NSX-T Data Center Administration Guide

Modified on 19 MAR 2021

VMware NSX-T Data Center 2.4

You can find the most up-to-date technical documentation on the VMware website at:

https://docs.vmware.com/

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

About Administering VMware NSX-T Data Center

The *NSX-T Data Center Administration Guide* provides information about configuring and managing networking for VMware NSX-T™ Data Center, including how to create logical switches and ports and how to set up networking for tiered logical routers, configure NAT, firewalls, SpoofGuard, grouping and DHCP. It also describes how to configure NSX Cloud.

Intended Audience

This information is intended for anyone who wants to configure NSX-T Data Center. The information is written for experienced Windows or Linux system administrators who are familiar with virtual machine technology, networking, and security operations.

VMware Technical Publications Glossary

VMware Technical Publications provides a glossary of terms that might be unfamiliar to you. For definitions of terms as they are used in VMware technical documentation, go to https://www.vmware.com/topics/glossary.

Overview of the NSX Manager

1

The NSX Manager provides a web-based user interface where you can manage the NSX-T environment. It also hosts the API server that processes API calls.

The NSX Manager web interface provides two methods of configuring resources.

- The Policy interface: the **Networking**, **Security**, **Inventory**, and **Plan & Troubleshoot** tabs.
- The Advanced interface: the **Advanced Networking & Security** tab.

When to Use Policy or Advanced Interfaces

Be consistent about which user interface you use. There are a few reasons to use one user interface over another.

- If you are deploying a new environment with NSX-T Data Center 2.4 or later, using the new policy-based user interface to create and manage your environment is the best choice in most situations.
  - Some features are not available in the policy-based user interface. If you need these features, use the Advanced user interface for all configurations.
- If you are upgrading to NSX-T Data Center 2.4 or later, continue to make configuration changes using the **Advanced Networking & Security** user interface.

Table 1-1. When to Use Policy or Advanced Interfaces

Policy Interface	Advanced Interface
Most new deployments should use the policy-based interface.	Deployments which were created using the advanced interface, for example, upgrades from versions before the policy-based interface was present.
NSX Cloud deployments	Deployments which integrate with other plugins. For example, NSX Container Plug-in, Openstack, and other cloud management platforms.

Table 1-1. When to Use Policy or Advanced Interfaces (continued)

Policy Interface	Advanced Interface
Networking features available in the Policy interface only: ■ DNS Services and DNS Zones ■ VPN ■ Forwarding policies for NSX Cloud	Networking features available in the Advanced interface only: ■ Layer 3 forwarding for IPv4 and IPv6 ■ Forwarding up timer ■ Change internal transit network IP ■ VIP HA support on Tier-0 ■ Standby relocation ■ Route advertisement filtering based on list of prefixes on Tier-1 ■ Loopback creation ■ BGP multihop ■ BGP source addresses ■ Static routes with BFD and interface as next-hop ■ Metadata proxy ■ DHCP server attached to an isolated segment and static binding
Security features available in the Policy interface only: ■ Endpoint Protection ■ Network Introspection (East-West Service Insertion) ■ Context Profiles ■ L7 applications ■ FQDN ■ New Distributed Firewall and Gateway Firewall Layout ■ Categories ■ Auto service rules	Security features available in the Advanced interface only: ■ Ability to enable or disable Distributed Firewall, Identity Firewall, and Gateway Firewall ■ Distributed Firewall session timers ■ Exclusion lists ■ CPU and memory thresholds ■ Sections for stateless rules ■ Bridge Firewall ■ Section Locking ■ Distributed Firewall rule IDs ■ Distributed Firewall rules based on IPs in source and destination

Using the Policy Interface

If you decide to use the policy interface, use it to create all objects. Do not use the advanced interface to create objects.

You can use the advanced interface to modify objects that have been created in the policy interface. The settings for a policy-created object might include a link for **Advanced Configuration**. This link takes you to the advanced interface where you can fine-tune the configuration. You can also view policy-created objects in the advanced interface directly. Settings that are managed by policy but are visible in the advanced interface have this icon next to them: ⊖. You cannot modify them from the advanced user interface.

Where to Find the Policy Interfaces and Advanced Interfaces

The policy-based and advanced interfaces appear in different parts of the NSX Manager user interface, and use different API URIs.

Table 1-2. Policy Interfaces and Advanced Interfaces

Policy Interface	Advanced Interface
■ **Networking** tab ■ **Security** tab ■ **Inventory** tab ■ **Plan & Troubleshoot** tab	**Advanced Networking & Security** tab
API URIs that begin with `/policy/api`	API URIs that begin with `/api`

Note The **System** tab is used for all environments. If you modify Edge nodes, Edge clusters, or transport zones, it can take up to 5 minutes for those changes to be visible on the policy-based user interface. You can synchronize immediately using `POST /policy/api/v1/infra/sites/default/enforcement-points/default?action=reload`.

For more information about using the policy API, see the NSX-T Policy API Getting Started Guide.

Names for Objects Created in the Policy and Advanced Interfaces

The objects you create have different names depending on which interface was used to create them.

Table 1-3. Object Names

Objects Created Using the Policy Interface	Objects Created Using the Advanced Interface
Segment	Logical switch
Tier-1 gateway	Tier-1 logical router
Tier-0 gateway	Tier-0 logical router
Group	NSGroup, IP Sets, MAC Sets
Security Policy	Firewall section
Rule	Firewall rule
Gateway firewall	Edge firewall

Tier-0 Gateways

2

A tier-0 gateway performs the functions of a tier-0 logical router. It processes traffic between the logical and physical networks.

NSX Cloud Note If using NSX Cloud, see How to use NSX-T Data Center Features with the Public Cloud for a list of auto-generated logical entities, supported features, and configurations required for NSX Cloud.

An Edge node can support only one tier-0 gateway or logical router. When you create a tier-0 gateway or logical router, make sure you do not create more tier-0 gateways or logical routers than the number of Edge nodes in the NSX Edge cluster.

Note In the **Advanced Networking & Security** tab, the term tier-0 logical router is used to refer to a tier-0 gateway.

This chapter includes the following topics:

- Add a Tier-0 Gateway
- Create an IP Prefix List
- Create a Community List
- Configure a Static Route
- Create a Route Map
- Configure BGP

Add a Tier-0 Gateway

A tier-0 gateway has downlink connections to tier-1 gateways and uplink connections to physical networks.

You can configure the HA (high availability) mode of a tier-0 gateway to be active-active or active-standby. The following services are only supported in active-standby mode:

- NAT
- Load balancing

- Stateful firewall
- VPN

Tier-0 and tier-1 gateways support the following addressing configurations for all interfaces (uplinks, service ports and downlinks) in both single tier and multi-tiered topologies:

- IPv4 only
- IPv6 only
- Dual Stack - both IPv4 and IPv6

To use IPv6 or dual stack addressing, enable **IPv4 and IPv6** as the L3 Forwarding Mode in **Networking > Networking Settings > Global Networking Config** .

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > Tier-0 Gateways**.
3. Click **Add Tier-0 Gateway**.
4. Enter a name for the gateway.
5. (Required) Select a high-availability mode.

   The default is active-active. In the active-active mode, traffic is load balanced across all members. In active-standby mode, all traffic is processed by an elected active member. If the active member fails, a new member is elected to be active.

   Important After you create the gateway, the HA mode cannot be changed.

6. If the HA mode is active-standby, select a failover mode.

Option	Description
Preemptive	If the preferred node fails and recovers, it will preempt its peer and become the active node. The peer will change its state to standby.
Non-preemptive	If the preferred node fails and recovers, it will check if its peer is the active node. If so, the preferred node will not preempt its peer and will be the standby node.

7. Select an NSX Edge cluster.
8. Click **Save**.
9. To configure route redistribution, click **Route Redistribution** and **Set**.

   Select one or more of the sources:

   - Tier-0 subnets: **Static Routes**, **NAT**, **IPSec Local IP**, **DNS Forwarder IP**, **Service Interface Subnet**, **External Interface Subnet**, **Connected Segment**.
   - Advertised tier-1 subnets: **DNS Forwarder IP**, **Static Routes**, **LB VIP**, **Connected Subnets**, **NAT**, **LB SNAT**.

10 To configure interfaces, click **Interfaces** and **Set**.

   a Click **Add Interface**.

   b Enter a name and an IP address in CIDR format.

   c Select a segment.

   d Select an NSX Edge node.

   e (Optional) Change the MTU value and add tags.

11 Click **Routing** to add IP prefix lists, community lists, static routes, and route maps.

12 Click **BGP** to configure BGP.

13 (Optional) Click **Advanced Configuration** to go to the **Advanced Networking & Security > Routers** page to make additional configurations.

Create an IP Prefix List

An IP prefix list contains single or multiple IP addresses that are assigned access permissions for route advertisement. The IP addresses in this list are processed sequentially. IP prefix lists are referenced through BGP neighbor filters or route maps with in or out direction.

For example, you can add the IP address 192.168.100.3/27 to the IP prefix list and deny the route from being redistributed to the northbound router. You can also append an IP address with less-than-or-equal-to (le) and greater-than-or-equal-to (ge) modifiers to grant or limit route redistribution. For example, 192.168.100.3/27 ge 24 le 30 modifiers match subnet masks greater than or equal to 24-bits and less than or equal to 30-bits in length.

Note The default action for a route is **Deny**. When you create a prefix list to deny or permit specific routes, be sure to create an IP prefix with no specific network address (select **Any** from the dropdown list) and the **Permit** action if you want to permit all other routes.

Prerequisites

Verify that you have a tier-0 gateway configured. See Create a Tier-0 Logical Router.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > Tier-0 Gateways**.

3 To edit a tier-0 gateway, click the menu icon (three dots) and select **Edit**.

4 Click **Routing**.

5 Click **Set** next to **IP Prefix List**.

6 Click **Add IP Prefix List**.

7 Enter a name for the IP prefix list.

8 Click **Set** to add IP prefixes.

9 Click **Add Prefix**.

   a Enter an IP address in CIDR format.

     For example, 192.168.100.3/27.

   b (Optional) Set a range of IP address numbers in the **le** or **ge** modifiers.

     For example, set **le** to 30 and **ge** to 24.

   c Select **Deny** or **Permit** from the drop-down menu.

   d Click **Add**.

10 Repeat the previous step to specify additional prefixes.

11 Click **Save**.

Create a Community List

You can create BGP community lists so that you can configure route maps based on community lists.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > Tier-0 Gateways**.

3 To edit a tier-0 gateway, click the menu icon (three dots) and select **Edit**.

4 Click **Routing**.

5 Click **Set** next to **Community List**.

6 Click **Add Community List**.

7 Enter a name for the community list.

8 Specify a community using the aa:nn format, for example, 300:500, and press Enter. Repeat to add additional communities.

   In addition, you can select one or more of the following:

   - NO_EXPORT_SUBCONFED - Do not advertise to EBGP peers.
   - NO_ADVERTISE - Do not advertise to any peer.
   - NO_EXPORT - Do not advertise outside BGP confederation

9 Click **Save**.

Configure a Static Route

You can configure a static route on the tier-0 gateway to external networks. After you configure a static route, there is no need to advertise the route from tier-0 to tier-1, because tier-1 gateways automatically have a static default route towards their connected tier-0 gateway.

Recursive static routes are supported.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > Tier-0 Gateways**.

3 To edit a tier-0 gateway, click the menu icon (three dots) and select **Edit**.

4 Click **Routing**.

5 Click **Set** next to **Static Routes**.

6 Click **Add Static Route**.

7 Enter a name and network address in CIDR format. Static routes based on IPv6 are supported. IPv6 prefixes can only have an IPv6 next hop.

8 Click **Set Next Hops** to add next-hop information.

9 Click **Add Next Hop**.

10 Enter an IP address.

11 Specify the administrative distance.

12 Select an interface from the dropdown list.

13 Click the **Add** button.

What to do next

Check that the static route is configured properly. See Verify the Static Route.

Create a Route Map

A route map consists of a sequence of IP prefix lists, BGP path attributes, and an associated action. The router scans the sequence for an IP address match. If there is a match, the router performs the action and scans no further.

Route maps can be referenced at the BGP neighbor level and for route redistribution.

Prerequisites

- Verify that an IP prefix list or a community list is configured. See Create an IP Prefix List or Create a Community List.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > Tier-0 Gateways**.

3 To edit a tier-0 gateway, click the menu icon (three dots) and select **Edit**.

4 Click **Routing**.

5 Click **Set** next to **Route Maps**.

6 Click **Add Route Map**.

7 Enter a name and click **Set** to add match criteria.

8 Click **Add Match Criteria** to add one or more match criteria.

9 For each criterion, select **IP Prefix** or **Community List** and click **Set** to specify one or more match expressions.

   a If you selected **Community List**, specify match expressions that define how to match members of community lists. For each community list, the following match options are available:

      - **MATCH ANY** - perform the set action in the route map if any of the communities in the community list is matched.
      - **MATCH ALL** - perform the set action in the route map if all the communities in the community list are matched regardless of the order.
      - **MATCH EXACT** - perform the set action in the route map if all the communities in the community list are matched in the exact same order.
      - **MATCH REGEXP** - perform the set action in the route map if all the communities associated with the NRLI match the regular expression.

   For any match criterion, the match expressions are applied in an AND operation, which means that all match expressions must be satisfied for a match to occur. If there are multiple match criteria, they are applied in an OR operation, which means that a match will occur if any one match criterion is satified.

10 Set BGP attributes.

BGP Attribute	Description
AS-path Prepend	Prepend a path with one or more AS (autonomous system) numbers to make the path longer and therefore less preferred.
MED	Multi-Exit Discriminator indicates to an external peer a preferred path to an AS.
Weight	Set a weight to influence path selection. The range is 0 - 65535.

BGP Attribute	Description
Community	Specify a list of communities using the aa:nn format, for example, 300:500. Or use the drop-down menu to select one of the following: ■ NO_EXPORT_SUBCONFED - Do not advertise to EBGP peers. ■ NO_ADVERTISE - Do not advertise to any peer. ■ NO_EXPORT - Do not advertise outside BGP confederation
Local Preference	Use this value to choose the outbound external BGP path. The path with the highest value is preferred.

11 In the Action column, select **Permit** or **Deny**.

   You can permit or deny IP addresses matched by the IP prefix lists or community lists from being advertised.

12 Click **Save**.

Configure BGP

To enable access between your VMs and the outside world, you can configure an external BGP (eBGP) connection between a tier-0 gateway and a router in your physical infrastructure.

When configuring BGP, you must configure a local Autonomous System (AS) number for the tier-0 gateway. BGP multihop is supported.

BGPv6 is supported for single hop and multihop. A BGPv6 neighbor only supports IPv6 addresses. Redistribution, prefix list, and route maps are supported with IPv6 prefixes.

A tier-0 gateway in active-active mode supports inter-SR (service router) iBGP. If gateway #1 is unable to communicate with a northbound physical router, traffic is re-routed to gateway #2 in the active-active cluster. If gateway #2 is able to communicate with the physical router, traffic between gateway #1 and the physical router will not be affected.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > Tier-0 Gateways**.

3 To edit a tier-0 gateway, click the menu icon (three dots) and select **Edit**.

4 Click **BGP**.

   a Enter the local AS number.

   b Click the **BGP** toggle to enable or disable BGP.

   c If this gateway is in active-active mode, click the **Inter SR iBGP** toggle to enable or disable inter-SR iBGP.

   d Click the **ECMP** toggle button to enable or disable ECMP.

   e. Click the **Multipath Relax** toggle button to enable or disable load-sharing across multiple paths that differ only in AS-path attribute values but have the same AS-path length.

      Note **ECMP** must be enabled for **Multipath Relax** to work.

   f. Click the **Graceful Restart** toggle to enable or disable graceful restart.

      Graceful restart is only supported if the NSX Edge cluster associated with the tier-0 gateway has only one edge node.

5. Configure **Route Aggregation** by adding IP address prefixes.

   a. Click **Add Prefix**.

   b. Enter a IP address prefix in CIDR format.

   c. For the option **Summary Only**, select **Yes** or **No**.

6. Configure **BGP Neighbors**.

   a. Enter the IP address of the neighbor.

   b. Enable or disable BFD.

   c. Enter the remote AS number.

   d. Configure the out filter.

   e. Configure the in filter.

   f. Enable or disable the **Allowas-in** feature.

      This is disabled by default. With this feature enabled, BGP neighbors can receive routes with the same AS, for example, when you have two locations interconnected using the same service provider. This feature applies to all the address families and cannot be applied to specific address families.

   g. Click **Timers & Password**.

   h. Enter a value for **BFD Interval**.

   i. Enter a value for **BFD Multiplier**.

   j. Enter a value for **Hold Down Time**.

   k. Enter a value for **Keep Alive Time**.

   l. Enter a password.

      This is required if you configure MD5 authentication between BGP peers.

7. Click **Save**.

Tier-1 Gateway

3

A tier-1 gateway performs the functions of a tier-1 logical router. It has downlink connections to segments and uplink connections to tier-0 gateways.

Note In the **Advanced Networking & Security** tab, the term tier-1 logical router is used to refer to a tier-1 gateway.

You can configure route advertisements and static routes on a tier-1 gateway. Recursive static routes are supported.

This chapter includes the following topics:

Add a Tier-1 Gateway

A tier-1 gateway is typically connected to a tier-0 gateway in the northbound direction and to segments in the southbound direction.

Tier-0 and tier-1 gateways support the following addressing configurations for all interfaces (uplinks, service ports and downlinks) in both single tier and multi-tiered topologies:

- IPv4 only
- IPv6 only
- Dual Stack - both IPv4 and IPv6

To use IPv6 or dual stack addressing, enable **IPv4 and IPv6** as the L3 Forwarding Mode in **Networking > Networking Settings > Global Networking Config** .

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > Tier-1 Gateways**.
3. Click **Add Tier-1 Gateway**.
4. Enter a name for the gateway.

5. (Optional) Select a tier-0 gateway to connect to this tier-1 gateway to create a multi-tier topology.
6. Select a failover mode.

Option	Description
Preemptive	If the preferred NSX Edgenode fails and recovers, it will preempt its peer and become the active node. The peer will change its state to standby. This is the default option.
Non-preemptive	If the preferred NSX Edge node fails and recovers, it will check if its peer is the active node. If so, the preferred node will not preempt its peer and will be the standby node.

7. (Optional) Select an NSX Edge cluster if you want this tier-1 gateway to host stateful services (NAT, LB, FW).

   If an NSX Edge cluster is selected, a service router will always be created (even if you do not configure stateful services), affecting the north/south traffic pattern.
8. (Optional) Select an NSX Edge node.
9. Click **Save**.
10. (Optional) Click **Route Advertisement**.

    Select one or more of the following:

    - **All Static Routes**
    - **All NAT IP's**
    - **All DNS Forwarder Routes**
    - **All LB VIP Routes**
    - **All Connected Segments and Service Ports**
    - **All LB SNAT IP Routes**
11. (Optional) Click **Service Interfaces** and **Set** to configure connections to segments. Required in some topologies such as VLAN-backed segments or one-arm load balancing.
    a. Click **Add Interface**.
    b. Enter a name and IP address in CIDR format.
    c. Select a segment.
    d. Click **Save**.
12. (Optional) Click **Static Routes** and **Set** to configure static routes.
    a. Click **Add Static Route**.
    b. Enter a name and a network address in the CIDR or IPv6 CIDR format.
    c. Click **Set Next Hops** to add next hop information.
    d. Click **Save**.

Segments

4

A segment performs the functions of a logical switch.

Note In the **Advanced Networking & Security** tab, the term logical switch is used to refer to a segment.

This chapter includes the following topics:

- Segment Profiles
- Add a Segment

Segment Profiles

Segment profiles include Layer 2 networking configuration details for segments and segment ports. NSX Manager supports several types of segment profiles.

The following types of segment profiles are available:

- QoS (Quality of Service)
- IP Discovery
- SpoofGuard
- Segment Security
- MAC Management

Note You cannot edit or delete the default segment profiles. If you require alternate settings from what is in the default segment profile you can create a custom segment profile. By default all custom segment profiles except the segment security profile will inherit the settings of the appropriate default segment profile. For example, a custom IP discovery segment profile by default will have the same settings as the default IP discovery segment profile.

Each default or custom segment profile has a unique identifier. You use this identifier to associate the segment profile to a segment or a segment port.

A segment or segment port can be associated with only one segment profile of each type. You cannot have, for example, two QoS segment profiles associated with a segment or segment port.

If you do not associate a segment profile when you create a segment, then the NSX Manager associates a corresponding default system-defined segment profile. The children segment ports inherit the default system-defined segment profile from the parent segment.

When you create or update a segment or segment port you can choose to associate either a default or a custom segment profile. When the segment profile is associated or disassociated from a segment the segment profile for the children segment ports is applied based on the following criteria.

- If the parent segment has a profile associated with it, the child segment port inherits the segment profile from the parent.
- If the parent segment does not have a segment profile associated with it, a default segment profile is assigned to the segment and the segment port inherits that default segment profile.
- If you explicitly associate a custom profile with a segment port, then this custom profile overrides the existing segment profile.

Note If you have associated a custom segment profile with a segment, but want to retain the default segment profile for one of the child segment port, then you must make a copy of the default segment profile and associate it with the specific segment port.

You cannot delete a custom segment profile if it is associated to a segment or a segment port. You can find out whether any segments and segment ports are associated with the custom segment profile by going to the Assigned To section of the Summary view and clicking on the listed segments and segment ports.

Understanding QoS Segment Profile

QoS provides high-quality and dedicated network performance for preferred traffic that requires high bandwidth. The QoS mechanism does this by prioritizing sufficient bandwidth, controlling latency and jitter, and reducing data loss for preferred packets even when there is a network congestion. This level of network service is provided by using the existing network resources efficiently.

For this release, shaping and traffic marking namely, CoS and DSCP is supported. The Layer 2 Class of Service (CoS) allows you to specify priority for data packets when traffic is buffered in the segment due to congestion. The Layer 3 Differentiated Services Code Point (DSCP) detects packets based on their DSCP values. CoS is always applied to the data packet irrespective of the trusted mode.

NSX-T Data Center trusts the DSCP setting applied by a virtual machine or modifying and setting the DSCP value at the segment level. In each case, the DSCP value is propagated to the outer IP header of encapsulated frames. This enables the external physical network to prioritize the traffic based on the DSCP setting on the external header. When DSCP is in the trusted mode, the DSCP value is copied from the inner header. When in the untrusted mode, the DSCP value is not preserved for the inner header.

Note DSCP settings work only on tunneled traffic. These settings do not apply to traffic inside the same hypervisor.

You can use the QoS switching profile to configure the average ingress and egress bandwidth values to set the transmit limit rate. The peak bandwidth rate is used to support burst traffic a segment is allowed to prevent congestion on the northbound network links. These settings do not guarantee the bandwidth but help limit the use of network bandwidth. The actual bandwidth you will observe is determined by the link speed of the port or the values in the switching profile, whichever is lower.

The QoS switching profile settings are applied to the segment and inherited by the child segment port.

Create a QoS Segment Profile

You can define the DSCP value and configure the ingress and egress settings to create a custom QoS switching profile.

Prerequisites

- Familiarize yourself with the QoS switching profile concept. See Understanding QoS Switching Profile.
- Identify the network traffic you want to prioritize.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > Segments > Segment Profiles**.
3. Click **Add Segment Profile** and select **QoS**.

4 Complete the QoS switching profile details.

Option	Description
Name	Name of the profile.
Mode	Select either a **Trusted** or **Untrusted** option from the Mode drop-down menu. When you select the Trusted mode the inner header DSCP value is applied to the outer IP header for IP/IPv6 traffic. For non IP/IPv6 traffic, the outer IP header takes the default value. Trusted mode is supported on an overlay-based logical port. The default value is 0. Untrusted mode is supported on overlay-based and VLAN-based logical port. For the overlay-based logical port, the DSCP value of the outbound IP header is set to the configured value irrespective to the inner packet type for the logical port. For the VLAN-based logical port, the DSCP value of IP/IPv6 packet will be set to the configured value. The DSCP values range for untrusted mode is between 0 to 63. **Note** DSCP settings work only on tunneled traffic. These settings do not apply to traffic inside the same hypervisor.
Priority	Set the CoS priority value. The priority values range from 0 to 63, where 0 has the highest priority.
Class of Service	Set the CoS value. CoS is supported on VLAN-based logical port. CoS groups similar types of traffic in the network and each type of traffic is treated as a class with its own level of service priority. The lower priority traffic is slowed down or in some cases dropped to provide better throughput for higher priority traffic. CoS can also be configured for the VLAN ID with zero packet. The CoS values range from 0 to 7, where 0 is the best effort service.
Ingress	Set custom values for the outbound network traffic from the VM to the logical network. You can use the average bandwidth to reduce network congestion. The peak bandwidth rate is used to support burst traffic and the burst size is based on the duration with peak bandwidth. You set burst duration in the burst size setting. You cannot guarantee the bandwidth. However, you can use the Average, Peak, and Burst Size settings to limit network bandwidth. For example, if the average bandwidth is 30 Mbps, peak bandwidth is 60 Mbps, and the allowed duration is 0.1 second, then the burst size is 60 * 1000000 * 0.10/8 = 750000 Bytes. The default value 0 disables rate limiting on the ingress traffic.
Ingress Broadcast	Set custom values for the outbound network traffic from the VM to the logical network based on broadcast. For example, when you set the average bandwidth for a logical switch to 3000 Kbps, peak bandwidth is 6000 Kbps, and the allowed duration is 0.1 second, then the burst size is 6000 * 1000 * 0.10/8 = 75000 Bytes. The default value 0 disables rate limiting on the ingress broadcast traffic.
Egress	Set custom values for the inbound network traffic from the logical network to the VM. The default value 0 disables rate limiting on the egress traffic.

If the ingress, ingress broadcast, and egress options are not configured, the default values are used.

5 Click **Save**.

Understanding IP Discovery Segment Profile

IP Discovery uses DHCP and DHCPv6 snooping, ARP (Address Resolution Protocol) snooping, ND (Neighbor Discovery) snooping, and VM Tools to learn MAC and IP addresses.

The discovered MAC and IP addresses are used to achieve ARP/ND suppression, which minimizes traffic between VMs connected to the same segment. The addresses are also used by the SpoofGuard and distributed firewall (DFW) components. DFW uses the address bindings to determine the IP address of objects in firewall rules.

DHCP/DHCPv6 snooping inspects the DHCP/DHCPv6 packets exchanged between the DHCP/DHCPv6 client and server to learn the IP and MAC addresses.

ARP snooping inspects the outgoing ARP and GARP (gratuitous ARP) packets of a VM to learn the IP and MAC addresses.

VM Tools is software that runs on an ESXi-hosted VM and can provide the VM's configuration information including MAC and IP or IPv6 addresses. This IP discovery method is available for VMs running on ESXi hosts only.

ND snooping is the IPv6 equivalent of ARP snooping. It inspects neighbor solicitation (NS) and neighbor advertisement (NA) messages to learn the IP and MAC addresses.

Duplicate address detection checks whether a newly discovered IP address is already present on the realized binding list for a different port. This check is performed for ports on the same segment. If a duplicate address is detected, the newly discovered address is added to the discovered list, but is not added to the realized binding list. All duplicate IPs have an associated discovery timestamp. If the IP that is on the realized binding list is removed, either by adding it to the ignore binding list (see below) or by disabling snooping, the duplicate IP with the oldest timestamp is moved to the realized binding list. The duplicate address information is available through an API call.

By default, the discovery methods ARP snooping and ND snooping operate in a mode called trust on first use (TOFU). In TOFU mode, when an address is discovered and added to the realized bindings list, that binding remains in the realized list forever. TOFU applies to the first 'n' unique <IP, MAC, VLAN> bindings discovered using ARP/ND snooping, where 'n' is the binding limit that you can configure. You can disable TOFU for ARP/ND snooping. The methods will then operate in trust on every use (TOEU) mode. In TOEU mode, when an address is discovered, it is added to the realized bindings list and when it is deleted or expired, it is removed from the realized bindings list. The methods DHCP snooping and VM Tools always operate in TOEU mode.

Note TOFU does not mean SpoofGuard and it does not block traffic like SpoofGuard does. For more information about SpoofGuard, see Understanding SpoofGuard Segment Profile.

For each port, NSX Manager maintains maintains an ignore bindings list, which contains IP addresses that that cannot be bound bound to the port. You can only update this list using the API. You can also use this method By navigating to delete a previously discovered IP for a given port. For more information, see the NSX-T API Reference and search for ignore_address_bindings

Note For Linux VMs, the ARP flux problem might cause ARP snooping to obtain incorrect information. The problem can be prevented with an ARP filter. For more information, see http://linux-ip.net/html/ether-arp.html#ether-arp-flux.

Create an IP Discovery Segment Profile

NSX-T Data Center has several default IP Discovery switching profiles. You can also create additional ones.

Prerequisites

Familiarize yourself with the IP Discovery switching profile concepts. See Understanding IP Discovery Switching Profile

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > Segments > Segment Profiles**.

3 Click **Add Segment Profile** and select **IP Discovery**.

4 Specify the IP Discovery switching profile details.

Option	Description
Name	Enter a name.
ARP Snooping	For an IPv4 environment. Applicable if VMs have static IP addresses.
ARP Binding Limit	The maximum number of IPv4 IP addresses that can be bound to a port.
ARP ND Binding Limit Timeout	The timeout value, in minutes, for IP addresses in the ARP/ND binding table if TOFU is disabled. If an address times out, a newly discovered address will replace it.
DHCP Snooping	For an IPv4 environment. Applicable if VMs have IPv4 addresses.
DHCP V6 Snooping	For an IPv6 environment. Applicable if VMs have IPv6 addresses.
VM Tools	Available for ESXi-hosted VMs only.
VM Tools for IPv6	Available for ESXi-hosted VMs only.
Neighbor Discovery Snooping	For an IPv6 environment. Applicable if VMs have static IP addresses.
Neighbor Discovery Binding Limit	The maximum number of IPv6 addresses that can be bound to a port.
Trust on First use	Applicable to ARP and ND snooping.
Duplicate IP Detection	For all snooping methods and both IPv4 and IPv6 environments.

5 Click **Save**.

Understanding SpoofGuard Segment Profile

SpoofGuard helps prevent a form of malicious attack called "web spoofing" or "phishing." A SpoofGuard policy blocks traffic determined to be spoofed.

SpoofGuard is a tool that is designed to prevent virtual machines in your environment from sending traffic with an IP address it is not authorized to end traffic from. In the instance that a virtual machine’s IP address does not match the IP address on the corresponding logical port and segment address binding in SpoofGuard, the virtual machine’s vNIC is prevented from accessing the network entirely. SpoofGuard can be configured at the port or segment level. There are several reasons SpoofGuard might be used in your environment:

- Preventing a rogue virtual machine from assuming the IP address of an existing VM.
- Ensuring the IP addresses of virtual machines cannot be altered without intervention – in some environments, it’s preferable that virtual machines cannot alter their IP addresses without proper change control review. SpoofGuard facilitates this by ensuring that the virtual machine owner cannot simply alter the IP address and continue working unimpeded.
- Guaranteeing that distributed firewall (DFW) rules will not be inadvertently (or deliberately) bypassed – for DFW rules created utilizing IP sets as sources or destinations, the possibility always exists that a virtual machine could have it’s IP address forged in the packet header, thereby bypassing the rules in question.

NSX-T Data Center SpoofGuard configuration covers the following:

- MAC SpoofGuard - authenticates MAC address of packet
- IP SpoofGuard - authenticates MAC and IP addresses of packet
- Dynamic Address Resolution Protocol (ARP) inspection, that is, ARP and Gratuitous Address Resolution Protocol (GARP) SpoofGuard and Neighbor Discovery (ND) SpoofGuard validation are all against the MAC source, IP Source and IP-MAC source mapping in the ARP/GARP/ND payload.

At the port level, the allowed MAC/VLAN/IP whitelist is provided through the Address Bindings property of the port. When the virtual machine sends traffic, it is dropped if its IP/MAC/VLAN does not match the IP/MAC/VLAN properties of the port. The port level SpoofGuard deals with traffic authentication, i.e. is the traffic consistent with VIF configuration.

At the segment level, the allowed MAC/VLAN/IP whitelist is provided through the Address Bindings property of the segment. This is typically an allowed IP range/subnet for the segment and the segment level SpoofGuard deals with traffic authorization.

Traffic must be permitted by port level AND segment level SpoofGuard before it will be allowed into segment. Enabling or disabling port and segment level SpoofGuard, can be controlled using the SpoofGuard segment profile.

Create a SpoofGuard Segment Profile

When SpoofGuard is configured, if the IP address of a virtual machine changes, traffic from the virtual machine may be blocked until the corresponding configured port/segment address bindings are updated with the new IP address.

Enable SpoofGuard for the port group(s) containing the guests. When enabled for each network adapter, SpoofGuard inspects packets for the prescribed MAC and its corresponding IP address.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking** > **Segments** > **Segment Profiles**.
3. Click **Add Segment Profile** and select **Spoof Guard**.
4. Enter a name.
5. To enable port level SpoofGuard, set **Port Bindings** to **Enabled**.
6. Click **Save**.

Understanding Segment Security Segment Profile

Segment security provides stateless Layer2 and Layer 3 security by checking the ingress traffic to the segment and dropping unauthorized packets sent from VMs by matching the IP address, MAC address, and protocols to a set of allowed addresses and protocols. You can use segment security to protect the segment integrity by filtering out malicious attacks from the VMs in the network.

You can configure the Bridge Protocol Data Unit (BPDU) filter, DHCP Snooping, DHCP server block, and rate limiting options to customize the segment security segment profile on a segment.

Create a Segment Security Segment Profile

You can create a custom segment security segment profile with MAC destination addresses from the allowed BPDU list and configure rate limiting.

Prerequisites

Familiarize yourself with the segment security segment profile concept. See Understanding Switch Security Switching Profile.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking** > **Segments** > **Segment Profiles**.
3. Click **Add Segment Profile** and select **Segment Security**.

4 Complete the segment security profile details.

Option	Description
Name	Name of the profile.
BPDU Filter	Toggle the **BPDU Filter** button to enable BPDU filtering. Disabled by default. When the BPDU filter is enabled, all of the traffic to BPDU destination MAC address is blocked. The BPDU filter when enabled also disables STP on the logical switch ports because these ports are not expected to take part in STP.
BPDU Filter Allow List	Click the destination MAC address from the BPDU destination MAC addresses list to allow traffic to the permitted destination. You must enable **BPDU Filter** to be able to select from this list.
DHCP Filter	Toggle the **Server Block** button and **Client Block** button to enable DHCP filtering. Both are disabled by default. DHCP Server Block blocks traffic from a DHCP server to a DHCP client. Note that it does not block traffic from a DHCP server to a DHCP relay agent. DHCP Client Block prevents a VM from acquiring a DHCP IP address by blocking DHCP requests.
DHCPv6 Filter	Toggle the **V6 Server Block** button and **V6 Client Block** button to enable DHCP filtering. Both are disabled by default. DHCPv6 Server Block blocks traffic from a DHCPv6 server to a DHCPv6 client. Note that it does not block traffic from a DHCP server to a DHCP relay agent. Packets whose UDP source port number is 547 are filtered. DHCPv6 Client Block prevents a VM from acquiring a DHCP IP address by blocking DHCP requests. Packets whose UDP source port number is 546 are filtered.
Block Non-IP Traffic	Toggle the **Block Non-IP Traffic** button to allow only IPv4, IPv6, ARP, and BPDU traffic. The rest of the non-IP traffic is blocked. The permitted IPv4, IPv6, ARP, GARP and BPDU traffic is based on other policies set in address binding and SpoofGuard configuration. By default, this option is disabled to allow non-IP traffic to be handled as regular traffic.
RA Guard	Toggle the **RA Guard** button to filter out ingress IPv6 router advertisements. ICMPv6 type 134 packets are filtered out. This option is enabled by default.
Rate Limits	Set a rate limit for broadcast and multicast traffic. This option is enabled by default. Rate limits can be used to protect the logical switch or VMs from events such as broadcast storms. To avoid any connectivity problems, the minimum rate limit value must be >= 10 pps.

5 Click **Save**.

Understanding MAC Discovery Segment Profile

The MAC management segment profile supports two functionalities: MAC learning and MAC address change.

The MAC address change feature allows a VM to change its MAC address. A VM connected to a port can run an administrative command to change the MAC address of its vNIC and still send and receive traffic on that vNIC. This feature is supported on ESXi only and not on KVM. This property is disabled by default.

MAC learning provides network connectivity to deployments where multiple MAC addresses are configured behind one vNIC, for example, in a nested hypervisor deployment where an ESXi VM runs on an ESXi host and multiple VMs run inside the ESXi VM. Without MAC learning, when the ESXi VM's vNIC connects to a segment port, its MAC address is static. VMs running inside the ESXi VM do not have network connectivity because their packets have different source MAC addresses. With MAC learning, the vSwitch inspects the source MAC address of every packet coming from the vNIC, learns the MAC address and allows the packet to go through. If a MAC address that is learned is not used for a certain period of time, it is removed. This time period is not confurable. The field **MAC Learning Aging Time** displays the pre-defined value, which is 600.

MAC learning also supports unknown unicast flooding. Normally, when a packet that is received by a port has an unknown destination MAC address, the packet is dropped. With unknown unicast flooding enabled, the port floods unknown unicast traffic to every port on the switch that has MAC learning and unknown unicast flooding enabled. This property is enabled by default, but only if MAC learning is enabled.

The number of MAC addresses that can be learned is configurable. The maximum value is 4096, which is the default. You can also set the policy for when the limit is reached. The options are:

- **Drop** - Packets from an unknown source MAC address are dropped. Packets inbound to this MAC address will be treated as unknown unicast. The port will receive the packets only if it has unknown unicast flooding enabled.
- **Allow** - Packets from an unknown source MAC address are forwarded although the address will not be learned. Packets inbound to this MAC address will be treated as unknown unicast. The port will receive the packets only if it has unknown unicast flooding enabled.

If you enable MAC learning or MAC address change, to improve security, configure SpoofGuard as well.

Create a MAC Discovery Segment Profile

You can create a MAC discovery segment profile to manage MAC addresses.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > Segments > Segment Profiles**.
3. Click **Add Segment Profile** and select **MAC Discovery**.

4 Complete the MAC discovery profile details.

Option	Description
Name	Name of the profile.
MAC Change	Enable or disable the MAC address change feature. The default is disabled.
MAC Learning	Enable or disable the MAC learning feature. The default is disabled.
MAC Limit Policy	Select **Allow** or **Drop**. The default is **Allow**. This option is available if you enable MAC learning
Unknown Unicast Flooding	Enable or disable the unknown unicast flooding feature. The default is enabled. This option is available if you enable MAC learning
MAC Limit	Set the maximum number of MAC addresses. The default is 4096. This option is available if you enable MAC learning
MAC Learning Aging Time	For information only. This option is not configurable. The pre-defined value is 600.

5 Click **Save**.

Add a Segment

A segment connects to gateways and VMs. A segment performs the functions of a logical switch.

For information on how to find the VIF ID of a VM, see Connecting a VM to a Logical Switch.

Note An N-VDS switch configured in the Enhanced Datapath mode supports IP Discovery, SpoofGuard, and IPFIX profiles.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > Segments**.

3 Click **Add**.

4 Enter a name for the segment.

5 Select an uplink.

You can select an existing tier-0 or tier-1 gateway, or select **None**. If you select **None**, the segment is simply a logical switch. With a subnet configured, it can link to a tier-0 or tier-1 gateway.

6 If the uplink is a tier-1 gateway, select a type, **Flexible** or **Fixed**.

A flexible segment can be unlinked from gateways. A fixed segment can be deleted but not unlinked from a gateway.

7 Click **Set Subnets** to specify a subnet.

8 Select a transport zone.

9 If the transport zone is of type VLAN, specify a list of VLAN IDs.

10 Click **Save**.

11 Click **Ports** and **Set** to add segment ports.

    a Click **Add Segment Port**.

    b Enter a port name.

    c For **ID**, enter the VIF UUID of the VM or server that connects to this port.

    d Select a type: **Parent**, **Child**, or **Independent**.

    Leave this field blank except for use cases such as containers or VMware HCX. If this port is for a container in a VM, select **Child**. If this port is for a container host VM, select **Parent**. If this port is for a bare metal container or server, select **Independent**.

    e Enter a context ID.

    Enter the parent VIF ID if **Type** is **Child**, or transport node ID if **Type** is **Independent**.

    f Enter a traffic tag.

    Enter the VLAN ID in container and other use cases.

    g Select an address allocation method: **IP Pool**, **MAC Pool**, **Both**, or **None**.

    h Specify tags.

    i Select segment profiles for this port.

12 Click **Segment Profiles** to select segment profiles.

13 Click **Save**.

Virtual Private Network (VPN)

5

NSX-T Data Center supports IPSec Virtual Private Network (IPSec VPN) and Layer 2 VPN (L2 VPN) on an NSX Edge node. IPSec VPN offers site-to-site connectivity between an NSX Edge node and remote sites. With L2 VPN, you can extend your data center by allowing virtual machines to keep their network connectivity across geographical boundaries while using the same IP address.

Note IPSec VPN and L2 VPN are not supported in the NSX-T Data Center limited export release.

You must have a working NSX Edge node, with at least one configured Tier-0 gateway, before you can configure a VPN service. For more information, see "NSX Edge Installation" in the *NSX-T Data Center Installation Guide*.

Beginning with NSX-T Data Center 2.4, you can also configure new VPN services using the NSX Manager user interface. In earlier releases of NSX-T Data Center, you can only configure VPN services using REST API calls.

Important When using NSX-T Data Center 2.4 or later to configure VPN services, you must use new objects, such as Tier-0 gateways, that were created using the NSX Manager UI or Policy APIs that are included with NSX-T Data Center 2.4 or later release. To use existing Tier-0 logical routers that were configured before the NSX-T Data Center 2.4 release, you must continue to use API calls to configure a VPN service.

System-default configuration profiles with predefined values and settings are made available for your use during a VPN service configuration. You can also define new profiles with different settings and select them during the VPN service configuration.

This chapter includes the following topics:

- Check the Realized State of an IPSec VPN Session
- Monitor and Troubleshoot VPN Sessions

Understanding IPSec VPN

Internet Protocol Security (IPSec) VPN secures traffic flowing between two networks connected over a public network through IPSec gateways called endpoints. NSX Edge supports site-to-site IPSec VPN between an NSX Edge node and remote sites.

IPSec VPN secures traffic flowing between two networks connected over a public network through IPSec gateways called endpoints. NSX Edge only supports a tunnel mode that uses IP tunneling with Encapsulating Security Payload (ESP). ESP operates directly on top of IP, using IP protocol number 50.

IPSec VPN uses the IKE protocol to negotiate security parameters. The default UDP port is set to 500. If NAT is detected in the gateway, the port is set to UDP 4500.

In NSX-T Data Center, IPSec VPN services are only supported on Tier-0 gateways that must be in `Active-Standby` high-availability mode. See Add a Tier-0 Gateway for information. You can use segments that are connected to either Tier-0 or Tier-1 gateways when configuring an IPSec VPN service.

IPsec VPN service in NSX-T Data Center uses the gateway-level failover functionality to support high-availability. Tunnels are re-established on failover and VPN configuration data is synchronized. The IPSec VPN state is not synchronized as tunnels are re-established.

Pre-shared key mode authentication and IP unicast traffic are supported between the NSX Edge node and remote VPN sites. In addition, certificate authentication is supported beginning with NSX-T Data Center 2.4. Only certificate types signed by one of the following signature hash algorithms are supported.

- SHA256withRSA
- SHA384withRSA
- SHA512withRSA

NSX Edge supports two types of IPSec VPN: policy-based IPSec VPN and route-based IPSec VPN.

Using Policy-Based IPSec VPN

Policy-based IPSec VPN requires a VPN policy to be applied to packets to determine which traffic is to be protected by IPSec before being passed through the VPN tunnel.

This type of VPN is considered static because when a local network topology and configuration change, the VPN policy settings must also be updated to accommodate the changes.

When using a policy-based IPSec VPN with NSX-T Data Center, you use IPSec tunnels to connect one or more local subnets behind the NSX Edge node with the peer subnets on the remote VPN site.

You can deploy an NSX Edge node behind a NAT device. In this deployment, the NAT device translates the VPN address of an NSX Edge node to a publicly accessible address facing the Internet. Remote VPN sites use this public address to access the NSX Edge node.

You can place remote VPN sites behind a NAT device as well. You must provide the remote VPN site's public IP address and its ID (either FQDN or IP address) to set up the IPSec tunnel. On both ends, static one-to-one NAT is required for the VPN address.

The size of the NSX Edge node determines the maximum number of supported tunnels, as shown in the following table.

Table 5-1. Number of IPSec Tunnels Supported

Edge Node Size	# of IPSec Tunnels Per VPN Session (Policy-Based)	# of Sessions Per VPN Service	# of IPSec Tunnels Per VPN Service (16 tunnels per session)
Small	N/A (POC/Lab Only)	N/A (POC/Lab Only)	N/A (POC/Lab Only)
Medium	128	128	2048
Large	128 (soft limit)	256	4096
Bare Metal	128 (soft limit)	512	6000

Restriction The inherent architecture of policy-based IPSec VPN restricts you from setting up a VPN tunnel redundancy.

For information about configuring a policy-based IPSec VPN, see Add an IPSec VPN Service.

Using Route-Based IPSec VPN

Route-based IPSec VPN provides tunneling on traffic based on the routes learned dynamically over a special interface called virtual tunnel interface (VTI) using, for example, BGP as the protocol. IPSec secures all the traffic flowing through the VTI.

Route-based IPSec VPN is similar to Generic Routing Encapsulation (GRE) over IPSec, with the exception that no additional encapsulation is added to the packet before applying IPSec processing.

In this VPN tunneling approach, VTIs are created on the NSX Edge node. Each VTI is associated with an IPSec tunnel. The encrypted traffic is routed from one site to another site through the VTI interfaces. IPSec processing happens only at the VTI.

VPN Tunnel Redundancy

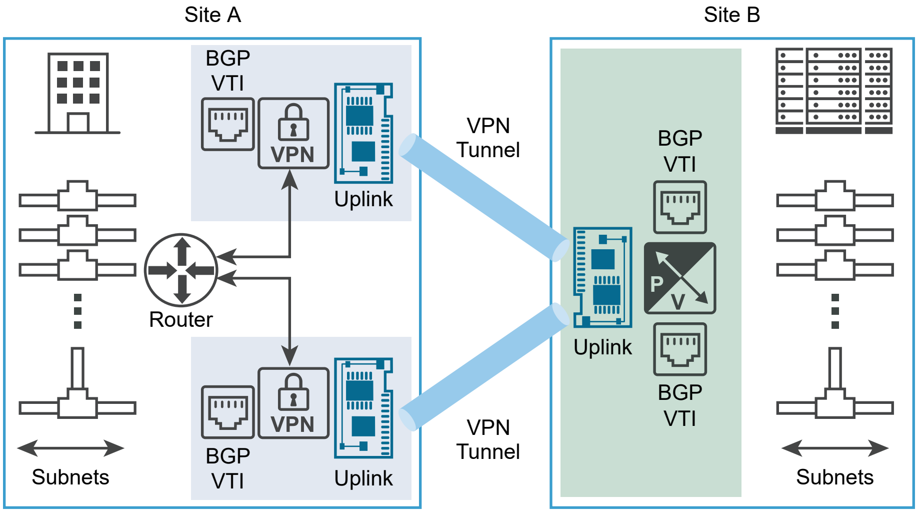
You can configure VPN tunnel redundancy with a route-based IPSec VPN service. Tunnel redundancy provides an uninterrupted data path connectivity between the two sites when the ISP link fails, or when the remote VPN gateway fails.

Important

- In NSX-T Data Center, IPSec VPN tunnel redundancy is supported only using BGP. OSPF dynamic routing is not supported for routing through IPSec VPN tunnels.
- Do not use static routing for route-based IPSec VPN tunnels to achieve VPN tunnel redundancy.

The following figure shows a logical representation of IPSec VPN tunnel redundancy between two sites. In this figure, Site A and Site B represent two data centers. For this example, assume that NSX-T Data Center is not managing the Edge VPN Gateways in Site A, and that NSX-T Data Center is managing an Edge Gateway virtual appliance in Site B.

Figure 5-1. Tunnel Redundancy in Route-Based IPSec VPN



As shown in the figure, you can configure two independent IPSec VPN tunnels by using VTIs. Dynamic routing is configured using BGP protocol to achieve tunnel redundancy. If both IPSec VPN tunnels are available, they remain in service. All the traffic destined from Site A to Site B through the NSX Edge node is routed through the VTI. The data traffic undergoes IPSec processing and goes out of its associated NSX Edge node uplink interface. All the incoming IPSec traffic received from Site B VPN Gateway on the NSX Edge node uplink interface is forwarded to the VTI after decryption, and then usual routing takes place.

You must configure BGP HoldDown timer and KeepAlive timer values to detect loss of connectivity with peer within the required failover time. See Configure BGP.

For information about configuring a policy-based IPSec VPN, see Add an IPSec VPN Service.

Understanding Layer 2 VPN

With Layer 2 VPN (L2 VPN), you can extend Layer 2 networks (VLANs or VNIs) across multiple sites on the same broadcast domain. Virtual machines (VMs) in the Layer 2 can seamlessly communicate with each other over L2 VPN even if they are located across data centers.

With an L2 VPN connectivity, Layer 2 networks can be extended from an on-premise data center to the cloud such as, VMware Cloud on Amazon (VMC). This connection is secured with a route-based IPSec tunnel between the L2 VPN client and the L2 VPN server.

Each L2 VPN session has one Generic Routing Encapsulation (GRE) tunnel. Tunnel redundancy is not supported. An L2 VPN session can extend up to 4094 Layer 2 networks.

NSX-T Data Center L2 VPN services are supported only on Tier-0 gateways. Segments can be connected to either Tier-0 or Tier-1 gateways and use L2 VPN services.

Note This L2 VPN feature is available only for NSX-T Data Center and does not have any third-party interoperability.

The L2 VPN service support is provided in the following scenarios.

- Between an NSX-T Data Center L2 VPN server and an L2 VPN client hosted on an NSX Edge managed in an NSX Data Center for vSphere. A managed L2 VPN client is limited to supporting VNIs.
- Between an NSX-T Data Center L2 VPN server and an L2 VPN client hosted on a standalone or unmanaged NSX Edge. An unmanaged L2 VPN client supports VLANs .
- Beginning with NSX-T Data Center 2.4 release, L2 VPN service support is available between an NSX-T Data Center L2 VPN server and NSX-T Data Center L2 VPN clients. In this scenario, you can extend the logical L2 segments between two on-premises software-defined data centers (SDDCs).

The extended network is a single subnet with a single broadcast domain, so virtual machines (VMs) remain on the same subnet when they are moved between network sites and their IP addresses remain the same.

You can migrate workloads between different physical sites and their IP addresses do not change. The workloads can run on either VXLAN-based or VLAN-based networks. For cloud providers, L2 VPN provides a mechanism to onboard tenants without modifying existing IP addresses used by their workloads and applications.

In addition to supporting data center migration, an on-premise network extended with an L2 VPN is useful for a disaster recovery plan and dynamically engaging off-premise compute resources to meet the increased demand.

Adding VPN Services

You can add either an IPSec VPN (policy-based or route-based) or an L2 VPN using the NSX Manager user interface (UI).

The following sections provide high-level information of the workflows required to set up the VPN service that you need. The topics that follow these sections provide details on how to add either an IPSec VPN or an L2 VPN using the NSX Manager user interface.

Policy-Based IPSec VPN Configuration Workflow

Configuring a policy-based IPSec VPN service workflow requires the following high-level steps.

1. Create and enable an IPSec VPN service using an existing Tier-0 gateway. See Add an IPSec VPN Service.
2. Create a DPD (dead peer detection) profile, if you prefer not to use the system default. See Add DPD Profiles.
3. To use a non-system default IKE profile, define an IKE (Internet Key Exchange) profile . See Add IKE Profiles.
4. Configure an IPSec profile using Add IPSec Profiles.
5. Use Add Local Endpoints to create a local endpoint.
6. Configure a policy-based IPSec VPN session, apply the profiles, and attach the local endpoint to it. See Add a Policy-Based IPSec Session.

Route-Based IPSec VPN Configuration Workflow

A route-based IPSec VPN configuration workflow requires the following high-level steps.

1. Configure and enable an IPSec VPN service using an existing Tier-0 gateway. See Add an IPSec VPN Service.
2. Specify the local and peer subnets to be used for the tunnel.
3. Create a DPD profile. See Add DPD Profiles.
4. Define an IKE profile if you prefer not to use the default IKE profile. See Add IKE Profiles.
5. If you decide not to use the system default IPSec profile, create one using Add IPSec Profiles.
6. Add a local endpoint using Add Local Endpoints.
7. Create a route-based IPSec VPN session. See Add a Route-Based IPSec Session.

L2 VPN Configuration Workflow

Configuring an L2 VPN requires that you configure an L2 VPN service in Server mode and then another L2 VPN service in Client mode. You also must configure the sessions for the L2 VPN server and L2 VPN client. Following is a high-level workflow for configuring an L2 VPN service.

1. Create an L2 VPN Service in Server mode.
    a. Configure a route-based IPSec VPN tunnel with a Tier-0 gateway and an L2 VPN Server service using that route-based IPSec tunnel. See Add an L2 VPN Server Service.

   b Configure an L2 VPN server session, which binds the newly created route-based IPSec VPN service and the L2 VPN server service, and automatically allocates the GRE IP addresses. See Add an L2 VPN Server Session.

   c Add segments to the L2 VPN Server sessions. This step is also described in Add an L2 VPN Server Session.

   d Use Download the Remote Side L2 VPN Configuration to obtain the peer code for the L2 VPN Server service session, which is used to configure the L2 VPN Client session automatically.

2 Create an L2 VPN Service in Client mode.

   a Configure another route-based IPSec VPN service using a different Tier-0 gateway and configure an L2 VPN Client service using that Tier-0 gateway that you just configured. See Add an L2 VPN Client Service for information.

   b Define the L2 VPN Client sessions by importing the peer code generated by the L2 VPN Server service. See Add an L2 VPN Client Session.

   c Add segments to the L2 VPN Client sessions defined in the previous step. This step is described in Add an L2 VPN Client Session.

Add an IPSec VPN Service

NSX-T Data Center supports a site-to-site IPSec VPN service between a Tier-0 gateway and remote sites. You can create a policy-based or a route-based IPSec VPN service. You must create the IPSec VPN service first before you can configure either a policy-based or a route-based IPSec VPN session.

Note IPSec VPN is not supported in the NSX-T Data Center limited export release.

IPSec VPN is not supported when the local endpoint IP address goes through NAT in the same logical router that the IPSec VPN session is configured.

Prerequisites

- Familiarize yourself with the IPSec VPN. See Understanding IPSec VPN.
- You must have at least one Tier-0 gateway configured and available for use. See Add a Tier-0 Gateway for more information.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Navigate to **Networking > VPN > VPN Services**.

3 Select **Add Service > IPSec**.

4 Enter a name for the IPSec service.

   This name is required.

5 From the **Tier-0 Gateway** drop-down menu, select the Tier-0 gateway to associate with this IPSec VPN service.

6 Enable or disable **Admin Status**.

   By default, the value is set to `Enabled`, which means the IPSec VPN service is enabled on the Tier-0 gateway after the new IPSec VPN service is configured.

7 Set the value for **IKE Log Level**.

   The Internet Key Exchange (IKE) logging level determines the amount of information you want collected for the IPSec VPN traffic. The default is set to the `Info` level.

8 Enter a value for **Tags** if you want to include this service in a tag group.

9 Click **Global Bypass Rules** if you want to allow data packets to be exchanged between the specified local and remote IP addresses without any IPSec protection, even if the IP addresses are specified in the IPSec session rules. In **Local Networks** and **Remote Networks**, enter the list of local and remote subnets between which the bypass rules are applied.

   The default is to use the IPSec protection when data is exchanged between local and remote sites. These rules apply for all IPSec VPN sessions created within this IPSec VPN service.

10 Click **Save**.

   After the new IPSec VPN service is created successfully, you are asked whether you want to continue with the rest of the IPSec VPN configuration. If you click **Yes**, you are taken back to the Add IPSec VPN Service panel. The **Session** link is now enabled and you can click it to add an IPSec VPN session.

What to do next

Use information in Adding IPSec VPN Sessions to guide you in adding an IPSec VPN session. You also provide information for the profiles and local endpoint that are required to finish the IPSec VPN configuration.

Add an L2 VPN Service

You can configure an L2 VPN service over an IPSec tunnel by creating a route-based IPSec VPN tunnel first. You then configure an L2 VPN tunnel between an L2 VPN server (destination gateway) and an L2 VPN client (source gateway) by consuming the route-based IPSec VPN tunnel.

To configure an L2 VPN service over an IPSec tunnel, use the information in the topics that follow in this section.

Prerequisites

- Familiarize yourself with IPsec VPN and L2 VPN. See Understanding IPSec VPN and Understanding Layer 2 VPN.

- You must have at least one Tier-0 gateway configured and available for use. See Add a Tier-0 Gateway.

Procedure

1. Add an L2 VPN Server Service

   To configure an L2 VPN Server service, you must configure the L2 VPN service in server mode on the destination NSX Edge to which the L2 VPN client is to be connected.

2. Add an L2 VPN Client Service

   After you configure the L2 VPN server, configure the L2 VPN service in the client mode on another Edge instance, which can be either an NSX-managed Edge, a standalone Edge, or an NSX-T software-defined data center (SDDC).

Add an L2 VPN Server Service

To configure an L2 VPN Server service, you must configure the L2 VPN service in server mode on the destination NSX Edge to which the L2 VPN client is to be connected.

Before configuring an L2 VPN server, you must first create a route-based IPSec VPN tunnel. You then use this route-based IPSec VPN tunnel to create an L2 VPN tunnel that stretches your Layer 2 networks between two sites.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Create a route-based IPSec tunnel with the NSX Edge you want to configure as the L2 VPN server mode.
   a. Navigate to the **Networking > VPN > VPN Services** tab and select **Add Service > IPSec**.
   b. Enter a name for the IPSec VPN service.
   c. From the **Tier-0 Gateway** drop-down menu, select a Tier-0 gateway to use with the L2 VPN server.
   d. If you want to use values different from the system defaults, set the rest of the properties on the Add IPSec Service pane, as needed.
   e. Click **Save** and when prompted if you want to continue configuring the IPSec VPN service, select **No**.
3. Navigate to the **Networking > VPN > VPN Services** tab and select **Add Service > L2 VPN Server** to create an L2 VPN server.
4. Enter a name for the L2 VPN server.
5. From the **Tier-0 Gateway** drop-down menu, select the same Tier-0 gateway that you used with the IPSec service you created a moment ago.
6. Enter an optional description for this L2 VPN server.

7 Enter a value for **Tags** if you want to include this service in a tag group.

8 Enable or disable the **Hub & Spoke** property.

   By default, the value is set to `Disabled`, which means the traffic received from the L2 VPN clients is only replicated to the segments connected to the L2 VPN server. If this property is set to `Enabled`, the traffic from any L2 VPN client is replicated to all other L2 VPN clients.

9 Click **Save**.

   After the new L2 VPN server is created successfully, you are asked whether you want to continue with the rest of the L2 VPN service configuration. If you click **Yes**, you are taken back to the Add L2 VPN Server pane and the **Session** link is enabled. You can use that link to create an L2 VPN server session or use the **Networking > VPN > L2 VPN Sessions** tab.

What to do next

Configure an L2 VPN server session for the L2 VPN server that you configured using information in Add an L2 VPN Server Session as a guide.

Add an L2 VPN Client Service

After you configure the L2 VPN server, configure the L2 VPN service in the client mode on another Edge instance, which can be either an NSX-managed Edge, a standalone Edge, or an NSX-T software-defined data center (SDDC).

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Create a route-based IPSec tunnel for the L2 VPN client service.

   a Navigate to the **Networking > VPN > VPN Services** tab and select **Add Service > IPSec**.

   b Enter a name for the IPSec VPN service.

   c From the **Tier-0 Gateway** drop-down menu, select a Tier-0 gateway to use with the L2 VPN client.

   d If you want to use values different from the system defaults, set the rest of the properties on the Add IPSec Service pane, as needed.

   e Click **Save** and when prompted if you want to continue configuring the IPSec VPN service, select **No**.

3 Navigate to the **Networking > VPN > VPN Services** tab and select **Add Service > L2 VPN Client**.

4 Enter a name for the L2 VPN Client service.

5 From the **Tier-0 Gateway** drop-down menu, select the same Tier-0 gateway that you used with the route-based IPSec tunnel you created a moment ago.

6 Define the other properties on the Add L2 VPN Client pane if you want to use values other than the system defaults.

7 Click **Save**.

   After the new L2 VPN client service is created successfully, you are asked whether you want to continue with the rest of the L2 VPN client configuration. If you click **Yes**, you are taken back to the Add L2 VPN Client pane and the **Session** link is enabled. You can use that link to create an L2 VPN client session or use the **Networking > VPN > L2 VPN Sessions** tab.

What to do next

Configure an L2 VPN client session for the L2 VPN Client service that you configured. Use the information in Add an L2 VPN Client Session as a guide.

Adding IPSec VPN Sessions

After you have configured an IPSec VPN service, you must add either a policy-based IPSec VPN session or a route-based IPSec VPN session, depending on the type of IPSec VPN you want to configure. You also provide the information for the local endpoint and profiles to use to finish the IPSec VPN service configuration.

Add a Policy-Based IPSec Session

When you add a policy-based IPSec VPN, IPSec tunnels are used to connect multiple local subnets that are behind the NSX Edge node with peer subnets on the remote VPN site.

The following steps use the **IPSec Sessions** tab on the NSX Manager UI to create a policy-based IPSec session. You also add information for the tunnel, IKE, and DPD profiles, and select an existing local endpoint to use with the policy-based IPSec VPN.

Note You can also add the IPSec VPN sessions immediately after you have successfully configured the IPSec VPN service. You click **Yes** when prompted to continue with the IPSec VPN service configuration and select **Sessions > Add Sessions** on the Add IPsec Service panel. The first few steps in the following procedure assume you selected **No** to the prompt to continue with the IPSec VPN service configuration. If you selected **Yes**, proceed to step 3 in the following steps to guide you with the rest of the policy-based IPSec VPN session configuration.

Prerequisites

- You must have configured an IPSec VPN service before proceeding. See Add an IPSec VPN Service.
- Obtain the information for the local endpoint, IP address for the peer site, local network subnet, and remote network subnet to use with the policy-based IPSec VPN session you are adding. To create a local endpoint, see Add Local Endpoints.
- If you are using a Pre-Shared Key (PSK) for authentication, obtain the PSK value.

- If you are using a certificate for authentication, ensure that the necessary server certificates and corresponding CA-signed certificates are already imported. See Setting Up Certificates.
- If you do not want to use the defaults for the IPSec tunnel, IKE, or dead peer detection (DPD) profiles provided by NSX-T Data Center, configure the profiles you want to use instead. See Adding Profiles for information.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Navigate to the **Networking > VPN > IPSec Sessions** tab.

3 Select **Add IPSec Session > Policy Based**.

4 Enter a name for the policy-based IPSec VPN session.

5 From the **VPN Service** drop-down menu, select the IPSec VPN service to which you want to add this new IPSec session.

Note If you are adding this IPSec session from the **Add IPSec Sessions** dialog box, the VPN Service name is already indicated above the **Add IPSec Session** button.

6 Select an existing local endpoint from the drop-down menu.

This local endpoint value is required and identifies the local NSX Edge node. If you want to create a different local endpoint, click the three-dot menu (⋮) and select **Add Local Endpoint**.

7 In the **Remote IP** text box, enter the required IP address of the remote site.

This value is required.

8 Enter an optional description for this policy-based IPSec VPN session.

The maximum length is 1024 characters.

9 To enable or disable the IPSec VPN session, click **Admin Status** .

By default, the value is set to `Enabled`, which means the IPSec VPN session is to be configured down to the NSX Edge node.

10 Select a mode from the **Authentication Mode** drop-down menu.

The default authentication mode used is `PSK`, which means a secret key shared between NSX Edge and the remote site is used for the IPSec VPN session. If you select `Certificate`, the site certificate that was used to configure the local endpoint is used for authentication.

11 If you selected `PSK` for the authentication mode, enter the key value in the **Pre-shared Key** text box.

This secret key can be a string with a maximum length of 128 characters.

Caution Be careful when sharing and storing a PSK value because it contains sensitive information.

12 In the **Local Networks** and **Remote Networks** text boxes, enter at least one IP subnet address to use for this policy-based IPSec VPN session.

These subnets must be in a CIDR format.

13 To identify the peer site, enter a value in **Remote ID**.

For peer sites using PSK authentication, this ID value must be the public IP address or the FQDN of the peer site. For peer sites using certificate authentication, this ID value must be the common name (CN) or distinguished name (DN) used in the peer site's certificate.

Note If the peer site's certificate contains an email address in the DN string, for example,

```
C=US, ST=California, O=MyCompany, OU=MyOrg, CN=Site123/emailAddress=user1@mycompany.com
```

then enter the **Remote ID** value using the following format as an example.

```
C=US, ST=California, O=MyCompany, OU=MyOrg, CN=Site123, MAILTO=user1@mycompany.com"
```

If the local site's certificate contains an email address in the DN string and the peer site uses the strongSwan IPsec implementation, enter the local site's ID value in that peer site as shown in the following example.

```
C=US, ST=California, O=MyCompany, OU=MyOrg, CN=Site123, E=user1@mycompany.com"
```

---

14 If you want to include this session as part of a specific group, enter the tag name in **Tags**.

15 To change the profiles and initiation mode used by the policy-based IPSec VPN session, click **Profiles and Initiation Mode**.

By default, the system generated profiles are used. Select another available profile if you do not want to use the default. If you want to use a profile that is not configured yet, click the three-dot menu (⋮) to create another profile. See Adding Profiles.

a From the **IKE Profiles** drop-down menu, select the IKE profile to use.

b Select the preferred DPD profile from the **DPD Profiles** drop-down menu.

c In **IPSec Profiles**, select the IPsec tunnel profile to use with the IPSec session.

d Select the preferred mode from the **Connection Initiation Mode** drop-down menu.

Connection initiation mode defines the policy used by the local endpoint in the process of tunnel creation. The default value is `Initiator`. The following table describes the different connection initiation modes available.

Table 5-2. Connection Initiation Modes

Connection Initiation Mode	**Description**
`Initiator`	The default value. In this mode, the local endpoint initiates the IPSec VPN tunnel creation and responds to incoming tunnel setup requests from the peer gateway.
`On Demand`	In this mode, the local endpoint initiates the IPSec VPN tunnel creation after the first packet matching the policy rule is received. It also responds to the incoming initiation request.
`Respond Only`	The IPSec VPN never initiates a connection. The peer site always initiates the connection request and the local endpoint responds to that connection request.

16 Click **Save**.

Results

When the new policy-based IPSec VPN session is configured successfully, it is added to the list of available IPsec VPN sessions. It is in read-only mode.

What to do next

- Verify that the IPSec VPN tunnel status is Up. See Monitor and Troubleshoot VPN Sessions for information.
- If necessary, manage the IPSec VPN session information by clicking the three-dot menu (⋮) on the left-side of the session's row. Select one of the actions you are allowed to perform.

Add a Route-Based IPSec Session

When you add a route-based IPSec VPN, tunneling is provided on traffic that is based on routes that were learned dynamically over a virtual tunnel interface (VTI) using a preferred protocol, such as BGP. IPSec secures all the traffic flowing through the VTI.

The steps described in this topic use the **IPSec Sessions** tab to create a route-based IPSec session. You also add information for the tunnel, IKE, and DPD profiles, and select an existing local endpoint to use with the route-based IPSec VPN.

Note You can also add the IPSec VPN sessions immediately after you have successfully configured the IPSec VPN service. You click **Yes** when prompted to continue with the IPSec VPN service configuration and select **Sessions > Add Sessions** on the Add IPsec Service panel. The first few steps in the following procedure assume you selected **No** to the prompt to continue with the IPSec VPN service configuration. If you selected **Yes**, proceed to step 3 in the following steps to guide you with the rest of the route-based IPSec VPN session configuration.

Prerequisites

- You must have configured an IPSec VPN service before proceeding. See Add an IPSec VPN Service.
- Obtain the information for the local endpoint, IP address for the peer site, and tunnel service IP subnet address to use with the route-based IPSec session you are adding. To create a local endpoint, see Add Local Endpoints.
- If you are using a Pre-Shared Key (PSK) for authentication, obtain the PSK value.
- If you are using a certificate for authentication, ensure that the necessary server certificates and corresponding CA-signed certificates are already imported. See Setting Up Certificates.
- If you do not want to use the default values for the IPSec tunnel, IKE, or dead peer detection (DPD) profiles provided by NSX-T Data Center, configure the profiles you want to use instead. See Adding Profiles for information.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to **Networking > VPN > IPSec Sessions**.
3. Select **Add IPSec Session > Route Based**.
4. Enter a name for the route-based IPSec session.
5. From the **VPN Service** drop-down menu, select the IPSec VPN service to which you want to add this new IPSec session.

   Note If you are adding this IPSec session from the **Add IPSec Sessions** dialog box, the VPN Service name is already indicated above the **Add IPSec Session** button.

6. Select an existing local endpoint from the drop-down menu.

   This local endpoint value is required and identifies the local NSX Edge node. If you want to create a different local endpoint, click the three-dot menu (⋮) and select **Add Local Endpoint**.

7 In the **Remote IP** text box, enter the IP address of the remote site.

   This value is required.

8 Enter an optional description for this route-based IPSec VPN session.

   The maximum length is 1024 characters.

9 To enable or disable the IPSec session, click **Admin Status** .

   By default, the value is set to `Enabled`, which means the IPSec session is to be configured down to the NSX Edge node.

10 Select a mode from the **Authentication Mode** drop-down menu.

   The default authentication mode used is `PSK`, which means a secret key shared between NSX Edge and the remote site is to be used for the IPSec VPN session. If you select `Certificate`, the site certificate that was used to configure the local endpoint is used for authentication.

11 If you selected `PSK` for the authentication mode, enter the key value in the **Pre-shared Key** text box.

   This secret key can be a string with a maximum length of 128 characters.

   Caution Be careful when sharing and storing a PSK value because it contains some sensitive information.

   ---

12 Enter an IP subnet address in **Tunnel Interface** in the CIDR notation.

   This address is required.

13 Enter a value **Remote ID**.

   For peer sites using PSK authentication, this ID value must be the public IP address or the FQDN of the peer site. For peer sites using certificate authentication, this ID value must be the common name (CN) or distinguished name (DN) used in the peer site's certificate.

   Note If the peer site's certificate contains an email address in the DN string, for example,

   ```
   C=US, ST=California, O=MyCompany, OU=MyOrg, CN=Site123/emailAddress=user1@mycompany.com
   ```

   then enter the **Remote ID** value using the following format as an example.

   ```
   C=US, ST=California, O=MyCompany, OU=MyOrg, CN=Site123, MAILTO=user1@mycompany.com"
   ```

   If the local site's certificate contains an email address in the DN string and the peer site uses the strongSwan IPsec implementation, enter the local site's ID value in that peer site. The following is an example.

   ```
   C=US, ST=California, O=MyCompany, OU=MyOrg, CN=Site123, E=user1@mycompany.com"
   ```

   ---

14 If you want to include this IPSec session as part of a specific group tag, enter the tag name in **Tags**.

15 To change the profiles and initiation mode to be used by the route-based IPSec VPN session, click **Profiles and Initiation Mode**.

By default, the system-generated profiles are used. Select another available profile if you do not want to use the default. If you want to use a profile that is not configured yet, click the three-dot menu (⋮) to create another profile. See Adding Profiles.

a From the **IKE Profiles** drop-down menu, select the IKE profile to use.

b Select the preferred DPD profile from the **DPD Profiles** drop-down menu.

c In **IPSec Profiles**, select the IPsec tunnel profile to use with the IPSec session.

d Select the preferred mode from the **Connection Initiation Mode** drop-down menu.

Connection initiation mode defines the policy used by the local endpoint in the process of tunnel creation. The default value is `Initiator`. The following table describes the different connection initiation modes available.

Table 5-3. Connection Initiation Modes

Connection Initiation Mode	Description
`Initiator`	The default value. In this mode, the local endpoint initiates the IPSec VPN tunnel creation and responds to incoming tunnel setup requests from the peer gateway.
`On Demand`	Do not use with the route-based VPN. This mode applies to policy-based VPN only.
`Respond Only`	The IPSec VPN never initiates a connection. The peer site always initiates the connection request and the local endpoint responds to that connection request.

16 Click **Save**.

Results

When the new route-based IPSec VPN session is configured successfully, it is added to the list of available IPsec VPN sessions. It is in read-only mode.

What to do next

- Verify that the IPSec VPN tunnel status is Up. See Monitor and Troubleshoot VPN Sessions for information.
- Configure routing using either a static route or BGP. See Configure a Static Route or Configure BGP.
- If necessary, manage the IPSec VPN session information by clicking the three-dot menu (⋮) on the left-side of the session's row. Select one of the actions you can perform.

Adding L2 VPN Sessions

After you have configured an L2 VPN server and an L2 VPN client, you must add L2 VPN sessions for both to complete the L2 VPN service configuration.

Add an L2 VPN Server Session

After creating an L2 VPN Server service, you must add an L2 VPN session and attach it to an existing segment.

The following steps use the **L2 VPN Sessions** tab on the NSX Manager UI to create an L2 VPN Server session. You also select an existing local endpoint and segment to attach to the L2 VPN Server session.

Note You can also add an L2 VPN Server session immediately after you have successfully configured the L2 VPN Server service. You click **Yes** when prompted to continue with the L2 VPN Server configuration and select **Sessions > Add Sessions** on the Add L2 VPN Server panel. The first few steps in the following procedure assume you selected **No** to the prompt to continue with the L2 VPN Server configuration. If you selected **Yes**, proceed to step 3 in the following steps to guide you with the rest of the L2 VPN Server session configuration.

Prerequisites

- You must have configured an L2 VPN Server service before proceeding. See Add an L2 VPN Server Service.
- Obtain the information for the local endpoint and remote IP to use with the L2 VPN Server session you are adding. To create a local endpoint, see Add Local Endpoints.
- Obtain the values for the pre-shared key (PSK) and the tunnel interface subnet to use with the L2 VPN Server session.
- Obtain the name of the existing segment you want to attach to the L2 VPN Server session you are creating. See Add a Segment for information.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to the **Networking > VPN > L2 VPN Sessions** tab.
3. Select **Add L2 VPN Session > L2 VPN Server**.
4. Enter a name for the L2 VPN Server session.
5. From the **L2 VPN Service** drop-down menu, select the L2 VPN Server service for which the L2 VPN session is being created.

   Note If you are adding this L2 VPN Server session from the Set L2VPN Server Sessions dialog box, the L2 VPN Server service is already indicated above the **Add L2 Session** button.

6. Select an existing local endpoint from the drop-down menu.

   If you want to create a different local endpoint, click the three-dot menu (⋮) and select **Add Local Endpoint**.

7. Enter the IP address of the remote site.

8. To enable or disable the L2 VPN Server session, click **Admin Status**.

   By default, the value is set to `Enabled`, which means the L2 VPN Server session is to be configured down to the NSX Edge node.

9. Enter the secret key value in **Pre-shared Key**.

   Caution Be careful when sharing and storing a PSK value because it contains sensitive information.

10. Enter an IP subnet address in the **Tunnel Interface** using the CIDR notation.

    For example, 4.5.6.6/24. This subnet address is required.

11. Enter a value in **Remote ID**.

    For peer sites using certificate authentication, this ID must be the common name in the peer site's certificate. For PSK peers, this ID can be any string. Preferably, use the public IP address of the VPN or an FQDN for the VPN services as the `Remote ID`.

12. If you want to include this session as part of a specific group, enter the tag name in **Tags**.

13. Click **Save** and click **Yes** when prompted if you want to continue with the VPN service configuration.

    You are returned to the Add L2VPN Sessions panel and the **Segments** link is now enabled.

14. Attach an existing segment to the L2 VPN Server session.

    a Click **Segments > Set Segments**.

    b In the **Set Segments** dialog box, click **Set Segment** to attach an existing segment to the L2 VPN Server session.

    c From the **Segment** drop-down menu, select the segment you want to attach to the session.

    d Enter a value in the **VPN Tunnel ID** that is used to identify uniquely the segment you selected.

    e Click **Save** and then **Close**.

    In the Set L2VPN Sessions pane or dialog box, the system has incremented the **Segments** count for the L2 VPN Server session.

15. To finish the L2 VPN Server session configuration, click **Close Editing**.

Results

In the **VPN Services** tab, the system incremented the **Sessions** count for the L2 VPN Server service that you configured.

What to do next

To complete the L2 VPN service configuration, you must also create an L2 VPN service in Client mode and an L2 VPN client session. See Add an L2 VPN Client Service and Add an L2 VPN Client Session.

Add an L2 VPN Client Session

You must add an L2 VPN Client session after creating an L2 VPN Client service, and attach it to an existing segment.

The following steps use the **L2 VPN Sessions** tab on the NSX Manager UI to create an L2 VPN Client session. You also select an existing local endpoint and segment to attach to the L2 VPN Client session.

Note You can also add an L2 VPN Client session immediately after you have successfully configured the L2 VPN Client service. Click **Yes** when prompted to continue with the L2 VPN Client configuration and select **Sessions > Add Sessions** on the Add L2 VPN Client panel. The first few steps in the following procedure assume you selected **No** to the prompt to continue with the L2 VPN Client configuration. If you selected **Yes**, proceed to step 3 in the following steps to guide you with the rest of the L2 VPN Client session configuration.

Prerequisites

- You must have configured an L2 VPN Client service before proceeding. See Add an L2 VPN Client Service.
- Obtain the IP addresses information for the local IP and remote IP to use with the L2 VPN Client session you are adding.
- Obtain the peer codes that were generated during the L2 VPN server configuration. See Download the Remote Side L2 VPN Configuration.
- Obtain the name of the existing segment you want to attach to the L2 VPN Client session you are creating. See Add a Segment.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select the **Networking > VPN > L2 VPN Sessions**.
3. Select **Add L2 VPN Session > L2 VPN Client**.
4. Enter a name for the L2 VPN Client session.

5 From the **VPN Service** drop-down menu, select the L2 VPN Client service with which the L2 VPN session is to be associated.

   Note If you are adding this L2 VPN Client session from the Set L2VPN Client Sessions dialog box, the L2 VPN Client service is already indicated above the **Add L2 Session** button.

   ---

6 In the **Local IP address** text box, enter the IP address of the L2 VPN Client session.

7 Enter the remote IP address of the IPSec tunnel used for the L2 VPN Client service.

8 In the **Peer Configuration** text box, enter the peer code generated when you configured the L2 VPN Server service.

   a Navigate to where you downloaded the `L2VPNSession_`*<L2VPN-Server-Session>*`_config.txt` using Download the Remote Side L2 VPN Configuration.

   b Copy the file's content and paste it in the **Peer Configuration** text box.

9 Enable or disable **Admin Status**.

   By default, the value is set to `Enabled`, which means the L2 VPN Server session is to be configured down to the NSX Edge node.

10 Click **Save** and click **Yes** when prompted if you want to continue with the VPN service configuration.

11 Attach an existing segment to the L2 VPN Client session.

   a Select **Segments > Add Segments**.

   b In the **Set Segments** dialog box, click **Add Segment**.

   c From the **Segment** drop-down menu, select the segment you want to attach to the L2 VPN Server session.

   d Enter a value in the **VPN Tunnel ID**.

   e Click **Close**.

12 To finish the L2 VPN Client session configuration, click **Close Editing**.

Results

In the **VPN Services** tab, the sessions count is updated for the L2 VPN Client service that you configured.

Download the Remote Side L2 VPN Configuration

To configure the L2 VPN client session, you must obtain the peer code that was generated when you configured the L2 VPN server session.

Prerequisites

- You must have configured an L2 VPN server service and a session successfully before proceeding. See Add an L2 VPN Server Service.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Navigate to the **Networking > VPN > L2 VPN Sessions** tab.

3 In the table of L2 VPN sessions, expand the row for the L2 VPN server session you plan to use for the L2 VPN client session configuration.

4 Click **Download Config** and click **Yes** on the Warning dialog box.

   A text file with the name `L2VPNSession_<name-of-L2-VPN-server-session>_config.txt` is downloaded. It contains the peer code for the remote side L2 VPN configuration.

   Caution Be careful when storing and sharing the peer code because it contains a PSK value, which is sensitive information.

   For example, `L2VPNSession_L2VPNSess1_config.txt` contains the following configuration.

```
[{"transport_tunnel_path":"/infra/tier-0s/T0-gateway-1-AS/locale-services/1f309c00-277f-11e9-8074-
a18943ad6b99/ipsec-vpn-services/
IPS01-01/sessions/093ad8d0-2fad-11e9-8e5b-15a7211d1582",
"peer_code":"MCxiYTNjZmIwLHsic2l0ZU5hbWUiOiJMMlZQTi1MMlZTZXNzMSIsInNyY1RhcElwIjoiMTY5LjI1NC42NC4yI
iwiZHN0VGFwSXAiOiIxNjkuMjU0LjY0
LjEiLCJpa2VPcHRpb24iOiJpa2V2MiIsImVuY2FwUHJvdG8iOiJncmUvaXBzZWMiLCJkaEdyb3VwIjoiZGgxNCIsImVuY3J5cH
RBbmREaWdlc3QiOiJhZXMtZ2NtL3NoY
S0yNTYiLCJwc2siOiIxMTIyMzM0NDU1NjYiLCJ0dW5uZWxzIjpbeyJsb2NhbElkIjoiNC41LjYuNiIsInBlZXJJZCI6IjEuMS4
yLjIiLCJsb2NhbFZ0aUlwIjoiNC41Lj
YuMS8yNCJ9XX0="}]2NhbFZ0aUlwIjoiNC41LjYuMS8yNCJ9XX0="}]
```

What to do next

Configure the L2 VPN Client service and session. See Add an L2 VPN Client Service and Add an L2 VPN Client Session.

Add Local Endpoints

You must configure a local endpoint to use with the IPSec VPN that you are configuring.

The following steps use the **Local Endpoints** tab on the NSX Manager UI. You can also create a local endpoint while in the process of adding an IPSec VPN session by clicking the three-dot menu (⋮) and selecting **Add Local Endpoint**. If you are in the middle of configuring an IPSec VPN session, proceed to step 3 in the following steps to guide you with creating a new local endpoint.

Prerequisites

- If you are using a certificate-based authentication mode for the IPSec VPN session that is to use the local endpoint you are configuring, obtain the information about the certificate that the local endpoint must use.

- Ensure that you have configured an IPSec VPN service to which this local endpoint is to be associated.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to **Networking > VPN > Local Endpoints** and click **Add Local Endpoint**.
3. Enter a name for the local endpoint.
4. From the **VPN Service** drop-down menu, select the IPSec VPN service to which to associate this local endpoint.
5. Enter an IP address for the local endpoint.
6. If you are using a certificate-based authentication mode for the IPSec VPN session, from the **Site Certificate** drop-down menu, select the certificate that is to be used by the local endpoint.
7. Enter the **Local ID** value that is used for identifying the local NSX Edge instance.

   This local ID is the peer ID on the remote site. The local ID must be either the public IP address or FQDN of the remote site. For certificate-based VPN sessions that were defined using the local endpoint, the local ID is derived from the certificate associated with the local endpoint. The ID specified in the **Local ID** text box is ignored. The local ID derived from the certificate for a VPN session depends on the extensions present in the certificate.

   - If the X509v3 extension `X509v3 Subject Alternative Name` is not present in the certificate, then the Distinguished Name (DN) is used as the local ID value.
   - If the X509v3 extension `X509v3 Subject Alternative Name` is found in the certificate, then one of the Subject Alternative Name is taken as the local ID value.
8. From the **Trust CA Certificate** and **Trust CLR Certificate** drop-down menus, select the appropriate certificates that are required.
9. Specify a tag, if needed.
10. Click **Save**.

Adding Profiles

NSX-T Data Center provides the system-generated IPSec tunnel profile and an IKE profile that are assigned by default when you configure either an IPSec VPN or L2 VPN service. A system-generated DPD profile is created for an IPSec VPN configuration.

The IKE and IPSec profiles provide information about the algorithms that are used to authenticate, encrypt, and establish a shared secret between network sites. The DPD profile provides information about the number of seconds to wait in between probes.

If you decide not to use the default profiles provided by NSX-T Data Center, you can configure your own using the information in the topics that follow in this section.

Add IKE Profiles

The Internet Key Exchange (IKE) profiles provide information about the algorithms that are used to authenticate, encrypt, and establish a shared secret between network sites when you establish an IKE tunnel.

NSX-T Data Center provides system-generated IKE profiles that are assigned by default when you configure an IPSec VPN or L2 VPN service. The following table lists the default profiles provided.

Table 5-4. Default IKE Profiles Used for IPSec VPN or L2 VPN Services

Default IKE Profile Name	Description
nsx-default-l2vpn-ike-profile	■ Used for an L2 VPN service configuration. ■ Configured with IKE V2, AES 128 encryption algorithm, SHA2 256 algorithm, and Diffie-Hellman group14 key exchange algorithm.
nsx-default-l3vpn-ike-profile	■ Used for an IPSec VPN service configuration. ■ Configured with IKE V2, AES 128 encryption algorithm, SHA2 256 algorithm, and Diffie-Hellman group 14 key exchange algorithm.

If you decide not to use the default IKE profiles provided, you can configure your own using the following steps.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Click the **Networking > VPN > Profiles** tab.
3. Select the **IKE Profiles** profile type, and click **Add IKE Profile**.
4. Enter a name for the IKE profile.

5 From the **IKE Version** drop-down menu, select the IKE version to use to set up a security association (SA) in the IPSec protocol suite.

Table 5-5. IKE Versions

IKE Version	Description
IKEv1	When selected, the IPSec VPN initiates and responds to an IKEv1 protocol only.
IKEv2	This version is the default. When selected, the IPSec VPN initiates and responds to an IKEv2 protocol only.
IKE-Flex	If this version is selected and if the tunnel establishment fails with the IKEv2 protocol, the source site does not fall back and initiate a connection with the IKEv1 protocol. Instead, if the remote site initiates a connection with the IKEv1 protocol, then the connection is accepted.

6 Select the encryption, digest, and Diffie-Hellman group algorithms from the drop-down menus. You can select multiple algorithms to apply or deselect any selected algorithms you do not want to be applied.

Table 5-6. Algorithms Used

Type of Algorithm	Valid Values	Description
Encryption	■ AES 128 (default) ■ AES 256 ■ AES GCM 128 ■ AES GCM 192 ■ AES GCM 256	The encryption algorithm used during the Internet Key Exchange (IKE) negotiation. The AES-GCM algorithms are supported when used with IKEv2. They are not supported when used with IKEv1.
Digest	■ SHA2 256 (default) ■ SHA1 ■ SHA2 384 ■ SHA2 512	The secure hashing algorithm used during the IKE negotiation. If AES-GCM is the only encryption algorithm selected in the **Encryption Algorithm** text box, then no hash algorithms can be specified in the **Digest Algorithm** text box, per section 8 in RFC 5282. In addition, the Psuedo-Random Function (PRF) algorithm PRF-HMAC-SHA2-256 is implicitly selected and used in the IKE security association (SA) negotiation. The PRF-HMAC-SHA2-256 algorithm must also be configured on the peer gateway in order for the phase 1 of the IKE SA negotiation to succeed. If more algorithms are specified in the **Encryption Algorithm** text box, in addition to the AES-GCM algorithm, then one or more hash algorithms can be selected in the **Digest Algorithm** text box. In addition, the PRF algorithm used in the IKE SA negotiation is implicitly determined based on the hash algorithms configured. At least one of the matching PRF algorithms must also be configured on the peer gateway in order for the phase 1 of the IKE SA negotiation to succeed. For example, if the **Encryption Algorithm** text box contains AES 128 and AES GCM 128 and SHA1 is specified in the **Digest Algorithm** text box, then the PRF-HMAC-SHA1 algorithm is used during the IKE SA negotiation and must also be configured in the peer gateway.
Diffie-Hellman Group	■ Group 14 (default) ■ Group 2 ■ Group 5 ■ Group 15 ■ Group 16 ■ Group 19 ■ Group 20 ■ Group 21	The cryptography schemes that the peer site and the NSX Edge use to establish a shared secret over an insecure communications channel.

Note When you attempt to establish an IPSec VPN tunnel with a GUARD VPN Client (previously QuickSec VPN Client) using two encryption algorithms or two digest algorithms, the GUARD VPN Client adds additional algorithms in the proposed negotiation list. For example, if you specified AES 128 and AES 256 as the encryption algorithms and SHA2 256 and SHA2 512 as the digest algorithms to use in the IKE profile you are using to establish the IPSec VPN tunnel, the GUARD VPN Client also proposes AES 192 and SHA2 384 in the negotiation list. In this case, NSX-T Data Center uses the first encryption algorithm you selected when establishing the IPSec VPN tunnel.

7. Enter a security association (SA) lifetime value, in seconds, if you want it different from the default value of 86400 seconds (24 hours).
8. Provide a description and add a tag, as needed.
9. Click **Save**.

Results

A new row is added to the table of available IKE profiles. To edit or delete a non-system created profile, click the three-dot menu (⋮) and select from the list of actions available.

Add IPSec Profiles

The Internet Protocol Security (IPSec) profiles provide information about the algorithms that are used to authenticate, encrypt, and establish a shared secret between network sites when you establish an IPSec tunnel.

NSX-T Data Center provides system-generated IPSec profiles that are assigned by default when you configure an IPSec VPN or L2 VPN service. The following table lists the default profiles provided.

Table 5-7. Default IPSec Profiles Used for IPSec VPN or L2 VPN Services

Filename of Default IPSec Profile	Description
nsx-default-l2vpn-tunnel-profile	■ Used for L2 VPN. ■ Configured with AES GCM 128 encryption algorithm and Diffie-Hellman group 14 key exchange algorithm.
nsx-default-l3vpn-tunnel-profile	■ Used for IPSec VPN. ■ Configured with AES GCM 128 encryption algorithm and Diffie-Hellman group 14 key exchange algorithm.

If you decide not to use the default IPSec profiles provided, you can configure your own using the following steps.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2. Navigate to the **Networking > VPN > Profiles** tab.
3. Select the **IPSec Profiles** profile type, and click **Add IPSec Profile**.
4. Enter a name for the IPSec profile.
5. From the drop-down menus, select the encryption, digest, and Diffie-Hellman algorithms. You can select multiple algorithms to apply.

   Deselect the ones you do not want used.
6. Deselect **PFS Group** if you decide not to use the PFS Group protocol on your VPN service.

   It is selected by default.
7. In the **SA Lifetime** text box, modify the default number of seconds before the IPSec tunnel must be re-established.

   By default, an SA lifetime of 24 hours (86400 seconds) is used.
8. Select the value for **DF Bit** to use with the IPSec tunnel.

   The value determines how to handle the "Don't Fragment" (DF) bit included in the data packet received. The acceptable values are described in the following table.

Table 5-8. DF Bit Values

DF Bit Value	Description
`COPY`	The default value. When this value is selected, NSX-T Data Center copies the value of the DF bit from the received packet into the packet which is forwarded. This value implies that if the data packet received has the DF bit set, after encryption, the packet also has the DF bit set.
`CLEAR`	When this value is selected, NSX-T Data Center ignores the value of the DF bit in the data packet received, and the DF bit is always 0 in the encrypted packet.

9. Provide a description and add a tag, if necessary.
10. Click **Save**.

Results

A new row is added to the table of available IPSec profiles. To edit or delete a non-system created profile, click the three-dot menu (⋮) and select from the list of actions available.

Add DPD Profiles

A DPD (Dead Peer Detection) profile provides information about the number of seconds to wait in between probes to detect if an IPSec peer is alive or not.

NSX-T Data Center provides a system-generated DPD profile, named `nsx-default-l3vpn-dpd-profile`, that is assigned by default when you configure an IPSec VPN service.

If you decide not to use the default DPD profile provided, you can configure your own using the following steps.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to **Networking > VPN > Profiles**.
3. Select the **DPD Profiles** profile type, and click **Add DPD Profile**.
4. Enter a name for the DPD profile.
5. In the **DPD Probe Interval** text box, enter the number of seconds you want NSX-T Data Center to wait before sending the next DPD probe. The default is 60 seconds.

   If the NSX Edge node receives a response from the remote peer site, the DPD probe interval timer is restarted. If the NSX Edge node does not hear back from the peer site within 0.5 seconds after the next DPD probe is sent, a retransmission timer is set to 0.5 seconds. The NSX Edge node retransmits the next DPD probe after the retransmission timer is reached. If the remote peer site continues not to respond, the retransmission timer is exponentially increased to the maximum limit of 6 seconds. The NSX Edge node continues to retransmit the DPD probe every time the retransmission timer expires. The NSX Edge node retransmits up to a maximum of 30 times before it declares the peer site to be dead and it tears down the security association (SA) on the dead peer's link. The total time it takes to retransmit the DPD probe 30 times is about 2 minutes and 45 seconds.
6. Provide a description and add a tag, as needed.
7. Click **Save**.

Results

A new row is added to the table of available DPD profiles. To edit or delete a non-system created profile, click the three-dot menu (⋮) and select from the list of actions available.

Check the Realized State of an IPSec VPN Session

After you send a configuration update request for an IPSec VPN session, you can check to see if the requested state has been successfully processed in the NSX-T Data Center local control plane on the transport nodes.

When you create an IPSec VPN session, multiple entities are created: IKE profile, DPD profile, tunnel profile, local endpoint, IPSec VPN service, and IPSec VPN session. These entities all share the same `IPSecVPNSession` span, so you can obtain the realization state of all the entities of the IPSec VPN session by using the same `GET` API call. You can check the realization state using only the API.

Prerequisites

- Familiarize yourself with IPSec VPN. See Understanding IPSec VPN.
- Verify the IPSec VPN is configured successfully. See Add an IPSec VPN Service.

- You must have access to the NSX Manager API.

Procedure

1 Send a `POST`, `PUT`, or `DELETE` request API call.

For example:

```
PUT https://<nsx-mgr>/api/v1/vpn/ipsec/sessions/8dd1c386-9b2c-4448-85b8-51ff649fae4f
{
   "resource_type": "PolicyBasedIPSecVPNSession",
   "id": "8dd1c386-9b2c-4448-85b8-51ff649fae4f",
   "display_name": "Test RZ_UPDATED",
   "ipsec_vpn_service_id": "7adfa455-a6fc-4934-a919-f5728957364c",
   "peer_endpoint_id":  "17263ca6-dce4-4c29-bd8a-e7d12bd1a82d",
   "local_endpoint_id": "91ebfa0a-820f-41ab-bd87-f0fb1f24e7c8",
   "enabled": true,
   "policy_rules": [
       {
           "id": "1026",
           "sources": [
               {
                   "subnet": "1.1.1.0/24"
               }
           ],
           "logged": true,
           "destinations": [
               {
                   "subnet": "2.1.4..0/24"
               }
           ],
           "action": "PROTECT",
           "enabled": true,
           "_revision": 1
       }
   ]
}
```

2 Locate and copy the value of `x-nsx-requestid` from the response header returned.

For example:

```
x-nsx-requestid   e550100d-f722-40cc-9de6-cf84d3da3ccb
```

3 Request the realization state of the IPSec VPN session using the following `GET` call.

```
GET https://<nsx-mgr>/api/v1/vpn/ipsec/sessions/<ipsec-vpn-session-id>/state?request_id=<request-id>
```

The following API call uses the `id` and `x-nsx-requestid` values in the examples used in the previous steps.

```
GET https://<nsx-mgr>/api/v1/vpn/ipsec/sessions/8dd1c386-9b2c-4448-85b8-51ff649fae4f/state?
request_id=e550100d-f722-40cc-9de6-cf84d3da3ccb
```

Following is an example of a response you receive when the realization state is `in_progress`.

```
{
  "details": [
    {
      "sub_system_type": "TransportNode",
      "sub_system_id": "fe651e63-04bd-43a4-a8ec-45381a3b71b9",
      "state": "in_progress",
      "failure_message": "CCP Id:ab5958df-d98a-468e-a72b-d89dcdae5346, Message:State realization
is in progress at the node."
    },
    {
      "sub_system_type": "TransportNode",
      "sub_system_id": "ebe174ac-e4f1-4135-ba72-3dd2eb7099e3",
      "state": "in_sync"
    }
  ],
  "state": "in_progress",
  "failure_message": "The state realization is in progress at transport nodes."
}
```

Following is an example of a response you receive when the realization state is `in_sync`.

```
{
    "details": [
        {
            "sub_system_type": "TransportNode",
            "sub_system_id":  "7046e8f4-a680-11e8-9bc3-020020593f59",
            "state": "in_sync"
        }
    ],
    "state": "in_sync"
}
```

The following are examples of possible responses you receive when the realization state is `unknown`.

```
{
    "state": "unknown",
    "failure_message": "Unable to get response from any CCP node. Please retry operation after
some time."
}
```

```
{
    "details": [
       {
          "sub_system_type": "TransportNode",
          "sub_system_id": "3e643776-5def-11e8-94ae-020022e7749b",
          "state": "unknown",
          "failure_message": "CCP Id:ab5958df-d98a-468e-a72b-d89dcdae5346, Message: Unable to get
response from the node. Please retry operation after some time."
       },
       {
          "sub_system_type": "TransportNode",
          "sub_system_id": "4784ca0a-5def-11e8-93be-020022f94b73",
          "state": "in_sync"
       }
    ],
    "state": "unknown",
    "failure_message": "The state realization is unknown at transport nodes"
}
```

After you perform an entity `DELETE` operation, you might receive the status of `NOT_FOUND`, as shown in the following example.

```
{
   "http_status": "NOT_FOUND",
   "error_code": 600,
   "module_name": "common-services",
   "error_message": "The operation failed because object identifier LogicalRouter/
61746f54-7ab8-4702-93fe-6ddeb804 is missing: Object identifiers are case sensitive.."
}
```

If the IPSec VPN service associated with the session is disabled, you receive the `BAD_REQUEST` response, as shown in the following example.

```
{
    "httpStatus": "BAD_REQUEST",
    "error_code": 110199,
    "module_name": "VPN",
    "error_message": "VPN service f9cfe508-05e3-4e1d-b253-fed096bb2b63 associated with the
session 8dd1c386-9b2c-4448-85b8-51ff649fae4f is disabled. Can not get the realization status."
}
```

Monitor and Troubleshoot VPN Sessions

After you configure an IPSec or L2 VPN session, you can monitor the VPN tunnel status and troubleshoot any reported tunnel issues using the NSX Manager user interface.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to the **Networking > VPN > IPSec Sessions** or **Networking > VPN > L2 VPN Sessions** tab.
3. Expand the row for the VPN session that you want to monitor or troubleshoot.

