** to save.

   Note that after you create the Service Device Cluster, it is highlighted in red in the **Template Properties** (template-level configuration) page. At this point, you have defined redirection to a firewall service, but you must still provide the firewall information and the redirect policy you want to use at the fabric-local level.

6. Provide fabric-local configuration for the Service Device Cluster you created in the previous step.

   a. In the **Service Device Template** screen, choose the **<fabric-name>** tab.

   b. At the fabric level, choose the Service Device Cluster you created.

   c. In the properties sidebar, choose the **Domain Type**.

You can choose whether the firewall device in this fabric is Physical or VMM (virtual and hosted by a hypervisor that is part of a VMM domain).

d. Click **Select Domain** to choose the domain to which this firewall device belongs.

   You can choose either a physical or a virtual domain.

   - If you choose a physical domain, provide the following information:
     - **VLAN** - you must provide the VLAN ID used for traffic between the fabric and the firewall device.
     - **Fabric to Device Connectivity** - provide the switch node and interface information for the fabric's connectivity to the firewall device.
   - If you choose a VMM domain, provide the additional options:
     - **Trunking Port** - used to enable tagged traffic for the L4-L7 VM.

       By default, the ACI service graph configuration creates access-mode port groups and attaches them to the vNIC of the L4-L7 VM automatically.
     - **Promiscuous Mode** - required if the L4-L7 virtual appliance must receive traffic destined to a MAC address that is not the vNIC MAC owned by the VM.
     - **VLAN** - optional configuration for VMM domains and will be allocated from the dynamic VLAN pool associated with the domain if not specified.
     - **Enhanced LAG Option** - if you are using enhanced LACP for the port channel between the hypervisor and the fabric.
     - **VM Name** - choose the firewall's VM from the list of all VMs available in this VMM domain and the interface (**VNIC**) used for the firewall traffic.

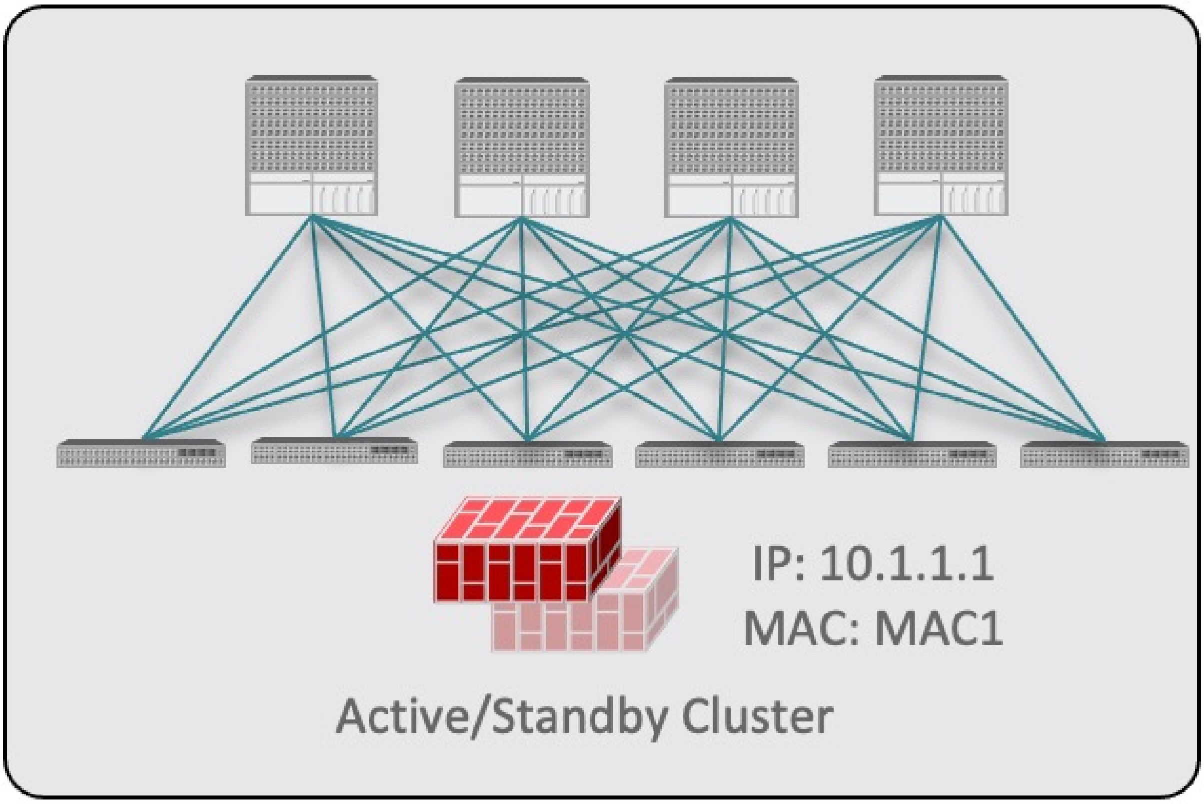
       Depending on the kind of device cluster you are deploying, click **+Add VM information** to provide additional cluster nodes.