4. To view the status of the VPN tunnel status, click the info icon.

   The Status dialog box appears and displays the available statuses.

5. To view the VPN tunnel traffic statistics, click **View Statistics** in the Status column.

   The Statistics dialog box displays the traffic statistics for the VPN tunnel.

6. To view the error statistics, click the **View More** link in the Statistics dialog box.

7. To close the **Statistics** dialog box, click **Close**.

Network Address Translation

6

Network address translation (NAT) maps one IP address space to another. You can configure NAT on tier-0 and tier-1 gateways.

This chapter includes the following topics:

Configure NAT on a Gateway

You can configure source NAT (SNAT), destination NAT (DNAT), or reflexive NAT on a tier-0 or tier-1 gateway.

If a tier-0 gateway is running in active-active mode, you cannot configure SNAT or DNAT because asymmetrical paths might cause issues. You can only configure reflexive NAT (sometimes called stateless NAT). If a tier-0 gateway is running in active-standby mode, you can configure SNAT, DNAT, or reflexive NAT.

You can also disable SNAT or DNAT for an IP address or a range of addresses. If an address has multiple NAT rules, the rule with the highest priority is applied.

SNAT configured on a tier-0 gateway's external interface will process traffic from a tier-1 gateway as well as from another external interface on the tier-0 gateway.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > NAT**.
3. Select a gateway.
4. Click **Add NAT Rule**.
5. Select an action.

   For a tier-1 gateway, the available actions are **SNAT**, **DNAT**, **Reflexive**, **NO SNAT**, and **NO DNAT**.

For a tier-0 gateway in active-standby mode, the available actions are **SNAT**, **DNAT**, **NO SNAT**, and **NO DNAT**.

For a tier-0 gateway in active-active mode, the available action is **Reflexive**.

6 In the **Service** column, click **Set** to select services.

7 (Required) For **Source IP**, specify an IP address or an IP address range in CIDR format.

   If you leave this field blank, this NAT rule applies to all sources outside of the local subnet.

8 For **Destination IP**, specify an IP address or an IP address range in CIDR format.

9 For **Translated IP**, specify an IP address or an IP address range in CIDR format.

10 Enter a value for **Translated Port**.

11 Select a firewall setting from the following options:

   - **Match External Address** - The packet is processed by firewall rules that match the combination of translated IP address, and translated port.
   - **Match Internal Address** - The packet is processed by firewall rules that match the combination of original IP address, and original port.
   - **Bypass** - The packet bypasses firewall rules.

12 (Required) Change the logging status.

13 (Required) For **Applied To**, select objects that this rule applies to.

   The available objects are **Tier-0 Gateways**, **Interfaces**, **Labels**, **Service Instance Endpoints**, and **Virtual Endpoints**.

14 Specify a priority value.

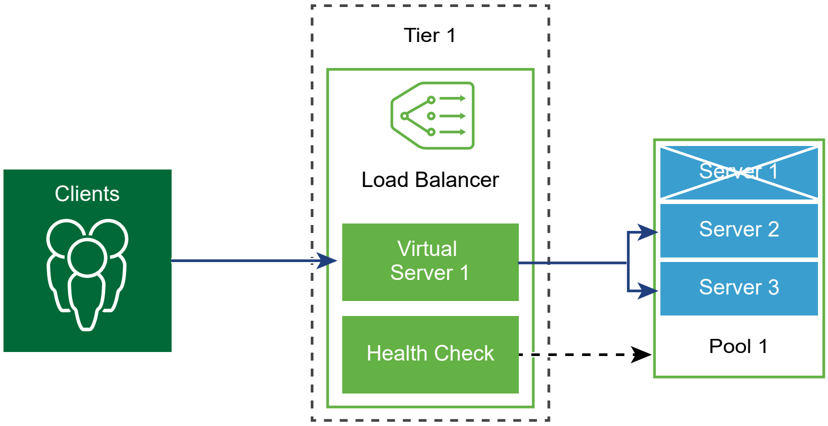
   A lower value means a higher priority. The default is 100.

15 Click **Save**.

Load Balancing

7

The NSX-T Data Center logical load balancer offers high-availability service for applications and distributes the network traffic load among multiple servers.



The load balancer distributes incoming service requests evenly among multiple servers in such a way that the load distribution is transparent to users. Load balancing helps in achieving optimal resource utilization, maximizing throughput, minimizing response time, and avoiding overload.

You can map a virtual IP address to a set of pool servers for load balancing. The load balancer accepts TCP, UDP, HTTP, or HTTPS requests on the virtual IP address and decides which pool server to use.

Depending on your environment needs, you can scale the load balancer performance by increasing the existing virtual servers and pool members to handle heavy network traffic load.

Note Logical load balancer is supported only on the tier-1 gateway. One load balancer can be attached only to a tier-1 gateway.

This chapter includes the following topics:

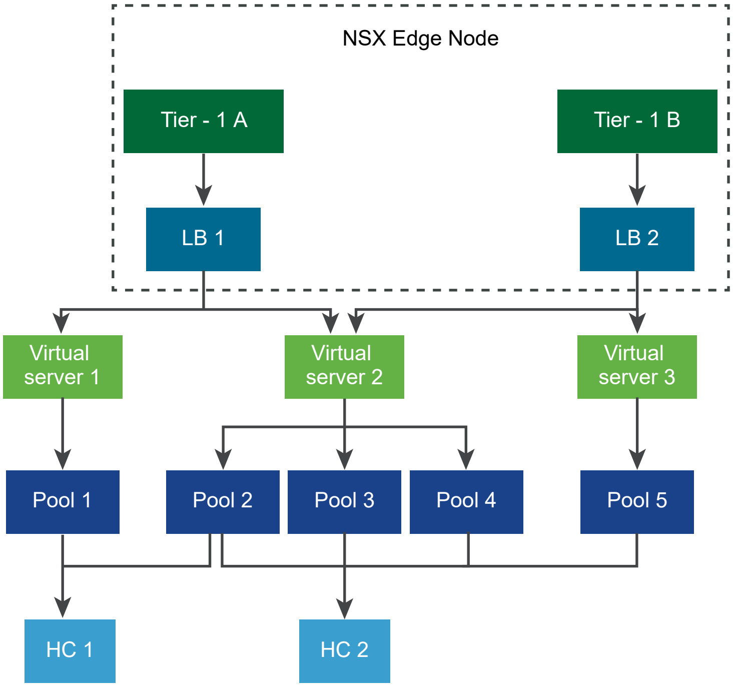
- Key Load Balancer Concepts
- Setting Up Load Balancer Components

Key Load Balancer Concepts

Load balancer includes virtual servers, server pools, and health checks monitors.

A load balancer is connected to a Tier-1 logical router. The load balancer hosts single or multiple virtual servers. A virtual server is an abstract of an application service, represented by a unique combination of IP, port, and protocol. The virtual server is associated to single to multiple server pools. A server pool consists of a group of servers. The server pools include individual server pool members.

To test whether each server is correctly running the application, you can add health check monitors that check the health status of a server.



Scaling Load Balancer Resources

Load balancers are available in small, medium, and large sizes. Based on the load balancer size, the load balancer can host different virtual servers and pool members.

Note In the **Advanced & Security** tab, the term tier-1 logical router is used to refer to a tier-1 gateway.

Table 7-1. Load Balancer Scale for Load Balancer Service

Load Balancer Service	Small Load Balancer	Medium Load Balancer	Large Load Balancer
Number of virtual servers per load balancer	20	100	1000
Number of pools per load balancer	60	300	3000
Number of pool members per load balancer	300	2000	7500

A load balancer is attached to one tier-1 logical router. This tier-1 logical router which must be in Active-Standby mode is hosted on the NSX Edge nodes.

NSX Edge has the form factor Bare Metal, small, medium, and large sized VM appliances. Based on the form factor, the NSX Edge node can host a different number of load balancers.

Table 7-2. Load Balancer Scale for NSX Edge Node

Load Balancer Per NSX Edge Node	Small Load Balancer	Medium Load Balancer	Large Load Balancer	Maximum Pool Members
NSX Edge VM - Small	N/A	N/A	N/A	N/A
NSX Edge VM - Medium	1	N/A	N/A	300
NSX Edge VM - Large	40	4	1	7500
NSX Edge VM - Bare Metal	750	75	18	30,000

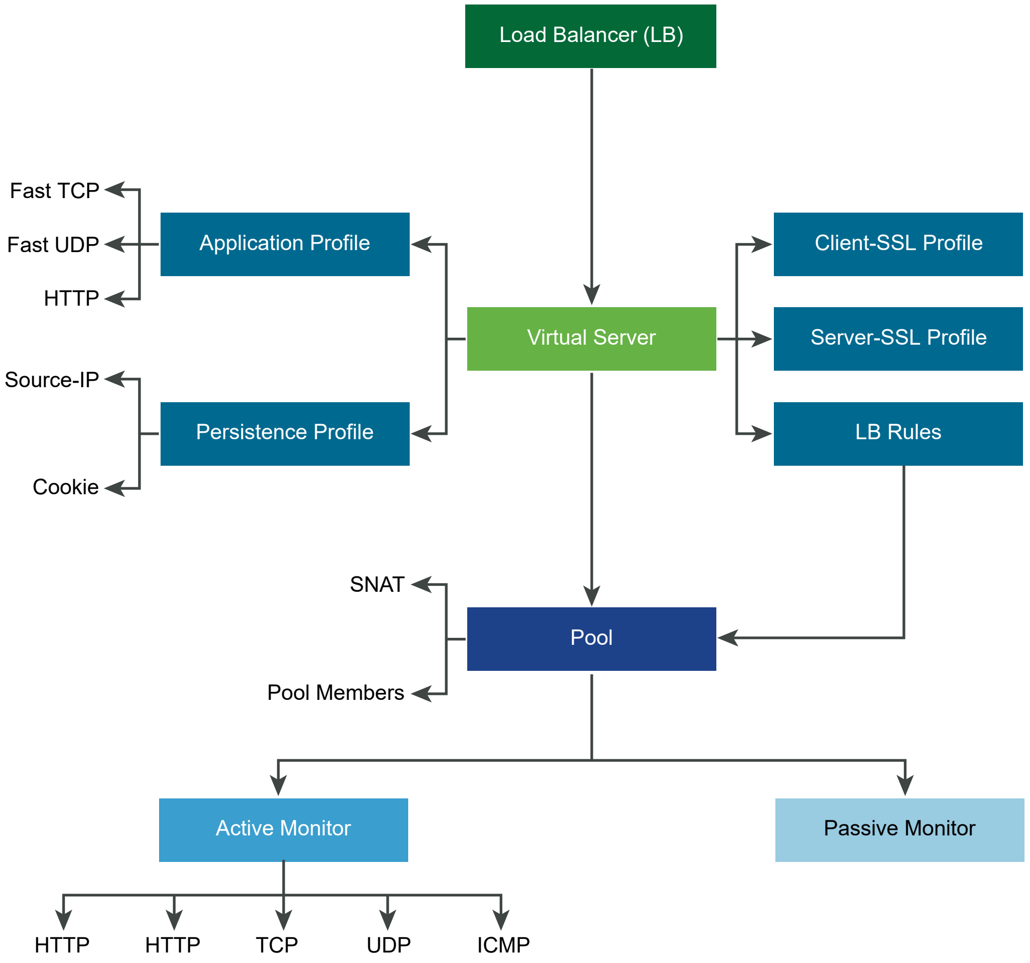
Supported Load Balancer Features

NSX-T Data Center load balancer supports the following features.

- Layer 4 - TCP and UDP
- Layer 7 - HTTP and HTTPS with load balancer rules support
- Server pools - static and dynamic with NSGroup
- Persistence - Source-IP and Cookie persistence mode
- Health check monitors - Active monitor which includes HTTP, HTPPS, TCP, UDP, and ICMP, and passive monitor
- SNAT - Transparent, Automap, and IP List
- HTTP upgrade - For applications using HTTP upgrade such as WebSocket, the client or server requests for HTTP Upgrade, which is supported. By default, NSX-T Data Center supports and accepts HTTPS upgrade client request using the HTTP application profile.

  To detect an inactive client or server communication, the load balancer uses the HTTP application profile response timeout feature set to 60 seconds. If the server does not send traffic during the 60 seconds interval, NSX-T Data Center ends the connection on the client and server side.

Note: SSL -Terminate-mode and proxy-mode is not supported in NSX-T Data Center limited export release.

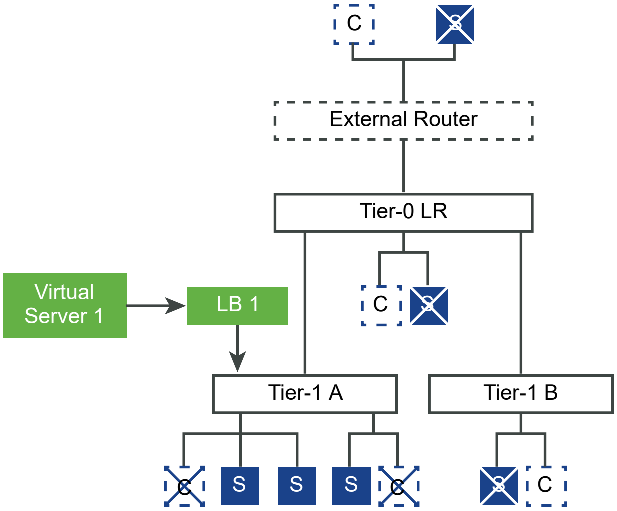


Load Balancer Topologies

Load balancers are typically deployed in either inline or one-arm mode. One-arm mode requires virtual server Source NAT (SNAT) configuration, and inline mode does not.

Inline Topology

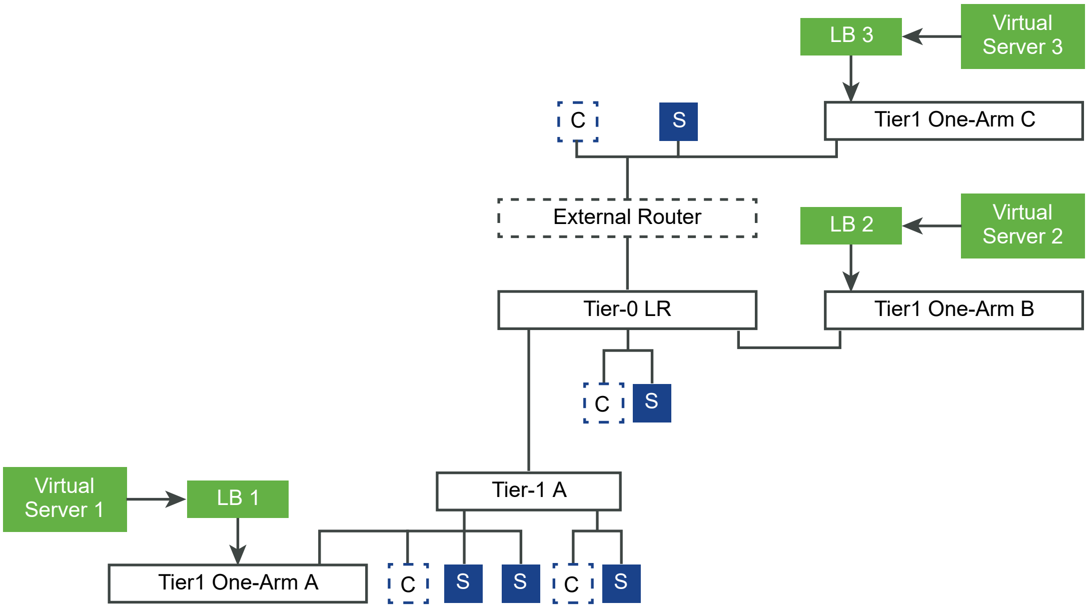
In the inline mode, the load balancer is in the traffic path between the client and the server. Clients and servers should not be connected to overlay segments on the same tier-1 logical router if SNAT on the load balancer is not desired. If clients and servers are connected to overlay segments on the same tier-1 logical router, SNAT is required.



One-Arm Topology

In one-arm mode, the load balancer is not in the traffic path between the client and the server. In this mode, the client and the server can be anywhere. The load balancer performs Source NAT (SNAT) to force return traffic from the server destined to the client to go through the load balancer. This topology requires virtual server SNAT to be enabled.

When the load balancer receives the client traffic to the virtual IP address, the load balancer selects a server pool member and forwards the client traffic to it. In the one-arm mode, the load balancer replaces the client IP address with the load balancer IP address so that the server response is always sent to the load balancer. The load balancer forwards the response to the client.



Tier-1 Service Chaining

If a tier-1 gateway or logical router hosts different services, such as NAT, firewall, and load balancer, the services are applied in the following order:

- Ingress

  DNAT - Firewall - Load Balancer

  Note: If DNAT is configured with Firewall Bypass, Firewall is skipped but not Load Balancer.

- Egress

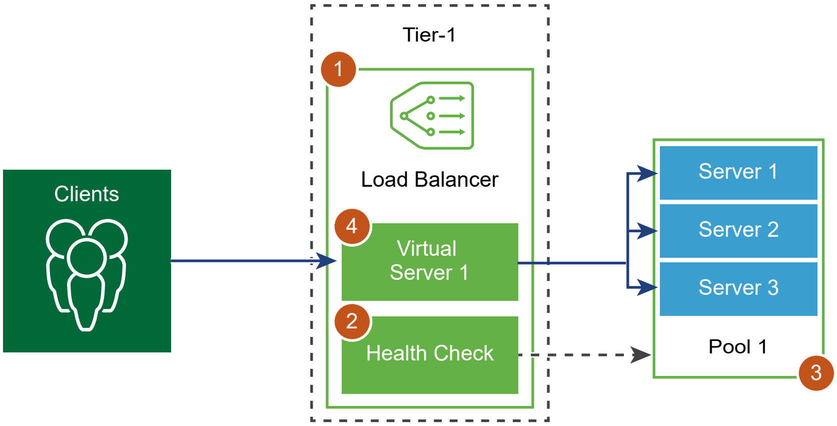
  Load Balancer - Firewall - SNAT

Setting Up Load Balancer Components

To use logical load balancers, you must start by configuring a load balancer and attaching it to a tier-1 gateway.

Note In the **Advanced & Security** tab, the term tier-1 logical router is used to refer to a tier-1 gateway.

Next, you set up health check monitoring for your servers. You must then configure server pools for your load balancer. Finally, you must create a layer 4 or layer 7 virtual server for your load balancer and attach the newly created virtual server to the load balancer.



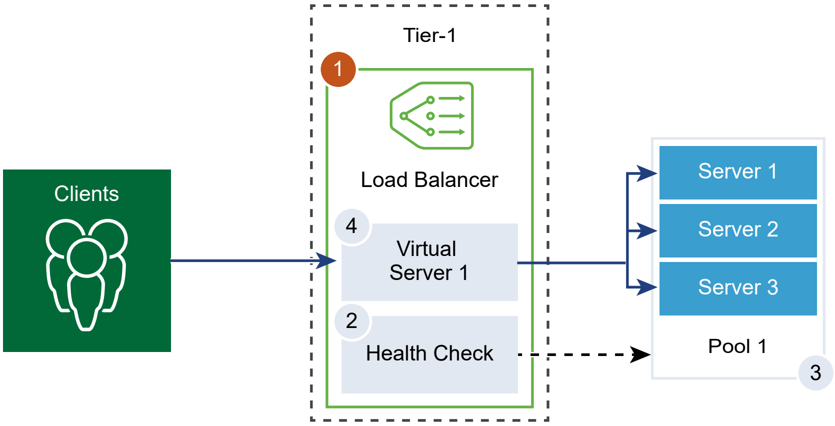
Add Load Balancers

Load balancer is created and attached to the tier-1 gateway.

Note In the **Advanced & Security** tab, the term tier-1 logical router is used to refer to a tier-1 gateway.

You can configure the level of error messages you want the load balancer to add to the error log.

Note Avoid setting the log level to DEBUG on load balancers with a significant traffic due to the number of messages printed to the log that affect performance.



Prerequisites

Verify that a tier-1 gateway is configured. See Chapter 3 Tier-1 Gateway.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > Load Balancing > Add Load Balancer**.

3 Enter a name and a description for the load balancer.

4 Select the load balancer virtual server size and number of pool members based on your available resources.

5 Select the already configured tier-1 gateway to attach to this load balancer from the drop-down menu.

   The tier-1 gateway must be in the Active-Standby mode.

6 Define the severity level of the error log from the drop-down menu.

   Load balancer collects information about encountered issues of different severity levels to the error log.

7 (Optional) Enter tags to make searching easier.

   You can specify a tag to set a scope of the tag.

8 Toggle the button to disable the admin state of the load balancer.

9 Click **Save**.

   The load balancer creation and attaching the load balancer to the tier-1 gateway takes about three minutes and the configuration status to appear green and Up.

If the status is Down, click the information icon and resolve the error before you proceed.

10 (Optional) Delete the load balancer.

   a Detach the load balancer from the virtual server and tier-1 gateway.

   b Select the load balancer.

   c Click the vertical ellipses button.

   d Select **Delete**.

Add an Active Monitor

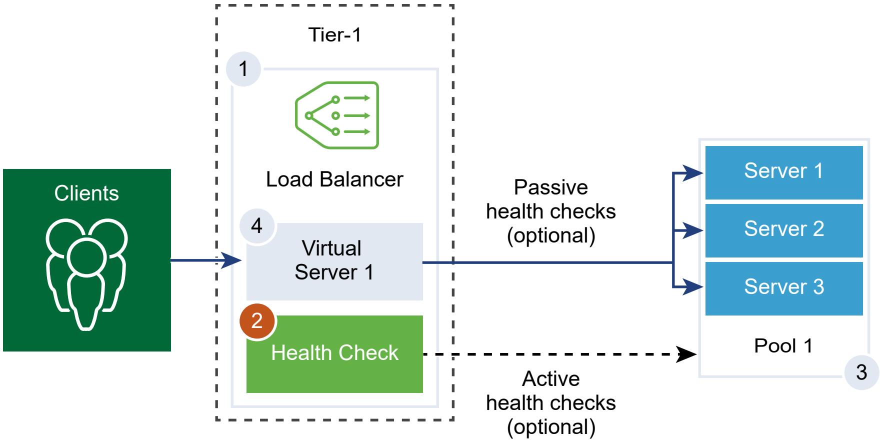
The active health monitor is used to test whether a server is available. The active health monitor uses several types of tests such as sending a basic ping to servers or advanced HTTP requests to monitor an application health.

Note In the **Advanced & Security** tab, the term tier-1 logical router is used to refer to a tier-1 gateway.

Servers that fail to respond within a certain time period or respond with errors are excluded from future connection handling until a subsequent periodic health check finds these servers to be healthy.

Active health checks are performed on server pool members after the pool member is attached to a virtual server and that virtual server is attached to a tier-1 gateway. The tier-1 uplink IP address is used for the health check.

Note One active health monitor can be configured per server pool.



Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking** > **Load Balancing** > **Monitors** > **Active** > **Add Active Monitor**.

3 Select a protocol for the server from the drop-down menu.

   You can also use predefined protocols; HTTP, HTTPS, ICMP, TCP, and UDP for NSX Manager.

4 Select the **HTTP** protocol.

5 Configure the values to monitor a service pool.

   You can also accept the default active health monitor values.

Option	Description
Name and Description	Enter a name and description for the active health monitor.
Monitoring Port	Set the value of the monitoring port.
Monitoring Interval	Set the time in seconds that the monitor sends another connection request to the server.
Timeout Period	Set the number of times the server is tested before it is considered as DOWN.
Fall Count	Set a value when the consecutive failures reach this value, the server is considered temporarily unavailable.
Rise Count	Set a number after this timeout period, the server is tried again for a new connection to see if it is available.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

   For example, if the monitoring interval is set as 5 seconds and the timeout as 15 seconds, the load balancer send requests to the server every 5 seconds. In each probe, if the expected response is received from the server within 15 seconds, then the health check result is OK. If not, then the result is CRITICAL. If the recent three health check results are all UP, the server is considered as UP.

6 Click **Configure**.

7 Enter the HTTP request and response configuration details.

Option	Description
HTTP Method	Select the method to detect the server status from the drop-down menu, GET, OPTIONS, POST, HEAD, and PUT.
HTTP Request URL	Enter the request URI for the method.
HTTP Request Version	Select the supported request version from the drop-down menu. You can also accept the default version, HTTP_VERSION_1.
HTTP Response Header	Click **Add** and enter the HTTP response header name and corresponding value. The default header value is 4000. The maximum header value is 64,000.
HTTP Request Body	Enter the request body. Valid for the POST and PUT methods.

Option	Description
HTTP Response Code	Enter the string that the monitor expects to match in the status line of HTTP response body. The response code is a comma-separated list. For example, 200,301,302,401.
HTTP Response Body	If the HTTP response body string and the HTTP health check response body match, then the server is considered as healthy.

8 Select the **HTTPS** protocol.

9 Complete step 5.

10 Click **Configure**.

11 Enter the HTTP request and response and SSL configuration details.

Option	Description
Name and Description	Enter a name and description for the active health monitor.
HTTP Method	Select the method to detect the server status from the drop-down menu, GET, OPTIONS, POST, HEAD, and PUT.
HTTP Request URL	Enter the request URI for the method.
HTTP Request Version	Select the supported request version from the drop-down menu. You can also accept the default version, HTTP_VERSION_1.
HTTP Response Header	Click **Add** and enter the HTTP response header name and corresponding value. The default header value is 4000. The maximum header value is 64,000.
HTTP Request Body	Enter the request body. Valid for the POST and PUT methods.
HTTP Response Code	Enter the string that the monitor expects to match in the status line of HTTP response body. The response code is a comma-separated list. For example, 200,301,302,401.
HTTP Response Body	If the HTTP response body string and the HTTP health check response body match, then the server is considered as healthy.
Server SSL	Toggle the button to enable the SSL server.
Client Certificate	(Optional) Select a certificate from the drop-down menu to be used if the server does not host multiple host names on the same IP address or if the client does not support an SNI extension.
Server SSL Profile	(Optional) Assign a default SSL profile from the drop-down menu that defines reusable and application-independent client-side SSL properties. Click the vertical ellipses and create a custom SSL profile.
Trusted CA Certificates	(Optional) You can require the client to have a CA certificate for authentication.
Mandatory Server Authentication	(Optional) Toggle the button to enable server authentication.

Option	Description
Certificate Chain Depth	(Optional) Set the authentication depth for the client certificate chain.
Certificate Revocation List	(Optional) Set a Certificate Revocation List (CRL) in the client-side SSL profile to reject compromised client certificates.

12 Select the **ICMP** protocol.

13 Complete step 5 and assign the data size in byte of the ICMP health check packet.

14 Select the **TCP** protocol.

15 Complete step 5 and you can leave the TCP data parameters empty.

If both the data sent and expected are not listed, then a three-way handshake TCP connection is established to validate the server health. No data is sent.

Expected data if listed has to be a string. Regular expressions are not supported.

16 Select the **UDP** protocol.

17 Complete step 5 and configure the UDP data.

Required Option	Description
UDP Data Sent	Enter the string to be sent to a server after a connection is established.
UDP Data Expected	Enter the string expected to receive from the server. Only when the received string matches this definition, is the server is considered as UP.

What to do next

Associate the active health monitor with a server pool. See Add a Server Pool.

Add a Passive Monitor

Load balancers perform passive health checks to monitor failures during client connections and mark servers causing consistent failures as DOWN.

Passive health check monitors client traffic going through the load balancer for failures. For example, if a pool member sends a TCP Reset (RST) in response to a client connection, the load balancer detects that failure. If there are multiple consecutive failures, then the load balancer considers that server pool member to be temporarily unavailable and stops sending connection requests to that pool member for some time. After some time, the load balancer sends a connection request to verify that the pool member has recovered. If that connection is successful, then the pool member is considered healthy. Otherwise, the load balancer waits for some time and tries again.

Passive health check considers the following scenarios to be failures in the client traffic.

- For server pools associated with Layer 7 virtual servers, if the connection to the pool member fails. For example, if the pool member sends a TCP RST when the load balancer tries to connect or perform an SSL handshake between the load balancer and the pool member fails.

- For server pools associated with Layer 4 TCP virtual servers, if the pool member sends a TCP RST in response to client TCP SYN or does not respond at all.
- For server pools associated with Layer 4 UDP virtual servers, if a port is unreachable or a destination unreachable ICMP error message is received in response to a client UDP packet.

Server pools associated to Layer 7 virtual servers, the failed connection count is incremented when any TCP connection errors, for example, TCP RST failure to send data or SSL handshake failures occur.

Server pools associated to Layer 4 virtual servers, if no response is received to a TCP SYN sent to the server pool member or if a TCP RST is received in response to a TCP SYN, then the server pool member is considered as DOWN. The failed count is incremented.

For Layer 4 UDP virtual servers, if an ICMP error such as, port or destination unreachable message is received in response to the client traffic, then it is considered as DOWN.

Note One passive health monitor can be configured per server pool.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > Load Balancing > Monitors > Passive > Add Passive Monitor**.
3. Enter a name and description for the passive health monitor.
4. Configure the values to monitor a service pool.

   You can also accept the default active health monitor values.

Option	Description
Fall Count	Set a value when the consecutive failures reach this value, the server is considered temporarily unavailable.
Timeout Period	Set the number of times the server is tested before it is considered as DOWN.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

   For example, when the consecutive failures reach the configured value 5, that member is considered temporarily unavailable for 5 seconds. After this period, that member is tried again for a new connection to see if it is available. If that connection is successful, then the member is considered available and the failed count is set to zero. However, if that connection fails, then it is not used for another timeout interval of 5 seconds.

What to do next

Associate the passive health monitor with a server pool. See Add a Server Pool.

Add a Server Pool

Server pool consists of one or more servers that are configured and running the same application. A single pool can be associated to both Layer 4 and Layer 7 virtual servers.

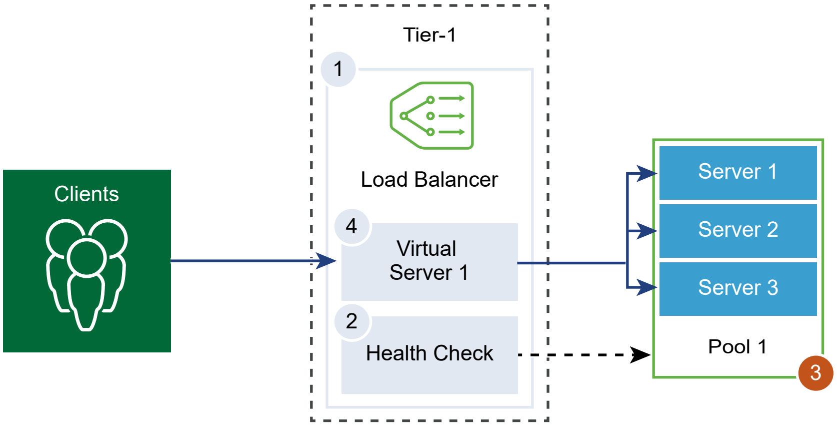
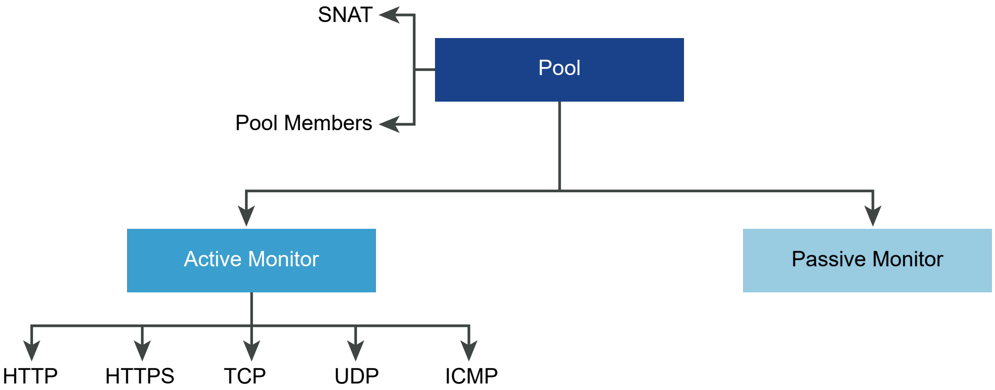


Figure 7-1. Server Pool Parameter Configuration



Prerequisites

- If you use dynamic pool members, a NSGroup must be configured. See Create an NSGroup.
- Verify that a passive health monitors is configured. See Add a Passive Monitor.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking** > **Load Balancing** > **Server Pools** > **Add Server Pool**.

3 Enter a name and description for the load balancer server pool.

You can optionally describe the connections managed by the server pool.

4 Select the algorithm balancing method for the server pool.

Load balancing algorithm controls how the incoming connections are distributed among the members. The algorithm can be used on a server pool or a server directly.

All load balancing algorithms skip servers that meet any of the following conditions:

- Admin state is set to DISABLED
- Admin state is set to GRACEFUL_DISABLED and no matching persistence entry
- Active or passive health check state is DOWN
- Connection limit for the maximum server pool concurrent connections is reached.

Option	Description
ROUND_ROBIN	Incoming client requests are cycled through a list of available servers capable of handling the request. Ignores the server pool member weights even if they are configured.
WEIGHTED_ROUND_ROBIN	Each server is assigned a weight value that signifies how that server performs relative to other servers in the pool. The value determines how many client requests are sent to a server compared to other servers in the pool. This load balancing algorithm focuses on fairly distributing the load among the available server resources.
LEAST_CONNECTION	Distributes client requests to multiple servers based on the number of connections already on the server. New connections are sent to the server with the fewest connections. Ignores the server pool member weights even if they are configured.
WEIGHTED_LEAST_CONNECTION	Each server is assigned a weight value that signifies how that server performs relative to other servers in the pool. The value determines how many client requests are sent to a server compared to other servers in the pool. This load balancing algorithm focuses on using the weight value to distribute the load among the available server resources. By default, the weight value is 1 if the value is not configured and slow start is enabled.
IP-HASH	Selects a server based on a hash of the source IP address and the total weight of all the running servers.

5 Select the server pool members.

Server pool consists of single or multiple pool members.

Option	Description
Enter individual members	Enter a pool member name, IP address, and a port. Each server pool member can be configured with a weight for use in the load balancing algorithm. The weight indicates how much more or less load a given pool member can handle relative to other members in the same pool. You can set the server pool admin state. By default, the option is enable when a server pool member is added. If the option is disabled, active connections are processed and the server pool member is not selected for new connections. New connections are assigned to other members of the pool. If gracefully disabled, it allows you to remove servers for maintenance. The existing connections to a member in the server pool in this state continue to be processed. Toggle the button to designate a pool member as a backup member to work with the health monitor to provide an Active-Standby state. Traffic failover occurs for backup members if active members fail a health check. Backup members are skipped during the server selection. When the server pool is inactive, the incoming connections are sent to only the backup members that are configured with a sorry page indicating an application is unavailable. Max Concurrent Connection value assigns a connection maximum so that the server pool members are not overloaded and skipped during server selection. If a value is not specified, then the connection is unlimited.
Select a group	Select a pre-configured group of server pool members. Enter a group name, optional description, and domain. Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1. Set the compute member from existing list or create one. You can specify membership criteria, select members of the group, add IP and MAC addresses as group members, and add Active Directory groups. The identity members intersect with the compute member to define membership of the group. Enter tags to make searching easier. You can specify a tag to set a scope of the tag. You can optionally, define the maximum group IP address list.

6 Select the active health check monitor for the server pool from the drop-down menu.

The load balancer periodically sends an ICMP ping to the servers to verify health independent of data traffic. You can configure only one active health check monitor per server pool.

7 Select the Source NAT (SNAT) translation mode.

Depending on the topology, SNAT might be required so that the load balancer receives the traffic from the server destined to the client. SNAT can be enabled per server pool.

Mode	Description
Auto Map Mode	Load Balancer uses the interface IP address and ephemeral port to continue the communication with a client initially connected to one of the server's established listening ports. SNAT is required. Enable port overloading to allow the same SNAT IP and port to be used for multiple connections if the tuple (source IP, source port, destination IP, destination port, and IP protocol) is unique after the SNAT process is performed. You can also set the port overload factor to allow the maximum number of times a port can be used simultaneously for multiple connections.
Disable	Disable the SNAT translation mode.
IP Pool	Specify a single IP address range, for example, 1.1.1.1-1.1.1.10 to be used for SNAT while connecting to any of the servers in the pool. By default, from 4000 through 64000-port range is used for all configured SNAT IP addresses. Port ranges from 1000 through 4000 are reserved for purposes such as, health checks and connections initiated from Linux applications. If multiple IP addresses are present, then they are selected in a Round Robin manner. Enable port overloading to allow the same SNAT IP and port to be used for multiple connections if the tuple (source IP, source port, destination IP, destination port, and IP protocol) is unique after the SNAT process is performed. You can also set the port overload factor to allow the maximum number of times a port can be used simultaneously for multiple connections.

8 Toggle the button to enable TCP Multiplexing.

With TCP multiplexing, you can use the same TCP connection between a load balancer and the server for sending multiple client requests from different client TCP connections.

9 Set the maximum number of TCP multiplexing connections per pool that are kept alive to send future client requests.

10 Enter the minimum number of active members the server pool must always maintain.

11 Select a passive health monitor for the server pool from the drop-down menu.

12 Enter tags to make searching easier.

You can specify a tag to set a scope of the tag.

Setting Up Virtual Server Components

You can set up the Layer 4 and Layer 7 virtual servers and configure several virtual server components such as, application profiles, persistent profiles, and load balancer rules.

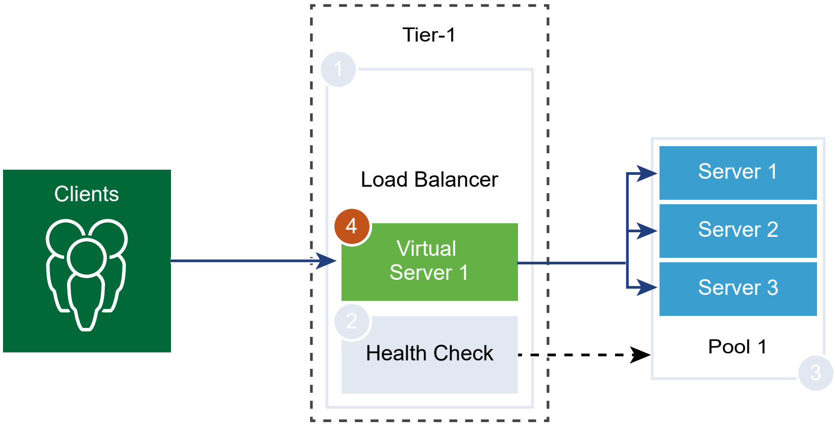
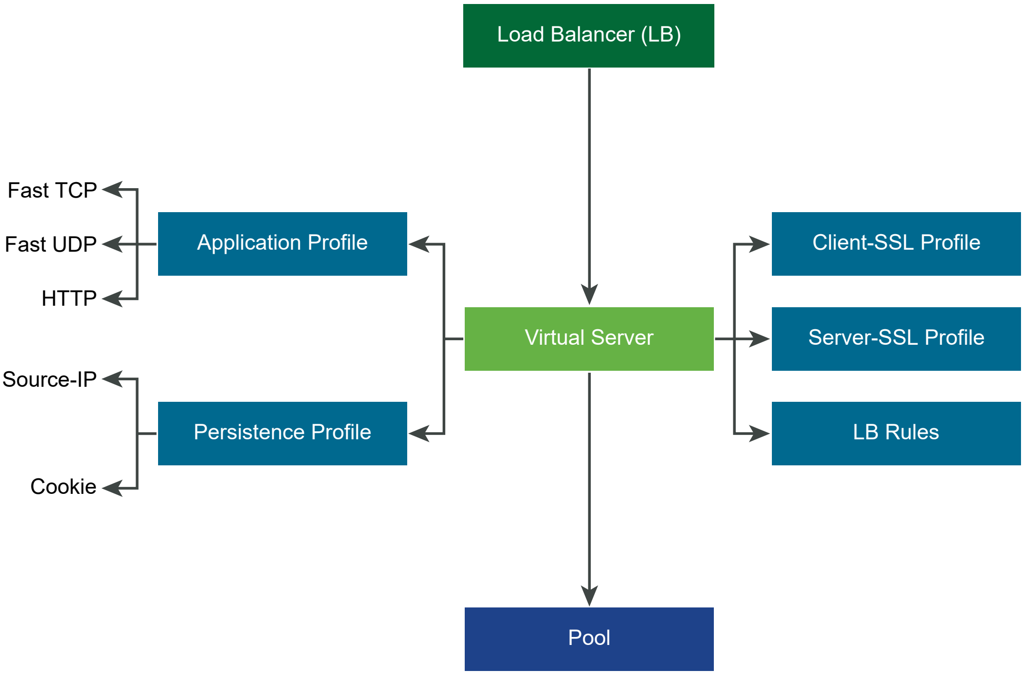


Figure 7-2. Virtual Server Components



Add an Application Profile

Application profiles are associated with virtual servers to enhance load balancing network traffic and simplify traffic-management tasks.

Application profiles define the behavior of a particular type of network traffic. The associated virtual server processes network traffic according to the values specified in the application profile. Fast TCP, Fast UDP, and HTTP application profiles are the supported types of profiles.

TCP application profile is used by default when no application profile is associated to a virtual server. TCP and UDP application profiles are used when an application is running on a TCP or UDP protocol and does not require any application level load balancing such as, HTTP URL load balancing. These profiles are also used when you only want Layer 4 load balancing, which has a faster performance and supports connection mirroring.

HTTP application profile is used for both HTTP and HTTPS applications when the load balancer must take actions based on Layer 7 such as, load balancing all images requests to a specific server pool member or stopping HTTPS to offload SSL from pool members. Unlike the TCP application profile, the HTTP application profile stops the client TCP connection before selecting the server pool member.

Figure 7-3. Layer 4 TCP and UDP Application Profile

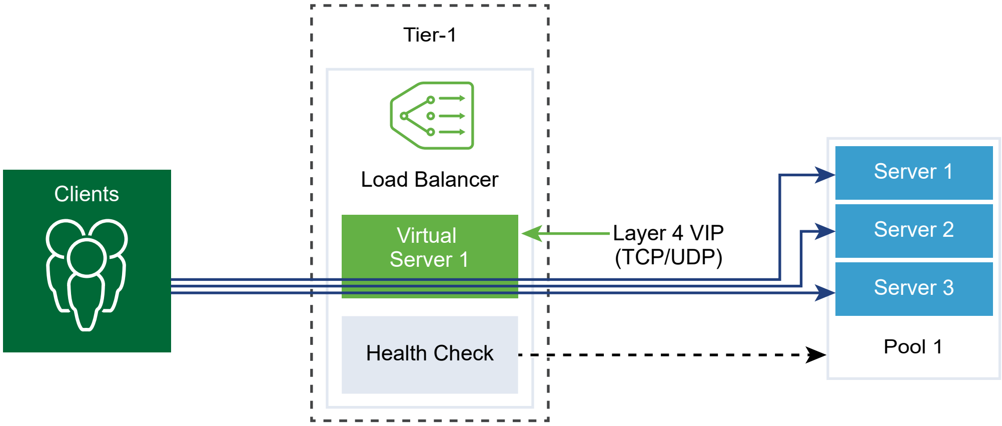
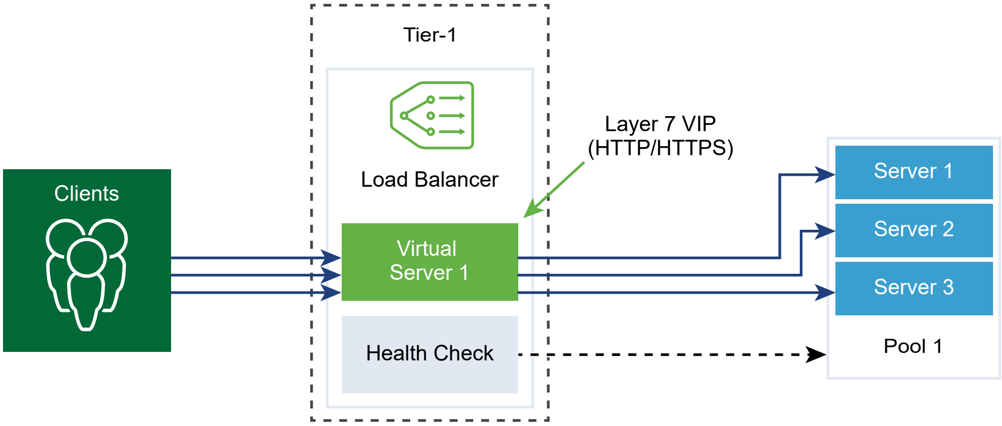


Figure 7-4. Layer 7 HTTPS Application Profile



Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking** > **Load Balancing** > **Profiles** > **Application** > **Add Application Profiles**.

3 Select a **Fast TCP** application profile and enter the profile details.

You can also accept the default FAST TCP profile settings.

Option	Description
Name and Description	Enter a name and a description for the Fast TCP application profile.
Idle Timeout	Enter the time in seconds on how long the server can remain idle after a TCP connection is established. Set the idle time to the actual application idle time and add a few more seconds so that the load balancer does not close its connections before the application does.
HA Flow Mirroring	Toggle the button to make all the flows to the associated virtual server mirrored to the HA standby node.
Connection Close Timeout	Enter the time in seconds that the TCP connection both FINs or RST must be kept for an application before closing the connection. A short closing timeout might be required to support fast connection rates.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

4 Select a **Fast UDP** application profile and enter the profile details.

You can also accept the default UDP profile settings.

Option	Description
Name and Description	Enter a name and a description for the Fast UDP application profile.
Idle Timeout	Enter the time in seconds on how long the server can remain idle after a UDP connection is established. UDP is a connectionless protocol. For load balancing purposes, all the UDP packets with the same flow signature such as, source and destination IP address or ports and IP protocol received within the idle timeout period are considered to belong to the same connection and sent to the same server. If no packets are received during the idle timeout period, the connection which is an association between the flow signature and the selected server is closed.
HA Flow Mirroring	Toggle the button to make all the flows to the associated virtual server mirrored to the HA standby node.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

5 Select a **HTTP** application profile and enter the profile details.

You can also accept the default HTTP profile settings.

HTTP application profile is used for both HTTP and HTTPS applications.

Option	Description
Name and Description	Enter a name and a description for the HTTP application profile.
Idle Timeout	Enter the time in seconds on how long an HTTP application can remain idle, instead of the TCP socket setting which must be configured in the TCP application profile.
Request Header Size	Specify the maximum buffer size in bytes used to store HTTP request headers.
X-Forwarded-For (XFF)	■ **Insert** - If the XFF HTTP header is not present in the incoming request, the load balancer inserts a new XFF header with the client IP address. If the XFF HTTP header is present in the incoming request, the load balancer appends the XFF header with the client IP address. ■ **Replace** - If the XFF HTTP header is present in the incoming request, the load balancer replaces the header. Web servers log each request they handle with the requesting client IP address. These logs are used for debugging and analytics purposes. If the deployment topology requires SNAT on the load balancer, then server uses the client SNAT IP address which defeats the purpose of logging. As a workaround, the load balancer can be configured to insert XFF HTTP header with the original client IP address. Servers can be configured to log the IP address in the XFF header instead of the source IP address of the connection.
Request Body Size	Enter value for the maximum size of the buffer used to store the HTTP request body. If the size is not specified, then the request body size is unlimited.
Redirection	■ None - If a website is temporarily down, user receives a page not found error message. ■ HTTP Redirect - If a website is temporarily down or has moved, incoming requests for that virtual server can be temporarily redirected to a URL specified here. Only a static redirection is supported. For example, if HTTP Redirect is set to http://sitedown.abc.com/sorry.html, then irrespective of the actual request, for example, http://original_app.site.com/home.html or http://original_app.site.com/somepage.html, incoming requests are redirected to the specified URL when the original website is down. ■ HTTP to HTTPS Redirect - Certain secure applications might want to force communication over SSL, but instead of rejecting non-SSL connections, they can redirect the client request to use SSL. With HTTP to HTTPS Redirect, you can preserve both the host and URI paths and redirect the client request to use SSL. For HTTP to HTTPS redirect, the HTTPS virtual server must have port 443 and the same virtual server IP address must be configured on the same load balancer. For example, a client request for http://app.com/path/page.html is redirected to https://app.com/path/page.html. If either the host name or the URI must be modified while redirecting, for example, redirect to https://secure.app.com/path/page.html, then load balancing rules must be used.

Option	Description
NTLM Authentication	Toggle the button for the load balancer to turn off TCP multiplexing and enable HTTP keep-alive. NTLM is an authentication protocol that can be used over HTTP. For load balancing with NTLM authentication, TCP multiplexing must be disabled for the server pools hosting NTLM-based applications. Otherwise, a server-side connection established with one client's credentials can potentially be used for serving another client's requests. If NTLM is enabled in the profile and associated to a virtual server, and TCP multiplexing is enabled at the server pool, then NTLM takes precedence. TCP multiplexing is not performed for that virtual server. However, if the same pool is associated to another non-NTLM virtual server, then TCP multiplexing is available for connections to that virtual server. If the client uses HTTP/1.0, the load balancer upgrades to HTTP/1.1 protocol and the HTTP keep-alive is set. All HTTP requests received on the same client-side TCP connection are sent to the same server over a single TCP connection to ensure that reauthorization is not required.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

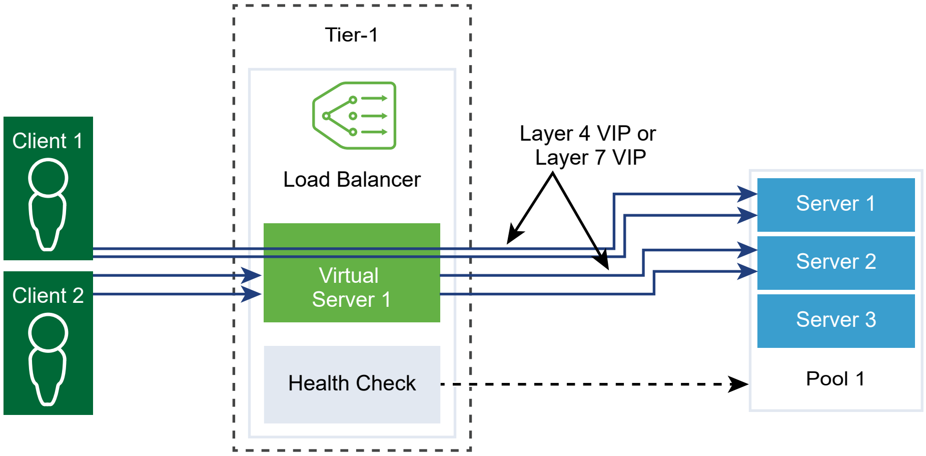
Add a Persistence Profile

To ensure stability of stateful applications, load balancers implement persistence which directs all related connections to the same server. Different types of persistence are supported to address different types of application needs.

Some applications maintain the server state such as, shopping carts. Such state might be per client and identified by the client IP address or per HTTP session. Applications might access or modify this state while processing subsequent related connections from the same client or HTTP session.

The source IP persistence profile tracks sessions based on the source IP address. When a client requests a connection to a virtual server that enables the source address persistence, the load balancer checks if that client was previously connected, if so, returns the client to the same server. If not, you can select a server pool member based on the pool load balancing algorithm. Source IP persistence profile is used by Layer 4 and Layer 7 virtual servers.

The cookie persistence profile inserts a unique cookie to identify the session the first time a client accesses the site. The client forwards the HTTP cookie in subsequent requests and the load balancer uses that information to provide the cookie persistence. Layer 7 virtual servers can only use the cookie persistence profile.



Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking** > **Load Balancing** > **Profiles** > **Persistence** > **Add Persistence Profiles**.

3 Select **Source IP** to add a source IP persistence profile and enter the profile details.

You can also accept the default Source IP profile settings.

Option	Description
Name and Description	Enter a name and a description for the Source IP persistence profile.
Share Persistence	Toggle the button to share the persistence so that all virtual servers this profile is associated with can share the persistence table. If the persistence sharing is not enabled in the Source IP persistence profile associated to a virtual server, each virtual server that the profile is associated to maintain a private persistence table.
Persistence Entry Timeout	Enter the persistence expiration time in seconds. The load balancer persistence table maintains entries to record that client requests are directed to the same server. ■ If no new connection requests are received from the same client within the timeout period, the persistence entry expires and is deleted. ■ If a new connection request from the same client is received within the timeout period, the timer is reset, and the client request is sent to a sticky pool member. After the timeout period has expired, new connection requests are sent to a server allocated by the load balancing algorithm. For the L7 load balancing TCP source IP persistence scenario, the persistence entry times out if no new TCP connections are made for some time, even if the existing connections are still alive.
Purge Entries When Full	Toggle the button to purge entries when the persistence table is full. A large timeout value might lead to the persistence table quickly filling up when the traffic is heavy. When the persistence table fills up, the oldest entry is deleted to accept the newest entry.

Option	Description
HA Persistence Mirroring	Toggle the button to synchronize persistence entries to the HA peer.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

4 Select a **Cookie** persistence profile and enter the profile details.

Option	Description
Name and Description	Enter a name and a description for the Cookie persistence profile.
Share Persistence	Toggle the button to share persistence across multiple virtual servers that are associated to the same pool members. The Cookie persistence profile inserts a cookie with the format, *<name>.<profile-id>.<pool-id>*. If the persistence shared is not enabled in the Cookie persistence profile associated with a virtual server, the private Cookie persistence for each virtual server is used and is qualified by the pool member. The load balancer inserts a cookie with the format, *<name>.<virtual_server_id>.<pool_id>*.
Cookie Mode	Select a mode from the drop-down menu. ■ INSERT - Adds a unique cookie to identify the session. ■ PREFIX - Appends to the existing HTTP cookie information. ■ REWRITE - Rewrites the existing HTTP cookie information.
Cookie Name	Enter the cookie name.
Cookie Domain	Enter the domain name. HTTP cookie domain can be configured only in the INSERT mode.
Cookie Fallback	Toggle the button so that the client request is rejected if cookie points to a server that is in a DISABLED or is in a DOWN state. Selects a new server to handle a client request if the cookie points to a server that is in a DISABLED or is in a DOWN state.
Cookie Path	Enter the cookie URL path. HTTP cookie path can be set only in the INSERT mode.
Cookie Garbling	Toggle the button to disable encryption. When garbling is disabled, the cookie server IP address and port information is in a plain text. Encrypt the cookie server IP address and port information.
Cookie Type	Select a cookie type from the drop-down menu. **Session Cookie** - Not stored. Will be lost when the browser is closed. **Persistence Cookie** - Stored by the browser. Not lost when the browser is closed.
Max Idle Time	Enter the time in seconds that the cookie type can be idle before a cookie expires.
Max Cookie Age	For the session cookie type, enter the time in seconds a cookie is available.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

Add an SSL Profile

SSL profiles configure application-independent SSL properties such as, cipher lists and reuse these lists across multiple applications. SSL properties are different when the load balancer is acting as a client and as a server, as a result separate SSL profiles for client-side and server-side are supported.

Note SSL profile is not supported in the NSX-T Data Center limited export release.

Client-side SSL profile refers to the load balancer acting as an SSL server and stopping the client SSL connection. Server-side SSL profile refers to the load balancer acting as a client and establishing a connection to the server.

You can specify a cipher list on both the client-side and server-side SSL profiles.

SSL session caching allows the SSL client and server to reuse previously negotiated security parameters avoiding the expensive public key operation during the SSL handshake. SSL session caching is disabled by default on both the client-side and server-side.

SSL session tickets are an alternate mechanism that allows the SSL client and server to reuse previously negotiated session parameters. In SSL session tickets, the client and server negotiate whether they support SSL session tickets during the handshake exchange. If supported by both, server can send an SSL ticket, which includes encrypted SSL session parameters to the client. The client can use that ticket in subsequent connections to reuse the session. SSL session tickets are enabled on the client-side and disabled on the server-side.

Figure 7-5. SSL Offloading

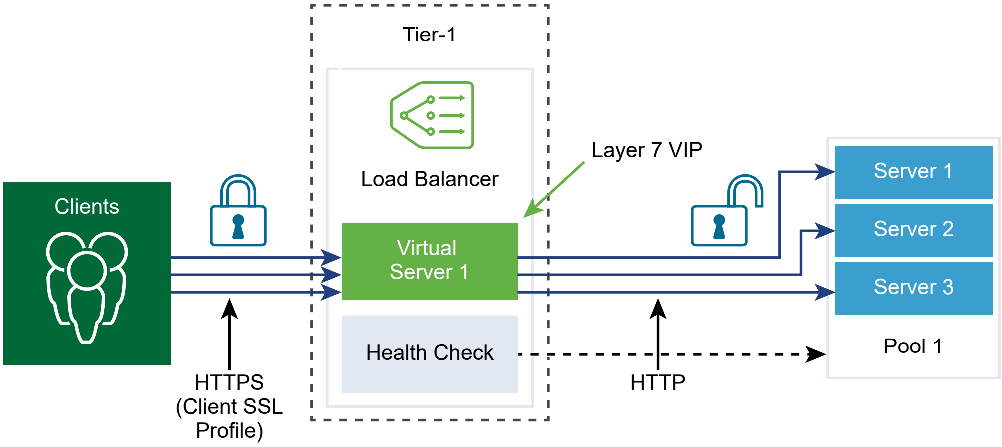
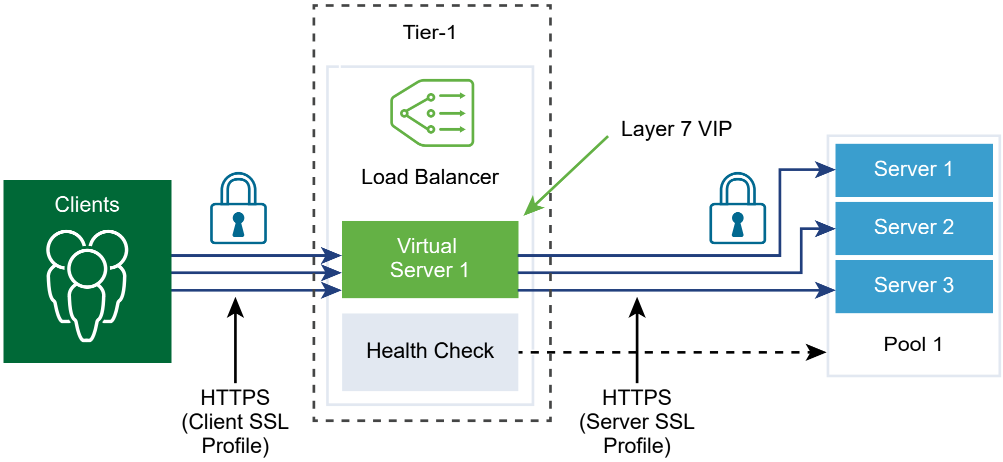


Figure 7-6. End-to-End SSL



Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > Load Balancing > Profiles > SSL Profile**.
3. Select a **Client SSL Profile** and enter the profile details.

Option	Description
Name and Description	Enter a name and a description for the Client SSL profile.
SSL Suite	Select the SSL Cipher group from the drop-down menu and available SSL Ciphers and SSL protocols to be included in the Client SSL profile are populated. Balanced SSL Cipher group is the default.
Session Caching	Toggle the button to allow the SSL client and server to reuse previously negotiated security parameters avoiding the expensive public key operation during an SSL handshake.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.
Supported SSL Ciphers	Depending on the SSL suite, you assigned the supported SSL Ciphers are populated here. Click **View More** to view the entire list. If you selected **Custom**, you must select the SSL Ciphers from the drop-down menu.
Supported SSL Protocols	Depending on the SSL suite, you assigned the supported SSL protocols are populated here. Click **View More** to view the entire list. If you selected **Custom**, you must select the SSL Ciphers from the drop-down menu.

Option	Description
Session Cache Entry Timeout	Enter the cache timeout in seconds to specify how long the SSL session parameters must be kept and can be reused.
Prefer Server Cipher	Toggle the button so that the server can select the first supported cipher from the list it can support. During an SSL handshake, the client sends an ordered list of supported ciphers to the server.

4 Select a **Server SSL Profile** and enter the profile details.

Option	Description
Name and Description	Enter a name and a description for the Server SSL profile.
SSL Suite	Select the SSL Cipher group from the drop-down menu and available SSL Ciphers and SSL protocols to be included in the Server SSL profile are populated. Balanced SSL Cipher group is the default.
Session Caching	Toggle the button to allow the SSL client and server to reuse previously negotiated security parameters avoiding the expensive public key operation during an SSL handshake.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.
Supported SSL Ciphers	Depending on the SSL suite, you assigned the supported SSL Ciphers are populated here. Click **View More** to view the entire list. If you selected **Custom**, you must select the SSL Ciphers from the drop-down menu.
Supported SSL Protocols	Depending on the SSL suite, you assigned the supported SSL protocols are populated here. Click **View More** to view the entire list. If you selected **Custom**, you must select the SSL Ciphers from the drop-down menu.
Session Cache Entry Timeout	Enter the cache timeout in seconds to specify how long the SSL session parameters must be kept and can be reused.
Prefer Server Cipher	Toggle the button so that the server can select the first supported cipher from the list it can support. During an SSL handshake, the client sends an ordered list of supported ciphers to the server.

Add Layer 4 Virtual Servers

Virtual servers receive all the client connections and distribute them among the servers. A virtual server has an IP address, a port, and a protocol. For Layer 4 virtual servers, lists of ports ranges can be specified instead of a single TCP or UDP port to support complex protocols with dynamic ports.

A Layer 4 virtual server must be associated to a primary server pool, also called a default pool.

If a virtual server status is disabled, any new connection attempts to the virtual server are rejected by sending either a TCP RST for the TCP connection or ICMP error message for UDP. New connections are rejected even if there are matching persistence entries for them. Active connections continue to be processed. If a virtual server is deleted or disassociated from a load balancer, then active connections to that virtual server fail.

Prerequisites

- Verify that application profiles are available. See Add an Application Profile.
- Verify that persistent profiles are available. See Add a Persistence Profile.
- Verify that SSL profiles for the client and server are available. See Add an SSL Profile.
- Verify that server pools are available. See Add a Server Pool.
- Verify that load balancer is available. See Add Load Balancers.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > Load Balancing > Virtual Servers > Add Virtual Server**.
3. Select a **L4 TCP** protocol and enter the protocol details.

   Layer 4 virtual servers support either the Fast TCP or Fast UDP protocol, but not both.

   For Fast TCP or Fast UDP protocol support on the same IP address and port, for example DNS, a virtual server must be created for each protocol.

Option	Description
Name and Description	Enter a name and a description for the Layer 4 virtual server.
IP Address	Enter the virtual server IP address.
Ports	Enter the virtual server port number.
Load Balancer	Select an existing load balancer to attach to this Layer 4 virtual server from the drop down menu.
Server Pool	Select an existing server pool from the drop-down menu. The server pool consists of one or more servers, also called pool members that are similarly configured and running the same application. You can click the vertical ellipses to create a server pool.
Application Profile	Based on the protocol type, the existing application profile is automatically populated. You can click the vertical ellipses to create an application profile.
Persistence	Select an existing persistence profile from the drop-down menu. Persistence profile can be enabled on a virtual server to allow Source IP related client connections to be sent to the same server.

Option	Description
Max Concurrent Connection	Set the maximum concurrent connection allowed to a virtual server so that the virtual server does not deplete resources of other applications hosted on the same load balancer.
Max New Connection Rate	Set the maximum new connection to a server pool member so that a virtual server does not deplete resources.
Sorry Server Pool	Select an existing sorry server pool from the drop-down menu. The sorry server pool serves the request when a load balancer cannot select a backend server to the serve the request from the default pool. You can click the vertical ellipses to create a server pool.
Default Pool Member Port	Enter a default pool member port if the pool member port for a virtual server is not defined. For example, if a virtual server is defined with port range 2000-2999 and the default pool member port range is set as 8000-8999, then an incoming client connection to the virtual server port 2500 is sent to a pool member with a destination port set to 8500.
Admin State	Toggle the button to disable the admin state of the Layer 4 virtual server.
Access Log	Toggle the button to enable logging for the Layer 4 virtual server.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

4 Select a **L4 UDP** protocol and enter the protocol details.

Option	Description
Name and Description	Enter a name and a description for the Layer 4 virtual server.
IP Address	Enter the virtual server IP address.
Ports	Enter the virtual server port number.
Load Balancer	Select an existing load balancer to attach to this Layer 4 virtual server from the drop down menu.
Server Pool	Select an existing server pool from the drop-down menu. The server pool consists of one or more servers, also called pool members that are similarly configured and running the same application. You can click the vertical ellipses to create a server pool.
Application Profile	Based on the protocol type, the existing application profile is automatically populated. You can click the vertical ellipses to create an application profile.
Persistence	Select an existing persistence profile from the drop-down menu. Persistence profile can be enabled on a virtual server to allow Source IP related client connections to be sent to the same server.
Max Concurrent Connection	Set the maximum concurrent connection allowed to a virtual server so that the virtual server does not deplete resources of other applications hosted on the same load balancer.
Max New Connection Rate	Set the maximum new connection to a server pool member so that a virtual server does not deplete resources.

Option	Description
Sorry Server Pool	Select an existing sorry server pool from the drop-down menu. The sorry server pool serves the request when a load balancer cannot select a backend server to the serve the request from the default pool. You can click the vertical ellipses to create a server pool.
Default Pool Member Port	Enter a default pool member port if the pool member port for a virtual server is not defined. For example, if a virtual server is defined with port range 2000-2999 and the default pool member port range is set as 8000-8999, then an incoming client connection to the virtual server port 2500 is sent to a pool member with a destination port set to 8500.
Admin State	Toggle the button to disable the admin state of the Layer 4 virtual server.
Access Log	Toggle the button to enable logging for the Layer 4 virtual server.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

Add Layer 7 HTTP Virtual Servers

Virtual servers receive all the client connections and distribute them among the servers. A virtual server has an IP address, a port, and a protocol TCP.

Load balancer rules are supported for only Layer 7 virtual servers with an HTTP application profile. Different load balancer services can use load balancer rules.

Each load balancer rule consists of single or multiple match conditions and single or multiple actions. If the match conditions are not specified, then the load balancer rule always matches and is used to define default rules. If more than one match condition is specified, then the matching strategy determines if all conditions must match or any one condition must match for the load balancer rule to be considered a match.

Each load balancer rule is implemented at a specific phase of the load balancing processing; HTTP Request Rewrite, HTTP Request Forwarding, and HTTP Response Rewrite. Not all the match conditions and actions are applicable to each phase.

Note Layer 7 SSL passthrough is supported in NSX-T Data Center 3.0 and later.

If a virtual server status is disabled, any new connection attempts to the virtual server are rejected by sending either a TCP RST for the TCP connection or ICMP error message for UDP. New connections are rejected even if there are matching persistence entries for them. Active connections continue to be processed. If a virtual server is deleted or disassociated from a load balancer, then active connections to that virtual server fail.

Note SSL profile is not supported in the NSX-T Data Center limited export release.

If a client-side SSL profile binding is configured on a virtual server but not a server-side SSL profile binding, then the virtual server operates in an SSL-terminate mode, which has an encrypted connection to the client and plain text connection to the server. If both the client-side and server-side SSL profile bindings are configured, then the virtual server operates in SSL-proxy mode, which has an encrypted connection both to the client and the server.

Associating server-side SSL profile binding without associating a client-side SSL profile binding is currently not supported. If a client-side and a server-side SSL profile binding is not associated with a virtual server and the application is SSL-based, then the virtual server operates in an SSL-unaware mode. In this case, the virtual server must be configured for Layer 4. For example, the virtual server can be associated to a fast TCP profile.

Prerequisites

- Verify that application profiles are available. See Add an Application Profile.
- Verify that persistent profiles are available. See Add a Persistence Profile.
- Verify that SSL profiles for the client and server are available. See Add an SSL Profile.
- Verify that server pools are available. See Add a Server Pool.
- Verify that CA and client certificate are available. See Create a Certificate Signing Request File.
- Verify that a certification revocation list (CRL) is available. See Import a Certificate Revocation List.
- Verify that load balancer is available. See Add Load Balancers.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > Load Balancing > Virtual Servers > Add Virtual Server**.
3. Select a **L7 HTTP** protocol and enter the protocol details.

   Layer 7 virtual servers support the HTTP and HTTPS protocols.

Option	Description
Name and Description	Enter a name and a description for the Layer virtual server.
IP Address	Enter the virtual server IP address.
Ports	Enter the virtual server port number.
Load Balancer	Select an existing load balancer to attach to this Layer 4 virtual server from the drop down menu.
Server Pool	Select an existing server pool from the drop-down menu. The server pool consists of one or more servers, also called pool members that are similarly configured and running the same application. You can click the vertical ellipses to create a server pool.

Option	Description
Application Profile	Based on the protocol type, the existing application profile is automatically populated. You can click the vertical ellipses to create an application profile.
Persistence	Select an existing persistence profile from the drop-down menu. Persistence profile can be enabled on a virtual server to allow Source IP and Cookie related client connections to be sent to the same server.

4 Click **Configure** to set the Layer 7 virtual server SSL.

You can configure the Client SSL and Server SSL.

5 Configure the Client SSL.

Option	Description
Client SSL	Toggle the button to enable the profile. Client-side SSL profile binding allows multiple certificates, for different host names to be associated to the same virtual server.
Default Certificate	Select a default certificate from the drop-down menu. This certificate is used if the server does not host multiple host names on the same IP address or if the client does not support Server Name Indication (SNI) extension.
Client SSL Profile	Select the Client-side SSL Profile from the drop-down menu.
SNI Certificates	Select the available SNI certificate from the drop-down menu.
Trusted CA Certificates	Select the available CA certificate.
Mandatory Client Authentication	Toggle the button to enable this menu item.
Certificate Chain Depth	Set the certificate chain depth to verify the depth in the server certificates chain.
Certificate Revocation List	Select the available CRL to disallow compromised server certificates.

6 Configure the Server SSL.

Option	Description
Server SSL	Toggle the button to enable the profile.
Client Certificate	Select a client certificate from the drop-down menu. This certificate is used if the server does not host multiple host names on the same IP address or if the client does not support Server Name Indication (SNI) extension.
Server SSL Profile	Select the Server-side SSL Profile from the drop-down menu.
Trusted CA Certificates	Select the available CA certificate.

Option	Description
Mandatory Server Authentication	Toggle the button to enable this menu item. Server-side SSL profile binding specifies whether the server certificate presented to the load balancer during the SSL handshake must be validated or not. When validation is enabled, the server certificate must be signed by one of the trusted CAs whose self-signed certificates are specified in the same server-side SSL profile binding.
Certificate Chain Depth	Set the certificate chain depth to verify the depth in the server certificates chain.
Certificate Revocation List	Select the available CRL to disallow compromised server certificates. OCSP and OCSP stapling are not supported on the server-side.

7 Configure additional Layer 7 virtual server properties.

Option	Description
Max Concurrent Connection	Set the maximum concurrent connection allowed to a virtual server so that the virtual server does not deplete resources of other applications hosted on the same load balancer.
Max New Connection Rate	Set the maximum new connection to a server pool member so that a virtual server does not deplete resources.
Sorry Server Pool	Select an existing sorry server pool from the drop-down menu. The sorry server pool serves the request when a load balancer cannot select a backend server to the serve the request from the default pool. You can click the vertical ellipses to create a server pool.
Default Pool Member Port	Enter a default pool member port if the pool member port for a virtual server is not defined. For example, if a virtual server is defined with port range 2000-2999 and the default pool member port range is set as 8000-8999, then an incoming client connection to the virtual server port 2500 is sent to a pool member with a destination port set to 8500.
Admin State	Toggle the button to disable the admin state of the Layer 7 virtual server.
Access Log	Toggle the button to enable logging for the Layer 7 virtual server.
Tags	Enter tags to make searching easier. You can specify a tag to set a scope of the tag.

Add Load Balancer Rules

With Layer 7 HTTP virtual servers, you can optionally configure load balancer rules and customize load balancing behavior using match or action rules.

Load Balancer rules support REGEX for match types. PCRE style REGEX patters is supported with a few limitations on advanced use cases. When REGEX is used in match conditions, named capturing groups are supported.

REGEX restrictions include:

- Character unions and intersections are not supported. For example, do not use [a-z[0-9]] and [a-z&&[aeiou]] instead use [a-z0-9] and [aeiou] respectively.
- Only 9 back references are supported and \1 through \9 can be used to refer to them.
- Use \0dd format to match octal characters, not the \ddd format.
- Embedded flags are not supported at the top level, they are only supported within groups. For example, do not use "Case (?i:s)ensitive" instead use "Case ((?i:s)ensitive)".
- Preprocessing operations \l, \u, \L, \U are not supported. Where \l - lowercase next char \u - uppercase next char \L - lower case until \E \U - upper case to \E.
- (?(condition)X), (?{code}), (??{Code}) and (?#comment) are not supported.
- Predefined Unicode character class \X is not supported
- Using named character construct for Unicode characters is not supported. For example, do not use \N{name} instead use \u2018.

When REGEX is used in match conditions, named capturing groups are supported. For example, REGEX match pattern /news/(?<year>\d+)-(?<month>\d+)-(?<day>\d+)/(?<article>.*) can be used to match a URI like /news/2018-06-15/news1234.html.

Then variables are set as follows, $year = "2018" $month = "06" $day = "15" $article = "news1234.html". After the variables are set, these variables can be used in load balancer rule actions. For example, URI can be rewritten using the matched variables like, /news.py?year=$year&month=$month&day=$day&article=$article. Then the URI gets rewritten as /news.py?year=2018&month=06&day=15&article=news1234.html.

Rewrite actions can use a combination of named capturing groups and built-in variables. For example, URI can be written as /news.py?year=$year&month=$month&day=$day&article=$article&user_ip=$_remote_addr. Then the example URI gets rewritten as /news.py?year=2018&month=06&day=15&article=news1234.html&user_ip=1.1.1.1.

Note For named capturing groups, the name cannot start with an _ character.

In addition to named capturing groups, the following built-in variables can be used in rewrite actions. All the built-in variable names start with _.

- $_args - arguments from the request
- $_cookie_<name> - value of <name> cookie
- $_host - in the order of precedence - host name from the request line, or host name from the "Host" request header field, or the server name matching a request
- $_hostname - host name
- $_http_<name> - arbitrary request header field and <name> is the field name converted to lower case with dashes replaced by underscores
- $_https - "on" if connection operates in SSL mode, or "" otherwise

- $_is_args - "?" if a request line has arguments, or "" otherwise
- $_query_string - same as $_args
- $_remote_addr - client address
- $_remote_port - client port
- $_request_uri - full original request URI (with arguments)
- $_scheme - request scheme, "http" or "https"
- $_server_addr - address of the server which accepted a request
- $_server_name - name of the server which accepted a request
- $_server_port - port of the server which accepted a request
- $_server_protocol - request protocol, usually "HTTP/1.0" or "HTTP/1.1"
- $_ssl_client_cert - returns the client certificate in the PEM format for an established SSL connection, with each line except the first prepended with the tab character
- $_ssl_server_name - returns the server name requested through SNI
- $_uri - URI path in request

Prerequisites

Verify a Layer 7 HTTP virtual server is available. See Add Layer 7 HTTP Virtual Servers.

Procedure

1. Open the Layer 7 HTTP virtual server.
2. In the Load Balancer Rules section, click **Set > Add Rule** to configure the load balancer rules for the HTTP Request Rewrite phase.

   Supported match types are, REGEX, STARTS_WITH, ENDS_WITH, etc and inverse option.

Supported Match Condition	Description
HTTP Request Method	Match an HTTP request method. http_request.method - value to match
HTTP Request URI	Match an HTTP request URI without query arguments. http_request.uri - value to match
HTTP Request URI Arguments	Match an HTTP request URI query argument. http_request.uri_arguments - value to match
HTTP Request Version	Match an HTTP request version. http_request.version - value to match
HTTP Request Header	Match any HTTP request header. http_request.header_name - header name to match http_request.header_value - value to match

Supported Match Condition	Description
HTTP Request Cookie	Match any HTTP request cookie. http_request.cookie_value - value to match
HTTP Request Body	Match an HTTP request body content. http_request.body_value - value to match
Client SSL	Match client SSL profile ID. ssl_profile_id - value to match
TCP Header Port	Match a TCP source or the destination port. tcp_header.source_port - source port to match tcp_header.destination_port - destination port to match
IP Header Source	Match an IP source or destination address. ip_header.source_address - source address to match ip_header.destination_address - destination address to match
Variable	Create a variable and assign a value to the variable.
Case Sensitive	Set a case-sensitive flag for HTTP header value comparison.

Actions	Description
HTTP Request URI Rewrite	Modify an URI. http_request.uri - URI (without query arguments) to write http_request.uri_args - URI query arguments to write
HTTP Request Header Rewrite	Modify value of an HTTP header. http_request.header_name - header name http_request.header_value - value to write
HTTP Request Header Delete	Delete HTTP header. http_request.header_delete - header name http_request.header_delete - value to write

3 Click **Request Forwarding > Add Rule** to configure the load balancer rules for the HTTP Request Forwarding.

All match values accept regular expressions.

Supported Match Condition	Description
HTTP Request Method	Match an HTTP request method. http_request.method - value to match
HTTP Request URI	Match an HTTP request URI. http_request.uri - value to match
HTTP Request Version	Match an HTTP request version. http_request.version - value to match
HTTP Request Header	Match any HTTP request header. http_request.header_name - header name to match http_request.header_value - value to match

Supported Match Condition	Description
HTTP Request Cookie	Match any HTTP request cookie. http_request.cookie_value - value to match
HTTP Request Body	Match an HTTP request body content. http_request.body_value - value to match
Client SSL	Match client SSL profile ID. ssl_profile_id - value to match
TCP Header Port	Match a TCP source or the destination port. tcp_header.source_port - source port to match tcp_header.destination_port - destination port to match
IP Header Source	Match an IP source or destination address. ip_header.source_address - source address to match ip_header.destination_address - destination address to match
Variable	Create a variable and assign a value to the variable.
Case Sensitive	Set a case-sensitive flag for HTTP header value comparison.

Action	Description
HTTP Reject	Reject a request, for example, by setting status to 5xx. http_forward.reply_status - HTTP status code used to reject http_forward.reply_message - HTTP rejection message
HTTP Redirect	Redirect a request. Status code must be set to 3xx. http_forward.redirect_status - HTTP status code for redirect http_forward.redirect_url - HTTP redirect URL
Select Pool	Force the request to a specific server pool. Specified pool member's configured algorithm (predictor) is used to select a server within the server pool. http_forward.select_pool - server pool UUID
Reply Status	Shows the status of the reply.
Reply Message	Server responds with a reply message that contains confirmed addresses and configuration.

4 Click **Response Rewrite > Add Rule** to configure the load balancer rules for the HTTP Response Rewrite.

All match values accept regular expressions.

Supported Match Condition	Description
HTTP Response Header	Match any HTTP response header. http_response.header_name - header name to match http_response.header_value - value to match
HTTP Response Method	Match an HTTP response method. http_response.method - value to match

Supported Match Condition	Description
HTTP Response URI	Match an HTTP response URI. http_response.uri - value to match
HTTP Response URI Arguments	Match an HTTP response URI arguments. http_response.uri_args - value to match
HTTP Response Version	Match an HTTP response version. http_response.version - value to match
HTTP Response Cookie	Match any HTTP response cookie. http_response.cookie_value - value to match
Client SSL	Match client SSL profile ID. ssl_profile_id - value to match
TCP Header Port	Match a TCP source or the destination port. tcp_header.source_port - source port to match tcp_header.destination_port - destination port to match
IP Header Source	Match an IP source or destination address. ip_header.source_address - source address to match ip_header.destination_address - destination address to match
Variable	Create a variable and assign a value to the variable.
Case Sensitive	Set a case-sensitive flag for HTTP header value comparison.

Action	Description
HTTP Response Header Rewrite	Modify the value of an HTTP response header. http_response.header_name - header name http_response.header_value - value to write
HTTP Response Header Delete	Delete HTTP header. http_request.header_delete - header name http_request.header_delete - value to write

Forwarding Policies

8

This feature pertains to NSX Cloud.

Forwarding Policies or Policy-Based Routing (PBR) rules define how NSX-T handles traffic from an NSX-managed VM. This traffic can be steered to NSX-T overlay or it can be routed through the cloud provider's (underlay) network.

Note Your public cloud workload VMs are managed by NSX-T after you tag them with `nsx.network=default` in your public cloud and install the NSX agent on them. See Onboard Workload VMs for details.

Three default forwarding policies are set up automatically after you either deploy a PCG on a Transit VPC/VNet or link a Compute VPC/VNet to the Transit.

1. **Route to Underlay** for all traffic that is addressed within the Transit/Compute VPC/VNet
2. **Route to Underlay** for all traffic destined to the metadata services of the public cloud.
3. **Route to Overlay** for all other traffic, for example, traffic that is headed outside the Transit/Compute VPC/VNet. Such traffic is routed over the NSX-T overlay tunnel to the PCG and further to its destination.

   Note **For traffic destined to another VPC/VNET managed by the same PCG**: Traffic is routed from the source NSX-managed VPC/VNet via the NSX-T overlay tunnel to the PCG and then routed to the destination VPC/VNet.

   For traffic destined to another VPC/VNet managed by a different PCG: Traffic is routed from one NSX-managed VPC/VNet over the NSX overlay tunnel to the PCG of the source VPC/VNet and forwarded to the PCG of the destination NSX-managed VPC/VNet.

   If traffic is headed to the internet, the PCG routes it to the destination in the internet.

Micro-segmentation while Routing to Underlay

Micro-segmentation is enforced even for workload VMs whose traffic is routed to the underlay network.

If you have direct connectivity from an NSX-managed workload VM to a destination outside the managed VPC/VNet and want to bypass the PCG, set up a forwarding policy to route traffic from this VM via underlay.

When traffic is routed through the underlay network, the PCG is bypassed and therefore the north-south firewall is not encountered by traffic. However, you still have to manage rules for east-west or distributed firewall (DFW) because those rules are applied at the VM-level before reaching the PCG.

Currently Supported Forwarding Policies

You may see a list of forwarding policies in the drop-down menu but in this release only the following forwarding policies are supported:

- **Route to Underlay**: Access a service on underlay from an NSX-managed VM. For example, access to the AWS S3 service on the AWS underlay network.
- **Route from Underlay**: Access a service hosted on an NSX-managed VM from the underlay network. For example, access from AWS ELB to the NSX-managed VM.

This chapter includes the following topics:

- Add or Edit Forwarding Policies

Add or Edit Forwarding Policies

You can edit the auto-created forwarding policies or add new ones.

For example, to use services provided by the public cloud, such as S3 by AWS, you can manually create a policy to allow a set of IP addresses to access this service by being routed through underlay.

Prerequisites

You must have a VPC or VNet with a PCG deployed on it.

Procedure

1. Click **Add Section**. Name the section appropriately, for example, `AWS Services`.
2. Select the check box next to the section and click **Add Rule**. Name the rule, for example, `S3 Rules`.
3. In the **Sources** tab, select the VPC or VNet where you have the workload VMs to which you want to provide the service access, for example, the AWS VPC. You can also create a **Group** here to include multiple VMs matching one or more criteria.
4. In the **Destinations** tab, select the VPC or VNet where the service is hosted, for example, a **Group** that contains the IP address of the S3 service in AWS.

5 In the **Services** tab, select the service from the drop-down menu. If the service does not exist, you can add it. You can also leave the selection to **Any** because you can provide the routing details under **Destinations**.

6 In the **Action** tab, select how you want the routing to work, for example, select **Route to Underlay** if setting up this policy for the AWS S3 service.

7 Click **Publish** to finish setting up the Forwarding Policy.

IP Address Management (IPAM)

9

To manage IP addresses, you can configure DNS (Domain Name System), DHCP (Dynamic Host Configuration Protocol), IP address pools, and IP address blocks.

Note IP blocks are used by NSX Container Plug-in (NCP). For more info about NCP, see the *NSX Container Plug-in for Kubernetes and Cloud Foundry - Installation and Administration Guide*.

This chapter includes the following topics:

Add a DNS Zone

You can configure DNS zones for your DNS service. A DNS zone is a distinct portion of the domain name space in DNS.

When you configure a DNS zone, you can specify a source IP for a DNS forwarder to use when forwarding DNS queries to an upstream DNS server. If you do not specify a source IP, the DNS query packet's source IP will be the DNS forwarder's listener IP. Specifying a source IP is needed if the listener IP is an internal address that is not reachable from the external upstream DNS server. To ensure that the DNS response packets are routed back to the forwarder, a dedicated source IP is needed. Alternatively, you can configure SNAT on the logical router to translate the listener IP to a public IP. In this case, you do not need to specify a source IP.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > IP Address Management > DNS**.

3 Click the **DNS Zones** tab.

4 To add a default zone, select **Add DNS Zone > Add Default Zone**

  a Enter a name and optionally a description.

  b Enter the IP address of up to three DNS servers.

  c (Optional) Enter an IP address in the **Source IP** field.

5 To add an FQDN zone, select **Add DNS Zone > Add FQDN Zone**

  a Enter a name and optionally a description.

  b Enter a FQDN for the domain.

  c Enter the IP address of up to three DNS servers.

  d (Optional) Enter an IP address in the **Source IP** field.

6 Click **Save**.

Add a DNS Forwarder Service

You can configure a DNS forwarder to forward DNS queries to external DNS servers.

Before you configure a DNS forwarder, you must configure a default DNS zone. Optionally, you can configure one or more FQDN DNS zones. Each DNS zone is associated with up to 3 DNS servers. When you configure a FQDN DNS zone, you specify one or more domain names. A DNS forwarder is associated with a default DNS zone and up to 5 FQDN DNS zones. When a DNS query is received, the DNS forwarder compares the domain name in the query with the domain names in the FQDN DNS zones. If a match is found, the query is forwarded to the DNS servers specified in the FQDN DNS zone. If a match is not found, the query is forwarded to the DNS servers specified in the default DNS zone.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > IP Address Management > DNS**.

3 Click **Add DNS Service**.

4 Enter a name and optionally a description.

5 Select a tier-0 or tier-1 gateway.

6 Enter the IP address of the DNS service.

  Clients send DNS queries to this IP address, which is also known as the DNS forwarder's listener IP.

7 Select a default DNS zone.

8 Select a log level.

9 Select up to five FQDN zones.

10 Click the **Admin Status** toggle to enable or disable the DNS service.

11 Click **Save**.

Add a DHCP Server

DHCP (Dynamic Host Configuration Protocol) allows clients to automatically obtain network configuration, such as IP address, subnet mask, default gateway, and DNS configuration, from a DHCP server. You can create DHCP servers to handle DHCP requests.

Note The DHCP server that is created using this procedure is not supported on a VLAN-backed segment. You must use the DHCP feature under **Advanced Networking & Security** to create a DHCP server that is supported on a VLAN-backed logical switch.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > IP Address Management > DHCP**.

3 Click **Add Server**.

4 Select **DHCP Server** as the server type.

5 Enter a name for the server.

6 Enter the server's IP address in CIDR format.

   This step will create a two logical ports (one for a logical interface and one for the DHCP server itself) and connect the DHCP server to a specific DHCP logical switch. This interface will appear on the tier-0 or tier-1 gateway as a connected interface, so make sure you choose a non-overlapping subnet for the tier-1 or tier-0 gateway that you want to assign the DHCP server to. You can specify <IP address>/30 for this purpose. The subnet range used here does not get advertised to the connected tier-0 gateway, but does appear in the tier-1 gateway's forwarding table.

7 Enter a lease time.

8 Select an NSX Edge cluster.

9 Click **Save**.

10 To assign a DHCP server to a tier-0 or tier-1 gateway:

   a Navigate to **Networking > Tier-0 Gateways** or **Networking > Tier-1 Gateways**.

   b Edit an existing gateway.

   c In the **IP Address Management** field, click **No IP Allocation**.

   d Select **DHCP Local Server** from the Type dropdown list.

   e Select a DHCP server.
   f Click **Save**.
   g Click **Save**.

11 To assign a DHCP server to a segment:

   a Navigate to **Networking > Segments**.
   b Add or edit a segment.

     The segment must be associated with a tier-0 or tier-1 gateway.

   c Click **Set Subnets** if you are adding a new segment, or click the number under **Subnets** to add or modify a subnet.
   d Enter the appropriate DHCP ranges.
   e Click **Apply**.
   f Click **Save**.

Configure a DHCP Relay Server for a Tier-0 or Tier-1 Gateway

DHCP (Dynamic Host Configuration Protocol) allows clients to automatically obtain network configuration, such as IP address, subnet mask, default gateway, and DNS configuration, from a DHCP server. You can create a DHCP relay server to relay DHCP traffic to external DHCP servers.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > IP Address Management > DHCP**.
3. Click **Add Server**.
4. Select **DHCP Relay** as the server type.
5. Enter a name for the relay server.
6. Enter one or more IP addresses for the server.
7. Click **Save**.
8. Go to **Networking > Tier-0 Gateways** or **Networking > Tier-1 Gateways** to configure a DHCP relay server for a gateway.
9. Edit the appropriate gateway.
10. In the **IP Address Management** field, click **No IP Allocation** for a tier-0 gateway or **No IP Allocation Set** for a tier-1 gateway.
11. In the **Type** field, select **DHCP Relay**.

12 In the **DHCP Relay** field, select the DHCP relay server you created earlier.

13 Click **Save**.

14 For each segment connected to the gateway that will use this DHCP relay service, you must specify DHCP ranges for the relay to function.

    a Go to **Networking > Segments**.

    b Add or edit a segment.

    c Click **Set Subnets** if you are adding a new segment, or click the number under **Subnets** to modify a subnet.

    d Specify one or more DHCP ranges.

    This is required for the relay to function.

    e Click **Apply**.

    f Click **Save**.

Add an IP Address Pool

You can configure IP address pools for use by components such as DHCP.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Networking > IP Address Management > IP Address Pools**.

3 Click **Add IP Address Pool**.

4 Enter a name and optionally a description.

5 To specify an address block, select **Add Sunet > IP Block**.

    a Select an IP block.

    b Specify a size.

    c Click **Add**.

6 To specify IP ranges, select **Add Sunet > IP Ranges**.

    a Enter IPv4 or IPv6 IP ranges.

    b Enter IP ranges in CIDR format.

    c Enter an address for **Gateway IP**.

    d Click **Add**.

7 Click **Save**.

Add an IP Address Block

You can configure IP address blocks for use by other components.

Note You can also add an IP address block by navigating to **Advanced Networking & Security > Networking > IPAM**.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Networking > IP Address Management > IP Address Pools**.
3. Click the **IP Address Blocks** tab.
4. Click **Add IP Address Block**.
5. Enter a name and optionally a description.
6. Enter an IP block in CIDR format.
7. Click **Save**.

Security

10

The topics in this section cover north-south and east-west security for distributed firewall rules, identity firewall, network introspection, gateway firewall, and endpoint protection policies.

This chapter includes the following topics:

Security Configuration Overview

Configure east-west and north-south firewall policies under predefined categories for your environment.

Distributed Firewall (east-west) and Gateway Firewall (north-south) offer multiple sets of configurable rules divided by categories. You can configure an exclusion list that contains logical switches, logical ports, or groups, to be excluded from firewall enforcement.

Security policies are enforced as follows:

- Rules are processed in categories, left to right.
- Rules are processed in top-to-bottom ordering.
- Each packet is checked against the top rule in the rule table before moving down the subsequent rules in the table.
- The first rule in the table that matches the traffic parameters is enforced.

No subsequent rules can be enforced as the search is then terminated for that packet. Because of this behavior, it is always recommended to put the most granular policies at the top of the rule table. This ensures they will be enforced before more specific rules.

Security Terminology

The following terms are used throughout distributed firewall.

Table 10-1. Security-Related Terminology

Construct	Definition
Domain	A domain represents an environment or security zone that includes firewall rules, and groups. Creating a domain is optional. The default domain represents the entire NSX environment. Rules in a domain must have at least one group in the source or destination that is a member of the same domain. Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1.
Policy	A security policy includes various security elements including firewall rules and service configurations. Policy was previously called a firewall section.
Rule	A set of parameters with which flows are evaluated against, and define which actions will be taken upon a match. Rules include parameters such as source and destination, service, context profile , logging, and tags.
Group	Groups include different objects that are added both statically and dynamically, and can be used as the source and destination field of a firewall rule. Groups can be configured to contain a combination of virtual machines, IP sets, MAC sets, logical ports, logical switches, AD user groups, and other nested groups. Dynamic inclusion of groups can be based on tag, machine name, OS name, or computer name. When you create a group, you must include a domain that it belongs to, and by default this is the default domain. Groups were previously called NSGroup or security group.
Service	Defines a combination or port and protocol. Used to classify traffic based on port and protocol. Pre-defined services and user-defined services can be used in firewall rules.
Context Profile	Defines context aware attributes including APP-ID and domain name. Also includes sub attributes such as application version, or cipher set. Firewall rules can include a context profile to enable Layer-7 firewall rules.

Identity Firewall

Identity Firewall (IDFW) features allow an NSX administrator to create Active Directory user-based Distributed Firewall (DFW) rules.

IDFW can be used for Virtual Desktops (VDI) or Remote desktop session (RDSH support), enabling simultaneous logins by multiple users, user application access based on requirements, and the ability to maintain independent user environments. VDI management systems control what users are granted access to the VDI virtual machines. NSX-T controls access to the destination servers from the source VM. IDFW is processed at the source VM. With RDSH administrators create security groups with different users in Active Directory (AD), and allow or deny those users access to an application server based on their role. For example, the Human Resources and Engineering can connect to the same RDSH server and have access to different applications from that server.

Note IDFW relies on the security and integrity of the guest operating system. There are multiple methods for a malicious local administrator to spoof their identity to bypass firewall rules. User identity information is provided by the Guest Introspection Agent inside guest VMs. Security administrators need to ensure that NSX Guest Introspection Agent is installed and running in each guest VM. Logged-in users should not have the privilege to remove or stop the agent.

Linux based operating systems are not supported.

IDFW is supported on:

Microsoft Active Directory Windows Server:

- 2008
- 2012
- 2012R2
- 2016
- 2019

VMware Tools version 10.3 or later: NSX File Introspection driver, NSX Network Introspection driver, VMCI driver.

Host operating system: ESXi only

Guest Operating systems:

- Desktop enforcement: Windows 8, Windows 10
- RDSH enforcement: Windows 2012R2, Windows 2016

A high level overview of the IDFW configuration workflow begins with preparing the infrastructure. This includes the administrator installing the host preparation components on each protected cluster, and setting up Active Directory synchronization so that NSX can consume AD users and groups. Next, IDFW must know which desktop an Active Directory user logs onto in order to apply IDFW rules. When network events are generated by a user, the thin agent installed with VMware Tools on the VM, gathers the information and forwards the information and sends it to the Context Engine. This information is used to provide enforcement for the Distributed Firewall.

IDFW workflow:

1. A user logs in to a VM and starts a network connection, by opening Skype or Outlook.
2. A user login event is detected by the Thin Agent, which gathers connection information and identity information and sends it to the Context Engine.

3 The context engine forwards the connection and the identity information to Distributed Firewall Wall for any applicable rule enforcement.

Identity Firewall Workflow

IDFW enhances traditional firewall by allowing firewall rules based on user identity. For example, administrators can allow or disallow customer support staff to access an HR database with a single firewall policy.

User-based distributed firewall rules are determined by membership in an Active Directory (AD) group membership. Identity Firewall requires a Thin Agent.

Note IDFW relies on the security and integrity of the guest operating system. There are multiple methods for a malicious local administrator to spoof their identity to bypass firewall rules. User identity information is provided by the Guest Introspection Agent inside guest VMs. Security administrators need to ensure that NSX Guest Introspection Agent is installed and running in each guest VM. Logged-in users should not have the privilege to remove or stop the agent.

Note For Identity Firewall rule enforcement, Windows Time service should be **on** for all VMs using Active Directory. This ensures that the date and time is synchronized between Active Directory and VMs. Additionally, AD group membership changes, including enabling and deleting users, do not immediately take effect for logged in users. For changes to take effect, users must log out and then log back in. AD administrator's should force a log out when group membership is modified. This behavior is a limitation of Active Directory.

Prerequisites

Microsoft Active Directory Windows Server:

- 2008
- 2012
- 2012R2
- 2016
- 2019

VMware Tools version 10.3 or later: NSX File Introspection driver, NSX Network Introspection driver, VMCI driver.

Host operating system: ESXi only

Guest Operating systems:

- Desktop enforcement: Windows 8, Windows 10
- RDSH enforcement: Windows 2012R2, Windows 2016

Procedure

1. Enable NSX File Introspection driver and NSX Network Introspection driver. VMware Tools full installation adds these by default.
2. Enable IDFW on cluster or standalone host: Enable Identity Firewall.
3. Configure Active Directory domain: Add an Active Directory.
4. Configure Active Directory sync operations: Synchronize Active Directory.
5. Create security groups (SG) with Active Directory group members: Add a Group.
6. Assign SG with AD group members to a distributed firewall rule: Add a Distributed Firewall .

Enable Identity Firewall

Identity Firewall must be enabled for IDFW firewall rules to take effect.

Procedure

1. Select **Security > Distributed Firewall** from the navigation panel.
2. Click **Enable IDFW** on the banner.
3. Again, click **Enable IDFW** on the banner. Click the status button to enable IDFW.

   The **Edit Identity Firewall** screen appears.
4. Toggle the status button to enable IDFW.
5. (Optional) Toggle the status button to enable IDFW on Standalone Hosts.
6. (Optional) Change the status of each available cluster to enable IDFW on a per cluster basis.
7. Click **Save**.

Identity Firewall Best Practices

The following best practices will help maximize the success of identity firewall rules.

- IDFW supports only TCP-based firewall rules.
- A single ID based group can be used within a firewall rule. If IP and ID based groups are needed at the source, create two separate firewall rules.
- Windows 2008 is not supported as an Active Directory server or RDSH Server OS.
- Any change on a domain, including a domain name change, will trigger a full sync with Active Directory. Because a full sync can take a long time, we recommend syncing during off-peak or non business hours.
- The default LDAP port 389 and LDAPs port 636 are used for the Active Directory sync, and should not be edited from the default values. Custom ports are not supported.

Layer 7 Context Profile

Layer 7 Application Identity is configured as part of a context profile.

A context profile can specify one or more Application Identification GUIDS, and can also include sub-attributes. When a sub-attribute, such as TLS version 1.2 is defined, multiple application identity attributes are not supported. In addition to APP-IDs, a Fully Qualified Domain Name (FQDN) or URL can also be set in a context profile for FQDN whitelisting. FQDN can be configured along with APP-ID in a Context Profile, or each can be set in different context profiles. Once a context profile has been defined, it can be applied to one or more distributed firewall rules.

When a context-profile has been used in a rule, any traffic coming in from a virtual machine is matched against the rule-table based on 5-tuple. If the rule matches the flow also includes a Layer 7 Context profile, that packet will be redirected to a user-space component called the Deep Packet Inspection (DPI) engine. A small number of subsequent packets are punted to that DPI engine for each flow, and once it has determined the APP_ID, this information is stores in the in-kernel context-table. When the next packet for the flow comes in, the information in the context table is compared with the rule table and is matched on 5-tuple, and on the layer 7 APP-ID. The appropriate action as defined in the rule is taken, and in case of an ALLOW-rule, all subsequent packets for the flow are process in kernel, and matched against the connection table. Logs generated by the Distributed Firewall will include the Layer 7 APP_ID if that flow was punted to DPI.

Rule processing for an incoming packet:

1. Upon entering a DFW filter, packets are looked up in the flow table based on 5-tuple.
2. If no flow/state is found, the flow is matched against the rule-table based on 5-tuple and an entry is created in the flow table.
3. If the flow matches a rule with a Layer 7 service object, the flow table state is marked as “DPI In Progress.”
4. The traffic is then punted to the DPI engine. The DPI Engine determines the APP_ID.
5. Once the APP_ID has been determined, the DPI Engine sends down the attribute which is inserted into the context table for this flow. The ”DPI In Progress” flag is removed and traffic is no longer punted to the DPI engine.
6. The flow (now with APP-ID) is reevaluated against all rules that match the APP_ID, starting with the original rule that was matched based on 5-tuple, and ensuring that no matching L4 rules take precedence. The appropriate action is taken (allow/deny) and the flow table entry is updated accordingly.

Layer 7 Distributed Firewall Rule Workflow

Layer 7 App IDs are used in creating a context profile and then used in creating distributed firewall rules. Rule enforcement based on application identity enables users to allow or deny applications to run on any port.

NSX-T provides built in Application Identification GUIDS for common infrastructure and enterprise applications. App IDs include versions (SSL/TLS and CIFS/SMB) and Cipher Suite (SSL/TLS). App IDs are used in rules through context profiles, and can be combined with FQDN whitelisting and blacklisting. Supported on ESXi hosts only.

Supported App Ids and FQDNs:

- For FQDN, users need to configure a high priority rule with a DNS App Id for the specified DNS servers on port 53.
- ALG App Ids (FTP, ORACLE, DCERPC, TFTP), require the corresponding ALG service for the firewall rule.
- SYSLOG App Id is detected only on standard ports.

Procedure

1. Create a custom context profile: Add a Context Profile.
2. Use the context profile in a distributed firewall rule: Add a Distributed Firewall .

Application Identification GUIDS

Layer 7 application identification identifies which application a particular packet or flow is generated by, independent of the port that is being used.

Enforcement based on application identity enables users to allow or deny applications to run on any port, or to force applications to run on their standard port. Deep Packet Inspection (DPI) enables matching packet payload against defined patterns, commonly referred to as signatures. Signature-based identification and enforcement enables customers not just to match the particular application/protocol a flow belongs to, but also the version of that protocol, for example TLS version 1.0 version TLS version 1.2 or different versions of CIFS traffic. This allows customers to get visibility into or restrict the use of protocols that have known vulnerabilities for all deployed applications and their E-W flows within the datacenter.

Supported App Ids and FQDNs:

- For FQDN, users need to configure a high priority rule with a DNS App Id for the specified DNS servers on port 53.
- ALG App Ids (FTP, ORACLE, DCERPC, TFTP), require the corresponding ALG service for the firewall rule.
- SYSLOG App Id is detected only on standard ports.

KVM Supported App Ids and FQDNs:

- Sub attributes are not supported on KVM.
- FTP and TFTP ALG App Ids are supported on KVM.

Layer 7 APP-IDs are used in context profiles in distributed firewall and are supported only on ESXi hosts.

GUID	Description	Type
360ANTIV	360 Safeguard is a program developed by Qihoo 360, an IT company based in China	Web Services
ACTIVDIR	Microsoft Active Directory	Networking
AD_BKUP	Microsoft Active Directory Backup Service	Networking
AD_NSP	Microsoft Active Directory Service Provider	Networking
AMQP	Advanced Messaging Queuing Protocol is application layer protocol which supports business message communication between applications or organizations	Networking
AVAST	Traffic generated by browsing Avast.com official website of Avast! Antivirus downloads	Web Services
AVG	AVG Antivirus/Security software download and updates	File Transfer
AVIRA	Avira Antivirus/Security software download and updates	File Transfer
BLAST	A remote access protocol that compresses, encrypts, and encodes a computing experiences at a data center and transmits it across any standard IP network for VMware Horizon desktops.	Remote Access
BDEFNDER	BitDefender Antivirus/Security software download and updates.	File Transfer
CA_CERT	Certification authority (CA) issues digital certificates which certifies the ownership of a public key for message encryption	Networking
CIFS	CIFS (Common Internet File System) is used to provide shared access to directories, files, printers, serial ports, and miscellaneous communications between nodes on a network	File Transfer
CLDAP	Connectionless Lightweight Directory Access Protocol is an application protocol for accessing and maintaining distributed directory information services over an Internet Protocol (IP) network using UDP.	
CLRCASE	A software tool for revision control of source code and other software development assets. It is developed by the Rational Software division of IBM. ClearCase forms the base of revision control for many large and medium sized businesses and can handle projects with hundreds or thousands of developers	Networking
CTRXCGP	Citrix Common Gateway Protocol is an application protocol for accessing and maintaining distributed directory information services over an Internet Protocol (IP) network using UDP.	Database
CTRXGOTO	Hosting Citrix GoToMeeting, or similar sessions based on the GoToMeeting platform. Includes voice, video, and limited crowd management functions	Collaboration
CTRXICA	ICA (Independent Computing Architecture) is a proprietary protocol for an application server system, designed by Citrix Systems	Remote Access
DCERPC	Distributed Computing Environment / Remote Procedure Calls, is the remote procedure call system developed for the Distributed Computing Environment (DCE)	Networking
DIAMETER	An authentication, authorization, and accounting protocol for computer networks	Networking

GUID	Description	Type
DNS	Querying a DNS server over TCP or UDP	Networking
EPIC	Epic EMR is an electronic medical records application that provides patient care and healthcare information.	Client Server
ESET	Eset Antivirus/Security software download and updates	File Transfer
FPROT	F-Prot Antivirus/Security software download and updates	File Transfer
FTP	FTP (File Transfer Protocol) is used to transfer files from a file server to a local machine	File Transfer
GITHUB	Web-based Git or version control repository and Internet hosting service	Collaboration
HTTP	(HyperText Transfer Protocol) the principal transport protocol for the World Wide Web	Web Services
HTTP2	Traffic generated by browsing websites that support the HTTP 2.0 protocol	Web Services
IMAP	IMAP (Internet Message Access Protocol) is an Internet standard protocol for accessing email on a remote server	Mail
KASPRSKY	Kaspersky Antivirus/Security software download and updates	File Transfer
KERBEROS	Kerberos is a network authentication protocol designed to provide strong authentication for client/server applications by using secret-key cryptography	Networking
LDAP	LDAP (Lightweight Directory Access Protocol) is a protocol for reading and editing directories over an IP network	Database
MAXDB	SQL connections and queries made to a MaxDB SQL server	Database
MCAFEE	McAfee Antivirus/Security software download and updates	File Transfer
MSSQL	Microsoft SQL Server is a relational database.	Database
NFS	Allows a user on a client computer to access files over a network in a manner similar to how local storage is accessed	File Transfer
NNTP	An Internet application protocol used for transporting Usenet news articles (netnews) between news servers, and for reading and posting articles by end user client applications.	File Transfer
NTBIOSNS	NetBIOS Name Service. In order to start sessions or distribute datagrams, an application must register its NetBIOS name using the name service	Networking
NTP	NTP (Network Time Protocol) is used for synchronizing the clocks of computer systems over the network	Networking
OCSP	An OCSP Responder verifying that a user's private key has not been compromised or revoked	Networking
ORACLE	An object-relational database management system (ORDBMS) produced and marketed by Oracle Corporation.	Database
PANDA	Panda Security Antivirus/Security software download and updates.	File Transfer

GUID	Description	Type
PCOIP	A remote access protocol that compresses, encrypts, and encodes a computing experiences at a data center and transmits it across any standard IP network.	Remote Access
POP2	POP (Post Office Protocol) is a protocol used by local e-mail clients to retrieve e-mail from a remote server.	Mail
POP3	Microsoft's implementation of NetBIOS Name Service (NBNS), a name server and service for NetBIOS computer names.	Mail
RADIUS	Provides centralized Authentication, Authorization, and Accounting (AAA) management for computers to connect and use a network service	Networking
POSTGRES		
RDP	RDP (Remote Desktop Protocol) provides users with a graphical interface to another computer	Remote Access
RTCP	RTCP (Real-Time Transport Control Protocol) is a sister protocol of the Real-time Transport Protocol (RTP). RTCP provides out-of-band control information for an RTP flow.	Streaming Media
RTP	RTP (Real-Time Transport Protocol) is primarily used to deliver real-time audio and video	Streaming Media
RTSP	RTSP (Real Time Streaming Protocol) is used for establishing and controlling media sessions between end points	Streaming Media
RTSPS	A secure network control protocol designed for use in entertainment and communications systems to control streaming media servers. The protocol is used for establishing and controlling media sessions between end points.	Streaming Media
SAP	Connections to generic components of several SAP products such as Netweaver, BusinessObjects XI, and Crystal Enterprise Server.	Collaboration
SIP	SIP (Session Initiation Protocol) is a common control protocol for setting up and controlling voice and video calls	Streaming Media
SKIP	Simple Key Management for Internet Protocols (SKIP) is hybrid Key distribution protocol Simple Key Management for Internet Protocols (SKIP) is similar to SSL, except that it establishes a long-term key once, and then requires no prior communication in order to establish or exchange keys on a session-by-session basis.	Networking
SMTP	SMTP (Simple Mail Transfer Protocol) An Internet standard for electronic mail (e-mail) transmission across Internet Protocol (IP) networks.	Mail
SNMP	SNMP (Simple Network Management Protocol) is an Internet-standard protocol for managing devices on IP networks.	Network Monitoring
SQLNET	Networking software that allows remote data-access between programs and the Oracle Database, or among multiple Oracle Databases.	Database
SQLSERV	SQL Services	Database

GUID	Description	Type
SSH	SSH (Secure Shell) is a network protocol that allows data to be exchanged using a secure channel between two networked devices.	Remote Access
SSL	SSL (Secure Sockets Layer) is a cryptographic protocol that provides security over the Internet.	Web Services
SVN	Managing content on a Subversion server.	Database
SYMUPDAT	Symantec LiveUpdate traffic, this includes spyware definitions, firewall rules, antivirus signature files, and software updates.	File Transfer
SYSLOG	Symantec LiveUpdate traffic, this includes spyware definitions, firewall rules, antivirus signature files, and software updates.	Network Monitoring
TELNET	A network protocol used on the Internet or local area networks to provide a bidirectional interactive text-oriented communications facility using a virtual terminal connection.	Remote Access
TFTP	TFTP (Trivial File Transfer Protocol) being used to list, download, and upload files to a TFTP server like SolarWinds TFTP Server, using a client like WinAgents TFTP client.	File Transfer
VNC	Traffic for Virtual Network Computing.	Remote Access
WINS	Microsoft's implementation of NetBIOS Name Service (NBNS), a name server and service for NetBIOS computer names.	Networking

Distributed Firewall

Distributed firewall comes with predefined categories for firewall rules. Rules are evaluated top down, and left to right. The category names can be changed using the API.

Table 10-2. Categories

Ethernet	Used for Layer 2 based rules
Emergency	Used for quarantine and allow rules
Infrastructure	Define access to shared services. Global rules - AD, DNS, NTP, DHCP, Backup, Managment Servers
Environment	Rules between zones - production vs development, inter business unit rules
Application	Rules between applications, application tiers, or the rules between micro services

Add a Distributed Firewall

Distributed firewall monitors all the East-West traffic on your virtual machines.

Prerequisites

To be DFW-protected, guest virtual machines must have their vNIC connected to an N-VDS logical switch associated with a transport zone.

If you are creating rules for Identity Firewall, first create a group with Active Directory members. IDFW only supports TCP-based firewall rules.

Note For Identity Firewall rule enforcement, Windows Time service should be **on** for all VMs using Active Directory. This ensures that the date and time is synchronized between Active Directory and VMs. Additionally, AD group membership changes, including enabling and deleting users, do not immediately take effect for logged in users. For changes to take effect, users must log out and then log back in. AD administrator's should force a log out when group membership is modified. This behavior is a limitation of Active Directory.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Security > Distributed Firewall** from the navigation panel.
3. Ensure that you are in the correct pre-defined category, and click **Add Policy**. For more about categories see Distributed Firewall .
4. Enter a **Name** for the new policy section.
5. Select the policy **Destination** domain. Keep the default policy domain or add or create another domain. A domain is a logical construct that represents a security zone and all the security groups and rules.

   Note that the domain object is an experimental feature in NSX-T Data Center 2.4, and is not available in NSX-T Data Center 2.4.1.

6 (Optional) Click the gear icon to configure the following policy settings:

Menu Option	Description
TCP Strict	A TCP connection begins with a three-way handshake (SYN, SYN-ACK, ACK) and typically ends with a two-way exchange (FIN, ACK). In certain circumstances, the distributed firewall may not see the three-way handshake for a particular flow (i.e. due to asymmetric traffic or the distributed firewall being enabled while a flow exists). By default, the distributed firewall does not enforce the need to see a three-way handshake, and will pick-up sessions that are already established. TCP strict can be enabled on a per section basis to turn off mid-session pick-up and enforce the requirement for a three-way handshake. When enabling TCP strict mode for a particular Distributed Firewall Section, and using a default ANY-ANY Block rule, packets that do not complete the three-way handshake connection requirements and that match a TCP-based rule in this section are dropped. Strict is only applied to stateful TCP rules and is enabled at the distributed firewall section level. TCP strict is not enforced for packets that match a default ANY-ANY Allow which has no TCP service specified.
Stateful	A stateful firewall monitors the state of active connections and uses this information to determine which packets to let though the firewall.
Locked	The policy can be locked to prevent multiple users from making changes to the same sections. When locking a section, you must include a comment. Some roles such as enterprise administrator have full access credentials, and cannot be locked out. See Role-Based Access Control.

7 Click **Publish**. Multiple policies can be added and then published together at one time.

The new policy is shown on the screen.

8 Select a policy section and click **Add Rule**.

9 Enter a name for the rule.

10 In the **Sources** column, click the edit icon and select the source of the rule. Groups with Active Directory members can be used for the source text box of an IDFW rule. See Add a Group for more information.

11 In the **Destinations** column, click the edit icon and select the destination of the rule. If not defined, the destination matches any. See Add a Group for more information.

12 In the **Services** column, click the edit icon and select services. The service matches **Any** if not defined.

13 This **Profiles** column is not available when adding a rule to the Ethernet category. For all other rule categories, in the **Profiles** column, click the edit icon and select a context profile. See Add a Context Profile.

Context profiles use layer 7 APP ID attributes for use in distributed firewall rules.

14 By default, the **Applied to** column is set to DFW, and the rule is applied to all workloads. You can also apply the rule or policy to selected groups. **Applied to** defines the scope of enforcement per rule, and is used mainly for optimization or resources on ESXi and KVM hosts. It helps in defining a targeted policy for specific zones and tenants, without interfering with other policy defined for other tenants and zones.

Groups consisting of only IP addresses, MAC Addresses, or Active Directory groups cannot be used in the **Applied-to** text box.

15 In the **Action** column, select an action.

Option	Description
Allow	Allows all L3 or L2 traffic with the specified source, destination, and protocol to pass through the current firewall context. Packets that match the rule, and are accepted, traverse the system as if the firewall is not present.
Drop	Drops packets with the specified source, destination, and protocol. Dropping a packet is a silent action with no notification to the source or destination systems. Dropping the packet causes the connection to be retried until the retry threshold is reached.
Reject	Rejects packets with the specified source, destination, and protocol. Rejecting a packet is a more graceful way to deny a packet, as it sends a destination unreachable message to the sender. If the protocol is TCP, a TCP RST message is sent. ICMP messages with administratively prohibited code are sent for UDP, ICMP, and other IP connections. One benefit of using Reject is that the sending application is notified after only one attempt that the connection cannot be established.

16 Click the status toggle button to enable or disable the rule.

17 (Optional) Click the gear icon to configure the following rule options:

Option	Description
Logging	Logging is turned off by default. Logs are stored at /var/log/dfwpktlogs.log file on ESXi and KVM hosts.
Direction	This text box refers to the direction of traffic from the point of view of the destination object. IN means that only traffic to the object is checked, OUT means that only traffic from the object is checked, and In/Out, means that traffic in both directions is checked.
IP Protocol	Enforce the rule based on IPv4, IPv6, or both IPv4-IPv6.
Tag	Tags can make searching easier.

18 Click **Publish**. Multiple rules can be added and then published together at one time.

Add a Firewall Rule to Whitelist FQDN/URLs

Set up a distributed firewall rule to whitelist specific east-west traffic to go to specific domains identified with FQDN/URLs, for example, **.office365.com*.

Currently a predefined list of domains is supported. You can see the list of FQDNs when you add a new Context Profile of Attribute type *Domain (FQDN) Name*.

You must set up a DNS rule first, and then the FQDN whitelist rule below it. This is because NSX-T Data Center uses DNS Snooping to obtain a mapping between the IP address and the FQDN. To protect against the risk of DNS spoofing attacks, wherein a malicious VM can inject spoofed DNS responses to redirect traffic to malicious endpoints or bypass the DFW, Spoofguard should be enabled across the switch on all logical ports. For more information about Spoofguard, see Understanding SpoofGuard Segment Profile.

FQDN-based rules are retained during vMotion.

Note In the current release, only ESXi is supported.

Prerequisites

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Security > Distributed Firewall** from the navigation panel.
3. Add a firewall policy section by following the steps in Add a Distributed Firewall . Alternatively you can use an existing firewall policy section.
4. Select the new or existing firewall policy section and click **Add Rule** to create the DNS firewall rule first.
5. Provide a name for the firewall rule, such as, `DNS rule`, and provide the following details:

Option	Description
Services	Click the edit icon and select the DNS or DNS-UDP service as applicable to your environment.
Profile	Click the edit icon and select the DNS context profile. This is precreated and is available in your deployment by default.
Applied To	Select DFW or a group as required.
Action	Select **Allow**.

6. Click **Add Rule** again to set up the FQDN whitelisting rule.
7. Name the rule appropriately, such as, `FQDN/URL Whitelist`. Drag the rule under the DNS rule under this policy section.

8 Provide the following details:

Option	Description
Services	Click the edit icon and select the service you want to associate with this rule, for example, HTTP.
Profile	Click the edit icon and click **Add New Context Profile**. Click in the column titled **Attribute**, and select **Domain (FQDN) Name**. Select the list of Attribute Name/Values from the predefined list. Click **Add**. See Add a Context Profile for details.
Applied To	Select DFW or a group as required.
Action	Select **Allow**.

9 Click **Publish**.

Distributed Firewall Packet Logs

If logging is enabled for firewall rules, you can look at the firewall packet logs to troubleshoot issues.

The log file is `/var/log/dfwpktlogs.log` for both ESXi and KVM hosts.

The following is a regular log sample for distributed firewall rules:

```
2018-07-03T19:44:09.749Z b6507827 INET match PASS mainrs/1024 IN 52 TCP 192.168.4.3/49627-
>192.168.4.4/49153 SEW

2018-07-03T19:46:02.338Z 7396c504 INET match DROP mainrs/1024 OUT 52 TCP 192.168.4.3/49676-
>192.168.4.4/135 SEW

2018-07-06T18:15:49.647Z 028cd586 INET match DROP mainrs/1027 IN 36 PROTO 2 0.0.0.0->224.0.0.1

2018-07-06T18:19:54.764Z 028cd586 INET6 match DROP mainrs/1027 OUT 143 UDP
fe80:0:0:0:68c2:8472:2364:9be/546->ff02:0:0:0:0:0:1:2/547
```

The elements of a DFW log file format include the following, separated by a space:

- timestamp:
- last eight digits of the VIF ID of the interface
- INET type (v4 or v6)
- reason (match)
- action (PASS, DROP, REJECT)
- rule set name/ rule ID
- packet direction (IN/OUT)
- packet size
- protocol (TCP, UDP, or PROTO #)

- SVM direction for netx rule hit
- source IP address/source port>destination IP address/destination port
- TCP flags (SEW)

For passed TCP packets there is a termination log when the session has ended:

```
2018-07-03T19:44:30.585Z 7396c504 INET TERM mainrs/1024 OUT TCP RST 192.168.4.3/49627-
>192.168.4.4/49153 20/16 1718/76308
```

The elements of a TCP termination log include the following, separated by a space:

- timestamp:
- last 8 digits of the VIF ID of the interface
- INET type (v4 or v6)
- action (TERM)
- ruleset name/ rule ID
- packet direction (IN/OUT)
- protocol (TCP, UDP, or PROTO #)
- TCP RST flag
- SVM direction for netx rule hit
- source IP address/source port>destination IP address/destination port
- IN packet count/OUT packet count (all accumulated)
- IN packet size/OUT packet size

The following is a sample of FQDN log file for distributed firewall rules:

```
2019-01-15T00:34:45.903Z 7c607b29 INET match PASS 1031 OUT 48 TCP 10.172.178.226/32808-
>23.72.199.234/80 S www.sway.com(034fe78d-5857-0680-81e4-d8da6b28d1b4)
```

The elements of an FQDN log include the following, separated by a space:

- timestamp:
- last eight digits of the VIF ID of the interface
- INET type (v4 or v6)
- reason (match)
- action (PASS, DROP, REJECT)
- ruleset name/ rule ID
- packet direction (IN/OUT)
- packet size
- protocol (TCP, UDP, or PROTO #)

- source IP address/source port>destination IP address/destination port
- domain name/UUID where UUID is the binary internal representation of the domain name

The following is a sample of Layer 7 log file for distributed firewall rules:

```
2019-01-15T00:35:07.221Z 82f365ae INET match REJECT 1034 OUT 48 TCP 10.172.179.6/49818-
>23.214.173.202/80 S APP_HTTP

2019-01-15T00:34:46.486Z 7c607b29 INET match PASS 1030 OUT 48 UDP 10.172.178.226/42035-
>10.172.40.1/53 APP_DNS
```

The elements of a Layer 7 log include the following, separated by a space:

- timestamp:
- last eight digits of the VIF ID of the interface
- INET type (v4 or v6)
- reason (match)
- action (PASS, DROP, REJECT)
- ruleset name/ rule ID
- packet direction (IN/OUT)
- packet size
- protocol (TCP, UDP, or PROTO #)
- source IP address/source port>destination IP address/destination port
- APP_XXX is the discovered application

Select a Default Connectivity Strategy

You can select a default connectivity strategy to enforce your security model.

The default connectivity strategy creates either an allow-all (blacklist) or deny-all (whitelist) firewall policy on top of the other firewall rules you create instead of your having to modifying individual rules.

The following options are available:

- **Blacklist (with or without logging)**: This is the default option and creates an allow-all rule on the DFW.
- **Whitelist (with or without logging)**: Creates a deny-all traffic firewall rule. Only communication from sites or applications that have been defined in firewall rules is allowed, and all other communication is denied access, this includes DHCP traffic.
- **None**: Select this option to disable both blacklisting or whitelisting of firewall rules. This is useful if you have a set of rules already configured using previous versions of NSX-T Data Center.

Configuring a Gateway Firewall

Gateway firewall represents rules applied at the perimeter firewall.

There are predefined categories under the **All Shared Rules** view, where rules across all gateways are visible. Rules are evaluated top down, and left to right. The category names can be changed using the API.

Table 10-3. Categories for Gateway Firewall Rules

Rule Category	Purpose
Emergency	Used for Quarantine. Can also be used for Allow rules.
System	These rules are automatically generated by NSX-T Data Center and are specific to internal control plane traffic, such as, BFD rules, VPN rules and so on. **Note** Do not edit System rules.
Shared Pre Rules	These rules are globally applied across gateways.
Local Gateway	These rules are specific to a particular gateway.
Auto Service Rules	These are auto-plumbed rules applied to the data plane. You can edit these rules as required.
Default	These rules define the default gateway firewall behavior.

Add a Gateway Firewall Policy and Rule

Implement gateway firewall rules by adding them under a firewall policy section that belongs to a predefined category.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Security > North South Security > Gateway Firewall** and navigate to the category where you want to add the new policy.
3. click **Add Policy**. For more about categories see Configuring a Gateway Firewall.
4. Enter a **Name** for the new policy section.
5. Select the policy **Destination** domain. Keep the default policy domain or add or create another domain. A domain is a logical construct that represents a security zone and all of the security groups and rules.

   Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1.

6 Click the gear icon to configure the following policy settings:

Menu Option	Description
TCP Strict	A TCP connection begins with a three-way handshake (SYN, SYN-ACK, ACK), and typically ends with a two-way exchange (FIN, ACK). In certain circumstances, the firewall may not see the three-way handshake for a particular flow (i.e. due to asymmetric traffic). By default, the firewall does not enforce the need to see a three-way handshake, and will pick-up sessions that are already established. TCP strict can be enabled on a per section basis to turn off mid-session pick-up, and enforce the requirement for a three-way handshake. When enabling TCP strict mode for a particular firewall policy and using a default ANY-ANY Block rule, packets that do not complete the three-way handshake connection requirements and that match a TCP-based rule in this policy section are dropped. Strict is only applied to stateful TCP rules, and is enabled at the gateway firewall policy level. TCP strict is not enforced for packets that match a default ANY-ANY Allow which has no TCP service specified.
Stateful	A stateful firewall monitors the state of active connections and uses this information to determine which packets to let though the firewall.
Locked	The policy can be locked to prevent multiple users from making changes to the same sections. When locking a section, you must include a comment.

7 Click **Publish**. Multiple Policies can be added and then published together at one time.

The new policy is shown on the screen.

8 Select a policy section and click **Add Rule**.

9 Enter a name for the rule.

10 In the **Sources** column, click the edit icon and select the source of the rule. See Add a Group for more information.

11 In the **Destinations** column, click the edit icon and select the destination of the rule. The destination will match any if not defined. See Add a Group for more information.

12 In the **Services** column, click the edit icon and select services. The service will match any if not defined.

13 The **Applied to** column defines the scope of enforcement per rule and is used mainly for optimization of resources on ESXi and KVM hosts. You can define a targeted policy for specific zones and tenants without interfering with policy defined for other tenants and zones. You can choose a Logical Router (Tier-0 or Tier-1) or interfaces on Logical Routers or Route-based VPN sessions in this column.

14 In the **Action** column, select an action.

Option	Description
Allow	Allows all traffic with the specified source, destination, and protocol to pass through the current firewall context. Packets that match the rule, and are accepted, traverse the system as if the firewall is not present.
Drop	Drops packets with the specified source, destination, and protocol. Dropping a packet is a silent action with no notification to the source or destination systems. Dropping the packet causes the connection to be retried until the retry threshold is reached.

15 Click the status toggle button to enable or disable the rule.

16 Click the gear icon to set logging, direction, IP protocol, tag, and notes.

Option	Description
Logging	Logging can be turned off or on. Logs are stored at /var/log/dfwpktlogs.log file on ESXi and KVM hosts.
Direction	The options are `In`, `Out`, and `In/Out`. The default is `In/Out`. This field refers to the direction of traffic from the point of view of the destination object. `In` means that only traffic to the object is checked, `Out` means that only traffic from the object is checked, and `In/Out` means traffic in both directions is checked.
IP Protocol	The options are `IPv4`, `IPv6`, and `IPv4_IPv6`. The default is `IPv4_IPv6`.
Tags	Tags that have been added to the rule.

Note Click the graph icon to view the flow statistics of the firewall rule. You can see information such as the byte, packet count, and sessions.

17 Click **Publish**. Multiple rules can be added and then published together at one time.

Configure Network Introspection East-West

After partners register network services such as Intrusion Detection System or Intrusion Protection System (IDS/IPS) with NSX-T Data Center, as an administrator you can configure network services to introspect east-west traffic moving between VMs on an on-premises data center.

High-Level Tasks for East-West Network Security

Follow these steps to set up network security for east-west traffic.

Table 10-4. List of Tasks to Configure East-West Network Introspection

Workflow Tasks	Persona	Implementation
Register Service	Partner	Only API
Register Vendor Template	Partner	Only API
Register Service Manager	Partner	Only API
Deploy a Service for East-West Traffic Introspection	Administrator	API and NSX Manager UI
Add a Service Profile	Administrator	API and NSX Manager UI
Add a Service Chain	Administrator	API and NSX Manager UI
Add Redirection Rules for East-West Traffic	Administrator	API and NSX Manager UI

Key Concepts of Network Protection East-West

Traffic flowing between Guest VMs on an on-premises data center is protected by third-party services provided by partners. There are a few concepts that aid your understanding of the workflow.

- Service: Partners register services with NSX-T Data Center . A service represents the security functionality offered by the partner, service deployment details such as OVF URL of service VMs, point to attach the service, state of the service.
- Vendor Template: It consists of functionality that a service can perform on a network traffic. Partners define vendor templates. For example, a vendor template can provide a network operation service such as tunneling with IPSec service.
- Service Profile: Is an instance of a vendor template. An NSX-T Data Center administrator can create a service profile to be consumed by service VMs.
- Guest VM: a source or destination of traffic in the network. The incoming or outgoing traffic is introspected by a service chain defined for a rule running east-west network services.
- Service VM: A VM that runs the OVA or OVF appliance specified by a service. It is connected over the service plane to receive redirected traffic.
- Service Instance: Is created when a service is deployed on a host. Each service instance has a corresponding service VM.
- Service Segment: A segment of a service plane that is associated to a transport zone. Each service attachment is segregated from other service attachments and from the regular L2 or L3 network segments provided by NSX-T. The service plane manages service attachments.
- Service Manager: Is the partner service manager that points to a set of services.

- Service Chain: Is a logical sequence of service profiles defined by an administrator. Service profiles introspect network traffic in the order defined in the service chain. For example, the first service profile is firewall, second service profile is monitor, and so on. Service chains can specify different sequence of service profiles for different directions of traffic (egress/ingress).
- Redirection Policy: Ensures that traffic classified for a specific service chain is redirected to that service chain. It is based on traffic patterns that match NSX-T Data Center security group and a service chain. All traffic matching the pattern is redirected along the service chain.
- Service Path: Is a sequence of service VMs that implement the service profiles of a service chain. An administrator defines the service chain, which consists of a pre-defined order of service profiles. NSX-T Data Center generates multiple service paths from a service chain based on the number, and locations of guest VMs and service VMs. It selects the optimum service path for the traffic flow to be introspected. Each service path is identified by a Service Path Index (SPI) and each hop along a path has a unique Service Index (SI).

Deploy a Service for East-West Traffic Introspection

After partners register services, as an administrator you must deploy an instance of the service on member hosts of a cluster.

Deploy partner service VMs that run the partner security engine on all the NSX-T Data Center hosts in a cluster. After you deploy the SVMs, you can create policy rules used by SVM to protect guest VMs.

Prerequisites

- All hosts are managed by a vCenter Server.
- Partner services are registered with NSX-T Data Center and are ready for deployment.
- NSX-T Data Center administrators can access partner services and vendor templates.
- Both the service VM and the partner service manager (console) must be able to communicate with each other at the management network level.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Service Deployments > Deployment > Deploy Service**.
3. From the Partner Service field, select the partner service.
4. Enter the service deployment name.
5. In the Compute Manager field, select the compute resource on the vCenter Server to deploy the service.
6. In the Cluster field, select the cluster where the services need to be deployed.

7. In the Data Store drop-down menu, select a data store as the repository for the service virtual machine.
8. In the Network column, click **Set** and enter the Management Network interface by choosing DHCP or static IP address type, control network and data network.
9. In the Service Segments field, select a service segment from the list or click the Action icon to add or edit a service segment. A service segment determines the guest VMs associated with an overlay transport zone that are to be provided east-west network traffic protection.
10. In the Deployment Specification field, select the service and the form factor of the service VM to be deployed on cluster hosts. There can be multiple services available for deployment.
11. In the Deployment Template field, select the vendor template with attributes to protect the workload you want to run on guest VMs groups.
12. In the Clustered Deployment Count, enter the number of service VMs to deploy on the cluster. The vCenter Server decides on which host to deploy the service VMs.
13. Click **Save**.

Results

After service deployment, the partner Service Manager is notified about the update.

What to do next

Know deployment details and heath status about service instances deployed on hosts. See View Service Instance Details.

Add a Service Profile

A service profile is an instance of a partner vendor template. Administrators can customize attributes of a vendor template to create an instance of the template.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to **Security > East West Security > Network Introspection > Service Profiles**.
3. From the Partner Service drop-down field, select a service. You can create a service profile for the selected service.
4. Enter the service profile name and select the vendor template.
5. Click **Save**.

Results

A new service profile is created for the partner service.

What to do next

Add a service chain. See Add a Service Chain.

Add a Service Chain

A service chain is a logical sequence of service profiles defined by the network administrator.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Security > East West Security > Network Introspection > Service Chain > Add Chain**.
3. Enter the service chain name.
4. In the Service Segments field, select the service segment to which you want to apply the service chain. A service segment is a segment of service plane that connects multiple service VMs of an overlay transport zone. Each service VM in the service chain is separate from another service VM and L2 and L3 network segments run by NSX-T Data Center. The service plane controls access to service VMs.
5. To set the forward path, click the **Set Forward Path field** and click **Add Profile in Sequence**.
6. Add the first profile in the service chain and click **Add**.
7. To specify the next service profile, click **Add Profile in Sequence** and enter details. You can also rearrange the profile order by using the Up and Down arrow icons.
8. Click **Save** to finish adding a forward path for the service chain.
9. In the Reverse Path column, select **Inverse Forward Path** for the service plane to use the forward path in reverse order. To set a new reverse path, click **Set Reverse Path** and add a new reverse path.
10. Click **Save** to finish adding a reverse path for the service chain.
11. In the Failure Policy field,
    - Select **Allow** to send traffic to the destination VM when the service VM fails. Service VM failure is detected by the liveness detection mechanism which can be enabled only by partners.
    - Select **Block** to not send traffic to the destination VM when the service VM fails.
12. Click **Save**.

Results

After adding a service chain, the partner Service Manager is notified about the update.

What to do next

Create a redirection rule to introspect east-west network traffic. See Add Redirection Rules for East-West Traffic.

Add Redirection Rules for East-West Traffic

Add rules to redirect an east-west traffic for network introspection.

Rules are defined in a policy. Policy as a concept is similar to the concept of sections in firewalls. When you add a policy, select the service chain to redirect the traffic for introspection by service profiles of the service chain.

A rule definition consists of source and destination of the traffic, introspection service, the NSX object to apply the rule to, and traffic redirection policy. After you publish the rule, NSX Manager triggers the rule when a matching traffic pattern is found. The rule begins to introspect the traffic. For example, when NSX Manager classifies a traffic flow that must be introspected, it does not forward it to the regular distributed firewall, rather it redirects that traffic along the specified service chain in the policy. The service profiles defined in the service chain introspect the traffic for network services the partner offers. If a service profile finishes introspection without detecting any security issues in the traffic, the traffic is forwarded to the next service profile in the service chain. At the end of the service chain, the traffic is forwarded to the destination target.

All notifications are sent to the partner Service Manager and NSX-T Data Center.

Prerequisites

A service chain is available to redirect the traffic for a network introspection.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. **Security > East West Security > Network Introspection > Rules > Add Policy**.

   A policy section is similar to a firewall section where you define rules that determine how traffics flows.
3. (Optional) Click the default domain to select a different domain.

   Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1.
4. Select a service chain.
5. To add a policy, click **Publish**.
6. Click the ⋮ vertical ellipsis on a section and click **Add Rule**.
7. Edit the **Source** field to add a group by defining membership criteria, static members, IP/MAC addresses, or active directory groups. Membership criteria can be defined from one of these types: Virtual Machine, Logical Switch, Logical Port, IP Set. You can select static members from one of these categories: Group, Segment, Segment Port, Virtual Network Interface, or Virtual Machine.
8. Click **Save**.

9 To add a destination group, edit the **Destination** field.

10 In the Applied To field, you can do one of the following:

- Select **DFW** to apply the rule to all virtual NICs attached to the logical switch.
- Select **VM groups** to apply the rule on virtual NICs of member VMs of the group. Members can be selected from a static list or based on dynamic criteria. The supported NSX-T Data Center objects are: Virtual Machine, Logical Switch, Logical Port, IP Set and so on.

11 In the Action field, select **Redirect** to redirect traffic along the service chain or **Do Not Redirect** not to apply network introspection on the traffic.

12 Click **Publish**.

13 To revert a published rule, select a rule and click **Revert**.

14 To add a policy, click **+ Add Policy**.

15 To clone a policy or a rule, select the policy or rule and click **Clone**.

16 To enable a rule, enable the Enable/Disable icon or select the rule and from the menu click **Enable > Enable Rule**.

17 After enabling or disabling a rule, click **Publish** to enforce the rule.

Results

Traffic going to the source is redirected to the service chain for network introspection. After service profiles in the chain introspect the traffic, it is delivered to the destination.

During deployment, it is possible that the VM group membership for a particular policy changes. NSX-T Data Center notifies the partner Service Manager about these updates.

Configure Network Introspection North-South

After partners register network services with NSX-T Data Center, as an administrator you can configure network services to introspect north-south traffic moving between VMs in a data center and the external network.

High-Level Tasks for North-South Network Security

Follow these steps to set up network security for north-south traffic.

Table 10-5. List of Tasks to Configure North-South Network Introspection

Workflow Tasks	Persona	Implementation
Register Service with NSX-T Data Center	Partner	Only API
Deploy a Service for North-South Traffic Introspection	Administrator	API and NSX Manager UI
Configure Traffic Redirection	Administrator	API and NSX Manager UI

Deploy a Service for North-South Traffic Introspection

After you register a service, you must deploy an instance of the service for the service to start processing network traffic.

Deploy partner service VM at tier-0 or tier-1 logical router that acts as a gateway between the physical world and the logical network on vCenter Server. After you deploy the SVM as a standalone service instance or an active-standby service instance, you can create redirection rules to redirect traffic to the SVM for network introspection.

Prerequisites

- All hosts are managed by a vCenter Server.
- Partner services are registered with NSX-T Data Center and are ready for deployment.
- NSX-T Data Center administrators can access partner services and vendor templates.

  Ensure that the High Availability mode for your logical router is active-standby.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Partner Services > Service Instances > Catalog**.
3. The Catalog tab displays the registered services.
4. Select the service displayed in OVF form factor and click **Deploy** to begin deployment of the service instance.
5. In the Partner Service Insertion window, click **Proceed**.
6. In the Partner Service window, enter the details.

   Table 10-6. Partner Service Details

Field	Description
Instance Name	Enter a name to identify the service instance.
Description	Description about the service instance.
Partner Service	Select the partner service registered with NSX-T Data Center.

Table 10-6. Partner Service Details (continued)

Field	Description
Deployment Specification	Select the form factor to deploy.
Logical Router	Select the tier-0 logical router where the service instance must be deployed.

7 Click **Next**.

8 In the Instance Configuration window, enter the details.

Table 10-7. Service Instance Details

Field	Description
Deployment Mode	Select **Standalone** to deploy a single service instance at the tier-0 logical router. Select **High Availability** to deploy a couple of service instances in active-standby mode at the tier-0 logical router.
Failure Policy	Select **Allow** or **Block**.
Service Instance IP Address	Enter the IP address to be used by the service instance.
Gateway	Enter the gateway address.
Subnet Mask	Enter the subnet mask.
Network ID	Enter the network ID of the logical switch where you want to connect the management network.
Compute Manager	Select the registered vCenter Server.
Resource Pool	Select the resource pool that provides resources to deploy the service instance.
Datastore	Select the repository to store service instance data.

9 Click **Next**.

10 In the Advanced Configuration window, enter the details.

Table 10-8.

Field	Description
Deployment Template	Select the template to be used during deployment of the service instance.
License	Enter the license of the template.

11 Click **Finish**.

Results

The Service Instances tab displays the deployment progress. It might take a few minutes for deployment to finish. Verify the deployment state to ensure that the service instance is successfully deployed at the tier-0 logical router.

Alternatively, go to the vCenter Server and verify the deployment status.

What to do next

Configure rules to redirect traffic to the service instance deployed at the tier-0 router. See Configure Traffic Redirection

Configure Traffic Redirection

After you deploy a service instance, configure the type of traffic that the router redirects to the service. Configuring traffic redirection is similar to configuring a firewall.

For information about configuring a firewall, see Firewall Sections and Firewall Rules.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Partner Services > Service Instances** .
3. Click the service instance.
4. Click the **Traffic Redirection** tab.
5. To add a section, select an existing section and click **Add Section**.
   - From the menu, select **Add Section Above** or **Add Section Below**.

   A new section is created. The traffic type to be redirected is set to **L3 Redirect**, service is of the type **Stateless**, the **Applied To** field is associated to a Tier-0 logical router that is configured on the host. After you define rules, the **Rules** field is auto-populated.
6. Click **Publish** to persist configuration details of the section.
7. To add a rule within that section, select the section and click **Add Rule**.
8. In the rule row, enter the following details:
   a. Enter rule name.
   b. Enter the source and destination of L3 traffic. The partner service VM introspects traffic flowing in from the source before redirecting it to the destination VM.
   c. In the **Applied To** field, select the uplink of Tier-0 router.
   d. In the **Action** field, select **Redirect** if traffic needs to be introspected by the service VMs or select **Don't Redirect** if traffic does not need to be introspected for north-south introspection.
9. Each rule can be enabled individually. After you enable a rule, it is applied to the traffic that matches the rule.
10. Click Advanced Settings to configure the traffic direction and to enable logging.

11. At the end of a section containing rules, click **Publish** to persist the rules in the section or click **Revert** to cancel the operation.

Results

The traffic is sent to network introspection rules where policy rules are applied to the traffic.

What to do next

See Add Redirection Rules for North-South Traffic.

Add Redirection Rules for North-South Traffic

Use the **Advanced Networking and Security** UI to set up north-south redirection rules. Traffic redirection happens only for services inserted at the Tier-0 router.

Follow instructions at Configure Traffic Redirection.

Prerequisites

- Register and deploy third-party services on NSX-T.
- Configure Tier-0 router.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. **Security > North South Firewall > Network Introspection (N-S) > Add Policy**.

   A policy section is similar to a firewall section where you define rules that determine how traffics flows.
3. (Optional) Click the default domain to select a different domain.

   Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1.
4. Set **Redirection To** to the service instance that is registered with NSX-T to perform network introspection of traffic flowing between source and destination entities.
5. To add a policy, click **Publish**.
6. Click the ⋮ vertical ellipsis on a section and click **Add Rule**.
7. Edit the **Source** field to add a group by defining membership criteria, static members, IP/MAC addresses, or active directory groups. Membership criteria can be defined from one of these types: Virtual Machine, Logical Switch, Logical Port, IP Set. You can select static members from one of these categories: Group, Segment, Segment Port, Virtual Network Interface, or Virtual Machine.
8. Click **Save**.
9. To add a destination group, edit the **Destination** field.

10 In the Applied To field, you can do one of the following:

- Select **DFW** to apply the rule to all virtual NICs attached to the logical switch.
- Select **VM groups** to apply the rule on virtual NICs of member VMs of the group. Members can be selected from a static list or based on dynamic criteria. The supported NSX-T Data Center objects are: Virtual Machine, Logical Switch, Logical Port, IP Set and so on.

11 In the Action field, select **Redirect** to redirect traffic along the service instance or **Do Not Redirect** not to apply network introspection on the traffic.

12 Click **Publish**.

13 To revert a published rule, select a rule and click **Revert**.

14 To add a policy, click **+ Add Policy**.

15 To clone a policy or a rule, select the policy or rule and click **Clone**.

16 To enable a rule, enable the Enable/Disable icon or select the rule and from the menu click **Enable > Enable Rule**.

17 After enabling or disabling a rule, click **Publish** to enforce the rule.

Results

Based on the actions set, north-south traffic is redirected to the service instance for network introspection.

Monitor Traffic Redirection

After you deploy a service instance and configure traffic redirection, you can monitor the amount of traffic that goes into and out of the service instance.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Partner Services > Service Instances**.

3 Click the name of a service instance.

The **Overview** tab shows the configuration and status of the service instance.

4 Click the **Statistics** tab.

Information about the number of packets and the amount of data that go into and out of the service instance is displayed.

5 Click **Refresh** to update the statistics..

Configure Endpoint Protection

Apply endpoint protection policies to guest VM groups after partners register their services with NSX-T Data Center. Before you configure endpoint protection for guest VMs, you must deploy partner services as part of the service insertion workflow.

Understand Endpoint Protection

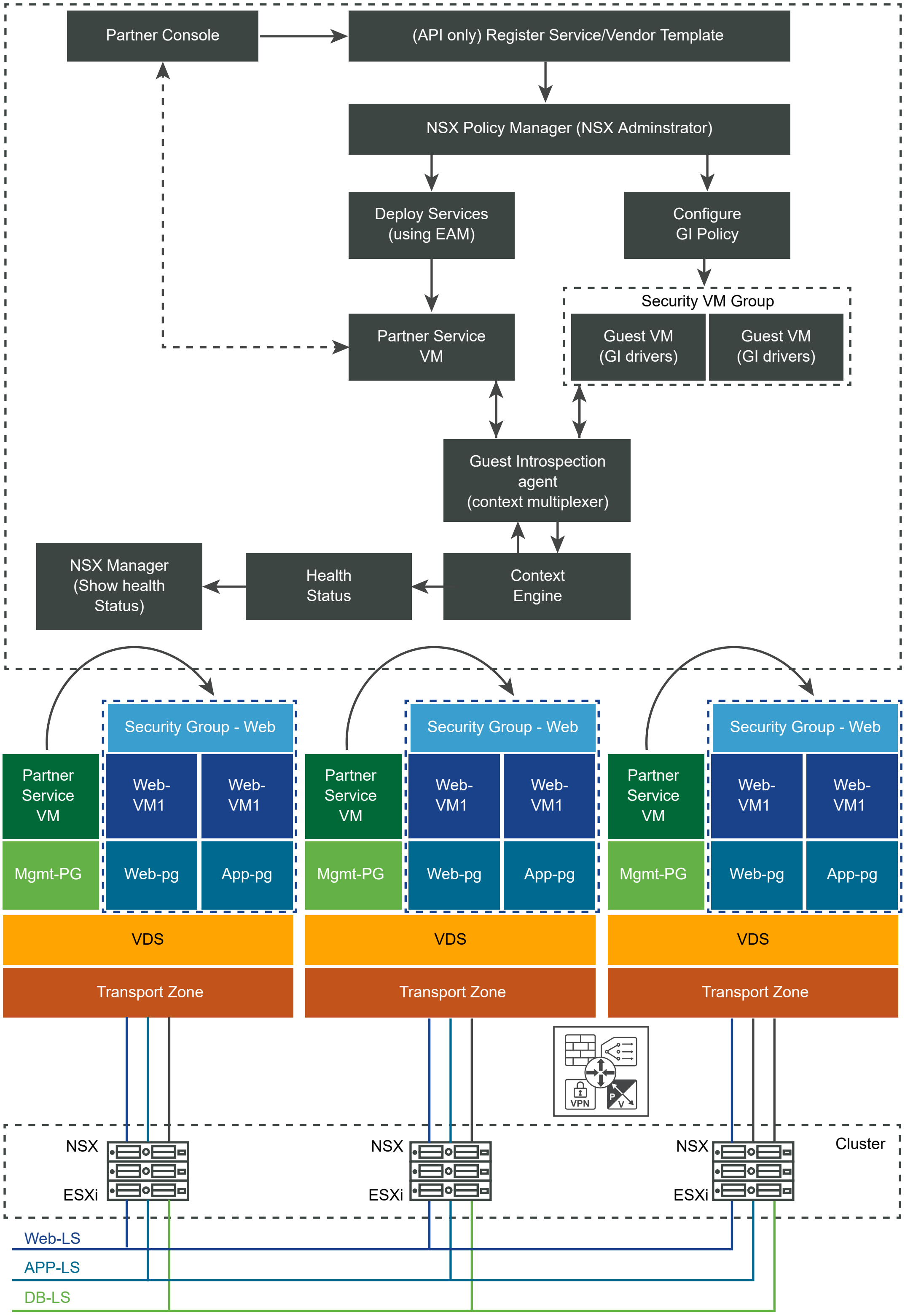
Know the use case, workflow, and key concepts of endpoint protection.

Endpoint Protection Use Case

NSX-T provides L2-L4 stateful firewall services to virtual networks. If your environment requires anitmalware security services to protect guest VMs, NSX provides a powerful way of introspecting guest VMs by integrating services from third-party vendors in hosts to protect against malware.

During host node preparation, NSX-T installs the Guest Introspection host agent as part of host bundle installation on all hosts of a cluster. Thus the Guest Introspection host agent does not need to be separately installed on a host node. The partner service virtual machine (SVM) is installed as a virtual appliance on a host node. The SVM uses Guest Introspection API library (EPSec API library) to introspect and protect guest VMs from malware.

Figure 10-1. Endpoint Protection Use Case



As an NSX administrator you implement an antimalware solution that is deployed as a Service Virtual Machine (Service VM, or SVM) to monitor a file activity on a Guest VM. Whenever a file is accessed, such as a file open attempt, the antimalware Service VM is notified of the event. The Service VM then determines how to respond to the event-for example, to inspect the file for virus signatures.

- If the Service VM determines that the file contains no viruses, then it allows the file open operation to succeed.
- If the Service VM detects a virus in the file, it attempts to clean it.
  - If the file is cleaned successfully, the Service VM allows the file open operation to succeed.
  - If the Service VM is not able to clean the file, then it prevents the file open operation and tags the file (and the VM) as infected. Moreover, you can define a rule that automatically moved the VM to a security group that contains infected VMs.

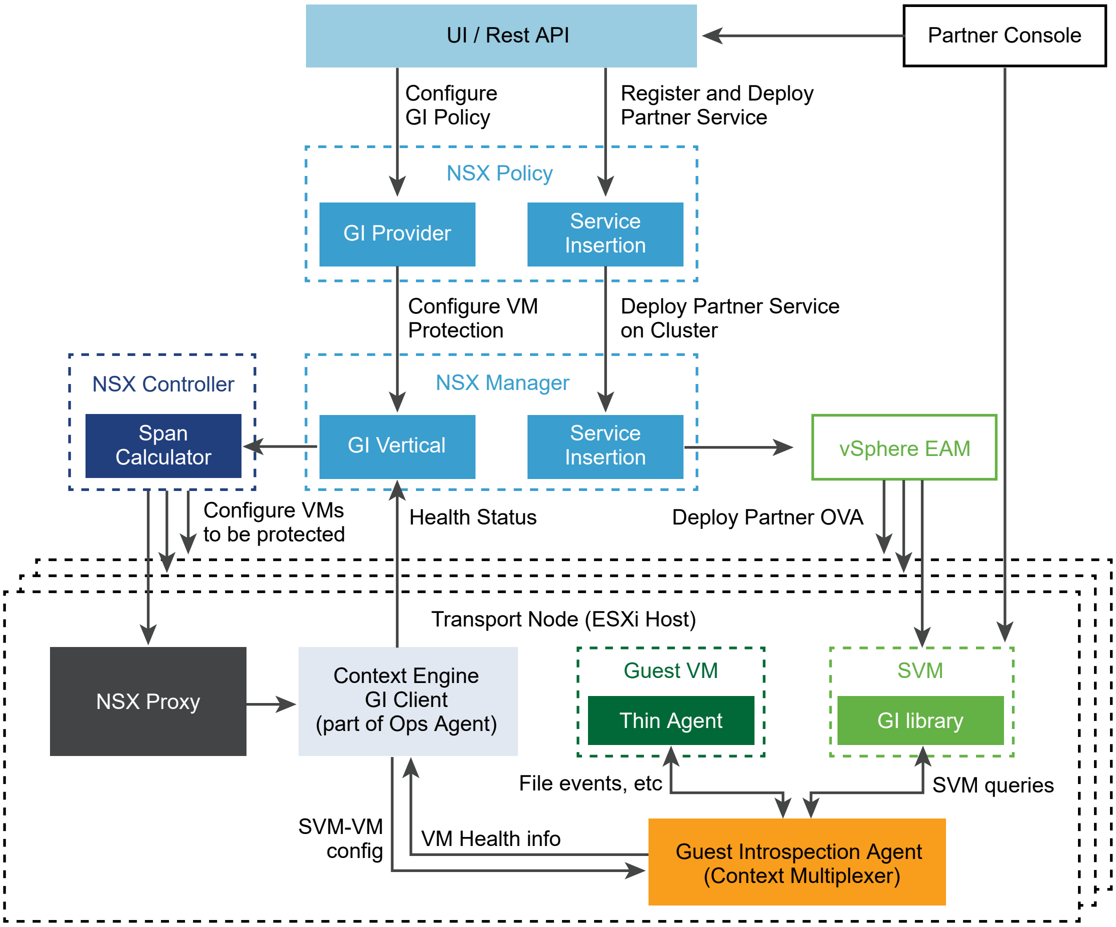
Note In case guest VMs are disconnected from or are unable to reach an ESXi host agent (MUX) or the SVM, then file accesses on the guest may be allowed without going through an antivirus scan.

Unlike a guest VM, which can go offline, a Service VM is continuously running. Therefore, it can continuously update antivirus signatures, provide an uninterrupted protection to the virtual machines on the host, and provide an immediate protection to new VMs that come online. Because Guest Introspection enables Service VMs to read and write specific files on guest VMs, it provides an efficient way to avoid resource bottlenecks and optimize the memory use.

Guest Introspection Architecture

Understand the architecture of service insertion and guest introspection components in NSX-T Data Center.

Figure 10-2. Guest Introspection Architecture



Partner registration:

- Partners register services by invoking Guest Introspection Rest API library, provided by NSX Manager APIs.
- Later in the workflow, when partner services (service VMs) are deployed on a host, partner console registers with the SVM to receive notifications related to maintenance activities and event notifications happening on guest VM groups.

Service Deployment:

- Partner services are deployed on an NSX prepared host using the service insertion framework.
- The vSphere Enterprise Agency Manager (EAM) deploys the partner service VMs on NSX-T hosts.
- Each host of the cluster runs an instance of the service, which is an SVM.

Guest Introspection Driver Installation:

- Before getting SVM to communicate with guest VMs and other components, install GI drivers on guest VMs.

- Administrator uses VMTools to install a Thin Agent on every Guest VM.
- Thin Agents perform the following functions.
  - Communicates with a component known as the Guest Introspection agent (MUX) over a fast channel, called Virtual Machine Communication Interface (VMCI).
  - Captures file access events on guest VMs.
  - Notifies the partner SVM about events on guest VMs.
  - Implements protection policy on guest VMs. For example, allow or deny a file access, or quarantine a file or VM.

Policy creation:

An admin creates a policy that protects a VM group by associating the VM group with service profiles.

- NSX Policy Manager composes the GI policies and interacts with the GI component (running on NSX Manager).
- This GI component is responsible for configuring GI policies on VM groups and for sending this configuration to the control plane, specifically the CCP Span Calculator component.

The control plane manages VM configuration:

- The Control plane receives configuration about GI policies applied to VM groups. It calculates the span of transport nodes that host the VMs from a particular group.
- CCP Span calculator: NSX Manager sends configuration details of a group - VMs and its associated policy, to the CCP. The span calculator determines the transport nodes on which these VMs belong to. It then pushes the VM ID list along with the associated policy to transport nodes that hosts these VMs. LCP receives this information and stores in a database on the host.
- Context engine listens on any updates made to the database and updates the Guest Introspection agent (MUX) component.

Establish communication between SVM , Guest VM, and Context Multiplexer:

- SVM: Partner services run on a separate appliance known as the service VM (SVM) on each host of a cluster. Partners provide the OVF location to deploy SVMs while registering services. It communicates with the following components:
  - A guest VM and Guest Introspection agent communicate over a fast channel (VMCI) on the ESXi hypervisor, whereas Guest VM and SVM communicate over a TCP/IP channel. The thin running inside a guest VM gathers information about the OS, file activities. SVMs gather context provided by the thin agent through the EPSec API library. The GI drivers sends events to SVMs. SVMs determines whether the file is malware or a clean file. SVM reads into the EPSec API library to determine an action based on the context gathered.

- Once SVMs are deployed on each host of the cluster, the Guest Introspection components in NSX Manager sends down the SVM configuration down to Context Engine. The Context Engine updates the Guest Introspection Agent with a new SVM configuration information. SVM registers for any events happening on a VM, or a file.

  The Guest Introspection agent establishes communication with the guest introspection library, which results in the SVM receiving a VM Power on event. The SVM is now ready to receive file events from the thin agent.
- Guest Introspection agent: It is the Guest Introspection host module (context multiplexer) that multiplexes and forwards messages from all protected Guest VMs to the SVM. It is installed as a vSphere Installation Bundle (VIB) on NSX-T hosts. The NSX Manager installs and configures this module on the ESX host. The Guest Introspection agent configuration file (`/var/run/muxconfig.xml`) on the host specifies configuration information about the partner solution. The `VMConfig` file lists protected VMs and corresponding solution. The `SolutionConfig` file lists SVM details such as includes solution ID, IP address, listener port, UUID.

Role of Context Engine:

- Context Engine: This component sends configuration details of SVM associated to VMs to Guest Introspection agent. Upon receiving configuration details, Guest Introspection agent records SVM configuration updates in the `muxconfig.xml` file. The configuration information also contains the service profile tag for the SVM to query and identify the policy. During introspection, Guest Introspection agent forwards events only from VMs associated to that SVM. This component is responsible for sending the health status of Thin Agent and Guest Introspection agent to the GI Vertical component in the NSX Manager.
- Health Status: The GI component (running on NSX Manager) requests health information periodically from the Context Engine.
- Context Engine gathers health status information from Guest Introspection agent and sends it to GI component (running on NSX Manager). The health status is determined by the following factors: status of the partner solution, connectivity between Guest Introspection agent (Context Multiplexer) and Context Engine (Ops Agent), and availability of Guest Introspection agent information, SVM protocol information with NSX Manager.

Key Concepts of Endpoint Protection

Guest VMs are protected from malware. The endpoint protection workflow needs partners to register their services with NSX-T Data Center and an administrator to consume these services. There are a few concepts that aid your understanding of the workflow.

- Service Definition: Partners define services with these attributes: name, description, supported form factors, deployment attributes such as storage, network stores.
- Service Insertion: NSX provides a framework that allows partners use the Service Definition API to register their services with NSX-T. The service insertion is to deploy partner services on hosts to perform guest introspection against malware.

- Span Calculator: For a VM group that needs to be protected, the control plane finds out which transport nodes host the VMs that are part of this group. A VM group may have VMs hosted on different transport nodes. The control plane calculates the span of transport nodes that host these VMs. After calculating the span, NSX Manager pushes the VM configuration (a VM and its associated policy) to each transport node. This is required as transport nodes need to know the policy associated with VMs. The control plane also pushes the VM Id list along with the SVM Policy to transport nodes.
- Service Profiles and Vendor Templates: Partners register vendor templates which expose protection levels for policies. Protection levels can be Gold, Silver or Platinum. Vendor templates might also provide deployment attributes that are specific to partners, such as name or license key, and so on. These attributes are part of the service definition. These attributes allow an NSX admin to customize the vendor template to create many service profiles from a single vendor template. If there are no deployment attributes made available in the vendor template, then the admin can only create a single service profile from that vendor template.
- Guest Introspection Library and SVM: Guest Introspection library (earlier known as EPSec) is a library that runs on the partner SVM. It also acts as an interface between the partner SVM and the Guest Introspection Thin Agent.
- The Guest Introspection agent (MUX) and SVM: This component is responsible for forwarding Guest Introspection Thin Agent events to the configured SVMs. It also forwards SVM requests to Guest Introspection Thin Agent.
- Context Engine GI Client: This component is responsible for :
  - Sending health status of the Thin Agent and Guest Introspection agent (MUX) to the GI component in NSX Manager.
  - Providing NestDb configuration to the Guest Introspection agent (MUX).
- Health Status: Context Engine sends health status of SVM, VM health, Guest Introspection agent health, Guest Introspection Client health to the Guest Introspection (running on the NSX Manager).
- Domain and VM groups: Domain is an environment that hosts VM groups and policy rules. VM groups are a list of VMs that are hosted on a single or multiple transport nodes. The NSX administrator creates a group of VMs in a domain before applying protection policy to that VM group. For example, a domain can be created for a PCI-DSS security domain, which consists of different VM groups that need to compliant to highest security standards. Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1. In NSX-T Data Center 2.4.1 it is not necessary to create any domain.
- Sequence Number: Determines the sequence of executing rules across multiple domains. If there are multiple domains, each having rules, guest introspection sequences rules from a higher ranked domain and then sequences rules from a lower ranked domain till all the rules are sequenced. Once the rules are published, they immediately are applied to the VM groups

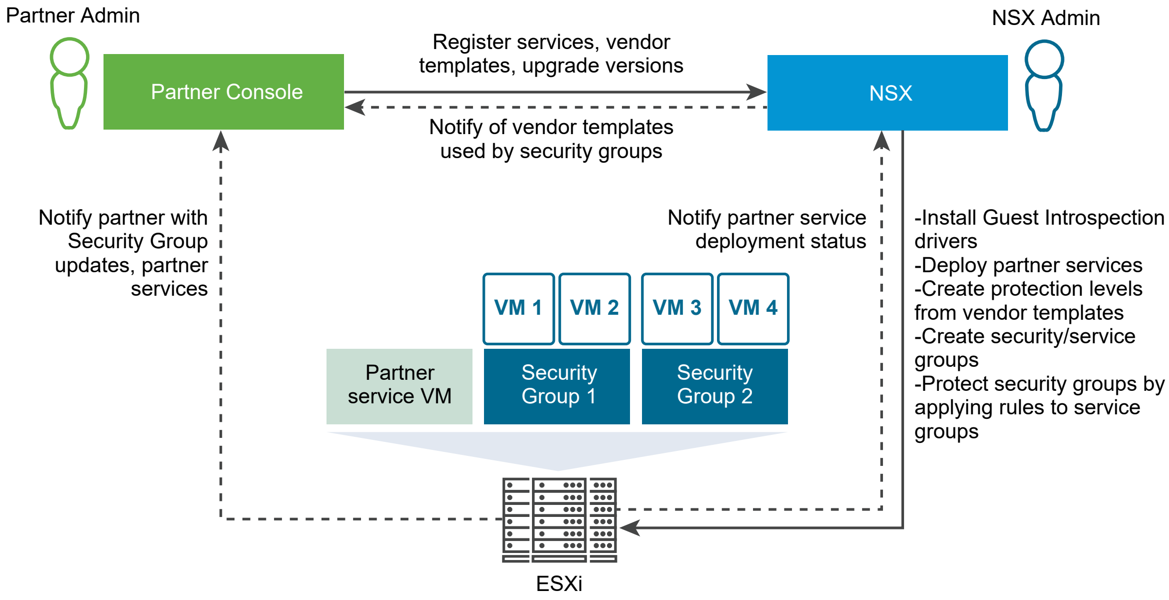
that need to be protected and guest introspection begins. The sequence number can be defined explicitly through API calls or through the UI. Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1. In NSX-T Data Center 2.4.1 it is not necessary to create any domain.

Endpoint Protection Workflow

In the first part of the workflow, partners register services with NSX-T. In the last part of the workflow, the NSX administrator deploys the registered services and applies end point protection policies to VM groups.

The guest introspection workflow for endpoint protection is as follows:

Figure 10-3. Endpoint Protection Workflow


Partner Admin
Partner Console
Register services, vendor templates, upgrade versions
Notify of vendor templates used by security groups
NSX
NSX Admin
Notify partner with Security Group updates, partner services
Notify partner service deployment status
-Install Guest Introspection drivers
-Deploy partner services
-Create protection levels from vendor templates
-Create security/service groups
-Protect security groups by applying rules to service groups
VM 1
VM 2
VM 3
VM 4
Partner service VM
Security Group 1
Security Group 2
ESXi

At a high-level, partner services prepare a service virtual machine (SVM) by consuming EPSec API (GI) libraries. Service registration happens through the partner Service Manager console by invoking NSX-T Policy APIs. The Service Manager console is managed by partners. In addition to services, partners also register vendor templates that contain configuration to protect guest VMs when they are applied in NSX-T. After registration, NSX administrators must bind the service with a specific IP address and port number to the partner Service Manager.

After partners register their services, an NSX-T administrator can view all the registered partner services on the NSX-T Policy Manager user interface. The administrator deploys these services on a cluster. When the deployment is complete, each host of the cluster runs an SVM, which runs the security engine. SVMs use the EPSec API library to communicate with guest VMs to intercept events. To apply policies on guest VMs, admins specify rules that associate VM groups with service profiles (an instance of vendor template), which defines what type of protection level is applied to the guest VMs.

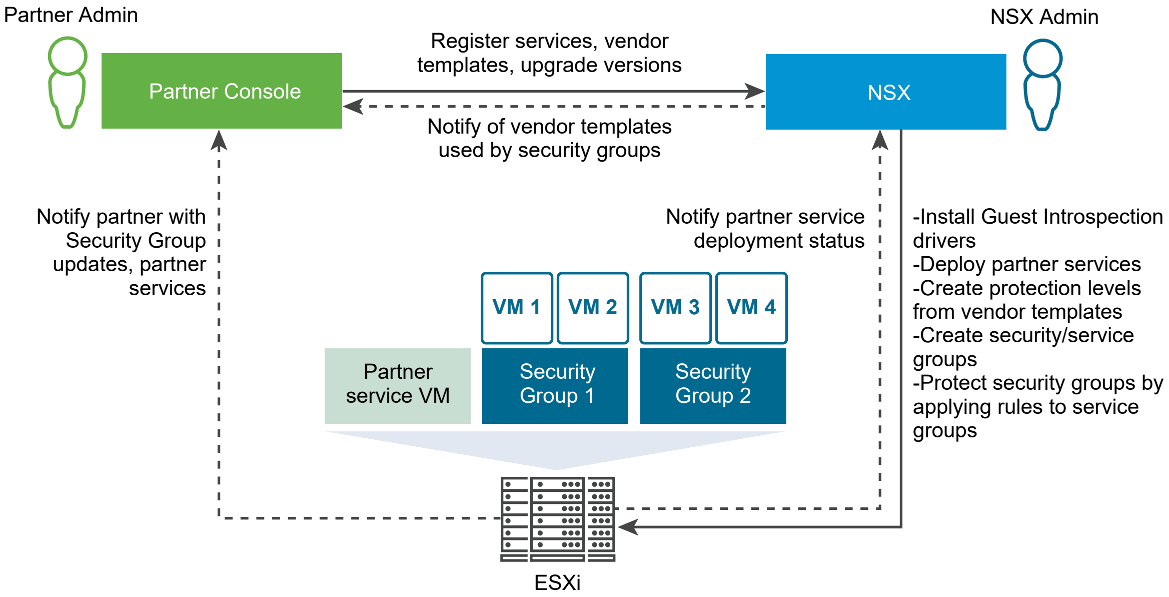
After deploying and configuring guest introspection services, the SVM starts introspecting the guest VMs. When an event occurs on a guest VM, SVM intercepts and remediates the event. SVM also notifies the partner console and the NSX-T Manager.

Endpoint Protection Workflow

In the first part of the workflow, partners register services with NSX-T. In the last part of the workflow, the NSX administrator deploys the registered services and applies end point protection policies to VM groups.

The guest introspection workflow for endpoint protection is as follows:

Figure 10-4. Endpoint Protection Workflow


Partner Admin
Partner Console
Register services, vendor templates, upgrade versions
NSX Admin
NSX
Notify of vendor templates used by security groups
Notify partner with Security Group updates, partner services
Notify partner service deployment status
-Install Guest Introspection drivers
-Deploy partner services
-Create protection levels from vendor templates
-Create security/service groups
-Protect security groups by applying rules to service groups
VM 1
VM 2
VM 3
VM 4
Partner service VM
Security Group 1
Security Group 2
ESXi

At a high-level, partner services prepare a service virtual machine (SVM) by consuming EPSec API (GI) libraries. Service registration happens through the partner Service Manager console by invoking NSX-T Policy APIs. The Service Manager console is managed by partners. In addition to services, partners also register vendor templates that contain configuration to protect guest VMs when they are applied in NSX-T. After registration, NSX administrators must bind the service with a specific IP address and port number to the partner Service Manager.

After partners register their services, an NSX-T administrator can view all the registered partner services on the NSX-T Policy Manager user interface. The administrator deploys these services on a cluster. When the deployment is complete, each host of the cluster runs an SVM, which runs the security engine. SVMs use the EPSec API library to communicate with guest VMs to intercept events. To apply policies on guest VMs, admins specify rules that associate VM groups with service profiles (an instance of vendor template), which defines what type of protection level is applied to the guest VMs.

After deploying and configuring guest introspection services, the SVM starts introspecting the guest VMs. When an event occurs on a guest VM, SVM intercepts and remediates the event. SVM also notifies the partner console and the NSX-T Manager.

Prerequisites to Configure Endpoint Protection

Before you configure endpoint protection for guest VMs, ensure that the prerequisites are met.

Prerequisites

- NSX Manager is installed on all the hosts.
- Prepare and configure NSX-T Data Center cluster as transport nodes by applying transport node profiles. After the host is configured as the transport node, guest introspection components are installed. See *NSX-T Data Center Installation Guide*.
- Partner console is installed and configured to register services with NSX-T Data Center.
- Ensure that the guest VMs run VM Hardware Configuration file version 9 or higher.
- Configure VMware Tools and install thin agents.
  - See Install the Guest Introspection Thin Agent on Linux Virtual Machines.
  - See Install the Guest Introspection Thin Agent on Windows Virtual Machines.
  - See Install the Linux Thin Agent for Network Introspection.

Install the Guest Introspection Thin Agent on Windows Virtual Machines

To protect VMs using a Guest Introspection security solution, you must install Guest Introspection thin agent, also called Guest Introspection drivers, on the VM. Guest Introspection drivers are included with VMware Tools for Windows, but are not part of the default installation. To install Guest Introspection on a Windows VM, you must perform a custom install and select the drivers.

Windows virtual machines with the Guest Introspection drivers installed are automatically protected whenever they are started up on an ESXi host that has the security solution installed. Protected virtual machines retain the security protection through shut downs and restarts, and even after a vMotion move to another ESXi host with the security solution installed.

- If you are using vSphere 6.0, see these instructions for installing VMware Tools, see Manually Install or Upgrade VMware Tools in a Windows Virtual Machine.
- If you are using vSphere 6.5, see these instructions for installing VMware Tools: https://www.vmware.com/support/pubs/vmware-tools-pubs.html.

Prerequisites

Ensure that the guest virtual machine has a supported version of Windows installed. The following Windows operating systems are supported for NSX Guest Introspection:

- Windows XP SP3 and above (32 bit)
- Windows Vista (32 bit)

- Windows 7 (32/64 bit)
- Windows 8 (32/64 bit)
- Windows 8.1 (32/64) (vSphere 6.0 and later)
- Windows 10
- Windows 2003 SP2 and above (32/64 bit)
- Windows 2003 R2 (32/64 bit)
- Windows 2008 (32/64 bit)
- Windows 2008 R2 (64 bit)
- Win2012 (64)
- Win2012 R2 (64) (vSphere 6.0 and later)

Procedure

1 Start the VMware Tools installation, following the instructions for your version of vSphere. Select **Custom install**.

2 Expand the VMCI Driver section.

The options available vary depending on the version of VMware Tools.

3 Select the driver to be installed on the VM.

Driver	Description
vShield Endpoint Drivers	Installs File Introspection (`vsepflt`) and Network Introspection (`vnetflt`) drivers.
Guest Introspection Drivers	Installs File Introspection (`vsepflt`) and Network Introspection (`vnetflt`) drivers.
NSX File Introspection Driver and NSX Network Introspection Driver	Select NSX File Introspection Driver to install vsepflt. Optionally select NSX Network Introspection Driver to install `vnetflt` (`vnetWFP` on Windows 10 or later). **Note** Select NSX Network Introspection Driver only if you are using the Identity Firewall or Endpoint Monitoring features.

4 In the drop-down menu next to the drivers you want to add, select This feature is installed on the local hard drive.

5 Follow the remaining steps in the procedure.

What to do next

Verify whether the thin agent is running using the `fltmc` command with the administrative privileges. The Filter Name column in the output lists the thin agent with an entry `vsepflt`.

Install the Guest Introspection Thin Agent on Linux Virtual Machines

Guest Introspection supports File Introspection in Linux for anti-virus only. To protect Linux VMs using a Guest Introspection security solution, you must install the Guest Introspection thin agent.

The Linux thin agent is available as part of the operating system specific packages (OSPs). The packages are hosted on VMware packages portal. Enterprise or Security Administrator (non-NSX Administrator) can install the agent on guest VMs outside of NSX.

Installing VMware Tools is not required.

Based on your Linux operating system, perform the following steps with root privilege:

Prerequisites

- Ensure that the guest virtual machine has a supported version of Linux installed:
  - Red Hat Enterprise Linux (RHEL) 7.4 (64 bit) GA
  - SUSE Linux Enterprise Server (SLES) 12 (64 bit) GA
  - Ubuntu 16.04.5 LTS (64 bit) GA
  - CentOS 7.4 GA
- Verify GLib 2.0 is installed on the Linux VM.

Procedure

1. For Ubuntu systems
   a. Obtain and import the VMware packaging public keys using the following commands.

      ```
      curl -O https://packages.vmware.com/packages/nsx-gi/keys/VMWARE-PACKAGING-NSX-GI-GPG-RSA-
      KEY.pub
      apt-key add VMWARE-PACKAGING-NSX-GI-GPG-RSA-KEY.pub
      ```

   b. Create a new file named `vmware.list` file under `/etc/apt/sources.list.d`

   c. Edit the file with the following content:

      ```
      deb [arch=amd64] https://packages.vmware.com/packages/nsx-gi/latest/ubuntu/ xenial main
      ```

   d. Install the package.

      ```
      apt-get update
      apt-get install vmware-nsx-gi-file
      ```

2 For RHEL7 systems

  a Obtain and import the VMware packaging public keys using the following commands.

```
curl -O https://packages.vmware.com/packages/nsx-gi/keys/VMWARE-PACKAGING-NSX-GI-GPG-RSA-
KEY.pub
rpm --import VMWARE-PACKAGING-NSX-GI-GPG-RSA-KEY.pub
```

  b Create a new file named `vmware.repo` file under `/etc/yum.repos.d`.

  c Edit the file with the following content:

```
[vmware]
name = VMware
baseurl = https://packages.vmware.com/packages/nsx-gi/latest/rhel7/x86_64
enabled = 1
gpgcheck = 1
metadata_expire = 86400
ui_repoid_vars = basearch
```

3 Install the package.

```
yum install vmware-nsx-gi-file
```

4 For SLES systems

  a Obtain and import the VMware packaging public keys using the following commands.

```
curl -O https://packages.vmware.com/packages/nsx-gi/keys/VMWARE-PACKAGING-NSX-GI-GPG-RSA-
KEY.pub
rpm --import VMWARE-PACKAGING-NSX-GI-GPG-RSA-KEY.pub
```

  b Add the following repository:

```
zypper ar -f "https://packages.vmware.com/packages/nsx-gi/latest/sle12/x86_64/" VMware
```

  c Install the package.

```
zypper install vmware-nsx-gi-file
```

5 For CentOS systems

   a Obtain and import the VMware packaging public keys using the following commands.

   ```
   curl -O https://packages.vmware.com/packages/nsx-gi/keys/VMWARE-PACKAGING-NSX-GI-GPG-RSA-
   KEY.pub
   rpm --import VMWARE-PACKAGING-NSX-GI-GPG-RSA-KEY.pub
   ```

   b Create a new file named `vmware.repo` file under `/etc/yum.repos.d`.

   c Edit the file with the following content:

   ```
   [vmware]
   name = VMware
   baseurl = https://packages.vmware.com/packages/nsx-gi/latest/centos7/x86_64
   enabled = 1
   gpgcheck = 1
   metadata_expire = 86400
   ui_repoid_vars = basearch
   ```

What to do next

Verify whether the thin agent is running using the service `vsepd status` command with the administrative privileges. The status must be running.

Install the Linux Thin Agent for Network Introspection

Install the Linux thin agent to introspect network traffic.

Important To protect guest VMs against antivirus, you do not need to install the Linux thin agent for network introspection.

The Linux thin agent driver that is used to introspect network traffic depends on an open-source driver.

Prerequisites

Install the following packages:

- glib2
- libnetfilter-conntrack3/ libnetfilter-conntrack
- libnetfilter-queue1/ libnetfilter-queue
- iptables

Procedure

1 To install the open-source driver provided by guest introspection.

  a Add following URL as the base URL for your operating system.

  ```
  deb [arch=amd64] https://packages.vmware.com/guest-introspection-for-vmware-nsx/latest/
  ```

  b Import the VMware packaging key.

  ```
  https://packages.vmware.com/packages/nsx-gi/keys/VMWARE-PACKAGING-NSX-GI-GPG-RSA-KEY.pub
  ```

  c Update the repository and install the open-source driver.

  ```
  apt-get install Guest-Introspection-for-VMware-NSX
  ```

2 To install the Linux thin agent that is used to introspect file and or network traffic.

  - To install file and network introspection packages, select `vmware-nsx-gi` package in step c.
  - To install network introspection packages, select the `vmware-nsx-gi-net` package in step c.

  a Add following URL as the base URL for your operating system.

  ```
  deb [arch=amd64] https://packages.vmware.com/packages/nsx-gi/latest
  ```

  b Import the VMware packaging key.

  ```
  https://packages.vmware.com/packages/nsx-gi/keys/VMWARE-PACKAGING-NSX-GI-GPG-RSA-KEY.pub
  ```

  c Install one of the drivers.

  ```
  vmware-nsx-gi
  vmware-nsx-gi-net
  ```

Supported Software

Guest Introspection is interoperable with specific versions of software.

VMware Tools

VMware Tool 10.3.10 version is supported.

Check out interoperability between VMware Tools and NSX-T. See VMware Product Interoperability Matrices.

Supported OS

Only Microsoft Windows operating system is supported.

- Windows 7
- Windows 8/8.1

- Windows 10
- Windows 2008 server R2
- Windows 2012 server R2
- Windows 2016 Server

Supported Hosts

For supported ESXi hosts, see the VMware Product Interoperability Matrices.

Create a User with Guest Introspection Partner Admin Role

Assign a user with the Guest Introspection Partner Admin role that is available in NSX-T Data Center.

Note: It is recommended to register partner services by a user that is associated with the Guest Introspection Partner Admin role to avoid any security issues.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System** → **User**→ **Role Assignments**.
3. Click **Add**.
4. Select the user and assign that user the `GI Partner Admin` role.

What to do next

Register services with NSX-T Data Center. See Register a Service with NSX-T Data Center.

Register a Service with NSX-T Data Center

Register third-party security services with NSX-T Data Center.

Prerequisites

- Ensure that prerequisites are met. See Prerequisites to Configure Endpoint Protection.
- Ensure that a vIDM user is assigned the GI Partner Admin role. This role is used to register services with NSX-T Data Center.

Procedure

1. Log in with the GI Partner Admin privileges to the partner console.
2. Register a service, vendor template, and configure the partner solution with NSX-T Data Center. See partner documentation.

What to do next

View catalog of partner services. See View Catalog of Partner Services.

View Catalog of Partner Services

The catalog page displays all the partners and their services that are registered with NSX-T Data Center.

Prerequisites

- Partners register services with NSX-T Data Center.
- Services are deployed on a cluster.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Service Deployments > Catalog**.
3. Click **View** on a service. The Deployment page displays the details about the service, such as status of deployment, network details, cluster details, and so on.

What to do next

Deploy a service. See Deploy a Service.

Deploy a Service

After you register a service, you must deploy an instance of the service for the service to start processing network traffic.

Deploy partner service VMs that run the partner security engine on all the NSX-T Data Center hosts in a cluster. The vSphere ESX Agency Manager (EAM) service is used to deploy the partner service VMs on each host. After you deploy the SVMs, you can create policy rules used by SVM to protect guest VMs.

Prerequisites

- All hosts are managed by a vCenter Server.
- Partner services are registered with NSX-T Data Center and are ready for deployment.
- NSX-T Data Center administrators can access partner services and vendor templates.
- Both the service VM and the partner Service Manager (console) must be able to communicate with each other at the management network level.
- Prepare hosts as NSX-T Data Center transport nodes:
  - Create a transport zone.
  - Create an IP pool for tunnel endpoint IP addresses.
  - Create an uplink profile.
  - Add a transport node profile to prepare a cluster for auto deployment of NSX-T Data Center transport nodes.

- Configure a standalone or managed host.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Go to the **System** tab and click **Service Deployment**.
3. Click **Deployment** and click **Deploy Service**.
4. Enter the service deployment name.
5. In the Compute Manager field, select the compute resource on the vCenter Server to deploy the service.
6. In the Cluster field, select the cluster where the services need to be deployed.
7. In the Data Store drop-down menu, you can:
   a. Select a datastore as the repository for the service virtual machine.
   b. Select **Configure on Host**. This setting means that you do not need to select a datastore and port group on this wizard. You can directly configure agent settings on EAM in vCenter Server to point to a specific datastore and port group to be used for service deployment. Proceed to step 11.

   To know how to configure EAM, refer to the vSphere documentation.
8. In the Network column, click **Set** and enter the Management Network interface by selecting a DHCP or static IP address type, control network and data network.
9. In the Deployment Specification field, select the service and the form factor of the service VM to be deployed on cluster hosts. There can be multiple services available for deployment.
10. In the Deployment Template field, select the vendor template with attributes to protect the workload you want to run on guest VMs groups.
11. Click **Save**.

Results

When a new host is added to the cluster, EAM automatically deploys the service VM on the new host. The deployment process might take some time, depending on the vendor's implementation. You can view the status in the NSX Manager user interface. The service is successfully deployed on the host when the status turns `Deployment Successful`.

To remove host from a cluster, first move it into maintenance mode. Then, select the option to migrate the guest VMs to another host to complete migration.

What to do next

Know deployment details and heath status about service instances deployed on hosts. See View Service Instance Details.

View Service Instance Details

Know deployment details and health status of service instance deployed on member hosts of a cluster.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Service Deployments > Service Instances**.
3. From the Partner Service drop-down menu, select the partner service to view details related to service instances.

Table 10-9.

Field	Description
Service Instance Name	A unique ID identifying the service instance on a particular host.
Service Deployment Name	A unique ID identifying the service definition
Deployed To	Host IP address
Deployment Mode	Cluster or Standalone
Deployment Status	Up status to determine a successful deployment
Health Status	Health status is Up when the following parameters are successfully realized by NSX-T Data Center. ■ Solution status: Up ■ Connectivity between NSX-T Data Center Guest Introspection agent and NSX-T Data Center Ops Agent: Up ■ Service VM protocol is defined ■ Protocol compatibility between service VM and NSX-T Data Center Guest Introspection agent

What to do next

View catalog of registered services. See View Catalog of Partner Services.

Bring up Service Instance

After deploying the service instance, certain parameters need to be realized in NSX-T Data Center for the health status to be Up.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Service Deployments > Service Instances**.

3 From the Partner Service drop-down menu, select the partner service to view details related to service instances.

4 The Health Status column displays state of the service instance as `Ready`. It indicates that the service instance is ready to be configured with endpoint protection policy rules to protect VMs.

5 The following parameters must be realized in NSX-T Data Center for the health status to change to `Up`.

   - Guest virtual machines must be available on the host.
   - Guest virtual machines must be powered on.
   - Endpoint protection rules must be applied to the guest virtual machines.
   - Guest virtual machines must be configured with the supported version of VMtools and file introspection drivers.

What to do next

Add a service profile. See Add Endpoint Protection Service Profile.

Add Endpoint Protection Service Profile

Guest introspection policies can be implemented only when a service profile is available in NSX-T Data Center. Service profiles are created from a template provided by the partner. Service Profiles are a way for the administrator to choose protection levels (Gold, Silver, Platinum policy) for a VM by choosing the vendor templates provided by the vendor.

For example, a vendor can provide Gold, Platinum, and Silver policy levels. Each profile created might serve a different type of workload. A Gold service profile provides complete antimalware to a PCI-type workload, while a silver service profile only provides basic antimalware protection to a regular workload.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Security > Endpoint Protection > Service Profiles** .

3 From the Partner Service field, select the service for which you want to create a service profile.

4 Click **Add Service Profile**.

5 Enter the service profile name and select the vendor template. Optionally, add description and tags.

6 Click **Save**.

   The vendor template ID used to create the service profile is passed on to the partner console. Partners store the vendor template ID to track usage of which guest VMs are protected by these vendor template.

Results

After creating service profile, an NSX admin creates rules to associate a service profile to a group of VMs before publishing the policy rule.

What to do next

Apply endpoint protection policy on guest VM groups that need to be protected from malware.

Consume Guest Introspection Policy

Policy can be enforced on VM groups by creating rules that associate service profiles with VM groups. Protection begins immediately after rules are applied to a VM group.

The endpoint protection policy is a protection service offered by partners to protect guest VMs from malware by implementing service profiles on guest VMs. With a rule applied to a VM group, all guest VMs within that group are protected by that service profile. When a file access event on a guest VM occurs, the GI thin agent (running on each guest VM) collects context of the file (file attributes, file handle, and other context details) and notifies the event to SVM. If the SVM wants to scan the file content, it request for details using the EPSec API library. Upon a clean verdict from SVM, the GI thin agent allows the user to access the file. In case SVM reports the file as infected, the GI thin agent denies user access to the file.

To implement the endpoint protection policy, you begin by creating a domain that caters to a specific kind of workload. Then, you define EPP rules by associating a VM group with a service profile, which defines the service, protection level to protect VMs. Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1. In NSX-T Data Center 2.4.1 it is not necessary to create any domain.

To execute an security service on a VM group, you need to:

Procedure

1 Define a domain, an environment that hosts VM groups and rules.

   Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1.

2 Define membership criteria to form VM group.

3 Define rules for VM groups.

4 Publish the rule.

Add and Publish Endpoint Protection Rules

Publishing policy rules to VM groups means associating VM groups that need to be protected with a specific service profile.

Procedure

1. In the policy section, select the policy section.
2. Click **Add** -> **Add Rule**.
3. In the Name column, enter a name for the rule.
4. In the Group column, select the VM group.
5. In the Service Profiles column, select the service profile that provides the desired protection level to the guest VMs in the group.
6. Click **Publish**.

Results

Endpoint protection policies protect VM groups.

What to do next

You might want to change the sequence of rules depending on the type of protection required for different VM groups. See How Guest Introspection Runs Endpoint Protection Policy

Monitor Endpoint Protection Status

Monitor the configuration status of protected and unprotected VMs, issues with Host agent and service VMs, and VMs configured with the file introspection driver that was installed as part of the VMtools installation.

You can view:

- View Service Deployment Status.
- View Configuration Status of Endpoint Protection.
- View Capacity Status Set for Endpoint Protection.

View Service Deployment Status

View service deployment details on the Monitoring Dashboard.

View the system-wide status of EPP policy.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to **Home** > **Monitoring - Dashboards**.
3. From the drop-down menu, click **Monitoring - System**.

4 To view the deployment status across clusters in the system, navigate to the Endpoint Protection widget, click the doughnut chart to view successful or unsuccessful deployments.

The Service Deployments page displays the deployment details.

View Capacity Status Set for Endpoint Protection

View capacity status of the endpoint protection service.

View the capacity status of EPP policy.

Procedure

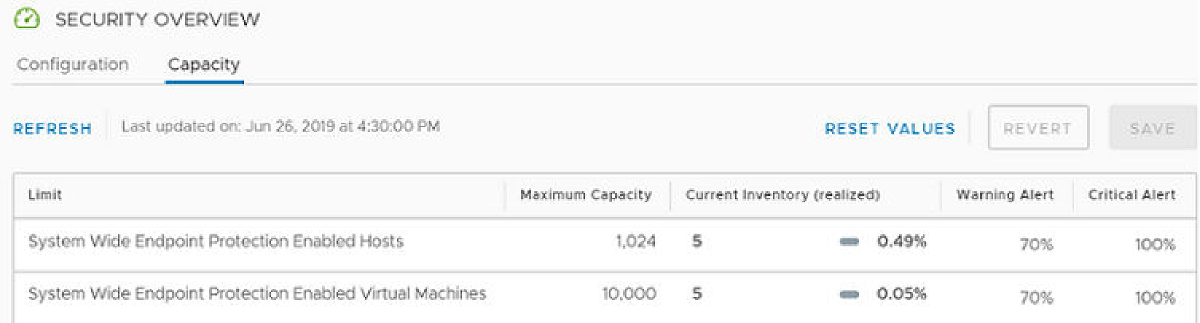
1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Navigate to **Home** > **Monitoring - Dashboards**.

3 From the drop-down menu, click **Monitoring - Networking and Security**.

4 To view status of EPP on clusters, click the Security widget.

5 In the Security Overview page, click **Capacity** and view capacity status of these parameters.



Last updated on: Jun 26, 2019 at 4:30:00 PM

Limit	Maximum Capacity	Current Inventory (realized)	Warning Alert	Critical Alert
System Wide Endpoint Protection Enabled Hosts	1,024	5 0.49%	70%	100%
System Wide Endpoint Protection Enabled Virtual Machines	10,000	5 0.05%	70%	100%

a **System Wide Endpoint Protection Enabled Hosts**: If the number of host numbers protected reaches the threshold limit, NSX Manager notifies a warning alert or critical alert when corresponding threshold limits are reached.

b **System Wide Endpoint Protection Enabled Virtual Machines**: If the number of virtual machine numbers protected reaches the threshold limit, NSX Manager notifies a warning alert or critical alert when corresponding threshold limits are reached.

Note You can set threshold limits for these parameters, view status and receive alerts when these parameters reach the set threshold limit.

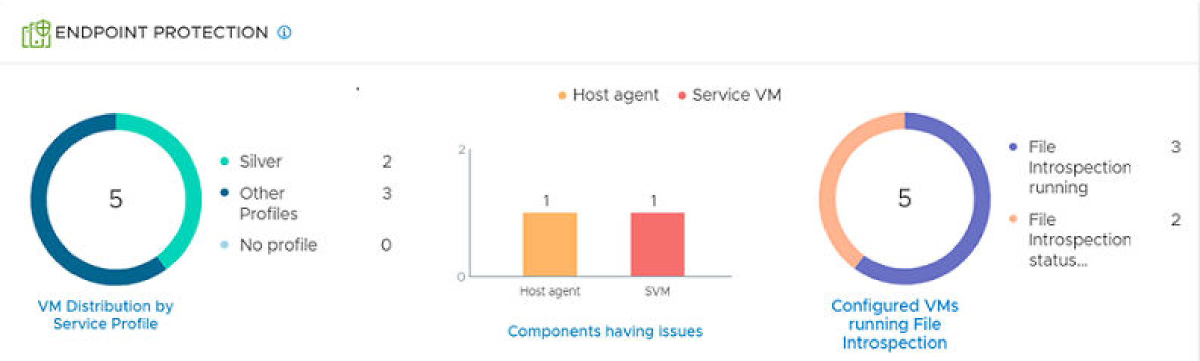
View Configuration Status of Endpoint Protection

View configuration status of the endpoint protection service.

View the system-wide status of EPP policy.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to **Home** > **Security** > **Security Overview**.
3. To view status of EPP on clusters, click the Security widget.
4. In the Security Overview page, click **Configuration**.


ENDPOINT PROTECTION
5
Silver 2
Other Profiles 3
No profile 0
VM Distribution by Service Profile
Host agent
Service VM
2
1
1
0
Host agent
SVM
Components having issues
5
File Introspection running 3
File Introspection status... 2
Configured VMs running File Introspection

5. In the Endpoint Protection section, view:

   a. VM Distribution by Service Profile widget displays:

      1. Number of VMs protected by top profile. Top profile represents a profile that protects the maximum number of VMs on a cluster.
      2. VMs protected by remaining service profiles categorized under Other Profiles.
      3. VMs not protected categorized under No Profile.

      The Endpoint Protection Rules page displays VMs protected by Endpoint Protection policies.

   b. Components having issues widget displays:

      1. Host: Issues related to the context multiplexer.
      2. SVM: Issues related to service VMs. For example, the SVM state is down, SVM connection with guest VM is down.

      The Status column on the Deployment page displays health issues.

   c. Configure VMs running File Introspection widget displays:

      1. VMs protected by File Introspection driver.
      2. VMs where the File Introspection driver status is unknown.

   ESXi Agency Manager (EAM) attempts to resolve a few issues related to hosts, SVMs, and configuration errors. See Ensure Partner Services are Functional on Each Host.

Add Domain and VM Groups

Create a domain that represents an environment to which the policies and VM security groups belong.

Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1. In NSX-T Data Center 2.4.1 it is not necessary to create any domain.

Procedure

1. Select **Security > Endpoint Protection > Rules** .
2. Click **Add Policy**.
3. In the Name column, enter a name for the policy.
4. In the Domain field, click **default** to select a domain or create a new one.
5. In the Select Domain window, at the bottom of the window click **CREATE NEW DOMAIN**.
6. In the Name column, enter a name for the domain.
7. Click **Save**.
8. Click **Yes** to configure the groups in this domain.
9. Click **Add Group**.
10. In the Add Groups window, enter the name for the group.
11. In the Compute Members column, select Members.
12. In the Select Members window, set the membership criteria for VMs to join the group or manually select VMs to be part of a group.
13. Click **Add Criteria**. Membership criteria can be defined by either a tag, OS Name, or Computer Name.
14. After you add the desired criteria for VMs to join the group, click **Save** and click **Close**.
15. Click **Save**.

What to do next

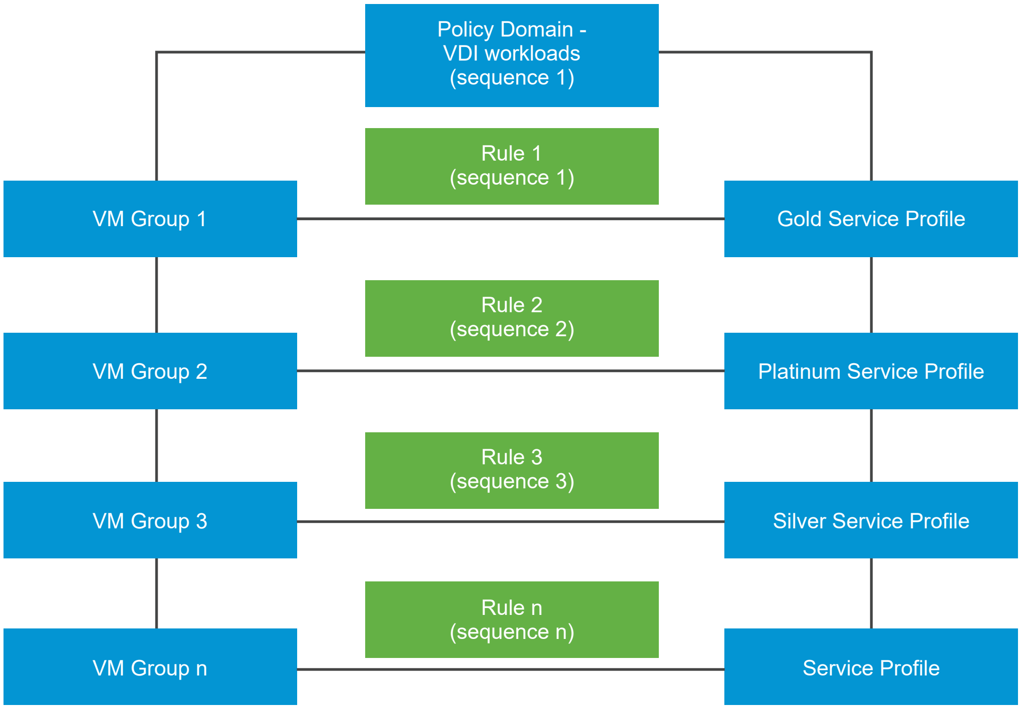
Create and publish rules. See Add and Publish Endpoint Protection Rules.

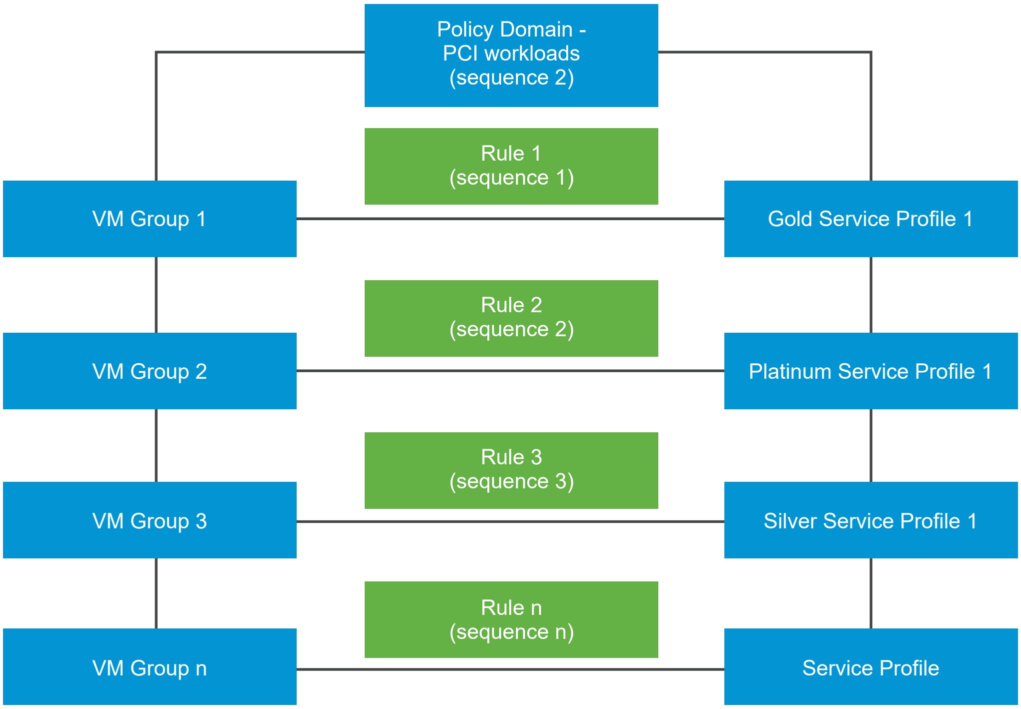
How Guest Introspection Runs Endpoint Protection Policy

Endpoint protection policies are enforced in a specific order. When you design policies, consider the sequence number associated to rules and the domains that host the rules.