7. Provide the FW device information and PBR destination IP addresses.

   As previously mentioned, this release supports 3 deployment options for high-availability FW clusters: active/standby clusters, active/active clusters, and independent active nodes. In all three deployment options, the use of an IP SLA policy (mentioned in Step 3) allows to specify only the IP address of the firewall nodes, and the corresponding MAC address will be automatically discovered.

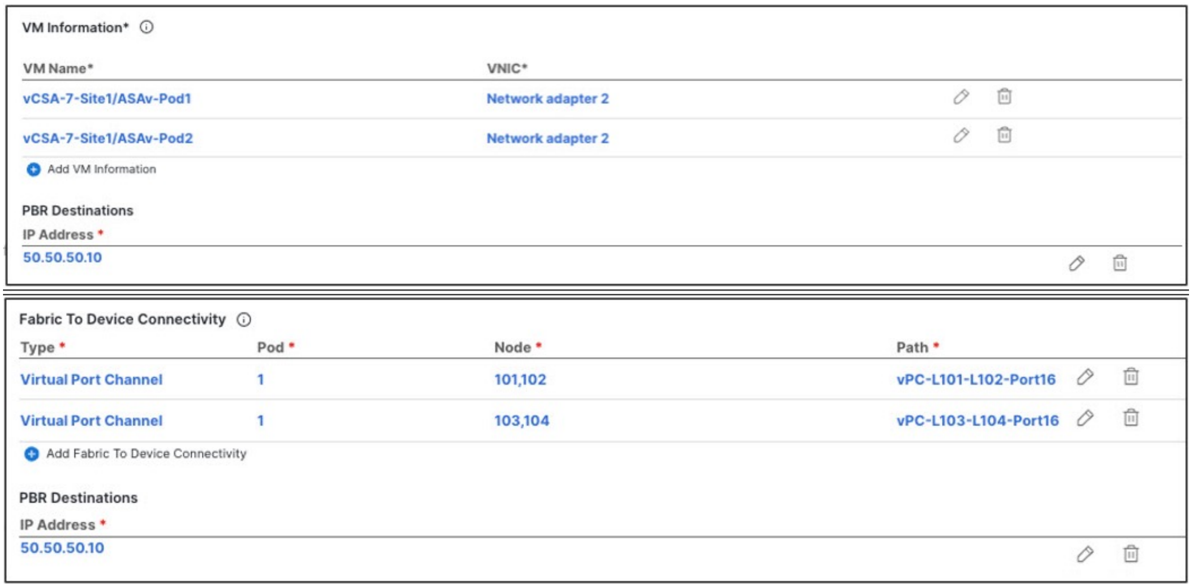
   You can deploy different designs in different fabrics.

   - Active/standby clusters are identified by a single MAC/IP pair.



In this case, you need to provide a single PBR destination IP address identifying the active firewall node and also include information about every node in the cluster.
For example, for a 2-nodes active/standby cluster, you would provide the following:

- For a virtual firewall cluster, the VMs representing the active and standby firewall nodes and the IP address of the active firewall as PBR destination.
- For a physical firewall cluster, the interfaces used to connect the active and standby firewall nodes to the leaf switches of the fabric (vPC interfaces in the specific example below) and the IP address of the active firewall as PBR destination.