Note The domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1. In NSX-T Data Center 2.4.1 it is not necessary to create any domain.

Scenario: Out of the many workloads that run in your organization, for the purposes of illustration we consider two kinds of workloads - VMs running Virtual Desktop Infrastructure (VDI), and VMs running Payments Cards Industry Data Security Standards (PCI-DSS) workloads. A section of employees in the organization requires remote desktop access, which makes up the virtual desktop infrastructure (VDI) workload. These VDI workloads might require a Gold protection policy level based on the compliance rules set up by the organization. Whereas a PCI-DSS workload needs the highest level of protection, Platinum level protection.





As there are two workload types, create two policies one each for VDI workloads and server workloads. Within each policy or section, define a domain to reflect the workload type and within that section define rules for that workload. Publish the rules to start GI services on guest VMs. GI internally uses the two sequence numbers: Policy sequence number and rule sequence number to determine the complete sequence of rules to run. Each rule serves two purposes: determines which VMs to protect and the protection policy that must be applied to protect the VMs.

To change the sequence order, drag a rule in the NSX-T Policy Manager UI to change its sequence order. Alternatively, you can explicitly assign sequence number for rules using API.

Alternatively make an NSX-T Data Center API call to manually define a rule by associating a service profile with a VM group and declare the sequence number of the rules. The API and parameter details are detailed in the NSX-T Data Center *API guide*. Make Service configuration APIs calls to apply profiles to entities such as VM groups and so on.

Table 10-10. NSX-T Data Center APIs used to define rule that apply service profile to VM groups

API	Details
Get all service configuration details.	`GET /api/v1/service-configs` The service configuration API returns details of the service profile applied to a VM group, the VM group protected, and the sequence or precedence number that decides priority of the rule.
Create a service configuration.	`POST /api/v1/service-configs` The service configuration API takes input parameters of a service profile, VM group to be protected, and sequence or precedence number that must be applied to the rule.
Delete a service configuration.	`DELETE /api/v1/service-configs/<config-set-id>` The service configuration API deletes the configuration applied to the VM group.
Get details of a specific configuration.	`GET /api/v1/service-configs/<config-set-id>` Get details of a specific configuration
Update a service configuration.	`PUT /api/v1/service-configs/<config-set-id>` Update a service configuration.
Get effective profiles.	`GET /api/v1/service-configs/effective-profiles?resource_id=<resource-id>&resource_type=<resource-type>` The service configuration API returns only that profile which is applied to a particular VM group.

Efficiently manage rules by following these recommendations:

- Set a higher sequence number for a policy for which rules must be ran first. From the UI, you can drag policies to change their priority.
- Similarly, set a higher sequence number for rules within each policy.
- Depending on how many rules you need, you can position rules apart in multiples of 2, 3, 4, or even 10. So, two consecutive rules that are 10 positions apart give you more flexibility to resequence rules without having to change the sequence order of all the rules. For example, if you do not plan to define many rules, you can select to position rules 10 positions apart. So, rule 1 gets a sequence number of 1, rule 2 gets a sequence number of 10, rule 3 gets a sequence number of 20, and so on. This recommendation provides flexibility to efficiently manage rules so that you do not need to resequence all the rules.

Internally, guest introspection sequences these policy rules in the following way.

```
Policy 1 ↔ Sequence Number 1 (1000)

 – Rule 1 : Group 1↔ Service Profile ↔ Sequence Number 1 (1001)

 – Rule 2 : Group 1↔ Service Profile ↔ Sequence Number 10 (1010)

 – Rule 3 : Group 1↔ Service Profile ↔ Sequence Number 20 (1020)

 – Rule 4 : Group 1↔ Service Profile ↔ Sequence Number 30 (1030)



Policy 2  ↔ Sequence Number 2 (2000)

 – Rule 1 : Group 1↔ Service Profile ↔ Sequence Number 1 (2001)

 – Rule 2 : Group 1↔ Service Profile ↔ Sequence Number 10 (2010)

 – Rule 3 : Group 1↔ Service Profile ↔ Sequence Number 20 (2020)

 – Rule 4 : Group 1↔ Service Profile ↔ Sequence Number 30 (2030)
```

Based on the above sequence numbers, GI runs rules of Policy 1 before it runs rules of Policy 2.

But there are situations when the intended rules are not applied to a VM group or a VM. These conflicts need to be resolved to apply the desired policy protection levels.

Ensure Partner Services are Functional on Each Host

Without partner service VM functional, guest VMs are not protected against malware.

On each host, the following services or process must be up and running:

- ESXi Agency Manager (EAM) service must be up and running. To verify, the following URL must be accessible.

  ```
  https://<vCenter_Server_IP_Address>/eam/mob
  ```

  Run command to verify whether ESXi Agency Manager is online.

  ```
  root> service-control --status vmware-eam
  ```

- Port groups related to SVMs that are automatically created by NSX-T Data Center are not deleted because these are required to ensure SVM continues to protect guest VMs.

  ```
  https://<vCenter_Server_IP_Address>/ui
  ```

  In vCenter Server, go to the virtual machine, click **Networks** tab, and check whether **vmservice-vshield-pg** is listed.

- Context Multiplexer (MUX) service is up and running. Check `nsx-context-mux` VIB is UP and running on the host.

- Management interface: The SVM interface on which NSX-T Data Center communicates with the partner service console.
- Control interface: The SVM interface enabling communication between MUX and SVM. Port group connecting MUX with SVM is created. This interface and port group is required for the partner service to be functional.

ESXi Agency Manager Issues

The table lists the ESXi Agency Manager issues that can be resolved using the Resolve button on the NSX Manager user interface. It notifies NSX Manager with error details.

Issue	Category	Description
Cannot Access Agent OVF	VM Not Deployed	An agent virtual machine is expected to be deployed on a host, but the agent virtual machine cannot be deployed because the ESXi Agent Manager is unable to access the OVF package for the agent. This happens because the web server providing the OVF package is down. The web server is often internal to the solution that created the Agency.
Incompatible Host Version	VM Not Deployed	An agent virtual machine is expected to be deployed on a host, but the agent could not be deployed because it was incompatible with the host.
Insufficient Resources	VM Not Deployed	An agent virtual machine is expected to be deployed on a host, but the agent virtual machine could not be deployed because the host does not have enough free CPU or memory resources.
Insufficient Space	VM Not Deployed	An agent virtual machine is expected to be deployed on a host, but the agent virtual machine could not be deployed because the host's agent datastore did not have enough free space.
No Agent VM Network	VM Not Deployed	An agent virtual machine is expected to be deployed on a host, but the agent cannot be deployed because the agent network has not been configured on the host.

Ovf Invalid Format	VM Not Deployed	An Agent virtual machine is expected to be provisioned on a host, but it failed to do so because the provisioning of the OVF package failed. The provisioning is unlikely to succeed until the solution that provides the OVF package has been upgraded or patched to provide a valid OVF package for the agent virtual machine.
Missing Agent IP Pool	VM Powered Off	An agent virtual machine is expected to be powered on, but the agent virtual machine is powered off because there there are no IP addresses defined on the agent's virtual machine network.
No Agent VM Datastore	VM Powered Off	An agent virtual machine is expected to be deployed on a host, but the agent cannot be deployed because the agent datastore has not been configured on the host.
No Custom Agent VM Network	No Agent VM Network	An agent virtual machine is expected to be deployed on a host, but the agent cannot be deployed because the agent network has not been configured on the host. The host needs to be added to one of the networks listed in custom agent VM network.
No Custom Agent VM Datastore	No Agent VM Datastore	An agent virtual machine is expected to be deployed on a host, but the agent cannot be deployed because the agent datastore has not been configured on the host. The host needs to be added to one of the datastores listed in custom agent VM datastore.
Orphaned Agency	Agency Issue	The solution that created the agency is no longer registered with the vCenter Server.
Orphaned DvFilter Switch	Host Issue	A dvFilter switch exists on a host but no agents on the host depend on dvFilter. This typically happens if a host is disconnected when an agency configuration changed.
Unknown Agent VM	Host Issue	An agent virtual machine has been found in thevCenter Server inventory that does not belong to any agency in this vSphere ESX Agent Manager server instance.
Ovf Invalid Property	VM Issue	An agent virtual machine needs to be powered on, but an OVF property is either missing or has an invalid value.

VM Corrupted	VM Issue	An agent virtual machine is corrupt.
VM Orphaned	VM Issue	An agent virtual machine exists on a host, but the host is no longer part of scope for the agency. This typically happens if a host is disconnected when the agency configuration is changed.
VM Deployed	VM Issue	An agent virtual machine is expected to be removed from a host, but the agent virtual machine has not been removed. The specific reason why vSphere ESX Agent Manager was unable to remove the agent virtual machine, such as the host is in maintenance mode, powered off or in standby mode.
VM Powered Off	VM Issue	An agent virtual machine is expected to be powered on, but the agent virtual machine is powered off.
VM Powered On	VM Issue	An agent virtual machine is expected to be powered off, but the agent virtual machine is powered off.
VM Suspended	VM Issue	An agent virtual machine is expected to be powered on, but the agent virtual machine is suspended.
VM Wrong Folder	VM Issue	An agent virtual machine is expected to be located in a designated agent virtual machine folder, but is found in a different folder.
VM Wrong Resource Pool	VM Issue	An agent virtual machine is expected to be located in a designated agent virtual machine resource pool, but is found in a different resource pool.
VM Not Deployed	Agent Issue	An agent virtual machine is expected to be deployed on a host, but the agent virtual machine has not been deployed. Specific reasons why ESXi Agent Manager was unable to deploy the agent, such as being unable to access the OVF package for the agent or a missing host configuration. This issue can also happen if the agent virtual machine is explicitly deleted from the host.

Next, configure endpoint protection for VM groups. See Configure Endpoint Protection.

Endpoint Policy Conflict Resolution

Consider a scenario where two policy domains exist, each consisting of multiple rules. As an admin you are not always certain of which VMs can end up getting membership of a group because VMs get associated to a group based on dynamic membership criteria, such as OS Name, Computer Name, User, Tagging.

Note The domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1. In NSX-T Data Center 2.4.1 it is not necessary to create any domain.

Conflicts arise in the following scenarios:

- A VM is part of two groups, where each group is protected by a different profile.
- A partner service VM is associated with more than one service profile.
- An unexpected rule ran on a guest VM, or when a rule does not run on a VM group.
- Sequence number is not assigned to policy rules or domains.

Table 10-12. Resolve policy conflicts

Scenario	Expected Endpoint Protection Flow	Resolution
When a VM gets membership to multiple groups. And each group is protected by a different type of service profile. Expected protection was not applied to the VM.	A VM group created with a membership criteria means that VMs are added to the group dynamically. In such a case, the same VM can be part of multiple groups. There is no way to pre-determine which group that VM is going to be part of because the membership criteria dynamically populates VM into the group. Consider VM 1 is part of Group 1 and Group 2. ■ Rule 1: Group 1 (by OS name) is applied Gold (Service Profile) with Sequence Number 1 ■ Rule 2: Group 2 (by tag) is applied Platinum with Sequence Number 10 Endpoint protection policy runs the Gold service profile on VM 1 but does not run Platinum service profile on VM1.	Change the Sequence Number of Rule 2 such that it runs before Rule 1. ■ On the NSX-T Policy Manager UI, drag the Rule 2 before Rule 1 on the rule list. ■ Using NSX-T Policy Manager API, manually add a higher sequence number for Rule 2.
When a rule associates the same service profile to protect two VM groups. Endpoint protection does not run the rule on the second VM group.	Endpoint protection only runs the first service profile on the VM because the same service profile cannot be applied again to any other rule across policies or domain. Consider VM 1 is part of Group 1 and Group 2. Rule 1: Group 1 (by OS name) is applied Gold (service profile) Rule 2: Group 2 (by tag) is applied Gold (service profile)	■ Add Group 2 to Rule 1. (Rule 1: Group 1, Group 2 is applied Profile 1)

Quarantine VMs

After rules are applied to VM groups, based on the protection level and tag set by partners, there might be VMs that are identified as infected that need to be quarantined.

Partners use the API with tag `virus_found=true` to tag VMs that are infected. Affected VMs are attached with the `virus_found=true` tag.

As an administrator, you can create a pre-defined quarantine group based on tag with `virus_found=true` value, such that the group gets populated with infected VMs as and when they are tagged. As an admin, you might choose to set specific firewall rules for the quarantine group. You can set firewall rules for the quarantine group. For example, you might choose to block all traffic incoming and outgoing from the quarantine group.

Verify Health Status of Service Instances

Health status of a service instance depends on many factors: status of the partner solution, connectivity between Guest Introspection Agent (Context Multiplexer) and Context Engine (Ops Agent), and availability of Guest Introspection Agent information, SVM protocol information with NSX Manager.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Service Deployments > Service Instances**.
3. In the Health Status column, click ⓘ to know the health of the service instance.

   Table 10-13. Health Status of Third-Party Service Instance

Parameter	Description
Health Status received at	The latest timestamp when NSX Manager received the health status details of the service instance.
Solution Status	Status of partner solution running on an SVM. Status UP indicates that the partner solution is correctly running.
Connectivity between NSX-T Data Center Guest Introspection Agent and NSX-T Data Center Ops Agent	Status is UP when NSX-T Data Center Guest Introspection agent (context multiplexer) is connected with the Ops agent (includes the context engine). The context multiplexer forwards health information of SVMs to the context engine. They also share SVM-VM configuration between each other to know which guest VMs are protected by the SVM.
Service VM Protocol Version	Transport protocol version used internally for troubleshooting issues.
NSX-T Data Center Guest Introspection Agent Information	Represents protocol version compatibility between NSX-T Data Center Guest Introspection agent and SVM.

4. If the Health Status is `Up` (status displayed in green) and the partner console displays all guest VMs as protected, the health status of the service instance is `Up`.

5 If the Health Status is `Up` (status displayed in green) but the partner console displays guest VMs in unprotected state, perform the following step:

   a Contact VMware support to resolve the issue. The health status of the service instance might be down not correctly reflected by the NSX Manager user interface.

6 If the Health Status is `Down` (status displayed in red), then one or more factors that determine the service instance health are down.

Table 10-14. Troubleshoot Health Status

Health Status Attribute	Resolution
Solution Status is `Down` or `Not available`.	1 Verify that service deployment status is `Up` (green). If you encounter errors, see Ensure Partner Services are Functional on Each Host. 2 Ensure that at least one guest VM in the affected host is protected with an endpoint protection policy. 3 From the partner console, verify whether the solution service is running on the SVM on the host. See the Partner documentation for more details. 4 If none of the above steps resolve the issue, contact VMware support.
Connectivity between NSX-T Data Center Guest Introspection Agent and NSX-T Data Center Ops Agent is `Down`.	1 Verify that service deployment status is `Up` (green). If you encounter errors, see Ensure Partner Services are Functional on Each Host. 2 Ensure that at least one guest VM in the affected host is protected with an endpoint protection policy. 3 From the partner console, verify whether the solution service is running on the SVM on the host. See the Partner documentation for more details. 4 If none of the above steps resolve the issue, contact VMware support.
Service VM Protocol Version is `Unavailable`.	1 Verify that service deployment status is `Up` (green). If you encounter errors, see Ensure Partner Services are Functional on Each Host. 2 Ensure that at least one guest VM in the affected host is protected with an endpoint protection policy. 3 From the partner console, verify whether the solution service is running on the SVM on the host. See the Partner documentation for more details. 4 If none of the above steps resolve the issue, contact VMware support.
NSX-T Data Center Guest Introspection Agent Information is `Unavailable`.	Contact VMware support.

Delete Partner Services

To delete partner services, make an API call . Before you make the API call to delete partner services or SVMs deployed on a host, you need to do the following from the NSX Manager user interface.

To delete partner services:

Procedure

1. Remove EPP rules applied to VM groups running on the host.
2. Remove service profile protection applied to VM groups.
3. To remove solution binding SVMs with partner service manager, make the following API call.

   ```
   /DEL https://<NSX_Manager_IPaddress>/api/v1/serviceinsertion/services/{{service_id}}/solution-
   configs/<solution-config-id>
   ```

4. To delete the service deployment, make the following API call.

   ```
   /DEL https://<NSX_Manager_IPaddress>/api/v1/serviceinsertion/services/<service-id>/service-
   deployments/<service-deployment-id>
   ```

   Refer to the *NSX-T Data Center API guide* for more information on API parameters.

Inventory

11

You can configure services, groups, domains, and context profiles for the NSX-T Data Center inventory.

Note that the domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1.

This chapter includes the following topics:

Add a Domain

A domain is a logical collection of workloads and objects which serve a common business goal. Creating domains makes it easier to manage objects in your environment.

Note The domain object is an experimental feature in NSX-T Data Center 2.4 but is not available in NSX-T Data Center 2.4.1.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Inventory > Domains**.

3 Click **Add Domain**.

4 Enter a name and optionally a description.

5 Click **Save** and continue configuring the groups.

6 Click **Add Group**.

7 Enter a name.

8 Click **Set Members**.

   You can select members using one or more of the following methods:

   - Specifying member criteria
   - Selecting members
   - Entering IP or MAC addresses
   - Selecting AD groups

9 Click **Add Criteria** to select members by specifying membership criteria.

10 Click the **Members** tab to select objects.

11 Click the **IP/MAC Addresses** tab to enter IP or MAC addresses.

12 Click the **AD Groups** tab to select AD groups.

13 Click **Save**.

Add a Service

You can configure a Service, and specify parameters for matching network traffic such as a port and protocol pairing.

You can also use a Service to allow or block certain types of traffic in firewall rules. You cannot change the type after you create an Service. Some Services are predefined and cannot be modified or deleted.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Inventory > Services**.

3 Click **Add New Service**.

4 Enter a name.

5 Click **Set Service Entries**. Select a predefined Service from the list, or click **Add New Service Entry**.

6 For a new service, select a type of service, and specify additional properties.

   The available types are **IP**, **IGMP**, **ICMPv4**, **ICMPv6**,**ALG**, **TCP**, **UDP** and **Ether**.

7 Click **Save**.

8 (Optional) Enter a scope.

9 Click **Save**.

Add a Group

Groups include different objects that are added both statically and dynamically and can be used as the source and destination field of a firewall rule.

Groups can be configured to contain a combination of virtual machines, IP sets, MAC sets, logical ports, logical switches, AD user groups, and other nested groups. Dynamic inclusion of groups can be based on tag, machine name, OS name, or computer name.

A single ID based group can be used within a firewall rule. If IP and ID based groups are needed at the source, create two separate firewall rules.

Groups consisting of only IP addresses, MAC Addresses, or Active Directory groups cannot be used in the **Applied-to** text box.

Note When a host is added to or removed from a vCenter Server, the external ID of the VMs on the host changes. If a VM is a static member of a group and the VM's external ID changes, the NSX Manager UI will no longer show the VM as a member of the group. However, the API that lists the groups will still show that the group contains the VM with its original external ID. If you add a VM as a static member of a group and the VM's external ID changes, you need to add the VM again using its new external ID. You can also use dynamic membership criteria to avoid this issue.

Procedure

1. Select **Inventory > Groups** from the navigation panel.
2. Click **ADD GROUP**.
3. Enter a group name.
4. (Required) Choose a domain from the drop-down menu, or use the default domain. A domain is a logical construct representing a security zone and all rules and groups. The default domain represents the entire NSX environment.

   Note that the domain object is an experimental feature in NSX-T Data Center 2.4, and is not available in NSX-T Data Center 2.4.1. In NSX-T Data Center 2.4.1 it is not necessary to create any domain.
5. (Optional) Click **Set Members**.

   For each membership criterion, you can specify up to five rules, which are combined with the logical AND operator. The available member criterion can apply to the following:

   - **Logical Port** - can specify a tag and optional scope.
   - **Logical Switch** - can specify a tag and optional scope.
   - **Virtual Machine** - can specify a name, tag, computer OS name, or computer name that equals, contains, starts with, ends with, or does not equal a particular string.
   - **Transport Node** - can specify a node type that equals an edge node or a host node.

6 (Optional) Click **Members** to select members.

   The available member types are:

   - **Group**
   - **Segment**
   - **Segment Port**
   - **Virtual Network Interface**
   - **Virtual Machine**

7 Click **IP/MAC Addresses** to add IP and MAC addresses as group members.

8 Click **AD Groups** to add Active Directory Groups. Groups with Active Directory members can be used to in the source or destination field of a distributed firewall rule for Identity Firewall, for and must be the only members in the group. For example, there cannot be an group with both ADGroup and IPSet together as members.

9 Click **Apply**

   Groups are listed, with an option to view members and where the group is used.

Add a Context Profile

Context profiles use layer 7 APP ID attributes for use in distributed firewall rules. After a context profile has been defined, it can be used in one or more distributed firewall rules.

There are two attributes for use in context profiles: APP ID and Domain (FQDN) Name. Select APP IDs also have the sub attributes, TLS_Version and CIPHER_SUITE. Both APP ID and domain name can be used in a singe context profile. Multiple APP IDs can be used in the same profile. One APP ID with sub attributes can be used - sub attributes are cleared when multiple APP ID attributes are used in a single profile.

Procedure

1 Select **Inventory > Context Profiles**.

2 Click **Add New Context Profile**.

3 Enter a **Profile Name**.

4 In the Attributes column, click **Set**.

5 Select an attribute, or click **Add Attribute**, and select **App Id** or **Domain (FQDN) Name**.

6 Select one or more attributes.

7 (Optional) If you have selected an attribute with sub attributes such as SSL or CIFS, click **Set** in the Sub Attributes/Values column.

   a Click **Add Sub Attribute** and select a sub attribute category from the drop-down menu.

   b Select one or more sub attributes.

   c Click **Add**. Another sub attribute can be added by clicking **Add Sub Attribute**.

   d Click **Apply**.

8 Click **Add**.

9 (Optional) To add another type of attribute, click **Add Attribute** again.

10 Click **Apply**.

11 (Optional) Enter a description.

12 (Optional) Enter a tag.

13 Click **Save**.

What to do next

Apply this context profile to a layer 7 distributed firewall rule.

Monitoring

12

The topics in this section show you how to configure monitoring using Internet Protocol Flow Information Export (IPFIX) profiles for the firewall and switches, as well as how to configure an IPFIX collector.

This chapter includes the following topics:

Add a Firewall IPFIX Profile

You can configure IPFIX profiles for firewalls.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Tools > Monitoring Profiles > IPFIX**.
3. Click the **Firewall IPFIX Profiles** tab.
4. Click **Add Firewall IPFIX Profile**.
5. Complete the following details.

Setting	Description
Name and Description	Enter a name and optionally a description. **Note** If you want to create a global profile, name the profile `Global`. A global profile cannot be edited or deleted from the UI, but you can do so using NSX-T Data Center APIs.
Active Flow Export Timeout (Minutes)	The length of time after which a flow will time out, even if more packets associated with the flow are received. Default is 1.

Setting	Description
Observation Domain ID	This parameter identifies which observation domain the network flows originate from. The default is 0 and indicates no specific observation domain.
Collector Configuration	Select a collector from the drop-down list.
Priority	This parameter resolves conflicts when multiple profiles apply. The IPFIX exporter will use the profile with the highest priority only. A lower value means a higher priority.

6 Click **Save** and then **Yes** to continue configuring the profile.

7 Click **Applied To** to apply the profile to objects.

Select one or more of the objects.

8 Click **Save**.

Add a Switch IPFIX Profile

You can configure IPFIX profiles for switches, also known as segments.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Tools > Monitoring Profiles > IPFIX**.

3 Click the **Switch IPFIX Profiles** tab.

4 Click **Add Switch IPFIX Profile**.

5 Complete the following details.

Setting	Description
Name and Description	Enter a name and optionally a description. **Note** If you want to create a global profile, name the profile `Global`. A global profile cannot be edited or deleted from the UI, but you can do so using NSX-T Data Center APIs.
Active Timeout (seconds)	The length of time after which a flow will time out, even if more packets associated with the flow are received. Default is 300.
Idle Timeout (seconds)	The length of time after which a flow will time out, if no more packets associated with the flow are received (ESXi only, KVM times out all flows based on active timeout). Default is 300.
Packet Sampling Probability (%)	The percentage of packets that will be sampled (approximately). Increasing this setting may have a performance impact on the hypervisors and collectors. If all hypervisors are sending more IPFIX packets to the collector, the collector may not be able to collect all packets. Setting the probability at the default value of 0.1% will keep the performance impact low.
Collector Configuration	Select a collector from the drop-down list.

Setting	Description
Priority	This parameter resolves conflicts when multiple profiles apply. The IPFIX exporter will use the profile with the highest priority only. A lower value means a higher priority.
Max Flows	The maximum flows cached on a bridge (KVM only, not configurable on ESXi). Default is 16384.
Observation Domain ID	The observation domain ID identifies which observation domain the network flows originate from. Enter 0 to indicate no specific observation domain.

6 Click **Save** and then **Yes** to continue configuring the profile.

7 Click **Applied To** to apply the profile to objects.

Select one or more of the objects.

8 Click **Save**.

Add an IPFIX Collector

You can configure IPFIX collectors for firewalls and switches.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Tools > Monitoring Profiles > IPFIX**.

3 Click the **Collectors** tab.

4 Select **Add New Collector > IPFIX Switch** or **Add New Collector > IPFIX Firewall**.

5 Enter a name.

6 Enter the IP address and port of up to four collectors. Both IPv4 and IPv6 addresses are supported.

7 Click **Save**.

Add a Port Mirroring Profile

You can configure port mirroring profiles for port mirroring sessions.

Note that logical SPAN is supported for overlay segments only and not VLAN segments.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Tools > Port Mirroring**

3 Select **Add Profile > Remote L3 Span** or **Add Profile > Logical Span**.

4 Enter a name and optionally a description.

5 Complete the following profile details.

Session Type	Parameters
Remote L3 SPAN	■ **Direction** - Select **Bidirectional**, **Ingress**, or **Egress**. ■ **Snap Length** - Specify the number of bytes to capture from a packet. ■ **Encapsulation Type** - Select **GRE**, **ERSPAN TWO**, or **ERSPAN THREE**. ■ **GRE Key** - Specify a GRE key if encapsulation type is **GRE**. ■ **ERSPAN ID** - Specify an ERSPAN ID if encapsulation type is **ERSPAN TWO** or **ERSPAN THREE**.
Logical SPAN	■ **Direction** - Select **Bidirectional**, **Ingress**, or **Egress**. ■ **Snap Length** - Specify the number of bytes to capture from a packet.

6 Click **Set** in the **Destination** column to set a destination.

7 Click **Save** and then **Yes** to continue configuring the profile.

8 Click **Sources** and then **Set** to set sources.

For Logical SPAN, the available sources are **Segment Port**, **Group of Virtual Machines**, and **Group of Virtual Network Interfaces**.

For Remote L3 SPAN, the available sources are **Segment**, **Segment Port**, **Group of Virtual Machines**, and **Group of Virtual Network Interfaces**.

9 Click **Save**.

Advanced Monitoring Tools

NSX-T support advanced monitoring methods, including viewing port connections, traceflow, port mirroring, activity monitoring, and so on.

View Port Connection Information

You can use the port connection tool to quickly visualize and troubleshoot the connection between two VMs.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Tools > Port Connection** from the navigation panel.

3 Select a VM from the **Source Virtual Machine** drop-down menu.

4 Select a VM from the **Destination Virtual Machine** drop-down menu.

5 Click **Go**.

A visual map of the port connection topology is displayed. You can click on any of the components in the visual output to reveal more information about that component.

Traceflow

Traceflow allows you to inject a packet into the network and monitor its flow across the network. This flow allows you to monitor your network and identify issues such as bottlenecks or disruptions.

Traceflow allows you to identify the path (or paths) a packet takes to reach its destination or, conversely, where a packet is dropped along the way. Each entity reports the packet handling on input and output, so you can determine whether issues occur when receiving a packet or when forwarding the packet.

Traceflow is not the same as a ping request/response that goes from guest-VM stack to guest-VM stack. Traceflow observes a marked packet as it traverses the overlay network, and each packet is monitored as it crosses the overlay network until it is delivered to the destination guest VM. The injected marked packet is never actually delivered to the destination guest VM, which enables traceflow to be successful even when the guest VM is powered down.

Trace flow can be used on transport nodes and supports both IPV4 and IPv6 protocols including: ICMP, TCP, UDP, DHCP, DNS and ARP/NDP.

Traceflow supports the following traffic types:

- Layer 2 unicast
- Layer 3 unicast
- Layer 2 broadcast
- Layer 2 multicast

You can construct packets with custom header fields and packet sizes. The source or destination for the trace flow can be a logical switch port, logical router uplink port, CSP or DHCP port. The destination endpoint can be any device in the NSX overlay or in the underlay. However, you cannot select a destination that is north of an NSX Edge node. The destination must be on the same subnet, or must be reachable through NSX distributed logical routers.

The traceflow operation is considered Layer 2 if the source and destination are in the same Layer 2 domain. In NSX, this means that they are on the same VXLAN network identifier (VNI or segment ID). This happens, for example, when two VMs are attached to the same logical switch.

If NSX bridging is configured, unknown Layer 2 packets are always sent to the bridge. Typically, the bridge forwards these packets to a VLAN and reports the traceflow packet as delivered. A packet reported as delivered does not necessarily mean that the trace packet was delivered to the specified destination.

For Layer 3 traceflow unicast traffic, the two end points are on different logical switches and have different VNIs, connected to a distributed logical router (DLR).

For multicast traffic, the source is a VM vNIC or a logical port, and the destination is a multicast group address.

Traceflow observations may include observations of broadcasted traceflow packets. The ESXi host broadcasts a traceflow packet if it does not know the destination host's MAC address. For broadcast traffic, the source is a VM vNIC. The Layer 2 destination MAC address for broadcast traffic is FF:FF:FF:FF:FF:FF. To create a valid packet for firewall inspection, the broadcast traceflow operation requires a subnet prefix length. The subnet mask enables NSX to calculate an IP network address for the packet.

Trace the Path of a Packet with Traceflow

Use Traceflow to inspect the path of a packet. Traceflow traces the transport node-level path of a packet. The trace packet traverses the logical switch overlay, but is not visible to interfaces attached to the logical switch. In other words, no packet is actually delivered to the test packet’s intended recipients.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Tools > Traceflow**.
3. Select an IPv4 or IPv6 address type.
4. Select a traffic type.

   For IPv4 addresses the traffic type choices are Unicast, Multicast, and Broadcast. For IPv6 address the traffic type choices are Unicast or Multicast.

5 Specify the source and destination information according to the traffic type.

Traffic Type	Source	Destination
Unicast	Select a VM or a logical port. For a VM: ■ Select a VM from the drop-down list. ■ Select a virtual interface. ■ The IP address and MAC address are displayed if VMtools is installed in the VM, or if the VM is deployed using OpenStack plug-in (address bindings will be used in this case). If the VM has more than one IP address, select one from the drop-down list. ■ If the IP address and MAC address are not displayed, enter the IP address and MAC address in the text boxes. For a logical port: ■ Select an attachment type: **VIF**, **DHCP**, **Edge Uplink**, or **Edge Centralized Service**. ■ Select a port.	Select a VM, a logical port, or IP-MAC. For a VM: ■ Select a VM from the drop-down list. ■ Select a virtual interface. ■ The IP address and MAC address are displayed if VMtools is installed in the VM or if the VM is deployed using OpenStack plug-in (address bindings will be used in this case). If the VM has more than one IP address, select one from the drop-down list. ■ If the IP address and MAC address are not displayed, enter the IP address and MAC address in the text boxes. For a logical port: ■ Select an attachment type: **VIF**, **DHCP**, **Edge Uplink**, or **Edge Centralized Service**. ■ Select a port. For IP-MAC: ■ Select the trace type (layer 2 or layer 3). For layer 2, enter an IP address and a MAC address. For layer 3, enter an IP address.
Multicast	Same as above.	Enter an IP Address. It must be a multicast address from 224.0.0.0 - 239.255.255.255.
Broadcast	Same as above.	Enter a subnet prefix length.

6 (Optional) Click **Advanced** to see the advanced options.

7 (Optional) In the left column, enter the desired values or input for the following fields:

Option	Description
Frame Size	The default is 128.
TTL	The default is 64.
Timeout (ms)	The default is 10000.
Ethertype	The default is 2048.
Payload Type	Select **Base64**, **Hex**, **Plaintext**, **Binary**, or **Decimal**.
Payload Data	Payload formatted based on selected type.

8 (Optional) Select a protocol and provide related information.

Protocol	Step 1
TCP	Specify a source port, a destination port, and TCP flags.
UDP	Specify a source port and a destination port.
ICMP	Specify an ICMP ID and a sequence.

Protocol	Step 1
DHCPv6	Select a DHCP message type: **Solicit**, **Advertise**, **Request**, or **Reply**.
DHCP	Select a DHCP OP code: **Boot Request** or **Boot Reply**.
DNS	Specify an address and select a message type: **Query** or **Response**.

9 Click **Trace**.

Information about the connections, components, and layers is displayed. The output includes a table listing Observation Type (Delivered, Dropped, Received, Forwarded), Transport Node, and Component, and a graphical map of the topology if unicast and logical switch as a destination are selected. You can apply a filter (**All**, **Delivered**, **Dropped**) on the observations that are displayed. If there are dropped observations, the **Dropped** filter is applied by default. Otherwise, the **All** filter is applied. The graphical map shows the backplane and router links. Note that bridging information is not displayed.

Monitor Port Mirroring Sessions

You can monitor port mirroring sessions for troubleshooting and other purposes.

Note that logical SPAN is supported for overlay logical switches only and not VLAN logical switches.

NSX Cloud Note If using NSX Cloud, see How to use NSX-T Data Center Features with the Public Cloud for a list of auto-generated logical entities, supported features, and configurations required for NSX Cloud.

This feature has the following restrictions:

- A source mirror port cannot be in more than one mirror session.
- With KVM, multiple NICs can be attached to the same OVS port. The mirroring happens at the OVS uplink port, meaning that traffic on all the pNICs attached to the OVS port is mirrored.
- For a local SPAN session, the mirror session source and destination ports must be on the same host vSwitch. Therefore, if you vMotion the VM that has the source or destination port to another host, traffic on that port can no longer be mirrored.
- On ESXi, when mirroring is enabled on the uplink, raw production TCP packets are encapsulated using the Geneve protocol by VDL2 into UDP packets. A physical NIC that supports TSO (TCP segmentation offload) can change the packets and mark the packets with the MUST_TSO flag. On a monitor VM with VMXNET3 or E1000 vNICs, the driver treats the packets as regular UDP packets and cannot handle the MUST_TSO flag, and will drop the packets.

If a lot of traffic is mirrored to a monitor VM, there is a potential for the driver's buffer ring to become full and packets to be dropped. To alleviate the problem, you can take one or more of the following actions:

- Increase the rx buffer ring size.

- Assign more CPU resources to the VM.
- Use the Data Plane Development Kit (DPDK) to improve packet processing performance.

Note Make sure that the monitor VM's MTU setting (in the case of KVM, the hypervisor's virtual NIC device's MTU setting also) is large enough to handle the packets. This is especially important for encapsulated packets because encapsulation increases the size of packets. Otherwise, packets might be dropped. This is not an issue with ESXi VMs with VMXNET3 NICs, but is a potential issue with other types of NICs on both ESXi and KVM VMs.

Note In an L3 port mirroring session involving VMs on KVM hosts, you must set the MTU size to be large enough to handle the extra bytes required by encapsulation. The mirror traffic goes through an OVS interface and OVS uplink. You must set the OVS interface's MTU to be at least 100 bytes larger than the size of the original packet (before encapsulation and mirroring). If you see dropped packets, increase the MTU setting for the host's virtual NIC and the OVS interface. Use the following command to set the MTU for an OVS interface:

```
ovs-vsctl -- set interface <ovs_Interface> mtu_request=<MTU>
```

Note When you monitor the logical port of a VM and the uplink port of a host where the VM resides, you will see different behaviors depending on whether the host is ESXi or KVM. For ESXi, the logical-port mirror packets and the uplink mirror packets are tagged with the same VLAN ID and appear the same to the monitor VM. For KVM, the logical-port mirror packets are not tagged with a VLAN ID but the uplink mirror packets are tagged, and they appear different to the monitor VM.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
3. Select **Advanced Networking & Security > Tools > Port Mirroring Session**.
4. Click **Add** and select a session type.

   The available types are **Local SPAN**, **Remote SPAN**, **Remote L3 SPAN**, and **Logical SPAN**.
5. Enter a session name and optionally a description.

6 Provide additional parameters.

Session Type	Parameters
Local SPAN	■ **Transport Node** - Select a transport node. ■ **Direction** - Select **Bidirectional**, **Ingress**, or **Egress**. ■ **Packet Truncation** - Select a packet truncation value.
Remote SPAN	■ **Session Type** - Select **RSPAN Source session** or **RSPAN Destination session**. ■ **Transport Node** - Select a transport node. ■ **Direction** - Select **Bidirectional**, **Ingress**, or **Egress**. ■ **Packet Truncation** - Select a packet truncation value. ■ **Encap. VLAN ID** - Specify an encapsulation VLAN ID. ■ **Preserve Orig. VLAN** - Select whether to preserve the original VLAN ID.
Remote L3 SPAN	■ **Encapsulation** - Select **GRE**, **ERSPAN TWO**, or **ERSPAN THREE**. ■ **GRE Key** - Specify a GRE key if encapsulation is **GRE**. **ERSPAN ID** - Specify an ERSPAN ID if encapsulation is **ERSPAN TWO** or **ERSPAN THREE**. ■ **Direction** - Select **Bidirectional**, **Ingress**, or **Egress**. ■ **Packet Truncation** - Select a packet truncation value.
Logical SPAN	■ **Logical Switch** - Select a logical switch. ■ **Direction** - Select **Bidirectional**, **Ingress**, or **Egress**. ■ **Packet Truncation** - Select a packet truncation value.

7 Click **Next**.

8 Provide source information.

Session Type	Parameters
Local SPAN	■ Select an N-VDS. ■ Select physical interfaces. ■ Enable or disable encapsulated packet. ■ Select virtual machines. ■ Select virtual interfaces.
Remote SPAN	■ Select virtual machines. ■ Select virtual interfaces.
Remote L3 SPAN	■ Select virtual machines. ■ Select virtual interfaces. ■ Select a logical switch.
Logical SPAN	■ Select logical ports.

9 Click **Next**.

10 Provide destination information.

Session Type	Parameters
Local SPAN	■ Select virtual machines. ■ Select virtual interfaces.
Remote SPAN	■ Select an N-VDS. ■ Select physical interfaces.
Remote L3 SPAN	■ Specify an IPv4 address.
Logical SPAN	■ Select logical ports.

11 Click **Save**.

You cannot change the source or destination after saving the port mirroring session.

Configure Filters for a Port Mirroring Session

You can configure filters for port mirroring sessions to limit the amount of data that is mirrored.

This feature has the following capabilities and restrictions:

- Only ESXi and KVM host transport nodes are supported.
- IP address, IP prefix, and IP ranges are supported for source and destination.
- IPSet for source or destination is not supported.
- Mirror statistics on ESXi or KVM are not supported.

You must configure filters using the API. Using the NSX Manager UI is not supported. For more information about the port mirroring API and the `PortMirroringFilter` schema, see the *NSX-T Data Center API Reference*.

Procedure

1 Configure a port mirroring session using the NSX Manager UI or API.

2 Call the `GET /api/v1/mirror-sessions` API to get information about the port mirroring session.

3 Call the `GET /api/v1/mirror-sessions/<mirror-session-id>` API to add one or more filters. For example,

```
PUT https://<nsx-mgr>/api/v1/mirror-sessions/e57e8b2d-3047-4550-b230-dd1ee0e10b49
{
  "resource_type": "PortMirroringSession",
  "id": "e57e8b2d-3047-4550-b230-dd1ee0e10b49",
  "display_name": "port-mirror-session-1",
  "description": "Pnic port mirror session 1",
  "mirror_sources": [
    {
      "resource_type": "LogicalPortMirrorSource",
      "port_ids": [
          "6a361832-43e4-430d-a48a-b84a6cba73c3"
      ]
    }
  ],
  "mirror_destination": {
    "resource_type": "LogicalPortMirrorDestination",
    "port_ids": [
        "3e42e8b2d-3047-4550-b230-dd1ee0e10b34"
    ]
  },
  "port_mirrorring_filters": [
      {
          "filter_action": "MIRROR",
          "src_ips": {
             "ip-addresses": [
                 "192.168.175.250",
                 "2001:bd6::c:2957:160:126"
             ]
          }
          "dst_ips": {
             "ip-addresses": [
                 "192.168.160.126",
                 "2001:bd6::c:2957:175:250"
             ]
          }
      }
  }
  "session_type": "LogicalPortMirrorSession",
  "preserve_original_vlan": false,
  "direction": "BIDIRECTIONAL",
  "_revision": 0
}
```

4 (Optional) You can call the `get mirroring-session <session-number>` CLI command to show the properties of the port mirroring session, including the filters.

Configure IPFIX

IPFIX (Internet Protocol Flow Information Export) is a standard for the format and export of network flow information. You can configure IPFIX for switches and firewalls. For switches, network flow at VIFs (virtual interfaces) and pNICs (physical NICs) is exported. For firewalls, network flow that is managed by the distributed firewall component is exported.

NSX Cloud Note If using NSX Cloud, see How to use NSX-T Data Center Features with the Public Cloud for a list of auto-generated logical entities, supported features, and configurations required for NSX Cloud.

This feature is compliant with the standards specified in RFC 7011 and RFC 7012.

When you enable IPFIX, all configured host transport nodes will send IPFIX messages to the IPFIX collectors using port 4739. In the case of ESXi, NSX-T Data Center automatically opens port 4739. In the case of KVM, if firewall is not enabled, port 4739 is open, but if firewall is enabled, you must ensure that the port is open because NSX-T Data Center does not automatically open the port.

IPFIX on ESXi and KVM sample tunnel packets in different ways. On ESXi the tunnel packet is sampled as two records:

- Outer packet record with some inner packet information
  - SrcAddr, DstAddr, SrcPort, DstPort, and Protocol refer to the outer packet.
  - Contains some enterprise entries to describe the inner packet.
- Inner packet record
  - SrcAddr, DstAddr, SrcPort, DstPort, and Protocol refer to the inner packet.

On KVM the tunnel packet is sampled as one record:

- Inner packet record with some outer tunnel information
  - SrcAddr, DstAddr, SrcPort, DstPort, and Protocol refer to the inner packet.
  - Contains some enterprise entries to describe the outer packet.

Configure Switch IPFIX Collectors

You can configure IPFIX collectors for switches.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Tools > IPFIX**
3. Click the **Switch IPFIX Collectors** tab.
4. Click **Add** to add a collector.
5. Enter a name and optionally a description.
6. Click **Add** and enter the IP address and port of a collector.

   You can add up to 4 collectors.
7. Click **Add**.

Configure Switch IPFIX Profiles

You can configure IPFIX profiles for switches.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Tools > IPFIX**

3 Click the **Switch IPFIX Profiles** tab.

4 Click **Add** to add a profile.

Setting	Description
Name and Description	Enter a name and optionally a description. **Note** If you want to create a global profile, name the profile `Global`. A global profile cannot be edited or deleted from the UI, but you can do so using NSX-T Data Center APIs.
Active Timeout (seconds)	The length of time after which a flow will time out, even if more packets associated with the flow are received. Default is 300.
Idle Timeout (seconds)	The length of time after which a flow will time out, if no more packets associated with the flow are received (ESXi only, KVM times out all flows based on active timeout). Default is 300.
Max Flows	The maximum flows cached on a bridge (KVM only, not configurable on ESXi). Default is 16384.
Sampling Probability (%)	The percentage of packets that will be sampled (approximately). Increasing this setting may have a performance impact on the hypervisors and collectors. If all hypervisors are sending more IPFIX packets to the collector, the collector may not be able to collect all packets. Setting the probability at the default value of 0.1% will keep the performance impact low.
Observation Domain ID	The observation domain ID identifies which observation domain the network flows originate from. Enter 0 to indicate no specific observation domain.
Collector Profile	Select a switch IPFIX collector that you configure in the previous step.
Priority	This parameter resolves conflicts when multiple profiles apply. The IPFIX exporter will use the profile with the highest priority only. A lower value means a higher priority.

5 Click **Applied To** to apply the profile to one or more objects.

The types of object are logical ports, logical switches, and NSGroups. If you select an NSGroup, it must contain one or more logical switches or logical ports. If the NSGroup contains only IP sets or MAC sets, it will be ignored.

6 Click **Save**.

Configure Firewall IPFIX Collectors

You can configure IPFIX collectors for firewalls.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Tools > IPFIX**

3 Click the **Firewall IPFIX Collectors** tab.

4 Click **Add** to add a collector.

5 Enter a name and optionally a description.

6 Click **Add** and enter the IP address and port of a collector.

You can add up to 4 collectors.

7 Click **Add**.

Configure Firewall IPFIX Profiles

You can configure IPFIX profiles for firewalls.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Tools > IPFIX**

3 Click the **Firewall IPFIX Profiles** tab.

4 Click **Add** to add a profile.

Setting	Description
Name and Description	Enter a name and optionally a description. **Note** If you want to create a global profile, name the profile `Global`. A global profile cannot be edited or deleted from the UI, but you can do so using NSX-T Data Center APIs.
Collector Configuration	Select a collector from the drop-down list.
Active Flow Export Timeout (Minutes)	The length of time after which a flow will time out, even if more packets associated with the flow are received. Default is 1.
Priority	This parameter resolves conflicts when multiple profiles apply. The IPFIX exporter will use the profile with the highest priority only. A lower value means a higher priority.
Observation Domain ID	This parameter identifies which observation domain the network flows originate from. The default is 0 and indicates no specific observation domain.

5 Click **Add**.

ESXi IPFIX Templates

An ESXi host transport node supports eight logical switch IPFIX flow templates and two distributed firewall IPFIX flow templates.

The following table lists VMware-specific elements in logical switch IPFIX packets.

Element ID	Parameter Name	Data Type	Unit
880	tenantProtocol	unsigned8	1 byte
881	tenantSourceIPv4	ipv4Address	4 bytes

Element ID	Parameter Name	Data Type	Unit
882	tenantDestIPv4	ipv4Address	4 bytes
883	tenantSourceIPv6	ipv6Address	16 bytes
884	tenantDestIPv6	ipv6Address	16 bytes
886	tenantSourcePort	unsigned16	2 bytes
887	tenantDestPort	unsigned16	2 bytes
888	egressInterfaceAttr	unsigned16	2 bytes
889	vxlanExportRole	unsigned8	1 byte
890	ingressInterfaceAttr	unsigned16	2 bytes
898	virtualObsID	string	variable length

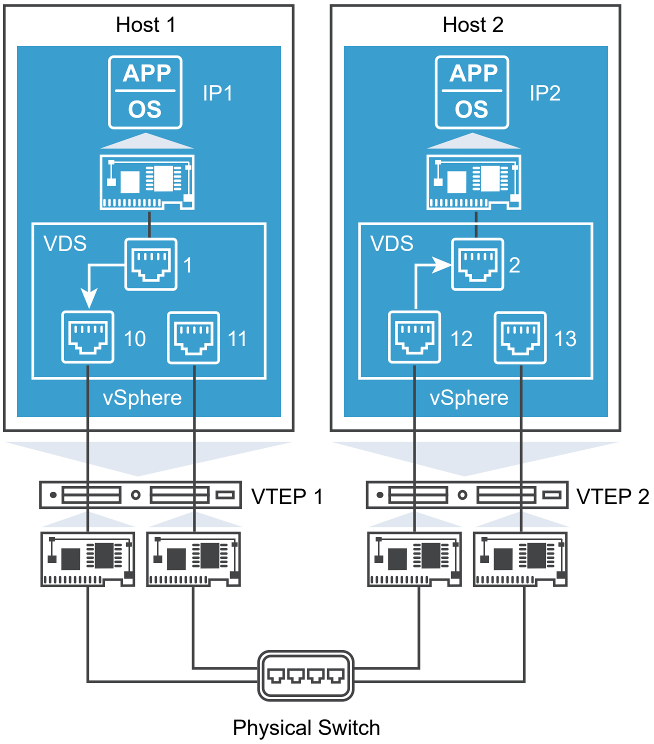
The following table lists VMware-specific elements in distributed firewall IPFIX packets.

Element ID	Parameter Name	Data Type	Unit
950	ruleId	unsigned32	4 bytes
951	vmUuid	string	16 bytes
952	vnicIndex	unsigned32	4 bytes
953	sessionFlags	unsigned8	1 byte
954	flowDirection	unsigned8	1 byte
955	algControlFlowId	unsigned64	8 bytes
956	algType	unsigned8	1 byte
957	algFlowType	unsigned8	1 byte
958	averageLatency	unsigned32	4 bytes
959	retransmissionCount	unsigned32	4 bytes
960	vifUuid	octetArray	16 bytes
961	vifId	string	variable length

ESXi Logical Switch IPFIX Templates

An ESXi host transport node supports eight logical switch IPFIX flow templates.

The following diagram shows the flow of traffic between VMs attached to ESXi hosts monitored by the IPFIX feature:



The IPv4 Encapsulated template will have the following elements:

- standard elements
- SrcAddr: VTEP1
- DstAddr: VTEP2
- tenantSourceIPv4: IP1
- tenantDestIPv4: IP2
- tenantSourcePort: 10000
- tenantDestPort: 80
- tenantProtocol: TCP
- ingressInterfaceAttr: 0x03 (tunnel port)
- egressInterfaceAttr: 0x01
- encapExportRole: 01
- virtualObsID: 89fd5032-2dc9-4fc3-993a-9bb4b616de54 (logical port ID)

IPv4 Template

Template ID: 256

```
IPFIX_TEMPLATE_START(IPFIX_FLOW_TYPE_IPv4)
IPFIX_TEMPLATE_FIELD(sourceIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(destinationIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(octetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(flowStartSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(flowEndSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(sourceTransportPort, 2)
IPFIX_TEMPLATE_FIELD(destinationTransportPort, 2)
IPFIX_TEMPLATE_FIELD(ingressInterface, 4)
IPFIX_TEMPLATE_FIELD(egressInterface, 4)
IPFIX_TEMPLATE_FIELD(encapId, 8)
IPFIX_TEMPLATE_FIELD(protocolIdentifier, 1)
IPFIX_TEMPLATE_FIELD(flowEndReason, 1)
IPFIX_TEMPLATE_FIELD(tcpFlags, 1)
IPFIX_TEMPLATE_FIELD(IPv4TOS, 1)
IPFIX_TEMPLATE_FIELD(maxTTL, 1)
IPFIX_TEMPLATE_FIELD(flowDir, 1)
// Specify the Interface port- Uplink Port, Access port,N.A
IPFIX_VMW_TEMPLATE_FIELD(ingressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(egressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(encapExportRole, 1)
IPFIX_VMW_TEMPLATE_VAR_LEN_FIELD(virtualObsID, virtualObsDataLen)
IPFIX_TEMPLATE_PADDING(paddingOctets, 1)
IPFIX_TEMPLATE_END()
```

IPv4 Encapsulated Template

Template ID: 257

```
IPFIX_TEMPLATE_START(IPFIX_FLOW_TYPE_IPv4_ENCAP)
IPFIX_TEMPLATE_FIELD(sourceIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(destinationIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(octetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(flowStartSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(flowEndSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(sourceTransportPort, 2)
IPFIX_TEMPLATE_FIELD(destinationTransportPort, 2)
IPFIX_TEMPLATE_FIELD(ingressInterface, 4)
IPFIX_TEMPLATE_FIELD(egressInterface, 4)
IPFIX_TEMPLATE_FIELD(protocolIdentifier, 1)
IPFIX_TEMPLATE_FIELD(flowEndReason, 1)
IPFIX_TEMPLATE_FIELD(tcpFlags, 1)
IPFIX_TEMPLATE_FIELD(IPv4TOS, 1)
IPFIX_TEMPLATE_FIELD(maxTTL, 1)
IPFIX_TEMPLATE_FIELD(flowDir, 1)
IPFIX_TEMPLATE_FIELD(encapId, 8)
IPFIX_VMW_TEMPLATE_FIELD(tenantSourceIPv4, 4)
IPFIX_VMW_TEMPLATE_FIELD(tenantDestIPv4, 4)
IPFIX_VMW_TEMPLATE_FIELD(tenantSourcePort, 2)
IPFIX_VMW_TEMPLATE_FIELD(tenantDestPort, 2)
IPFIX_VMW_TEMPLATE_FIELD(tenantProtocol, 1)
// Specify the Interface port - Uplink Port, Access port, N.A
IPFIX_VMW_TEMPLATE_FIELD(ingressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(egressInterfaceAttr, 2)
```

```
// TUNNEL-GW or no.
IPFIX_VMW_TEMPLATE_FIELD(encapExportRole, 1)
IPFIX_VMW_TEMPLATE_VAR_LEN_FIELD(virtualObsID, virtualObsDataLen)
IPFIX_TEMPLATE_END()
```

IPv4 ICMP Template

Template ID: 258

```
IPFIX_TEMPLATE_START(IPFIX_FLOW_TYPE_IPv4_ICMP)
IPFIX_TEMPLATE_FIELD(sourceIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(destinationIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(octetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(flowStartSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(flowEndSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(ingressInterface, 4)
IPFIX_TEMPLATE_FIELD(egressInterface, 4)
IPFIX_TEMPLATE_FIELD(protocolIdentifier, 1)
IPFIX_TEMPLATE_FIELD(flowEndReason, 1)
IPFIX_TEMPLATE_FIELD(IPv4TOS, 1)
IPFIX_TEMPLATE_FIELD(maxTTL, 1)
IPFIX_TEMPLATE_FIELD(flowDir, 1)
IPFIX_TEMPLATE_FIELD(encapId, 8)
// Specify the Interface port - Uplink Port, Access Port, or NA.
IPFIX_VMW_TEMPLATE_FIELD(ingressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(egressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(encapExportRole, 1)
IPFIX_VMW_TEMPLATE_VAR_LEN_FIELD(virtualObsID, virtualObsDataLen)
IPFIX_TEMPLATE_PADDING(paddingOctets, 2)
IPFIX_TEMPLATE_END()
```

IPv4 ICMP Encapsulated Template

Template ID: 259

```
IPFIX_TEMPLATE_START(IPFIX_FLOW_TYPE_IPv4_ICMP_ENCAP)
IPFIX_TEMPLATE_FIELD(sourceIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(destinationIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(octetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(flowStartSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(flowEndSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(sourceTransportPort, 2)
IPFIX_TEMPLATE_FIELD(destinationTransportPort, 2)
IPFIX_TEMPLATE_FIELD(ingressInterface, 4)
IPFIX_TEMPLATE_FIELD(egressInterface, 4)
IPFIX_TEMPLATE_FIELD(protocolIdentifier, 1)
IPFIX_TEMPLATE_FIELD(flowEndReason, 1)
IPFIX_TEMPLATE_FIELD(IPv4TOS, 1)
IPFIX_TEMPLATE_FIELD(maxTTL, 1)
IPFIX_TEMPLATE_FIELD(flowDir, 1)
IPFIX_TEMPLATE_FIELD(encapId, 8)
IPFIX_VMW_TEMPLATE_FIELD(tenantSourceIPv4, 4)
IPFIX_VMW_TEMPLATE_FIELD(tenantDestIPv4, 4)
IPFIX_VMW_TEMPLATE_FIELD(tenantProtocol, 1)
```

```
// Specify the Interface port- Uplink Port, Access port,N.A
IPFIX_VMW_TEMPLATE_FIELD(ingressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(egressInterfaceAttr, 2)
// TUNNEL-GW or no.
IPFIX_VMW_TEMPLATE_FIELD(encapExportRole, 1)
IPFIX_VMW_TEMPLATE_VAR_LEN_FIELD(virtualObsID, virtualObsDataLen)
IPFIX_TEMPLATE_PADDING(paddingOctets, 1)
IPFIX_TEMPLATE_END()
```

IPv6 Template

Template ID: 260

```
IPFIX_TEMPLATE_START(IPFIX_FLOW_TYPE_IPv6)
IPFIX_TEMPLATE_FIELD(sourceIPv6Address, 16)
IPFIX_TEMPLATE_FIELD(destinationIPv6Address, 16)
IPFIX_TEMPLATE_FIELD(octetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(flowStartSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(flowEndSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(sourceTransportPort, 2)
IPFIX_TEMPLATE_FIELD(destinationTransportPort, 2)
IPFIX_TEMPLATE_FIELD(ingressInterface, 4)
IPFIX_TEMPLATE_FIELD(egressInterface, 4)
IPFIX_TEMPLATE_FIELD(encapId, 8)
IPFIX_TEMPLATE_FIELD(protocolIdentifier, 1)
IPFIX_TEMPLATE_FIELD(flowEndReason, 1)
IPFIX_TEMPLATE_FIELD(tcpFlags, 1)
IPFIX_TEMPLATE_FIELD(IPv6TOS,1)
IPFIX_TEMPLATE_FIELD(maxTTL, 1)
IPFIX_TEMPLATE_FIELD(flowDir, 1)
// Specify the Interface port - Uplink Port, Access Port, or NA.
IPFIX_VMW_TEMPLATE_FIELD(ingressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(egressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(encapExportRole, 1)
IPFIX_VMW_TEMPLATE_VAR_LEN_FIELD(virtualObsID, virtualObsDataLen)
IPFIX_TEMPLATE_PADDING(paddingOctets, 1)
IPFIX_TEMPLATE_END()
```

IPv6 Encapsulated Template

Template ID: 261

```
IPFIX_TEMPLATE_START(IPFIX_FLOW_TYPE_IPv6_ENCAP)
IPFIX_TEMPLATE_FIELD(sourceIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(destinationIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(octetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(flowStartSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(flowEndSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(sourceTransportPort, 2)
IPFIX_TEMPLATE_FIELD(destinationTransportPort, 2)
IPFIX_TEMPLATE_FIELD(ingressInterface, 4)
IPFIX_TEMPLATE_FIELD(egressInterface, 4)
IPFIX_TEMPLATE_FIELD(protocolIdentifier, 1)
IPFIX_TEMPLATE_FIELD(flowEndReason, 1)
IPFIX_TEMPLATE_FIELD(tcpFlags, 1)
IPFIX_TEMPLATE_FIELD(IPv6TOS, 1)
IPFIX_TEMPLATE_FIELD(maxTTL, 1)
IPFIX_TEMPLATE_FIELD(flowDir, 1)
//ENCAP specific
IPFIX_TEMPLATE_FIELD(encapId, 8)
IPFIX_VMW_TEMPLATE_FIELD(tenantSourceIPv6, 16)
IPFIX_VMW_TEMPLATE_FIELD(tenantDestIPv6, 16)
IPFIX_VMW_TEMPLATE_FIELD(tenantSourcePort, 2)
IPFIX_VMW_TEMPLATE_FIELD(tenantDestPort, 2)
IPFIX_VMW_TEMPLATE_FIELD(tenantProtocol, 1)
// Specify the Interface port - Uplink Port, Access Port, or NA
IPFIX_VMW_TEMPLATE_FIELD(ingressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(egressInterfaceAttr, 2)
// TUNNEL-GW or no.
IPFIX_VMW_TEMPLATE_FIELD(encapExportRole, 1)
IPFIX_VMW_TEMPLATE_VAR_LEN_FIELD(virtualObsID, virtualObsDataLen)
IPFIX_TEMPLATE_END()
```

IPv6 ICMP Template

Template ID: 262

```
IPFIX_TEMPLATE_START(IPFIX_FLOW_TYPE_IPv6_ICMP)
IPFIX_TEMPLATE_FIELD(sourceIPv6Address, 16)
IPFIX_TEMPLATE_FIELD(destinationIPv6Address, 16)
IPFIX_TEMPLATE_FIELD(octetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(flowStartSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(flowEndSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(ingressInterface, 4)
IPFIX_TEMPLATE_FIELD(egressInterface, 4)
IPFIX_TEMPLATE_FIELD(protocolIdentifier, 1)
IPFIX_TEMPLATE_FIELD(flowEndReason, 1)
IPFIX_TEMPLATE_FIELD(IPv6TOS, 1)
IPFIX_TEMPLATE_FIELD(maxTTL, 1)
IPFIX_TEMPLATE_FIELD(flowDir, 1)
IPFIX_TEMPLATE_FIELD(encapId, 8)
// Specify the Interface port - Uplink Port, Access Port, or NA.
IPFIX_VMW_TEMPLATE_FIELD(ingressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(egressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(encapExportRole, 1)
IPFIX_VMW_TEMPLATE_VAR_LEN_FIELD(virtualObsID, virtualObsDataLen)
IPFIX_TEMPLATE_PADDING(paddingOctets, 2)
IPFIX_TEMPLATE_END()
```

IPv6 ICMP Encapsulated Template

Template ID: 263

```
IPFIX_TEMPLATE_START(IPFIX_FLOW_TYPE_IPv6_ICMP_ENCAP)
IPFIX_TEMPLATE_FIELD(sourceIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(destinationIPv4Address, 4)
IPFIX_TEMPLATE_FIELD(octetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount, 8)
IPFIX_TEMPLATE_FIELD(flowStartSysUpTime, 8)
IPFIX_TEMPLATE_FIELD(flowEndSysUpTime, 8)
IPFIX_VMW_TEMPLATE_FIELD(sourceTransportPort, 2)
IPFIX_VMW_TEMPLATE_FIELD(destinationTransportPort, 2)
IPFIX_TEMPLATE_FIELD(ingressInterface, 4)
IPFIX_TEMPLATE_FIELD(egressInterface, 4)
IPFIX_TEMPLATE_FIELD(protocolIdentifier, 1)
IPFIX_TEMPLATE_FIELD(IPv6TOS, 1)
IPFIX_TEMPLATE_FIELD(maxTTL, 1)
IPFIX_TEMPLATE_FIELD(flowDir, 1)
IPFIX_TEMPLATE_FIELD(flowEndReason, 1)
//ENCAP Specific
IPFIX_TEMPLATE_FIELD(encapId, 8)
IPFIX_VMW_TEMPLATE_FIELD(tenantSourceIPv6, 16)
IPFIX_VMW_TEMPLATE_FIELD(tenantDestIPv6, 16)
IPFIX_VMW_TEMPLATE_FIELD(tenantProtocol, 1)
// Specify the Interface port - Uplink Port, Access Port, or NA
IPFIX_VMW_TEMPLATE_FIELD(ingressInterfaceAttr, 2)
IPFIX_VMW_TEMPLATE_FIELD(egressInterfaceAttr, 2)
// TUNNEL-GW or no.
IPFIX_VMW_TEMPLATE_FIELD(encapExportRole, 1)
IPFIX_VMW_TEMPLATE_VAR_LEN_FIELD(virtualObsID, virtualObsDataLen)
IPFIX_TEMPLATE_PADDING(paddingOctets, 1)
IPFIX_TEMPLATE_END()
```

ESXi Distributed Firewall IPFIX Templates

An ESXi host transport node supports two distributed firewall IPFIX flow templates.

IPv4 Template

Template ID: 288

```
IPFIX_TEMPLATE_FIELD(sourceIPv4Address,4)
IPFIX_TEMPLATE_FIELD(destinationIPv4Address,4)
IPFIX_TEMPLATE_FIELD(sourceTransportPort,2)
IPFIX_TEMPLATE_FIELD(destinationTransportPort,2)
IPFIX_TEMPLATE_FIELD(protocolIdentifier,1)
IPFIX_TEMPLATE_FIELD(icmpTypeIPv4,1)
IPFIX_TEMPLATE_FIELD(icmpCodeIPv4,1)
IPFIX_TEMPLATE_FIELD(flowStartSeconds,4)
IPFIX_TEMPLATE_FIELD(flowEndSeconds,4)
IPFIX_TEMPLATE_FIELD(octetDeltaCount,8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount,8)
IPFIX_TEMPLATE_FIELD(firewallEvent,1)
IPFIX_TEMPLATE_FIELD(direction,1)
IPFIX_TEMPLATE_FIELD(ruleId,4)
IPFIX_TEMPLATE_FIELD(vifUuid,16)
IPFIX_TEMPLATE_FIELD(sessionFlags,1)
IPFIX_TEMPLATE_FIELD(flowDirection,1)
IPFIX_TEMPLATE_FIELD(flowId,8)
IPFIX_TEMPLATE_FIELD(algControlFlowId,8)
IPFIX_TEMPLATE_FIELD(algType,1)
IPFIX_TEMPLATE_FIELD(algFlowType,1)
IPFIX_TEMPLATE_FIELD(averageLatency,4)
IPFIX_TEMPLATE_FIELD(retransmissionCount,4)
```

IPv6 Template

Template ID: 289

```
IPFIX_TEMPLATE_FIELD(sourceIPv6Address,16)
IPFIX_TEMPLATE_FIELD(destinationIPv6Address,16)
IPFIX_TEMPLATE_FIELD(sourceTransportPort,2)
IPFIX_TEMPLATE_FIELD(destinationTransportPort,2)
IPFIX_TEMPLATE_FIELD(protocolIdentifier,1)
IPFIX_TEMPLATE_FIELD(icmpTypeIPv6,1)
IPFIX_TEMPLATE_FIELD(icmpCodeIPv6,1)
IPFIX_TEMPLATE_FIELD(flowStartSeconds,4)
IPFIX_TEMPLATE_FIELD(flowEndSeconds,4)
IPFIX_TEMPLATE_FIELD(octetDeltaCount,8)
IPFIX_TEMPLATE_FIELD(packetDeltaCount,8)
IPFIX_TEMPLATE_FIELD(firewallEvent,1)
IPFIX_TEMPLATE_FIELD(direction,1)
IPFIX_TEMPLATE_FIELD(ruleId,4)
IPFIX_TEMPLATE_FIELD(vifUuid,16)
IPFIX_TEMPLATE_FIELD(sessionFlags,1)
IPFIX_TEMPLATE_FIELD(flowDirection,1)
IPFIX_TEMPLATE_FIELD(flowId,8)
IPFIX_TEMPLATE_FIELD(algControlFlowId,8)
IPFIX_TEMPLATE_FIELD(algType,1)
IPFIX_TEMPLATE_FIELD(algFlowType,1)
IPFIX_TEMPLATE_FIELD(averageLatency,4)
IPFIX_TEMPLATE_FIELD(retransmissionCount,4)
```

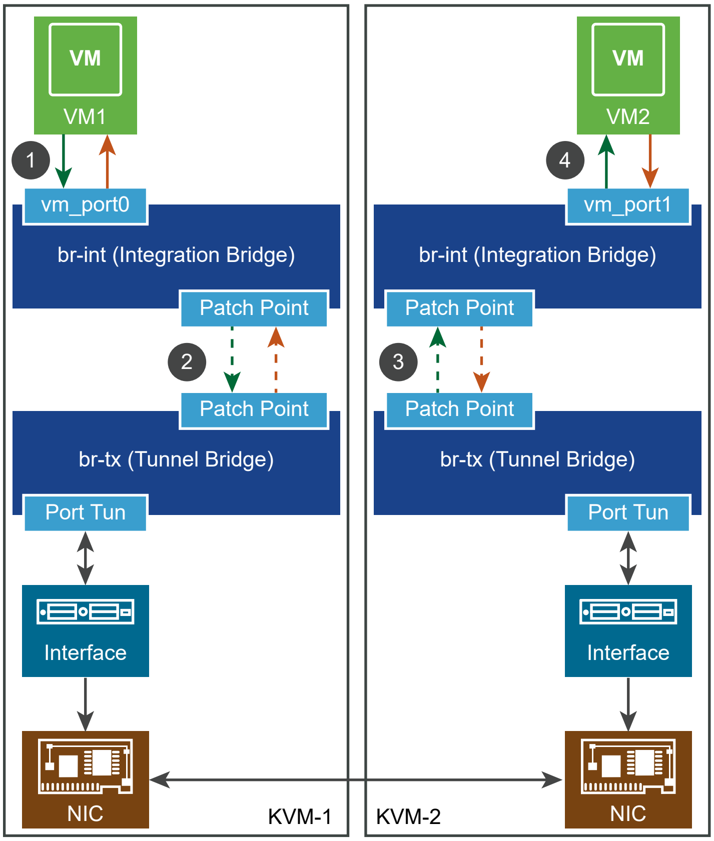
KVM IPFIX Templates

A KVM host transport node supports 88 IPFIX flow templates and one options template.

The following table lists VMware-specific elements in the KVM IPFIX packets.

Element ID	Parameter Name	Data Type	Unit
891	tunnelType	unsigned8	1 byte
892	tunnelKey	bytes	variable length
893	tunnelSourceIPv4Address	unsigned32	4 bytes
894	tunnelDestinationIPv4Address	unsigned32	4 bytes
895	tunnelProtocolIdentifier	unsigned8	1 byte
896	tunnelSourceTransportPort	unsigned16	2 bytes
897	tunnelDestinationTransportPort	unsigned16	2 bytes
898	virtualObsID	string	variable length

The following diagram shows the flow of traffic between VMs attached to KVM hosts monitored by the IPFIX feature:



The KVM IPv4 IPFIX ingress template will have the following elements:

- standard elements
- virtualObsID: 6d876a1c-e0ac-4bcf-85ee-bdd42fa7ba34 (logical port ID)

KVM Ethernet IPFIX Templates

There are four KVM Ethernet IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

Ethernet Ingress

Template ID: 256. Field count: 27.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)

- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)

Ethernet Egress

Template ID: 257. Field count: 31.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)

- IF_DESC (length: variable)
- OUTPUT_SNMP (length: 4)
- Unknown(369) (length: 8)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)

Ethernet Ingress with Tunnel

Template ID: 258. Field count: 34.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)

- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)

Ethernet Egress with Tunnel

Template ID: 259. Field count: 38.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)

- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (length: 4)
- Unknown(369) (length: 8)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)

- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)

KVM IPv4 IPFIX Templates

There are four KVM IPv4 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

IPv4 Ingress

Template ID: 276. Field count: 45.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)

- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv4 Egress

Template ID: 277. Field count: 49.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)

- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)

- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv4 Ingress with Tunnel

Template ID: 278. Field count: 52.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)

- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)

- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv4 Egress with Tunnel

Template ID: 279. Field count: 56.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))

- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM TCP over IPv4 IPFIX Templates

There are four KVM TCP over IPv4 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

TCP over IPv4 Ingress

Template ID: 280. Field count: 53.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)

- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv4 Egress

Template ID: 281. Field count: 57.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)

- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)

- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv4 Ingress with Tunnel

Template ID: 282. Field count: 60.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)

- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)

- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv4 Egress with Tunnel

Template ID: 283. Field count: 64.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)

- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)

- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

KVM UDP over IPv4 IPFIX Templates

There are four KVM UDP over IPv4 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

UDP over IPv4 Ingress

Template ID: 284. Field count: 47.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)

- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)

- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv4 Egress

Template ID: 285. Field count: 51.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)

- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)

- postMCastOctetTotalCount (Length: 8)

UDP over IPv4 Ingress with Tunnel

Template ID: 286. Field count: 54.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))

- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv4 Egress with Tunnel

Template ID: 287. Field count: 58.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)

- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)

- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM SCTP over IPv4 IPFIX Templates

There are four KVM SCTP over IPv4 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

SCTP over IPv4 Ingress

Template ID: 288. Field count: 47.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)

- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)

- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv4 Egress

Template ID: 289. Field count: 51.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)

- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)

- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv4 Ingress with Tunnel

Template ID: 290. Field count: 54.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))

- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv4 Egress with Tunnel

Template ID: 291. Field count: 58.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)

- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)

- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM ICMPv4 IPFIX Templates

There are four KVM ICMPv4 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

ICMPv4 Ingress

Template ID: 292. Field count: 47.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)

- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- ICMP_IPv4_TYPE (Length: 1)
- ICMP_IPv4_CODE (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)

- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv4 Egress

Template ID: 293. Field count: 51.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)

- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- ICMP_IPv4_TYPE (Length: 1)
- ICMP_IPv4_CODE (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)

- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv4 Ingress with Tunnel

Template ID: 294. Field count: 54.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- ICMP_IPv4_TYPE (Length: 1)
- ICMP_IPv4_CODE (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))

- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv4 Egress with Tunnel

Template ID: 295. Field count: 58.

The fields are:

- observationPointId (length: 4)

- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- ICMP_IPv4_TYPE (Length: 1)
- ICMP_IPv4_CODE (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)

- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM IPv6 IPFIX Templates

There are four KVM IPv6 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

IPv6 Ingress

Template ID: 296. Field count: 46.

The fields are:

- observationPointId (length: 4)

- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)

- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv6 Egress

Template ID: 297. Field count: 50.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)

- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)

- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv6 Ingress with Tunnel

Template ID: 298. Field count: 53.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))

- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv6 Egress with Tunnel

Template ID: 299. Field count: 57.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)

- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)

- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM TCP over IPv6 IPFIX Templates

There are four KVM TCP over IPv6 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

TCP over IPv6 Ingress

Template ID: 300. Field count: 54.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)

- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)

- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv6 Egress

Template ID: 301. Field count: 58.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)

- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)

- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv6 Ingress with Tunnel

Template ID: 302. Field count: 61.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)

- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)

- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv6 Egress with Tunnel

Template ID: 303. Field count: 65.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)

- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)

- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

KVM UDP over IPv6 IPFIX Templates

There are four KVM UDP over IPv6 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

UDP over IPv6 Ingress

Template ID: 304. Field count: 48.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)

- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)

- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv6 Egress

Template ID: 305. Field count: 52.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)

- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv6 Ingress with Tunnel

Template ID: 306. Field count: 55.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))

- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv6 Egress with Tunnel

Template ID: 307. Field count: 59.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)

- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)

- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM SCTP over IPv6 IPFIX Templates

There are four KVM SCTP over IPv6 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

SCTP over IPv6 Ingress

Template ID: 308. Field count: 48.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)

- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)

- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv6 Egress

Template ID: 309. Field count: 52.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)

- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)

- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv6 Ingress with Tunnel

Template ID: 310. Field count: 55.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))

- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv6 Egress with Tunnel

Template ID: 311. Field count: 59.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))

- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM ICMPv6 IPFIX Templates

There are four KVM ICMPv6 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

ICMPv6 Ingress

Template ID: 312. Field count: 48.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- ICMP_IPv6_TYPE (Length: 1)
- ICMP_IPv6_CODE (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)

- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv6 Egress

Template ID: 313. Field count: 52.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)

- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- ICMP_IPv6_TYPE (Length: 1)
- ICMP_IPv6_CODE (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)

- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv6 Ingress with Tunnel

Template ID: 314. Field count: 55.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)

- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- ICMP_IPv6_TYPE (Length: 1)
- ICMP_IPv6_CODE (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)

- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv6 Egress with Tunnel

Template ID: 315. Field count: 59.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)

- FLOW_LABEL (Length: 4)
- ICMP_IPv6_TYPE (Length: 1)
- ICMP_IPv6_CODE (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)

- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM Ethernet VLAN IPFIX Templates

There are four KVM Ethernet VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

Ethernet VLAN Ingress

Template ID: 316. Field count: 30.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)

- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)

Ethernet VLAN Egress

Template ID: 317. Field count: 34.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (length: 4)
- Unknown(369) (length: 8)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)

- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)

Ethernet VLAN Ingress with Tunnel

Template ID: 318. Field count: 37.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))

- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)

Ethernet VLAN Egress with Tunnel

Template ID: 319. Field count: 41.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)

- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (length: 4)
- Unknown(369) (length: 8)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)

- Unknown(353) (length: 8)
- flowEndReason (length: 1)

KVM IPv4 VLAN IPFIX Templates

There are four KVM IPv4 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

IPv4 VLAN Ingress

Template ID: 336. Field count: 48.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)

- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv4 VLAN Egress

Template ID: 337. Field count: 52.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)

- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)

- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv4 VLAN Ingress with Tunnel

Template ID: 338. Field count: 55.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)

- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)

- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv4 VLAN Egress with Tunnel

Template ID: 339. Field count: 59.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)

- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)

- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM TCP over IPv4 VLAN IPFIX Templates

There are four KVM TCP over IPv4 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

TCP over IPv4 VLAN Ingress

Template ID: 340. Field count: 56.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)

- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)

- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv4 VLAN Egress

Template ID: 341. Field count: 60.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)

- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)

- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv4 VLAN Ingress with Tunnel

Template ID: 342. Field count: 63.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))

- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv4 VLAN Egress with Tunnel

Template ID: 343. Field count: 67.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)

- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)

- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

KVM UDP over IPv4 VLAN IPFIX Templates

There are four KVM UDP over IPv4 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

UDP over IPv4 VLAN Ingress

Template ID: 344. Field count: 50.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)

- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)

- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv4 VLAN Egress

Template ID: 345. Field count: 54.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)

- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv4 VLAN Ingress with Tunnel

Template ID: 346. Field count: 57.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))

- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv4 VLAN Egress with Tunnel

Template ID: 347. Field count: 61.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)

- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))

- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM SCTP over IPv4 VLAN IPFIX Templates

There are four KVM SCTP over IPv4 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

SCTP over IPv4 VLAN Ingress

Template ID: 348. Field count: 50.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)

- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv4 VLAN Egress

Template ID: 349. Field count: 54.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)

- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)

- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv4 VLAN Ingress with Tunnel

Template ID: 350. Field count: 57.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)

- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)

- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv4 VLAN Egress with Tunnel

Template ID: 351. Field count: 61.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)

- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)

- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM ICMPv4 VLAN IPFIX Templates

There are four KVM ICMPv4 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

ICMPv4 VLAN Ingress

Template ID: 352. Field count: 50.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)

- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- ICMP_IPv4_TYPE (Length: 1)
- ICMP_IPv4_CODE (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv4 VLAN Egress

Template ID: 353. Field count: 54.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- ICMP_IPv4_TYPE (Length: 1)
- ICMP_IPv4_CODE (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)

- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv4 VLAN Ingress with Tunnel

Template ID: 354. Field count: 57.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)

- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- ICMP_IPv4_TYPE (Length: 1)
- ICMP_IPv4_CODE (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)

- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv4 VLAN Egress with Tunnel

Template ID: 355. Field count: 61.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)

- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IP_SRC_ADDR (Length: 4)
- IP_DST_ADDR (Length: 4)
- ICMP_IPv4_TYPE (Length: 1)
- ICMP_IPv4_CODE (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)

- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM IPv6 VLAN IPFIX Templates

There are four KVM IPv6 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

IPv6 VLAN Ingress

Template ID: 356. Field count: 49.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)

- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)

- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv6 VLAN Egress

Template ID: 357. Field count: 53.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)

- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)

- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv6 VLAN Ingress with Tunnel

Template ID: 358. Field count: 56.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))

- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

IPv6 VLAN Egress with Tunnel

Template ID: 359. Field count: 60.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))

- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM TCP over IPv6 VLAN IPFIX Templates

There are four KVM TCP over IPv6 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

TCP over IPv6 VLAN Ingress

Template ID: 360. Field count: 57.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)

- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv6 VLAN Egress

Template ID: 361. Field count: 61.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))

- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv6 VLAN Ingress with Tunnel

Template ID: 362. Field count: 64.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))

- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)

- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

TCP over IPv6 VLAN Egress with Tunnel

Template ID: 363. Field count: 68.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)

- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)

- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)
- tcpAckTotalCount (Length: 8)
- tcpFinTotalCount (Length: 8)
- tcpPshTotalCount (Length: 8)
- tcpRstTotalCount (Length: 8)
- tcpSynTotalCount (Length: 8)
- tcpUrgTotalCount (Length: 8)

KVM UDP over IPv6 VLAN IPFIX Templates

There are four KVM UDP over IPv6 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

UDP over IPv6 VLAN Ingress

Template ID: 364. Field count: 51.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)

- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)

- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv6 VLAN Egress

Template ID: 365. Field count: 55.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)

- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv6 VLAN Ingress with Tunnel

Template ID: 366. Field count: 58.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))

- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

UDP over IPv6 VLAN Egress with Tunnel

Template ID: 367. Field count: 62.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)

- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))

- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM SCTP over IPv6 VLAN IPFIX Templates

There are four KVM SCTP over IPv6 VLAN IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

SCTP over IPv6 VLAN Ingress

Template ID: 368. Field count: 51.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)

- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv6 VLAN Egress

Template ID: 369. Field count: 55.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)

- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)

- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv6 VLAN Ingress with Tunnel

Template ID: 370. Field count: 58.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)

- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)

- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

SCTP over IPv6 VLAN Egress with Tunnel

Template ID: 371. Field count: 62.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)

- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- L4_SRC_PORT (Length: 2)
- L4_DST_PORT (Length: 2)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)

- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM ICMPv6 VLAN IPFIX Templates

There are four KVM ICMPv6 IPFIX templates: ingress, egress, ingress with tunnel, and egress with tunnel.

ICMPv6 Ingress

Template ID: 372. Field count: 51.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)

- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- ICMP_IPv6_TYPE (Length: 1)
- ICMP_IPv6_CODE (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)

- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv6 Egress

Template ID: 373. Field count: 55.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)

- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- ICMP_IPv6_TYPE (Length: 1)
- ICMP_IPv6_CODE (Length: 1)
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)

- postMCastOctetTotalCount (Length: 8)

ICMPv6 Ingress with Tunnel

Template ID: 374. Field count: 58.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- ICMP_IPv6_TYPE (Length: 1)
- ICMP_IPv6_CODE (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))
- 896 (length: 2, PEN: VMware Inc. (6876))

- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

ICMPv6 Egress with Tunnel

Template ID: 375. Field count: 62.

The fields are:

- observationPointId (length: 4)
- DIRECTION (length: 1)
- SRC_MAC (length: 6)
- DESTINATION_MAC (length: 6)
- ethernetType (length: 2)
- ethernetHeaderLength (length: 1)
- INPUT_SNMP (length: 4)
- Unknown(368) (length: 4)
- IF_NAME (length: variable)
- IF_DESC (length: variable)
- OUTPUT_SNMP (Length: 4)
- Unknown(369) (Length: 4)
- IF_NAME (Length: variable)
- IF_DESC (Length: variable)
- SRC_VLAN (Length: 2)
- dot1qVlanId (Length: 2)
- dot1qPriority (Length: 1)
- IP_PROTOCOL_VERSION (Length: 1)
- IP_TTL (Length: 1)
- PROTOCOL (Length: 1)
- IP_DSCP (Length: 1)
- IP_PRECEDENCE (Length: 1)
- IP_TOS (Length: 1)
- IPV6_SRC_ADDR (Length: 4)
- IPV6_DST_ADDR (Length: 4)
- FLOW_LABEL (Length: 4)
- ICMP_IPv6_TYPE (Length: 1)
- ICMP_IPv6_CODE (Length: 1)
- 893 (length: 4, PEN: VMware Inc. (6876))
- 894 (length: 4, PEN: VMware Inc. (6876))
- 895 (length: 1, PEN: VMware Inc. (6876))

- 896 (length: 2, PEN: VMware Inc. (6876))
- 897 (length: 2, PEN: VMware Inc. (6876))
- 891 (length: 1, PEN: VMware Inc. (6876))
- 892 (length: variable, PEN: VMware Inc. (6876))
- 898 (length: variable, PEN: VMware Inc. (6876))
- flowStartDeltaMicroseconds (length: 4)
- flowEndDeltaMicroseconds (length: 4)
- DROPPED_PACKETS (length: 8)
- DROPPED_PACKETS_TOTAL (length: 8)
- PKTS (length: 8)
- PACKETS_TOTAL (length: 8)
- Unknown(354) (length: 8)
- Unknown(355) (length: 8)
- Unknown(356) (length: 8)
- Unknown(357) (length: 8)
- Unknown(358) (length: 8)
- MUL_DPKTS (length: 8)
- postMCastPacketTotalCount (length: 8)
- Unknown(352) (length: 8)
- Unknown(353) (length: 8)
- flowEndReason (length: 1)
- DROPPED_BYTES (Length: 8)
- DROPPED_BYTES_TOTAL (Length: 8)
- BYTES (Length: 8)
- BYTES_TOTAL (Length: 8)
- BYTES_SQUARED (Length: 8)
- BYTES_SQUARED_PERMANENT (Length: 8)
- IP LENGTH MINIMUM (Length: 8)
- IP LENGTH MAXIMUM (Length: 8)
- MUL_DOCTETS (Length: 8)
- postMCastOctetTotalCount (Length: 8)

KVM Options IPFIX Templates

There is one KVM options template, based on IETF RFC 7011, section 3.4.2.

Options Template

Template ID: 462. Scope count: 1. Data count: 1.

Monitor a Logical Switch Port Activity

You can monitor the logical port activity for example, to troubleshoot network congestion and packets being dropped

Prerequisites

Verify that a logical switch port is configured. See Connecting a VM to a Logical Switch.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Switching > Ports**
3. Click the name of a port.
4. Click the **Monitor** tab.

   The port status and statistics are displayed.
5. To download a CSV file of the MAC addresses that has been learned by the host, click **Download MAC Table**.
6. To monitor activity on the port, click **Begin Tracking**.

   A port tracking page opens. You can view the bidirectional port traffic and identify dropped packets. The port tracker page also lists the switching profiles attached to the logical switch port.

Results

If you notice dropped packets because of network congestion, you can configure a QoS switching profile for the logical switch port to prevent data loss on preferred packets. See Understanding QoS Switching Profile.

Monitor Fabric Nodes

You can monitor fabric nodes such as hosts, edges, NSX Edge clusters, bridges, and transport nodes from the NSX Manager UI.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Fabric > Nodes** from the navigation panel.

3 Select one of the following tabs.

   - Hosts
   - Edges
   - Edge Clusters
   - Bridges
   - Transport Nodes

Results

Note On the Hosts screen, if the MPA Connectivity status is Down or Unknown for a host, ignore the LCP Connectivity status because it might be inaccurate.

Logical Switches

13

You can configure logical switches and related objects from the **Advanced Networking & Security** tab. A logical switch reproduces switching functionality, broadcast, unknown unicast, multicast (BUM) traffic, in a virtual environment decoupled from the underlying hardware.

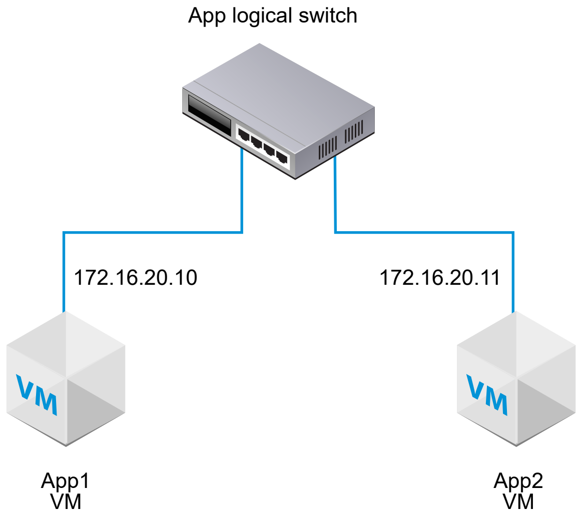
Note If you use the **Advanced Networking & Security** user interface to modify objects created in the policy interface, some settings might not be configurable. These read-only settings have this icon next to them: ⊖. See Chapter 1 Overview of the NSX Manager for more information.

Logical switches are similar to VLANs, in that they provide network connections to which you can attach virtual machines. The VMs can then communicate with each other over tunnels between hypervisors if the VMs are connected to the same logical switch. Each logical switch has a virtual network identifier (VNI), like a VLAN ID. Unlike VLAN, VNIs scale well beyond the limits of VLAN IDs.

To see and edit the VNI pool of values, log in to NSX Manager, navigate to **Fabric > Profiles**, and click the **Configuration** tab. Note that if you make the pool too small, creating a logical switch will fail if all the VNI values are in use. If you delete a logical switch, the VNI value will be re-used, but only after 6 hours.

When you add logical switches, it is important that you map out the topology that you are building.

Figure 13-1. Logical Switch Topology



For example, the topology above shows a single logical switch connected to two VMs. The two VMs can be on different hosts or the same host, in different host clusters or in the same host cluster. Because the VMs in the example are on the same virtual network, the underlying IP addresses configured on the VMs must be in the same subnet.

NSX Cloud Note If using NSX Cloud, see How to use NSX-T Data Center Features with the Public Cloud for a list of auto-generated logical entities, supported features, and configurations required for NSX Cloud.

This chapter includes the following topics:

- Understanding BUM Frame Replication Modes
- Create a Logical Switch
- Connecting a VM to a Logical Switch
- Create a Logical Switch Port
- Test Layer 2 Connectivity
- Create a VLAN Logical Switch for the NSX Edge Uplink
- Switching Profiles for Logical Switches and Logical Ports
- Layer 2 Bridging

Understanding BUM Frame Replication Modes

Each host transport node is a tunnel endpoint. Each tunnel endpoint has an IP address. These IP addresses can be in the same subnet or in different subnets, depending on your configuration of IP pools or DHCP for your transport nodes.

When two VMs on different hosts communicate directly, unicast-encapsulated traffic is exchanged between the two tunnel endpoint IP addresses associated with the two hypervisors without any need for flooding.

However, as with any Layer 2 network, sometimes traffic that is originated by a VM needs to be flooded, meaning that it needs to be sent to all of the other VMs belonging to the same logical switch. This is the case with Layer 2 broadcast, unknown unicast, and multicast traffic (BUM traffic). Recall that a single NSX-T Data Center logical switch can span multiple hypervisors. BUM traffic originated by a VM on a given hypervisor needs to be replicated to remote hypervisors that host other VMs that are connected to the same logical switch. To enable this flooding, NSX-T Data Center supports two different replication modes:

- Hierarchical two-tier (sometimes called MTEP)
- Head (sometimes called source)

Hierarchical two-tier replication mode is illustrated by the following example. Say you have Host A, which has VMs connected to virtual network identifiers (VNIs) 5000, 5001, and 5002. Think of VNIs as being similar to VLANs, but each logical switch has a single VNI associated with it. For this reason, sometimes the terms VNI and logical switch are used interchangeably. When we say a host is on a VNI, we mean that it has VMs that are connected to a logical switch with that VNI.

A tunnel endpoint table shows the host-VNI connections. Host A examines the tunnel endpoint table for VNI 5000 and determines the tunnel endpoint IP addresses for other hosts on VNI 5000.

Some of these VNI connections will be on the same IP subnet, also called an IP segment, as the tunnel endpoint on Host A. For each of these, Host A creates a separate copy of every BUM frame and sends the copy directly to each host.

Other hosts’ tunnel endpoints are on different subnets or IP segments. For each segment where there is more than one tunnel endpoint, Host A nominates one of these endpoints to be the replicator.

The replicator receives from Host A one copy of each BUM frame for VNI 5000. This copy is flagged as Replicate locally in the encapsulation header. Host A does not send copies to the other hosts in the same IP segment as the replicator. It becomes the responsibility of the replicator to create a copy of the BUM frame for each host it knows about that is on VNI 5000 and in the same IP segment as that replicator host.

The process is replicated for VNI 5001 and 5002. The list of tunnel endpoints and the resulting replicators might be different for different VNIs.

With head replication also known as headend replication, there are no replicators. Host A simply creates a copy of each BUM frame for each tunnel endpoint it knows about on VNI 5000 and sends it.

If all the host tunnel endpoints are on the same subnet, the choice of replication mode does not make any difference because the behaviour will not differ. If the host tunnel endpoints are on different subnets, hierarchical two-tier replication helps distribute the load among multiple hosts. Hierarchical two-tier is the default mode.

Create a Logical Switch

Logical switches attach to single or multiple VMs in the network. The VMs connected to a logical switch can communicate with each other using the tunnels between hypervisors.

Prerequisites

- Verify that a transport zone is configured. See the *NSX-T Data Center Installation Guide*.
- Verify that fabric nodes are successfully connected to NSX-T Data Center management plane agent (MPA) and NSX-T Data Center local control plane (LCP).

  In the `GET https://<nsx-mgr>/api/v1/transport-nodes/<transport-node-id>/state` API call, the `state` must be `success`. See the *NSX-T Data Center Installation Guide*.
- Verify that transport nodes are added to the transport zone. See the *NSX-T Data Center Installation Guide*.
- Verify that the hypervisors are added to the NSX-T Data Center fabric and VMs are hosted on these hypervisors.
- Familiarize yourself with the logical switch topology and BUM frame replication concepts. See Chapter 13 Logical Switches and Understanding BUM Frame Replication Modes.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Switching > Switches > Add**.
3. Enter a name for the logical switch and optionally a description.
4. Select a transport zone for the logical switch.

   VMs that are attached to logical switches that are in the same transport zone can communicate with each other.
5. Enter the name of an uplink teaming policy.
6. Set **Admin Status** to either **Up** or **Down**.
7. Select a replication mode for the logical switch.

   The replication mode (hierarchical two-tier or head) is required for overlay logical switches, but not for VLAN-based logical switches.

Replication Mode	Description
Hierarchical two-tier	The replicator is a host that performs replication of BUM traffic to other hosts within the same VNI. Each host nominates one host tunnel endpoint in every VNI to be the replicator. This is done for each VNI.
Head	Hosts create a copy of each BUM frame and send the copy to each tunnel endpoint it knows about for each VNI.

8 (Optional) Specify a VLAN ID or ranges of VLAN IDs for VLAN tagging.

   To support guest VLAN tagging for VMs connected to this switch, you must specify VLAN ID ranges, also called trunk VLAN ID ranges. The logical port will filter packets based on the trunk VLAN ID ranges, and a guest VM can tag its packets with its own VLAN ID based on the trunk VLAN ID ranges.

9 (Optional) Click the **Switching Profiles** tab and select switching profiles.

10 Click **Save**.

   In the NSX Manager UI, the new logical switch is a clickable link.

What to do next

Attach VMs to your logical switch. See Connecting a VM to a Logical Switch.

Connecting a VM to a Logical Switch

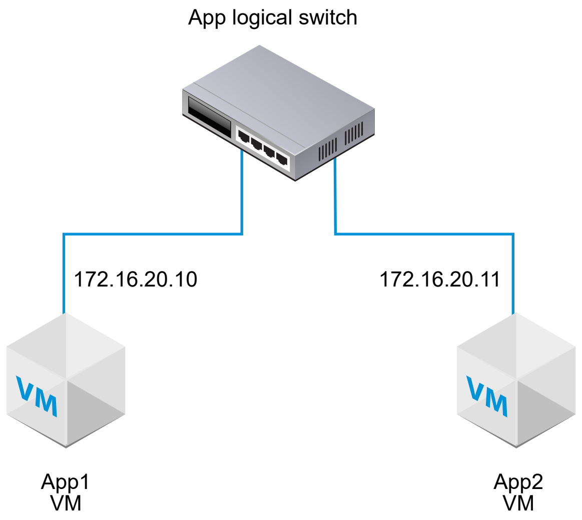
Depending on your host, the configuration for connecting a VM to a logical switch can vary.

The supported hosts that can connect to a logical switch are; an ESXi host that is managed in vCenter Server, a standalone ESXi host, and a KVM host.

Attach a VM Hosted on vCenter Server to an NSX-T Data Center Logical Switch

If you have a ESXi host that is managed in vCenter Server, you can access the host VMs through the Web-based vSphere Web Client. In this case, you can use this procedure to attach VMs to NSX-T Data Center logical switches.

The example shown in this procedure shows how to attach a VM called app-vm to a logical switch called app-switch.



The installation-based vSphere Client application does not support attaching a VM to an NSX-T Data Center logical switch. If you do not have the (Web-based) vSphere Web Client, see Attach a VM Hosted on Standalone ESXi to an NSX-T Data Center Logical Switch.

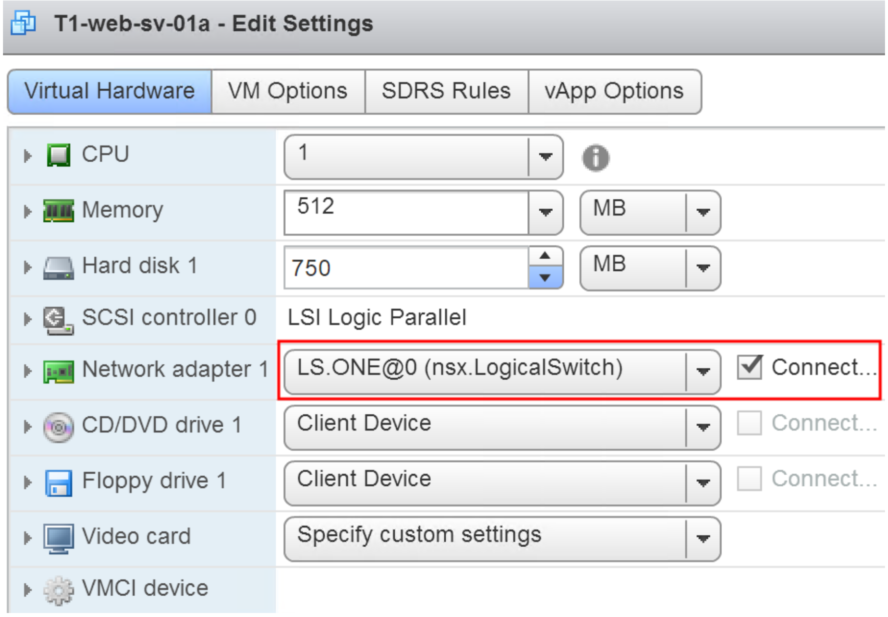
Prerequisites

- The VMs must be hosted on hypervisors that have been added to the NSX-T Data Center fabric.
- The fabric nodes must have NSX-T Data Center management plane (MPA) and NSX-T Data Center control plane (LCP) connectivity.
- The fabric nodes must be added to a transport zone.
- A logical switch must be created.

Procedure

1 In the vSphere Web Client, edit the VM settings, and attach the VM to the NSX-T Data Center logical switch.

For example:



T1-web-sv-01a - Edit Settings

CPU	1	
Memory	512	MB
Hard disk 1	750	MB
SCSI controller 0	LSI Logic Parallel	
Network adapter 1	LS.ONE@0 (nsx.LogicalSwitch)	☑ Connect...
CD/DVD drive 1	Client Device	☐ Connect...
Floppy drive 1	Client Device	☐ Connect...
Video card	Specify custom settings	
VMCI device		

2 Click **OK**.

Results

After attaching a VM to a logical switch, logical switch ports are added to the logical switch. You can view logical switch ports on the NSX Manager in **Advanced Networking & Security > Networking > Switching > Ports**.

In the NSX-T Data Center API, you can view NSX-T Data Center-attached VMs with the `GET https://<nsx-mgr>/api/v1/fabric/virtual-machines` API call

In the NSX-T Data Center Manager UI under **Advanced Networking & Security > Networking > Switching > Ports**, the VIF attachment ID matches the ExternalID found in the API call. Find the VIF attachment ID matching the VM's externalId and make sure that the Admin and Operational status are Up/Up.

If two VMs are attached to the same logical switch and have IP addresses configured in the same subnet, they should be able to ping each other.

What to do next

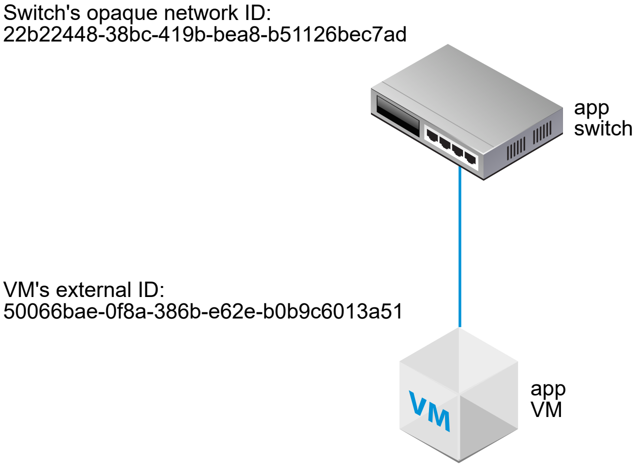
Add a logical router.

You can monitor the activity on the logical switch port to troubleshoot problems. See "Monitor a Logical Switch Port Activity" in the *NSX-T Data Center Administration Guide*.

Attach a VM Hosted on Standalone ESXi to an NSX-T Data Center Logical Switch

If you have a standalone ESXi host, you cannot access the host VMs through the web-based vSphere Web Client. In this case, you can use this procedure to attach VMs to NSX-T Data Center logical switches.

The example shown in this procedure shows how to attach a VM called app-vm to a logical switch called app-switch.


Switch's opaque network ID:
22b22448-38bc-419b-bea8-b51126bec7ad
app
switch
VM's external ID:
50066bae-0f8a-386b-e62e-b0b9c6013a51
VM
app
VM

Prerequisites

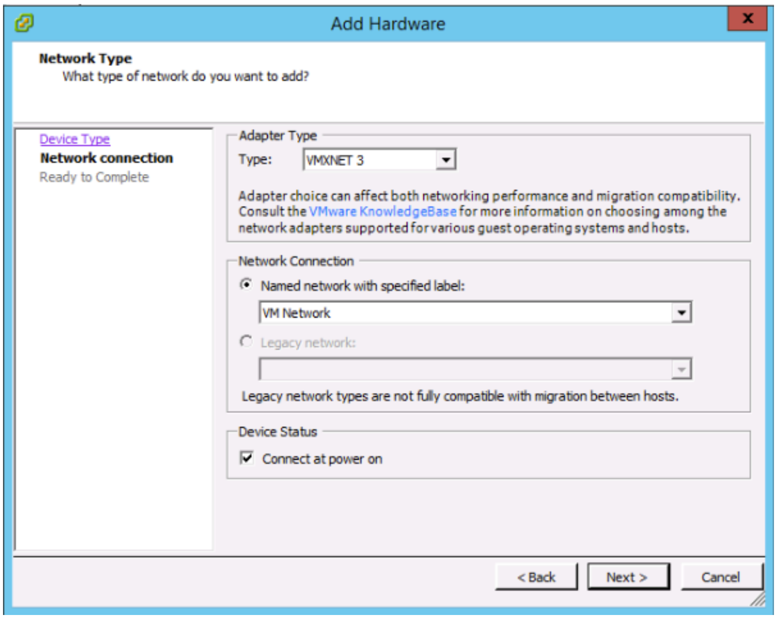
- The VM must be hosted on hypervisors that have been added to the NSX-T Data Center fabric.
- The fabric nodes must have NSX-T Data Center management plane (MPA) and NSX-T Data Center control plane (LCP) connectivity.
- The fabric nodes must be added to a transport zone.
- A logical switch must be created.
- You must have access to the NSX Manager API.

- You must have write access to the VM's VMX file.

Procedure

1 Using the (install-based) vSphere Client application or some other VM management tool, edit the VM and add a VMXNET 3 Ethernet adapter.

Select any named network. You will change the network connection in a later step.



Add Hardware

Network Type

What type of network do you want to add?

Adapter Type

Type: VMXNET 3

Adapter choice can affect both networking performance and migration compatibility. Consult the VMware KnowledgeBase for more information on choosing among the network adapters supported for various guest operating systems and hosts.

Network Connection

● Named network with specified label: VM Network

○ Legacy network:

Legacy network types are not fully compatible with migration between hosts.

Device Status

☑ Connect at power on

2 Use the NSX-T Data Center API to issue the `GET https://<nsx-mgr>/api/v1/fabric/virtual-machines/<VM-ID>` API call.

In the results, find the VM's externalId.

For example:

```
GET https://<nsx-mgr>/api/v1/fabric/virtual-machines/60a5a5d5-ea2b-407e-a806-4fdc8468f735

{
  "resource_type": "VirtualMachine",
  "id": "60a5a5d5-ea2b-407e-a806-4fdc8468f735",
  "display_name": "app-vm",
  "compute_ids": [
    "instanceUuid:50066bae-0f8a-386b-e62e-b0b9c6013a51",
    "moIdOnHost:5",
    "externalId:50066bae-0f8a-386b-e62e-b0b9c6013a51",
    "hostLocalId:5",
    "locationId:564dc020-1565-e3f4-f591-ee3953eef3ff",
    "biosUuid:4206f47d-fef7-08c5-5bf7-ea26a4c6b18d"
  ],
  "external_id": "50066bae-0f8a-386b-e62e-b0b9c6013a51",
  "type": "REGULAR",
  "host_id": "cb82b0fa-a8f1-11e5-92a9-6b7d1f8661fa",
  "local_id_on_host": "5"
}
```

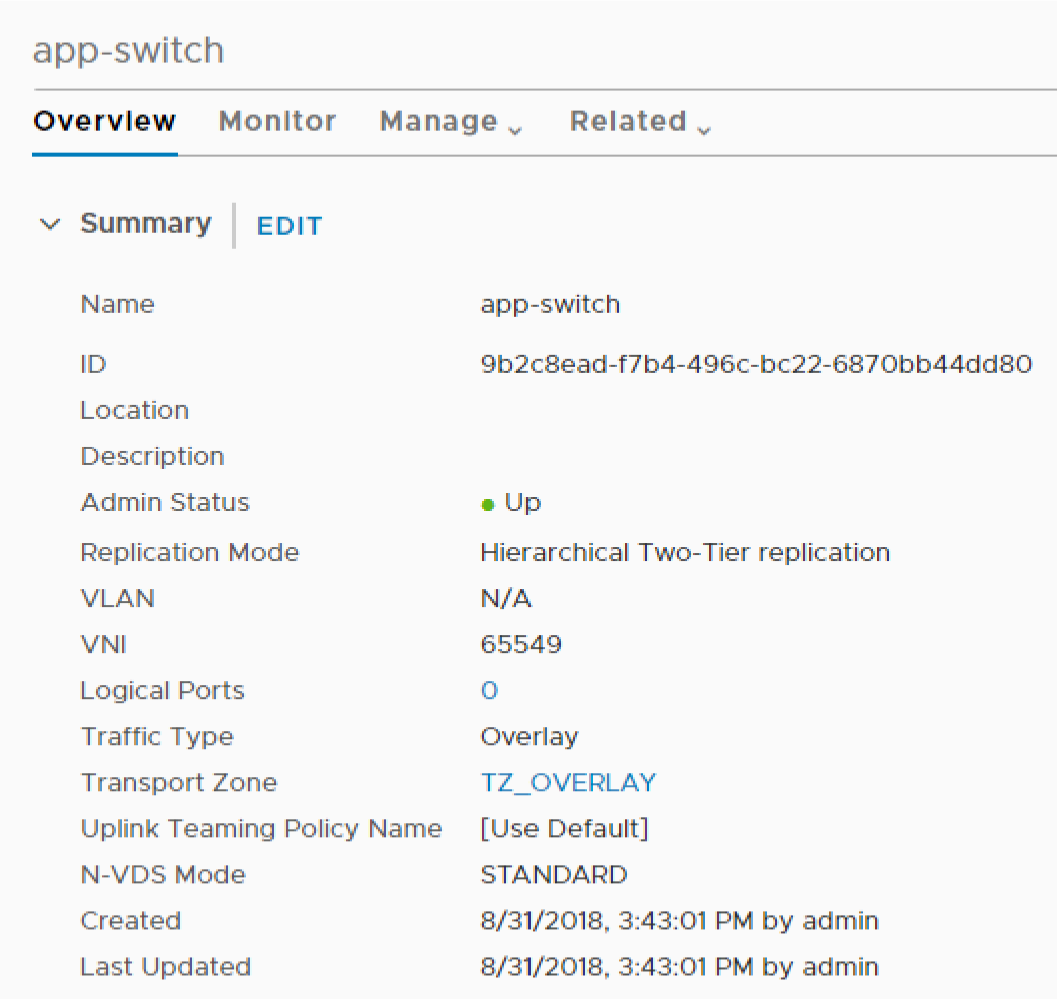
3 Power off and unregister the VM from the host.

   You can use your VM management tool or the ESXi CLI, as shown here.

```
[user@host:~] vim-cmd /vmsvc/getallvms
Vmid    Name             File               Guest OS      Version   Annotation
5      app-vm   [ds2] app-vm/app-vm.vmx   ubuntuGuest     vmx-08
8      web-vm   [ds2] web-vm/web-vm.vmx   ubuntu64Guest   vmx-08

[user@host:~] vim-cmd /vmsvc/power.off 5
Powering off VM:

[user@host:~] vim-cmd /vmsvc/unregister 5
```

4 From the NSX Manager UI, get the logical switch ID.

   For example:



Name	app-switch
ID	9b2c8ead-f7b4-496c-bc22-6870bb44dd80
Location	
Description	
Admin Status	● Up
Replication Mode	Hierarchical Two-Tier replication
VLAN	N/A
VNI	65549
Logical Ports	0
Traffic Type	Overlay
Transport Zone	TZ_OVERLAY
Uplink Teaming Policy Name	[Use Default]
N-VDS Mode	STANDARD
Created	8/31/2018, 3:43:01 PM by admin
Last Updated	8/31/2018, 3:43:01 PM by admin

5 Modify the VM's VMX file.

Delete the **ethernet1.networkName = "<name>"** field and add the following fields:

- ethernet1.opaqueNetwork.id = "<logical switch's ID>"
- ethernet1.opaqueNetwork.type = "nsx.LogicalSwitch"
- ethernet1.externalId = "<VM's externalId>"
- ethernet1.connected = "TRUE"
- ethernet1.startConnected = "TRUE"

For example:

```
OLD
ethernet1.pciSlotNumber = "224"
ethernet1.virtualDev = "vmxnet3"
ethernet1.networkName = "VM Network"
ethernet1.addressType = "vpx"
```

```
ethernet1.generatedAddress = "00:50:56:86:7b:d7"
ethernet1.uptCompatibility = "true"
ethernet1.present = "TRUE"
```

```
NEW
ethernet1.pciSlotNumber = "224"
ethernet1.virtualDev = "vmxnet3"
ethernet1.addressType = "vpx"
ethernet1.generatedAddress = "00:50:56:86:7b:d7"
ethernet1.uptCompatibility = "true"
ethernet1.present = "TRUE"
ethernet1.opaqueNetwork.id = "22b22448-38bc-419b-bea8-b51126bec7ad"
ethernet1.opaqueNetwork.type = "nsx.LogicalSwitch"
ethernet1.externalId = "50066bae-0f8a-386b-e62e-b0b9c6013a51"
ethernet1.connected = "TRUE"
ethernet1.startConnected = "TRUE"
```

6 In the NSX Manager UI, add a logical switch port, and use the VM's externalId for the VIF attachment.

7 Reregister the VM and power it on.

   You can use your VM management tool or the ESXi CLI, as shown here.

```
[user@host:~] vim-cmd /solo/register /path/to/file.vmx

For example:
[user@host:~] vim-cmd solo/registervm /vmfs/volumes/355f2049-6c704347/app-vm/app-vm.vmx
9

[user@host:~] vim-cmd /vmsvc/power.on 9
Powering on VM:
```

Results

In the NSX Manager UI under **Advanced Networking & Security > Networking > Switching > Ports**, find the VIF attachment ID matching the VM's externalId and make sure that the Admin and Operational status are Up/Up.

If two VMs are attached to the same logical switch and have IP addresses configured in the same subnet, they should be able to ping each other.

What to do next

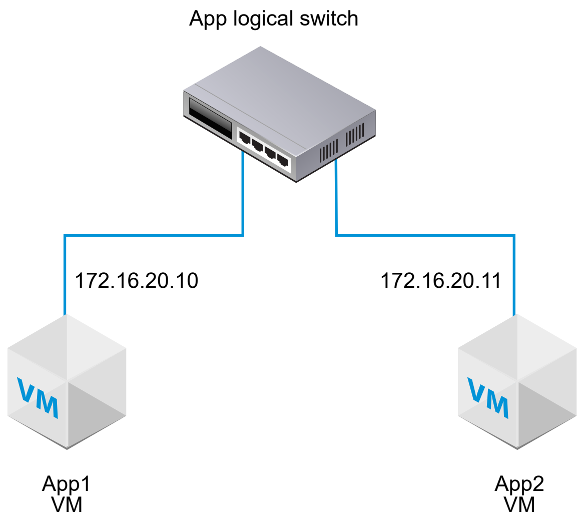
Add a logical router.

You can monitor the activity on the logical switch port to troubleshoot problems. See "Monitor a Logical Switch Port Activity" in the *NSX-T Data Center Administration Guide*.

Attach a VM Hosted on KVM to an NSX-T Data Center Logical Switch

If you have a KVM host, you can use this procedure to attach VMs to NSX-T Data Center logical switches.

The example shown in this procedure shows how to attach a VM called app-vm to a logical switch called app-switch.



Prerequisites

- The VM must be hosted on hypervisors that have been added to the NSX-T Data Center fabric.
- The fabric nodes must have NSX-T Data Center management plane (MPA) and NSX-T Data Center control plane (LCP) connectivity.
- The fabric nodes must be added to a transport zone.
- A logical switch must be created.

Procedure

1. From the KVM CLI, run the `virsh dumpxml <your vm> | grep interfaceid` command.
2. In the NSX Manager UI, add a logical switch port, and use the VM's interface ID for the VIF attachment.

Results

In the NSX Manager UI under **Advanced Networking & Security > Networking > Switching > Ports**, find the VIF attachment ID and make sure that the Admin and Operational status are Up/Up.

If two VMs are attached to the same logical switch and have IP addresses configured in the same subnet, they should be able to ping each other.

What to do next

Add a logical router.

You can monitor the activity on the logical switch port to troubleshoot problems. See "Monitor a Logical Switch Port Activity" in the *NSX-T Data Center Administration Guide*.

Create a Logical Switch Port

A logical switch has multiple switch ports. A logical switch port connects another network component, a VM, or a container to a logical switch.

If you connect a VM to a logical switch on an ESXi host that is managed by vCenter Server, a logical switch port is created automatically. For more information about connecting a VM to a logical switch, see Connecting a VM to a Logical Switch.

For more information about connecting a container to a logical switch, see the *NSX-T Container Plug-in for Kubernetes - Installation and Administration Guide*.

Note The IP address and MAC address bound to a logical switch port for a container are allocated by NSX Manager. Do not change the address binding manually.

To monitor activity on a logical switch port, see Monitor a Logical Switch Port Activity.

Prerequisites

Verify that a logical switch is created. See Chapter 13 Logical Switches.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Switching > Ports > Add**.
3. In the **General** tab, complete the port details.

Option	Description
Name and Description	Enter a name and optionally a description.
Logical Switch	Select a logical switch from the drop-down menu.
Admin Status	Select **Up** or **Down**.
Attachment Type	Select **None** or **VIF**.
Attachment ID	If the attachment type is VIF, enter the attachment ID.

   Using the API, you can set the attachment type to additional values (`LOGICALROUTER`, `BRIDGEENDPOINT`, `DHCP_SERVICE`, `METADATA_PROXY`, `L2VPN_SESSION`). If the attachment type is DHCP service, metadata proxy, or L2 VPN session, the switching profiles for the port must be the default ones. You cannot use any user-defined profile.

4. (Optional) In the **Switching Profiles** tab, select switching profiles.

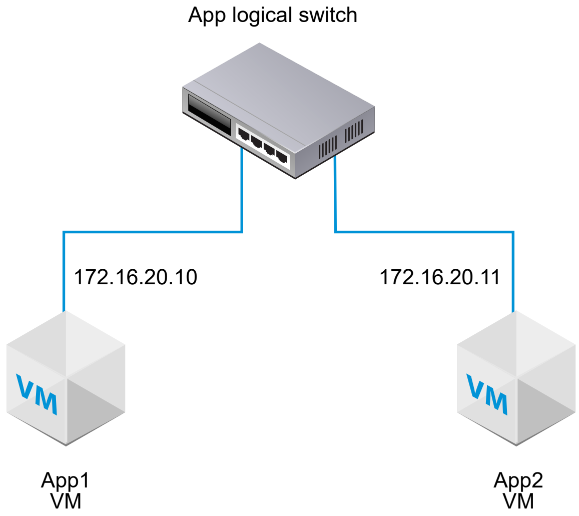
5 Click **Save**.

Test Layer 2 Connectivity

After you successfully set up your logical switch and attach VMs to the logical switch, you can test the network connectivity of the attached VMs.

If your network environment is configured properly, based on the topology the App2 VM can ping the App1 VM.

Figure 13-2. Logical Switch Topology



Procedure

1 Log in to one of the VMs attached to the logical switch using SSH or the VM console.

For example, App2 VM 172.16.20.11.

2 Ping the second VM attached to the logical switch to test connectivity.

```
$ ping –c 2 172.16.20.10
PING 172.16.20.10 (172.16.20.10) 56(84) bytes of data.
64 bytes from 172.16.20.10: icmp_seq=1 ttl=63 time=0.982 ms
64 bytes from 172.16.20.10: icmp_seq=2 ttl=63 time=0.654 ms
64 bytes from 172.16.20.10: icmp_seq=3 ttl=63 time=0.791 ms

––– 172.16.20.10 ping statistics –––
2 packets transmitted, 2 received, 0% packet loss, time 1990ms
rtt min/avg/max/mdev = 0.654/0.809/0.902/0.104 ms
```

3 (Optional) Identify the problem that causes the ping to fail.

   a Verify that the VM network settings are correct.

   b Verify that the VM network adapter is connected to the correct logical switch.

   c Verify that the logical switch Admin status is UP.

d From the NSX Manager, select **Advanced Networking & Security > Networking > Switching > Switches**.

e Click the logical switch and note the UUID and VNI information.

f Run the following commands to troubleshoot the problem.

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td><code>get logical-switch <vni-or-uuid> arp-table</code></td><td>Displays the ARP table for the specified logical switch.
Sample output.
<pre>
nsx-manager1> get logical-switch 41866 arp-table
VNI   IP   MAC   Connection-ID
41866 172.16.20.11 00:50:56:b1:70:5e   295422
</pre></td></tr>
<tr><td><code>get logical-switch <vni-or-uuid> connection-table</code></td><td>Displays the connections for the specified logical switch.
Sample output.
<pre>
nsx-manager1> get logical-switch 41866 connection-table
Host-IP   Port   ID
192.168.110.37   36923 295420
192.168.210.53   37883 295421
192.168.210.54   57278 295422
</pre></td></tr>
<tr><td><code>get logical-switch <vni-or-uuid> mac-table</code></td><td>Displays the MAC table for the specified logical switch.
Sample output.
<pre>
nsx-manager1> get logical-switch 41866 mac-table
VNI   MAC   VTEP-IP   Connection-ID
41866 00:50:56:86:f2:b2 192.168.250.102   295421
41866 00:50:56:b1:70:5e 192.168.250.101   295422
</pre></td></tr>
<tr><td><code>get logical-switch <vni-or-uuid> stats</code></td><td>Displays statistics information about the specified logical switch.
Sample output.
<pre>
nsx-manager1> get logical-switch 41866 stats
update.member 11
update.vtep 11
update.mac 4
update.mac.invalidate 0
update.arp 7
update.arp.duplicate 0
query.mac 2
query.mac.miss 0
query.arp 9
query.arp.miss 6
</pre></td></tr>
<tr><td><code>get logical-switch <vni-or-uuid> stats-sample</code></td><td>Displays a summary of all logical switch statistics over time.
Sample output.
<pre>
nsx-manager1> get logical-switch 41866 stats-sample
21:00:00 21:10:00 21:20:00 21:30:00 21:40:00
update.member 0 0 0 0 0
update.vtep 0 0 0 0 0
update.mac 0 0 0 0 0
update.mac.invalidate 0 0 0 0 0
update.arp 0 0 0 0 0
update.arp.duplicate 0 0 0 0 0
</pre></td></tr>
</table>

Command	Description

```
query.mac 0 0 0 0 0
query.mac.miss 0 0 0 0 0
query.arp 0 0 0 0 0
query.arp.miss 0 0 0 0 0
```

Command	Description
`get logical-switch <vni-or-uuid> vtep`	Displays all virtual tunnel end points related to the specified logical switch. Sample output.

```
nsx-manager1> get logical-switch 41866 vtep
VNI      IP           LABEL    Segment
MAC          Connection-ID
41866 192.168.250.102 0x8801  192.168.250.0
00:50:56:65:f5:fc 295421
41866 192.168.250.100 0x1F801 192.168.250.0
02:50:56:00:00:00 295420
41866 192.168.250.101 0x16001 192.168.250.0
00:50:56:64:7c:28 295422
```

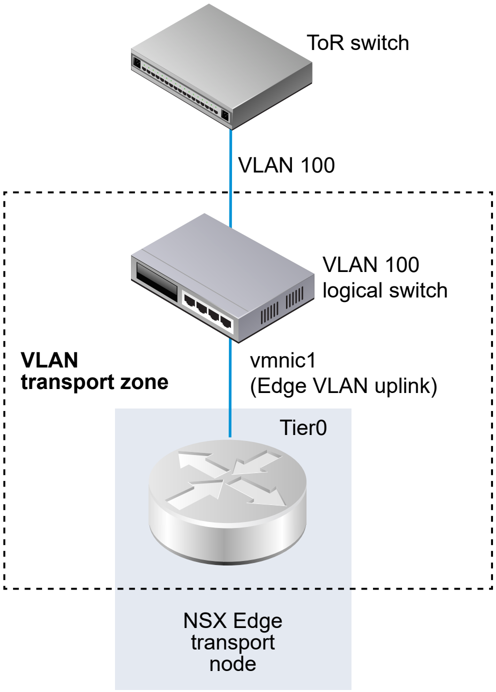
Results

The first VM attached to the logical switch is able to send packets to the second VM.

Create a VLAN Logical Switch for the NSX Edge Uplink

Edge uplinks go out through VLAN logical switches.

When you are creating a VLAN logical switch, it is important to have in mind a particular topology that you are building. For example, the following simple topology shows a single VLAN logical switch inside of a VLAN transport zone. The VLAN logical switch has VLAN ID 100. This matches the VLAN ID on the TOR port connected to the hypervisor host port used for the Edge's VLAN uplink.



Prerequisites

- To create a VLAN logical switch, you must first create a VLAN transport zone.
- An NSX-T Data Center vSwitch must be added to the NSX Edge. To confirm on an Edge, run the `get host-switches` command. For example:

```
nsx-edge1> get host-switches

Host Switch         : c0a78378-1c20-432a-9e23-ddb34f1c80c9
Switch Name         : hs1
Transport Zone      : c46dcd72-808a-423d-b4cc-8752c33f6b2c
Transport Zone      : 73def985-d122-4b7b-ab6a-a58176dfc32d
Physical Port       : fp-eth0
Uplink Name         : uplink-1
Transport VLAN      : 4096
Default Gateway     : 192.168.150.1
Subnet Mask         : 255.255.255.0
Local VTEP Device   : fp-eth0
Local VTEP IP       : 192.168.150.102
```

- Verify that fabric nodes are successfully connected to the NSX-T Data Center management plane agent (MPA) and the NSX-T Data Center local control plane (LCP).

  In the `GET https://<nsx-mgr>/api/v1/transport-nodes/<transport-node-id>/state` API call, the `state` must be `success`. See the *NSX-T Data Center Installation Guide*.

Procedure

1. From a browser, log in to an NSX Manager at `https://<nsx-mgr>`.
2. Select **Advanced Networking & Security > Networking > Switching > Switches > Add**.

3 Type a name for the logical switch.

4 Select a transport zone for the logical switch.

5 Select an uplink teaming policy.

6 For admin status, select **Up** or **Down**.

7 Type a VLAN ID.

   Enter 0 in the VLAN field if there is no VLAN ID for the uplink to the physical TOR.

8 (Optional) Click the **Switching Profiles** tab and select switching profiles.

Results

Note If you have two VLAN logical switches that have the same VLAN ID, they cannot be connected to the same Edge N-VDS (previously known as hostswitch). If you have a VLAN logical switch and an overlay logical switch, and the VLAN ID of the VLAN logical switch is the same as the transport VLAN ID of the overlay logical switch, they also cannot be connected to the same Edge N-VDS.

What to do next

Add a logical router.

Switching Profiles for Logical Switches and Logical Ports

Switching profiles include Layer 2 networking configuration details for logical switches and logical ports. NSX Manager supports several types of switching profiles, and maintains one or more system-defined default switching profiles for each profile type.

The following types of switching profiles are available.

- QoS (Quality of Service)
- Port Mirroring
- IP Discovery
- SpoofGuard
- Switch Security
- MAC Management

Note You cannot edit or delete the default switching profiles in the NSX Manager. You can create custom switching profiles instead.

Before using a default profile, make sure that the settings are what you need them to be. When you create a custom profile, some settings have default values. Do not assume that in the default profile, these settings will have the default values.

Each default or custom switching profile has a unique reserved identifier. You use this identifier to associate the switching profile to a logical switch or a logical port. For example, the default QoS switching profile ID is f313290b-eba8-4262-bd93-fab5026e9495.

A logical switch or logical port can be associated with one switching profile of each type. You cannot have for example, two QoS different switching profiles associated to a logical switch or logical port.

If you do not associate a switching profile type while creating or updating a logical switch, then the NSX Manager associates a corresponding default system-defined switching profile. The children logical ports inherit the default system-defined switching profile from the parent logical switch.

When you create or update a logical switch or logical port you can choose to associate either a default or a custom switching profile. When the switching profile is associated or disassociated from a logical switch the switching profile for the children logical ports is applied based on the following criteria.

- If the parent logical switch has a profile associated with it, the child logical port inherits the switching profile from the parent.
- If the parent logical switch does not have a switching profile associated with it, a default switching profile is assigned to the logical switch and the logical port inherits that default switching profile.
- If you explicitly associate a custom profile with a logical port, then this custom profile overrides the existing switching profile.

Note If you have associated a custom switching profile with a logical switch, but want to retain the default switching profile for one of the child logical port, then you must make a copy of the default switching profile and associate it with the specific logical port.

You cannot delete a custom switching profile if it is associated to a logical switch or a logical port. You can find out whether any logical switches and logical ports are associated with the custom switching profile by going to the Assigned To section of the Summary view and clicking on the listed logical switches and logical ports.

Understanding QoS Switching Profile

QoS provides high-quality and dedicated network performance for preferred traffic that requires high bandwidth. The QoS mechanism does this by prioritizing sufficient bandwidth, controlling latency and jitter, and reducing data loss for preferred packets even when there is a network congestion. This level of network service is provided by using the existing network resources efficiently.

For this release, shaping and traffic marking namely, CoS and DSCP is supported. The Layer 2 Class of Service (CoS) allows you to specify priority for data packets when traffic is buffered in the logical switch due to congestion. The Layer 3 Differentiated Services Code Point (DSCP) detects packets based on their DSCP values. CoS is always applied to the data packet irrespective of the trusted mode.

NSX-T Data Center trusts the DSCP setting applied by a virtual machine or modifying and setting the DSCP value at the logical switch level. In each case, the DSCP value is propagated to the outer IP header of encapsulated frames. This enables the external physical network to prioritize the traffic based on the DSCP setting on the external header. When DSCP is in the trusted mode, the DSCP value is copied from the inner header. When in the untrusted mode, the DSCP value is not preserved for the inner header.

Note DSCP settings work only on tunneled traffic. These settings do not apply to traffic inside the same hypervisor.

You can use the QoS switching profile to configure the average ingress and egress bandwidth values to set the transmit limit rate. The peak bandwidth rate is used to support burst traffic a logical switch is allowed to prevent congestion on the northbound network links. These settings do not guarantee the bandwidth but help limit the use of network bandwidth. The actual bandwidth you will observe is determined by the link speed of the port or the values in the switching profile, whichever is lower.

The QoS switching profile settings are applied to the logical switch and inherited by the child logical switch port.

Configure a Custom QoS Switching Profile

You can define the DSCP value and configure the ingress and egress settings to create a custom QoS switching profile.

Prerequisites

- Familiarize yourself with the QoS switching profile concept. See Understanding QoS Switching Profile.
- Identify the network traffic you want to prioritize.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Switching > Switching Profiles > Add**

3 Select **QoS** and complete the QoS switching profile details.

Option	Description
Name and Description	Assign a name to the custom QoS switching profile. You can optionally describe the setting that you modified in the profile.
Mode	Select either a **Trusted** or **Untrusted** option from the Mode drop-down menu. When you select the Trusted mode the inner header DSCP value is applied to the outer IP header for IP/IPv6 traffic. For non IP/IPv6 traffic, the outer IP header takes the default value. Trusted mode is supported on an overlay-based logical port. The default value is 0. Untrusted mode is supported on overlay-based and VLAN-based logical port. For the overlay-based logical port, the DSCP value of the outbound IP header is set to the configured value irrespective to the inner packet type for the logical port. For the VLAN-based logical port, the DSCP value of IP/IPv6 packet will be set to the configured value. The DSCP values range for untrusted mode is between 0 to 63. **Note** DSCP settings work only on tunneled traffic. These settings do not apply to traffic inside the same hypervisor.
Priority	Set the DSCP value. The priority values range from 0 to 63.
Class of Service	Set the CoS value. CoS is supported on VLAN-based logical port. CoS groups similar types of traffic in the network and each type of traffic is treated as a class with its own level of service priority. The lower priority traffic is slowed down or in some cases dropped to provide better throughput for higher priority traffic. CoS can also be configured for the VLAN ID with zero packet. The CoS values range from 0 to 7, where 0 is the best effort service.
Ingress	Set custom values for the outbound network traffic from the VM to the logical network. You can use the average bandwidth to reduce network congestion. The peak bandwidth rate is used to support burst traffic and the burst size is based on the duration with peak bandwidth. You set burst duration in the burst size setting. You cannot guarantee the bandwidth. However, you can use the Average, Peak, and Burst Size settings to limit network bandwidth. For example, if the average bandwidth is 30 Mbps, peak bandwidth is 60 Mbps, and the allowed duration is 0.1 second, then the burst size is 60 * 1000000 * 0.10/8 = 750000 Bytes. The default value 0 disables rate limiting on the ingress traffic.

Option	Description
Ingress Broadcast	Set custom values for the outbound network traffic from the VM to the logical network based on broadcast. Set custom values for the outbound network traffic from the VM to the logical network based on broadcast. For example, when you set the average bandwidth for a logical switch to 3000 Kbps, peak bandwidth is 6000 Kbps, and the allowed duration is 0.1 second, then the burst size is 6000 * 1000 * 0.10/8 = 75000 Bytes. The default value 0 disables rate limiting on the ingress broadcast traffic.
Egress	Set custom values for the inbound network traffic from the logical network to the VM. The default value 0 disables rate limiting on the egress traffic.

If the ingress, ingress broadcast, and egress options are not configured the default values are used.

4 Click **Save**.

Results

A custom QoS switching profile appears as a link.

What to do next

Attach this QoS customized switching profile to a logical switch or logical port so that the modified parameters in the switching profile are applied to the network traffic. See Associate a Custom Profile with a Logical Switch or Associate a Custom Profile with a Logical Port.

Understanding Port Mirroring Switching Profile

Logical port mirroring lets you replicate and redirect all of the traffic coming in or out of a logical switch port attached to a VM VIF port. The mirrored traffic is sent encapsulated within a Generic Routing Encapsulation (GRE) tunnel to a collector so that all of the original packet information is preserved while traversing the network to a remote destination.

Typically port mirroring is used in the following scenarios:

- Troubleshooting - Analyze the traffic to detect intrusion and debug and diagnose errors on a network.
- Compliance and monitoring - Forward all of the monitored traffic to a network appliance for analysis and remediation.

Compared to the physical port mirroring, logical port mirroring ensures that all of the VM network traffic is captured. If you implement port mirroring only in the physical network, some of the VM network traffic fails to be mirrored. This happens because communication between VMs residing on the same host never enters the physical network and therefore does not get mirrored. With logical port mirroring you can continue to mirror VM traffic even when that VM is migrated to another host.

The port mirroring process is similar for both VM ports in the NSX-T Data Center domain and ports of physical applications. You can forward the traffic captured by a workload connected to a logical network and mirror that traffic to a collector. The IP address should be reachable from the guest IP address on which the VM is hosted. This process is also true for physical applications connected to Gateway nodes.

Configure a Custom Port Mirroring Switching Profile

You can create a custom port mirroring switching profile with a different destination and key value.

Prerequisites

- Familiarize yourself with the port mirroring switching profile concept. See Understanding Port Mirroring Switching Profile.
- Identify the IP address of the destination logical port ID you want to redirect network traffic to.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security** > **Networking** > **Switching** > **Switching Profiles** > **Add**
3. Select **Port Mirroring** and complete the port mirroring switching profile details.

Option	Description
Name and Description	Assign a name to the custom port mirroring switching profile. You can optionally describe the setting you modified to customize this profile.
Direction	Select an option from the drop-down menu to use this source for **Ingress**, **Egress**, or **Bidirectional** traffic. Ingress is the outbound network traffic from the VM to the logical network. Egress is the inbound network traffic from the logical network to the VM. Bidirectional is the two-way of traffic from the VM to the logical network and from the logical network to the VM. This is the default option.
Packet Truncation	Optional. The range is 60 - 65535.

Option	Description
Key	Enter a random 32-bit value to identify mirrored packets from the logical port. This Key value is copied to the Key field in the GRE header of each mirror packet. If the Key value is set to 0, the default definition is copied to the Key field in the GRE header. The default 32-bit value is made of the following values. ■ The first 24-bit is a VNI value. VNI is part of the IP header of encapsulated frames. ■ The 25th bit indicates if the first 24-bit is a valid VNI value. One represents a valid value and zero represents an invalid value. ■ The 26th bit indicates the direction of the mirrored traffic. One represents an ingress direction and zero represents an egress direction. ■ The remaining six bits are not used.
Destinations	Enter the destination ID of the collector for the mirroring session. The destination IP address ID can only be an IPv4 address within the network or a remote IPv4 address not managed by NSX-T Data Center. You can add up to three destination IP addresses separated by a comma.

4 Click **Save**.

Results

A custom port mirroring switching profile appears as a link.

What to do next

Attach the switching profile to a logical switch or logical port. See Associate a Custom Profile with a Logical Switch or Associate a Custom Profile with a Logical Port.

Verify that the customized port mirroring switching profile works. See Verify Custom Port Mirroring Switching Profile.

Verify Custom Port Mirroring Switching Profile

Before you start using the custom port mirroring switching profile, verify that the customization works properly.

Prerequisites

- Verify that the custom port mirroring switching profile is configured. See Configure a Custom Port Mirroring Switching Profile.
- Verify that the customized port mirroring switching profile is attached to a logical switch. See Associate a Custom Profile with a Logical Switch.

Procedure

1 Locate two VMs with VIF attachments to the logical port configured for port mirroring.

For example, VM1 10.70.1.1 and VM2 10.70.1.2 have VIF attachments and they are located in the same logical network.

2. Run the `tcpdump` command on a destination IP address.

   ```
   sudo tcpdump -n -i eth0 dst host destination_IP_addres and proto gre
   ```

   For example, the destination IP address is 10.24.123.196.

3. Log in to the first VM and ping the second VM to verify that the corresponding ECHO requests and replies are received at the destination address.

What to do next

Attach this port mirroring customized switching profile to a logical switch so that the modified parameters in the switching profile are applied to the network traffic. See Associate a Custom Profile with a Logical Switch.

Understanding IP Discovery Switching Profile

IP Discovery uses DHCP and DHCPv6 snooping, ARP (Address Resolution Protocol) snooping, ND (Neighbor Discovery) snooping, and VM Tools to learn MAC and IP addresses.

The discovered MAC and IP addresses are used to achieve ARP/ND suppression, which minimizes traffic between VMs connected to the same logical switch. The addresses are also used by the SpoofGuard and distributed firewall (DFW) components. DFW uses the address bindings to determine the IP address of objects in firewall rules.

DHCP/DHCPv6 snooping inspects the DHCP/DHCPv6 packets exchanged between the DHCP/DHCPv6 client and server to learn the IP and MAC addresses.

ARP snooping inspects the outgoing ARP and GARP (gratuitous ARP) packets of a VM to learn the IP and MAC addresses.

VM Tools is software that runs on an ESXi-hosted VM and can provide the VM's configuration information including MAC and IP or IPv6 addresses. This IP discovery method is available for VMs running on ESXi hosts only.

ND snooping is the IPv6 equivalent of ARP snooping. It inspects neighbor solicitation (NS) and neighbor advertisement (NA) messages to learn the IP and MAC addresses.

Duplicate address detection checks whether a newly discovered IP address is already present on the realized binding list for a different port. This check is performed for ports on the same logical switch. If a duplicate address is detected, the newly discovered address is not added to the realized binding list but is added to the discovered list. All duplicate IPs have an associated discovery timestamp. If the IP that is on the realized binding list is removed, either by adding it to the ignore binding list (see below) or by disabling snooping, the duplicate IP with the oldest timestamp is moved to the realized binding list. The duplicate address information is available through an API call.

By default, the discovery methods ARP snooping and ND snooping operate in a mode called trust on first use (TOFU). In TOFU mode, when an address is discovered and added to the realized bindings list, that binding remains in the realized list forever. TOFU applies to the first 'n' unique <IP, MAC, VLAN> bindings discovered using ARP/ND snooping, where 'n' is the binding limit that you can configure. You can disable TOFU for ARP/ND snooping. The methods will then operate in trust on every use (TOEU) mode. In TOEU mode, when an address is discovered, it is added to the realized bindings list and when it is deleted or expired, it is removed from the realized bindings list. DHCP snooping and VM Tools always operate in TOEU mode

Note TOFU does not mean SpoofGuard and it does not block traffic like SpoofGuard does. For more information about SpoofGuard, see Understanding SpoofGuard.

For each port, NSX Manager maintains an ignore bindings list, which contains IP addresses that cannot be bound to the port. You can only update this list using the API. You can also use this method to delete a previously discovered IP for a given port. For more information, see the NSX-T API Reference and search for `ignore_address_bindings`.

Note For Linux VMs, the ARP flux problem might cause ARP snooping to obtain incorrect information. The problem can be prevented with an ARP filter. For more information, see http://linux-ip.net/html/ether-arp.html#ether-arp-flux.

Configure IP Discovery Switching Profile

NSX-T Data Center has several default IP Discovery switching profiles. You can also create additional ones.

Prerequisites

Familiarize yourself with the IP Discovery switching profile concepts. See Understanding IP Discovery Switching Profile

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Switching > Switching Profiles > Add**.

3 Select **IP Discovering** and specify the IP Discovery switching profile details.

Option	Description
Name and Description	Enter a name and optionally a description.
ARP Snooping	For an IPv4 environment. Applicable if VMs have static IP addresses.
ARP Binding Limit	The maximum number of IPv4 IP addresses that can be bound to a port.
ARP ND Binding Limit Timeout	The timeout value, in minutes, for IP addresses in the ARP/ND binding table if TOFU is disabled. If an address times out, a newly discovered address will replace it.
DHCP Snooping	For an IPv4 environment. Applicable if VMs have IPv4 addresses.
DHCP V6 Snooping	For an IPv6 environment. Applicable if VMs have IPv6 addresses.
VM Tools	Available for ESXi-hosted VMs only.

Option	Description
VM Tools for IPv6	Available for ESXi-hosted VMs only.
Neighbor Discovery Snooping	For an IPv6 environment. Applicable if VMs have static IP addresses.
Neighbor Discovery Binding Limit	The maximum number of IPv6 addresses that can be bound to a port.
Trust on First use	Applicable to ARP and ND snooping.
Duplicate IP Detection	For all snooping methods and both IPv4 and IPv6 environments.

4 Click **Add**.

What to do next

Attach this IP Discovery customized switching profile to a logical switch or logical port so that the modified parameters in the switching profile are applied to the network traffic. See Associate a Custom Profile with a Logical Switch or Associate a Custom Profile with a Logical Port.

Understanding SpoofGuard

SpoofGuard helps prevent a form of malicious attack called "web spoofing" or "phishing." A SpoofGuard policy blocks traffic determined to be spoofed.

SpoofGuard is a tool that is designed to prevent virtual machines in your environment from sending traffic with an IP address it is not authorized to end traffic from. In the instance that a virtual machine’s IP address does not match the IP address on the corresponding logical port and switch address binding in SpoofGuard, the virtual machine’s vNIC is prevented from accessing the network entirely. SpoofGuard can be configured at the port or switch level. There are several reasons SpoofGuard might be used in your environment:

- Preventing a rogue virtual machine from assuming the IP address of an existing VM.
- Ensuring the IP addresses of virtual machines cannot be altered without intervention – in some environments, it’s preferable that virtual machines cannot alter their IP addresses without proper change control review. SpoofGuard facilitates this by ensuring that the virtual machine owner cannot simply alter the IP address and continue working unimpeded.
- Guaranteeing that distributed firewall (DFW) rules will not be inadvertently (or deliberately) bypassed – for DFW rules created utilizing IP sets as sources or destinations, the possibility always exists that a virtual machine could have it’s IP address forged in the packet header, thereby bypassing the rules in question.

NSX-T Data Center SpoofGuard configuration covers the following:

- MAC SpoofGuard - authenticates MAC address of packet
- IP SpoofGuard - authenticates MAC and IP addresses of packet
- Dynamic Address Resolution Protocol (ARP) inspection, that is, ARP and Gratuitous Address Resolution Protocol (GARP) SpoofGuard and Neighbor Discovery (ND) SpoofGuard validation are all against the MAC source, IP Source and IP-MAC source mapping in the ARP/GARP/ND payload.

At the port level, the allowed MAC/VLAN/IP whitelist is provided through the Address Bindings property of the port. When the virtual machine sends traffic, it is dropped if its IP/MAC/VLAN does not match the IP/MAC/VLAN properties of the port. The port level SpoofGuard deals with traffic authentication, i.e. is the traffic consistent with VIF configuration.

At the switch level, the allowed MAC/VLAN/IP whitelist is provided through the Address Bindings property of the switch. This is typically an allowed IP range/subnet for the switch and the switch level SpoofGuard deals with traffic authorization.

Traffic must be permitted by port level AND switch level SpoofGuard before it will be allowed into switch. Enabling or disabling port and switch level SpoofGuard, can be controlled using the SpoofGuard switch profile.

Configure Port Address Bindings

Address bindings specify the IP and MAC address of a logical port and are used to specify the port whitelist in SpoofGuard.

With port address bindings you'll specify the IP and MAC address, and VLAN if applicable, of the logical port. When SpoofGuard is enabled, it ensures that the specified address bindings are enforced in the data path. In addition to SpoofGuard, port address bindings are used for DFW rule translations.

Procedure

1. In NSX Manager, select to**Advanced Networking & Security > Networking > Switching > Ports**.
2. Click the logical port to which you want apply address binding.

   The logical port summary appears.
3. In the **Overview** tab, expand **Address Bindings**.
4. Click **Add**.

   The Add Address Binding dialogue box appears
5. Specify the IP and MAC address of the logical port to which you want to apply address binding. You can also specify a VLAN ID.
6. Click **Add**.

What to do next

Use the port address bindings when you Configure a SpoofGuard Switching Profile.

Configure a SpoofGuard Switching Profile

When SpoofGuard is configured, if the IP address of a virtual machine changes, traffic from the virtual machine may be blocked until the corresponding configured port/switch address bindings are updated with the new IP address.

Enable SpoofGuard for the port group(s) containing the guests. When enabled for each network adapter, SpoofGuard inspects packets for the prescribed MAC and its corresponding IP address.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Switching > Switching Profiles > Add**.

3 Select **Spoof Guard**.

4 Enter a name and optionally a description.

5 To enable port level SpoofGuard, set **Port Bindings** to **Enabled**.

6 Click **Add**.

Results

A new switching profile has been created with a SpoofGuard Profile.

What to do next

Associate the SpoofGuard profile with a logical switch or logical port. See Associate a Custom Profile with a Logical Switch or Associate a Custom Profile with a Logical Port.

Understanding Switch Security Switching Profile

Switch security provides stateless Layer2 and Layer 3 security by checking the ingress traffic to the logical switch and dropping unauthorized packets sent from VMs by matching the IP address, MAC address, and protocols to a set of allowed addresses and protocols. You can use switch security to protect the logical switch integrity by filtering out malicious attacks from the VMs in the network.

You can configure the Bridge Protocol Data Unit (BPDU) filter, DHCP Snooping, DHCP server block, and rate limiting options to customize the switch security switching profile on a logical switch.

Configure a Custom Switch Security Switching Profile

You can create a custom switch security switching profile with MAC destination addresses from the allowed BPDU list and configure rate limiting.

Prerequisites

Familiarize yourself with the switch security switching profile concept. See Understanding Switch Security Switching Profile.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Switching**.

3 Click the **Switching Profiles** tab.

4 Click **Add** and select **Switch Security**.

5 Complete the switch security profile details.

Option	Description
Name and Description	Assign a name to the custom switch security profile. You can optionally describe the setting that you modified in the profile.
BPDU Filter	Toggle the **BPDU Filter** button to enable BPDU filtering. Disabled by default. When the BPDU filter is enabled, all of the traffic to BPDU destination MAC address is blocked. The BPDU filter when enabled also disables STP on the logical switch ports because these ports are not expected to take part in STP.
BPDU Filter Allow List	Click the destination MAC address from the BPDU destination MAC addresses list to allow traffic to the permitted destination. You must enable **BPDU Filter** to be able to select from this list.
DHCP Filter	Toggle the **Server Block** button and **Client Block** button to enable DHCP filtering. Both are disabled by default. DHCP Server Block blocks traffic from a DHCP server to a DHCP client. Note that it does not block traffic from a DHCP server to a DHCP relay agent. DHCP Client Block prevents a VM from acquiring a DHCP IP address by blocking DHCP requests.
DHCPv6 Filter	Toggle the **V6 Server Block** button and **V6 Client Block** button to enable DHCP filtering. Both are disabled by default. DHCPv6 Server Block blocks traffic from a DHCPv6 server to a DHCPv6 client. Note that it does not block traffic from a DHCP server to a DHCP relay agent. Packets whose UDP source port number is 547 are filtered. DHCPv6 Client Block prevents a VM from acquiring a DHCP IP address by blocking DHCP requests. Packets whose UDP source port number is 546 are filtered.
Block Non-IP Traffic	Toggle the **Block Non-IP Traffic** button to allow only IPv4, IPv6, ARP, and BPDU traffic. The rest of the non-IP traffic is blocked. The permitted IPv4, IPv6, ARP, GARP and BPDU traffic is based on other policies set in address binding and SpoofGuard configuration. By default, this option is disabled to allow non-IP traffic to be handled as regular traffic.
RA Guard	Toggle the **RA Guard** button to filter out ingress IPv6 router advertisements. ICMPv6 type 134 packets are filtered out. This option is enabled by default.
Rate Limits	Set a rate limit for broadcast and multicast traffic. This option is enabled by default. Rate limits can be used to protect the logical switch or VMs from events such as broadcast storms. To avoid any connectivity problems, the minimum rate limit value must be >= 10 pps.

6 Click **Add**.

Results

A custom switch security profile appears as a link.

What to do next

Attach this switch security customized switching profile to a logical switch or logical port so that the modified parameters in the switching profile are applied to the network traffic. See Associate a Custom Profile with a Logical Switch or Associate a Custom Profile with a Logical Port.

Understanding MAC Management Switching Profile

The MAC management switching profile supports two functionalities: MAC learning and MAC address change.

The MAC address change feature allows a VM to change its MAC address. A VM connected to a port can run an administrative command to change the MAC address of its vNIC and still send and receive traffic on that vNIC. This feature is supported on ESXi only and not on KVM. This property is disabled by default, except when the guest VM is deployed using VMware Integrated OpenStack, in which case the property is enabled by default.

MAC learning provides network connectivity to deployments where multiple MAC addresses are configured behind one vNIC, for example, in a nested hypervisor deployment where an ESXi VM runs on an ESXi host and multiple VMs run inside the ESXi VM. Without MAC learning, when the ESXi VM's vNIC connects to a switch port, its MAC address is static. VMs running inside the ESXi VM do not have network connectivity because their packets have different source MAC addresses. With MAC learning, the vSwitch inspects the source MAC address of every packet coming from the vNIC, learns the MAC address and allows the packet to go through. If a MAC address that is learned is not used for a certain period of time, it is removed. This aging property is not configurable.

MAC learning also supports unknown unicast flooding. Normally, when a packet that is received by a port has an unknown destination MAC address, the packet is dropped. With unknown unicast flooding enabled, the port floods unknown unicast traffic to every port on the switch that has MAC learning and unknown unicast flooding enabled. This property is enabled by default, but only if MAC learning is enabled.

The number of MAC addresses that can be learned is configurable. The maximum value is 4096, which is the default. You can also set the policy for when the limit is reached. The options are:

- **Drop** - Packets from an unknown source MAC address are dropped. Packets inbound to this MAC address will be treated as unknown unicast. The port will receive the packets only if it has unknown unicast flooding enabled.
- **Allow** - Packets from an unknown source MAC address are forwarded although the address will not be learned. Packets inbound to this MAC address will be treated as unknown unicast. The port will receive the packets only if it has unknown unicast flooding enabled.

If you enable MAC learning or MAC address change, to improve security, configure SpoofGuard as well.

Configure MAC Management Switching Profile

You can create a MAC management switching profile to manage MAC addresses.

Prerequisites

Familiarize yourself with the MAC management switching profile concept. See Understanding MAC Management Switching Profile.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Switching > Switching Profiles > Add**.
3. Select **MAC Management** and complete the MAC management profile details.

Option	Description
Name and Description	Assign a name to the MAC management profile. You can optionally describe the setting that you modified in the profile.
MAC Change	Enable or disable the MAC address change feature. The default is disabled.
Status	Enable or disable the MAC learning feature. The default is disabled.
Unknown Unicast Flooding	Enable or disable the unknown unicast flooding feature. The default is enabled. This option is available if you enable MAC learning
MAC Limit	Set the maximum number of MAC addresses. The default is 4096. This option is available if you enable MAC learning
MAC Limit Policy	Select **Allow** or **Drop**. The default is **Allow**. This option is available if you enable MAC learning

4. Click **Add**.

What to do next

Attach the switching profile to a logical switch or logical port. See Associate a Custom Profile with a Logical Switch or Associate a Custom Profile with a Logical Port.

Associate a Custom Profile with a Logical Switch

You can associate a custom switching profile to a logical switch so that the profile applies to all the ports on the switch.

When custom switching profiles are attached to a logical switch they override existing default switching profiles. The custom switching profile is inherited by children logical switch ports.

Note If you have associated a custom switching profile with a logical switch, but want to retain the default switching profile for one of the child logical switch port, then you must make a copy of the default switching profile and associate it with the specific logical switch port.

Prerequisites

- Verify that a logical switch is configured. See Create a Logical Switch.
- Verify that a custom switching profile is configured. See Switching Profiles for Logical Switches and Logical Ports.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Switching > Switches**.
3. Click the logical switch to apply the custom switching profile.
4. Click the **Manage** tab.
5. Select the custom switching profile type from the drop-down menu.
   - **QoS**
   - **Port Mirroring**
   - **IP Discovering**
   - **SpoofGuard**
   - **Switch Security**
   - **MAC Management**
6. Click **Change.**
7. Select the previously created custom switching profile from the drop-down menu.
8. Click **Save**.

   The logical switch is now associated with the custom switching profile.
9. Verify that the new custom switching profile with the modified configuration appears under the **Manage** tab.
10. (Optional) Click the **Related** tab and select **Ports** from the drop-down menu to verify that the custom switching profile is applied to child logical ports.

What to do next

If you do not want to use the inherited switching profile from a logical switch, you can apply a custom switching profile to the child logical switch port. See Associate a Custom Profile with a Logical Port.

Associate a Custom Profile with a Logical Port

A logical port provides a logical connection point for a VIF, a patch connection to a router, or a Layer 2 gateway connection to an external network. Logical ports also expose switching profiles, port statistics counters, and a logical link status.

You can change the inherited switching profile from the logical switch to a different custom switching profile for the child logical port.

Prerequisites

- Verify that a logical port is configured. See Connecting a VM to a Logical Switch.
- Verify that a custom switching profile is configured. See Switching Profiles for Logical Switches and Logical Ports.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Switching > Ports**.
3. Click the logical port to apply the custom switching profile.
4. Click the **Manage** tab.
5. Select the custom switching profile type from the drop-down menu.
    - **QoS**
    - **Port Mirroring**
    - **IP Discovering**
    - **SpoofGuard**
    - **Switch Security**
    - **MAC Management**
6. Click **Change**.
7. Select the previously created custom switching profile from the drop-down menu.
8. Click **Save**.

    The logical port is now associated with the custom switching profile.
9. Verify that the new custom switching profile with the modified configuration appears under the **Manage** tab.

What to do next

You can monitor the activity on the logical switch port to troubleshoot problems. See "Monitor a Logical Switch Port Activity" in the *NSX-T Data Center Administration Guide*.

Layer 2 Bridging

When an NSX-T Data Center logical switch requires a Layer 2 connection to a VLAN-backed port group or needs to reach another device, such as a gateway, that resides outside of an NSX-T Data Center deployment, you can use an NSX-T Data Center Layer 2 bridge. This Layer 2 bridge is especially useful in a migration scenario, in which you need to split a subnet across physical and virtual workloads.

The NSX-T Data Center concepts involved in Layer 2 bridging are Edge Clusters and Edge Bridge profiles. You can configure layer 2 bridging using NSX Edge transport nodes. To use NSX Edge transport nodes for bridging, you create an Edge bridge profile. An Edge Bridge profile specifies which Edge Cluster to use for bridging and which Edge Transport node acts as the primary and backup bridge.

The Edge Bridge Profile is attached to a logical switch and the mapping specifies the physical uplink on the Edge used for bridging and the VLAN ID to be associated with the logical switch. A logical switch can be attached to several bridge profiles.

Create an ESXi Bridge Cluster

An ESXi bridge cluster is a collection of ESXi host transport nodes that can provide layer 2 bridging to a logical switch.

An ESXi bridge cluster can have a maximum of two ESXi host transport nodes as bridge nodes. With two bridge nodes, an ESXi bridge cluster will provide high availability in an active-standby mode. Even if you want to have one bridge node, you still must create a bridge cluster. After creating the bridge cluster, you can add an additional bridge node later.

Prerequisites

- Create at least one NSX-T Data Center transport node for use as a bridge node.
- The transport node used as a bridge node must be an ESXi host. KVM is not supported for bridge nodes.
- It is recommended that bridge nodes not have any hosted VMs.
- A transport node can be added to only one bridge cluster. You cannot add the same transport node to multiple bridge clusters.

Procedure

1. Select **System > Fabric > Nodes > ESXi Bridge Clusters > Add**.
2. Enter a name for the bridge cluster and optionally a description.
3. Select a transport zone for the bridge cluster.

4 From the **Available** column, select transport nodes and click the right arrow to move them to the **Selected** column.

5 Click the **Add** button.

What to do next

You can now associate a logical switch with the bridge cluster.

Create an Edge Bridge Profile

An Edge bridge profile makes an NSX Edge cluster capable of providing layer 2 bridging to a logical switch.

Prerequisites

- Verify that you have an NSX Edge cluster with two NSX Edge transport nodes.

Procedure

1 Select **System > Fabric > Profiles > Edge Bridge Profiles > Add**.

2 Enter a name for the Edge bridge profile and optionally a description.

3 Select an NSX Edge cluster.

4 Select a primary node.

5 Select a backup node.

6 Select a failover mode.

   The options are **Preemptive** and **Non-Preemptive**.

7 Click the **Add** button.

What to do next

You can now associate a logical switch with the bridge profile.

Configure Edge-Based Bridging

When you configure edge-based bridging, after creating an edge brige profile for an edge cluster, some additonal configurations are required.

Note that bridging a logical switch twice on the same Edge node is not supported. However, you can bridge two VLANs to the same logical switch on two different Edge nodes.

There are three configuration options.

Option 1: Configure Promiscuous Mode

- Set promiscuous mode on the portgroup.
- Allow forged transmit on the portgroup.

- Run the following command to enable reverse filter on the ESXi host where the Edge VM is running:

  ```
  esxcli system settings advanced set -o /Net/ReversePathFwdCheckPromisc -i 1
  ```

  Then disable and enable promiscuous mode on the portgroup with the following steps:

  - Edit the portgroup's settings.
  - Disable promiscuous mode and save the settings.
  - Edit the portgroup's settings again.
  - Enable promiscuous mode and save the settings.
- Do not have other port groups in promiscuous mode on the same host sharing the same set of VLANs.
- The active and standby Edge VMs should be on different hosts. If they are on the same host the throughput might be reduced because VLAN traffic needs to be forwarded to both VMs in promiscuous mode.

Option 2: Configure MAC Learning

If the Edge is deployed on a host with NSX-T installed, it can connect to a VLAN logical switch or segment. The logical switch must have a MAC Management profile with MAC Learning enabled. Similarly, the segment must have a MAC Discovery profile with MAC Learning enabled.

Option 3: Configure a Sink Port

1. Retrieve the port number for the trunk vNIC that you want to configure as a sink port.
   1. Log in to the vSphere Web Client, and navigate to **Home > Networking**.
   2. Click the distributed port group to which the NSX Edge trunk interface is connected, and click **Ports** to view the ports and connected VMs. Note the port number associated with the trunk interface. Use this port number when fetching and updating opaque data.
2. Retrieve the dvsUuid value for the vSphere Distributed Switch.
   1. Log in to the vCenter Mob UI at `https://<vc-ip>/mob` .
   2. Click **content**.
   3. Click the link associated with the **rootFolder** (for example: *group-d1 (Datacenters)*).
   4. Click the link associated with the **childEntity** (for example: *datacenter-1*).
   5. Click the link associated with the **networkFolder** (for example: *group-n6*).
   6. Click the DVS name link for the vSphere distributed switch associated with the NSX Edges (for example: *dvs-1 (Mgmt_VDS)*).
   7. Copy the value of the uuid string. Use this value for dvsUuid when fetching and updating opaque data.

3 Verify if opaque data exists for the specified port.

   a Go to `https://<vc-ip>/mob/?moid=DVSManager&vmodl=1`.

   b Click **fetchOpaqueDataEx**.

   c In the **selectionSet** value box paste the following XML input:

```
<selectionSet xsi:type="DVPortSelection">
    <dvsUuid>c2 1d 11 50 6a 7c 77 68-e6 ba ce 6a 1d 96 2a 15</dvsUuid> <!-- example dvsUuid --
>
    <portKey>393</portKey>  <!-- example port number -->
</selectionSet>
```

   Use the port number and dvsUuid value that you retrieved for the NSX Edge trunk interface.

   d Set `isRuntime` to `false`.

   e Click **Invoke Method**. If the result shows values for `vim.dvs.OpaqueData.ConfigInfo`, then there is already opaque data set, use the `edit` operation when you set the sink port. If the value for `vim.dvs.OpaqueData.ConfigInfo` is empty, use the `add` operation when you set the sink port.

4 Configure the sink port in the vCenter managed object browser (MOB).

   a Go to `https://<vc-ip>/mob/?moid=DVSManager&vmodl=1`.

   b Click **updateOpaqueDataEx**.

   c In the **selectionSet** value box paste the following XML input. For example,

```
<selectionSet xsi:type="DVPortSelection">
    <dvsUuid>c2 1d 11 50 6a 7c 77 68-e6 ba ce 6a 1d 96 2a 15</dvsUuid> <!-- example dvsUuid --
>
    <portKey>393</portKey>  <!-- example port number -->
</selectionSet>
```

   Use the dvsUuid value that you retrieved from the vCenter MOB.

   d On the opaqueDataSpec value box paste one of the following XML inputs.

   Use this input to enable a SINK port if opaque data is not set (`operation` is set to `add`):

```
<opaqueDataSpec>
    <operation>add</operation>
    <opaqueData>
        <key>com.vmware.etherswitch.port.extraEthFRP</key>
        <opaqueData
xsi:type="vmodl.Binary">AAABAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAA=</opaqueData>
    </opaqueData>
</opaqueDataSpec>
```

Use this input to enable a SINK port if opaque data is already set (`operation` is set to `edit`):

```
<opaqueDataSpec>
    <operation>edit</operation>
    <opaqueData>
        <key>com.vmware.etherswitch.port.extraEthFRP</key>
        <opaqueData
xsi:type="vmodl.Binary">AAABAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAA=</opaqueData>
    </opaqueData>
</opaqueDataSpec>
```

Use this input to disable a SINK port:

```
<opaqueDataSpec>
    <operation>edit</operation>
        <opaqueData>
            <key>com.vmware.etherswitch.port.extraEthFRP</key>
            <opaqueData
xsi:type="vmodl.Binary">AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAA=</opaqueData>
    </opaqueData>
</opaqueDataSpec>
```

e Set `isRuntime` to `false`.

f Click **Invoke Method**.

Create a Layer 2 Bridge-Backed Logical Switch

When you have VMs that are connected to the NSX-T Data Center overlay, you can configure a bridge-backed logical switch to provide layer 2 connectivity with other devices or VMs that are outside of your NSX-T Data Center deployment.

Prerequisites

- Verify that you have a bridge cluster or a bridge profile.
- At least one ESXi or KVM host to serve as a regular transport node. This node has hosted VMs that require connectivity with devices outside of a NSX-T Data Center deployment.
- A VM or another end device outside of the NSX-T Data Center deployment. This end device must be attached to a VLAN port matching the VLAN ID of the bridge-backed logical switch.
- One logical switch in an overlay transport zone to serve as the bridge-backed logical switch.

Procedure

1 From a browser, log in to an NSX Manager at `https://<nsx-mgr>`.

2 Select **Advanced Networking & Security > Networking > Switching**.

3. Click the name of an overlay switch (traffic type: overlay).
4. Click **Related > ESXi Bridge Clusters** or **Related > Edge Bridge Profiles**.
5. Click **Attach**.
6. To attach to a bridge cluster,
   - a Select a bridge cluster.
   - b Enter a VLAN ID.
   - c Enable or disable **HA on VLAN**.
   - d Click **Attach**.
7. To attach to a bridge profile,
   - a Select a bridge profile.
   - b Select a transport zone.
   - c Enter a VLAN ID.
   - d Click **Save**.
8. Connect VMs to the logical switch if they are not already connected.

   The VMs must be on transport nodes in the same transport zone as the bridge cluster or bridge profile.

Results

You can test the functionality of the bridge by sending a ping from the NSX-T Data Center-internal VM to a node that is external to NSX-T Data Center.

You can monitor traffic on the bridge switch by clicking the **Monitor** tab.

You can also view the bridge traffic with the `GET https://192.168.110.31/api/v1/bridge-endpoints/<endpoint-id>/statistics` API call:

```
{
  "tx_packets": {
    "total": 134416,
    "dropped": 0,
    "multicast_broadcast": 0
  },
  "rx_bytes": {
    "total": 22164,
    "multicast_broadcast": 0
  },
  "tx_bytes": {
    "total": 8610134,
    "multicast_broadcast": 0
  },
  "rx_packets": {
    "total": 230,
    "dropped": 0,
    "multicast_broadcast": 0
  },
  "last_update_timestamp": 1454979822860,
  "endpoint_id": "ba5ba59d-22f1-4a02-b6a0-18ef0e37ef31"
}
```

Logical Routers

14

NSX-T Data Center supports a 2-tier routing model.

In the top tier is the tier-0 logical router. Northbound, the tier-0 logical router connects to one or more physical routers or layer 3 switches and serves as a gateway to the physical infrastructure. Southbound, the tier-0 logical router connects to one or more tier-1 logical routers or directly to one or more logical switches.

In the bottom tier is the tier-1 logical router. Northbound, the tier-1 logical router connects to a tier-0 logical router. Southbound, it connects to one or more logical switches.

Note If you use the **Advanced Networking & Security** user interface to modify objects created in the policy interface, some settings might not be configurable. These read-only settings have this icon next to them: ⊖. See Chapter 1 Overview of the NSX Manager for more information.

This chapter includes the following topics:

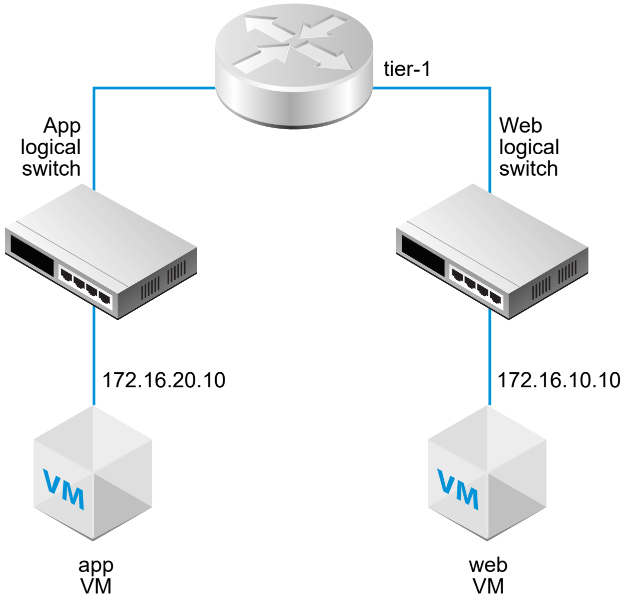
- Tier-1 Logical Router
- Tier-0 Logical Router

Tier-1 Logical Router

Tier-1 logical routers have downlink ports to connect to logical switches and uplink ports to connect to tier-0 logical routers.

When you add a logical router, it is important that you plan the networking topology you are building.

Figure 14-1. Tier-1 Logical Router Topology



For example, this simple topology shows two logical switches connected to a tier-1 logical router. Each logical switch has a single VM connected. The two VMs can be on different hosts or the same host, in different host clusters or in the same host cluster. If a logical router does not separate the VMs, the underlying IP addresses configured on the VMs must be in the same subnet. If a logical router does separate them, the IP addresses on the VMs must be in different subnets.

In some scenarios, external clients send ARP queries for MAC addresses bound to LB VIP ports. However, LB VIP ports do not have MAC addresses and cannot handle such queries. Proxy ARP is implemented on the centralized service ports of a tier-1 logical router to handle ARP queries on behalf of the LB VIP ports.

When a tier-1 logical router is configured with DNAT, Edge firewall, and load balancer, traffic to and from another tier-1 logical router is processed in this order: DNAT first, then Edge firewall, and then load balancer. Traffic within the tier-1 logical router is processed through DNAT first and then load balancer. Edge firewall processing is skipped.

On a tier-0 or tier-1 logical router, you can configure different types of ports. One type is called centralized service port (CSP). You must configure a CSP on a tier-0 logical router in active-standby mode or a tier-1 logical router to connect to a VLAN-backed logical switch, or to create a standalone tier-1 logical router. A CSP supports the following services on a tier-0 logical router in active-standby mode or a tier-1 logical router:

- NAT
- Load balancing
- Stateful firewall
- VPN (IPsec and L2VPN)

Create a Tier-1 Logical Router

The tier-1 logical router must be connected to the tier-0 logical router to get the northbound physical router access.

Prerequisites

- Verify that the logical switches are configured. See Create a Logical Switch.
- Verify that an NSX Edge cluster is deployed to perform network address translation (NAT) configuration. See the *NSX-T Data Center Installation Guide*.
- Familiarize yourself with the tier-1 logical router topology. See Tier-1 Logical Router.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Routers > Routers > Add**.
3. Select **Tier-1 Router** and enter a name for the logical router and optionally a description.
4. (Optional) Select a tier-0 logical router to connect to this tier-1 logical router.

   If you do not yet have any tier-0 logical routers configured, you can leave this field blank for now and edit the router configuration later.
5. (Optional) Select an NSX Edge cluster.

   To deselect a cluster that you selected, click the **x** icon. If the tier-1 logical router is going to be used for NAT configuration, it must be connected to an NSX Edge cluster. If you do not yet have any NSX Edge clusters configured, you can leave this field blank for now and edit the router configuration later.
6. (Optional) Click the **StandBy Relocation** toggle to enable or disable standby relocation.

   Standby relocation means that if the Edge node where the active or standby logical router is running fails, a new standby logical router is created on another Edge node to maintain high availability. If the Edge node that fails is running the active logical router, the original standby logical router becomes the active logical router and a new standby logical router is created. If the Edge node that fails is running the standby logical router, the new standby logical router replaces it.
7. (Optional) If you selected an NSX Edge cluster, select a failover mode.

Option	Description
Preemptive	If the preferred node fails and recovers, it will preempt its peer and become the active node. The peer will change its state to standby. This is the default option.
Non-preemptive	If the preferred node fails and recovers, it will check if its peer is the active node. If so, the preferred node will not preempt its peer and will be the standby node.

8. (Optional) Click the **Advanced** tab and enter a value for **Intra Tier-1 Transit Subnet**.

9 Click **Add**.

Results

After the logical router is created, if you want to remove the Edge cluster from the router's configuration, perform the following steps:

- Click the name of the router to see the configuration details.
- Select **Services > Edge Firewall**.
- Click **Disable Firewall**.
- Click the **Overview** tab and click **Edit**.
- In the **Edge Cluster** field, click the **x** icon.
- Click **Save**.

If this logical router supports more than 5000 VMs, you must run the following commands on each node of the NSX Edge cluster to increase the size of the ARP table.

```
set debug-mode
set dataplane neighbor max-arp-logical-router 10000
```

You must re-run the commands after a dataplane restart or a node reboot because the change is not persistent.

What to do next

Create downlink ports for your tier-1 logical router. See Add a Downlink Port on a Tier-1 Logical Router.

Add a Downlink Port on a Tier-1 Logical Router

When you create a downlink port on a tier-1 logical router, the port serves as a default gateway for the VMs that are in the same subnet.

Prerequisites

Verify that a tier-1 logical router is configured. See Create a Tier-1 Logical Router .

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Click the name of a router.

4 Click the **Configuration** tab and select **Router Ports**.

5 Click **Add**.

6 Enter a name for the router port and optionally a description.

7 In the **Type** field, select **Downlink**.

8 For **URPF Mode**, select **Strict** or **None**.

   URPF (unicast Reverse Path Forwarding) is a security feature.

9 (Optional) Select a logical switch.

10 Select whether this attachment creates a switch port or updates an existing switch port.

   If the attachment is for an existing switch port, select the port from the drop-down menu.

11 Enter the router port IP address in CIDR notation.

   For example, the IP address can be 172.16.10.1/24.

12 (Optional) Select a DHCP relay service.

13 Click **Add**.

What to do next

Enable route advertisement to provide North-South connectivity between VMs and external physical networks or between different tier-1 logical routers that are connected to the same tier-0 logical router. See Configure Route Advertisement on a Tier-1 Logical Router.

Add a VLAN Port on a Tier-0 or Tier-1 Logical Router

If you have only VLAN-backed logical switches, you can connect the switches to VLAN ports on a tier-0 or tier-1 router so that NSX-T Data Center can provide layer-3 services.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Click the name of a router.

4 Click the **Configuration** tab and select **Router Ports**.

5 Click **Add**.

6 Enter a name for the router port and optionally a description.

7 In the **Type** field, select **Centralized**.

8 For **URPF Mode**, select **Strict** or **None**.

   URPF (unicast Reverse Path Forwarding) is a security feature.

9 (Required) Select a logical switch.

10 Select whether this attachment creates a switch port or updates an existing switch port.

   If the attachment is for an existing switch port, select the port from the drop-down menu.

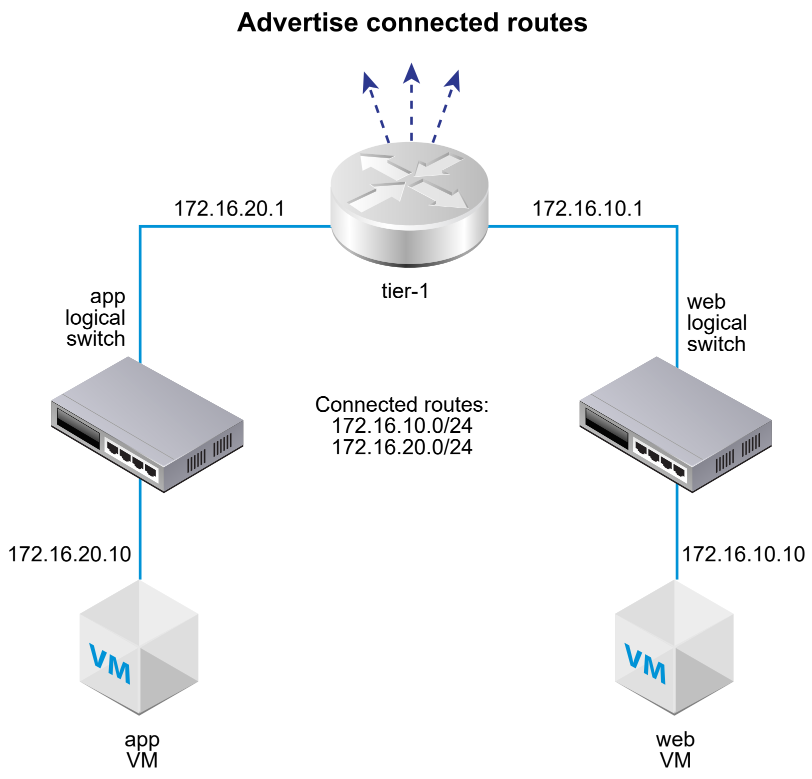
11 Enter the router port IP address in CIDR notation.

12 Click **Add**.

Configure Route Advertisement on a Tier-1 Logical Router

To provide Layer 3 connectivity between VMs connected to logical switches that are attached to different tier-1 logical routers, it is necessary to enable tier-1 route advertisement towards tier-0. You do not need to configure a routing protocol or static routes between tier-1 and tier-0 logical routers. NSX-T Data Center creates NSX-T Data Center static routes automatically when you enable route advertisement.

For example, to provide connectivity to and from the VMs through other peer routers, the tier-1 logical router must have route advertisement configured for connected routes. If you don't want to advertise all connected routes, you can specify which routes to advertise.



Prerequisites

- Verify that VMs are attached to logical switches. See Chapter 13 Logical Switches.
- Verify that downlink ports for the tier-1 logical router are configured. See Add a Downlink Port on a Tier-1 Logical Router.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Click the name of a tier-1 router.

4 Select **Route Advertisement** from the **Routing** drop-down menu.

5 Click **Edit** to edit the route advertisement configuration.

   You can toggle the following switches:

   - **Status**
   - **Advertise All NSX Connected Routes**
   - **Advertise All NAT Routes**
   - **Advertise All Static Routes**
   - **Advertise All LB VIP Routes**
   - **Advertise All LB SNAT IP Routes**
   - **Advertise All DNS Forwarder Routes**

   a Click **Save**.

6 Click **Add** to advertise routes.

   a Enter a name and optionally a description.

   b Enter a route prefix in CIDR format.

   c Click **Apply Filter** to set the following options:

Action	Specify **Allow** or **Deny**.
Match route types	Select one or more of the following: ■ **Any** ■ **NSX Connected** ■ **Tier-1 LB VIP** ■ **Static** ■ **Tier-1 NAT** ■ **Tier-1 LB SNAT**
Prefix operator	Select **GE** or **EQ**.

   d Click **Add**.

What to do next

Familiarize yourself with the tier-0 logical router topology and create the tier-0 logical router. See Tier-0 Logical Router.

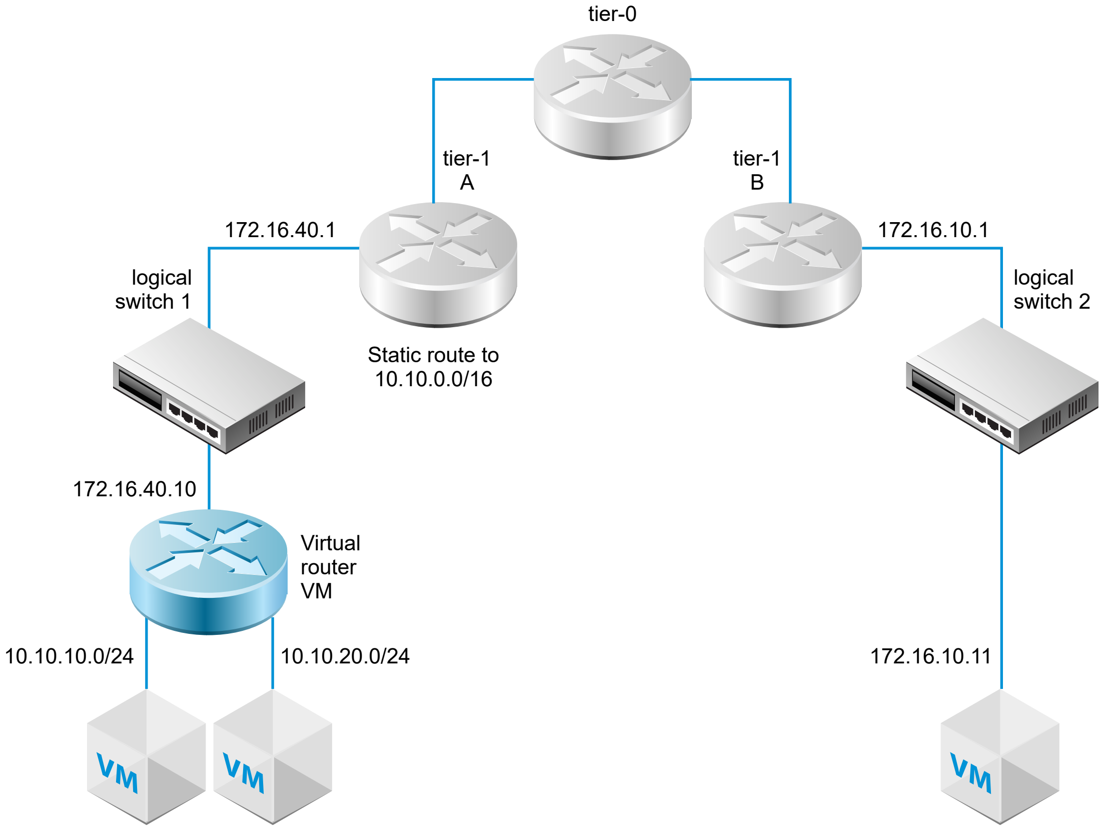
If you already have a tier-0 logical router connected to the tier-1 logical router, you can verify that the tier-0 router is learning the tier-1 router connected routes. See Verify that a Tier-0 Router Has Learned Routes from a Tier-1 Router.

Configure a Tier-1 Logical Router Static Route

You can configure a static route on a tier-1 logical router to provide connectivity from NSX-T Data Center to a set of networks that are accessible through a virtual router.

For example, in the following diagram, the tier-1 A logical router has a downlink port to an NSX-T Data Center logical switch. This downlink port (172.16.40.1) serves the default gateway for the virtual router VM. The virtual router VM and tier-1 A are connected through the same NSX-T Data Center logical switch. The tier-1 logical router has a static route 10.10.0.0/16 that summarizes the networks available through the virtual router. Tier-1 A then has route advertisement configured to advertise the static route to tier-1 B.

Figure 14-2. Tier-1 Logical Router Static Route Topology



Recursive static routes are supported.

Prerequisites

Verify that a downlink port is configured. See Add a Downlink Port on a Tier-1 Logical Router.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Click the name of a tier-1 router.

4 Click the **Routing** tab and select **Static Routes** from the drop-down menu.

5 Click **Add**.

6 Enter a network address in the CIDR format.

Static route based on IPv6 is supported. IPv6 prefixes can only have an IPv6 next hop.

For example, 10.10.10.0/16 or an IPv6 address.

7 Click **Add** to add a next-hop IP address.

For example, 172.16.40.10. You can also specify a null route by clicking the pencil icon and selecting **NULL** from the drop-down. To add another next hop addresses, click **Add** again.

8 Click **Add** at the bottom of the dialog box.

The newly created static route network address appears in the row.

9 From the tier-1 logical router, select **Routing > Route Advertisement**.

10 Click **Edit** and select **Advertise All Static Routes**.

11 Click **Save**.

The static route is propagated across the NSX-T Data Center overlay.

Create a Standalone Tier-1 Logical Router

A standalone tier-1 logical router has no downlink and no connection to a tier-0 router. It has a service router but no distributed router. The service router can be deployed on one NSX Edge node or two NSX Edge nodes in active-standby mode.

A standalone tier-1 logical router:

- Must not have a connection to a tier-0 logical router.
- Must not have a downlink.
- Can have only one centralized service port (CSP) if it is used to attach a load balancer (LB) service.
- Can connect to an overlay logical switch or a VLAN logical switch.
- Supports any combination of the services IPSec, DNAT, firewall, load balancer, and service insertion. For ingress, the order of processing is: IPSec – DNAT – firewall – load balancer - service insertion. For egress, the order of processing is: service insertion - load balancer - firewall - DNAT - IPSec.

Typically, a standalone tier-1 logical router is connected to a logical switch that a regular tier-1 logical router is also connected to. The standalone tier-1 logical router can communicate with other devices through the regular tier-1 logical router after static routes and route advertisements are configured.

Before using the standalone tier-1 logical router, note the following:

- To specify the default gateway for the standalone tier-1 logical router, you must add a static route. The subnet should be 0.0.0.0/0 and the next hop is the IP address of a regular tier-1 router connected to the same switch.
- ARP proxy on the standalone router is supported. You can configure an LB virtual server IP or LB SNAT IP in the CSP's subnet. For example, if the CSP IP is 1.1.1.1/24, the virtual IP can be 1.1.1.2. It can also be an IP in another subnet such as 2.2.2.2 if routing is properly configured so that traffic for 2.2.2.2 can reach the standalone router.
- For an NSX Edge VM, you cannot have more than one CSPs which are connected to the same VLAN-backed logical switch or different VLAN-backed logical switches that have the same VLAN ID.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Routers > Routers > Add**.
3. Select **Tier-1 Router** and enter a name for the logical router, and optionally a description.
4. (Required) Select an NSX Edge cluster to connect to this tier-1 logical router.
5. (Required) Select a failover mode and cluster members.

Option	Description
Preemptive	If the preferred node fails and recovers, it will preempt its peer and become the active node. The peer will change its state to standby. This is the default option.
Non-preemptive	If the preferred node fails and recovers, it will check if its peer is the active node. If so, the preferred node will not preempt its peer and will be the standby node.

6. Click **Add**.
7. Click the name of the router that you just created.
8. Click the **Configuration** tab and select **Router Ports**.
9. Click **Add**.
10. Enter a name for the router port and optionally a description.
11. In the **Type** field, select **Centralized**.
12. For **URPF Mode**, select **Strict** or **None**.

    URPF (Unicast Reverse Path Forwarding) is a security feature.
13. (Required) Select a logical switch.
14. Select whether this attachment creates a switch port or updates an existing switch port.
15. Enter the router port IP address in CIDR notation.

16 Click **Add**.

Tier-0 Logical Router

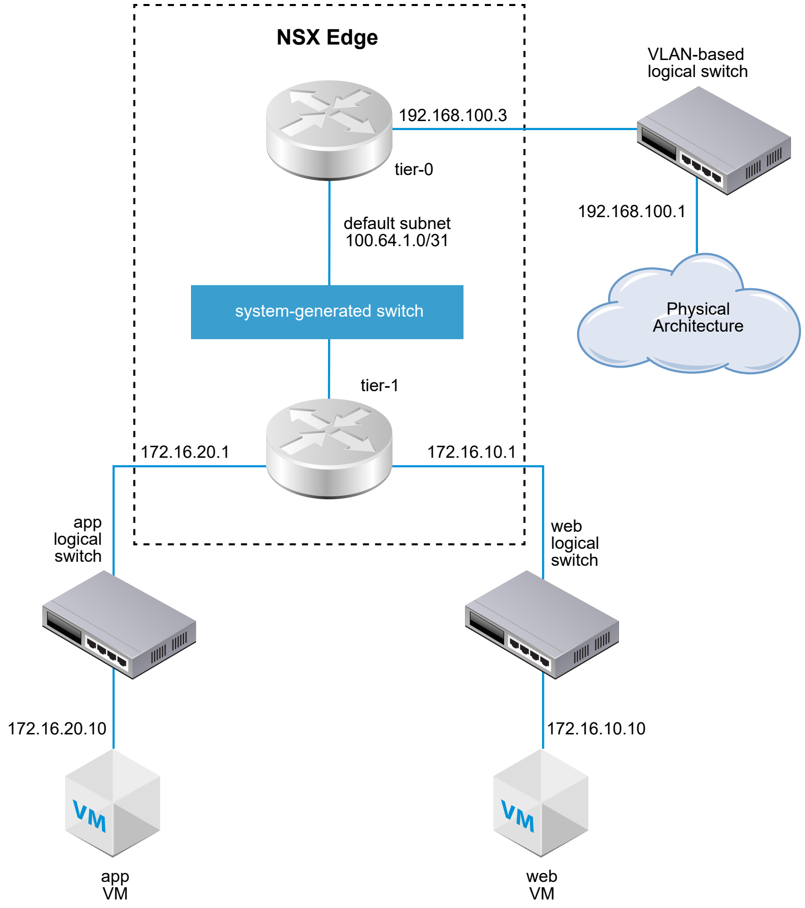
A tier-0 logical router provides a gateway service between the logical and physical network.

NSX Cloud Note If using NSX Cloud, see How to use NSX-T Data Center Features with the Public Cloud for a list of auto-generated logical entities, supported features, and configurations required for NSX Cloud.

An Edge node can support only one tier-0 gateway or logical router. When you create a tier-0 gateway or logical router, make sure you do not create more tier-0 gateways or logical routers than the number of Edge nodes in the NSX Edge cluster.

When you add a tier-0 logical router, it is important that you map out the networking topology you are building.

Figure 14-3. Tier-0 Logical Router Topology



For simplicity, the sample topology shows a single tier-1 logical router connected to a single tier-0 logical router hosted on a single NSX Edge node. Keep in mind that this is not a recommended topology. Ideally, you should have a minimum of two NSX Edge nodes to take full advantage of the logical router design.

The tier-1 logical router has a web logical switch and an app logical switch with respective VMs attached. The router-link switch between the tier-1 router and the tier-0 router is created automatically when you attach the tier-1 router to the tier-0 router. Thus, this switch is labeled as system generated.

In some scenarios, external clients send ARP queries for MAC addresses bound to loopback or IKE IP ports. However, loopback and IKE IP ports do not have MAC addresses and cannot handle such queries. Proxy ARP is implemented on the uplink and centralized service ports of a tier-0 logical router to handle ARP queries on behalf of the loopback and IKE IP ports.

When a tier-0 logical router is configured with DNAT, IPsec, and Edge firewall, traffic is processed in this order: IPsec first, then DNAT, and then Edge firewall.

On a tier-0 or tier-1 logical router, you can configure different types of ports. One type is called centralized service port (CSP). You must configure a CSP on a tier-0 logical router in active-standby mode or a tier-1 logical router to connect to a VLAN-backed logical switch, or to create a standalone tier-1 logical router. A CSP supports the following services on a tier-0 logical router in active-standby mode or a tier-1 logical router:

- NAT
- Load balancing
- Stateful firewall
- VPN (IPsec and L2VPN)

Create a Tier-0 Logical Router

Tier-0 logical routers have downlink ports to connect to NSX-T Data Center tier-1 logical routers and uplink ports to connect to external networks.

Prerequisites

- Verify that at least one NSX Edge is installed. See the *NSX-T Data Center Installation Guide*
- Verify that an NSX Edge cluster is configured. See the *NSX-T Data Center Installation Guide*.
- Familiarize yourself with the networking topology of the tier-0 logical router. See Tier-0 Logical Router.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security** > **Routers** > **Routers** > **Add**.
3. Select **Tier-0 Router** from the drop-down menu.

4. Assign a name for the tier-0 logical router.
5. Select an existing NSX Edge cluster from the drop-down menu to back this tier-0 logical router.
6. (Optional) Select a high-availability mode.

   By default, the active-active mode is used. In the active-active mode, traffic is load balanced across all members. In active-standby mode, all traffic is processed by an elected active member. If the active member fails, a new member is elected to be active.
7. (Optional) Click the **Advanced** tab to enter a subnet for the intra-tier 0 transit subnet.

   This is the subnet that connects to the tier-0 services router to its distributed router. If you leave this blank, the default 169.0.0.0/28 subnet is used.
8. (Optional) Click the **Advanced** tab to enter a subnet for the tier-0-tier-1 transit subnet.

   This is the subnet that connects the tier-0 router to any tier-1 routers that connect to this tier-0 router. If you leave this blank, the default address space assigned for these tier-0-to-tier-1 connections is 100.64.0.0/16. Each tier-0-to-tier-1 peer connection is provided a /31 subnet within the 100.64.0.0/16 address space.
9. Click **Save**.

   The new tier-0 logical router appears as a link.
10. (Optional) Click the tier-0 logical router link to review the summary.

What to do next

Attach tier-1 logical routers to this tier-0 logical router.

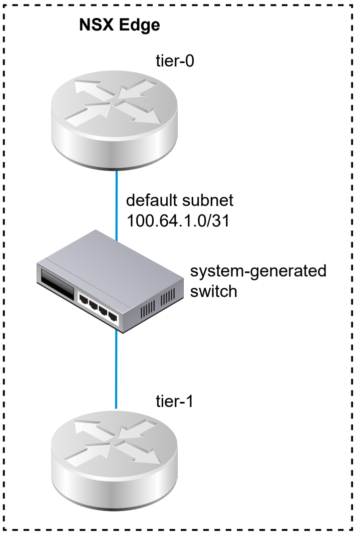
Configure the tier-0 logical router to connect it to a VLAN logical switch to create an uplink to an external network. See Connect a Tier-0 Logical Router to a VLAN Logical Switch for the NSX Edge Uplink.

Attach Tier-0 and Tier-1

You can attach the tier-0 logical router to the tier-1 logical router so that the tier-1 logical router gets northbound and east-west network connectivity.

When you attach a tier-1 logical router to a tier-0 logical router, a router-link switch between the two routers is created. This switch is labeled as system-generated in the topology. The default address space assigned for these tier-0-to-tier-1 connections is 100.64.0.0/16. Each tier-0-to-tier-1 peer connection is provided a /31 subnet within the 100.64.0.0/16 address space. Optionally, you can configure the address space in the tier-0 **Summary > Advanced** configuration.

The following figure shows a sample topology.



Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Select the tier-1 logical router.

4 From the **Summary** tab, click **Edit**.

5 Select the tier-0 logical router from the drop-down menu.

6 (Optional) Select an NSX Edge cluster from the drop-down menu.

   The tier-1 router needs to be backed by an edge device if the router is going to be used for services, such as NAT. If you do not select an NSX Edge cluster, the tier-1 router cannot perform NAT.

7 Specify members and a preferred member.

   If you select an NSX Edge cluster and leave the members and preferred member fields blank, NSX-T Data Center sets the backing edge device from the specified cluster for you.

8 Click **Save**.

9 Click the **Configuration** tab of the tier-1 router to verify that a new point-to-point linked port IP address is created.

   For example, the IP address of the linked port can be 100.64.1.1/31.

10 Select the tier-0 logical router from the navigation panel.

11 Click the **Configuration** tab of the tier-0 router to verify that a new point-to-point linked port IP address is created.

   For example, the IP address of the linked port can be 100.64.1.1/31.

What to do next

Verify that the tier-0 router is learning routes that are advertised by the tier-1 routers.

Verify that a Tier-0 Router Has Learned Routes from a Tier-1 Router

When a tier-1 logical router advertises routes to a tier-0 logical router, the routes are listed in the tier-0 router's routing table as NSX-T Data Center static routes.

Procedure

1 On the NSX Edge, run the `get logical-routers` command to find the VRF number of the tier-0 service router.

```
nsx-edge-1> get logical-routers
Logical Router
UUID        : 736a80e3-23f6-5a2d-81d6-bbefb2786666
vrf         : 0
type        : TUNNEL

Logical Router
UUID        : 421a2d0d-f423-46f1-93a1-2f9e366176c8
vrf         : 5
type        : SERVICE_ROUTER_TIER0

Logical Router
UUID        : f3ce9d7d-7123-47d6-aba6-45cf1388ca7b
vrf         : 6
type        : DISTRIBUTED_ROUTER

Logical Router
UUID        : c8e64eff-02b2-4462-94ff-89f3788f1a61
vrf         : 7
type        : SERVICE_ROUTER_TIER1

Logical Router
UUID        : fb6c3f1f-599f-4421-af8a-99692dff3dd4
vrf         : 8
type        : DISTRIBUTED_ROUTER
```

2 Run the `vrf <number>` command to enter the tier-0 service router context.

```
nsx-edge-1> vrf 5
nsx-edge1(tier0_sr)>
```

3 On the tier-0 service router, run the `get route` command and make sure the expected routes appear in the routing table.

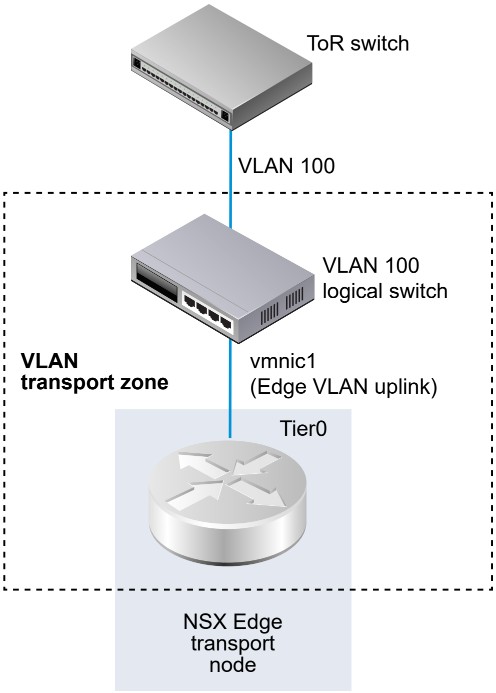
Notice that the NSX-T Data Center static routes (ns) are learned by the tier-0 router because the tier-1 router is advertising routes.

```
nsx-edge1(tier0_sr)> get route

Flags: c - connected, s - static, b - BGP, ns - nsx_static
nc - nsx_connected, rl - router_link, t0n: Tier0-NAT, t1n: Tier1-NAT

Total number of routes: 7

b    10.10.10.0/24        [20/0]        via 192.168.100.254
rl   100.91.176.0/31      [0/0]         via 169.254.0.1
c    169.254.0.0/28       [0/0]         via 169.254.0.2
ns   172.16.10.0/24       [3/3]         via 169.254.0.1
ns   172.16.20.0/24       [3/3]         via 169.254.0.1
c    192.168.100.0/24     [0/0]         via 192.168.100.2
```

Connect a Tier-0 Logical Router to a VLAN Logical Switch for the NSX Edge Uplink

To create an NSX Edge uplink, you must connect a tier-0 router to a VLAN switch.

The following simple topology shows a VLAN logical switch inside of a VLAN transport zone. The VLAN logical switch has a VLAN ID that matches the VLAN ID on the TOR port for the Edge's VLAN uplink.



Prerequisites

Create a VLAN logical switch. See Create a VLAN Logical Switch for the NSX Edge Uplink.

Create a tier-0 router.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Select the tier-0 logical router.
4. From the **Configuration** tab, add a new logical router port.
5. Type a name for the port, such as uplink.
6. Select the **Uplink** type.
7. Select an edge transport node.
8. Select a VLAN logical switch.
9. Type an IP address in CIDR format in the same subnet as the connected port on the TOR switch.

Results

A new uplink port is added for the tier-0 router.

What to do next

Configure BGP or a static route.

Verify the Tier-0 Logical Router and TOR Connection

For routing to work on the uplink from the tier-0 router, connectivity with the top-of-rack device must be in place.

Prerequisites

- Verify that the tier-0 logical router is connected to a VLAN logical switch. See Connect a Tier-0 Logical Router to a VLAN Logical Switch for the NSX Edge Uplink.

Procedure

1. Log in to the NSX Manager CLI.
2. On the NSX Edge, run the `get logical-routers` command to find the VRF number of the tier-0 service router.

```
nsx-edge-1> get logical-routers
Logical Router
UUID        : 736a80e3-23f6-5a2d-81d6-bbefb2786666
```

```
vrf         : 0
type        : TUNNEL

Logical Router
UUID        : 421a2d0d-f423-46f1-93a1-2f9e366176c8
vrf         : 5
type        : SERVICE_ROUTER_TIER0

Logical Router
UUID        : f3ce9d7d-7123-47d6-aba6-45cf1388ca7b
vrf         : 6
type        : DISTRIBUTED_ROUTER

Logical Router
UUID        : c8e64eff-02b2-4462-94ff-89f3788f1a61
vrf         : 7
type        : SERVICE_ROUTER_TIER1

Logical Router
UUID        : fb6c3f1f-599f-4421-af8a-99692dff3dd4
vrf         : 8
type        : DISTRIBUTED_ROUTER
```

3 Run the `vrf <number>` command to enter the tier-0 service router context.

```
nsx-edge-1> vrf 5
nsx-edge1(tier0_sr)>
```

4 On the tier-0 service router, run the `get route` command and make sure the expected route appears in the routing table.