VM Information*

VM Name*	VNIC*
vCSA-7-Site1/ASAv-Pod1	Network adapter 2
vCSA-7-Site1/ASAv-Pod2	Network adapter 2

Add VM Information

PBR Destinations

IP Address *
50.50.50.10

Fabric To Device Connectivity

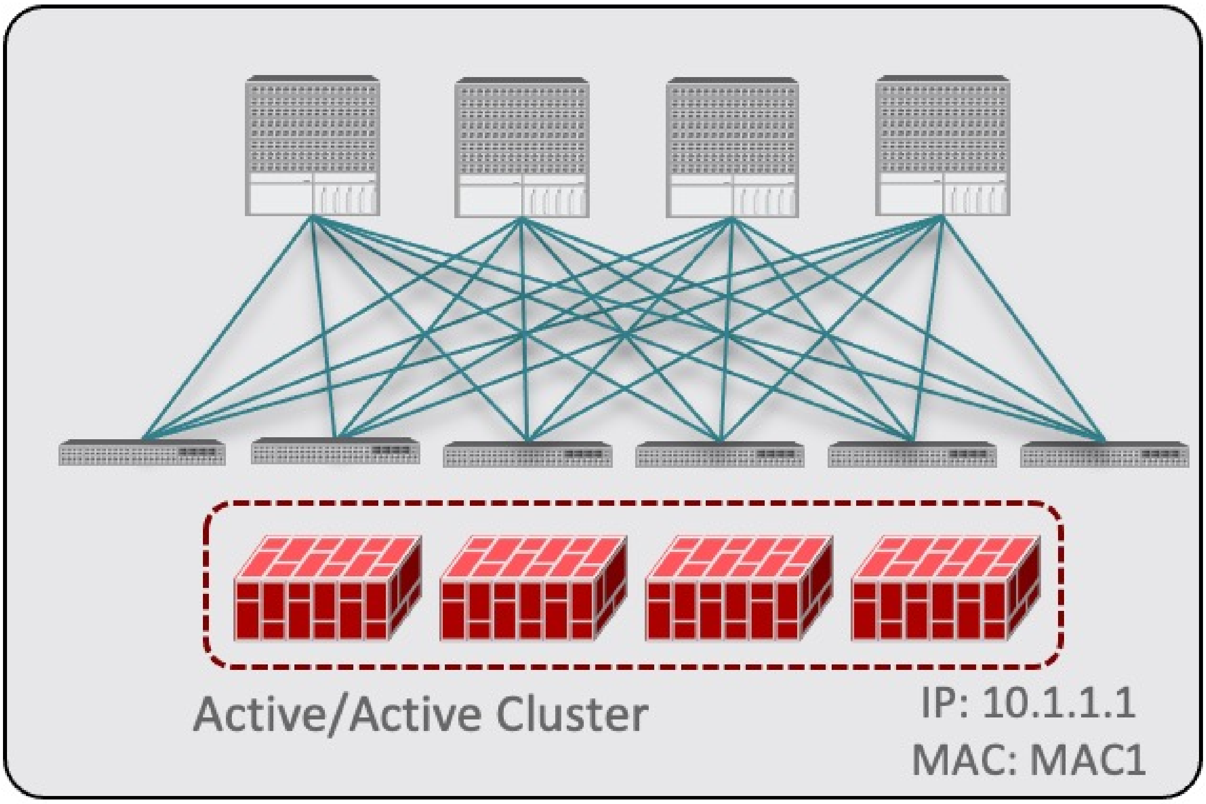
Type *	Pod *	Node *	Path *
Virtual Port Channel	1	101,102	vPC-L101-L102-Port16
Virtual Port Channel	1	103,104	vPC-L103-L104-Port16

Add Fabric To Device Connectivity

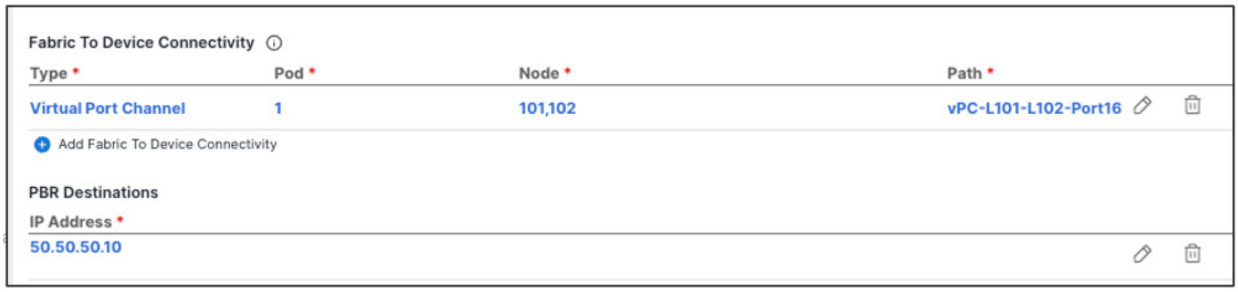
PBR Destinations

IP Address *
50.50.50.10

- Active/active clusters are also identified by a single MAC/IP pair.