Notice that the route to the TOR appears as connected (c).

```
nsx-edge1(tier0_sr)> get route

Flags: c - connected, s - static, b - BGP, ns - nsx_static
nc - nsx_connected, rl - router_link, t0n: Tier0-NAT, t1n: Tier1-NAT

Total number of routes: 7

b    10.10.10.0/24        [20/0]        via 192.168.100.254
rl   100.91.176.0/31      [0/0]         via 169.254.0.1
c    169.254.0.0/28       [0/0]         via 169.254.0.2
ns   172.16.10.0/24       [3/3]         via 169.254.0.1
ns   172.16.20.0/24       [3/3]         via 169.254.0.1
c    192.168.100.0/24     [0/0]         via 192.168.100.2
```

5 Ping the TOR.

```
nsx-edge1(tier0_sr)> ping    192.168.100.254
PING 192.168.100.254 (192.168.100.254): 56 data bytes
64 bytes from 192.168.100.254: icmp_seq=0 ttl=64 time=2.822 ms
64 bytes from 192.168.100.254: icmp_seq=1 ttl=64 time=1.393 ms
^C
nsx-edge1>
--- 192.168.100.254 ping statistics ---
3 packets transmitted, 2 packets received, 33.3% packet loss
round-trip min/avg/max/stddev = 1.393/2.107/2.822/0.715 ms
```

Results

Packets are sent between the tier-0 logical router and physical router to verify a connection.

What to do next

Depending on your networking requirements, you can configure a static route or BGP. See Configure a Static Route or Configure eBGP on a Tier-0 Logical Router.

Add a Loopback Router Port

You can add a loopback port to a tier-0 logical router.

The loopback port can be used for the following purposes:

- Router ID for routing protocols
- NAT
- BFD
- Source address for routing protocols

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Select the tier-0 logical router.

4 Select **Configuration > Router Ports**

5 Click **Add**.

6 Enter a name and optionally a description.

7 Select the **Loopback** type.

8 Select an edge transport node.

9 Enter an IP address in CIDR format.

Results

A new port is added for the tier-0 router.

Add a VLAN Port on a Tier-0 or Tier-1 Logical Router

If you have only VLAN-backed logical switches, you can connect the switches to VLAN ports on a tier-0 or tier-1 router so that NSX-T Data Center can provide layer-3 services.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Click the name of a router.
4. Click the **Configuration** tab and select **Router Ports**.
5. Click **Add**.
6. Enter a name for the router port and optionally a description.
7. In the **Type** field, select **Centralized**.
8. For **URPF Mode**, select **Strict** or **None**.

   URPF (unicast Reverse Path Forwarding) is a security feature.
9. (Required) Select a logical switch.
10. Select whether this attachment creates a switch port or updates an existing switch port.

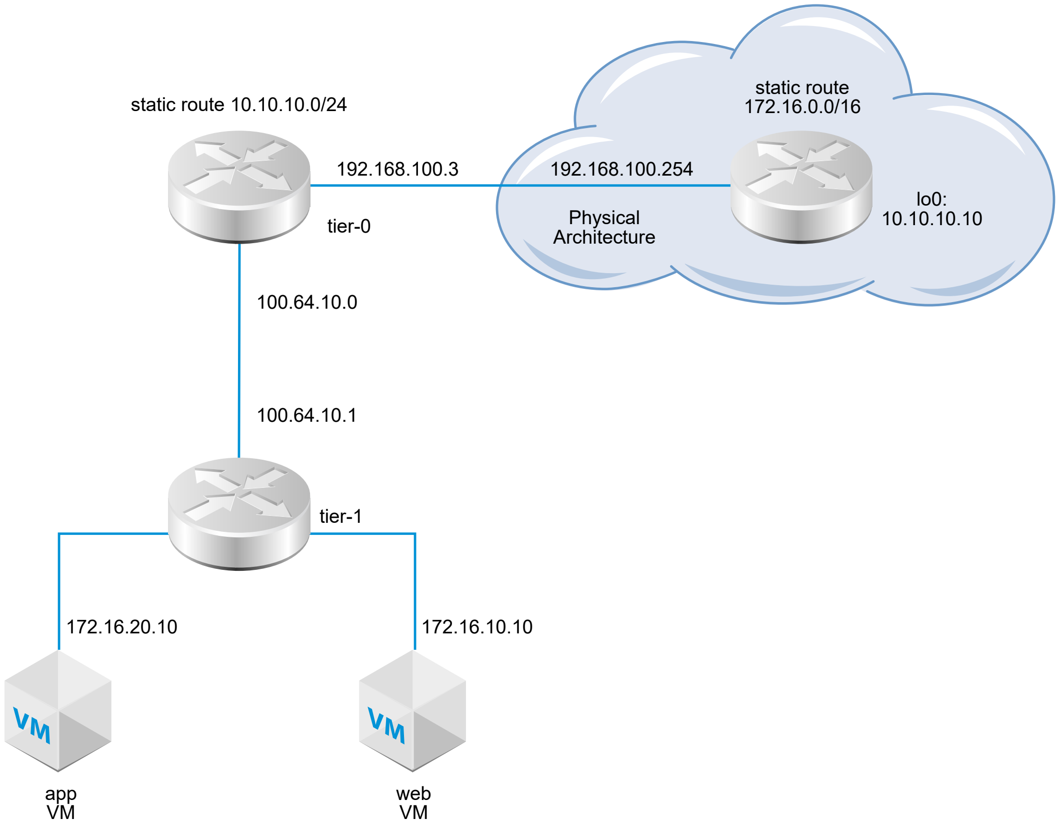
    If the attachment is for an existing switch port, select the port from the drop-down menu.
11. Enter the router port IP address in CIDR notation.
12. Click **Add**.

Configure a Static Route

You can configure a static route on the tier-0 router to external networks. After you configure a static route, there is no need to advertise the route from tier-0 to tier-1, because tier-1 routers automatically have a static default route towards their connected tier-0 router.

The static route topology shows a tier-0 logical router with a static route to the 10.10.10.0/24 prefix in the physical architecture. For test purposes, the 10.10.10.10/32 address is configured on the external router loopback interface. The external router has a static route to the 172.16.0.0/16 prefix to reach the app and web VMs.

Figure 14-4. Static Route Topology



Recursive static routes are supported.

Prerequisites

- Verify that the physical router and tier-0 logical router are connected. See Verify the Tier-0 Logical Router and TOR Connection.
- Verify that the tier-1 router is configured to advertise connected routes. See Create a Tier-1 Logical Router .

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Select the tier-0 logical router.
4. Click the **Routing** tab and select **Static Route** from the drop-down menu.
5. Select **Add**.
6. Enter a network address in the CIDR format.

   For example, 10.10.10.0/24.

7 Click **+ Add** to add a next-hop IP address.

   For example, 192.168.100.254. You can also specify a null route by clicking the pencil icon and selecting **NULL** from the drop-down.

8 Specify the administrative distance.

9 Select a logical router port from the dropdown list.

   The list includes IPSec Virtual Tunnel Interface (VTI) ports.

10 Click the **Add** button.

What to do next

Check that the static route is configured properly. See Verify the Static Route.

Verify the Static Route

Use the CLI to verify that the static route is connected. You must also verify the external router can ping the internal VMs and the internal VMs can ping the external router.

Prerequisites

Verify that a static route is configured. See Configure a Static Route.

Procedure

1 Log in to the NSX Manager CLI.

2. Confirm the static route.

   a. Get the service router UUID information.

      ```
      get logical-routers
      ```

      ```
      nsx-edge1> get logical-routers
      Logical Router
      UUID        : 736a80e3-23f6-5a2d-81d6-bbefb2786666
      vrf         : 2
      type        : TUNNEL

      Logical Router
      UUID        : d40bbfa4-3e3d-4178-8615-6f42ea335037
      vrf         : 4
      type        : SERVICE_ROUTER_TIER0

      Logical Router
      UUID        : d0289ba4-250e-41b4-8ffc-7cab4a46c3e4
      vrf         : 5
      type        : DISTRIBUTED_ROUTER

      Logical Router
      UUID        : a6ee6316-2212-4171-99cc-930c98bcad7f
      vrf         : 6
      type        : DISTRIBUTED_ROUTER
      ```

   b. Locate the UUID information from the output.

      ```
      Logical Router
      UUID        : d40bbfa4-3e3d-4178-8615-6f42ea335037
      vrf         : 4
      type        : SERVICE_ROUTER_TIER0
      ```

   c. Verify that the static route works.

      ```
      get logical-router d40bbfa4-3e3d-4178-8615-6f42ea335037 route static
      ```

      ```
      Flags: c - connected, s - static, b - BGP, ns - nsx_static
      nc - nsx_connected, rl - router_link, t0n: Tier0-NAT, t1n: Tier1-NAT

      s    10.10.10.0/24       [1/1]        via 192.168.100.254
      rl   100.64.1.0/31       [0/0]        via 169.0.0.1
      ns   172.16.10.0/24      [3/3]        via 169.0.0.1
      ns   172.16.20.0/24      [3/3]        via 169.0.0.1
      ```

3 From the external router, ping the internal VMs to confirm that they are reachable through the NSX-T Data Center overlay.

   a Connect to the external router.

   `ping 172.16.10.10`

```
PING 172.16.10.10 (172.16.10.10) 56(84) bytes of data.
64 bytes from 172.16.10.10: icmp_req=1 ttl=62 time=127 ms
64 bytes from 172.16.10.10: icmp_req=2 ttl=62 time=1.96 ms
^C
--- 172.16.10.10 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.966/64.793/127.620/62.827 ms
```

   b Test the network connectivity.

   `traceroute 172.16.10.10`

```
traceroute to 172.16.10.10 (172.16.10.10), 30 hops max, 60 byte packets
 1  192.168.100.3 (192.168.100.3)  0.640 ms  0.575 ms  0.696 ms
 2  100.64.1.1 (100.64.1.1)  0.656 ms  0.604 ms  0.578 ms
 3  172.16.10.10 (172.16.10.10)  3.397 ms  3.703 ms  3.790 ms
```

4 From the VMs, ping the external IP address.

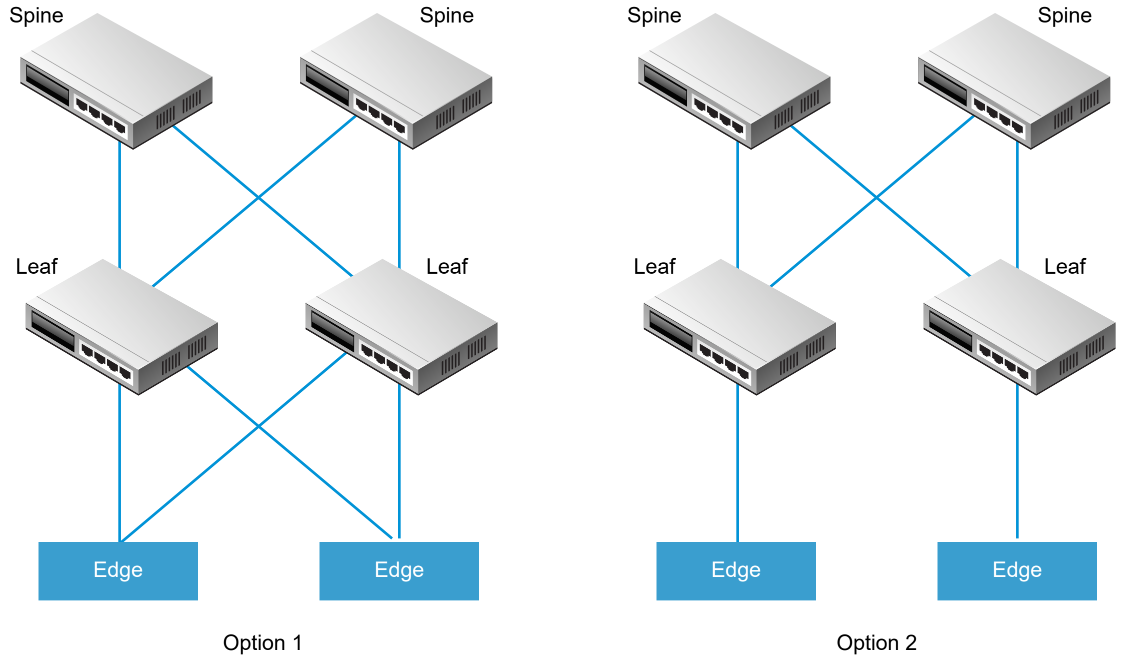
`ping 10.10.10.10`

```
PING 10.10.10.10 (10.10.10.10) 56(84) bytes of data.
64 bytes from 10.10.10.10: icmp_req=1 ttl=62 time=119 ms
64 bytes from 10.10.10.10: icmp_req=2 ttl=62 time=1.93 ms
^C
--- 10.10.10.10 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.936/60.865/119.795/58.930 ms
```

BGP Configuration Options

To take full advantage of the tier-0 logical router, the topology must be configured with redundancy and symmetry with BGP between the tier-0 routers and the external top-of-rack peers. This design helps to ensure connectivity in the event of link and node failures.

There are two modes of configuration: active-active and active-standby. The following diagram shows two options for symmetric configuration. There are two NSX Edge nodes shown in each topology. In the case of an active-active configuration, when you create tier-0 uplink ports, you can associate each uplink port with up to eight NSX Edge transport nodes. Each NSX Edge node can have two uplinks.



For option 1, when the physical leaf-node routers are configured, they should have BGP neighborships with the NSX Edges. Route redistribution should include the same network prefixes with equal BGP metrics to all of the BGP neighbors. In the tier-0 logical router configuration, all leaf-node routers should be configured as BGP neighbors.

When you are configuring the tier-0 router's BGP neighbors, if you do not specify a local address (the source IP address), the BGP neighbor configuration is sent to all NSX Edge nodes associated with the tier-0 logical router uplinks. If you do configure a local address, the configuration goes to the NSX Edge node with the uplink owning that IP address.

In the case of option1, if the uplinks are on the same subnet on the NSX Edge nodes, it makes sense to omit the local address. If the uplinks on the NSX Edge nodes are in different subnets, the local address should be specified in the tier-0 router's BGP neighbor configuration to prevent the configuration from going to all associated NSX Edge nodes.

For option 2, ensure that the tier-0 logical router configuration includes the tier-0 services router's local IP address. The leaf-node routers are configured with only the NSX Edges that they are directly connected to as the BGP neighbor.

Configure eBGP on a Tier-0 Logical Router

To enable access between your VMs and the outside world, you can configure an external or internal BGP (eBGP/iBGP) connection between a tier-0 logical router and a router in your physical infrastructure.

When configuring eBGP, you must configure a local Autonomous System (AS) number for the tier-0 logical router. For example, the following topology shows the local AS number is 64510. You must also configure the remote AS number of the physical router. In this example, the remote AS number is 64511. The remote neighbor IP address is 192.168.100.254. The neighbor must be in the same IP subnet as the uplink on the tier-0 logical router. BGP multi-hop is supported.

For test purposes, the 10.10.10.10/32 address is configured on the external router loopback interface.

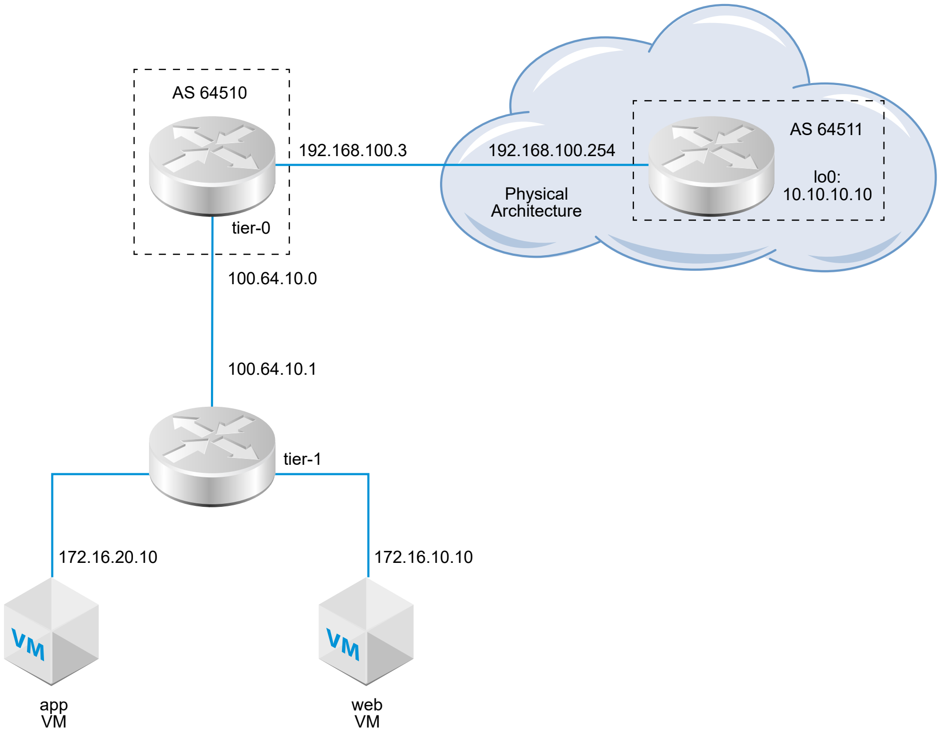
A tier-0 logical router in active-active mode supports inter-SR (service router) routing. If router #1 is unable to communicate with a northbound physical router, traffic is re-routed to router #2 in the active-active cluster. If router #2 is able to communicate with the physical router, traffic between router #1 and the physical router will not be affected.

In a topology with a tier-0 logical router in active-active mode attached to a tier-1 logical router in active-standby mode, you must enable inter-SR routing to handle asymmetric routing. You have asymmetric routing if you configure a static route on one of the SRs, or if one SR needs to reach another SR's uplink. In addition, note the following:

- In the case of a static route configured on one SR (for example, SR #1 on Edge node #1), another SR (for example, SR #2 on Edge node #2) might learn the same route from an eBGP peer and prefer the learned route to the static route on SR #1, which might be more efficient. To ensure that SR #2 uses the static route configured on SR #1, configure the tier-1 logical router in pre-emptive mode and configure Edge node #1 as the preferred node.
- If the tier-0 logical router has an uplink port on Edge node #1 and another uplink port on Edge node #2, ping traffic from tenant VMs to the uplinks works if the two uplinks are in different subnets. Ping traffic will fail if the two uplinks are in the same subnet.

Note Router ID used for forming BGP sessions on an edge node is automatically selected from the IP addresses configured on the uplinks of a tier-0 logical router. BGP sessions on an edge node can flap when router ID changes. This can happen when the IP address auto-selected for router ID is deleted or the logical router port on which this IP is assigned is deleted.

Figure 14-5. BGP Connection Topology



Note the following scenarios when there are connection failures involving BGP or BFD:

- With only BGP configured, if all BGP neighbors go down, the service router's state will be down.
- With only BFD configured, if all BFD neighbors go down, the service router's state will be down.
- With BGP and BFD configured, if all BGP and BFD neighbors go down, the service router's state will be down.
- With BGP and static routes configured, if all BGP neighbors go down, the service router's state will be down.
- With only static routes configured, the service router's state will always be up unless the node is experiencing a failure or in a maintenance mode.

Prerequisites

- Verify that the tier-1 router is configured to advertise connected routes. See Configure Route Advertisement on a Tier-1 Logical Router. This is not strictly a prerequisite for BGP configuration, but if you have a two-tier topology and you plan to redistribute your tier-1 networks into BGP, this step is required.
- Verify that a tier-0 router is configured. See Create a Tier-0 Logical Router.
- Make sure the tier-0 logical router has learned routes from the tier-1 logical router. See Verify that a Tier-0 Router Has Learned Routes from a Tier-1 Router.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Select the tier-0 logical router.
4. Click the **Routing** tab and select **BGP** from the drop-down menu.
5. Click **Edit**.
    a. Enter the local AS number.

       For example, 64510.
    b. Click the **Status** toggle to enable or disable BGP.
    c. Click the **ECMP** toggle to enable or disable ECMP.
    d. Click the **Graceful Restart** toggle to enable or disable graceful restart.

       Graceful restart is only supported if the NSX Edge cluster associated with the tier-0 router has only one edge node.
    e. If this logical router is in active-active mode, click the **Inter SR Routing** toggle to enable or disable inter-SR routing.
    f. Configure route aggregation.
    g. Click **Save**.
6. Click **Add** to add a BGP neighbor.
7. Enter the neighbor IP address.

   For example, 192.168.100.254.
8. Specify the maximum hop limit.

   The default is 1.
9. Enter the remote AS number.

   For example, 64511.
10. Configure the timers (keep alive time and hold down time) and a password.
11. Click the **Local Address** tab to select a local address.
    a. (Optional) Uncheck **All Uplinks** to see loopback ports as well as uplink ports.
12. Click the **Address Families** tab to add an address family.
13. Click the **BFD Configuration** tab to enable BFD.
14. Click **Save**.

What to do next

Test whether BGP is working properly. See Verify BGP Connections from a Tier-0 Service Router .

Configure iBGP on a Tier-0 Logical Router

You can configure internal BGP (iBGP) for tier-0 logical routers using the API. With iBGP configured, the tier-0 logical routers can exchange routing and reachability information.

The iBGP feature has the following capabilities and restrictions:

- Redistribution, prefix lists, and routes maps are supported.
- Route reflectors are not supported.
- BGP confederation is not supported.

Configuring iBGP using the NSX Manager UI is not supported in this release.

Procedure

1 Call the following API to add a BGP neighbor with the `remote_as` parameter set to the same value as the local AS. For example,

```
POST https://<nsx-mgr>/api/v1/logical-routers/7a62a0c5-1ea1-4b25-9d43-dce1c0fa4b8c/routing/bgp/
neighbors
{
  "display_name": "neighbor1",
  "neighbor_address": "2.2.2.2",
  "remote_as_num": "200",
  "maximum_hop_limit": 1,
  "enabled": true,
  "logical_router_id": "c831795d-dc7b-448c-92ce-21b16ec9a7ad",
  "address_families": [
    {
      "type" : "IPV4_UNICAST",
      "enabled" : true,
     }
  ],
  "remote_as": 200,
  "enable_bfd": false,
}
```

2 Call the following API to add a route map with the `nexthop_self` parameter set to **`true`** and the `local_preference` parameter set to 200. For example,

```
POST https://<nsx-mgr>/api/v1/logical-routers/7a62a0c5-1ea1-4b25-9d43-dce1c0fa4b8c/routing/route-
maps
{
  "description": "Route Map",
  "display_name": "Route Map",
  "logical_router_id": "c831795d-dc7b-448c-92ce-21b16ec9a7ad",
  "sequences": [
    {
      "match_criteria": {
        "match_community_expression": {
          "expression": [
            {
              "match_operator": "MATCH_ALL",
              "community_list_id": "c4b2b171-661b-4059-960c-fc931a612507"
            }
          ],
          "operator": "AND"
          }
      },
      "set_criteria": {
        "as_path_prepend" : "50",
        "weight" : 50,
        "community" : "30:40",
        "multi_exit_discriminator" : 10,
        "nexthop_self" : true,
        "local_preference" : 200
      },
      "action": "PERMIT"
    }
  ]
}
```

Verify BGP Connections from a Tier-0 Service Router

Use the CLI to verify from the tier-0 service router that a BGP connection to a neighbor is established.

Prerequisites

Verify that BGP is configured. See Configure eBGP on a Tier-0 Logical Router.

Procedure

1. Log in to the NSX Manager CLI.
2. On the NSX Edge, run the `get logical-routers` command to find the VRF number of the tier-0 service router.

```
nsx-edge-1> get logical-routers
Logical Router
UUID        : 736a80e3-23f6-5a2d-81d6-bbefb2786666
vrf         : 0
type        : TUNNEL

Logical Router
UUID        : 421a2d0d-f423-46f1-93a1-2f9e366176c8
vrf         : 5
type        : SERVICE_ROUTER_TIER0

Logical Router
UUID        : f3ce9d7d-7123-47d6-aba6-45cf1388ca7b
vrf         : 6
```

```
type        : DISTRIBUTED_ROUTER

Logical Router
UUID        : c8e64eff-02b2-4462-94ff-89f3788f1a61
vrf         : 7
type        : SERVICE_ROUTER_TIER1

Logical Router
UUID        : fb6c3f1f-599f-4421-af8a-99692dff3dd4
vrf         : 8
type        : DISTRIBUTED_ROUTER
```

3 Run the `vrf <number>` command to enter the tier-0 service router context.

```
nsx-edge-1> vrf 5
nsx-edge1(tier0_sr)>
```

4 Verify that the BGP state is `Established, up`.

`get bgp neighbor`

```
BGP neighbor: 192.168.100.254   Remote AS: 64511
BGP state: Established, up
Hold Time: 180s   Keepalive Interval: 60s
Capabilities:
         Route Refresh: advertised and received
         Address Family: IPv4 Unicast:advertised and received
         Graceful Restart: none
         Restart Remaining Time: 0
Messages: 28 received, 31 sent
Minimum time between advertisements: 30s (default)
For Address Family IPv4 Unicast:advertised and received
         Route Refresh: 0 received, 0 sent
         Prefixes: 2 received, 2 sent, 2 advertised
1 Connections established, 2 dropped
Local host: 192.168.100.3, Local port: 179
Remote host: 192.168.100.254, Remote port: 33044
```

What to do next

Check the BGP connection from the external router. See Verify North-South Connectivity and Route Redistribution.

Configure BFD on a Tier-0 Logical Router

BFD (Bidirectional Forwarding Detection) is a protocol that can detect forwarding path failures.

Note In this release, BFD over Virtual Tunnel Interface (VTI) ports is not supported.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Select the tier-0 logical router.
4. Click the **Routing** tab and select **BFD** from the drop-down menu.
5. Click **Edit** to configure BFD.
6. Click the **Status** toggle button to enable BFD.

   You can optionally change the global BFD properties **Receive interval**, **Transmit interval**, and **Declare dead interval**.
7. (Optional) Click **Add** under BFD Peers for Static Route Next Hops to add a BFD peer.

   Specify the peer IP address and set the admin status to **Enabled**. Optionally, you can override the global BFD properties **Receive interval**, **Transmit interval**, and **Declare dead interval**.

Enable Route Redistribution on the Tier-0 Logical Router

When you enable route redistribution, the tier-0 logical router starts sharing specified routes with its northbound router.

Prerequisites

- Verify that the tier-0 and tier-1 logical routers are connected so that you can advertise the tier-1 logical router networks to redistribute them on the tier-0 logical router. See Attach Tier-0 and Tier-1.
- If you want to filter specific IP addresses from route redistribution, verify that route maps are configured. See Create a Route Map.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Select the tier-0 logical router.
4. Click the **Routing** tab and select **Route Redistribution** from the drop-down menu.
5. Click **Edit** to enable or disable route redistribution.

6 Click **Add** to add a set of route redistribution criteria.

Option	Description
Name and Description	Assign a name to the route redistribution. You can optionally provide a description. An example name, advertise-to-bgp-neighbor.
Sources	Select one or more of the following sources: ■ **T0 Connected** ■ **T0 Uplink** ■ **T0 Downlink** ■ **T0 CSP** ■ **T0 Loopback** ■ **T0 Static** ■ **T0 NAT** ■ **T0 DNS Forwarder IP** ■ **T0 IPSec Local IP** ■ **T1 Connected** ■ **T1 CSP** ■ **T1 Downlink** ■ **T1 Static** ■ **T1 LB SNAT** ■ **T1 NAT** ■ **T1 LB VIP** ■ **T1 DNS Forwarder IP**
Route Map	(Optional) Assign a route map to filter a sequence of IP addresses from route redistribution.

Verify North-South Connectivity and Route Redistribution

Use the CLI to verify that the BGP routes are learned. You can also check from the external router that the NSX-T Data Center-connected VMs are reachable.

Prerequisites

- Verify that BGP is configured. See Configure eBGP on a Tier-0 Logical Router.
- Verify that NSX-T Data Center static routes are set to be redistributed. See Enable Route Redistribution on the Tier-0 Logical Router.

Procedure

1 Log in to the NSX Manager CLI.

2 View the routes learned from the external BGP neighbor.

```
nsx-edge1(tier0_sr)> get route bgp

Flags: c - connected, s - static, b - BGP, ns - nsx_static
nc - nsx_connected, rl - router_link, t0n: Tier0-NAT, t1n: Tier1-NAT
```

```
b    10.10.10.0/24        [20/0]        via 192.168.100.254
```

3 From the external router, check that BGP routes are learned and that the VMs are reachable through the NSX-T Data Center overlay.

   a List the BGP routes.

   ```
   user@router# run show ip route bgp
   Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
          I - ISIS, B - BGP, > - selected route, * - FIB route

   B>* 172.16.10.0/24 [20/0] via 192.168.100.2, eth2, 00:00:48
   B>* 172.16.20.0/24 [20/0] via 192.168.100.2, eth2, 00:00:48
   B>* 172.16.30.0/24 [20/0] via 192.168.100.2, eth2, 00:00:48
   ```

   b From the external router, ping the NSX-T Data Center-connected VMs.

   `ping 172.16.10.10`

   ```
   PING 172.16.10.10 (172.16.10.10) 56(84) bytes of data.
   64 bytes from 172.16.10.10: icmp_req=1 ttl=62 time=127 ms
   64 bytes from 172.16.10.10: icmp_req=2 ttl=62 time=1.96 ms
   ^C
   --- 172.16.10.10 ping statistics ---
   2 packets transmitted, 2 received, 0% packet loss, time 1001ms
   rtt min/avg/max/mdev = 1.966/64.793/127.620/62.827 ms
   ```

   c Check the path through the NSX-T Data Center overlay.

   `traceroute 172.16.10.10`

   ```
   traceroute to 172.16.10.10 (172.16.10.10), 30 hops max, 60 byte packets
    1  192.168.100.3 (192.168.100.3)  0.640 ms  0.575 ms  0.696 ms
    2  100.91.176.1 (100.91.176.1)  0.656 ms  0.604 ms  0.578 ms
    3  172.16.10.10 (172.16.10.10)  3.397 ms  3.703 ms  3.790 ms
   ```

4 From the internal VMs, ping the external IP address.

`ping 10.10.10.10`

```
PING 10.10.10.10 (10.10.10.10) 56(84) bytes of data.
64 bytes from 10.10.10.10: icmp_req=1 ttl=62 time=119 ms
64 bytes from 10.10.10.10: icmp_req=2 ttl=62 time=1.93 ms
^C
--- 10.10.10.10 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.936/60.865/119.795/58.930 ms
```

What to do next

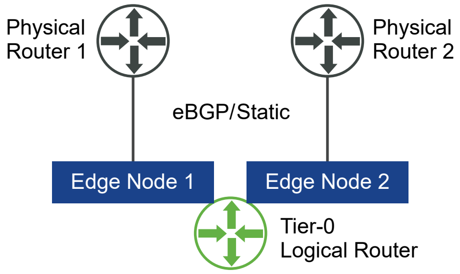
Configure additional routing functionality, such as ECMP.

Understanding ECMP Routing

Equal cost multi-path (ECMP) routing protocol increases the north and south communication bandwidth by adding an uplink to the tier-0 logical router and configure it for each Edge node in an NSX Edge cluster. The ECMP routing paths are used to load balance traffic and provide fault tolerance for failed paths.

The tier-0 logical router must be in active-active mode for ECMP to be available. A maximum of eight ECMP paths are supported.

Figure 14-6. ECMP Routing Topology



For example, the topology above shows a single tier-0 logical router in active-active mode running on a 2-node NSX Edge cluster. Two uplink ports are configured, one on each Edge node.

Add an Uplink Port for the Second Edge Node

Before you enable ECMP, you must configure an uplink to connect the tier-0 logical router to the VLAN logical switch.

Prerequisites

- Verify that a transport zone and two transport nodes are configured. See the *NSX-T Data Center Installation Guide*.
- Verify that two Edge nodes and an Edge cluster are configured. See the *NSX-T Data Center Installation Guide*.
- Verify that a VLAN logical switch for uplink is available. See Create a VLAN Logical Switch for the NSX Edge Uplink.
- Verify that a tier-0 logical router is configured. See Create a Tier-0 Logical Router.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Select the tier-0 logical router.

4 Click the **Configuration** tab to add a router port.

5 Click **Add**.

6 Complete the router port details.

Option	Description
Name	Assign a name for the router port.
Description	Provide additional description that shows that the port is for ECMP configuration.
Type	Accept the default type **Uplink**.
MTU	If you leave this field empty, the default is 1500.
Transport Node	Assign the Edge transport node from the drop-down menu.
URPF Mode	Unicast Reverse Path Forwarding is a security feature. Setting it to **None** is recommended if you have multiple active-active Edge nodes in ECMP mode. The default is **Strict**.
Logical Switch	Assign the VLAN logical switch from the drop-down menu.
Logical Switch Port	Assign a new switch port name. You can also use an existing switch port.
IP Address/Mask	Enter an IP address that is in the same subnet as the connected port on the ToR switch.

7 Click **Save**.

Results

A new uplink port is added to the tier-0 router and the VLAN logical switch. The tier-0 logical router is configured on both of the edge nodes.

What to do next

Create a BGP connection for the second neighbor and enable the ECMP routing. See Add a Second BGP Neighbor and Enable ECMP Routing.

Add a Second BGP Neighbor and Enable ECMP Routing

Before you enable ECMP routing, you must add a BGP neighbor and configure it with the newly added uplink information.

Prerequisites

Verify that the second edge node has an uplink port configured. See Add an Uplink Port for the Second Edge Node .

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2. Select **Advanced Networking & Security > Networking > Routers**.
3. Select the tier-0 logical router.
4. Click the **Routing** tab and select **BGP** from the drop-down menu.
5. Click **Add** under the Neighbors section to add a BGP neighbor.
6. Enter the neighbor IP address.

   For example, 192.168.200.254.
7. (Optional) Specify the maximum hop limit.

   The default is 1.
8. Enter the remote AS number.

   For example, 64511.
9. (Optional) Click the **Local Address** tab to select a local address.
   a. (Optional) Uncheck **All Uplinks** to see loopback ports as well as uplink ports.
10. (Optional) Click the **Address Families** tab to add an address family.
11. (Optional) Click the **BFD Configuration** tab to enable BFD.
12. Click **Save**.

    The newly added BGP neighbor appears.
13. Click **Edit** next to the BGP Configuration section.
14. Click the **ECMP** toggle button to enable ECMP.

    The Status button must be appear as Enabled.
15. Click **Save**.

Results

Multiple ECMP routing paths connect the VMs attached to logical switches and the two Edge nodes in the Edge cluster.

What to do next

Test whether the ECMP routing connections are working properly. See Verify ECMP Routing Connectivity.

Verify ECMP Routing Connectivity

Use CLI to verify that the ECMP routing connection to neighbor is established.

Prerequisites

Verify that ECMP routing is configured. See Add an Uplink Port for the Second Edge Node and Add a Second BGP Neighbor and Enable ECMP Routing.

Procedure

1. Log in to the NSX Manager CLI.
2. Get the distributed router UUID information.

   `get logical-routers`

   ```
   Logical Router
   UUID        : 736a80e3-23f6-5a2d-81d6-bbefb2786666
   vrf         : 2
   type        : TUNNEL

   Logical Router
   UUID        : d40bbfa4-3e3d-4178-8615-6f42ea335037
   vrf         : 4
   type        : SERVICE_ROUTER_TIER0

   Logical Router
   UUID        : d0289ba4-250e-41b4-8ffc-7cab4a46c3e4
   vrf         : 5
   type        : DISTRIBUTED_ROUTER

   Logical Router
   UUID        : a6ee6316-2212-4171-99cc-930c98bcad7f
   vrf         : 6
   type        : DISTRIBUTED_ROUTER
   ```

3. Locate the UUID information from the output.

   ```
   Logical Router
   UUID        : d0289ba4-250e-41b4-8ffc-7cab4a46c3e4
   vrf         : 5
   type        : DISTRIBUTED_ROUTER
   ```

4. Type the VRF for the tier-0 distributed router.

   `vrf 5`

5. Verify that the tier-0 distributed router is connected to the Edge nodes.

   `get forwarding`

   For example, edge-node-1 and edge-node-2.

6. Enter **`exit`** to leave the vrf context.
7. Verify that the tier-0 distributed router is connected.

   `get logical-router <UUID> route`

   The route type for the UUID should appear as `NSX_CONNECTED`.

8. Start a SSH session on the two Edge nodes.

9 Start a session to capture packets.

   ```
   set capture session 0 interface fp—eth1 dir tx
   ```

   ```
   set capture session 0 expression src net <IP_Address>
   ```

10 Use any tool that can generate traffic from a source VM connected to the tier-0 router to a destination VM.

11 Observe the traffic on the two Edge nodes.

Create an IP Prefix List

An IP prefix list contains single or multiple IP addresses that are assigned access permissions for route advertisement. The IP addresses in this list are processed sequentially. IP prefix lists are referenced through BGP neighbor filters or route maps with in or out direction.

For example, you can add the IP address 192.168.100.3/27 to the IP prefix list and deny the route from being redistributed to the northbound router. You can also append an IP address with less-than-or-equal-to (le) and greater-than-or-equal-to (ge) modifiers to grant or limit route redistribution. For example, 192.168.100.3/27 ge 24 le 30 modifiers match subnet masks greater than or equal to 24-bits and less than or equal to 30-bits in length.

Note The default action for a route is **Deny**. When you create a prefix list to deny or permit specific routes, be sure to create an IP prefix with no specific network address (select **Any** from the dropdown list) and the **Permit** action if you want to permit all other routes.

Prerequisites

Verify that you have a tier-0 logical router configured. See Create a Tier-0 Logical Router.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Select the tier-0 logical router.

4 Click the **Routing** tab and select **IP Prefix Lists** from the drop-down menu.

5 Click **Add**.

6 Enter a name for the IP prefix list.

7 Click **Add** to specify a prefix.

   a Enter an IP address in CIDR format.

      For example, 192.168.100.3/27.

   b Select **Deny** or **Permit** from the drop-down menu.

   c (Optional) Set a range of IP address numbers in the **le** or **ge** modifiers.

      For example, set **le** to 30 and **ge** to 24.

8 Repeat the previous step to specify additional prefixes.

9 Click **Add** at the bottom of the window.

Create a Community List

You can create BGP community lists so that you can configure route maps based on community lists.

Prerequisites

Verify that you have a tier-0 logical router configured. See Create a Tier-0 Logical Router.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Select the tier-0 logical router.

4 Click the **Routing** tab and select **Community Lists** from the drop-down menu.

5 Click **Add**.

6 Enter a name for the community list.

7 Specify a community using the aa:nn format, for example, 300:500, and press Enter. Repeat to add additional communities.

   In addition, you can click the dropdown arrow and select one or more of the following:

   - NO_EXPORT_SUBCONFED - Do not advertise to EBGP peers.
   - NO_ADVERTISE - Do not advertise to any peer.
   - NO_EXPORT - Do not advertise outside BGP confederation

8 Click **Add**.

Create a Route Map

A route map consists of a sequence of IP prefix lists, BGP path attributes, and an associated action. The router scans the sequence for an IP address match. If there is a match, the router performs the action and scans no further.

Route maps can be referenced at the BGP neighbor level and route redistribution. When IP prefix lists are referenced in route maps and the route map action of permitting or denying is applied, the action specified in the route map sequence overrides the specification within the IP prefix list.

Prerequisites

Verify that an IP prefix list is configured. See Create an IP Prefix List.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Select the tier-0 logical router.
4. Select **Routing > Route Maps**.
5. Click **Add**.
6. Enter a name and an optional description for the route map.
7. Click **Add** to add an entry in the route map.
8. Edit the column **Match IP Prefix List/Community List** to select either IP prefix lists, or community lists, but not both.
9. (Optional) Set BGP attributes.

BGP Attribute	Description
AS-path Prepend	Prepend a path with one or more AS (autonomous system) numbers to make the path longer and therefore less preferred.
MED	Multi-Exit Discriminator indicates to an external peer a preferred path to an AS.
Weight	Set a weight to influence path selection. The range is 0 - 65535.
Community	Specify a community using the aa:nn format, for example, 300:500. Or use the drop-down menu to select one of the following: ■ NO_EXPORT_SUBCONFED - Do not advertise to EBGP peers. ■ NO_ADVERTISE - Do not advertise to any peer. ■ NO_EXPORT - Do not advertise outside BGP confederation

10. In the Action column, select **Permit** or **Deny**.

    You can permit or deny IP addresses in the IP prefix lists from advertising their addresses.
11. Click **Save**.

Configure Forwarding Up Timer

You can configure forwarding up timer for a tier-0 logical router.

Forwarding up timer defines the time in seconds that the router must wait before sending the up notification after the first BGP session is established. This timer (previously known as forwarding delay) minimizes downtime in case of fail-overs for active-active or active-standby configurations of logical routers on NSX Edge that use dynamic routing (BGP). It should be set to the number of seconds an external router (TOR) takes to advertise all the routes to this router after the first BGP/BFD session. The timer value should be directly proportional to the number of northbound dynamic routes that the router must learn. This timer should be set to 0 on single edge node setups.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Select the tier-0 logical router.
4. Select **Routing > Global Configuration**
5. Click **Edit**.
6. Enter a value for the forwarding up timer.
7. Click **Save**.

Advanced NAT

15

You can configure NAT from the **Advanced Networking & Security** tab.

Note If you use the **Advanced Networking & Security** user interface to modify objects created in the policy interface, some settings might not be configurable. These read-only settings have this icon next to them: ⊖. See Chapter 1 Overview of the NSX Manager for more information.

This chapter includes the following topics:

- Network Address Translation

Network Address Translation

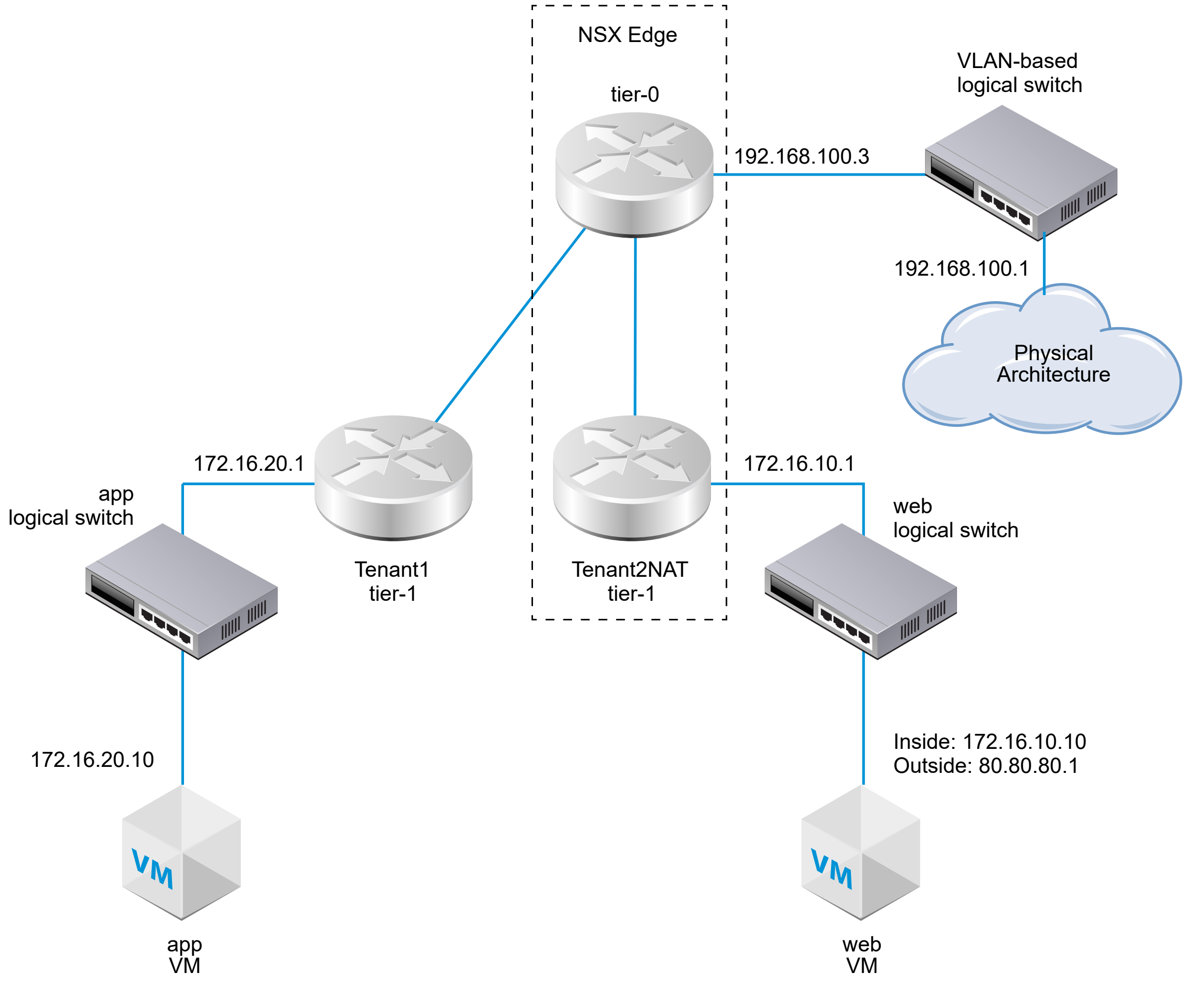
Network address translation (NAT) in NSX-T Data Center can be configured on tier-0 and tier-1 logical routers.

For example, the following diagram shows two tier-1 logical routers with NAT configured on Tenant2NAT. The web VM is simply configured to use 172.16.10.10 as its IP address and 172.16.10.1 as its default gateway.

NAT is enforced at the uplink of the Tenant2NAT logical router on its connection to the tier-0 logical router.

To enable NAT configuration, Tenant2NAT must have a service component on an NSX Edge cluster. Thus, Tenant2NAT is shown inside the NSX Edge. For comparison, Tenant1 can be outside of the NSX Edge because it is not using any Edge services.

Figure 15-1. NAT Topology



Tier-1 NAT

A tier-1 logical router supports source NAT (SNAT), destination NAT (DNAT) and reflexive NAT.

Configure Source NAT on a Tier-1 Router

Source NAT (SNAT) changes the source address in the IP header of a packet. It can also change the source port in the TCP/UDP headers. The typical usage is to change a private (rfc1918) address/port into a public address/port for packets leaving your network.

You can create a rule to either enable or disable source NAT.

In this example, as packets are received from the web VM, the Tenant2NAT tier-1 router changes the source IP address of the packets from 172.16.10.10 to 80.80.80.1. Having a public source IP address enables destinations outside of the private network to route back to the original source.

Prerequisites

- The tier-0 router must have an uplink connected to a VLAN-based logical switch. See Connect a Tier-0 Logical Router to a VLAN Logical Switch for the NSX Edge Uplink.

- The tier-0 router must have routing (static or BGP) and route redistribution configured on its uplink to the physical architecture. See Configure a Static Route, Configure eBGP on a Tier-0 Logical Router, and Enable Route Redistribution on the Tier-0 Logical Router.
- The tier-1 routers must each have an uplink to a tier-0 router configured. Tenant2NAT must be backed by an NSX Edge cluster. See Attach Tier-0 and Tier-1.
- The tier-1 routers must have downlink ports and route advertisement configured. See Add a Downlink Port on a Tier-1 Logical Router and Configure Route Advertisement on a Tier-1 Logical Router.
- The VMs must be attached to the correct logical switches.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Click a tier-1 logical router on which you want to configure NAT.

4 Select **Services > NAT**.

5 Click **ADD**.

6 Specify a priority value.

A lower value means a higher precedence for this rule.

7 For **Action**, select **SNAT** to enable source NAT, or **NO_SNAT** to disable source NAT.

8 Select the protocol type.

By default, **Any Protocol** is selected.

9 (Optional) For **Source IP**, specify an IP address or an IP address range in CIDR format.

If you leave this field blank, all sources on router's downlink ports are translated. In this example, the source IP address is 172.16.10.10.

10 (Optional) For **Destination IP**, specify an IP address or an IP address range in CIDR format.

If you leave this field blank, the NAT applies to all destinations outside of the local subnet.

11 If **Action** is **SNAT**, for **Translated IP**, specify an IP address or an IP address range in CIDR format.

In this example, the translated IP address is 80.80.80.1.

12 (Optional) For **Applied To**, select a router port.

13 (Optional) Set the status of the rule.

The rule is enabled by default.

14 (Optional) Change the logging status.

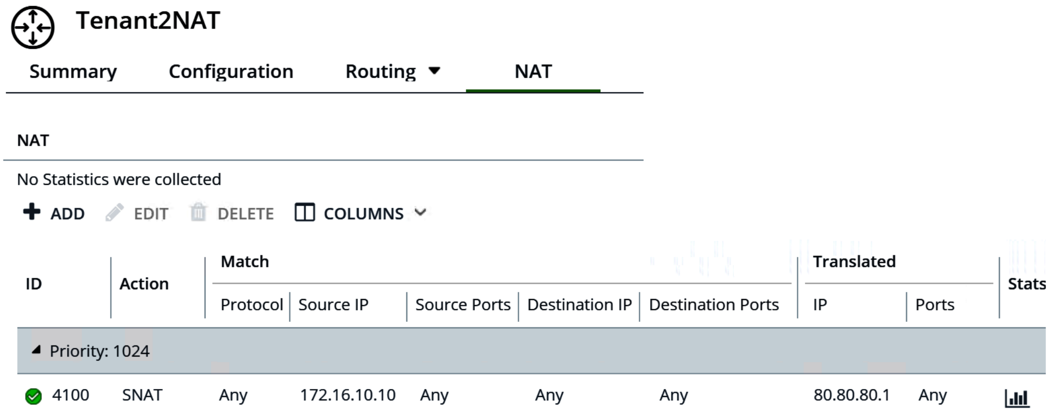
Logging is disabled by default.

15 (Optional) Change the firewall bypass setting.

The setting is enabled by default.

Results

The new rule is listed under NAT. For example:



No Statistics were collected

ID	Action	Match					Translated		Stats
		Protocol	Source IP	Source Ports	Destination IP	Destination Ports	IP	Ports	
Priority: 1024									
4100	SNAT	Any	172.16.10.10	Any	Any	Any	80.80.80.1	Any	

What to do next

Configure the tier-1 router to advertise NAT routes.

To advertise the NAT routes upstream from the tier-0 router to the physical architecture, configure the tier-0 router to advertise tier-1 NAT routes.

Configure Destination NAT on a Tier-1 Router

Destination NAT changes the destination address in IP header of a packet. It can also change the destination port in the TCP/UDP headers. The typical usage of this is to redirect incoming packets with a destination of a public address/port to a private IP address/port inside your network.

You can create a rule to either enable or disable destination NAT.

In this example, as packets are received from the app VM, the Tenant2NAT tier-1 router changes the destination IP address of the packets from 172.16.10.10 to 80.80.80.1. Having a public destination IP address enables a destination inside a private network to be contacted from outside of the private network.

Prerequisites

- The tier-0 router must have an uplink connected to a VLAN-based logical switch. See Connect a Tier-0 Logical Router to a VLAN Logical Switch for the NSX Edge Uplink.
- The tier-0 router must have routing (static or BGP) and route redistribution configured on its uplink to the physical architecture. See Configure a Static Route, Configure eBGP on a Tier-0 Logical Router, and Enable Route Redistribution on the Tier-0 Logical Router.
- The tier-1 routers must each have an uplink to a tier-0 router configured. Tenant2NAT must be backed by an NSX Edge cluster. See Attach Tier-0 and Tier-1.

- The tier-1 routers must have downlink ports and route advertisement configured. See Add a Downlink Port on a Tier-1 Logical Router and Configure Route Advertisement on a Tier-1 Logical Router.
- The VMs must be attached to the correct logical switches.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Click a tier-1 logical router on which you want to configure NAT.

4 Select **Services > NAT**.

5 Click **ADD**.

6 Specify a priority value.

A lower value means a higher precedence for this rule.

7 For **Action**, select **DNAT** to enable destination NAT, or **NO_DNAT** to disable destination NAT.

8 Select the protocol type.

By default, **Any Protocol** is selected.

9 (Optional) For **Source IP**, specify an IP address or an IP address range in CIDR format.

If you leave Source IP blank, the NAT applies to all sources outside of the local subnet.

10 For **Destination IP**, specify an IP address or an IP address range in CIDR format.

In this example, the destination IP address is 80.80.80.1.

11 If **Action** is **DNAT**, for **Translated IP**, specify an IP address or an IP address range in CIDR format.

In this example, the inside/translated IP address is 172.16.10.10.

12 (Optional) If **Action** is **DNAT**, for **Translated Ports**, specify the translated ports.

13 (Optional) For **Applied To**, select a router port.

14 (Optional) Set the status of the rule.

The rule is enabled by default.

15 (Optional) Change the logging status.

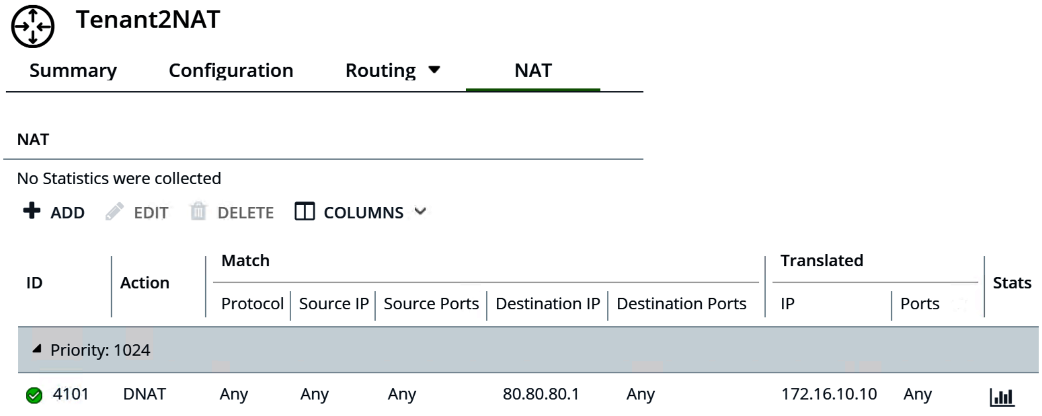
Logging is disabled by default.

16 (Optional) Change the firewall bypass setting.

The setting is enabled by default.

Results

The new rule is listed under NAT. For example:



No Statistics were collected

ID	Action	Match					Translated		Stats
		Protocol	Source IP	Source Ports	Destination IP	Destination Ports	IP	Ports	
Priority: 1024									
4101	DNAT	Any	Any	Any	80.80.80.1	Any	172.16.10.10	Any	

What to do next

Configure the tier-1 router to advertise NAT routes.

To advertise the NAT routes upstream from the tier-0 router to the physical architecture, configure the tier-0 router to advertise tier-1 NAT routes.

Advertise Tier-1 NAT Routes to the Upstream Tier-0 Router

Advertising tier-1 NAT routes enables the upstream tier-0 router to learn about these routes.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Click a tier-1 logical router on which you have configured NAT.

4 From the tier-1 router, select **Routing > Route Advertisement**.

5 Click **Edit** to edit the route advertisement configuration.

   You can toggle the following switches:

   - **Status**
   - **Advertise All NSX Connected Routes**
   - **Advertise All NAT Routes**
   - **Advertise All Static Routes**
   - **Advertise All LB VIP Routes**
   - **Advertise All LB SNAT IP Routes**
   - **Advertise All DNS Forwarder Routes**

6 Click **Save**.

What to do next

Advertise tier-1 NAT routes from the tier-0 router to the upstream physical architecture.

Advertise Tier-1 NAT Routes to the Physical Architecture

Advertising tier-1 NAT routes from the tier-0 router enables the upstream physical architecture to learn about these routes.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Routing**.
3. Click a tier-0 logical router that is connected to a tier-1 router on which you have configured NAT.
4. From the tier-0 router, select **Routing > Route Redistribution**.
5. Click **Edit** to enable or disable route redistribution.

6 Click **Add** to add a set of route redistribution criteria.

Option	Description
Name and Description	Assign a name to the route redistribution. You can optionally provide a description. An example name, advertise-to-bgp-neighbor.
Sources	Select one or more of the following sources: ■ **T0 Connected** ■ **T0 Uplink** ■ **T0 Downlink** ■ **T0 CSP** ■ **T0 Loopback** ■ **T0 Static** ■ **T0 NAT** ■ **T0 DNS Forwarder IP** ■ **T0 IPSec Local IP** ■ **T1 Connected** ■ **T1 CSP** ■ **T1 Downlink** ■ **T1 Static** ■ **T1 LB SNAT** ■ **T1 NAT** ■ **T1 LB VIP** ■ **T1 DNS Forwarder IP**
Route Map	(Optional) Assign a route map to filter a sequence of IP addresses from route redistribution.

Verify Tier-1 NAT

Verify that SNAT and DNAT rules are working correctly.

Procedure

1 Log in the NSX Edge.

2 Run `get logical-routers` to determine the VRF number for the tier-0 services router.

3 Enter the tier-0 services router context by running the `vrf <number>` command.

4 Run the `get route` command and make sure that the tier-1 NAT address appears.

```
nsx-edge(tier0_sr)> get route

Flags: c - connected, s - static, b - BGP, ns - nsx_static
nc - nsx_connected, rl - router_link, t0n: Tier0-NAT, t1n: Tier1-NAT
```

```
Total number of routes: 8

t1n  80.80.80.1/32        [3/3]         via 169.0.0.1
...
```

5 If your Web VM is set up to serve Web pages, make sure you can open a Web page at http://80.80.80.1.

6 Make sure that the tier-0 router's upstream neighbor in the physical architecture can ping 80.80.80.1.

7 While the ping is still running, check the stats column for the DNAT rule.

  There should be one active session.

Tier-0 NAT

A tier-0 logical router in active-standby mode supports source NAT (SNAT), destination NAT (DNAT) and reflexive NAT. A tier-0 logical router in active-active mode supports reflexive NAT only.

Configure Source and Destination NAT on a Tier-0 Logical Router

You can configure source and destination NAT on a tier-0 logical router that is running in active-standby mode.

You can also disable SNAT or DNAT for an IP address or a range of addresses. If multiple NAT rules apply to an address, the rule with the highest priority is applied.

SNAT configured on a tier-0 logical router's uplink will process traffic from a tier-1 logical router as well as from another uplink on the tier-0 logical router.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Routers**.

3 Click a tier-0 logical router.

4 Select **Services > NAT**.

5 Click **ADD** to add a NAT rule.

6 Specify a priority value.

  A lower value means a higher priority.

7 For **Action**, select **SNAT**, **DNAT**, **Reflexive**, **NO_SNAT**, or **NO_DNAT**.

8 Select the protocol type.

  By default, **Any Protocol** is selected.

9 (Required) For **Source IP**, specify an IP address or an IP address range in CIDR format.

   If you leave this field blank, this NAT rule applies to all sources outside of the local subnet.

10 For **Destination IP**, specify an IP address or an IP address range in CIDR format.

11 For **Translated IP**, specify an IP address or an IP address range in CIDR format.

12 (Optional) If **Action** is **DNAT**, for **Translated Ports**, specify the translated ports.

13 (Optional) For **Applied To**, select a router port.

14 (Optional) Set the status of the rule.

   The rule is enabled by default.

15 (Optional) Change the logging status.

   Logging is disabled by default.

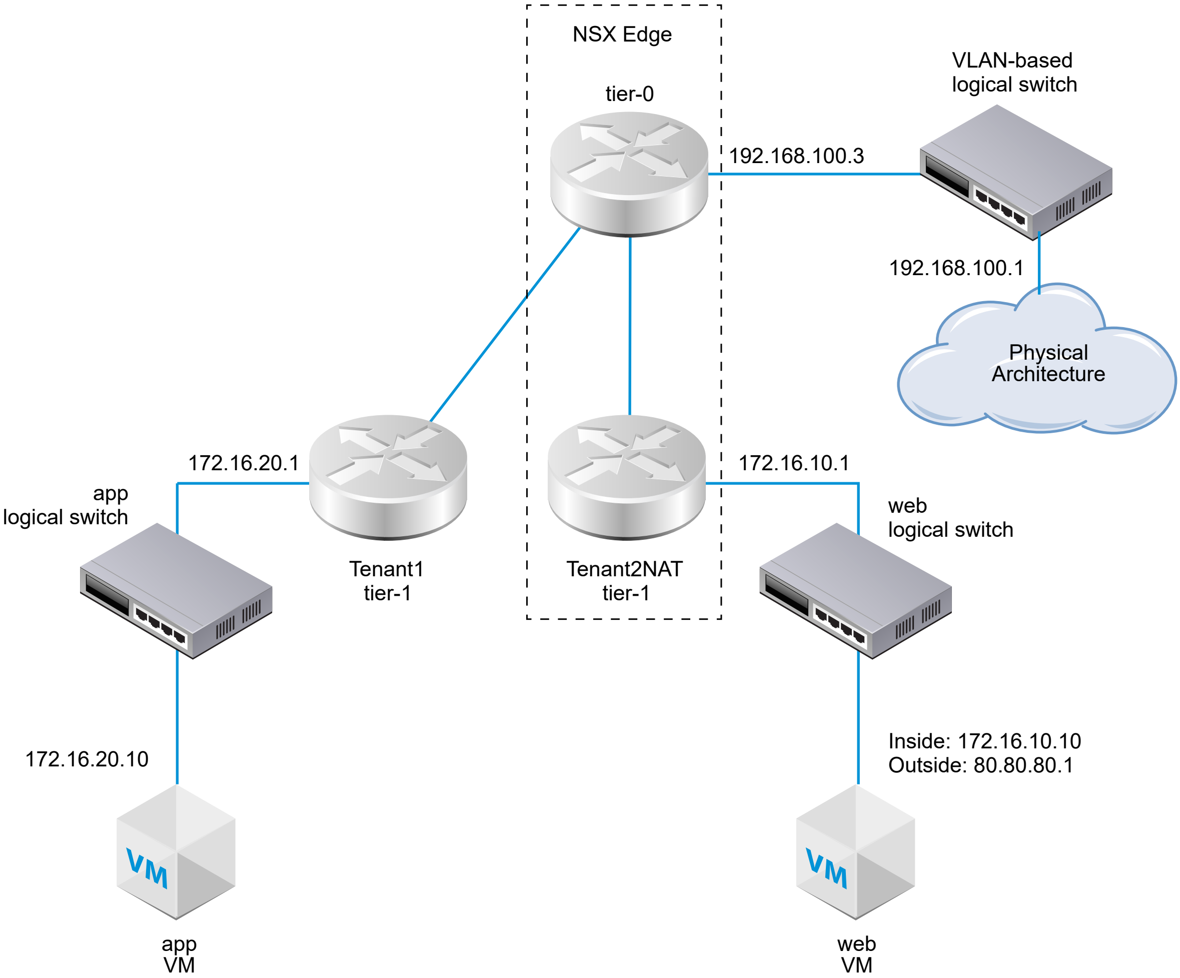
16 (Optional) Change the firewall bypass setting.

   The setting is enabled by default.

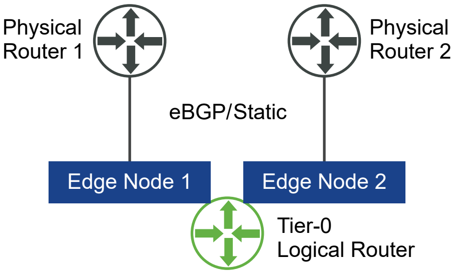
Reflexive NAT

When a tier-0 logical router is running in active-active mode, you cannot configure stateful NAT where asymmetrical paths might cause issues. For active-active routers, you can configure reflexive NAT (sometimes called stateless NAT).

In this example, as packets are received from the web VM, the Tenant2NAT tier-1 router changes the source IP address of the packets from 172.16.10.10 to 80.80.80.1. Having a public source IP address enables destinations outside of the private network to route back to the original source.



When there are two active-active tier-0 routers involved, as shown below, reflexive NAT must be configured.



Configure Reflexive NAT on a Tier-0 or Tier-1 Logical Router

When a tier-0 or tier-1 logical router is running in active-active mode, you cannot configure stateful NAT where asymmetrical paths might cause issues. For active-active routers, you can use reflexive NAT, which is sometimes called stateless NAT.

For reflexive NAT, you can configure a single source address to be translated, or a range of addresses. If you configure a range of source addresses, you must also configure a range of translated addresses. The size of the two ranges must be the same. The address translation will be deterministic, meaning that the first address in the source address range will be translated to the first address in the translated address range, the second address in the source range will be translated to the second address in the translated range, and so on.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Click a tier-0 or tier-1 logical router on which you want to configure reflexive NAT.
4. Select **Services > NAT**.
5. Click **ADD**.
6. Specify a priority value.

   A lower value means a higher precedence for this rule.
7. For **Action**, select **Reflexive**.
8. For **Source IP**, specify an IP address or an IP address range in CIDR format.
9. For **Translated IP**, specify an IP address or an IP address range in CIDR format.
10. (Optional) Set the status of the rule.

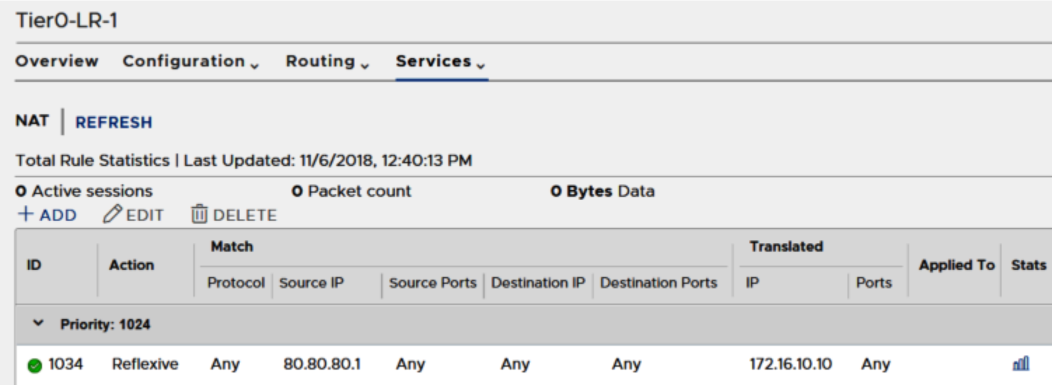
    The rule is enabled by default.
11. (Optional) Change the logging status.

    Logging is disabled by default.
12. (Optional) Change the firewall bypass setting.

    The setting is enabled by default.

Results

The new rule is listed under NAT. For example:



Total Rule Statistics | Last Updated: 11/6/2018, 12:40:13 PM

0 Active sessions **0** Packet count **0 Bytes** Data

ID	Action	Match					Translated		Applied To	Stats
		Protocol	Source IP	Source Ports	Destination IP	Destination Ports	IP	Ports		
Priority: 1024										
● 1034	Reflexive	Any	80.80.80.1	Any	Any	Any	172.16.10.10	Any		📊

Advanced Grouping Objects

16

You can create IP sets, IP pools, MAC sets, NSGroups, and NSServices. You can also manage tags for VMs.

Note If you use the **Advanced Networking & Security** user interface to modify objects created in the policy interface, some settings might not be configurable. These read-only settings have this icon next to them: ⊖. See Chapter 1 Overview of the NSX Manager for more information.

This chapter includes the following topics:

- Create an IP Set
- Create an IP Pool
- Create a MAC Set
- Create an NSGroup
- Configuring Services and Service Groups
- Manage Tags for a VM

Create an IP Set

An IP set is a group of IP addresses that can be used as sources and destinations in firewall rules.

An IP set can contain a combination of individual IP addresses, IP ranges, and subnets. You can specify IPv4 or IPv6 addresses, or both. An IP set can be a member of NSGroups. Any IP set created by this method will not be visible in Policy mode. In Policy mode, we can create a group and add members as IP addresses, ranges, network addresses, or MAC addresses by navigating to **Inventory > Groups > Set Members** and specifying IP or MAC addresses.

Note IPv4 addresses and IPv6 addresses are supported for source or destination ranges for firewall rules.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Inventory > Groups > IP Sets > Add**.

3 Enter a name.

4 (Optional) Enter a description.

5 In **Members**, enter individual IP addresses, IP ranges, and subnets in a comma separated list.

6 Click **Save**.

Create an IP Pool

You can use an IP Pool to allocate IP addresses or subnets when you create L3 subnets.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Inventory > Groups > IP Pools > Add**.

3 Enter a name for the new IP pool.

4 (Optional) Enter a description.

5 Click **Add**.

6 Click the IP Ranges cell and enter IP Ranges.

   Mouse over the upper right corner of any cell and click the pencil icon to edit it.

7 (Optional) Enter a Gateway.

8 Enter a CIDR IP address with suffix.

9 (Optional) Enter DNS Servers.

10 (Optional) Enter a DNS Suffix.

11 Click **Save**.

Create a MAC Set

A MAC Set is a group of MAC addresses that you can use as sources and destinations in layer 2 firewall rules and as a member of an NS Group.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Inventory > Groups > MAC Sets > Add**.

3 Enter a name.

4 (Optional) Enter a description.

5 Enter the MAC addresses in a comma-separated list.

6 Click **ADD**.

Create an NSGroup

NSGroups can be configured to contain a combination of IP sets, MAC sets, logical ports, logical switches, and other NSGroups. You can specify NSGroups with Logical Swithches, Logical ports and VMs as sources and destinations, and in the `Applied To` field of a firewall rule. NSGroups with IPset and MACSet will be ignored in a distributed firewall `Applied To` field.

NSX Cloud Note If using NSX Cloud, see How to use NSX-T Data Center Features with the Public Cloud for a list of auto-generated logical entities, supported features, and configurations required for NSX Cloud.

An NSGroup has the following characteristics:

- An NSGroup has direct members and effective members. Effective members include members that you specify using membership criteria, as well as all the direct and effective members that belong to this NSGroup's members. For example, assuming NSGroup-1 has direct member LogicalSwitch-1. You add NSGroup-2 and specify NSGroup-1 and LogicalSwitch-2 as members. Now NSGroup-2 has direct members NSGroup-1 and LogicalSwitch-2, and an effective member, LogicalSwitch-1. Next, you add NSGroup-3 and specify NSGroup-2 as a member. NSGroup-3 now has direct member NSGroup-2 and effective members LogicalSwitch-1 and LogicalSwitch-2. From the main groups table, clicking on a group and selecting **Related > NSGroups** would show NSGroup-1, NSGroup-2, and NSGroup-3 because all three have LogicalSwitch-1 as a member, either directly or indirectly.
- An NSGroup can have a maximum of 500 direct members.
- The recommended limit for the number of effective members in an NSGroup is 5000. The NSX Manager check the NSGroups regarding the limit twice a day, at 7 AM and 7 PM. Exceeding this limit does not affect any functionality but might have a negative impact on performance.
  - When the number of effective members for an NSGroup exceeds 80% of 5000, the warning message `NSGroup xyz is about to exceed the maximum member limit. Total number in NSGroup is ...` appears in the log file. When the number exceeds 5000, the warning message `NSGroup xyz has reached the maximum numbers limit. Total number in NSGroup = ...` appears.
  - When the number of translated VIFs/IPs/MACs in an NSGroup exceeds 5000, the warning message `Container xyz has reached the maximum IP/MAC/VIF translations limit. Current translations count in Container – IPs:..., MACs:..., VIFs:...` appears in the log file.
- The maximum supported number of VMs is 10,000.
- You can create a maximum of 10,000 NSGroups.

For all the objects that you can add to an NSGroup as members, you can navigate to the screen for any of the objects and select **Related > NSGroups**.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Inventory > Groups > Add**.
3. Enter a name for the NSGroup.
4. (Optional) Enter a description.
5. (Optional) Click **Membership Criteria**.

   For each criterion, you can specify up to five rules, which are combined with the logical AND operator. The available member criterion can apply to the following:

   - **Logical Port** - can specify a tag and optional scope.
   - **Logical Switch** - can specify a tag and optional scope.
   - **Virtual Machine** - can specify a name, tag, computer OS name, or computer name that equals, contains, starts with, ends with, or doesn't equal a particular string.
   - **Transport Node** - can specify a node type that equals an edge node or a host node.
6. (Optional) Click **Members** to select members.

   The available member types are:

   - **AD Group** - NSGroups with ADGroups can only be used in the extended_source field of a distributed firewall rule, and must be the only members in the group. For example, there cannot be an NSGroup with both ADGroup and IPSet together as members.
   - **IP Set** - can include both IPv4 an IPv6 addresses.
   - **Logical Port** - can include both IPv4 and IPv6 addresses.
   - **Logical Switch** - can include both IPv4 and IPv6 addresses.
   - **MAC Set**
   - **NSGroup**
   - **Transport Node**
   - **VIF**
   - **Virtual Machine**

7 Click **ADD**.

The group is added to the table of groups. Click a group name to display an overview and edit group information including membership criteria, members, applications, and related groups. Scroll to the bottom of the **Overview** tab to add and delete tags. See Add Tags to an Object for more information. Selecting **Related> NSGroups** displays all the NSGroups that have the selected NSGroup as a member.

Configuring Services and Service Groups

You can configure an NSService and specify parameters for matching network traffic such as a port and protocol pairing. You can also use an NSService to allow or block certain types of traffic in firewall rules.

An NSService can be of the following types:

- Ether
- IP
- IGMP
- ICMP
- ALG
- L4 Port Set

An L4 Port Set supports the identification of source ports and destination ports. You can specify individual ports or a range of ports, up to a maximum of 15 ports.

An NSService can also be a group of other NSServices. An NSService that is a group can be of the following types:

- Layer 2
- Layer 3 and above

You cannot change the type after you create an NSService. Some NSServices are predefined. You cannot modify or delete them.

Create an NSService

You can create an NSService to specify the characteristics that network matching uses, or to define the type of traffic to block or allow in firewall rules.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Inventory > Services > Add**.

3 Enter a name.

4 (Optional) Enter a description.

5 Select **Specify a protocol** to configure an individual service, or select **Group existing services** to configure a group of NSServices.

6 For an individual service, select a type of service and a protocol.

The available types are **Ether**, **IP**, **IGMP**, **ICMP**, **ALG**, and **L4 Port Set**

7 For a service group, select a type and members for the group.

The available types are **Layer 2** and **Layer 3 and above**.

8 Click **ADD**.

Manage Tags for a VM

You can see the list of VMs in the inventory. You can also add tags to a VM to make searching easier.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Inventory > Virtual Machines** from the navigation panel.

The list of VMs is displayed with 4 columns: Virtual Machine, External ID, Source, and Tag. Click the filter icon in the first three columns' heading to filter the list. Enter a string of characters to do a partial match. If the string in the column contains the string that you entered, the entry is displayed. Enter a string of characters enclosed in double quotes to do an exact match. If the string in the column exactly matches the string that you entered, the entry is displayed.

3 Select **Inventory > Virtual machines** from the navigation panel.

4 Select a VM.

5 Click **MANAGE TAGS**.

6 Add or delete tags.

Option	Action
Add a tag	Click **ADD** to specify a tag and optionally a scope.
Delete a tag	Select an existing tag and click **DELETE**.

The maximum number of tags that can be assigned from the NSX Manager to a virtual machine is 25. The maximum number of tags for all other managed objects such as logical switches or ports, is 30 .

7 Click **Save**.

17

Advanced DHCP

You can configure DHCP from the **Advanced Networking & Security** tab.

Note If you use the **Advanced Networking & Security** user interface to modify objects created in the policy interface, some settings might not be configurable. These read-only settings have this icon next to them: ⊖. See Chapter 1 Overview of the NSX Manager for more information.

This chapter includes the following topics:

- DHCP
- Metadata Proxies

DHCP

DHCP (Dynamic Host Configuration Protocol) allows clients to automatically obtain network configuration, such as IP address, subnet mask, default gateway, and DNS configuration, from a DHCP server.

You can create DHCP servers to handle DHCP requests and create DHCP relay services to relay DHCP traffic to external DHCP servers. However, you should not configure a DHCP server on a logical switch and also configure a DHCP relay service on a router port that the same logical switch is connected to. In such a scenario, DHCP requests will only go to the DHCP relay service.

If you configure DHCP servers, to improve security, configure a DFW rule to allow traffic on UDP ports 67 and 68 only for valid DHCP server IP addresses.

Note A DFW rule that has `Logical Switch/Logical Port/NSGroup` as the source, `Any` as the destination, and is configured to drop DHCP packets for ports 67 and 68, will fail to block DHCP traffic. To block DHCP traffic, configure `Any` as the source as well as the destination.

In this release, the DHCP server does not support guest VLAN tagging.

Create a DHCP Server Profile

A DHCP server profile specifies an NSX Edge cluster or members of an NSX Edge cluster. A DHCP server with this profile services DHCP requests from VMs on logical switches that are connected to the NSX Edge nodes that are specified in the profile.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > DHCP > Server Pofiles > Add**.

3 Enter a name and optional description.

4 Select an NSX Edge cluster from the drop-down menu.

5 (Optional) Select members of the NSX Edge cluster.

You can specify up to 2 members.

What to do next

Create a DHCP server. See Create a DHCP Server.

Create a DHCP Server

You can create DHCP servers to service DHCP requests from VMs that are connected to logical switches.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > DHCP > Servers > Add**.

3 Enter a name and optional description.

4 Enter the IP address of the DHCP server and its subnet mask in CIDR format.

For example, enter `192.168.1.2/24`.

5 (Required) Select a DHCP profile from the drop-down menu.

6 (Optional) Enter common options such as domain name, default gateway, DNS servers, and subnet mask.

7 (Optional) Enter classless static route options.

8 (Optional) Enter other options.

9 Click **Save**.

10 Select the newly created DHCP server.

11 Expand the IP Pools section.

12 Click **Add** to add IP ranges, default gateway, lease duration, warning threshold, error threshold, classless static route option, and other options.

13 Expand the Static Bindings section.

14 Click **Add** to add static bindings between MAC addresses and IP addresses, default gateway, hostname, lease duration, classless static route option, and other options.

What to do next

Attach a DHCP server to a logical switch. See Attach a DHCP Server to a Logical Switch.

Attach a DHCP Server to a Logical Switch

You must attach a DHCP server to a logical switch before the DHCP server can process DHCP requests from VMs connected to the switch.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Switching**.
   a. Click the checkbox of a logical switch.
   b. Click **Actions > Attach DHCP Server**.
3. Alternatively, select **Advanced Networking & Security > DHCP**.
   a. Click the **Servers** tab.
   b. Click the checkbox of a DHCP server.
   c. Click **Actions > Attach to Logical Switch**.

Detach a DHCP Server from a Logical Switch

You can detach a DHCP server from a logical switch to reconfigure your environment.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Switching**.
3. Click the logical switch that you intend to detach a DHCP server from.
4. Click **Actions > Detach DHCP Server**.

Create a DHCP Relay Profile

A DHCP relay profile specifies one or more external DHCP or DHCPv6 servers. When you create a DHCP/DHCPv6 relay service, you must specify a DHCP relay profile.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security** > **Networking** > **DHCP** > **Relay Profiles** > **Add**.

3 Enter a name and optional description.

4 Enter one or more external DHCP/DHCPv6 server addresses.

What to do next

Create a DHCP/DHCPv6 relay service. See Create a DHCP Relay Service.

Create a DHCP Relay Service

You can create a DHCP relay service to relay traffic between DHCP clients and DHCP servers that are not created in NSX-T Data Center.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security** > **Networking** > **DHCP** > **Relay Services** > **Add**.

3 Enter a name and optional description.

4 Select a DHCP relay profile from the drop-down menu.

What to do next

Add a DHCP service to a logical router port. See Add a DHCP Relay Service to a Logical Router Port.

Add a DHCP Relay Service to a Logical Router Port

You can add a DHCP relay service to a logical router port. VMs on the logical switch that is attached to that port can communicate with the DHCP servers that are configured in the relay service.

Prerequisites

- Verify you have a configured DHCP relay service. See Create a DHCP Relay Service.
- Verify that the router port is of type **Downlink**.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security** > **Networking** > **Routers**.

3 Select the appropriate router to display more information and configuration options.

4 Select **Configuration** > **Router Ports**.

5 Select the router port that connects to the desired logical switch and click **Edit**.

6 Select a DHCP relay service from the **Relay Service** drop-down list and click **Save**.

   You can also select a DHCP relay service when you add a new logical router port.

Delete a DHCP Lease

In some situations, you might want to delete a DHCP lease. For example, if you want a DHCP client to get a different IP address, or if a client shuts down without releasing its IP address and you want the address to be available to other clients.

You can use the following API to delete a DHCP lease:

```
DELETE /api/v1/dhcp/servers/<server-id>/leases?ip=<ip>&mac=<mac>
```

To ensure that the correct lease is deleted, call the following API before and after the `DELETE` API:

```
GET /api/v1/dhcp/servers/<server-id>/leases
```

After calling the `DELETE` API, make sure that the output of the `GET` API does not show the lease that was deleted.

For more information, see the *NSX-T Data Center API Reference*.

Metadata Proxies

With a metadata proxy server, VM instances can retrieve instance-specific metadata from an OpenStack Nova API server.

The following steps describe how a metadata proxy works:

1 A VM sends an HTTP GET to http://169.254.169.254:80 to request some metadata.

2 The metadata proxy server that is connected to the same logical switch as the VM reads the request, makes appropriate changes to the headers, and forwards the request to the Nova API server.

3 The Nova API server requests and receives information about the VM from the Neutron server.

4 The Nova API server finds the metadata and sends it to the metadata proxy server.

5 The metadata proxy server forwards the metadata to the VM.

A metadata proxy server runs on an NSX Edge node. For high availability, you can configure metadata proxy to run on two or more NSX Edge nodes in an NSX Edge cluster.

Add a Metadata Proxy Server

A metadata proxy server enables VMs to retrieve metadata from an OpenStack Nova API server.

Prerequisites

Verify that you have created an NSX Edge cluster. For more information, see *NSX-T Data Center Installation Guide*.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security** > **Networking** > **DHCP** > **Metadata Proxies** > **Add** .
3. Enter a name for the metadata proxy server.
4. (Optional) Enter a description.
5. Enter the URL and port for the Nova server.

   The valid port range is 3000 - 9000.
6. Enter a value for **Secret**.
7. Select an NSX Edge cluster from the drop-down list.
8. (Optional) Select members of the NSX Edge cluster.

What to do next

Attach the metadata proxy server to a logical switch.

Attach a Metadata Proxy Server to a Logical Switch

To provide metadata proxy services to VMs that are connected to a logical switch, you must attach a metadata proxy server to the switch.

Prerequisites

Verify that you have created a logical switch. For more information, see Create a Logical Switch.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security** > **Networking** > **DHCP** > **Metadata Proxies**.
3. Select a metadata proxy server.
4. Select the menu option **Actions** > **Attach to Logical Switch**
5. Select a logical switch from the drop-down list.

Results

You can also attach a metadata proxy server to a logical switch by navigating to **Switching** > **Switches**, selecting a switch, and selecting the menu option **Actions** > **Attach Metadata Proxy**.

Detach a Metadata Proxy Server from a Logical Switch

To stop providing metadata proxy services to VMs that are connected to a logical switch or use a different metadata proxy server, you can detach a metadata proxy server from a logical switch.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > DHCP > Metadata Proxies**.
3. Select a metadata proxy server.
4. Select the menu option **Actions > Detach from Logical Switch**
5. Select a logical switch from the drop-down list.

Results

You can also detach a metadata proxy server from a logical switch by navigating to **Switching > Switches**, selecting a switch, and selecting the menu option **Actions > Detach Metadata Proxy**.

Advanced IP Address Management

18

With IP address management (IPAM), you can create IP blocks to support NSX Container Plug-in (NCP). For more info about NCP, see the *NSX-T Container Plug-in for Kubernetes - Installation and Administration Guide*.

Note If you use the **Advanced Networking & Security** user interface to modify objects created in the policy interface, some settings might not be configurable. These read-only settings have this icon next to them: ⊖. See Chapter 1 Overview of the NSX Manager for more information.

This chapter includes the following topics:

- Manage IP Blocks
- Manage Subnets for IP Blocks

Manage IP Blocks

Setting up NSX Container Plug-in requires that you create IP blocks for the containers.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > IPAM**.
3. To add an IP block, click **Add**.
    a. Enter a name and optionally a description.
    b. Enter an IP block in CIDR format. For example, 10.10.10.0/24.
4. To edit an IP block, click the name of an IP block.
    a. In the **Overview** tab, click **Edit**.

       You can change the name, description, or the IP block value.

5 To manage the tags of an IP block, click the name of an IP block.

   a In the **Overview** tab, click **Manage**.

      You can add or delete tags.

6 To delete one or more IP blocks, select the blocks.

   a Click **Delete**.

      You cannot delete an IP block that has its subnet allocated.

Manage Subnets for IP Blocks

You can add or delete subnets for IP blocks.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > IPAM**.

3 Click the name of an IP block.

4 Click the **Subnets** tab.

5 To add a subnet, click **Add**.

   a Enter a name and optionally a description.

   b Enter the size of the subnet.

6 To delete one or more subnets, select the subnets.

   a Click **Delete**.

Advanced Load Balancing

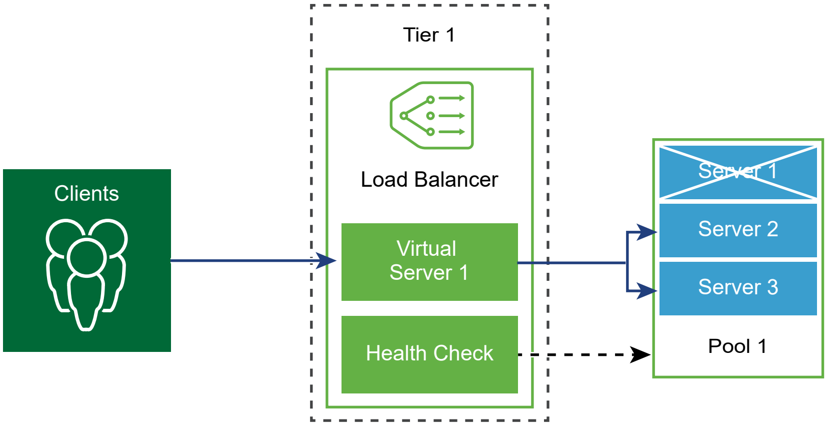
19

This information covers the NSX-T Data Center load balancing configuration found under the **Advanced Networking & Security** tab.

For information about NSX Advanced Load Balancer (Avi Networks) see https://www.vmware.com/products/nsx-advanced-load-balancer.html.

Note If you use the **Advanced Networking & Security** user interface to modify objects created in the policy interface, some settings might not be configurable. These read-only settings have this icon next to them: ⊖. See Chapter 1 Overview of the NSX Manager for more information.

The NSX-T Data Center logical load balancer offers high-availability service for applications and distributes the network traffic load among multiple servers.



The load balancer distributes incoming service requests evenly among multiple servers in such a way that the load distribution is transparent to users. Load balancing helps in achieving optimal resource utilization, maximizing throughput, minimizing response time, and avoiding overload.

You can map a virtual IP address to a set of pool servers for load balancing. The load balancer accepts TCP, UDP, HTTP, or HTTPS requests on the virtual IP address and decides which pool server to use.

Depending on your environment needs, you can scale the load balancer performance by increasing the existing virtual servers and pool members to handle heavy network traffic load.

Note Logical load balancer is supported only on the Tier-1 logical router. One load balancer can be attached only to a Tier-1 logical router.

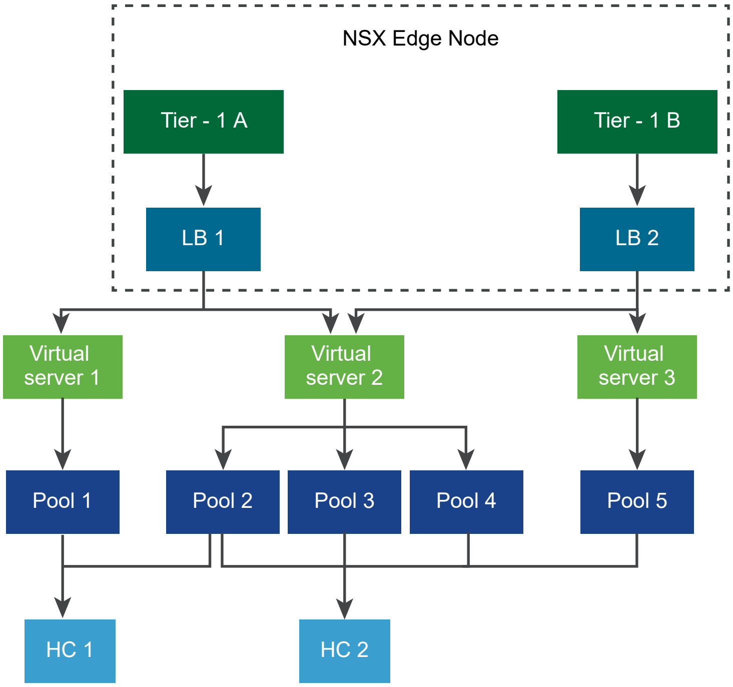
This chapter includes the following topics:

Key Load Balancer Concepts

Load balancer includes virtual servers, server pools, and health checks monitors.

A load balancer is connected to a Tier-1 logical router. The load balancer hosts single or multiple virtual servers. A virtual server is an abstract of an application service, represented by a unique combination of IP, port, and protocol. The virtual server is associated to single to multiple server pools. A server pool consists of a group of servers. The server pools include individual server pool members.

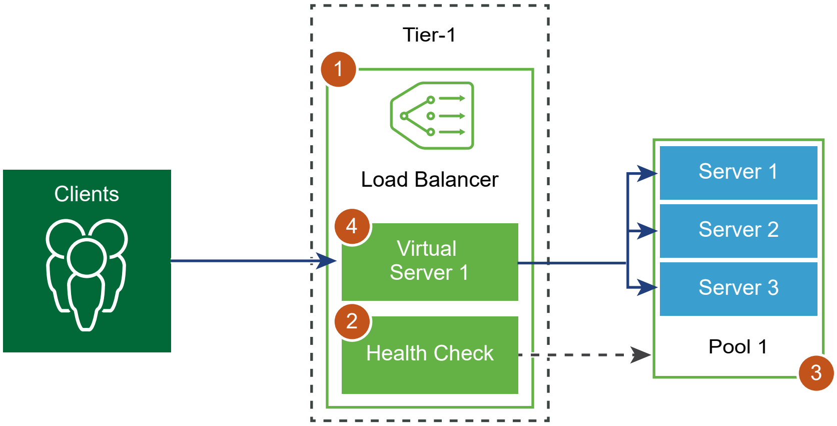
To test whether each server is correctly running the application, you can add health check monitors that check the health status of a server.



Configuring Load Balancer Components

To use logical load balancers, you must start by configuring a load balancer and attaching it to a Tier-1 logical router.

Next, you can set up health check monitoring for your servers. You must then configure server pools for your load balancer. Finally, you must create a layer 4 or layer 7 virtual server for your load balancer.

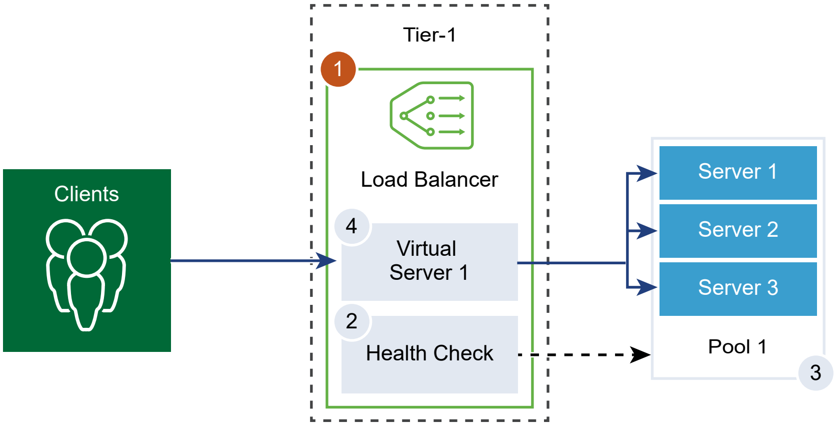


Create a Load Balancer

Load balancer is created and attached to the Tier-1 logical router.

You can configure the level of error messages you want the load balancer to add to the error log.

Note Avoid setting the log level to DEBUG on load balancers with significant traffic due to the number of messages printed to the log that affect performance.



Prerequisites

Verify that a Tier-1 logical router is configured. See Create a Tier-1 Logical Router .

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Load Balancer > Add**.

3 Enter a name and a description for the load balancer.

4 Select the load balancer virtual server size and number of pool members based on your available resources.

5 Define the severity level of the error log from the drop-down menu.

   Load balancer collects information about encountered issues of different severity levels to the error log.

6 Click **OK**.

7 Associate the newly created load balancer to a virtual server.

   a Select the load balancer and click **Actions > Attach to a Virtual Server**.

   b Select an existing virtual server from the drop-down menu.

   c Click **OK**.

8 Attach the newly created load balancer to a Tier-1 logical router.

   a Select the load balancer and click **Actions > Attach to a Logical Router**.

   b Select an existing Tier-1 logical router from the drop-down menu.

      The Tier-1router must be in the Active-Standby mode.

   c Click **OK**.

9 (Optional) Delete the load balancer.

   If you no longer want to use this load balancer, you must first detach the load balancer from the virtual server and Tier-1 logical router.

Configure an Active Health Monitor

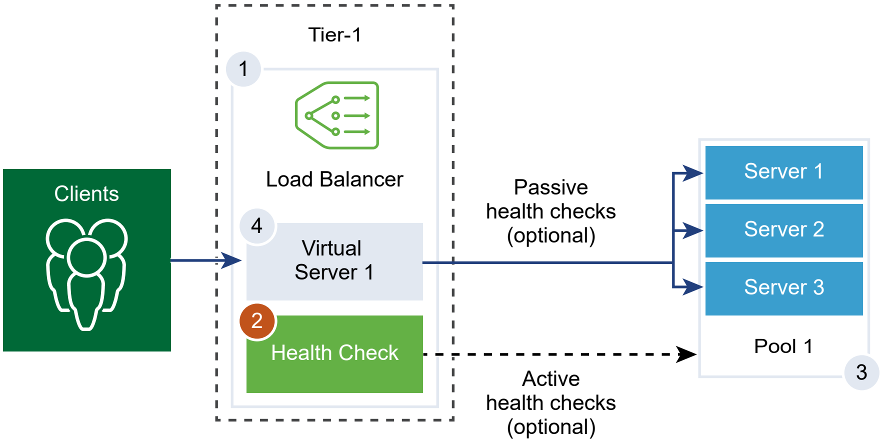
The active health monitor is used to test whether a server is available. The active health monitor uses several types of tests such as sending a basic ping to servers or advanced HTTP requests to monitor the application health.

Servers that fail to respond within a certain time period or respond with errors are excluded from future connection handling until a subsequent periodic health check finds these servers to be healthy.

Active health checks are performed on server pool members after the pool member is attached to a virtual server and that virtual server is attached to a Tier-1 gateway (previously called a Tier-1 logical router).

If the Tier-1 gateway is connected to a Tier-0 gateway, a router link port is created and its IP address (typically in the `100.64.x.x` format) is used to perform the health check for the load balancer service. If the Tier-1 gateway is standalone (has only one centralized service port and is not connected to a Tier-0 gateway), the centralized service port IP address is used to perform the health check for the load balancer service. See Create a Standalone Tier-1 Logical Router for information about standalone Tier-1 gateways.

Note One active health monitor can be configured per server pool.



Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Load Balancer > Monitors > Active Health Monitors > Add**.
3. Enter a name and description for the active health monitor.
4. Select a health check protocol for the server from the drop-down menu.

   You can also use predefined protocols in NSX Manager; `http-monitor`, `https-monitor`, `Icmp-monitor`, `Tcp-monitor`, and `Udp-monitor`.
5. Set the value of the monitoring port.
6. Configure the values to monitor a service pool.

   You can also accept the default active health monitor values.

Option	Description
Monitoring Interval	Set the time in seconds that the monitor sends another connection request to the server.
Fall Count	Set a value when the consecutive failures reach this value, the server is considered temporarily unavailable.

Option	Description
Rise Count	Set a number after this timeout period, the server is tried again for a new connection to see if it is available.
Timeout Period	Set the number of times the server is tested before it is considered as DOWN.

For example, if the monitoring interval is set as 5 seconds and the timeout as 15 seconds, the load balancer send requests to the server every 5 seconds. In each probe, if the expected response is received from the server within 15 seconds, then the health check result is OK. If not, then the result is CRITICAL. If the recent three health check results are all UP, the server is considered as UP.

7 If you select HTTP as the health check protocol, complete the following details.

Option	Description
HTTP Method	Select the method for detecting the server status from the drop-down menu, GET, OPTIONS, POST, HEAD, and PUT.
HTTP Request URL	Enter the request URI for the method.
HTTP Request Version	Select the supported request version from the drop-down menu. You can also accept the default version, HTTP_VERSION_1_1.
HTTP Request Body	Enter the request body. Valid for the POST and PUT methods.
HTTP Response Code	Enter the string that the monitor expects to match in the status line of HTTP response body. The response code is a comma-separated list. For example, 200,301,302,401.
HTTP Response Body	If the HTTP response body string and the HTTP health check response body match, then the server is considered as healthy.

8 If you select HTTPs as the health check protocol, complete the following details.

   a Select the SSL protocol list.

   TLS versions TLS1.1 and TLS1.2 versions are supported and enabled by default. TLS1.0 is supported, but disabled by default.

   b Click the arrow and move the protocols into the selected section.

c Assign a default SSL cipher or create a custom SSL cipher.

d Complete the following details for HTTP as the health check protocol.

Option	Description
HTTP Method	Select the method for detecting the server status from the drop-down menu: `GET`, `OPTIONS`, `POST`, `HEAD`, and `PUT`.
HTTP Request URL	Enter the request URI for the method.
HTTP Request Version	Select the supported request version from the drop-down menu. You can also accept the default version, HTTP_VERSION_1_1.
HTTP Request Body	Enter the request body. Valid for the POST and PUT methods.
HTTP Response Code	Enter the string that the monitor expects to match in the status line of HTTP response body. The response code is a comma-separated list. For example, 200,301,302,401.
HTTP Response Body	If the HTTP response body string and the HTTP health check response body match, then the server is considered as healthy.

9 If you select ICMP as the health check protocol, assign the data size in byte of the ICMP health check packet.

10 If you select TCP as the health check protocol, you can leave the parameters empty.

If both the sent and expected are not listed, then a three-way handshake TCP connection is established to validate the server health. No data is sent. Expected data if listed has to be a string and can be anywhere in the response. Regular expressions are not supported.

11 If you select UDP as the health check protocol, complete the following required details.

Required Option	Description
UDP Data Sent	Enter the string to be sent to a server after a connection is established.
UDP Data Expected	Enter the string expected to receive from the server. Only when the received string matches this definition, is the server is considered as UP.

12 Click **Finish**.

What to do next

Associate the active health monitor with a server pool. See Add a Server Pool for Load Balancing.