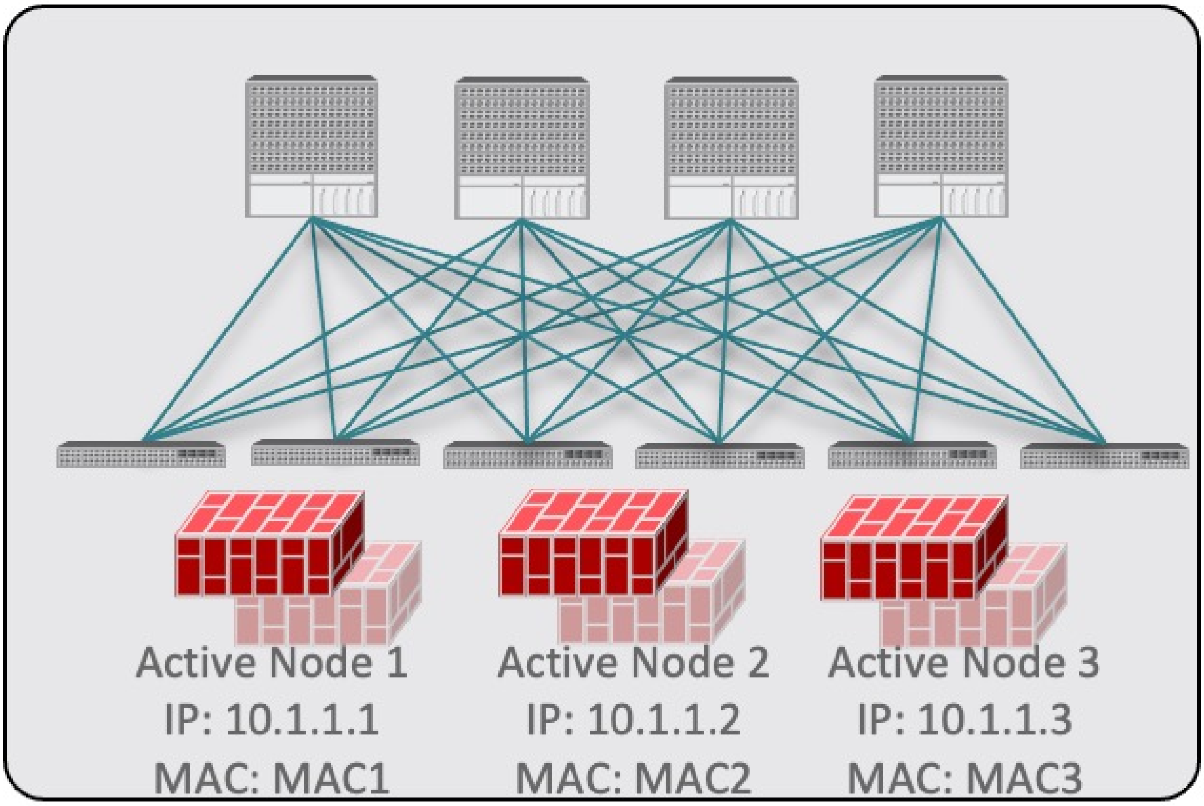


For Cisco Firewalls (ASA or FTD models), the Active/Active cluster is only supported for physical form factors, and all the cluster nodes own the same MAC/IP address and must be connected to the same vPC logical connection deployed on a pair of ACI leaf switches. As a result, the figure below shows how a single vPC interface and a single IP address should be configured on Nexus Dashboard, where the MAC address is dynamically discovered when using an IP SLA policy mentioned for the previous use case.



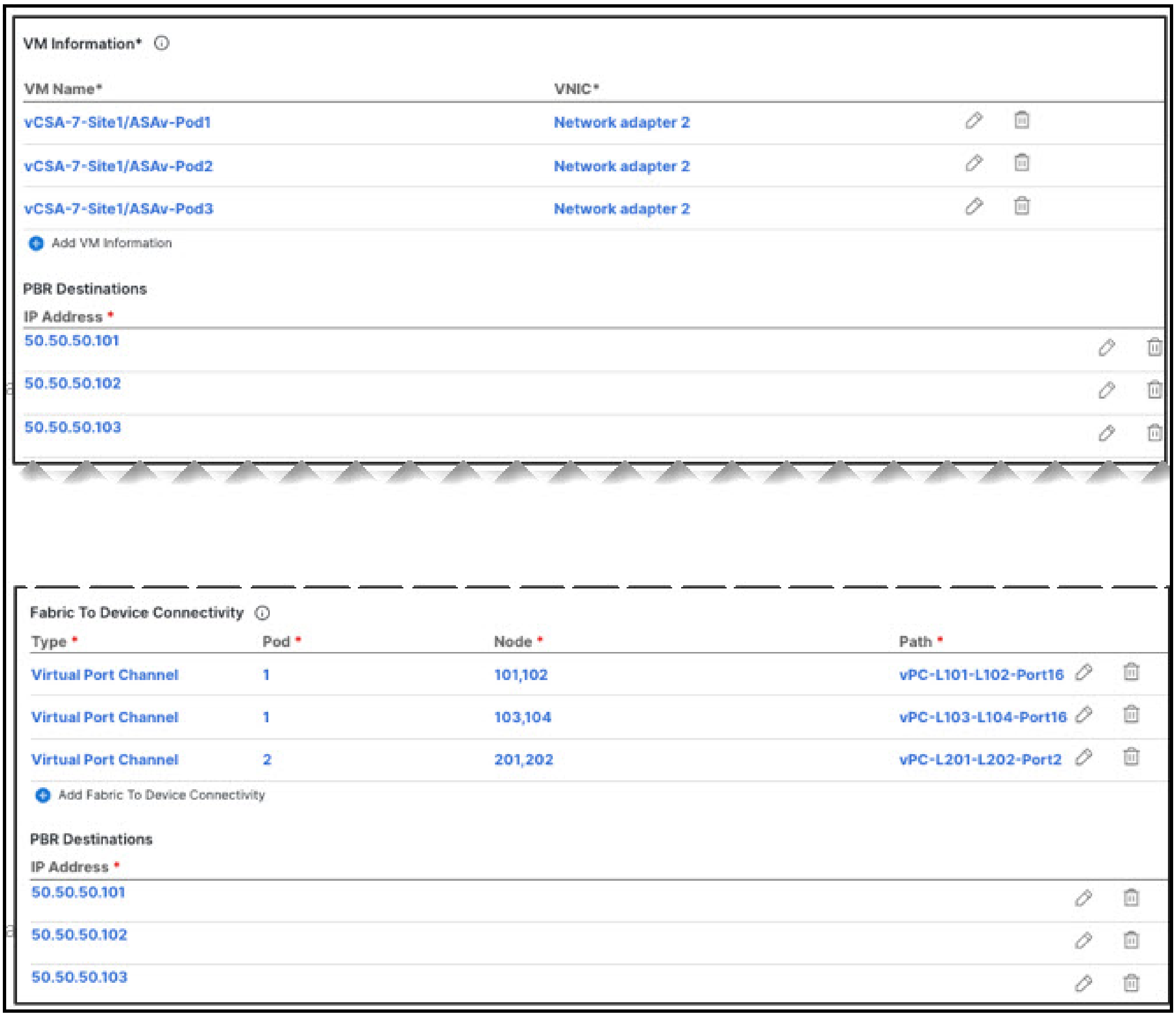
- For independent active nodes configuration, each active node is identified by a unique MAC/IP addresses pair.

  Note that symmetric PBR ensures that the traffic is handled by the same active node in both directions.



In this case, you must provide individual IP addresses for each active node as well as each node's information in your Nexus Dashboard configuration.
For example, for a deployment of 3 independent firewall nodes, you would provide the following:

- For a virtual firewall form factor, the VMs representing the 3 firewall nodes and their unique IP addresses as PBR destinations.
- For a physical firewall form factor, the interfaces used to connect each firewall node to the leaf switches of the fabric (vPC interfaces in the specific example below) and the unique IP addresses of each firewall node as PBR destinations.



a. Click **Add Fabric To Device Connectivity** (physical domain) or **Add VM Information** (VMM domain).

   Depending on whether you selected physical or VMM domain in the previous step, you will specify information for either the firewall VM or the physical fabric connectivity to the firewall device.

   For physical domains, provide the Pod, switch node, and the interface information.

   For VMM domains, provide the VM name and vNIC information.

b. Click **Add PBR Destination** to provide the IP address of the interface on the firewall that is connected to the service bridge domain.