Configure Passive Health Monitors

Load balancers perform passive health checks to monitor failures during client connections and mark servers causing consistent failures as DOWN.

Passive health check monitors client traffic going through the load balancer for failures. For example, if a pool member sends a TCP Reset (RST) in response to a client connection, the load balancer detects that failure. If there are multiple consecutive failures, then the load balancer considers that server pool member to be temporarily unavailable and stops sending connection requests to that pool member for some time. After some time, the load balancer sends a connection request to check if the pool member has recovered. If that connection is successful, then the pool member is considered healthy. Otherwise, the load balancer waits for some time and tries again.

Passive health check considers the following scenarios to be failures in client traffic.

- For server pools associated with Layer 7 virtual servers, if the connection to the pool member fails. For example, if the pool member sends a TCP RST when the load balancer tries to connect or perform a SSL handshake between load balancer and the pool member fails.
- For server pools associated with Layer 4 TCP virtual servers, if the pool member sends a TCP RST in response to client TCP SYN or does not respond at all.
- For server pools associated with Layer 4 UDP virtual servers, if a port is unreachable or a destination unreachable ICMP error message is received in response to a client UDP packet.

Server pools associated to Layer 7 virtual servers, the failed connection count is incremented when any TCP connection errors, for example, TCP RST failure to send data or SSL handshake failures occur.

Server pools associated to Layer 4 virtual servers, if no response is received to a TCP SYN sent to the server pool member or if a TCP RST is received in response to a TCP SYN, then the server pool member is considered as DOWN. The failed count is incremented.

For Layer 4 UDP virtual servers, if an ICMP error such as, port or destination unreachable message is received in response to client traffic, then it is considered as DOWN.

Note One passive health monitor can be configured per server pool.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Load Balancer > Monitors > Passive Health Monitors > Add**.
3. Enter a name and description for the passive health monitor.

4 Configure the values to monitor a service pool.

You can also accept the default active health monitor values.

Option	Description
Fall Count	Set a value when the consecutive failures reach this value, the server is considered temporarily unavailable.
Timeout Period	Set the number of times the server is tested before it is considered as DOWN.

For example, when the consecutive failures reach the configured value 5, that member is considered temporarily unavailable for 5 seconds. After this period, that member is tried again for a new connection to see if it is available. If that connection is successful, then the member is considered available and the failed count is set to zero. However, if that connection fails, then it is not used for another timeout interval of 5 seconds.

5 Click **OK**.

What to do next

Associate the passive health monitor with a server pool. See Add a Server Pool for Load Balancing.

Add a Server Pool for Load Balancing

Server pool consists of one or more servers that are configured and running the same application. A single pool can be associated to both Layer 4 and Layer 7 virtual servers.

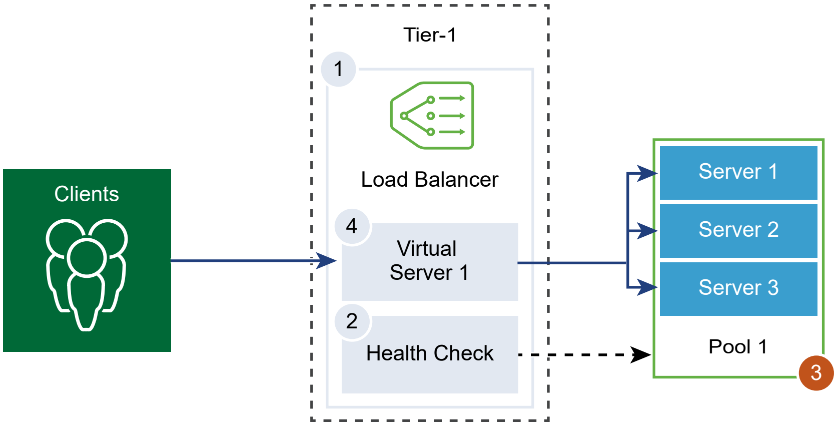
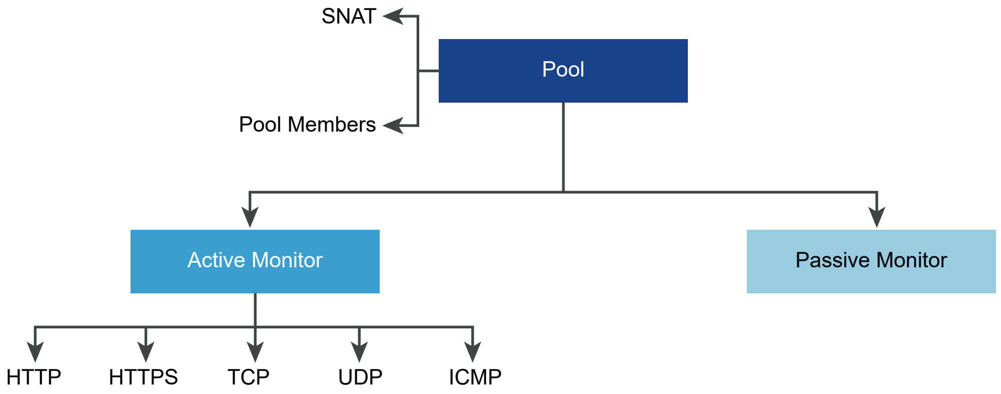


Figure 19-1. Server Pool Parameter Configuration



Prerequisites

- If you use dynamic pool members, a NSGroup must be configured. See Create an NSGroup.
- Depending on the monitoring you use, verify that active or passive health monitors are configured. See Configure an Active Health Monitor or Configure Passive Health Monitors.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security** > **Networking** > **Load Balancer** > **Server Pools** > **Add**.
3. Enter a name and description for the load balancer pool.

   You can optionally describe the connections managed by the server pool.
4. Select the algorithm balancing method for the server pool.

   Load balancing algorithm controls how the incoming connections are distributed among the members. The algorithm can be used on a server pool or a server directly.

   All load balancing algorithms skip servers that meet any of the following conditions:

   - Admin state is set to DISABLED.
   - Admin state is set to GRACEFUL_DISABLED and no matching persistence entry.
   - Active or passive health check state is DOWN.

- Connection limit for the maximum server pool concurrent connections is reached.

Option	Description
ROUND_ROBIN	Incoming client requests are cycled through a list of available servers capable of handling the request. Ignores the server pool member weights even if they are configured.
WEIGHTED_ROUND_ROBIN	Each server is assigned a weight value that signifies how that server performs relative to other servers in the pool. The value determines how many client requests are sent to a server compared to other servers in the pool. This load balancing algorithm focuses on fairly distributing the load among the available server resources.
LEAST_CONNECTION	Distributes client requests to multiple servers based on the number of connections already on the server. New connections are sent to the server with the fewest connections. Ignores the server pool member weights even if they are configured.
WEIGHTED_LEAST_CONNECTION	Each server is assigned a weight value that signifies how that server performs relative to other servers in the pool. The value determines how many client requests are sent to a server compared to other servers in the pool. This load balancing algorithm focuses on using the weight value to distribute the load among the available server resources fairly. By default, the weight value is 1 if the value is not configured and slow start is enabled.
IP-HASH	Selects a server based on a hash of the source IP address and the total weight of all the running servers.

5 Toggle the TCP Multiplexing button to enable this menu item.

With TCP multiplexing, you can use the same TCP connection between a load balancer and the server for sending multiple client requests from different client TCP connections.

6 Set the maximum number of TCP multiplexing connections per pool that are kept alive to send future client requests.

7 Select the Source NAT (SNAT) mode.

Depending on the topology, SNAT might be required so that the load balancer receives the traffic from the server destined to the client. SNAT can be enabled per server pool.

Mode	Description
Transparent Mode	Load balancer uses the client IP address and port spoofing while establishing connections to the servers. SNAT is not required.
Auto Map Mode	Load Balancer uses the interface IP address and ephemeral port to continue the communication with a client initially connected to one of the server's established listening ports. SNAT is required. Enable port overloading to allow the same SNAT IP and port to be used for multiple connections if the tuple (source IP, source port, destination IP, destination port, and IP protocol) is unique after the SNAT process is performed. You can also set the port overload factor to allow the maximum number of times a port can be used simultaneously for multiple connections.
IP List Mode	Specify a single IP address range, for example, 1.1.1.1-1.1.1.10 to be used for SNAT while connecting to any of the servers in the pool. By default, from 4000 through 64000 port range is used for all configured SNAT IP addresses. Port ranges from 1000 through 4000 are reserved for purposes such as, health checks and connections initiated from Linux applications. If multiple IP addresses are present, then they are selected in a Round Robin manner. Enable port overloading to allow the same SNAT IP and port to be used for multiple connections if the tuple (source IP, source port, destination IP, destination port, and IP protocol) is unique after the SNAT process is performed. You can also set the port overload factor to allow the maximum number of times a port can be used simultaneously for multiple connections.

8 Select the server pool members.

Server pool consists of single or multiple pool members. Each pool member has an IP address and a port.

Each server pool member can be configured with a weight for use in the load balancing algorithm. The weight indicates how much more or less load a given pool member can handle relative to other members in the same pool.

Designating a pool member as a backup member works with the health monitor to provide an active/standby state. If active members fail a health check, traffic failover occurs for backup members.

Option	Description
Static	Click **Add** to include a static pool member. You can also clone an existing static pool member.
Dynamic	Select the NSGroup from the drop-down menu. The server pool membership criteria is defined in the group. You can optionally, define the maximum group IP address list.

9 Enter the minimum number of active members the server pool must always maintain.

10 Select an active and passive health monitor for the server pool from the drop-down menu.

Setting an active and passive health monitor for the server pool is optional. When you select an active health monitor and if the Tier-1 gateway is connected to a Tier-0 gateway, a router link port is created. The router link port's IP address (typically in the `100.64.x.x` format) is used to perform the health check for the load balancer service. If the Tier-1 gateway is standalone (has only one centralized service port and is not connected to a Tier-0 gateway), the centralized service port IP address is used to perform the health check for the load balancer service. See Create a Standalone Tier-1 Logical Router for information about standalone Tier-1 gateways.

Add a firewall rule to allow the IP address to perform the health check for the load balancer service.

11 Click **Finish.**

Configuring Virtual Server Components

With the virtual server there are several components that you can configure such as, application profiles, persistent profiles, and load balancer rules.

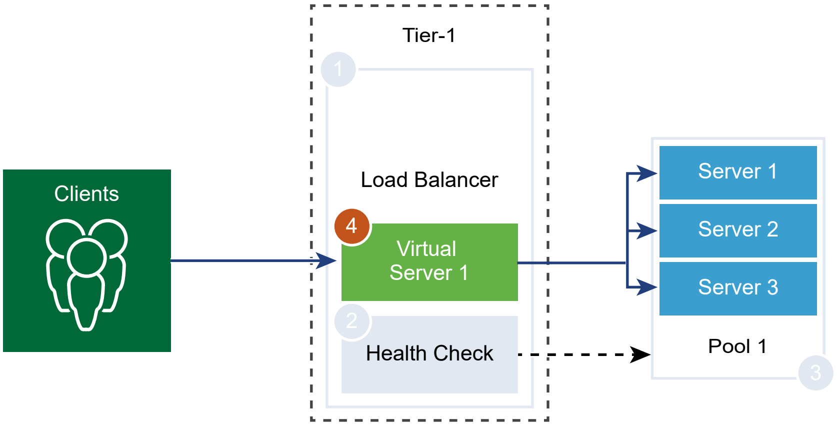
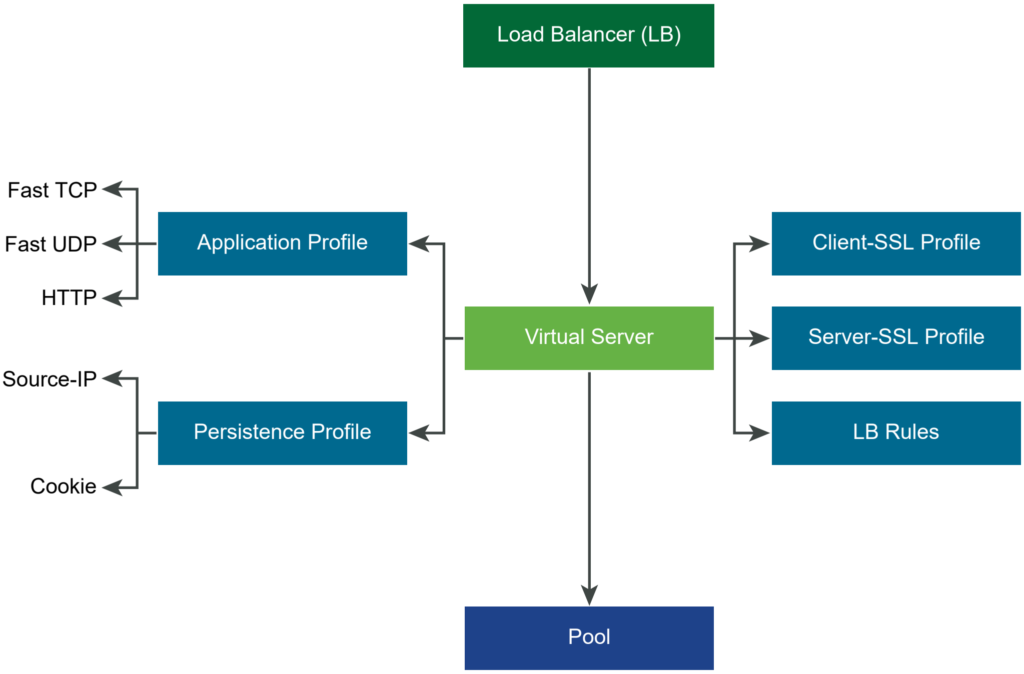


Figure 19-2. Virtual Server Components



Configure Application Profiles

Application profiles are associated with virtual servers to enhance load balancing network traffic and simplify traffic-management tasks.

Application profiles define the behavior of a particular type of network traffic. The associated virtual server processes network traffic according to the values specified in the application profile. Fast TCP, Fast UDP, and HTTP application profiles are the supported types of profiles.

TCP application profile is used by default when no application profile is associated to a virtual server. TCP and UDP application profiles are used when an application is running on a TCP or UDP protocol and does not require any application level load balancing such as, HTTP URL load balancing. These profiles are also used when you only want Layer 4 load balancing, which has faster performance and supports connection mirroring.

HTTP application profile is used for both HTTP and HTTPS applications when the load balancer needs to take actions based on Layer 7 such as, load balancing all images requests to a specific server pool member or terminating HTTPS to offload SSL from pool members. Unlike the TCP application profile, the HTTP application profile terminates the client TCP connection before selecting the server pool member.

Figure 19-3. Layer 4 TCP and UDP Application Profile

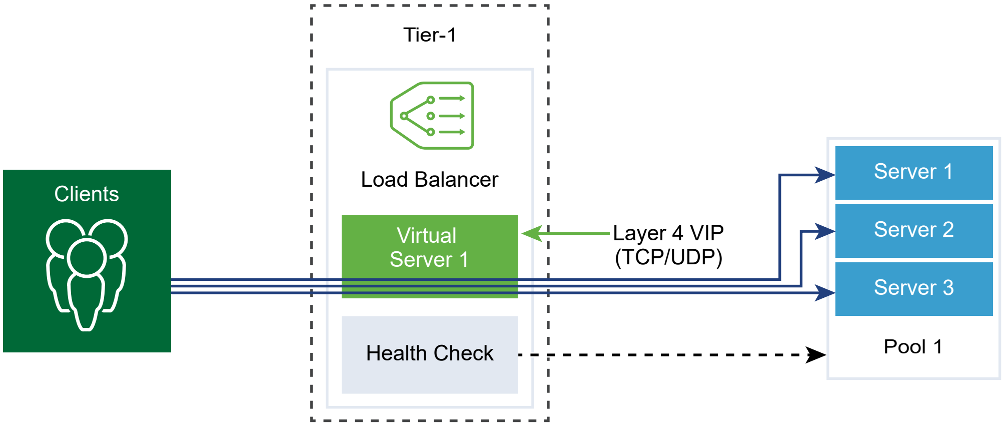
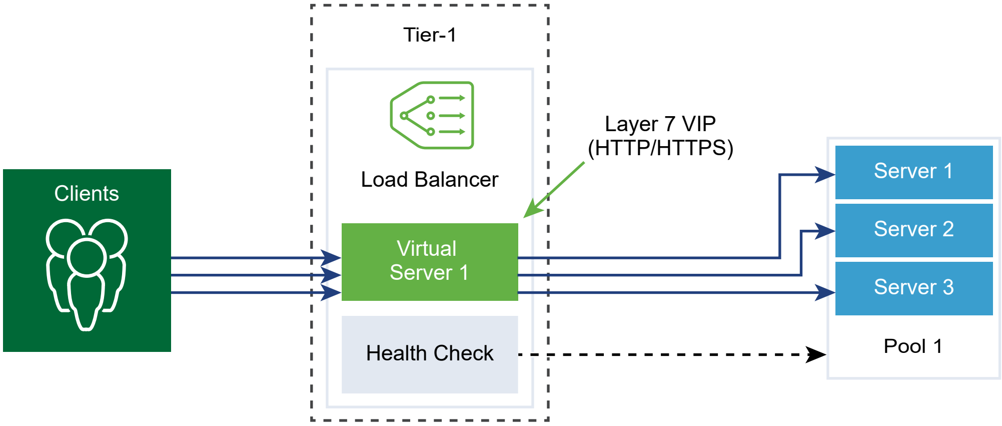


Figure 19-4. Layer 7 HTTPS Application Profile



Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Load Balancer > Profiles > Application Profiles**.
3. Create a Fast TCP application profile.
   a. Select **Add > Fast TCP Profile** from the drop-down menu.
   b. Enter a name and a description for the Fast TCP application profile.

c Complete the application profile details.

You can also accept the default FAST TCP profile settings.

Option	Description
Connection Idle Timeout	Enter the time in seconds on how long the server can remain idle after a TCP connection is established. Set the idle time to the actual application idle time and add a few more seconds so that the load balancer does not close its connections before the application does.
Connection Close Timeout	Enter the time in seconds that the TCP connection both FINs or RST must be kept for an application before closing the connection. A short closing timeout might be required to support fast connection rates.
HA Flow Mirroring	Toggle the button to make all the flows to the associated virtual server mirrored to the HA standby node.

d Click **OK**.

4 Create a Fast UDP application profile.

You can also accept the default UDP profile settings.

a Select **Add > Fast UDP Profile** from the drop-down menu.

b Enter a name and a description for the Fast UDP application profile.

c Complete the application profile details.

Option	Description
Idle Timeout	Enter the time in seconds on how long the server can remain idle after a UDP connection is established. UDP is a connectionless protocol. For load balancing purposes, all the UDP packets with the same flow signature such as, source and destination IP address or ports and IP protocol received within the idle timeout period are considered to belong to the same connection and sent to the same server. If no packets are received during the idle timeout period, the connection which is an association between the flow signature and the selected server is closed.
HA Flow Mirroring	Toggle the button to make all the flows to the associated virtual server mirrored to the HA standby node.

d Click **OK**.

5 Create an HTTP application profile.

You can also accept the default HTTP profile settings.

HTTP application profile is used for both HTTP and HTTPS applications.

a Select **Add > Fast HTTP Profile** from the drop-down menu.

b Enter a name and a description for the HTTP application profile.

c Complete the application profile details.

Option	Description
Redirection	■ None - If a website is temporarily down, user receives a page not found error message. ■ HTTP Redirect - If a website is temporarily down or has moved, incoming requests for that virtual server can be temporarily redirected to a URL specified here. Only a static redirection is supported. For example, if HTTP Redirect is set to http://sitedown.abc.com/sorry.html, then irrespective of the actual request, for example, http://original_app.site.com/home.html or http://original_app.site.com/somepage.html, incoming requests are redirected to the specified URL when the original website is down. ■ HTTP to HTTPS Redirect - Certain secure applications might want to force communication over SSL, but instead of rejecting non-SSL connections, they can redirect the client request to use SSL. With HTTP to HTTPS Redirect, you can preserve both the host and URI paths and redirect the client request to use SSL. For HTTP to HTTPS redirect, the HTTPS virtual server must have port 443 and the same virtual server IP address must be configured on the same load balancer. For example, a client request for http://app.com/path/page.html is redirected to https://app.com/path/page.html. If either the host name or the URI must be modified while redirecting, for example, redirect to https://secure.app.com/path/page.html, then load balancing rules must be used.
X-Forwarded-For (XFF)	■ **Insert** - If the XFF HTTP header is not present in the incoming request, the load balancer inserts a new XFF header with the client IP address. If the XFF HTTP header is present in the incoming request, the load balancer appends the XFF header with the client IP address. ■ **Replace** - If the XFF HTTP header is present in the incoming request, the load balancer replaces the header. Web servers log each request they handle with the requesting client IP address. These logs are used for debugging and analytics purposes. If the deployment topology requires SNAT on the load balancer, then server uses the client SNAT IP address which defeats the purpose of logging. As a workaround, the load balancer can be configured to insert XFF HTTP header with the original client IP address. Servers can be configured to log the IP address in the XFF header instead of the source IP address of the connection.
Connection Idle Timeout	Enter the time in seconds on how long an HTTP application can remain idle, instead of the TCP socket setting which must be configured in the TCP application profile.
Request Header Size	Specify the maximum buffer size in bytes used to store HTTP request headers.
NTLM Authentication	Toggle the button for the load balancer to turn off TCP multiplexing and enable HTTP keep-alive.

Option	Description
	NTLM is an authentication protocol that can be used over HTTP. For load balancing with NTLM authentication, TCP multiplexing must be disabled for the server pools hosting NTLM-based applications. Otherwise, a server-side connection established with one client's credentials can potentially be used for serving another client's requests. If NTLM is enabled in the profile and associated to a virtual server, and TCP multiplexing is enabled at the server pool, then NTLM takes precedence. TCP multiplexing is not performed for that virtual server. However, if the same pool is associated to another non-NTLM virtual server, then TCP multiplexing is available for connections to that virtual server. If the client uses HTTP/1.0, the load balancer upgrades to HTTP/1.1 protocol and the HTTP keep-alive is set. All HTTP requests received on the same client-side TCP connection are sent to the same server over a single TCP connection to ensure that reauthorization is not required.

d Click **OK**.

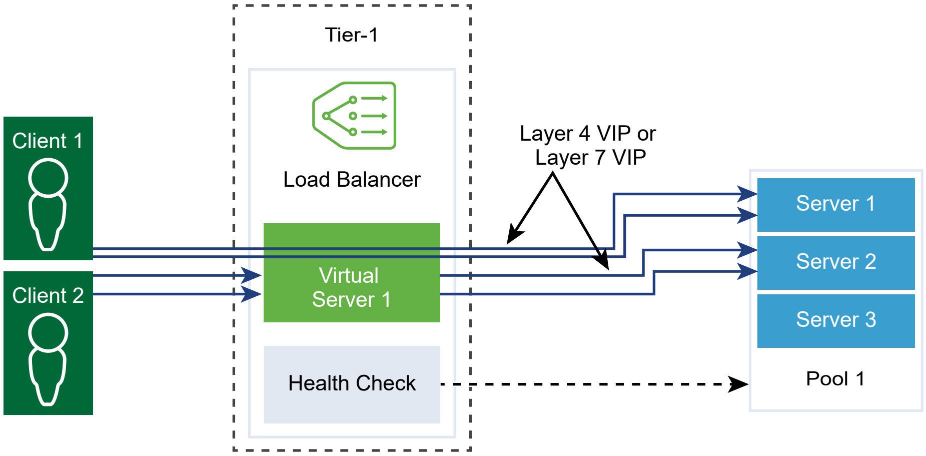
Configure Persistent Profiles

To ensure stability of stateful applications, load balancers implement persistence which directs all related connections to the same server. Different types of persistence are supported to address different types of application needs.

Some applications maintain the server state such as, shopping carts. Such state might be per client and identified by the client IP address or per HTTP session. Applications might access or modify this state while processing subsequent related connections from the same client or HTTP session.

Source IP persistence profile tracks sessions based on the source IP address. When a client requests a connection to a virtual server that enables the source address persistence, the load balancer checks if that client was previously connected, if so, returns the client to the same server. If not, you can select a server pool member based on the pool load balancing algorithm. Source IP persistence profile is used by Layer 4 and Layer 7 virtual servers.

Cookie persistence profile inserts a unique cookie to identify the session the first time a client accesses the site. The HTTP cookie is forwarded by the client in subsequent requests and the load balancer uses that information to provide the cookie persistence. Cookie persistence profile can only be used by Layer 7 virtual servers.



Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Load Balancer > Profiles > Persistence Profiles**.

3 Create a Source IP persistence profile.

   a Select **Add > Source IP Persistence** from the drop-down menu.

   b Enter a name and a description for the Source IP persistence profile.

c Complete the persistence profile details.

You can also accept the default Source IP profile settings.

Option	Description
Share Persistence	Toggle the button to share the persistence so that all virtual servers this profile is associated with can share the persistence table. If persistence sharing is not enabled in the Source IP persistence profile associated to a virtual server, each virtual server that the profile is associated to maintain a private persistence table.
Persistence Entry Timeout	Enter the persistence expiration time in seconds. The load balancer persistence table maintains entries to record that client requests are directed to the same server. ■ If no new connection requests are received from the same client within the timeout period, the persistence entry expires and is deleted. ■ If a new connection request from the same client is received within the timeout period, the timer is reset, and the client request is sent to a sticky pool member. After the timeout period has expired, new connection requests are sent to a server allocated by the load balancing algorithm. For the L7 load balancing TCP source IP persistence scenario, the persistence entry times out if no new TCP connections are made for some time, even if the existing connections are still alive.
HA Persistence Mirroring	Toggle the button to synchronize persistence entries to the HA peer.
Purge Entries When Full	Purge entries when the persistence table is full. A large timeout value might lead to the persistence table quickly filling up when the traffic is heavy. When the persistence table fills up, the oldest entry is deleted to accept the newest entry.

d Click **OK**.

4 Create a Cookie persistence profile.

a Select **Add > Cookie Persistence** from the drop-down menu.

b Enter a name and a description for the Cookie persistence profile.

c Toggle the **Share Persistence** button to share persistence across multiple virtual servers that are associated to the same pool members.

The Cookie persistence profile inserts a cookie with the format, *<name>.<profile-id>.<pool-id>*.

If the persistence shared is not enabled in the Cookie persistence profile associated with a virtual server, the private Cookie persistence for each virtual server is used and is qualified by the pool member. The load balancer inserts a cookie with the format, *<name>.<virtual_server_id>.<pool_id>*.

d Click **Next**.

e Complete the persistence profile details.

Option	Description
Cookie Mode	Select a mode from the drop-down menu. ■ INSERT - Adds a unique cookie to identify the session. ■ PREFIX - Appends to the existing HTTP cookie information. ■ REWRITE - Rewrites the existing HTTP cookie information.
Cookie Name	Enter the cookie name.
Cookie Domain	Enter the domain name. HTTP cookie domain can be configured only in the INSERT mode.
Cookie Path	Enter the cookie URL path. HTTP cookie path can be set only in the INSERT mode.
Cookie Garbling	Encrypt the cookie server IP address and port information. Toggle the button to disable encryption. When garbling is disabled, the cookie server IP address and port information is in a plain text.
Cookie Fallback	Select a new server to handle a client request if the cookie points to a server that is in a DISABLED or is in a DOWN state. Toggle the button so that the client request is rejected if cookie points to a server that is in a DISABLED or is in a DOWN state.

f Complete the Cookie expiry details.

Option	Description
Cookie Time Type	Select a cookie time type from the drop-down menu. **Session Cookie** is not stored and will be lost when the browser is closed. **Persistence Cookie** is stored by the browser and is not lost when the browser is closed.
Maximum Idle Time	Enter the time in seconds that a cookie can be idle before it expires.
Maximum Cookie Age	For **Session Cookie** only. Enter the maximum age in seconds that a cookie can be active.

g Click **Finish**.

Configure SSL Profile

SSL profiles configure application-independent SSL properties such as, cipher lists and reuse these lists across multiple applications. SSL properties are different when the load balancer is acting as a client and as a server, as a result separate SSL profiles for client-side and server-side are supported.

Note SSL profile is not supported in the NSX-T Data Center limited export release.

Client-side SSL profile refers to the load balancer acting as an SSL server and terminating the client SSL connection. Server-side SSL profile refers to the load balancer acting as a client and establishing a connection to the server.

You can specify a cipher list on both the client-side and server-side SSL profiles.

SSL session caching allows the SSL client and server to reuse previously negotiated security parameters avoiding the expensive public key operation during the SSL handshake. SSL session caching is disabled by default on both the client-side and server-side.

SSL session tickets are an alternate mechanism that allow the SSL client and server to reuse previously negotiated session parameters. In SSL session tickets, the client and server negotiate whether they support SSL session tickets during the handshake exchange. If supported by both, server can send an SSL ticket, which includes encrypted SSL session parameters to the client. The client can use that ticket in subsequent connections to reuse the session. SSL session tickets are enabled on the client-side and disabled on the server-side.

Figure 19-5. SSL Offloading

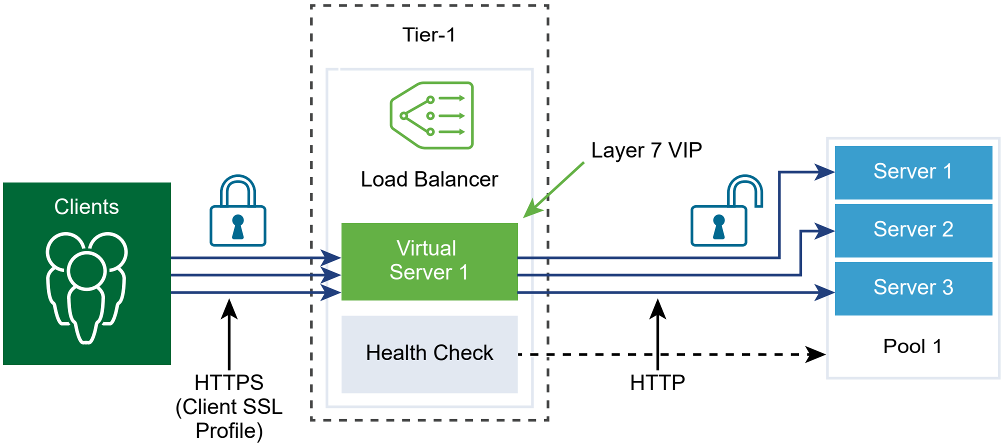
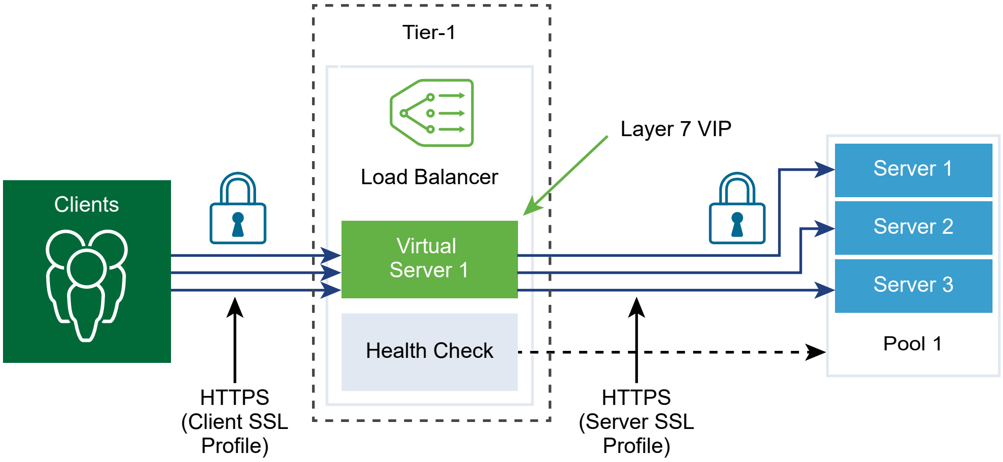


Figure 19-6. End-to-End SSL



Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Load Balancer > Profiles > SSL Profiles**.

3 Create a Client SSL profile.

   a Select **Add > Client Side SSL** from the drop-down menu.

   b Enter a name and a description for the Client SSL profile.

   c Assign the SSL Ciphers to be included in the Client SSL profile.

   You can also create custom SSL Ciphers.

   d Click the arrow to move the ciphers to the Selected section.

   e Click the **Protocols and Sessions** tab.

   f Select the SSL protocols to be included in the Client SSL profile.

   SSL protocol versions TLS1.1 and TLS1.2 are enabled by default. TLS1.0 is also supported, but disabled by default.

   g Click the arrow to move the protocol to the Selected section.

   h Complete the SSL protocol details.

   You can also accept the default SSL profile settings.

Option	Description
Session Caching	SSL session caching allows the SSL client and server to reuse previously negotiated security parameters avoiding the expensive public key operation during an SSL handshake.
Session Cache Entry Timeout	Enter the cache timeout in seconds to specify how long the SSL session parameters must be kept and can be reused.
Prefer Server Cipher	Toggle the button so that the server can select the first supported cipher from the list it can support. During an SSL handshake, the client sends an ordered list of supported ciphers to the server.

   i Click **OK**.

4 Create a Server SSL profile.

   a Select **Add > Server Side SSL** from the drop-down menu.

   b Enter a name and a description for the Server SSL profile.

   c Select the SSL Ciphers to be included in the Server SSL profile.

   You can also create custom SSL Ciphers.

   d Click the arrow to move the ciphers to the Selected section.

   e Click the **Protocols and Sessions** tab.

f Select the SSL protocols to be included in the Server SSL profile.

SSL protocol versions TLS1.1 and TLS1.2 are enabled by default. TLS1.0 is also supported, but disabled by default.

g Click the arrow to move the protocol to the Selected section.

h Accept the default session caching setting.

SSL session caching allows the SSL client and server to reuse previously negotiated security parameters avoiding the expensive public key operation during an SSL handshake.

i Click **OK**.

Configure Layer 4 Virtual Servers

Virtual servers receive all the client connections and distribute them among the servers. A virtual server has an IP address, a port, and a protocol. For Layer 4 virtual servers, lists of ports ranges can be specified instead of a single TCP or UDP port to support complex protocols with dynamic ports.

A Layer 4 virtual server must be associated to a primary server pool, also called a default pool.

If a virtual server status is disabled, any new connection attempts to the virtual server are rejected by sending either a TCP RST for the TCP connection or ICMP error message for UDP. New connections are rejected even if there are matching persistence entries for them. Active connections continue to be processed. If a virtual server is deleted or disassociated from a load balancer, then active connections to that virtual server fail.

Prerequisites

- Verify that application profiles are available. See Configure Application Profiles.
- Verify that persistent profiles are available. See Configure Persistent Profiles.
- Verify that SSL profiles for the client and server are available. See Configure SSL Profile.
- Verify that server pools are available. See Add a Server Pool for Load Balancing.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **Advanced Networking & Security > Networking > Load Balancer > Virtual Servers > Add**.

3 Enter a name and a description for the Layer 4 virtual server.

4 Select a Layer 4 protocol from the drop-down menu.

Layer 4 virtual servers support either the Fast TCP or Fast UDP protocol, but not both. For Fast TCP or Fast UDP protocol support on the same IP address and port, for example DNS, a virtual server must be created for each protocol.

Based on the protocol type, the existing application profile is automatically populated.

5 Toggle the Access Log button to enable logging for the Layer 4 virtual server.

6 Click **Next**.

7 Enter the virtual server IP address and port number.

You can enter the virtual server port number or port range.

8 Complete the advanced properties details.

Option	Description
Maximum Concurrent Connection	Set the maximum concurrent connection allowed to a virtual server so that the virtual server does not deplete resources of other applications hosted on the same load balancer.
Maximum New Connection Rate	Set the maximum new connection to a server pool member so that a virtual server does not deplete resources.
Default Pool Member Port	Enter a default pool member port if the pool member port for a virtual server is not defined. For example, if a virtual server is defined with port range 2000-2999 and the default pool member port range is set as 8000-8999, then an incoming client connection to the virtual server port 2500 is sent to a pool member with a destination port set to 8500.

9 Select an existing server pool from the drop-down menu.

The server pool consists of one or more servers, also called pool members that are similarly configured and running the same application.

10 Select an existing sorry server pool from the drop-down menu.

The sorry server pool serves the request when a load balancer cannot select a backend server to the serve the request from the default pool.

11 Click **Next**.

12 Select the existing persistence profile from the drop-down menu.

Persistence profile can be enabled on a virtual server to allow related client connections to be sent to the same server.

13 Click **Finish**.

Configure Layer 7 Virtual Servers

Virtual servers receive all the client connections and distribute them among the servers. A virtual server has an IP address, a port, and a protocol TCP.

Load balancer rules are supported for only Layer 7 virtual servers with an HTTP application profile. Different load balancer services can use load balancer rules.

Each load balancer rule consists of single or multiple match conditions and single or multiple actions. If the match conditions are not specified, then the load balancer rule always matches and is used to define default rules. If more than one match condition is specified, then the matching strategy determines if all conditions must match or any one condition must match for the load balancer rule to be considered a match.

Each load balancer rule is implemented at a specific phase of the load balancing processing; HTTP Request Rewrite, HTTP Request Forwarding, and HTTP Response Rewrite. Not all the match conditions and actions are applicable to each phase.

If a virtual server status is disabled, any new connection attempts to the virtual server are rejected by sending either a TCP RST for the TCP connection or ICMP error message for UDP. New connections are rejected even if there are matching persistence entries for them. Active connections continue to be processed. If a virtual server is deleted or disassociated from a load balancer, then active connections to that virtual server fail.

Prerequisites

- Verify that application profiles are available. See Configure Application Profiles.
- Verify that persistent profiles are available. See Configure Persistent Profiles.
- Verify that SSL profiles for the client and server are available. See Configure SSL Profile.
- Verify that server pools are available. See Add a Server Pool for Load Balancing.
- Verify that CA and client certificate are available. See Create a Certificate Signing Request File.
- Verify that a certification revocation list (CRL) is available. See Import a Certificate Revocation List.
- Configure Layer 7 Virtual Server Pool and Rules

  With Layer 7 virtual servers, you can optionally configure load balancer rules and customize load balancing behavior using match or action rules.
- Configure Layer 7 Virtual Server Load Balancing Profiles

  With Layer 7 virtual servers, you can optionally configure load balancer persistence, client-side SSL, and server-side SSL profiles.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Load Balancer > Virtual Servers > Add**.
3. Enter a name and a description for the Layer 7 virtual server.

4 Select the Layer 7 menu item.

   Layer 7 virtual servers support the HTTP and HTTPS protocols.

   The existing HTTP application profile is automatically populated.

5 (Optional) Click **Next** to configure server pool and load balancing profiles.

6 Click **Finish**.

Configure Layer 7 Virtual Server Pool and Rules

With Layer 7 virtual servers, you can optionally configure load balancer rules and customize load balancing behavior using match or action rules.

Load Balancer rules support REGEX for match types. PCRE style REGEX patters is supported with a few limitations on advanced use cases. When REGEX is used in match conditions, named capturing groups are supported.

REGEX restrictions include:

- Character unions and intersections are not supported. For example, do not use [a-z[0-9]] and [a-z&&[aeiou]] instead use [a-z0-9] and [aeiou] respectively.
- Only 9 back references are supported and \1 through \9 can be used to refer to them.
- Use \0dd format to match octal characters, not the \ddd format.
- Embedded flags are not supported at the top level, they are only supported within groups. For example, do not use "Case (?i:s)ensitive" instead use "Case ((?i:s)ensitive)".
- Preprocessing operations \l, \u, \L, \U are not supported. Where \l - lowercase next char \u - uppercase next char \L - lower case until \E \U - upper case to \E.
- (?(condition)X), (?{code}), (??{Code}) and (?#comment) are not supported.
- Predefined Unicode character class \X is not supported
- Using named character construct for Unicode characters is not supported. For example, do not use \N{name} instead use \u2018.

When REGEX is used in match conditions, named capturing groups are supported. For example, REGEX match pattern /news/(?<year>\d+)-(?<month>\d+)-(?<day>\d+)/(?<article>.*) can be used to match a URI like /news/2018-06-15/news1234.html.

Then variables are set as follows, $year = "2018" $month = "06" $day = "15" $article = "news1234.html". After the variables are set, these variables can be used in load balancer rule actions. For example, URI can be rewritten using the matched variables like, /news.py?year=$year&month=$month&day=$day&article=$article. Then the URI gets rewritten as /news.py?year=2018&month=06&day=15&article=news1234.html.

Rewrite actions can use a combination of named capturing groups and built-in variables. For example, URI can be written as /news.py?year=$year&month=$month&day=$day&article=$article&user_ip=$_remote_addr. Then the example URI gets rewritten as /news.py?year=2018&month=06&day=15&article=news1234.html&user_ip=1.1.1.1.

Note For named capturing groups, the name cannot start with an _ character.

In addition to named capturing groups, the following built-in variables can be used in rewrite actions. All the built-in variable names start with _.

- $_args - arguments from the request
- $_cookie_<name> - value of <name> cookie
- $_host - in the order of precedence - host name from the request line, or host name from the "Host" request header field, or the server name matching a request
- $_hostname - host name
- $_http_<name> - arbitrary request header field and <name> is the field name converted to lower case with dashes replaced by underscores
- $_https - "on" if connection operates in SSL mode, or "" otherwise
- $_is_args - "?" if a request line has arguments, or "" otherwise
- $_query_string - same as $_args
- $_remote_addr - client address
- $_remote_port - client port
- $_request_uri - full original request URI (with arguments)
- $_scheme - request scheme, "http" or "https"
- $_server_addr - address of the server which accepted a request
- $_server_name - name of the server which accepted a request
- $_server_port - port of the server which accepted a request
- $_server_protocol - request protocol, usually "HTTP/1.0" or "HTTP/1.1"
- $_ssl_client_cert - returns the client certificate in the PEM format for an established SSL connection, with each line except the first prepended with the tab character
- $_ssl_server_name - returns the server name requested through SNI
- $_uri - URI path in request
- $_ssl_ciphers: returns the client SSL ciphers
- $_ssl_client_i_dn: returns the "issuer DN" string of the client certificate for an established SSL connection according to RFC 2253
- $_ssl_client_s_dn: returns the "subject DN" string of the client certificate for an established SSL connection according to RFC 2253

- $_ssl_protocol: returns the protocol of an established SSL connection
- $_ssl_session_reused: returns "r" if an SSL session was reused, or "." otherwise

Prerequisites

Verify a Layer 7 virtual server is available. See Configure Layer 7 Virtual Servers.

Procedure

1 Open the Layer 7 virtual server.

2 Skip to the Virtual Server Identifiers page.

3 Enter the virtual server IP address and port number.

You can enter the virtual server port number or port range.

4 Complete the advanced properties details.

Option	Description
Maximum Concurrent Connection	Set the maximum concurrent connection allowed to a virtual server so that the virtual server does not deplete resources of other applications hosted on the same load balancer.
Maximum New Connection Rate	Set the maximum new connection to a server pool member so that a virtual server does not deplete resources.
Default Pool Member Port	Enter a default pool member port if the pool member port for a virtual server is not defined. For example, if a virtual server is defined with port range 2000–2999 and the default pool member port range is set as 8000-8999, then an incoming client connection to the virtual server port 2500 is sent to a pool member with a destination port set to 8500.

5 (Optional) Select an existing default server pool from the drop-down menu.

The server pool consists of one or more servers, called pool members that are similarly configured and running the same application.

6 Click **Add** to configure the load balancer rules for the HTTP Request Rewrite phase.

Supported match types are, REGEX, STARTS_WITH, ENDS_WITH, etc and inverse option.

Supported Match Condition	Description
HTTP Request Method	Match an HTTP request method. http_request.method - value to match
HTTP Request URI	Match an HTTP request URI without query arguments. http_request.uri - value to match
HTTP Request URI arguments	Match an HTTP request URI query argument. http_request.uri_arguments - value to match
HTTP Request Version	Match an HTTP request version. http_request.version - value to match

Supported Match Condition	Description
HTTP Request Header	Match any HTTP request header. http_request.header_name - header name to match http_request.header_value - value to match
HTTP Request Payload	Match an HTTP request body content. http_request.body_value - value to match
TCP Header Fields	Match a TCP source or the destination port. tcp_header.source_port - source port to match tcp_header.destination_port - destination port to match
IP Header Fields	Match an IP source or destination address. ip_header.source_address - source address to match ip_header.destination_address - destination address to match

Action	Description
HTTP Request URI Rewrite	Modify an URI. http_request.uri - URI (without query arguments) to write http_request.uri_args - URI query arguments to write
HTTP Request Header Rewrite	Modify value of an HTTP header. http_request.header_name - header name http_request.header_value - value to write

7 Click **Add** to configure the load balancer rules for the HTTP Request Forwarding.

All match values accept regular expressions.

Supported Match Condition	Description
HTTP Request Method	Match an HTTP request method. http_request.method - value to match
HTTP Request URI	Match an HTTP request URI. http_request.uri - value to match
HTTP Request URI args	Match an HTTP request URI query argument. http_request.uri_args - value to match
HTTP Request Version	Match an HTTP request version. http_request.version - value to match
HTTP Request Header	Match any HTTP request header. http_request.header_name - header name to match http_request.header_value - value to match
HTTP Request Payload	Match an HTTP request body content. http_request.body_value - value to match

Supported Match Condition	Description
TCP Header Fields	Match a TCP source or the destination port. tcp_header.source_port - source port to match tcp_header.destination_port - destination port to match
IP Header Fields	Match an IP source address. ip_header.source_address - source address to match

Action	Description
Reject	Reject a request, for example, by setting status to 5xx. http_forward.reply_status - HTTP status code used to reject http_forward.reply_message - HTTP rejection message
Redirect	Redirect a request. Status code must be set to 3xx. http_forward.redirect_status - HTTP status code for redirect http_forward.redirect_url - HTTP redirect URL
Select Pool	Force the request to a specific server pool. Specified pool member's configured algorithm (predictor) is used to select a server within the server pool. http_forward.select_pool - server pool UUID

8 Click **Add** to configure the load balancer rules for the HTTP Response Rewrite.

All match values accept regular expressions.

Supported Match Condition	Description
HTTP Response Header	Match any HTTP response header. http_response.header_name - header name to match http_response.header_value - value to match

Action	Description
HTTP Response Header Rewrite	Modify the value of an HTTP response header. http_response.header_name - header name http_response.header_value - value to write

9 (Optional) Click **Next** to configure load balancing profiles.

10 Click **Finish**.

Configure Layer 7 Virtual Server Load Balancing Profiles

With Layer 7 virtual servers, you can optionally configure load balancer persistence, client-side SSL, and server-side SSL profiles.

Note SSL profile is not supported in the NSX-T Data Center limited export release.

If a client-side SSL profile binding is configured on a virtual server but not a server-side SSL profile binding, then the virtual server operates in an SSL-terminate mode, which has an encrypted connection to the client and plain text connection to the server. If both the client-side and server-side SSL profile bindings are configured, then the virtual server operates in SSL-proxy mode, which has an encrypted connection both to the client and the server.

Associating server-side SSL profile binding without associating a client-side SSL profile binding is currently not supported. If a client-side and a server-side SSL profile binding is not associated with a virtual server and the application is SSL-based, then the virtual server operates in an SSL-unaware mode. In this case, the virtual server must be configured for Layer 4. For example, the virtual server can be associated to a fast TCP profile.

Prerequisites

Verify a Layer 7 virtual server is available. See Configure Layer 7 Virtual Servers.

Procedure

1 Open the Layer 7 virtual server.

2 Skip to the Load Balancing Profiles page.

3 Toggle the Persistence button to enable the profile.

   Persistence profile allows related client connections to be sent to the same server.

4 Select either the Source IP Persistence or Cookie Persistence profile.

5 Select the existing persistence profile from the drop-down menu.

6 Click **Next**.

7 Toggle the Client Side SSL button to enable the profile.

   Client-side SSL profile binding allows multiple certificates, for different host names to be associated to the same virtual server.

   The associated Client-side SSL profile is automatically populated.

8 Select a default certificate from the drop-down menu.

   This certificate is used if the server does not host multiple host names on the same IP address or if the client does not support Server Name Indication (SNI) extension.

9 Select the available SNI certificate and click the arrow to move the certificate to the Selected section.

10 (Optional) Toggle the Mandatory Client Authentication to enable this menu item.

11 Select the available CA certificate and click the arrow to move the certificate to the Selected section.

12 Set the certificate chain depth to verify the depth in the server certificates chain.

13 Select the available CRL and click the arrow to move the certificate to the Selected section.

   A CRL can be configured to disallow compromised server certificates.

14 Click **Next**.

15 Toggle the Server Side SSL button to enable the profile.

   The associated Server-side SSL profile is automatically populated.

16 Select a client certificate from the drop-down menu.

   The client certificate is used if the server does not host multiple host names on the same IP address or if the client does not support Server Name Indication (SNI) extension.

17 Select the available SNI certificate and click the arrow to move the certificate to the Selected section.

18 (Optional) Toggle the Server Authentication to enable this menu item.

   Server-side SSL profile binding specifies whether the server certificate presented to the load balancer during the SSL handshake must be validated or not. When validation is enabled, the server certificate must be signed by one of the trusted CAs whose self-signed certificates are specified in the same server-side SSL profile binding.

19 Select the available CA certificate and click the arrow to move the certificate to the Selected section.

20 Set the certificate chain depth to verify the depth in the server certificates chain.

21 Select the available CRL and click the arrow to move the certificate to the Selected section.

   A CRL can be configured to disallow compromised server certificates. OCSP and OCSP stapling are not supported on the server-side.

22 Click **Finish**.

Advanced Firewall

20

Note If you use the **Advanced Networking & Security** user interface to modify objects created in the policy interface, some settings might not be configurable. These read-only settings have this icon next to them: ⊖. See Chapter 1 Overview of the NSX Manager for more information.

This chapter includes the following topics:

- Firewall Sections and Firewall Rules

Firewall Sections and Firewall Rules

Firewall sections are used to group a set of firewall rules.

A firewall section is made up from one or more individual firewall rules. Each individual firewall rule contains instructions that determine whether a packet should be allowed or blocked; which protocols it is allowed to use; which ports it is allowed to use and so forth. Sections are used for multi-tenancy , such as specific rules for sales and engineering departments in separate sections.

A section can be defined as enforcing stateful or stateless rules. Stateless rules are treated as traditional stateless ACLs. Reflexive ACLs are not supported for stateless sections. A mix of stateless and stateful rules on a single logical switch port is not recommended and may cause undefined behavior.

Rules can be moved up and down within a section. For any traffic attempting to pass through the firewall, the packet information is subjected to the rules in the order shown in the section, beginning at the top and proceeding to the default rule at the bottom. The first rule that matches the packet has its configured action applied, and any processing specified in the rule's configured options is performed and all subsequent rules are ignored (even if a later rule is a better match). Thus, you should place specific rules above more general rules to ensure those rules are not ignored. The default rule, located at the bottom of the rule table, is a "catchall" rule; packets not matching any other rules will be enforced by the default rule.

Note A logical switch has a property called N-VDS mode. This property comes from the transport zone that the switch belongs to. If the N-VDS mode is `ENS` (also known as `Enhanced Datapath`), then you cannot create a firewall rule or section with the switch or its ports in the `Source`, `Destination`, or `Applied To` fields.

Add a Firewall Rule Section

A firewall rule section is edited and saved independently and is used to apply separate firewall configuration to tenants.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **General** tab for layer 3 (L3) rules or the **Ethernet** tab for layer 2 (L2) rules.
3. Click an existing section or rule.
4. Click the section icon on the menu bar and select **Add Section Above** or **Add Section Below**.

   Note For any traffic attempting to pass through the firewall, the packet information is subjected to the rules in the order shown in the Rules table, beginning at the top and proceeding to the default rules at the bottom. In some cases, the order of precedence of two or more rules might be important in determining the disposition of a packet.

5. Enter the section name.
6. To make the firewall stateless, select the **Enable Stateless Firewall**. This option is applicable for L3 only.

   Stateless firewalls watch network traffic, and restrict or block packets based on source and destination addresses or other static values. Stateful firewalls can watch traffic streams from end to end. Stateless firewalls are typically faster and perform better under heavier traffic loads. Stateful firewalls are better at identifying unauthorized and forged communications. There is no toggling between stateful and stateless once it is defined.

7. Select one or more objects to apply the section.

   The types of object are logical ports, logical switches, and NSGroups. If you select an NSGroup, it must contain one or more logical switches or logical ports. If the NSGroup contains only IP sets or MAC sets, it will be ignored.

   Note The **Applied To** in a section it will override any **Applied To** settings in the rules in that section.

8. Click **OK**.

What to do next

Add Firewall rules to the section.

Delete a Firewall Rule Section

A firewall rule section can be deleted when it is no longer used.

When you delete a firewall rule section, all rules in that section are deleted. You cannot delete a section and add it again at a different place in the firewall table. To do so, you must delete the section and publish the configuration. Then add the deleted section to the firewall table and re-publish the configuration.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **General** tab for L3 rules or the **Ethernet** tab for L2 rules.
3. Click the menu icon in the first column of the section and select **Delete Section**.

   You can also select the section and click the delete icon on the menu bar.

Enable and Disable Section Rules

You can enable or disable all rules in a firewall rule section.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **General** tab for L3 rules or the **Ethernet** tab for L2 rules.
3. Click the menu icon in the first column of the section and select **Enable All Rules** or **Disable All Rules**.
4. Click **Publish**.

Enable and Disable Section Logs

Enabling logs for section rules records information on packets for all of the rules in a section. Depending on the number of rules in a section, a typical firewall section will generate large amounts of log information and can affect performance.

Logs are stored in the /var/log/dfwpktlogs.log file on ESXi and KVM hosts.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **General** tab for L3 rules or the **Ethernet** tab for L2 rules.
3. Click the menu icon in the first column of the section and select **Enable Logs** or **Disable Logs**.
4. Click **Publish**.

About Firewall Rules

NSX-T Data Center uses firewall rules to specify traffic handling in and out of the network.

Firewall offers multiple sets of configurable rules: Layer 3 rules (General tab) and Layer 2 rules (Ethernet tab). Layer 2 firewall rules are processed before Layer 3 rules. You can configure an exclusion list that contains logical switches, logical ports, or groups that are to be excluded from firewall enforcement.

Firewall Rules are enforced as follows:

- Rules are processed in top-to-bottom ordering.
- Each packet is checked against the top rule in the rule table before moving down the subsequent rules in the table.
- The first rule in the table that matches the traffic parameters is enforced.

No subsequent rules can be enforced as the search is then terminated for that packet. Because of this behavior, it is always recommended to put the most granular policies at the top of the rule table. This will ensure they will be enforced before more specific rules.

The default rule, located at the bottom of the rule table, is a catchall rule; packets not matching any other rules will be enforced by the default rule. After the host preparation operation, the default rule is set to allow action. This ensures that VM-to-VM communication is not broken during staging or migration phases. It is a best practice to then change this default rule to block action and enforce access control through a positive control model (i.e., only traffic defined in the firewall rule is allowed onto the network).

Note TCP strict can be enabled on a per section basis to turn off mid-session pick-up and enforce the requirement for a three-way handshake. When enabling TCP strict mode for a particular Distributed Firewall Section, and using a default ANY-ANY Block rule, packets that do not complete the three-way handshake connection requirements, and that match a TCP-based rule in this section are dropped. Strict is only applied to stateful TCP rules and is enabled at the distributed firewall section level. TCP strict is not enforced for packets that match a default ANY-ANY Allow which as no TCP service specified.

Table 20-1. Properties of a Firewall Rule

Property	Description
Name	Name of the firewall rule.
ID	Unique system generated ID for each rule.
Source	The source of the rule can be either an IP or MAC address or an object other than an IP address. The source will match any if not defined. Both IPv4 and IPv6 are supported for source or destination range.
Destination	The destination IP or MAC address/netmask of the connection that is affected by the rule. The destination will match any if not defined. Both IPv4 and IPv6 are supported for source or destination range.
Service	The service can be a predefined port protocol combination for L3. For L2 it can be ether-type. For both L2 and L3 you can manually define a new service or service group. The service will match any, if it is not specified.
Applied To	Defines the scope at which this rule is applicable. If not defined the scope will be all logical ports. If you have added "applied to" in a section it will overwrite the rule.

Table 20-1. Properties of a Firewall Rule (continued)

Property	Description
Log	Logging can be turned off or on. Logs are stored at /var/log/dfwpktlogs.log file on ESX and KVM hosts.
Action	The action applied by the rule can be `Allow`, `Drop`, or `Reject`. The default is `Allow`.
IP Protocol	The options are `IPv4`, `IPv6`, and `IPv4_IPv6`. The default is `IPv4_IPv6`. To access this property, click the **Advanced Settings** icon.
Direction	The options are `In`, `Out`, and `In/Out`. The default is `In/Out`. This field refers to the direction of traffic from the point of view of the destination object. `In` means that only traffic to the object is checked, `Out` means that only traffic from the object is checked, and `In/Out` means traffic in both directions is checked. To access this property, click the **Advanced Settings** icon.
Rule Tags	Tags that have been added to the rule. To access this property, click the **Advanced Settings** icon.
Flow Statistics	Read-only field that displays the byte, packet count, and sessions. To access this property, click the graph icon.

Note If SpoofGuard is not enabled, automatically discovered address bindings cannot be guaranteed to be trustworthy because a malicious virtual machine can claim the address of another virtual machine. SpoofGuard, if enabled, verifies each discovered binding so that only approved bindings are presented.

Add a Firewall Rule

A firewall is a network security system that monitors and controls the incoming and outgoing network traffic based on predetermined firewall rules.

Firewall rules are added at the NSX Manager scope. Using the Applied To field, you can then narrow down the scope at which you want to apply the rule. You can add multiple objects at the source and destination levels for each rule, which helps reduce the total number of firewall rules to be added.

Note By default, a rule matches on the default of any source, destination, and service rule elements, matching all interfaces and traffic directions. If you want to restrict the effect of the rule to particular interfaces or traffic directions, you must specify the restriction in the rule.

Prerequisites

To use a group of addresses, first manually associate the IP and MAC address of each VM with their logical switch.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **General** tab for L3 rules or the **Ethernet** tab for L2 rules.
3. Click an existing section or rule.

4 Click the menu icon in the first column of a rule and select **Add Rule Above** or **Add Rule Below**.

A new row appears to define a firewall rule.

Note For any traffic attempting to pass through the firewall, the packet information is subjected to the rules in the order shown in the Rules table, beginning at the top and proceeding to the default rules at the bottom. In some cases, the order of precedence of two or more rules might be important in determining the disposition of a packet.

5 In the **Name** column, enter the rule name.

6 In the **Source** column, click the edit icon and select the source of the rule. The source will match any if not defined.

Option	Description
IP Addresses	Enter multiple IP or MAC addresses in a comma-separated list. The list can contain up to 255 characters. Both IPv4 and IPv6 formats are supported.
Container Objects	The available objects are IP Set, Logical Port, Logical Switch, and NS Group. Select the objects and click **OK**.

7 In the **Destination** column, click the edit icon and select the destination. The destination will match any if not defined.

Option	Description
IP Addresses	You can enter multiple IP or MAC addresses in a comma-separated list. The list can contain up to 255 characters. Both IPv4 and IPv6 formats are supported.
Container Objects	The available objects are IP Set, Logical Port, Logical Switch, and NS Group. Select the objects and click **OK**.

8 In the **Service** column, click the edit icon and select services. The service will match any if not defined.

9 To select a predefined service, select one of more available services.

10 To define a new service, click the **Raw Port-Protocol** tab and click **Add**..

Option	Description
Type of Service	■ ALG ■ ICMP ■ IGMP ■ IP ■ L4 Port Set
Protocol	Select one of the available protocols.
Source Ports	Enter the source port.
Destination Ports	Select the destination port.

11 In the **Applied To** column, click the edit icon and select objects.

12 In the **Log** column, set the logging option.

Logs are in the `/var/log/dfwpktlogs.log` file on ESXi and KVM hosts. Enabling logging can affect performance.

13 In the **Action** column, select an action.

Option	Description
Allow	Allows all L3 or L2 traffic with the specified source, destination, and protocol to pass through the current firewall context. Packets that match the rule, and are accepted, traverse the system as if the firewall is not present
Drop	Drops packets with the specified source, destination, and protocol. Dropping a packet is a silent action with no notification to the source or destination systems. Dropping the packet causes the connection to be retried until the retry threshold is reached.
Reject	Rejects packets with the specified source, destination, and protocol. Rejecting a packet is a more graceful way to deny a packet, as it sends a destination unreachable message to the sender. If the protocol is TCP, a TCP RST message is sent. ICMP messages with administratively prohibited code are sent for UDP, ICMP, and other IP connections. One benefit of using Reject is that the sending application is notified after only one attempt that the connection cannot be established.

14 Click the **Advanced Settings** icon to specify IP protocol, direction, rule tags, and comments.

15 Click **Publish**.

Delete a Firewall Rule

A firewall is a network security system that monitors and controls the incoming and outgoing network traffic based on predetermined firewall rules. Custom defined rules can be added and deleted.

Procedure

1 Select **Advanced Networking & Security > Security > Distributed Firewall**.

2 Click the **General** tab for L3 rules or the **Ethernet** tab for L2 rules.

3 Click the menu icon in the first column of the rule and select **Delete Rule**.

4 Click **Publish**.

Edit the Default Distributed Firewall Rule

You can edit the default firewall settings that apply to traffic that does not match any of the user-defined firewall rules.

The default firewall rules apply to traffic that does not match any of the user-defined firewall rules. The default Layer 3 rule is under the **General** tab and the default Layer 2 rule is under the **Ethernet** tab.

The default firewall rules allow all L3 and L2 traffic to pass through all prepared clusters in your infrastructure. The default rule is always at the bottom of the rules table and cannot be deleted. However, you can change the **Action** element of the rule from **Allow** to **Drop** or **Reject** (not recommended), and indicate whether traffic for that rule should be logged.

The default Layer 3 firewall rule applies to all traffic, including DHCP. If you change the **Action** to **Drop** or **Reject**, DHCP traffic will be blocked. You will need to create a rule to allow DHCP traffic.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **General** tab for L3 rules or the **Ethernet** tab for L2 rules.
3. In the **Name** column, enter a new name.
4. In the **Action** column, select one of the options.
   - Allow - Allows all L3 or L2 traffic with the specified source, destination, and protocol to pass through the current firewall context. Packets that match the rule, and are accepted, traverse the system as if the firewall is not present.
   - Drop - Drops packets with the specified source, destination, and protocol. Dropping a packet is a silent action with no notification to the source or destination systems. Dropping the packet causes the connection to be retried until the retry threshold is reached.
   - Reject - Rejects packets with the specified source, destination, and protocol. Rejecting a packet is a more graceful way to deny a packet, as it sends a destination unreachable message to the sender. If the protocol is TCP, a TCP RST message is sent. ICMP messages with administratively prohibited code are sent for UDP, ICMP, and other IP connections. One benefit of using Reject is that the sending application is notified after only one attempt that the connection cannot be established.

   Note Selecting **Reject** as the action for the default rule is not recommended.
5. In the **Log**, enable or disable logging.

   Enabling logging can affect performance.
6. Click **Publish**.

Change the Order of a Firewall Rule

Rules are processed in top-to-bottom ordering. You can change the order of the rules in the list.

For any traffic attempting to pass through the firewall, the packet information is subjected to the rules in the order shown in the rules table, beginning at the top and proceeding to the default rules at the bottom. In some cases, the order of precedence of two or more rules might be important in determining the traffic flow.

You can move a custom rule up or down in the table. The default rule is always at the bottom of the table and cannot be moved.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **General** tab for L3 rules or the **Ethernet** tab for L2 rules.
3. Select the rule and click the **Move Up** or **Move Down** icon on the menu bar.
4. Click **Publish**.

Filter Firewall Rules

When you navigate to the firewall section, initially all the rules are displayed. You can apply a filter to control what is displayed so that you see only a subset of the rules. This can make it easier to manage the rules.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **General** tab for L3 rules or the **Ethernet** tab for L2 rules.
3. In the search text field on the right side of the menu bar, select an object or enter the beginning characters of an object's name to narrow down the list of objects to select.

   After you select an object, the filter is applied and the list of rules is updated, showing only rules that contain the object in any of the following columns:

   - Sources
   - Destinations
   - Applied To
   - Services
4. To remove the filter, delete the object name from the text field.

Configure Firewall for a Logical Switch Bridge Port

You can configure firewall sections and firewall rules for the bridge port of a layer 2 bridge-backed logical switch. The bridge must be created using NSX Edge nodes.

Prerequisites

Verify that the switch is attached to a bridge profile. See Create a Layer 2 Bridge-Backed Logical Switch.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Security > Bridge Firewall**.

3. Select a logical switch.

   The switch must be attached to a bridge profile.

4. Follow the same steps in previous sections for configuring layer 2 or layer 3 firewall.

Configure a Firewall Exclusion List

A logical port, logical switch, or NSGroup can be excluded from a firewall rule.

After you've created a section with firewall rules you may want to exclude an NSX-T Data Center appliance port from the firewall rules.

Procedure

1. Select **Advanced Networking & Security > Security > Distributed Firewall > Exclusion List > Add**.
2. Select a type and an object.

   The available types are **Logical Port**, **Logical Switch**, and **NSGroup**.

3. Click **OK**.
4. To remove an object from the exclusion list, select the object and click **Delete** on the menu bar.

Enable and Disable Distributed Firewall

You can enable or disable the distributed firewall feature.

If it is disabled, no firewall rules are enforced at the dataplane level. Upon re-enablement rules are re-enforced.

Procedure

1. Navigate to **Advanced Networking & Security > Security > Distributed Firewall**.
2. Click the **Settings** tab.
3. Click Distributed Firewall **Edit**.
4. In the dialog box, toggle the firewall status to green (enabled) or gray (disabled).
5. Click **Save**.

Add or Delete a Firewall Rule to a Logical Router

You can add firewall rules to a tier-0 or tier-1 logical router to control communication into the router.

Edge fire-walling is implemented on uplink router ports, meaning that firewall rules will be applicable only if traffic hits uplink router ports on edge. To apply firewall rules to particular IP destination, you must configure groups with /32 network. If you provide a subnet other than /32, firewall rules will be applied to the complete subnet.

Prerequisites

Familiarize yourself with the parameters of a firewall rule. See Add a Firewall Rule.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Advanced Networking & Security > Networking > Routers**.
3. Click the **Routers** tab if it is not already selected.
4. Click the name of a logical router.
5. Select **Services > Edge Firewall**.
6. Click an existing section or rule.
7. To add a rule, click **Add Rule** on the menu bar and select **Add Rule Above** or **Add Rule Below**, or click the menu icon in the first column of a rule and select **Add Rule Above** or **Add Rule Below**, and specify the rule parameters.

   The Applied To field is not shown because this rule applies only to the logical router.
8. To delete a rule, select the rule, click **Delete** on the menu bar or click the menu icon in the first column and select **Delete**.

Results

Note If you add a firewall rule to a tier-0 logical router and the NSX Edge cluster backing the router is running in active-active mode, the firewall can only run in stateless mode. If you configure the firewall rule with stateful services such as HTTP, SSL, TCP, and so on, the firewall rule will not work as expected. To avoid this issue, configure the NSX Edge cluster to run in active-standby mode.

CPU and Memory Utilization Threshold Using API

Apply CPU and memory utilization thresholds to distributed firewall rules by using service configuration APIs. When you implement the service configuration API, you can apply a profile configuration to an entity such as VM groups, transport nodes, logical switches, and logical ports.

Get Service Configuration Details

Refer to the *NSX-T Data Center API* guide for syntax and usage details.

List of all service configuration.

```
GET https://<nsx-mgr>/api/v1/service-configs
```

Table 20-2. API Attributes

Attribute	Details
`Profile`	Profiles are configurations that are applied to a VM group. For example, `FirewallSessionTimerProfile` is the profile that is applied to a transport node to gather details on the CPU utilization rate of the transport node when distributed firewall rules are run. **Note** Only one profile can be included in a service configuration.
`Applied_To`	VM group on which the service profile is applied.
`Precedence`	Precedence is applied per profile type. NSX-T Data Center decides the priority of profiles that must be applied to a VM group by ascending precedence numbers. For example, a profile with sequence number 1 has higher priority than sequence number 2.

Create a Service Configuration

Creates a service configuration that can group profiles and configuration.

```
POST https://<nsx-mgr>/api/v1/service-config
{
  "display_name":"testServiceConfig",
  "profiles":[{"profile_type":"FirewallSessionTimerProfile",
              "target_id":"183e372b-854c-4fcc-a24e-05721ce89a60"
              }
            ],
  “precedence”: 10,
  “applied_to”: [{
    "target_id":"333e372b-854c-4fcc-a24e-05721ce89b71",
    "target_type" : "NSGroup"
              }]
}
```

```
Example Response:
{
  "id": "183e372b-854c-4fcc-a24e-05721ce89a60",
  "display_name":"testServiceConfig",
  "profiles":[{"profile_type":"FirewallSessionTimerProfile",
              "target_id":"183e372b-854c-4fcc-a24e-05721ce89a60"
              }
            ],
  “precedence”: 10,
  “applied_to”: [{
                  "target_id":"333e372b-854c-4fcc-a24e-05721ce89b71",
    "target_type" : "NSGroup"
              }]
  "_create_user": "system",
  "_last_modified_user": "system",
  "_last_modified_time": 1414057732203,
  "_create_time": 1414057732203
}
```

Delete a Service Configuration

Deletes the specified service config.

```
DELETE https://<nsx-mgr>/api/v1/service-configs/<183e372b-854c-4fcc-a24e-05721ce89a60>
```

Get Details of a Specific Configuration

Returns information about the specified Service Config.

```
GET https://<nsx-mgr>/api/v1/service-configs/<183e372b-854c-4fcc-a24e-05721ce89a60>
```

```
Example Response:
{
  "_revision": 1,
  "id": "183e372b-854c-4fcc-a24e-05721ce89a60",
    "display_name":"testServiceConfig1",
    "resource_type": "ServiceConfig",
    "profiles":[{"profile_type":"FirewallSessionTimerProfile",
                "target_id":"183e372b-854c-4fcc-a24e-05721ce89a45",
                "is_valid":true
              }],
    “precedence”: 10,
    “applied_to”: [{"target_id":"333e372b-854c-4fcc-a24e-05721ce89b71",
              "target_type": "LogicalSwitch",
              "is_valid":true
                }
              ]
    "_create_user": "system",
    "_last_modified_user": "system",
    "_last_modified_time": 1414057732203,
    "_create_time": 1414057732203
}
```

Update a Service Configuration

Updates the specified ServiceConfig.

```
PUT https://<nsx-mgr>/api/v1/service-configs/183e372b-854c-4fcc-a24e-05721ce89a60
{
  "id": "183e372b-854c-4fcc-a24e-05721ce89a60",
  "display_name":"testServiceConfig1",
  "resource_type": "ServiceConfig",
  "profiles":[{"profile_type":"FirewallSessionTimerProfile",
              "target_id":"183e372b-854c-4fcc-a24e-05721ce89a45"
            }],
  “precedence”: 10,
  “applied_to”: [{"target_id":"333e372b-854c-4fcc-a24e-05721ce89b71",
    "target_type" : "NSGroup"
              }]
  "_create_user": "system",
  "_last_modified_user": "system",
  "_last_modified_time": 1414057732203,
  "_create_time": 1414057732203,
  "_create_user": "admin",
  "_revision": 0
}
```

Get Effective Profiles

Returns the effective profiles applied to the specified resource.

```
GET https://<nsx-mgr>/api/v1/service-configs/effective-profiles?
resource_id=<144e372b-854c-4fcc-a24e-05721ce89a60>&resource_type=NSGroup
```

```
Example Response:
{
  "cursor": "00012",
  "sort_ascending": true,
  "result_count": 2,
  "results": [
           { "profile_type":"FirewallSessionTimerProfile",
             "target_id":"183e372b-854c-4fcc-a24e-05721ce89a45",
             "target_name":"Firewall Session Timer Profile
             "is_valid":true
            },
            { "profile_type":"FirewallCpuMemThresholdsProfile",
              "target_id":"5678372b-854c-4fcc-a24e-05721ce89a45",
              "target_name":"Firewall CPU Profile
              "is_valid":true
            },
          ]
  }
```

Operations and Management

21

You may need to change the configuration of the appliances you've installed, for example, adding licenses, certificates, and changing passwords. There are also routine maintenance tasks that you should perform, including running backups. Additionally, there are tools to help you find information about the appliances that are part of the NSX-T Data Center infrastructure and the logical networks created by NSX-T Data Center, including remote system logging, traceflow, and port connections.

This chapter includes the following topics:

- Find the SSH Fingerprint of a Remote Server
- View Data about Applications Running on VMs

Checking the Realized State of a Configuration Change

When you make a configuration change, NSX Manager typically sends a request to another component to implement the change. For some layer 3 entities, if you make the configuration change using the API, you can track the status of the request to see if the change is successfully implemented.

The configuration change that you initiate is called the desired state. The result of implementing the change is called the realized state. If NSX Manager implements the change successfully, the realized state will be the same as the desired state. If there is an error, the realized state will not be the same as the desired state.

For some layer 3 entities, when you call an API to make a configuration change, the response will include the parameter `request_id`. You can use the parameters `request_id` and the `entity_id` to make an API call to find out the status of the request.

This feature supports the following entities and APIs:

```
EdgeCluster
    POST /edge-clusters
    PUT /edge-clusters/<edge-cluster-id>
    DELETE /edge-clusters/<edge-cluster-id>
    POST /edge-clusters/<edge-cluster-id>?action=replace_transport_node

LogicalRouter
    POST /logical-routers
    PUT /logical-routers/<logical-router-id>
    DELETE /logical-routers/<logical-router-id>
    POST /logical-routers/<logical-router-id>?action=reprocess
    POST /logical-routers/<logical-router-id>?action=reallocate

LogicalRouterPort
    POST /logical-router-ports
    PUT /logical-router-ports/<logical-router-port-id>
    DELETE /logical-router-ports/<logical-router-port-id>

StaticRoute
    POST /logical-routers/<logical-router-id>/routing/static-routes
    PUT /logical-routers/<logical-router-id>/routing/static-routes/<static-route-id>
    DELETE /logical-routers/<logical-router-id>/routing/static-routes/<static-route-id>

BGPConfig
    PUT /logical-routers/<logical-router-id>/routing/bgp

BgpNeighbor
    POST /logical-routers/<logical-router-id>/routing/bgp/neighbors
    PUT /logical-routers/<logical-router-id>/routing/bgp/neighbors/<bgp-neighbor-id>
    DELETE /logical-routers/<logical-router-id>/routing/bgp/neighbors/<bgp-neighbor-id>
    POST /logical-routers/<logical-router-id>/routing/bgp/neighbors/<bgp-neighbor-id>
```

```
BGPCommunityList
    POST /logical-routers/<logical-router-id>/routing/bgp/community-lists
    PUT /logical-routers/<logical-router-id>/routing/bgp/community-lists/<community-list-id>
    DELETE /logical-routers/<logical-router-id>/routing/bgp/community-lists/<community-list-id>

AdvertisementConfig
    PUT /logical-routers/<logical-router-id>/routing/advertisement

AdvertiseRouteList
    PUT /logical-routers/<logical-router-id>/routing/advertisement/rules

NatRule
    POST /logical-routers/<logical-router-id>/nat/rules
    PUT /logical-routers/<logical-router-id>/nat/rules/<rule-id>
    DELETE /logical-routers/<logical-router-id>/nat/rules/<rule-id>

DhcpRelayService
    POST /dhcp/relays
    PUT /dhcp/relays/<relay-id>
    DELETE /dhcp/relays/<relay-id>

DhcpRelayProfile
    POST /dhcp/relay-profiles
    PUT /dhcp/relay-profiles/<relay-profile-id>
    DELETE /dhcp/relay-profiles/<relay-profile-id>

StaticHopBfdPeer
    POST /logical-routers/<logical-router-id>/routing/static-routes/bfd-peers
    PUT /logical-routers/<logical-router-id>/routing/static-routes/bfd-peers/<bfd-peers-id>
    DELETE /logical-routers/<logical-router-id>/routing/static-routes/bfd-peers/<bfd-peers-id>

IPPrefixList
    POST /logical-routers/<logical-router-id>/routing/ip-prefix-lists
    PUT /logical-routers/<logical-router-id>/routing/ip-prefix-lists/<ip-prefix-list-id>
    DELETE /logical-routers/<logical-router-id>/routing/ip-prefix-lists/<ip-prefix-list-id>

RouteMap
    POST /logical-routers/<logical-router-id>/routing/route-maps
    PUT /logical-routers/<logical-router-id>/routing/route-maps/<route-map-id>
    DELETE /logical-routers/<logical-router-id>/routing/route-maps/<route-map-id>

RedistributionConfig
    PUT /logical-routers/<logical-router-id>/routing/redistribution
RedistributionRuleList
    PUT /logical-routers/<logical-router-id>/routing/redistribution/rules

BfdConfig
    PUT /logical-routers/<logical-router-id>/routing/bfd-config

MplsConfig
    PUT /logical-routers/<logical-router-id>/routing/mpls

RoutingGlobalConfig
    PUT /logical-routers/<logical-router-id>/routing
```

```
IPSecVPNIKEProfile
    POST /vpn/ipsec/ike-profiles
    PUT /vpn/ipsec/ike-profiles/<ike-profile-id>
    DELETE /vpn/ipsec/ike-profiles/<ike-profile-id>

IPSecVPNDPDProfile
    POST /vpn/ipsec/dpd-profiles
    PUT /vpn/ipsec/dpd-profiles/<dpd-profile-id>
    DELETE /vpn/ipsec/dpd-profiles/<dpd-profile-id>

IPSecVPNTunnelProfile
    POST /vpn/ipsec/tunnel-profiles
    PUT /vpn/ipsec/tunnel-profiles/<tunnel-profile-id>
    DELETE /vpn/ipsec/tunnel-profiles/<tunnel-profile-id>

IPSecVPNLocalEndpoint
    POST /vpn/ipsec/local-endpoints
    PUT /vpn/ipsec/local-endpoints/<local-endpoint-id>
    DELETE /vpn/ipsec/local-endpoints/<local-endpoint-id>

IPSecVPNPeerEndpoint
    POST /vpn/ipsec/peer-endpoints
    PUT /vpn/ipsec/peer-endpoints/<peer-endpoint-id>
    DELETE /vpn/ipsec/peer-endpoints/<peer-endpoint-id>

IPSecVPNService
    POST /vpn/ipsec/services
    PUT /vpn/ipsec/services/<service-id>
    DELETE /vpn/ipsec/services/<service-id>

IPSecVPNSession
    POST /vpn/ipsec/sessions
    PUT /vpn/ipsec/sessions/<session-id>
    DELETE /vpn/ipsec/sessions/<session-id>
```

You can call the following APIs to get the realized states:

```
EdgeCluster
Request - GET /edge-clusters/<edge-cluster-id>/state?request_id=<request-id>
Response - An instance of EdgeClusterStateDto which will inherit ConfigurationState. If the edge
cluster is deleted then the state will be unknown and it will return the common entity not found
error.

LogicalRouter / All L3 Entites - All L3 entities can use this API to get realization state
Request - GET /logical-routers/<logical-router-id>/state?request_id=<request-id>
Response - An instance of LogicalRouterStateDto which will inherit ConfigurationState. Delete
operation of any entity other than logical router can be covered by getting the state of logical
router but if the logical router itself is deleted then the state will be unknown and it will return
the common entity not found error.

LogicalServiceRouterCluster - All L3 entities which are the part of services can use this API to get
the realization state
Request - GET /logical-routers/<logical-router-id>/service-cluster/state?request_id=<request-id>
Response - An instance of LogicalServiceRouterClusterState which will inherit ConfigurationState.
```

```
LogicalRouterPort / DhcpRelayService / DhcpRelayProfile
Request – GET /logical-router-ports/<logical-router-port-id>/state?request_id=<request-id>
Response – An instance of LogicalRouterPortStateDto which will inherit ConfigurationState.

IPSecVPNIKEProfile / IPSecVPNDPDProfile / IPSecVPNTunnelProfile / IPSecVPNLocalEndpoint /
IPSecVPNPeerEndpoint / IPSecVPNService / IPSecVPNSession
Request – GET /vpn/ipsec/sessions/<session-id>/state?request_id=<request-id>
Response – An instance of IPSecVPNSessionStateDto which will inherit ConfigurationState. If the
session is deleted then the state will be unknown and it will return the common entity not found
error. When IPSecVPNService is disabled, IKE itself is down and it does not respond. It will return
unknown state in such a case.
```

For more information about the APIs, see the *NSX-T Data Center API Reference*.

Search for Objects

You can search for objects using various criteria throughout the NSX-T Data Center inventory.

The search results are sorted by relevance and you can filter these results based on your search query.

Note If you have special characters in your search query that also function as operators, then you must add a leading backslash. The characters that function as operators are: +, -, =, &&, ||, <, >, !, (,), {, }, [,], ^, ", ~, ?, :, /, \.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. On the homepage, enter a search pattern for an object or object type.

   As you enter your search pattern, the search feature provides assistance by showing the applicable keywords.

Search	Search Query
Objects with Logical as the name or property	Logical
Exact logical switch name	display_name:LSP-301
Names with special characters such as, !	Logical\!

   All the related search results are listed and grouped by resource type in different tabs.

   You can click the tabs for specific search results for a resource type.
3. (Optional) In the search bar, click the save icon to save your refined search criteria.
4. In the search bar, click the icon to open the advanced search column where you can refine your search.

5 Specify one or more criteria to refine your search.

- Name
- Resource Type
- Description
- ID
- Created by
- Modified by
- Tags
- Creation Date
- Modified Date

You can also view your recent search results and saved search criteria.

6 (Optional) Click **Clear All** to reset your advanced search criteria.

Add a Compute Manager

A compute manager, for example, vCenter Server, is an application that manages resources such as hosts and VMs.

NSX-T Data Center polls compute managers to find out about changes such as, the addition or removal of hosts or VMs and updates its inventory accordingly. It is optional to add a compute manager, because NSX-T Data Center gets the inventory information even without a compute manager, such as standalone hosts and VMs.

When you add a vCenter Server compute manager, you must provide a vCenter Server user's credentials. You can provide the vCenter Server administrator's credentials, or create a role and a user specifically for NSX-T Data Center and provide this user's credentials. This role must have the following vCenter Server privileges:

`Extension.Register extension`
`Extension.Unregister extension`
`Extension.Update extension`
`Sessions.Message`
`Sessions.Validate session`
`Sessions.View and stop sessions`
`Host.Configuration.Maintenance`
`Host.Local Operations.Create virtual machine`
`Host.Local Operations.Delete virtual machine`
`Host.Local Operations.Reconfigure virtual machine`
`Tasks`

`Scheduled task`
`Global.Cancel task`
`Permissions.Reassign role permissions`
`Resource.Assign vApp to resource pool`
`Resource.Assign virtual machine to resource pool`
`Virtual Machine.Configuration`
`Virtual Machine.Guest Operations`
`Virtual Machine.Provisioning`
`Virtual Machine.Inventory`
`Network.Assign network`
`vApp`

For more information about vCenter Server roles and privileges, see the *vSphere Security* document.

Prerequisites

- Verify that you use the supported vSphere version. See Supported vSphere version.
- IPv6 and IPv4 communication with vCenter Server.
- Verify that you use the recommended number of compute managers. See https://configmax.vmware.com/home.

  Note NSX-T Data Center does not support the same vCenter Server to be registered with more than one NSX Manager.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Fabric > Compute Managers > Add**.
3. Complete the compute manager details.

Option	Description
Name and Description	Type the name to identify the vCenter Server. You can optionally describe any special details such as, the number of clusters in the vCenter Server.
Domain Name/IP Address	Type the IP address of the vCenter Server.
Type	Keep the default option.
Username and Password	Type the vCenter Server login credentials.
Thumbprint	Type the vCenter Server SHA-256 thumbprint algorithm value.

If you left the thumbprint value blank, you are prompted to accept the server provided thumbprint.

After you accept the thumbprint, it takes a few seconds for NSX-T Data Center to discover and register the vCenter Server resources.

4. If the progress icon changes from **In progress** to **Not registered**, perform the following steps to resolve the error.
   1. Select the error message and click **Resolve**. One possible error message is the following:

      ```
      Extension already registered at CM <vCenter Server name> with id <extension ID>
      ```

   2. Enter the vCenter Server credentials and click **Resolve**.

      If an existing registration exists, it will be replaced.

Results

It takes some time to register the compute manager with vCenter Server and for the connection status to appear as `UP`.

You can click the compute manager's name to view the details, edit the compute manager, or to manage tags that apply to the compute manager.

Add an Active Directory

Active Directory is used in creating user-based Identity Firewall rules.

Windows 2008 is not supported as an Active Directory server or RDSH Server OS.

You can register one or more Windows domains with an NSX Manager. NSX Manager gets group and user information and the relationship between them from each domain that it is registered with. NSX Manager also retrieves Active Directory (AD) credentials.

Once NSX Manager retrieves AD credentials, you can create security groups based on user identity, and create identity-based firewall rules.

Note For Identity Firewall rule enforcement, Windows Time service should be **on** for all VMs using Active Directory. This ensures that the date and time is synchronized between Active Directory and VMs. Additionally, AD group membership changes, including enabling and deleting users, do not immediately take effect for logged in users. For changes to take effect, users must log out and then log back in. AD administrator's should force a log out when group membership is modified. This behavior is a limitation of Active Directory.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to **System > Active Directory**.

3 Click **Add Active Directory**.

4 Enter the name of the active directory.

5 Enter the **NetBios Name** and **Base Distinguished Name**.

To retrieve the netBIOS name for your domain, enter nbtstat /n in a command window on a Windows Workstation that is part of a domain, or on a domain controller. In the NetBIOS Local Name Table, the entry with a <00> prefix and type Group is the NetBIOS name.

6 Set the **Delta Synchronization Interval** if necessary. A delta synchronization updates local AD objects that have changed since the last synchronization event.

Any changes made in Active Directory are NOT seen on NSX Manager until a delta or full synchronization has been performed.

7 Click **Save**.

Add an LDAP Server

LDAP (Lightweight Directory Access Protocol) server configuration and functionality is only for use with Identity Firewall.

LDAP provides a central place for authentication, meaning that when you configure a connection to your LDAP server, the user records are stored in your external LDAP server.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Navigate to **System > Active Directory**.

3 Select the **LDAP Server** tab.

4 Click **Add LDAP Server** .

5 Enter the **Host** name of the LDAP server.

6 Select the active directory the LDAP server is connected to from the **Connected to (Directory)** drop-down menu.

7 (Optional) Select the **protocol**: LDAP (unsecured) or LDAPS (secured).

8 The default LDAP port 389 and LDAPs port 636 are used for the Active Directory sync, and should not be edited from the default values. Custom ports are not supported.

9 Enter the **username** and **password** of an Active Directory account with a minimum of read-only access to the Active Directory domain.

10 Click **Save**.

11 To verify that you can connect to the LDAP server, click **Test Connection**.

Synchronize Active Directory

Active Directory objects can be used to create security groups based on user identity, and identity-based firewall rules.

Note For Identity Firewall rule enforcement, Windows Time service should be **on** for all VMs using Active Directory. This ensures that the date and time is synchronized between Active Directory and VMs. AD group membership changes, including enabling and deleting users, do not immediately take effect for logged in users. For changes to take effect, users must log out and then log back in. AD administrator's should force a log out when group membership is modified. This behavior is a limitation of Active Directory.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Navigate to **System > Active Directory**.
3. Click the three button menu icon next to the Active Directory that you want to synchronize, and select one of the following:

Sync Delta	Perform a delta synchronization, where local AD objects that have changed since the last synchronization are updated.
Sync All	Perform a full synchronization, where the local state of all AD objects is updated.

4. Click **View Sync Status** to see the current state of the Active Directory, the previous synchronization state, the synchronization status, and the last synchronization time.

Managing User Accounts and Role-Based Access Control

NSX-T Data Center appliances have two built-in users: admin and audit. You can integrate NSX-T Data Center with VMware Identity Manager (vIDM) and configure role-based access control (RBAC) for users that vIDM manages.

For users managed by vIDM, the authentication policy that applies is the one configured by the vIDM administrator, and not NSX-T Data Center's authentication policy, which applies to users admin and audit only.

Manage a User's Password

Each appliance has two built-in users, admin and audit, that you can use to log in to NSX Manager or SSH to the appliance and run CLI commands. You can manage the password for these users but cannot add or delete users.

By default, the password expires after 90 days.

The audit user is not active by default. To activate it, log in as admin and run the `set user audit` command and provide a new password. When prompted for the current password, press the Enter key.

Prerequisites

Familiarize yourself with the password complexity requirements for NSX Manager and NSX Edge. See "NSX Manager Installation" and "NSX Edge Installation" in the *NSX-T Data Center Installation Guide*.

Procedure

1. Log in to the appliance's CLI.
2. To change the password, run the `set user` command. For example,

   ```
   nsx> set user admin
   Current password:
   New password:
   Confirm new password:
   nsx>
   ```

3. To get the password expiration information, run the `get user <username> password-expiration` command. For example,

   ```
   nsx> get user audit password-expiration
   Password expires 90 days after last change
   nsx>
   ```

4. To set the password expiration time in days, run the `set user <username> password-expiration <number of days>` command. For example,

   ```
   nsx> set user audit password-expiration 120
   nsx>
   ```

5. To disable password expiration, run the `clear user <username> password-expiration` command. For example,

   ```
   nsx> clear user audit password-expiration
   nsx>
   ```

Resetting the Passwords of an Appliance

If you have forgotten the `root`, `admin`, or `audit` user's password, you can reset it by booting the appliance into single-user mode.

Note If you have an NSX Manager cluster, resetting the password for the `root`, `admin`, or `audit` user on one NSX Manager will automatically reset the password for the other NSX Managers in the cluster.

Important When you reboot an appliance, the GRUB boot menu does not appear by default. The following procedure requires that you have configured the appliance to display the GRUB boot menu and you know the GRUB root user's password. For more information, see "Configure NSX-T Data Center to Display the GRUB Menu at Boot Time" in the *NSX-T Data Center Installation Guide*.

Procedure

1. If you are reseting a password on an NSX Manager, perform the following steps:
   a. Shut down the NSX Manager.
   b. Download the Ubuntu 16.04 `.iso` file from http://releases.ubuntu.com/16.04/ubuntu-16.04.6-server-amd64.iso.
   c. Launch the vSphere or ESXi graphical user interface (GUI).
   d. Import the Ubuntu `.iso` file into the applicable datastore for the NSX Manager VM.
   e. Edit the settings of the NSX Manager VM and add a CD ROM Drive device if it does not exist.
   f. In the **CD ROM Drive** configuration, check the **Connect at power on** checkbox.
   g. In **CD/DVD Media**, press **Browse** and select `ubuntu-16.04.6-server-amd64.iso` from the applicable datastore.
   h. Click **Save** to exit the **Edit settings** page.
   i. Power on the NSX Manager.
2. Connect to the console of the appliance.
3. Reboot the system.
4. When the GRUB boot menu appears, press the left `SHIFT` or `ESC` key quickly. If you wait too long and the boot sequence does not pause, you must reboot the system again.
5. Press `e` to edit the menu.

   Enter the user name (`root`) and password. Note that this is GRUB root user, which is not the same as the appliance's root user.
6. Keep the cursor on the Ubuntu selection.
7. Press `e` to edit the selected option.

8 Search for the line starting with `linux`.

9 Remove all options after `root=UUID=`.

10 Add the following option.

```
rw single init=/bin/bash
```

11 Press `Ctrl-X` to boot.

12 When the log messages stop, press Enter.

You will see the prompt `root@(none):/#`.

13 If you are resetting the password for `root`, run the command `passwd`.

If you are resetting the password for `admin` or `audit`, run the command `passwd <admin or audit user ID>`.

You can run the `passwd` command multiple times.

14 Enter a new password.

15 Enter the password again.

16 Run the command `sync`.

17 Run the command `reboot -f`.

Important: If you are resetting a password on an NSX Manager, after running this command, press the **ESC** key in a timely manner so that you can perform the next step. If you wait too long and the boot sequence does not pause, reboot the system again.

18 If you are resetting a password on an NSX Manager, and you successfully paused the boot sequence in the previous step, perform the following steps:

  a Scroll down to **<Enter Setup>** using the down arrow key and press **Enter**.

  b Navigate to the Boot menu option using the right arrow key.

  c Make CD-ROM the first device using the **+** or **-** key.

  d Press **F10** to save and exit.

  e Press **Enter** for the **Yes** option to save configuration changes and exit.

  This will reboot and the BIOS banner page will be shown. Do not press any keys.

  f After a few seconds Ubuntu from the CD-ROM drive's `.iso` file will start.

  g Select a language and press **Enter**.

  You will see a Ubuntu menu.

  h Select **Rescue a broken system** using the down arrow key and press **Enter**.

  i On successive screens, select a language, country and keyboard layout and press **Enter**.

  j Enter a temporary hostname or accept the default.

  k Set the correct time and timezone if necessary.

l You will be prompted to enter a device to use as the root file system. Select the **Do not use a root file system** option using the down arrow key and press **Enter**.

m You will now be prompted to enter rescue mode. Select **Execute a shell in the installer environment** and press **Enter**.

n Confirm by selecting the **Continue** option and press **Enter**.

o You will now enter a Linux shell. Enter the following Linux commands:

```
mount /dev/sda2 /mnt
mount --bind /dev /mnt/dev
chroot /mnt
mount /config
touch /config/vmware/nsx-node-api/reset_cluster_credentials
umount /conifg
exit
umount /mnt/dev
umount /mnt
sync
exit
```

p You will now see the **Enter rescue mode** screen again, select the **Reboot the system** option using the down arrow key and press **Enter**.

When you see the BIOS banner page, press the **ESC** key quickly.

q Scroll down using the down arrow key to **<Enter Setup>** and press **Enter**.

r Navigate to the Boot menu option using the right arrow key.

s Navigate to the **Hard Drive** option using the down arrow key and press **+** until it is the first device.

t Press **F10** to save and exit.

u Press **Enter** for the **Yes** option to save configuration changes and exit. The system will reboot.

v When the GRUB menu appears, select the Ubuntu option and press **Enter**.

The NSX Manager will start and have the new password.

w As time permits, remove the CD ROM device using the **Edit Settings** option in the vSphere or ESXi GUI for the NSX Manager VM.

Authentication Policy Settings

You can view or change the authentication policy settings through the CLI.

You can view or set the minimum password length with the following commands:

```
get auth-policy minimum-password-length
set auth-policy minimum-password-length <password-length>
```

The following commands apply to logging in to the NSX Manager UI, or making an API call:

```
get auth-policy api lockout-period
get auth-policy api lockout-reset-period
get auth-policy api max-auth-failures
set auth-policy api lockout-period <lockout-period>
set auth-policy api lockout-reset-period <lockout-reset-period>
set auth-policy api max-auth-failures <auth-failures>
```

The following commands apply to logging in to the CLI on an NSX Manager or an NSX Edge node:

```
get auth-policy cli lockout-period
get auth-policy cli max-auth-failures
set auth-policy cli lockout-period <lockout-period>
set auth-policy cli max-auth-failures <auth-failures>
```

For more information about the CLI commands, see the *NSX-T Command-Line Interface Reference*.

By default, after five consecutive failed attempts to log in to the NSX Manager UI, the administrator account is locked for 15 minutes. You can disable account lockout with the following command:

```
set auth-policy api lockout-period 0
```

Similarly, you can disable account lockout for the CLI with the following command:

```
set auth-policy cli lockout-period 0
```

Obtain the Certificate Thumbprint from a vIDM Host

Before you configure the integration of vIDM with NSX-T, you must get the certificate thumbprint from the vIDM host.

You must use OpenSSL version 1.x or higher for the thumbprint. In the vIDM host, the command `openssl` runs an older OpenSSL version and therefore you must use the command `openssl1` in the vIDM host. This command is only available from the vIDM host.

In a server that is not the vIDM host, you can use the `openssl` command that is running OpenSSL version 1.x or higher.

Procedure

1 Log in to the vIDM host's console or by using SSH or log in to any server that can ping the vIDM host.

2 Use OpenSSL version 1.x or higher to get the thumbprint of the vIDM host.

- *openssl1*: If you are logged in to the vIDM host in a console or using SSH, run the following command to get the thumbprint:

  ```
  openssl1 s_client -connect <FQDN of vIDM host>:443 < /dev/null 2> /dev/null | openssl x509 -
  sha256 -fingerprint -noout -in /dev/stdin
  ```

- *openssl*: If you are logged in to a server that can ping the vIDM host but is not the vIDM host, run the following command to get the thumbprint:

  ```
  openssl s_client -connect <FQDN of vIDM host>:443 < /dev/null 2> /dev/null | openssl x509 -
  sha256 -fingerprint -noout -in /dev/stdin
  ```

Configure VMware Identity Manager Integration

You can integrate NSX-T Data Center with VMware Identity Manager (vIDM), which provides identity management services.

The vIDM server should have a certificate signed by a certificate authority (CA). Otherwise, logging in to vIDM from NSX Manager might not work with certain browsers, such as Microsoft Edge or Internet Explorer 11. For information about installing a CA-signed certificate on vIDM, see the VMware Identity Manager documentation at https://docs.vmware.com/en/VMware-Identity-Manager/index.html.

When you register NSX Manager with vIDM, you specify a redirect URI that points to NSX Manager. You can provide either the fully qualified domain name (FQDN) or the IP address. It is important to remember whether you use the FQDN or the IP address. When you try to log in to NSX Manager through vIDM, you must specify the host name in the URL the same way, that is, if you use the FQDN when registering the manager with vIDM, you must use the FQDN in the URL, and if you use the IP address when registering the manager with vIDM, you must use the IP address in the URL. Otherwise, login will fail.

Note NSX Managers and vIDM must be in the same time zone. The recommended way is to use UTC.

With vIDM enabled, you can still log in to NSX Manager with a local user account if you use the URL `https://<nsx-manager-ip-address>/login.jsp?local=true`.

If you use the UserPrincipalName (UPN) to log in to vIDM, authentication to NSX-T might fail. To avoid this issue, use a different type of credentials, for example, SAMAccountName.

If using NSX Cloud, you can log in to CSM separately using the URL `https://<csm-ip-address>/login.jsp?local=true`

Prerequisites

- Verify that you have the certificate thumbprint from the vIDM host. See Obtain the Certificate Thumbprint from a vIDM Host.

- Verify that NSX Manager is registered as an OAuth client to the vIDM host. During the registration process, note the client ID and the client secret. For more information, see the VMware Identity Manager documentation at https://docs.vmware.com/en/VMware-Identity-Manager/index.html

  NSX Cloud Note If using NSX Cloud, also verify that CSM is registered as an OAuth client on the vIDM host.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **System > Users**.