   Depending on the kind of device cluster you are deploying, you may need to provide one or more PBR destination IP addresses:

   > This does not provision the IP address on the firewall's interface, but simply configures redirection of traffic toward that IP address. The specific firewall configuration is not deployed from Nexus Dashboard, and you must provision it separately.

c. Click **Ok** to save the provided configuration.

d. Repeat this step for the other fabric with which you associated the template.

8. Save and deploy the template.

    a. At the **Service Device Template** level, click **Save** to save the template configuration.

    b. Choose the **Template Properties** tab and click **Deploy Template** to push the configuration to the fabrics.

    c. (Optional) Verify that the configuration was created at the fabric level.

    You can verify that the L4-L7 device is configured in the APIC by navigating to **<tenant-name> > Services > L4-L7 > Devices > <cluster-name>** in the APIC GUI. This shows the device cluster along with all the configuration you have provided in the previous steps.

    To verify that the PBR policy is now configured on the APIC, navigate to **<tenant-name> > Policies > Protocol > L4-L7 Policy-Based Redirect** and you should see the **<cluster-name>-one-arm** redirect defined with the IP SLA monitoring policy you chose in *Step 8i* and the IP address you provided in *Step 7d*.

What to do next:

After you have deployed the service device configuration, create the application template and a contract with which you will associate the service chaining as described in Create Application Template.

Create Application Template

Before you begin:

- Ensure that you have read and completed the requirements described in vzAny with PBR Guidelines and Limitations.
- You must have created a VRF for which you have or will enable vzAny and which you will use for these use cases.

  If you do not already have a VRF, you can create one in an Application template as you typically would. VRF configuration is described in detail in Create Contract and Filters.

The following steps describe how to create tenant template(s) and the configuration objects which you will use for the vzAny with PBR use cases.

1. Navigate to the **Orchestration** page.

   Manage > Orchestration

2. Choose **Tenant Templates > Applications**.
3. Choose the Schema where you want to define your configuration.

   If you have an existing Schema you want to update, simply click the Schema's name in the main page. Otherwise, if you want to create a new Schema, click the **Add Schema** button and provide the schema information as you typically would.

4. Choose the Template where you want to define your configuration.

   If you have an existing template you want to update, choose the template in the schema view.

   While these steps describe how to create a single application template and stretch all objects across both fabrics, only the service BD (BD FW-external) must be stretched. The EPG BDs can be configured as stretched or fabric-local; if you choose to configure fabric-local BDs for the EPGs, you will need to create additional application templates for those objects and assign them to the specific fabrics only.

   To create a new template:

   a. Click **Create Template**.
   b. In the **Select a Template Type** screen, choose ACI Multi-Cloud.
   c. Provide the **Display Name** for the template and **Select a Tenant**.
   d. For the **Deployment Mode**, you can choose either Multi-Fabric or Autonomous.

      The vzAny with PBR use cases described in this chapter can be deployed for both Multi-Fabric and autonomous templates. If you choose to create an autonomous template, the redirection policy would apply only for intra-fabric traffic flows.

   e. Click **Continue to Template** to save the information.
   f. Choose **ActionsAdd/Remove Fabrics** and associate the template with the fabrics.

g. Repeat these substeps if you want to create additional templates for non-stretched objects.

5. Create a contract.

   You will associate a service device previously defined in the Service Device template to this contract to enable the PBR functionality. The contract will then be used (consumed/provided) by vzAny and by EPG/ExtEPG depending on the specific use case to provision.

   a. In the **Template Properties** view, choose **Create Object > Contract** to add a new contract.
   b. Provide the name for the Contract.

      For example, vzAny-to-vzAny.

   c. From the **Scope** dropdown, choose VRF.

      You must set the contract's scope to VRF.

   d. Click **+Create Filter** to add one or more contract filters.

      For example, you can create a Permit-IP contract filter to redirect all traffic.

   e. Skip the **Service Chaining/Service Graph** configuration for now, you will associate a Service Device template to this contract in the next sections.
   f. Define the other contract options as you typically would and click **Ok** to save.

6. Enable the required settings on the VRF.

   a. Select the VRF you want to use for the vzAny with PBR use case.

      You can use an existing VRF or create a new one as you typically would.

   b. Enable **vzAny** and **Add Contract** that you created in the previous step.

      The contract **Type** depends on the use case you want to configure:

      - For any intra-VRF communication (vzAny-to-vzAny) use case, assign the contract to the VRF twice - once as consumer and again as a provider.
      - For many-to-one communication between all the EPGs in a VRF (vzAny) and specific EPG as part of same VRF, assign the contract as consumer if you want the vzAny EPGs as consumer and specific EPG as provider.
      - Similarly, for many-to-one communication between all the EPGs in a VRF (vzAny) and a specific External EPG that is part of the same VRF, assign the contract as consumer if you want the vzAny EPGs to consume a service provided by an L3Out External EPG.

   c. Enable **Fabric-aware Policy Enforcement Mode**

      You must enable the **Fabric-aware Policy Enforcement Mode** setting on the VRF to enable the new vzAny PBR use cases.

      > Enabling or disabling the **Fabric-aware Policy Enforcement Mode** option will cause a brief traffic disruption (including the already existing contracts between EPGs) because the zoning rules must be updated on the leaf switches. We recommend that you perform this operation during a maintenance window.

> Enabling **Fabric-aware Policy Enforcement Mode** increases TCAM usage on the leaf switches for the existing contracts and contracting permit logging cannot be used in conjunction with this option.

d. Enable **L3 Multicast**.

   The L3 Multicast option must be enabled for the vzAny VRF to enable the conversational learning functionality described earlier in this chapter.

e. Click **Ok** to save the changes.

7. Ensure that the service BD is associated with the same VRF as you used for the vzAny contract in the previous step.
8. Create application bridge domains, configured in hardware proxy mode.

   Each application EPG that you will create in the next step requires a BD to be associated with it.

   a. In the **Template Properties** view, choose **Create ObjectBridge Domain**.
   b. Provide the name for the BD.

      For example, BD-App.

   c. From the **Virtual Routing & Forwarding** dropdown, ensure to select the VRF from the previous step.
   d. Define the other BD options as you typically would.

      For additional information about all available BD configurations, see Configuring Bridge Domains.

   e. Click **Ok** to save the changes.
   f. Repeat this step to create the second BD.

      Following the illustration above, use BD-Web for the BD's name.

9. Create the EPGs.

   In this step, you will configure either two application EPGs or an application EPG and an External EPG depending on your specific use case.

   a. Create an Application Profile by choosing **+Create Object > Application Profile**.
   b. Choose **+Create ObjectEPG** and select the application profile you created.
   c. In the properties pane, provide the **Display Name** for the EPG and choose the BD you created for this EPG.

      For example, EPG-App. For additional information about all available BD configurations, see Configuring Application Profiles and EPGs.

   d. Define the other EPG options as you typically would.

      For additional information about all available BD configurations, see Configuring Bridge Domains.

e. Click **Ok** to save the changes.

f. Create a second EPG.

   The type of the EPG and its contract configuration depend on the use case you want to configure:

   - Any intra-VRF communication (vzAny-to-vzAny).

     This is the use case illustrated in Traffic Flow: Intra-VRF vzAny-to-vzAny, and you can simply create a second EPG in the same VRF. For example, create EPG-Web and assign the BD-Web bridge domain to it.

   - Many-to-one communication between all the EPGs in a VRF (vzAny) and a specific EPG that part of the same VRF

     In this case, create a second EPG within the same VRF but explicitly assign the contract to it as provider (the vzAny VRF contract for the specific EPG is assigned as consumer).

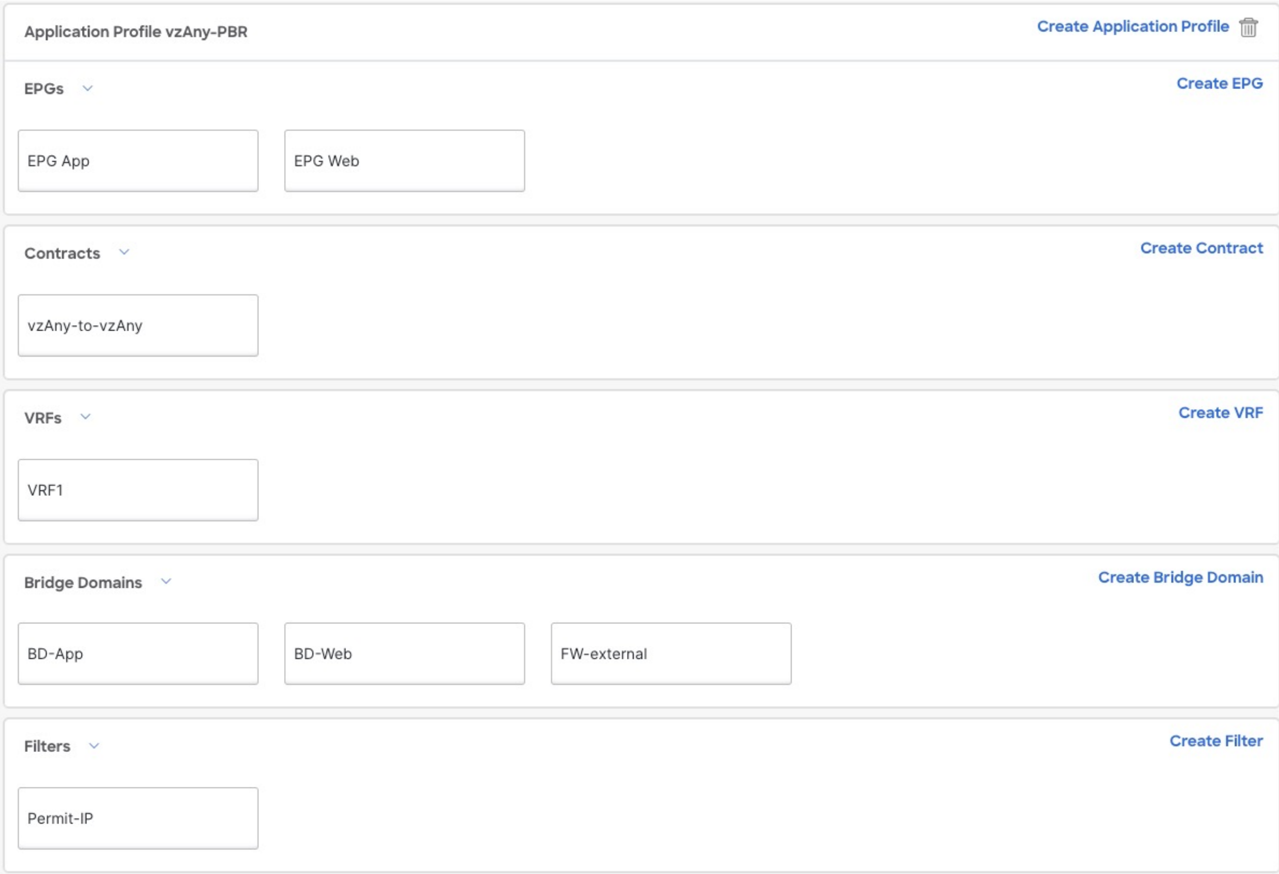
   - Many-to-one communication between all the EPGs in a VRF (vzAny) and a specific External EPG that is part of the same VRF

     In this case, you must create an External EPG instead (**+Create Object > External EPG**), associate an L3Out with the external EPG, and then explicitly assign the contract to the External EPG as provider.

10. Click **Save Schema** to save the defined configurations.

    We recommend not deploying the template until the service chaining has been configured as described in the next section to avoid undesirable communication between endpoints without firewall redirection.

    At this stage, you have effectively configured a basic use case for vzAny communication between two EPGs without adding service chaining with PBR:



The next section describes how to associate the service device you created in the previous section with the contract you created in the previous step.

What to do next:

After you have created the application template and the contract, proceed to associating the service device with the contract, as described in Add Service Chaining to Contract.

Add Service Chaining to Contract

Before you begin:

- You must have created and deployed the service device template containing the device configuration as described in Create Service Device Template.
- You must have created (but not yet deployed) the application template containing the application bridge domains and EPGs as described in Create Application Template.

After you have created the application and the service device templates, you can add policy-based redirection by associating the contract with the service devices you created in a previous section.

1. Navigate back to the application template that you created in the previous section.
2. Select the contract you created in the previous section.
3. In the **Service Chaining** area, click **+Service Chaining**.

    These steps assume that you have configured a brand new service device for this use case using the Service Device template workflow as described in Create Service Device Template. If you already have a Service Graph defined in an application template, choose Service Graph instead and then select the existing service graph. However, keep in mind that the Service Graph option will be deprecated in a future release.

4. For **Device Type**, choose Firewall.

    This release supports one-arm firewall service graphs only.

5. From the **Device** dropdown, choose the FW device cluster you created in the previous step.
6. Ensure that **Consumer Connector Type Redirect** is enabled.
7. Ensure that **Provider Connector Type Redirect** is enabled.
8. Click **Add** to continue.
9. Click **Save** to save the template.
10. Click **Deploy Template** to deploy it.

Copyright

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706

USA
https://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883