3 Click the **Configuration** tab.

4 Click **Edit**.

5 To enable external load balancer integration, click the **External Load Balancer Integration** toggle.

   Note If you have Virtual IP (VIP) set up (check **System > Appliances > Virtual IP**), you cannot use the **External Load Balancer Integration** even if you enable it. This is because you can either have VIP or the External Load Balancer while configuring vIDM but not both. Disable VIP if you want to use the External Load Balancer. See Configure a Virtual IP (VIP) Address for a Cluster in the *NSX-T Data Center Installation Guide* for details.

6 To enable VMware Identity Manager integration, click the **VMware Identity Manager Integration** toggle.

7 Provide the following information.

Parameter	Description
VMware Identity Manager Appliance	The fully qualified domain name (FQDN) of the vIDM host.
OAuth Client ID	The ID that is created when registering NSX Manager to the vIDM host.
OAuth Client Secret	The secret that is created when registering NSX Manager to the vIDM host.
SSL Thumbprint	The certificate thumbprint of the vIDM host.
NSX Appliance	The IP address or fully qualified domain name (FQDN) of NSX Manager. If you are using an NSX Manager cluster, use the load balancer FQDN or cluster VIP FQDN or IP address. If you specify a FQDN, you must access NSX Manager from a browser using the manager's FQDN in the URL, and if you specify an IP address, you must use the IP address in the URL. Alternatively, the vIDM administrator can configure the NSX Manager client so that you can connect using either the FQDN or the IP address.

8 Click **Save**.

9 If using NSX Cloud, repeat steps 1 through 8 from the CSM appliance by logging in to CSM instead of NSX Manager.

Time Synchronization between NSX Manager, vIDM, and Related Components

For authentication to work correctly, NSX Manager, vIDM and other service providers such as Active Directory must all be time synchronized. This section describes how to time synchronize these components.

VMware Infrastructure

Follow the instructions in the following KB articles to synchronize ESXi hosts.

- https://kb.vmware.com/kb/1003736
- https://kb.vmware.com/kb/2012069

Third-Party Infrastructure

Follow the vendor's documentation on how synchronize VMs and hosts.

Configuring NTP on the vIDM Server (Not Recommended)

If you are not able to synchronize time across the hosts, you can disable synchronizing to host and configure NTP on the vIDM server. This method is not recommend because it requires the opening of UDP port 123 on the vIDM server

- Check the clock on the vIDM server and make sure it is correct.

  ```
  # hwclock
  Tue May  9 12:08:43 2017  -0.739213 seconds
  ```

- Edit `/etc/ntp.conf` and add the following entries if they don't exist.

  ```
  server server time.nist.gov
  server server pool.ntp.org
  server server time.is dynamic
  ```

- Open UDP port 123.

  ```
  # iptables -A INPUT -p udp --dport 123 -j ACCEPT
  ```

  Run the following command to check that the port is open.

  ```
  # iptables -L -n
  ```

- Start the NTP service.

  ```
  /etc/init.d/ntp start
  ```

- Make NTP run automatically after a reboot.

  ```
  # chkconfig --add ntp
  # chkconfig ntp on
  ```

- Check that the NTP server can be reached.

  ```
  # ntpq -p
  ```

  The `reach` column should not show 0. The `st` column should show some number other than 16..

Role-Based Access Control

With role-based access control (RBAC), you can restrict system access to authorized users. Users are assigned roles and each role has specific permissions.

There are four types of permissions:

- Full access
- Execute
- Read
- None

Full access gives the user all permissions. The execute permission includes the read permission.

NSX-T Data Center has the following built-in roles. You cannot add any new roles.

- Enterprise Administrator
- Auditor
- Network Engineer
- Network Operations
- Security Engineer
- Security Operations
- Cloud Service Administrator
- Cloud Service Auditor
- Load Balancer Administrator
- Load Balancer Auditor
- VPN Administrator
- Guest Introspection Administrator
- Network Introspection Administrator

After an Active Directory (AD) user is assigned a role, if the username is changed on the AD server, you need to assign the role again using the new username.

Roles and Permissions

Table 21-1. Roles and Permissions shows the permissions each role has for different operations. The following abbreviations are used:

- EA - Enterprise Administrator
- A - Auditor
- NE - Network Engineer
- NO - Network Operations
- SE - Security Engineer
- SO - Security Operations
- CS Adm - Cloud Service Administrator
- CS Aud - Cloud Service Auditor
- LB Adm - Load Balancer Administrator
- LB Aud - Load Balancer Auditor
- VPN Adm - VPN Administrator
- GI Adm - Guest Introspection Administrator
- NI Adm - Network Introspection Administrator
- FA - Full access
- E - Execute
- R - Read

Table 21-1. Roles and Permissions

Operation	EA	A	NE	NO	SE	SO	CS Adm	CS Aud	LB Adm	LB Aud	VPN Adm	GI Adm	NI Adm
Tools > Port Connection	E	R	E	E	E	E	E	R	E	E	None	None	None
Tools > Traceflow	E	R	E	E	E	E	E	R	E	E	None	None	None
Tools > Port Mirroring	FA	R	FA	FA	FA	FA	FA	R	None	None	None	None	None
Tools > IPFIX	FA	R	FA	R	FA	R	FA	R	None	None	R	R	R
Firewall > General	FA	R	R	R	FA	R	FA	R	None	None	None	None	R

Table 21-1. Roles and Permissions (continued)

Operation	EA	A	NE	NO	SE	SO	CS Adm	CS Aud	LB Adm	LB Aud	VPN Adm	GI Adm	NI Adm
Firewall > Configuration	FA	R	R	R	FA	R	FA	R	None	None	None	None	None
Routing > Routers	FA	R	FA	R	R	R	FA	R	R	R	None	None	None
Routing > NAT	FA	R	FA	R	FA	R	FA	R	R	R	None	None	None
DHCP > Server Profiles	FA	R	FA	R	FA	None	FA	R	None	None	None	None	None
DHCP > Servers	FA	R	FA	R	FA	None	FA	R	None	None	None	None	None
DHCP > Relay Profiles	FA	R	FA	R	FA	None	FA	R	None	None	None	None	None
DHCP > Relay Services	FA	R	FA	R	FA	None	FA	R	None	None	None	None	None
DHCP > Metadata Proxies	FA	R	FA	R	FA	None	None	None	None	None	None	None	None
IPAM	FA	R	FA	R	FA	None	None	None	None	None	None	None	None
Switching > Switches	FA	R	FA	FA	R	R	FA	R	R	R	None	None	None
Switching > Ports	FA	R	FA	FA	R	R	FA	R	R	R	None	None	None
Switching > Switching Profiles	FA	R	FA	FA	FA	FA	FA	R	R	R	None	None	None
Policy > Networking > Load Balancers	FA	R	None	None	None	None	FA	R	FA	R	None	None	None
Load Balancing > Virtual Servers	FA	R	None	None	None	None	FA	R	FA	R	None	None	None

Table 21-1. Roles and Permissions (continued)

Operation	EA	A	NE	NO	SE	SO	CS Adm	CS Aud	LB Adm	LB Aud	VPN Adm	GI Adm	NI Adm
Load Balancing > Profiles > Application Profiles	FA	R	None	None	None	None	FA	R	FA	R	None	None	None
Load Balancing > Profiles > Persistence Profiles	FA	R	None	None	None	None	FA	R	FA	R	None	None	None
Load Balancing > Profiles > SSL Profiles	FA	R	None	None	FA	R	FA	R	FA	R	None	None	None
Load Balancing > Server Pools	FA	R	None	None	None	None	FA	R	FA	R	None	None	None
Load Balancing > Monitors	FA	R	None	None	None	None	FA	R	FA	R	None	None	None
Inventory > Groups	FA	R	FA	R	FA	R	FA	R	R	R	R	R	R
Inventory > IP Sets	FA	R	FA	R	FA	R	FA	R	R	R	R	R	R
Inventory > IP Pools	FA	R	FA	R	None	R	None	None	R	R	R	R	R
Inventory > MAC Sets	FA	R	FA	R	FA	R	FA	R	R	R	R	R	R
Inventory > Services	FA	R	FA	R	FA	R	FA	R	R	R	R	R	R
Inventory > Virtual Machines	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21-1. Roles and Permissions (continued)

Operation	EA	A	NE	NO	SE	SO	CS Adm	CS Aud	LB Adm	LB Aud	VPN Adm	GI Adm	NI Adm
Inventory > VM > Create & Assign Tags	FA	R	FA	FA	FA	FA	FA	R	R	R	R	FA	FA
Inventory > VM > Configure Tags	FA	None	None	None	FA	None	None	None	None	None	None	None	None
Fabric > Nodes > Hosts	FA	R	R	R	R	R	R	R	None	None	None	None	None
Fabric > Nodes > Nodes	FA	R	FA	R	FA	R	R	R	None	None	None	None	None
Fabric > Nodes > Edges	FA	R	FA	R	R	R	R	R	None	None	None	None	None
Fabric > Nodes > Edge Clusters	FA	R	FA	R	R	R	R	R	None	None	None	None	None
Fabric > Nodes > Bridges	FA	R	FA	R	R	R	None	None	R	R	None	None	None
Fabric > Nodes > Transport Nodes	FA	R	R	R	R	R	R	R	R	R	None	None	None
Fabric > Nodes > Tunnels	R	R	R	R	R	R	R	R	R	R	None	None	None
Fabric > Profiles > Uplink Profiles	FA	R	R	R	R	R	R	R	R	R	None	None	None
Fabric > Profiles > Edge Cluster Profiles	FA	R	FA	R	R	R	R	R	R	R	None	None	None

Table 21-1. Roles and Permissions (continued)

Operation	EA	A	NE	NO	SE	SO	CS Adm	CS Aud	LB Adm	LB Aud	VPN Adm	GI Adm	NI Adm
Fabric > Profiles > Configuration	FA	R	None	None	None	None	R	R	None	None	None	None	None
Fabric > Transport Zones > Transport Zones	FA	R	R	R	R	R	R	R	R	R	None	None	None
Fabric > Transport Zones > Transport Zone Profiles	FA	R	R	R	R	R	R	R	R	R	None	None	None
Fabric > Compute Managers	FA	R	R	R	R	R	R	R	None	None	None	R	R
System > Trust	FA	R	None	None	FA	R	None	None	FA	R	FA	None	None
System > Configuration	FA	R	None	None	None	None	None	None	None	None	None	None	None
System > Utilities > Support Bundle	FA	R	R	R	R	R	R	R	None	None	None	None	None
System > Utilities > Backup	FA	R	None	None	None	None	None	None	None	None	None	None	None
System > Utilities > Restore	FA	R	None	None	None	None	None	None	None	None	None	None	None
System > Utilities > Upgrade	FA	R	R	R	R	R	None	None	None	None	None	None	None

Table 21-1. Roles and Permissions (continued)

Operation	EA	A	NE	NO	SE	SO	CS Adm	CS Aud	LB Adm	LB Aud	VPN Adm	GI Adm	NI Adm
System > Users > Role Assignments	FA	R	None	None	None	None	None	None	None	None	None	None	None
System > Users > Configuration	FA	R	None	None	None	None	None	None	None	None	None	None	None

Add a Role Assignment or Principal Identity

You can assign roles to users or user groups if VMware Identity Manager is integrated with NSX-T Data Center. You can also assign roles to principal identities.

A principal is an NSX-T Data Center component or a third-party application such as an OpenStack product. With a principal identity, a principal can use the identity name to create an object and ensure that only an entity with the same identity name can modify or delete the object. A principal identity has the following properties:

- Name
- Node ID - this can be any alphanumeric value assigned to a principal identity
- Certificate
- RBAC role indicating the access rights of this principal

Users (local, remote, or principal identity) with the Enterprise Administrator role can modify or delete objects owned by principal identities. Users (local, remote, or principal identity) without the Enterprise Administrator role cannot modify or delete protected objects owned by principal identities, but can modify or delete unprotected objects.

If a principal identity user's certificate expires, you must import a new certificate and make an API call to update the principal identity user's certificate (see the procedure below). For more information about the NSX-T Data Center API, a link to the API resource is available at https://docs.vmware.com/en/VMware-NSX-T-Data-Center.

A principal identity user's certificate must satisfy the following requirements:

- SHA256 based.
- RSA/DSA message algorithm with 2048 bits or above key size.
- Cannot be a root certificate.

You can delete a principal identity using the API. However, deleting a principal identity does not automatically delete the corresponding certificate. You must delete the certificate manually.

Steps to delete a principal identity and its certificate:

1 Get the details of the principal identity to delete and note the `certificate_id` value in the response.

   ```
   GET /api/v1/trust-management/principal-identities/<principal-identity-id>
   ```

2 Delete the principal identity.

   ```
   DELETE /api/v1/trust-management/principal-identities/<principal-identity-id>
   ```

3 Delete the certificate using the `certificate_id` value obtained in step 1.

   ```
   DELETE /api/v1/trust-management/certificates/<certificate_id>
   ```

Prerequisites

- If you want to assign roles to users, verify that a vIDM host is associated with NSX-T. For more information, see Configure VMware Identity Manager Integration.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Users**.
3. To assign roles to users, select **Add > Role Assignment**.
   a. Select a user or user group.
   b. Select a role.
   c. Click **Save**.
4. To add a principal identity, select **Add > Principal Identity with Role**.
   a. Enter a name for the principal identity.
   b. Select a role.
   c. Enter a node ID.
   d. Enter a certificate in PEM format.
   e. Click **Save**.
5. (Optional) If using NSX Cloud, log in to the CSM appliance instead of NSX Manager and repeat steps 1 through 4.

6 If the certificate for the principal identity expires, perform the following steps:

   a Import a new certificate and note the certificate's ID. See Import a Certificate.

   b Call the following API to get the ID of the principal identity.

   ```
   GET https://<nsx-mgr>/api/v1/trust-management/principal-identities
   ```

   c Call the following API to update the principal identity's certificate. You must provide the imported certificate's ID and the principal identity user's ID.

   For example,

   ```
   POST https://<nsx-mgr>/api/v1/trust-management/principal-identities?action=update_certificate
   {
       "principal_identity_id": "ebd3032d-728e-44d4-9914-d4f81c9972cb",
       "certificate_id" : "abd3032d-728e-44d4-9914-d4f81c9972cc"
   }
   ```

Backing Up and Restoring the NSX Manager

If the NSX Manager cluster becomes inoperable, or if you want to restore your environment to a previous state, you can restore from a backup. While the NSX Manager is inoperable, the data plane is not affected, but you cannot make configuration changes.

There are two types of backups:

Cluster backup

This backup includes the desired state of the virtual network.

Node backup

This is a backup of the NSX Manager nodes.

There are two backup methods:

Manual

You manually run the backup at any time.

Automated

Automated backups run based on a schedule that you set. Automated backups are highly recommended to ensure that you have up-to-date backups.

You can restore an NSX-T Data Center configuration back to the state that is captured in any of the backups. When restoring a backup, you must restore to new NSX Manager appliances running the same version of NSX Manager as the appliances that were backed up.

Configure Backups

Before backups can occur, you must configure a backup file server. After a backup file server is configured, you can start a backup at any time, or configure a schedule for automatic backups.

Prerequisites

Verify that you have the SSH fingerprint of the backup file server. Only an SHA256 hashed ECDSA (256 bit) key is accepted as a fingerprint. See Find the SSH Fingerprint of a Remote Server.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Backup & Restore**.
3. Click **Edit** in the upper right of the page to configure backups.
4. Enter the IP address or host name of the backup file server.
5. Change the default port if required.
6. The protocol field is already filled in. Do not change the value.

   SFTP is the only supported protocol.
7. Enter the user name and password required to log in to the backup file server.

   The first time you configure a file server, you must provide a password. Subsequently, if you reconfigure the file server, and the server IP (or hostname), port, and user name are the same, you do not need to enter the password again.
8. In the **Destination Directory** field, enter the absolute directory path where the backups will be stored.

   The directory must already exist and cannot be /. If you have multiple NSX-T Data Center deployments, you must use a different directory for each deployment. If the backup file server is a Windows machine, you still use the forward slash when you specify the destination directory. For example, if the backup directory on the Windows machine is `c:\SFTP_Root\backup`, specify `/SFTP_Root/backup` as the destination directory.

   Note The backup process will generate a name for the backup file that can be quite long. On a Windows server, the length of the full path name of the backup file can exceed the limit set by Windows and cause backups to fail. To avoid this issue, see the KB article https://kb.vmware.com/s/article/76528.
9. To encrypt the backups, click the **Change Encryption Passphrase** toggle and enter the encryption passphrase.

   You will need this passphrase to restore a backup. If you forget the passphrase, you cannot restore any backups.
10. Enter the SSH fingerprint of the server that stores the backups.

    You can leave this blank and accept or reject the fingerprint provided by the server.
11. Click the **Schedule** tab.

12 To enable automatic backups, click the **Automatic Backup** toggle.

13 Click **Weekly** and set the days and time of the backup, or click **Interval** and set the interval between backups.

14 To trigger a backup when the configuration of the network changes, set the **Detect NSX configuration change** toggle to **Enabled**.

You can set the interval between the backups triggered by configuration changes. The default is 5 minutes.

15 Click **Save**.

Results

After you configure a backup file server, you can click **Backup Now** to start a backup at any time.

Removing Old Backups

Backups can accumulate on the backup file server and consume a large amount of storage. You can run a script that comes with NSX-T Data Center to automatically delete old backups.

You can find the Python script `nsx_backup_cleaner.py` in the directory `/var/vmware/nsx/file-store` on NSX Manager. You must log in as root to access this file. Typically, you schedule a job on the backup file server to run this script periodically to clean up old backups. The following usage information describes how to run the script:

```
nsx_backup_cleaner.py -d backup_dir [-k 1] [-l 5] [-h]
Or
nsx_backup_cleaner.py --dir backup_dir [--retention-period 1] [--min-count 5] [--help]

Required parameters:
    -d/--dir: Backup root directory
    -k/--retention-period: Number of days need to retain a backup file

Optional parameters:
    -l/--min-count: Minimum number of backup files to be kept, default value is 100
    -h/--help: Display help message
```

The age of a backup is calculated as the difference between the backup's timestamp and the time the script is run. If this value is larger than the retention period, the backup is deleted if there are more backups on the disk than the minimum number of backups.

For more information about setting up the script to run periodically on a Linux or Windows server, see the comments at the beginning of the script.

Listing Available Backups

The backup file server stores backups from all the NSX Managers. To get the list of backups so that you can find the one you want to restore, you must run the `get_backup_timestamps.sh` script.

The script is located on an NSX Manager. The full path name is `/var/vmware/nsx/file-store/get_backup_timestamps.sh`. You can run this script on any Linux machine or NSX-T Data Center appliance. As a best practice, you should copy this script after installing NSX-T Data Center to a machine that is not an NSX Manager so that you can run this script even if all the NSX Managers become inaccessible. If you need to restore a backup but have no access to this script, you can install a new NSX Manager and run the script there.

You can copy the script to another machine or to the backup file server by logging in to the NSX Manager as admin and running a CLI command. For example:

```
nsxmgr-1> copy file get_backup_timestamps.sh url scp://admin@10.127.1.20/tmp/
admin@10.127.1.20's password:
nsxmgr-1>
```

The script is interactive and will prompt you for the information that you specified when you configured the backup file server. You can specify the number of backups to display. Each backup is listed with a timestamp, the NSX Manager node's IP address or FQDN if the NSX Manager node is set up to publish its FQDN, and the node ID. For example,

```
admin@host1:/home/admin# ./get_backup_timestamps.sh
Enter file server ip:
10.108.115.108
Enter port:
22
Enter directory path:
/home/nsx/backups
Enter number of latest backup or press Enter to list all backups:

root@10.108.115.108's password:
Latest backups:
[Backup timestamp; IP address/FQDN; Node id]
2019-01-22;09:00:33 10.196.196.77 35163642-6623-8f6d-7af0-52e03f16faed
2019-01-22;09:01:52 10.196.196.77 35163642-6623-8f6d-7af0-52e03f16faed
2019-01-22;09:13:30 wdc75.platformqe.com 41893642-597b-915f-5117-7da576df4ff2
2019-01-22;09:14:42 wdc75.platformqe.com 41893642-597b-915f-5117-7da576df4ff2
2019-01-22;09:16:43 wdc75.platformqe.com 41893642-597b-915f-5117-7da576df4ff2
```

Restore a Backup

Restoring a backup results in restoring the state of the network at the time of the backup. In addition, the configurations maintained by the NSX Manager are also restored and any changes, such as adding or deleting nodes, that were made to the fabric since the backup was taken are reconciled.

You must restore a backup on a new installation of NSX Manager, as described in the first step below.

Prerequisites

- Verify that you have the login credential for the backup file server.

- Verify that you have the SSH fingerprint of the backup file server. Only an SHA256 hashed ECDSA (256 bit) key is accepted as a fingerprint. See Find the SSH Fingerprint of a Remote Server.
- Verify that you have the passphrase of the backup file.

Procedure

1 Install a new NSX Manager node on which to restore the backup.

   If the original NSX Manager was configured with the default setting of not publishing its FQDN, that is `"publish_fqdns": false`, then the new installation of NSX Manager must be installed with the same IP address that was used by the original NSX Manager. If the original NSX Manager was set up to publish its FQDN, that is `"publish_fqdns": true`, the new NSX Manager can be installed with a different IP address. However, the new NSX Manager must also be configured to publish its FQDN. If you had an NSX Manager cluster when the backup was taken, you must also restore to an NSX Manager cluster. The restore process restores one NSX Manager node first and then prompts you to add the other NSX Manager nodes.

   a Power down all NSX Manager nodes.

   b Deploy a new NSX Manager node with the same name and IP address of the original NSX Manager node.

      To identify the original NSX Manager node, open the NSX Manager dashboard and navigate to **System > Appliances** to view the Management Cluster. This displays the NSX Manager nodes. The original node is the one whose Deployment Type shows as Manual.

   After the new NSX Manager node is running and online, you can proceed with the rest of the procedure.

2 From your browser, log in with admin privileges to a new NSX Manager.

   The IP address or FQDN of this NSX Manager node must be the same as the IP address or FQDN of the NSX Manager where the backup was taken.

3 Select **System > Backup & Restore**.

4 Click the **Restore** tab.

5 To configure the backup file server, click **Edit**.

6 Enter the IP address or host name.

7 Change the port number, if necessary.

   The default is 22.

8 To log in to the server, enter the user name and password.

9 In the **Destination Directory** text box, enter the absolute directory path where the backups are stored.

10 Enter the passphrase that was used to encrypt the backup data.

11 Enter the SSH fingerprint of the server that stores the backups.

12 Click **Save**.

13 Select a backup.

14 Click **Restore**.

   The status of the restore operation is displayed. If you have deleted or added fabric nodes or transport nodes since the backup, you are prompted to take certain actions, for example, log in to a node and run a script.

   If the backup has information about an NSX Manager cluster, you are prompted to add NSX Manager nodes. If you decide not to add NSX Manager nodes, you can still proceed with the restore.

   After the restore operation is completed, the Restore Complete screen is displayed, showing the result of the restore, the timestamp of the backup file, and the start and end time of the restore operation. If the restore failed, the screen displays the step where the failure occurred, for example, `Current Step: Restoring Cluster (DB)` or `Current Step: Restoring Node`. If either cluster restore or node restore failed, the error might be transient. In that case, there is no need to click **Retry**. You can restart or reboot the manager and the restore continues. You can also determine that there was a cluster restore or node restore failure by running the following CLI command to view the system log file and searching for the strings `Cluster restore failed` and `Node restore failed`.

   ```
   get log-file syslog
   ```

   To restart the manager, run the following CLI command:

   ```
   restart service manager
   ```

   To reboot the manager, run the following CLI command:

   ```
   reboot
   ```

15 After the first NSX Manager node is up and functional, deploy two additional nodes to complete the NSX Manager cluster.

   See *DeployNSX Manager Nodes to Form a Cluster from UI* in the *NSX-T Data Center Installation Guide*.

16 After the new NSX Manager cluster is deployed, delete the original NSX Manager cluster VMs that you powered down in Step 1a. of this procedure.

Results

Note If you added a compute manager after the backup, after the restore, if you try to add the compute manager again, you will get an error message indicating that registration failed. You can click the **Resolve** button to resolve the error and successfully add the compute manager. For more information, see Add a Compute Manager, step 4. If you want to remove the information about NSX-T Data Center that is stored in a vCenter Server, follow the steps in Remove NSX-T Data Center Extension from vCenter Server.

Remove NSX-T Data Center Extension from vCenter Server

When you add a compute manager, NSX Manager adds its identity as an extension in vCenter Server. If you remove the compute manager, the extension in vCenter Server will be removed automatically. If the extension is not removed for some reason, you can maually remove the extension with the following procedure.

Prerequisites

Enable access to the vCenter Server Managed Object Browser (MOB) by following the procedure in https://kb.vmware.com/s/article/2042554.

Procedure

1 Login to the MOB at `https://<vCenter Server hostname or IP address>/mob`.

2 Click the **content** link, which is the value for the **content** property in the Properties table.

3 Click the **ExtensionManager** link, which is the value for **extensionManager** property in the Properties table.

4 Click the **UnregisterExtension** link in the Methods table.

5 Enter `com.vmware.nsx.management.nsxt` in the **value** text field.

6 Click the **Invoke Method** link on the right hand side of the page below the Parameters table.

   The method result says `void` but the extension will be removed.

7 To make sure the extension is removed, click the **FindExtension** method on the previous page and invoke it by entering the same value for the extension.

   The result should be `void`.

Managing the NSX Manager Cluster

You can reboot an NSX Manager if it becomes inoperable. You can also change the IP address of an NSX Manager.

In a production environment, it is highly recommended that the NSX Manager cluster has three members to provide high availability. If you delete an NSX Manager and deploy a new one, the new NSX Manager can have the same or a different IP address.

Note The primary NSX Manager node is the node that you create first, before you create a manager cluster. This node cannot be deleted. After you deploy two more manager nodes from the primary manager node's UI to form a cluster, only the second and the third manager nodes have the option (from the gear icon) to be deleted. For information about removing and adding a manager node, see Change the IP Address of an NSX Manager.

View the Configuration and Status of the NSX Manager Cluster

You can view the configuration and status of the NSX Manager cluster from the NSX Manager UI. You can get additional information using the CLI.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://*nsx-manager-ip-address*.

2 Select **System > Overview**

The status of the NSX Manager cluster is displayed.

3 To see additional information about the configuration, run the following CLI command:

```
manager1> get cluster config
Cluster Id: 18807edd-56d1-4107-b7b7-508d766a08e3
Cluster Configuration Version: 3
Number of nodes in the cluster: 3

Node UUID: 43cd0642-275c-af1d-fe46-1f5200f9e5f9
Node Status: JOINED
    ENTITY                               UUID                                       IP
ADDRESS      PORT     FQDN
    HTTPS                                5c8d01f1-f3ee-4f94-b517-a093d8fbfad3
10.160.71.225   443      ychin-nsxmanager-ob-12065118-1-F5
    CONTROLLER                           06fd0574-69c0-432e-a8af-53d140dbef8f
10.160.71.225   -        ychin-nsxmanager-ob-12065118-1-F5
    CLUSTER_BOOT_MANAGER                 da8d535e-7a0c-4dd8-8919-d88bdde006b8
10.160.71.225   -        ychin-nsxmanager-ob-12065118-1-F5
    DATASTORE                            3c9c4ec1-afef-47bd-aadb-1ed6a5536bc4
10.160.71.225   9000     ychin-nsxmanager-ob-12065118-1-F5
    MANAGER                              eb5e8922-23bd-4c3a-ae22-d13d9195a6bc
10.160.71.225   -        ychin-nsxmanager-ob-12065118-1-F5
    POLICY                               f9da1039-08ad-4a20-bacc-5b91c5d67730
10.160.71.225   -        ychin-nsxmanager-ob-12065118-1-F5

Node UUID: 8ebb0642-201e-6a5f-dd47-a1e38542e672
Node Status: JOINED
    ENTITY                               UUID                                       IP
ADDRESS      PORT     FQDN
    HTTPS                                3757f155-8a5d-4b53-828f-d67041d5a210
```

```
10.160.93.240   443      ychin-nsxmanager-ob-12065118-2-F5
    CONTROLLER                           7b1c9952-8738-4900-b68b-ca862aa4f6a9
10.160.93.240   -        ychin-nsxmanager-ob-12065118-2-F5
    CLUSTER_BOOT_MANAGER                 b5e12db1-5e0d-4e33-a571-6ba258dceb2e
10.160.93.240   -        ychin-nsxmanager-ob-12065118-2-F5
    DATASTORE                            bee1f629-4e23-4ab8-8083-9e0f0bb83178
10.160.93.240   9000     ychin-nsxmanager-ob-12065118-2-F5
    MANAGER                              45ccd6e3-1497-4334-944c-e6bbcd5c723e
10.160.93.240   -        ychin-nsxmanager-ob-12065118-2-F5
    POLICY                               d5ba5803-b059-4fbc-897c-3aace8cf1219
10.160.93.240   -        ychin-nsxmanager-ob-12065118-2-F5

Node UUID: 2e7e0642-df4a-b2ec-b9e8-633d1469f1ea
Node Status: JOINED
    ENTITY                               UUID                                       IP
ADDRESS      PORT     FQDN
    HTTPS                                bce3cc4c-7d60-45e2-aa7b-cdc75e445a14
10.160.76.33    443      ychin-nsxmanager-ob-12065118-3-F5
    CONTROLLER                           ced46f5c-9e52-4b31-a1cb-b3dead991c71
10.160.76.33    -        ychin-nsxmanager-ob-12065118-3-F5
    CLUSTER_BOOT_MANAGER                 88b70d31-3428-4ccc-ab57-55859f45030c
10.160.76.33    -        ychin-nsxmanager-ob-12065118-3-F5
    DATASTORE                            fb4aec3c-cae3-4386-b5b9-c0b99b7d9048
10.160.76.33    9000     ychin-nsxmanager-ob-12065118-3-F5
    MANAGER                              82b07440-3ff6-4f67-a1c9-e9327d1686ad
10.160.76.33    -        ychin-nsxmanager-ob-12065118-3-F5
    POLICY                               61f21a78-a56c-4af1-867b-3f24132d53c7
10.160.76.33    -        ychin-nsxmanager-ob-12065118-3-F5
```

4 To see additional information about the status, run the following CLI command:

```
manager1> get cluster status
Cluster Id: 18807edd-56d1-4107-b7b7-508d766a08e3
Group Type: DATASTORE
Group Status: STABLE

Members:
    UUID                                       FQDN
IP               STATUS
    43cd0642-275c-af1d-fe46-1f5200f9e5f9       ychin-nsxmanager-ob-12065118-1-F5
10.160.71.225    UP
    8ebb0642-201e-6a5f-dd47-a1e38542e672       ychin-nsxmanager-ob-12065118-2-F5
10.160.93.240    UP
    2e7e0642-df4a-b2ec-b9e8-633d1469f1ea       ychin-nsxmanager-ob-12065118-3-F5
10.160.76.33     UP

Group Type: CLUSTER_BOOT_MANAGER
Group Status: STABLE

Members:
    UUID                                       FQDN
IP               STATUS
    43cd0642-275c-af1d-fe46-1f5200f9e5f9       ychin-nsxmanager-ob-12065118-1-F5
10.160.71.225    UP
    8ebb0642-201e-6a5f-dd47-a1e38542e672       ychin-nsxmanager-ob-12065118-2-F5
```

```
10.160.93.240    UP
    2e7e0642-df4a-b2ec-b9e8-633d1469f1ea       ychin-nsxmanager-ob-12065118-3-F5
10.160.76.33     UP

Group Type: CONTROLLER
Group Status: STABLE

Members:
    UUID                                       FQDN
IP               STATUS
    7b1c9952-8738-4900-b68b-ca862aa4f6a9       ychin-nsxmanager-ob-12065118-2-F5
10.160.93.240    UP
    ced46f5c-9e52-4b31-a1cb-b3dead991c71       ychin-nsxmanager-ob-12065118-3-F5
10.160.76.33     UP
    06fd0574-69c0-432e-a8af-53d140dbef8f       ychin-nsxmanager-ob-12065118-1-F5
10.160.71.225    UP

Group Type: MANAGER
Group Status: STABLE

Members:
    UUID                                       FQDN
IP               STATUS
    43cd0642-275c-af1d-fe46-1f5200f9e5f9       ychin-nsxmanager-ob-12065118-1-F5
10.160.71.225    UP
    8ebb0642-201e-6a5f-dd47-a1e38542e672       ychin-nsxmanager-ob-12065118-2-F5
10.160.93.240    UP
    2e7e0642-df4a-b2ec-b9e8-633d1469f1ea       ychin-nsxmanager-ob-12065118-3-F5
10.160.76.33     UP

Group Type: POLICY
Group Status: STABLE

Members:
    UUID                                       FQDN
IP               STATUS
    43cd0642-275c-af1d-fe46-1f5200f9e5f9       ychin-nsxmanager-ob-12065118-1-F5
10.160.71.225    UP
    8ebb0642-201e-6a5f-dd47-a1e38542e672       ychin-nsxmanager-ob-12065118-2-F5
10.160.93.240    UP
    2e7e0642-df4a-b2ec-b9e8-633d1469f1ea       ychin-nsxmanager-ob-12065118-3-F5
10.160.76.33     UP

Group Type: HTTPS
Group Status: STABLE

Members:
    UUID                                       FQDN
IP               STATUS
    43cd0642-275c-af1d-fe46-1f5200f9e5f9       ychin-nsxmanager-ob-12065118-1-F5
10.160.71.225    UP
    8ebb0642-201e-6a5f-dd47-a1e38542e672       ychin-nsxmanager-ob-12065118-2-F5
10.160.93.240    UP
    2e7e0642-df4a-b2ec-b9e8-633d1469f1ea       ychin-nsxmanager-ob-12065118-3-F5
10.160.76.33     UP
```

Reboot an NSX Manager

You can reboot an NSX Manager with a CLI command to recover from critical errors.

If you need to reboot multiple NSX Managers, you must reboot them one at a time. Wait for the rebooted NSX Manager to be online before rebooting another.

Procedure

1. Log in to the CLI of the NSX Manager.
2. Run the following command.

   ```
   nsx-manager> reboot
   Are you sure you want to reboot (yes/no): y
   ```

Change the IP Address of an NSX Manager

You can change the IP address of an NSX Manager in an NSX Manager cluster. This section describes several approaches.

For example, if you have a cluster consisting of Manager A, Manager B, and Manager C, you can change the IP address of one or more of the managers in the following ways:

- Scenario A:
  - Manager A has IP address 172.16.1.11.
  - Manager B has IP address 172.16.1.12.
  - Manager C has IP address 172.16.1.13.
  - Add Manager D with a new IP address, for example, 192.168.55.11.
  - Remove Manager A.
  - Add Manager E with a new IP address, for example, 192.168.55.12.
  - Remove Manager B.
  - Add Manager F with a new IP address, for example, 192.168.55.13.
  - Remove Manager C.
- Scenario B:
  - Manager A has IP address 172.16.1.11.
  - Manager B has IP address 172.16.1.12.
  - Manager C has IP address 172.16.1.13.
  - Add Manager D with a new IP address, for example, 192.168.55.11.
  - Add Manager E with a new IP address, for example, 192.168.55.12.
  - Add Manager F with a new IP address, for example, 192.168.55.13.
  - Remove Manager A, Manager B, and Manager C.

- Scenario C:
  - Manager A has IP address 172.16.1.11.
  - Manager B has IP address 172.16.1.12.
  - Manager C has IP address 172.16.1.13.
  - Remove Manager A.
  - Add Manager D with a new IP address, for example, 192.168.55.11.
  - Remove Manager B.
  - Add Manager E with a new IP address, for example, 192.168.55.12.
  - Remove Manager C.
  - Add Manager F with a new IP address, for example, 192.168.55.13.

The first two scenarios require additional virtual RAM, CPU and disk for the additional NSX Managers during this IP address change.

Scenario C is not recommended because it temporarily reduces the number of NSX Managers and a loss of one of the two active managers during the IP address change will have an impact on the operations of NSX-T. This scenario is for a situation where additional virtual RAM, CPU and disk are not available and an IP address change is required.

Note If you are using the cluster VIP feature, you must either use the same subnet for the new IP addresses or disable the cluster VIP during the IP address changes because the cluster VIP requires all NSX Managers to be in the same subnet.

Prerequisites

Familiarize yourself with how to deploy an NSX Manager into a cluster. For more information, see the *NSX-T Data Center Installation Guide*.

Procedure

1 If the NSX Manager you want to remove was deployed manually, perform the following steps.

   a Run the following CLI command to detach the NSX Manager from the cluster.

   ```
   detach node <node-id>
   ```

   b Delete the NSX Manager VM.

2 If the NSX Manager you want to delete was deployed automatically through the NSX Manager UI, perform the following steps.

   a From your browser, log in with administrator privileges to an NSX Manager at https://*nsx-manager-ip-address*.

      This NSX Manager must not be the one that you want to delete.

   b Click the **Systems** tab.

      The status of the NSX Manager cluster is displayed.

   c For the NSX Manager that you want to delete, click the gear icon and select **Delete**.

3 Deploy a new NSX Manager.

Resize an NSX Manager Node

You can change the number of CPU cores or memory of an NSX Manager node at any time.

Note that in normal operating conditions all three manager nodes must have the same number of CPU cores and memory. A mismatch of CPU or memory between NSX Managers in an NSX management cluster should only be done when transitioning from one size of NSX Manager to another size of NSX Manager.

If you have configured resource allocation reservation for the NSX Manager VMs in vCenter Server, you might need to adjust the reservation. For more information, see the vSphere documentation.

Prerequisites

- Verify that the new size satisfies the system requirements for a manager node. For more information, see "NSX Manager VM System Requirements" in the *NSX-T Data Center Installation Guide*.
- Familiarize yourself with how to deploy an NSX Manager into a cluster. For more information, see the *NSX-T Data Center Installation Guide*.
- For information about how to remove a manager node from a cluster, see Change the IP Address of an NSX Manager.

Procedure

1 Deploy a new manager node with the new size.

2 Add the new manager node to the cluster.

3 Remove an old manager node.

4 Repeat steps 1 to 3 to replace the other two old manager nodes.

Multisite Deployment of NSX-T Data Center

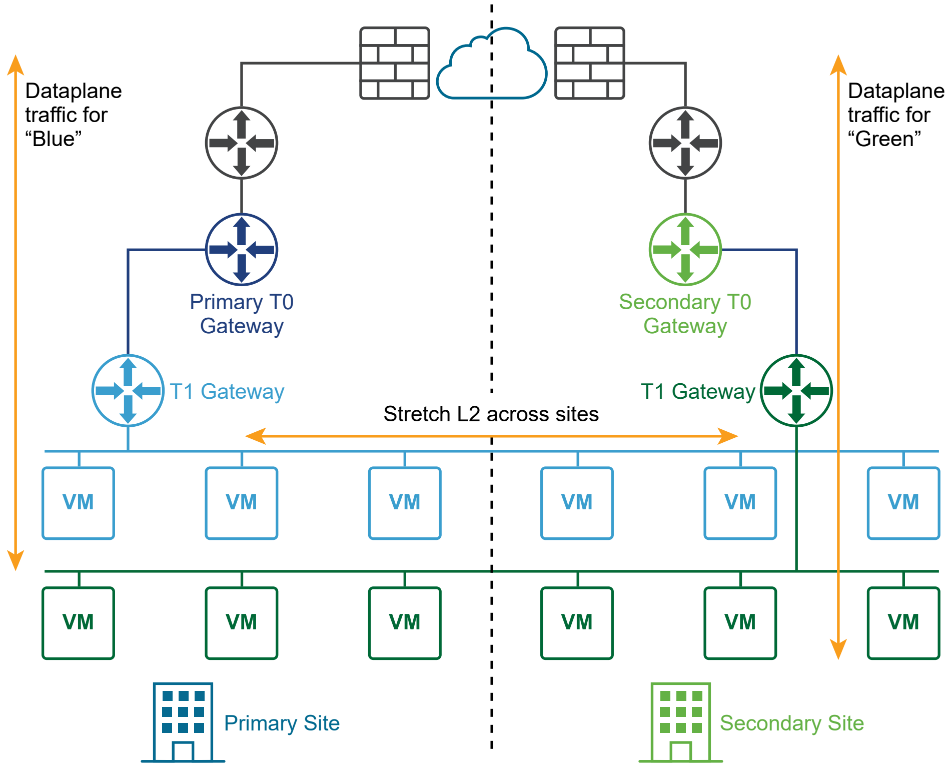
NSX-T Data Center supports multisite deployments where you can manage all the sites from one NSX Manager cluster.

Two types of multisite deployments are supported:

- Active-active
- Disaster recovery

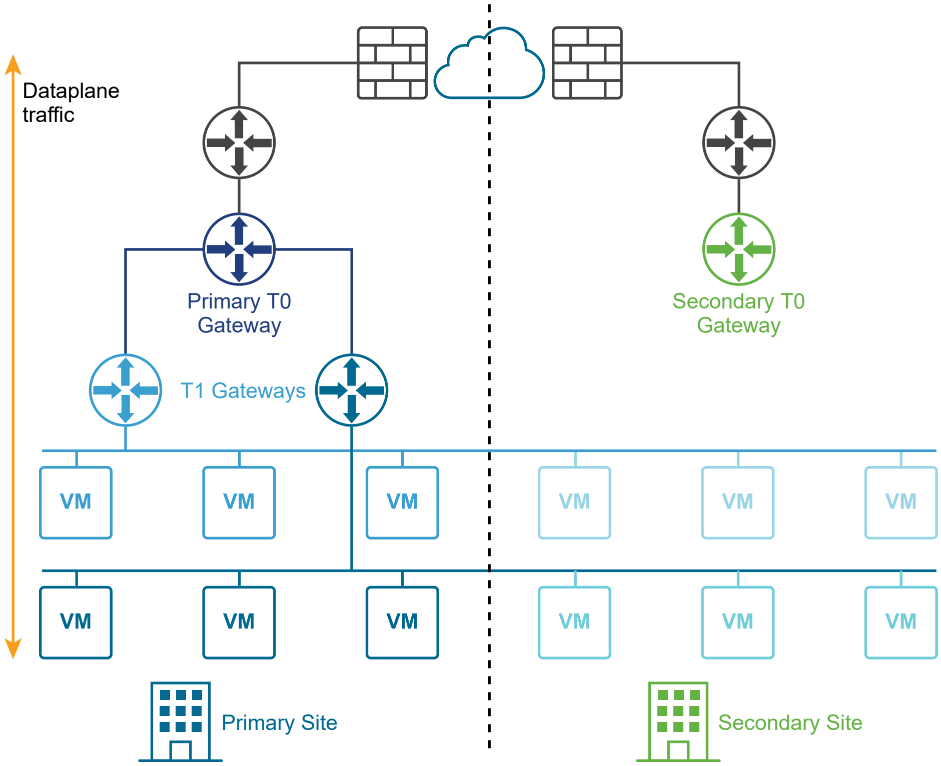
In an active-active deployment, all the sites are active and layer 2 traffic crosses the site boundaries. In a disaster recovery deployment, NSX-T Data Center at the primary site handles networking for the enterprise. The secondary site is standing by to take over if a catastrophic failure occurs at the primary site.

The following diagram illustrates an active-active deployment.

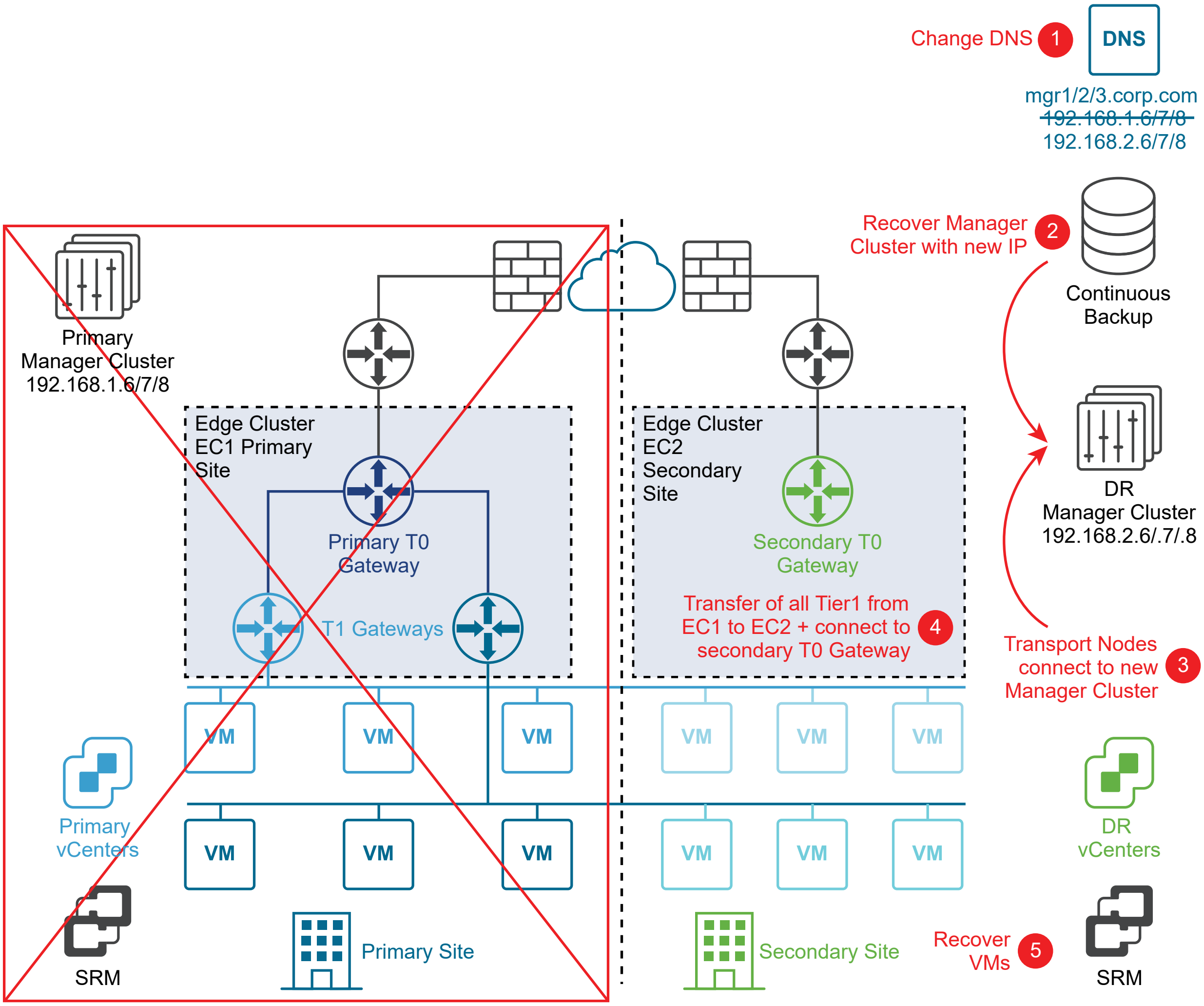


In an active-active deployment, if the primary gateway fails, it will fail over to the secondary gateway. If the primary site fails, all the steps described for disaster recovery below have to be completed.

The following diagram illustrates a disaster recovery deployment.



The following diagram illustrates how disaster recovery occurs.


Change DNS ①
DNS
mgr1/2/3.corp.com
~~192.168.1.6/7/8~~
192.168.2.6/7/8
Recover Manager Cluster with new IP ②
Continuous Backup
Primary Manager Cluster 192.168.1.6/7/8
Edge Cluster EC1 Primary Site
Primary T0 Gateway
T1 Gateways
Edge Cluster EC2 Secondary Site
Secondary T0 Gateway
Transfer of all Tier1 from EC1 to EC2 + connect to secondary T0 Gateway ④
DR Manager Cluster 192.168.2.6/.7/.8
Transport Nodes connect to new Manager Cluster ③
VM
Primary vCenters
DR vCenters
SRM
Primary Site
Secondary Site
Recover VMs ⑤
SRM

The recovery steps are:

1 Change the DNS record so that the NSX Manager cluster has different IP addresses.

2 Restore the NSX Manager cluster from a backup.

3 Connect the transport nodes to the new NSX Manager cluster.

4 Transfer tier-1 gateways from NSX Edge clusters at the primary site to NSX Edge clusters at the secondary site.

5 Recover the VMs.

Requirements for Multisite Deployments

Inter-site Communication

- The bandwidth must be at least 1 Gbps and the latency (RTT) must be less than 150 ms.
- MTU must be at least 1600. 9000 is recommended.

NSX Manager Configuration

- Automatic backup when NSX-T Data Center configuration changes must be enabled.
- NSX Manager must be set up to use FQDN.

Data Plane Recovery

- The same internet provider must be used if public IP addresses are exposed through services such as NAT or load balancer.

Cloud Management System

- The cloud management system (CMS) must support an NSX-T Data Center plug-in. In this release, VMware Integrated OpenStack (VIO) and vRealize Automation (vRA) satisfy this requirement.

Limitations

- No local-egress capabilities. All north-south traffic must occur within one site.
- Compute disaster recovery orchestration must support NSX-T Data Center.

Configuring Appliances

Some system configuration tasks must be done using the command line or API.

For complete command line interface information, see the *NSX-T Data Center Command-Line Interface Reference*. For complete API information, see the *NSX-T Data Center API Guide*.

Table 21-2. System configuration commands and API requests.

Task	**Command Line (NSX Manager and NSX Edge)**	**API Request (NSX Manager only)**
Set system timezone	`set timezone <timezone>`	`PUT https://<nsx-mgr>/api/v1/node`
Set NTP Server	`set ntp-server <ntp-server>`	`PUT https://<nsx-mgr>/api/v1/node/services/ntp`
Set a DNS server	`set name-servers <dns-server>`	`PUT https://<nsx-mgr>/api/v1/node/network/name-servers`
Set DNS Search Domain	`set search-domains <domain>`	`PUT https://<nsx-mgr>/api/v1/node/network/search-domains`

Add a License Key and Generate a License Usage Report

You can add license keys and generate a license usage report. The usage report is a file in CSV format.

The following non-evaluation NSX-T Data Center license types are available:

- Standard
- Professional

- Advanced
- Enterprise Plus

When you install NSX Manager, a pre-installed evaluation license becomes active and is valid for 60 days. The evaluation license provides all the features of an enterprise license. You cannot install or unassign an evaluation license.

You can install one or more of the non-evaluation licenses, but for each type, you can only install one key. When you install a standard, advanced, or enterprise license, the evaluation license is no longer available. You can also unassign non-evaluation licenses. If you unassign all non-evaluation licenses, the evaluation license is restored.

If you have multiple keys of the same license type and want to combine the keys, you must go to https://my.vmware.com and use the Combine Keys functionality. The NSX Manager UI does not provide this functionality.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Licenses > Add**.
3. Enter a license key.
4. To generate a license usage report, select **Export > License Usage Report**.

   The CSV report lists the VM, CPU, unique concurrent user, and vCPU usage numbers of the following features:

   - Switching and Routing
   - NSX Edge load balancer
   - VPN
   - DFW
   - Context Aware Micro-Segmentation - Application identification
   - Context Aware Micro-Segmentation - Identity firewall for remote desktop session host
   - Service Insertion
   - Identity Firewall
   - Enhanced Guest Introspection

Setting Up Certificates

You can import certificates, create a certificate signing request (CSR), generate self-signed certificates, and import a certificate revocation list (CRL).

After you install NSX-T Data Center, the manager nodes and cluster have self-signed certificates. To improve security, it is highly recommended that you replace the self-signed certificates with CA-signed certificates.

Import a Certificate

You can import a certificate with a private key to replace the default self-signed certificate after activation.

Note that only RSA-based certificates are supported.

Prerequisites

Verify that a certificate is available.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Certificates**.
3. Select **Import > Import Certificate** and enter the certificate details.

Option	Description
Name	Assign a name to the certificate.
Certificate Contents	Browse to the certificate file on your computer and add the file. The certificate must not be encrypted. If it is a CA-signed certificate, be sure to include the whole chain in this order: certificate - intermediate - root.
Private Key	Browse to the private key file on your computer and add the file.
Passphrase	Add a passphrase for this certificate if it is encrypted. In this release, this field is not used because encrypted certificate is not supported.
Description	Enter a description of what is included in this certificate.
Service Certificate	Set to **Yes** to use this certificate for services such as a load balancer and VPN. Set to **No** if this certificate is for the NSX Manager nodes.

4. Click **Import**.

Create a Certificate Signing Request File

Certificate signing request (CSR) is an encrypted text that contains specific information such as, organization name, common name, locality, and country. You send the CSR file to a certificate authority (CA) to apply for a digital identity certificate.

Prerequisites

- Gather the information that you need to fill out the CSR file. You must know the FQDN of the server and the organizational unit, organization, city, state, and country.
- Verify that the public and private key pairs are available.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **System > Certificates**.

3 Click the **CSRs** tab.

4 Click **Generate CSR**.

5 Complete the CSR file details.

Option	Description
Name	Assign a name for your certificate.
Common Name	Enter the fully qualified domain name (FQDN) of your server. For example, test.vmware.com.
Organization Name	Enter your organization name with applicable suffixes. For example, VMware Inc.
Organization Unit	Enter the department in your organization that is handling this certificate For example, IT department.
Locality	Add the city in which your organization is located. For example, Palo Alto.
State	Add the state in which your organization is located. For example, California.
Country	Add the country in which your organization is located. For example, United States (US).
Message Algorithm	Set the encryption algorithm for your certificate. RSA encryption - is used for digital signatures and encryption of the message. Therefore, it is slower than DSA when creating an encrypted token but faster to analyze and validate this token. This encryption is slower to decrypt and faster to encrypt. DSA encryption - is used for digital signatures. Therefore, it is faster than RSA when creating an encrypted token but slower to analyze and validate this token. This encryption is faster to decrypt and slower to encrypt.
Key Size	Set the key bits size of the encryption algorithm. The default value, 2048, is adequate unless you specifically need a different Key size. Many CAs require a minimum value of 2048. Larger key sizes are more secure but have a greater impact on performance.
Description	Enter specific details to help you identify this certificate at a later date.

6 Click **Generate**.

A custom CSR appears as a link.

7 Select the CSR and click **Actions**.

8 Select **Download CSR PEM** from the drop-down menu.

   You can save the CSR PEM file for your records and CA submission.

9 Use the contents of the CSR file to submit a certificate request to the CA in accordance with the CA enrollment process.

Results

The CA creates a server certificate based on the information in the CSR file, signs it with its private key, and sends you the certificate. The CA also sends you a root CA certificate.

Import a CA Certificate

You can import a signed CA certificate. After the import and activation, other certificates signed by that CA will be trusted by NSX-T Data Center.

Note that only RSA-based certificates are supported.

Prerequisites

Verify that a CA certificate is available.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **System > Certificates**.

3 Select **Import > Import CA Certificate** and enter the certificate details.

Option	Description
Name	Assign a name to the CA certificate.
Certificate Contents	Browse to the CA certificate file on your computer and add the file.
Description	Enter a summary of what is included in this CA certificate.
Service Certificate	Set to **Yes** to use this certificate for services such as a load balancer and VPN. Set to **No** if this certificate is for the NSX Manager nodes.

4 Click **Import**.

Create a Self-Signed Certificate

You can create a self-signed certificate. However, using a self-signed certificate is less secure than using a trusted certificate.

When you use a self-signed certificate the client user receives a warning message such as, `Invalid Security Certificate`. The client user must then accept the self-signed certificate when first connecting to the server in order to proceed. Allowing client users to select this option provides reduced security than other authorization methods.

Prerequisites

Verify that a CSR is available. See Create a Certificate Signing Request File.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Certificates**.
3. Click the **CSRs** tab.
4. Select a CSR.
5. Select **Actions > Self Sign Certificate for CSR**.
6. Enter the number of days the self-sign certificate is valid.

   The default is 10 years.
7. Click **Add**.

Results

The self-signed certificate appears in the **Certificates** tab.

Replace the Certificate for an NSX Manager Node or an NSX Manager Cluster Virtual IP

You can replace the certificate for a manager node or the manager cluster virtual IP (VIP) by making an API call.

After you install NSX-T Data Center, the manager nodes and cluster have self-signed certificates. To improve security, it is highly recommended that you replace the self-signed certificates with CA-signed certificates and that you use a different certificate for each node.

In release 2.4, replacing an existing certificate with a CA-signed certificate might fail. This issue is fixed in release 2.4.1.

Prerequisites

Verify that a certificate is available in the NSX Manager. See Import a Certificate.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Certificates**.
3. In the ID column, click the ID of the certificate you want to use and copy the certificate ID from the pop-up window.

   Make sure that when this certificate was imported, the option **Service Certificate** was set to **No**.

4. To replace the certificate of a manager node, use the `POST /api/v1/node/services/http?action=apply_certificate` API call. For example,

   ```
   POST https://<nsx-mgr>/api/v1/node/services/http?
   action=apply_certificate&certificate_id=e61c7537-3090-4149-b2b6-19915c20504f
   ```

   For more information, see the *NSX-T Data Center API Reference*.

5. To replace the certificate of the manager cluster VIP, use the `POST /api/v1/cluster/api-certificate?action=set_cluster_certificate` API call. For example,

   ```
   POST https://<nsx-mgr>/api/v1/cluster/api-certificate?
   action=set_cluster_certificate&certificate_id=d60c6a07-6e59-4873-8edb-339bf75711ac
   ```

   For more information, see the *NSX-T Data Center API Reference*. This step is not necessary if you did not configure VIP.

Import a Certificate Revocation List

A certificate revocation list (CRL) is a list of subscribers and their certificate status. When a potential user attempts to access a server, the server denies access based on the CRL entry for that particular user.

The list contains the following items:

- Revoked certificates and the reasons for revocation
- Dates the certificates are issued
- Entities that issued the certificates
- Proposed date for the next release

Prerequisites

Verify that a CRL is available.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Certificates**.
3. Click the **CRLs** tab.

4 Click **Import** and add the CRL details.

Option	Description
Name	Assign a name to the CRL.
Certificate Contents	Copy all of the items in the CRL and paste them in this section. A sample CRL.

```
-----BEGIN X509 CRL-----
MIIBODCB4zANBgkqhkiG9w0BAQQFADBgMQswCQYDVQQGEwJBVTEMMAoGA1UECBM
D
UUxEMRkwFwYDVQQKExBNaW5jb20gUHR5LiBMdGQuMQswCQYDVQQLEwJDUzEbMBk
G
A1UEAxMSU1NMZWF5IGRlbW8gc2VydmVyFw0wMTAxMTUxNjI2NTdaFw0wMTAyMTQ
x
NjI2NTdaMFIwEgIBARcNOTUxMDA5MjMzMjA1WjASAgEDFw05NTEyMDEwMTAwMDB
a
MBMCAhI0Fw0wMTAxMTUxNjE5NDdaMBMCAhI1Fw0wMTAxMTUxNjIzNDZaMA0GCSq
G
SIb3DQEBBAUAA0EAHPjQ3M93QOj8Ufi+jZM7Y78TfAzG4jJn/
E6MYBPFVQFYo/Gp
UZexfjSVo5CIyySOtYscz8oO7avwBxTiMpDEQg==
-----END X509 CRL--
```

Option	Description
Description	Enter a summary of what is included in this CRL.

5 Click **Import**.

Results

The imported CRL appears as a link.

Configuring NSX Manager to Retrieve a Certificate Revocation List

Using the API, you can configure NSX Manager to retrieve a certificate revocation list (CRL). You can then check the CRL by making an API call to NSX Manager instead of to the certificate authority.

This feature provides the following benefits:

- It is more efficient to have the CRL cached on the server, that is, NSX Manager.
- The client does not need to create any outbound connection to the certificate authority.

The following APIs related to certificate revocation lists are available:

```
GET /api/v1/trust-management
GET /api/v1/trust-management/crl-distribution-points
POST /api/v1/trust-management/crl-distribution-points
DELETE /api/v1/trust-management/crl-distribution-points/<crl-distribution-point-id>
GET /api/v1/trust-management/crl-distribution-points/<crl-distribution-point-id>
PUT /api/v1/trust-management/crl-distribution-points/<crl-distribution-point-id>
GET /api/v1/trust-management/crl-distribution-points/<crl-distribution-point-id>/status
POST /api/v1/trust-management/crl-distribution-points/pem-file
```

You can manage CRL distribution points and retrieve the CRLs stored in NSX Manager. For more information, see the *NSX-T Data Center API Reference*.

Import a Certificate for a CSR

You can import a signed certificate for a CSR.

When you use a self-signed certificate the client user receives a warning message such as, `Invalid Security Certificate`. The client user must then accept the self-signed certificate when first connecting to the server in order to proceed. Allowing client users to select this option provides reduced security than other authorization methods.

Prerequisites

Verify that a CSR is available. See Create a Certificate Signing Request File.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System** > **Certificates**.
3. Click the **CSRs** tab.
4. Select a CSR.
5. Select **Actions** > **Import Certificate for CSR**.
6. Browse to the signed certificate file on your computer and add the file.
7. Click **Add**.

Results

The self-signed certificate appears in the **Certificates** tab.

Storage of Public Certificates and Private Keys

Public certificates and private keys are stored on the NSX Managers. When a load balancer or a VPN service is created that requires a private key, NSX Manager sends a copy of the private key to the Edge node where the load balancer or VPN service is running.

Collect Support Bundles

You can collect support bundles on registered cluster and fabric nodes and download the bundles to your machine or upload them to a file server.

If you choose to download the bundles to your machine, you get a single archive file consisting of a manifest file and support bundles for each node. If you choose to upload the bundles to a file server, the manifest file and the individual bundles are uploaded to the file server separately.

NSX Cloud Note If you want to collect the support bundle for CSM, log in to CSM, go to **System > Utilities > Support Bundle** and click on **Download**. The support bundle for PCG is available from NSX Manager using the following instructions. The support bundle for PCG also contains logs for all the workload VMs.

Procedure

1 From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.

2 Select **System > Support Bundle**

3 Select the target nodes.

   The available types of nodes are **Management Nodes**, **Edges**, **Hosts**, and **Public Cloud Gateways**.

4 (Optional) Specify log age in days to exclude logs that are older than the specified number of days.

5 (Optional) Toggle the switch that indicates whether to include or exclude core files and audit logs.

   Note Core files and audit logs might contain sensitive information such as passwords or encryption keys.

6 (Optional) Select the check box to upload the bundles to a file server.

7 Click **Start Bundle Collection** to start collecting support bundles.

   Depending on how many log files exist, each node might take several minutes.

8 Monitor the status of the collection process.

   The status tab shows the progress of collecting support bundles.

9 Click **Download** to download the bundle if the option to send the bundle to a file server was not set.

Log Messages

Log messages from all NSX-T Data Center components, including those running on ESXi hosts, conform to the syslog format as specified in RFC 5424. Log messages from KVM hosts are in the RFC 3164 format. The log files are in the directory `/var/log`.

On NSX-T Data Center appliances, you can run the following NSX-T Data Center CLI command to view the logs:

```
get log-file <auth.log | http.log | kern.log | manager.log | node-mgmt.log | syslog> [follow]
```

On hypervisors, you can use Linux commands such as `tac`, `tail`, `grep`, and `more` to view the logs. You can also use these commands on NSX-T Data Center appliances.

For more information about RFC 5424, see https://tools.ietf.org/html/rfc5424. For more information about RFC 3164, see https://tools.ietf.org/html/rfc3164.

RFC 5424 defines the following format for log messages:

```
<facility * 8 + severity> version UTC-TZ hostname APP-NAME procid MSGID [structured-data] msg
```

A sample log message:

```
<187>1 2016-03-15T22:53:00.114Z nsx-manager NSX - SYSTEM [nsx@6876 comp="nsx-manager"
errorCode="MP4039" subcomp="manager"] Connection verification failed for broker '10.160.108.196'.
Marking broker unhealthy.
```

Every message has the component (`comp`) and sub-component (`subcomp`) information to help identify the source of the message.

NSX-T Data Center produces logs with facility `local6`, which has a numerical value of 22. Each API call produces one audit log, which contains `audit="true"` in the structured data field.

An audit log that is associated with an API call has the following information:

- An entity ID parameter `entId` to identify the object of the API.
- A request ID parameter `req-id` to identify a specific API call.
- An external request ID parameter `ereqId` if the API call contains the header `X-NSX-EREQID:<string>`.
- An external user parameter `euser` if the API call contains the header `X-NSX-EUSER:<string>`.

RFC 5424 defines the following severity levels:

Severity Level	Description
0	Emergency: system is unusable
1	Alert: action must be taken immediately
2	Critical: critical conditions
3	Error: error conditions
4	Warning: warning conditions
5	Notice: normal but significant condition
6	Informational: informational messages
7	Debug: debug-level messages

All logs with a severity of emergency, alert, critical, or error contain a unique error code in the structured data portion of the log message. The error code consists of a string and a decimal number. The string represents a specific module.

The `MSGID` field identifies the type of message. For a list of the message IDs, see Log Message IDs.

Configure Remote Logging

You can configure NSX-T Data Center appliances and hypervisors to send log messages to a remote logging server.

Remote logging is supported on NSX Manager, NSX Edge, and hypervisors. You must configure remote logging on each node individually.

On an KVM host, the NSX-T Data Center installation package automatically configures the rsyslog daemon by putting configuration files in the `/etc/rsyslog.d` directory.

Prerequisites

- Configure a logging server to receive the logs.

Procedure

1 To configure remote logging on an NSX-T Data Center appliance:

  a Run the following command to configure a log server and the types of messages to send to the log server. Multiple facilities or message IDs can be specified as a comma delimited list, without spaces.

```
set logging-server <hostname-or-ip-address[:port]> proto <proto> level <level> [facility
<facility>] [messageid <messageid>] [certificate <filename>] [structured-data <structured-
data>]
```

  For more information about this command, see the *NSX-T CLI Reference*. You can run the command multiple times to add multiple logging server configurations. For example:

```
nsx> set logging-server 192.168.110.60 proto udp level info facility syslog messageid
SYSTEM,FABRIC
nsx> set logging-server 192.168.110.60 proto udp level info facility auth,user
```

  b you can view the logging configuration with the `get logging-server` command. For example,

```
nsx> get logging-servers
192.168.110.60 proto udp level info facility syslog messageid SYSTEM,FABRIC
192.168.110.60 proto udp level info facility auth,user
```

2 To configure remote logging on an ESXi host:

   a Run the following commands to configure syslog and send a test message:

```
esxcli network firewall ruleset set -r syslog -e true
esxcli system syslog config set --loghost=udp://<log server IP>:<port>
esxcli system syslog reload
esxcli system syslog mark -s "This is a test message"
```

   b You can run the following command to display the configuration:

```
esxcli system syslog config get
```

3 To configure remote logging on a KVM host:

   a Edit the file `/etc/rsyslog.d/10-vmware-remote-logging.conf` for your environment.

   b Add the following line to the file:

```
*.* @<ip>:514;RFC5424fmt
```

   c Run the following command:

```
service rsyslog restart
```

Log Message IDs

In a log message, the message ID field identifies the type of message. You can use the `messageid` parameter in the `set logging-server` command to filter which log messages are sent to a logging server.

Table 21-3. Log Message IDs

Message ID	Examples
FABRIC	Host node Host preparation Edge node Transport zone Transport node Uplink profiles Cluster profiles Edge cluster Bridge clusters and endpoints
SWITCHING	Logical switch Logical switch ports Switching profiles switch security features

Table 21-3. Log Message IDs (continued)

Message ID	Examples
ROUTING	Logical router Logical router ports Static routing Dynamic routing NAT
FIREWALL	Firewall rules Firewall rule sections
FIREWALL-PKTLOG	Firewall connection logs Firewall packet logs
GROUPING	IP sets Mac sets NSGroups NSServices NSService groups VNI Pool IP Pool
DHCP	DHCP relay
SYSTEM	Appliance management (remote syslog, ntp, etc) Cluster management Trust management Licensing User and roles Task management Install Upgrade (NSX Manager, NSX Edge and host-packages upgrades) Realization Tags
MONITORING	SNMP Port connection Traceflow
-	All other log messages.

Customer Experience Improvement Program

NSX-T Data Center participates in VMware's Customer Experience Improvement Program (CEIP).

Details regarding the data collected through CEIP and the purposes for which it is used by VMware are set forth at the Trust & Assurance Center at https://www.vmware.com/solutions/trustvmware/ceip.html.

To join or leave the CEIP for NSX-T Data Center, or edit program settings, see Edit the Customer Experience Improvement Program Configuration.

Edit the Customer Experience Improvement Program Configuration

When you install or upgrade NSX Manager, you can decide to join the CEIP and configure data collection settings.

You can also edit the existing CEIP configuration to join or leave the CEIP program, define the frequency and the days the information is collected, and proxy server configuration.

Prerequisites

- Verify that the NSX Manager is connected and can synchronize with your hypervisor.
- Verify that NSX-T Data Center is connected to a public network for uploading data.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **System > Customer Program**.
3. Click **Edit** in the Customer Experience Improvement Program section.
4. In the Edit Customer Experience Program dialog box, select the **Join the VMware Customer Experience Improvement Program** check box.
5. Toggle the **Schedule** switch to disable or enable the data collection.

   The schedule is enabled by default.
6. (Optional) Configure the data collection and upload recurrence settings.
7. Click **Save**.

Add Tags to an Object

You can add tags to objects to make searching easier. When you specify a tag, you can also specify a scope.

NSX Cloud Note If using NSX Cloud, see How to use NSX-T Data Center Features with the Public Cloud for a list of auto-generated logical entities, supported features, and configurations required for NSX Cloud.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Edit an object.

   For example, go to the **Segments** tab and edit a segment.

3 Go the **Tags** field and add tags.

   Each tag has a tag value, which is required, and a scope value, which is optional. An object can have a maximum of 30 tags. The maximum length of a tag is 256 characters. The maximum length of a scope is 128 characters.

4 Click **Save**.

Find the SSH Fingerprint of a Remote Server

Some API requests that involve copying files to or from a remote server require that you provide the SSH fingerprint for the remote server in the request body. The SSH fingerprint is derived from a host key on the remote server.

To connect via SSH, the NSX Manager and the remote server must have a host key type in common. If there are multiple host keys types in common, whichever one is preferred according to the HostKeyAlgorithm configuration on the NSX Manager is used.

Having the fingerprint for a remote server helps you confirm you are connecting to the correct server, protecting you from man-in-the-middle attacks. You can ask the administrator of the remote server if they can provide the SSH fingerprint of the server. Or you can connect to the remote server to find the fingerprint. Connecting to the server over console is more secure than over the network.

The following table lists what NSX Manager supports in order from more preferred to less preferred.

Table 21-4. NSX Manager Host Keys in Preferred Order

Host key types supported by NSX Manager	Default Location of the Key
ECDSA (256 bit)	/etc/ssh/ssh_host_ecdsa_key.pub
ED25519	/etc/ssh/ssh_host_ed25519_key.pub

Procedure

1 Log in to the remote server as root.

   Logging in using a console is more secure than over the network.

2 List the public key files in the `/etc/ssh` directory.

```
$ ls -al /etc/ssh/*pub
-rw-r--r-- 1 root root 601 Apr  8 18:10 ssh_host_dsa_key.pub
-rw-r--r-- 1 root root  93 Apr  8 18:10 ssh_host_ed25519_key.pub
-rw-r--r-- 1 root root 393 Apr  8 18:10 ssh_host_rsa_key.pub
```

3 Compare the available keys to what NSX Manager supports.

   In this example, ED25519 is the only acceptable key.

4 Get the fingerprint of the key.

```
# awk '{print $2}' /etc/ssh/ssh_host_ed25519_key.pub | base64 -d | sha256sum -b | sed 's/ .*$//'
| xxd -r -p | base64 | sed 's/.//44g' | awk '{print "SHA256:"$1}'
SHA256:KemgftCfsd/hn7EEflhJ4m1698rRhMmNN2IW8y9iq2A
```

View Data about Applications Running on VMs

You can view information about applications running on VMs that are members of an NSGroup. This is a technical preview feature.

Procedure

1. From your browser, log in with admin privileges to an NSX Manager at https://<nsx-manager-ip-address>.
2. Select **Inventory > Groups** from the navigation panel.
3. Click the name of an NSGroup.
4. Click the **Applications** tab.
5. Click **COLLECT APPLICATION DATA**.

   This process can take a few minutes. When the process is completed, the following information is displayed:

   - The total number of processes.
   - Circles representing various tiers, for example, web tier, database tier, and application tier. Also displayed is the number of processes in each tier.
6. Click a circle to see more information about the processes in that tier.

Using NSX Cloud

22

NSX Cloud enables you to manage and secure your public cloud inventory using NSX-T Data Center.

See Installing NSX Cloud Components in the *NSX-T Data Center Installation Guide* for the NSX Cloud deployment workflow.

See also: public cloud.

This chapter includes the following topics:

The Cloud Service Manager

The Cloud Service Manager (CSM) provides a single pane of glass management endpoint for your public cloud inventory.

The CSM interface is divided into the following categories:

- **Search**: You can use the search text box to find public cloud accounts or related constructs.
- **Clouds**: Your public cloud inventory is managed through the sections under this category.
- **System**: You can access **Settings**, **Utilities**, and **Users** for Cloud Service Manager from this category.

You can perform all public cloud operations by going to the **Clouds** subsection of CSM.

To perform system-based operations, such as, backup, restore, upgrade, and user management, go to the **System** subsection.

Clouds

These are the sections under **Clouds**:

Clouds > Overview

Access your public cloud account by clicking **Clouds**.

Overview: Each tile on this screen represents your public cloud account with the number of accounts, regions, VPCs or VNets, and instances (workload VMs) it contains.

You can perform the following tasks:

Add a public cloud account or subscription	You can add one or more public cloud accounts or subscriptions. This enables you to view your public cloud inventory in CSM and indicates the number of VMs that are managed by NSX-T Data Center and their state. See **Add your Public Cloud Account** in the *NSX-T Data Center Installation Guide* for detailed instructions.
Deploy/Undeploy NSX Public Cloud Gateway	You can deploy or undeploy one or two (for High Availability) PCG(s). You can also undeploy PCG from CSM. See **Deploy PCG** or **Undeploy PCG** in the *NSX-T Data Center Installation Guide* for detailed instructions.
Enable or Disable Quarantine Policy	You can enable or disable Quarantine Policy. See Manage Quarantine Policy for details.
Switch between Grid and Card view	The cards display an overview of your inventory. The grid displays more details. Click the icons to switch between the view types.

CSM provides a holistic view of all your public cloud accounts that you have connected with NSX Cloud by presenting your public cloud inventory in different ways:

- You can view the number of regions you are operating in.
- You can view the number of private networks per region.
- You can view the number of workload VMs per private network.

There are four tabs under **Clouds**.

Clouds > {Your Public Cloud} > Accounts

The Accounts section of CSM provides information on the public cloud accounts you have already added.

Each card represents a public cloud account of the cloud provider you selected from under Clouds.

You can perform the following actions from this section:

- Add Account
- Edit Account

- Delete Account
- Resync Account

Clouds > {Your Public Cloud} > Regions

The Regions section displays your inventory for a selected region.

You can filter the Regions by your public cloud account. Each region has VPCs or VNets, and instances. If you have deployed any PCGs, you can see them here as the Gateways with an indicator for the PCG health.

Clouds > {Your Public Cloud} > VPCs or VNets

The VPCs or VNets section displays your private cloud inventory.

You can filter the inventory by Account and Region.

- Each card represents one VPC or VNet.
- You can have one or two (for HA) PCGs deployed on Transit VPCs/VNets.
- You can link Compute VPCs/VNets to Transit VPCs/VNets.
- You can view more details for each VPC or VNet by switching to the grid view.

  Note In the grid view you can see three tabs: **Overview**, **Instances**, and **Segments**.

  - **Overview** lists the options under Actions as described in the next step.
  - **Instances** displays a list of instances in the VPC/VNet.
  - **Segments** displays overlay segments in NSX-T. This feature is not supported in the current release for NSX Cloud. Do not tag your workload VMs in AWS or Microsoft Azure with tags shown on this screen.

- Click on **Actions** to access the following :
  - **Edit Configuration** (only available for Transit VPCs/VNets):
    - Enable or disable Quarantine Policy.
    - Change your proxy server selection.
  - **Link to Transit VPC/VNet**: This option is only available to VPCs/VNets that do not have any PCG deployed on them. Click to select a Transit VPC/VNet to link to.
  - **Deploy NSX Cloud Gateway**: This option is only available to VPCs/VNets that do not have a PCG deployed on them. Click this option to get started with deploying PCG on this VPC/VNet and make it a Transit or self-managed VPC/VNet. See **Deploy or Link NSX Public Cloud Gateways** in the *NSX-T Data Center Installation Guide* for detailed instructions.

Clouds > {Your Public Cloud} > Instances

The Instances section displays details of the instances in your VPC or VNet.

You can filter the instance inventory by Account, Region, and VPC or VNet.

Each card represents an instance (workload VM) and displays a summary.

For details on the instance, click on the card or switch to the grid view.

Note CSM displays the OS release value for NSX-managed VMs but for VMs not managed by NSX, the type of OS displayed is minimal in detail because it is obtained from the cloud provider APIs.

System

These are the sections under **System**:

System > Settings

These settings are first configured when you install CSM. You can edit them thereafter.

Join CSM with NSX Manager

You must connect the CSM appliance with NSX Manager to allow these components to communicate with each other.

Prerequisites

- NSX Manager must be installed and you must have the username and password for the admin account to log in to NSX Manager
- CSM must be installed and you must have the Enterprise Administrator role assigned in CSM.

Procedure

1. From a browser, log in to CSM.
2. When prompted in the setup wizard, click **Begin Setup**.
3. Enter the following details in the NSX Manager Credentials screen:

Option	Description
NSX Manager Host Name	Enter the fully qualified domain name (FQDN) of the NSX Manager, if available. You may also enter the IP address of NSX Manager.
Admin Credentials	Enter an Enterprise Administrator username and password for NSX Manager.
Manager Thumbprint	Optionally, enter the NSX Manager's thumbrpint value. If you leave this field blank, the system identifies the thumbprint and displays it in the next screen.

4. (Optional) If you did not provide a thumbprint value for NSX Manager, or if the value was incorrect, the **Verify Thumbprint** screen appears. Select the checkbox to accept the thumbprint discovered by the system.

5 Click **Connect**.

Note If you missed this setting in the setup wizard or if you want to change the associated NSX Manager, log in to CSM, click **System > Settings**, and click **Configure** on the panel titled **Associated NSX Node**.

CSM verifies the NSX Manager thumbprint and establishes connection.

6 (Optional) Set up the Proxy server. See instructions in (Optional) Configure Proxy Servers.

(Optional) Configure Proxy Servers

If you want to route and monitor all internet-bound HTTP/HTTPS traffic through a reliable HTTP Proxy, you can configure up to five proxy servers in CSM.

All public cloud communication from PCG and CSM is routed through the selected proxy server.

Proxy settings for PCG are independent of proxy settings for CSM. You can choose to have none or a different proxy server for PCG.

You can choose the following levels of authentication:

- Credentials-based authentication.
- Certificate-based authentication for HTTPS interception.
- No authentication.

Procedure

1 Click **System > Settings**. Then click **Configure** on the panel titled **Proxy Servers**.

Note You can also provide these details when using the CSM Setup Wizard that is available when you first install CSM.

2 In the Configure Proxy Servers screen, enter the following details:

Option	Description
Default	Use this radio button to indicate the default proxy server.
Profile Name	Provide a proxy server profile name. This is mandatory.
Proxy Server	Enter the proxy server's IP address. This is mandatory.
Port	Enter the proxy server's port. This is mandatory.
Authentication	Optional. If you want to set up additional authentication, select this check box and provide valid username and password.
Username	This is required if you select the Authentication checkbox.
Password	This is required if you select the Authentication checkbox.

Option	Description
Certificate	Optional. If you want to provide an authentication certificate for HTTPS interception, select this checkbox and copy-paste the certificate in the text box that appears.
No Proxy	Select this option if you do not want to use any of the proxy servers configured.

System > Utilities

The following utilities are available.

Backup and Restore

Follow the same instructions for backing up and restoring CSM, as you do for NSX Manager. See Backing Up and Restoring the NSX Manager for details.

Support Bundle

Click **Download** to retrieve the support bundle for CSM. This is used for r troubleshooting. See the *NSX-T Data Center Troubleshooting Guide* for more information.

System > Users

Users are managed using role-based access control (RBAC).

See Managing User Accounts and Role-Based Access Control for details.

Manage Quarantine Policy

Learn how to enable or disable Quarantine Policy and understand the implications thereof on your workload VMs.

NSX Cloud uses public cloud security groups for threat detection. For example, when Quarantine Policy is enabled, if NSX agent is forcibly stopped on a managed VM with malicious intent, the compromised VM is quarantined using the `quarantine` (in Microsoft Azure) or `default` (in AWS) security group.

General Recommendation:

Start with *disabled* for **Brownfield** deployments: Quarantine Policy is disabled by default. When you already have VMs set up in your public cloud environment, use the disabled mode for Quarantine Policy until you onboard your workload VMs. This ensures that your existing VMs are not automatically quarantined.

Start with *enabled* for **Greenfield** deployments: For greenfield deployments, it is recommended that you enable Quarantine Policy to allow threat detection for your VMs to be managed by NSX Cloud.

Note When Quarantine Policy is enabled, apply the `vm_override_sg` on workload VMs to be able to onboard them and then remove this security group after they are managed by NSX Cloud. Appropriate security groups are applied to the VMs within two minutes.

How to Enable or Disable Quarantine Policy

When deploying PCG on a Transit VPC/VNet or linking a Compute VPC/VNet to a Transit, you have the option to turn the Quarantine Policy on or off. Follow these steps to enable or disable the Quarantine Policy subsequently.

Prerequisites

One or a pair of PCGs must be deployed and running in your Transit VPC/VNet.

Procedure

1 Log in to CSM and go to your public cloud:

   a If using AWS, go to **Clouds > AWS > VPCs**. Click on the Transit or Compute VPC.

   b If using Microsoft Azure, go to **Clouds > Azure > VNets**. Click on the Transit or Compute VNet.

2 Enable the option using any one of the following:

   - In the tile view, click on **ACTIONS > Edit Configuration**.

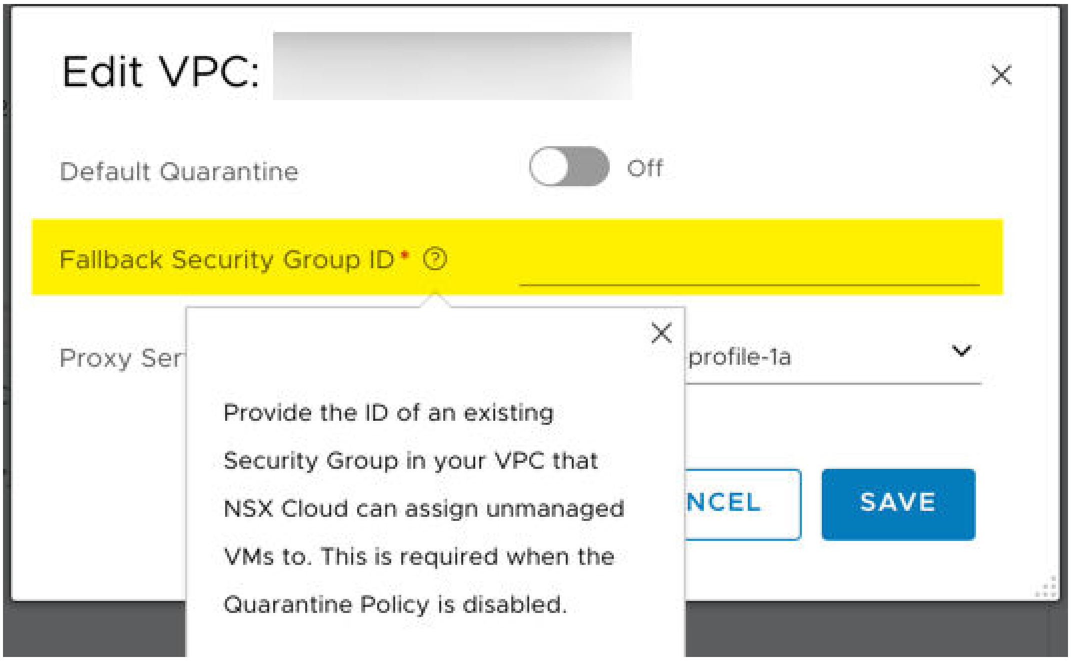
   - If you are in the grid view, select the checkbox next to the VPC or VNet and click **ACTIONS > Edit Configuration**.

   - If you are in the VPC or VNet's page, click the ACTIONS icon to go to **Edit Configurations**.

3 Turn **Default Quarantine** on or off to enable or disable it.

4 If you are disabling Quarantine Policy, you must provide a fallback security group.

Note The fallback security group must be an existing user-defined security group in your public cloud. You cannot use any of the NSX Cloud security groups as a fallback security group. See NSX Cloud Security Groups for the Public Cloud for a list of NSX Cloud security groups.



Default Quarantine: Off

Proxy Server: profile-1a

Provide the ID of an existing Security Group in your VPC that NSX Cloud can assign unmanaged VMs to. This is required when the Quarantine Policy is disabled.

  - All unmanaged or quarantined VMs in this VPC or VNet will get the fallback security group assigned to them upon disabling Quarantine Policy.
  - All managed VMs retain the security group assigned by NSX Cloud. The first time such VMs are untagged and become unmanaged after disabling Quarantine Policy, they also get the fallback security group assigned to them.

5 Click **SAVE**.

Quarantine Policy Impact when Disabled

Quarantine Policy: disabled

When Quarantine Policy is disabled:

- NSX Cloud does not assign any security groups to the VMs launched in this VPC or VNet. You must assign the appropriate NSX Cloud security groups to VMs to enable threat detection.

  From the Microsoft Azure portal or AWS console:

- - Assign `vm-underlay-sg` to VMs for which you want to use the underlay network provided by Microsoft Azure or AWS.
  - Ensure the following ports are open:
    - Inbound UDP 6081 : For overlay data packets. This should be allowed for (Active/Standby) PCG's VTEP IP address (eth1 interface).
    - Outbound TCP 5555 : For control packets. This should be allowed for (Active/Standby) PCG's management IP address (eth0 interface).
    - TCP 8080 : For install/upgrade on the PCG's management IP address.

  - TCP 80: For downloading any third party dependencies while installing the NSX agent.
  - UDP 67,68: For DHCP packets.
  - UDP 53: For DNS resolution.

Quarantine Policy: was enabled, then disabled

The following table captures the impact on security group assignments if the Quarantine Policy was enabled and then you disable it:

Table 22-1. Security Group Impact of Disabling Quarantine Policy

VM-ID	Managed?	Security Group	Security Group for VM after Quarantine Policy is Disabled
VM1	Yes	`vm_underlay_sg`	`vm_underlay_sg` . When you remove the `nsx.network` tag from this VM, to take it out from NSX management, this VM also gets the fallback security group assigned to it.
VM2	Yes	`default` (AWS) or `quarantine` (Microsoft Azure)	The fallback security group you specify when disabling Quarantine Policy. See How to Enable or Disable Quarantine Policy for details.
VM3	No	`vm_override_sg`	The fallback security group you specify when disabling Quarantine Policy.
VM4	No	`default` (AWS) or `quarantine` (Microsoft Azure)	The fallback security group you specify when disabling Quarantine Policy.

Note Disabling Quarantine Policy is required for undeploying PCG. See **Undeploying PCG** in the *NSX-T Data Center Installation Guide* for details.

Quarantine Policy Impact when Enabled

Quarantine Policy: enabled

When Quarantine Policy is enabled:

- The Security Group (SG) or Network Security Group (NSG) assignment for all interfaces for any workload VMs belonging to this VPC or VNet is managed by NSX Cloud as under:
  - Unmanaged VMs are assigned the `quarantine` NSG in Microsoft Azure and `default` Security Group in AWS and are quarantined. This limits the outbound traffic and stops all inbound traffic to such VMs.

- Unmanaged VMs can become NSX-Managed VMs when you install the NSX agent on the VM and tag them in the public cloud with `nsx.network`. In the default scenario, NSX Cloud assigns the `vm-underlay-sg` to allow appropriate inbound/outbound traffic.
- An NSX-Managed VM can still be assigned the `quarantine` or `default` security group and be quarantined if a threat is detected on the VM, for example, if the NSX agent is stopped on the VM.
- Any manual changes to the security groups will be reverted to the NSX-determined security group(s) within two minutes.
- If you want to move any VM out of quarantine, assign the `vm-override-sg` as the only security group for this VM. NSX Cloud does not auto-change the `vm-override-sg` security group and allows SSH and RDP access to the VM. Removing the `vm-override-sg` will again cause the VM security group(s) to revert to the NSX-determined security group.

Note When the Quarantine Policy is enabled, assign the `vm-override-sg` to your VMs before installing the NSX agent on them. After you follow the process of installing the NSX agent and tagging the VM as underlay, remove the `vm-override-sg` NSG from the VM. NSX Cloud wil automatically assign the appropriate security group to NSX-managed VMs thereafter. This step is necessary because it ensures the VM is not assigned the `quarantine` or `default` security group while you are preparing it for NSX Cloud.

Quarantine Policy: was disabled, then enabled

The following table captures the impact on security group assignments if the Quarantine Policy was disabled and then you enable it:

Table 22-2. Security Group Impact of Enabling Quarantine Policy

VM-ID	Managed?	Threat detected?	Security Group after enabling Quarantine Policy
VM1	Yes	No	`vm_underlay_sg` .

VM2	Yes	Yes	`default`
			(AWS) or
			`quarantine`
			(Microsoft Azure)
			Note You may manually assign `vm_override_sg` to managed VMs. This brings them out of quarantine mode and you can repair the problem by accessing such VMs through SSH or RDP. See Quarantine Policy: enabled

VM3	No	N/A	`default`
			(AWS) or
			`quarantine`
			(Microsoft Azure)

NSX Cloud Security Groups for the Public Cloud

The following security groups are created by NSX Cloud at the time of PCG deployment:

The **gw** security groups are applied to the respective PCG interfaces.

Table 22-3. Public Cloud Security Groups created by NSX Cloud for PCG interfaces

Security Group name	Available in Microsoft Azure?	Available in AWS?	Full Name
gw-mgmt-sg	Yes	Yes	Gateway Management Security Group
gw-uplink-sg	Yes	Yes	Gateway Uplink Security Group
gw-vtep-sg	Yes	Yes	Gateway Downlink Security Group

Table 22-4. Public Cloud Security Groups created by NSX Cloud for Workload VMs

Security Group name	Available in Microsoft Azure?	Available in AWS?	Descriptiom
quarantine	Yes	No	Quarantine security group for Microsoft Azure
default	No	Yes	Quarantine security group for AWS
vm-underlay-sg	Yes	Yes	VM Non-Overlay security group
vm-override-sg	Yes	Yes	VM Override Security Group
vm-overlay-sg	Yes	Yes	VM Overlay security group (this is not used in the current release)
vm-outbound-bypass-sg	Yes	Yes	VM Outbound Bypass Security Group (this is not used in the current release)
vm-inbound-bypass-sg	Yes	Yes	VM Inbound Bypass Security Group (this is not used in the current release)

Overview of Onboarding and Managing Workload VMs

Refer to the checklist for an overview of the steps involved in onboarding and managing workload VMs.

See Overview of Installing and Configuring NSX Cloud Components for your Public Cloud in the *NSX-T Data Center Installation Guide* for the Day-0 workflow.

How to Onboard and Manage Workload VMs

Refer to this workflow for an overview of the steps involved in onboarding and managing workload VMs from your public cloud.

Table 22-5. Day-N Workflow for onboarding your workload VMs into NSX Cloud

Task	Persona	Instructions
☐If Quarantine Policy is enabled, place the VMs in the **`vm_underlay_sg`** security group. If Quarantine Policy is disabled, place the VMs in the **`vm_override_sg`** security group.	Public Cloud Admin	Follow instructions in your public cloud documentation for placing workload VMs in specific security groups.
☐ Tag workload VMs with the key-value **`nsx.network=default`**.	Public Cloud Admin	Follow instructions in your public cloud documentation for tagging workload VMs.
☐ Install the NSX Agent on your Windows and Linux workload VMs. **Note** If **Auto Agent Installation** is turned on in CSM for Microsoft Azure accounts, the NSX Agent is automatically installed.	Public Cloud Admin	See Install NSX Agent.
☐ If Quarantine Policy is enabled, place the VMs in the **`default`** security group.	Public Cloud Admin	Follow instructions in your public cloud documentation for placing workload VMs in specific security groups.
☐ To allow inbound access to workload VMs, create distributed firewall (DFW) rules as required.	NSX-T Data Center Enterprise Administrator	See DFW Rules for NSX-Managed Workload VMs.
☐ Group your workload VMs using public cloud tags or NSX-T Data Center tags and set up micro-segmentation.	NSX-T Data Center Enterprise Administrator	See Group VMs using NSX-T Data Center and Public Cloud Tags .

Onboard Workload VMs

Onboarding your workload VMs to start managing them using NSX-T Data Center.

Supported Operating Systems

This is the list of operating systems currently supported by NSX Cloud for your workload VM.

Currently, the following operating systems are supported:

Note See the NSX Cloud Known Issues section in the *NSX-T Data Center Release Notes* for exceptions.

- Red Hat Enterprise Linux (RHEL) 7.2, 7.3, 7.4, 7.5
- CentOS 7.2, 7.3, 7.4, 7.5

- Ubuntu 14.04, 16.04
- Microsoft Windows Server 2012 R2
- Microsoft Windows Sever 2016
- Microsoft Windows 10

Tag VMs in the Public Cloud

Apply the **nsx.network** tag to VMs that you want to manage using NSX-T Data Center.

Prerequisites

The VPC or VNet where the workload VMs are hosted, must be onboarded with NSX Cloud. See **Adding your Public Cloud Inventory** in the *NSX-T Data Center Installation Guide* for details.

Procedure

1. Log in to your public cloud account and go to your VPC or VNet that has been onboarded with NSX Cloud.
2. Select the VMs that you want to manage using NSX-T Data Center.
3. Add the following tag details for the VMs and save your changes.

   ```
   Name: nsx.network
   Value: default
   ```

   Note You can apply this tag either at the VM level or the interface level with the same effect.

Example

What to do next

Install the NSX agent on these VMs. See Install NSX Agent.

If using Microsoft Azure, you have the option to auto-install the NSX agent on tagged VMs. See Install the NSX Agent Automatically for details.

Install NSX Agent

Install the NSX Agent on your workload VMs

See Generate Replicable Images for instructions on creating AMIs or Managed Images with the NSX Agent installed.

Install NSX Agent on Windows VMs

Follow these instructions to install the NSX agent on your Windows workload VM.

See Supported Operating Systems for a list of Microsoft Windows versions currently supported.

Note To verify the checksum of this script, go to **VMware Downloads > Drivers & Tools > NSX Cloud Scripts**.

Procedure

1 Log in to CSM and go to your public cloud:

   a If using AWS, go to **Clouds > AWS > VPCs**. Click on a Transit or Compute VPC.

   b If using Microsoft Azure, go to **Clouds > Azure > VNets**. Click on the VNet on which one or a pair of PCGs is deployed and running.

   Note: Transit VPC/VNet is where one or a pair of PCGs is deployed and running. Compute VPC/VNet is the one linked to a Transit and can use the PCGs deployed there.

2 From the **Agent Download & Installation** section of the screen, make a note of the **Download Location** and the **Installation Command** under **Windows**.

   Note For VNets, the DNS Suffix in the Installation Command is dynamically generated to match the DNS settings you choose when deploying PCG. For Transit VNets, the `-dnsServer <dns-server-ip>` parameter is optional. For Compute VNets, you must provide the DNS Forwarder IP address to complete this command.

3 Connect to your Windows workload VM as Administrator.

4 Download the installation script on your Windows VM from the **Download Location** you noted from CSM. You can use any browser, for example, Internet Explorer, to download the script. It is downloaded in your browser's default downloads directory, for example, *C:\Downloads*.

   Note To verify the checksum of this script, go to **VMware Downloads > Drivers & Tools > NSX Cloud Scripts**

   Note:

5 Open a PowerShell prompt and go to the directory containing the downloaded script.

6 Use the **Installation command** you noted from CSM to run the downloaded script.

   For example:

```
c:\> powershell -file ‘nsx_install.ps1” -operation install -dnsSuffix <>
```

   Note The file argument needs the full path unless you are in the same directory or if the PowerShell script is already in the path. For example, if you download the script to *C:\Downloads*, and you are currently not in that directory, then the script must contain the location: *powershell -file 'C:\Downloads\nsx_install.ps1' ...*

7 The script runs and when completed, displays a message indicating whether the NSX agent was installed successfully.

Note The script considers the primary network interface as the default.

What to do next

Manage Workload VMs

Install NSX Agent on Linux VMs

Follow these instructions to install the NSX agent on your Linux workload VMs.

See Supported Operating Systems for a list of Linux distributions currently supported.

Note To verify the checksum of this script, go to **VMware Downloads > Drivers & Tools > NSX Cloud Scripts**.

Prerequisites

You need the following commands to run the NSX agent installation script:

- **wget**
- **nslookup**
- **dmidecode**

Procedure

1. Log in to CSM and go to your public cloud:
   a. If using AWS, go to **Clouds > AWS > VPCs**. Click on a Transit or Compute VPC.
   b. If using Microsoft Azure, go to **Clouds > Azure > VNets**. Click on the VNet on which one or a pair of PCGs is deployed and running.

   Note: Transit VPC/VNet is where one or a pair of PCGs is deployed and running. Compute VPC/VNet is the one linked to a Transit and can use the PCGs deployed there.
2. From the **Agent Download & Installation** section of the screen, make a note of the **Download Location** and the **Installation Command** under **Linux**.

   Note For VNets, the DNS Suffix in the Installation Command is dynamically generated to match the DNS settings you choose when deploying PCG. For Transit VNets, the `-dnsServer <dns-server-ip>` parameter is optional. For Compute VNets, you must provide the DNS Forwarder IP address to complete this command.
3. Log in to the Linux workload VM with superuser privileges.

4 Use `wget` or equivalent to download the installation script on your Linux VM from the **Download Location** you noted from CSM. The installation script is downloaded in the directory where you run the `wget` command.

  Note To verify the checksum of this script, go to **VMware Downloads > Drivers & Tools > NSX Cloud Scripts**.

5 Change permissions on the installation script to make it executable if required, and run it:

```
$ chmod +x install_nsx_vm_agent.sh && sudo ./install_nsx_vm_agent.sh
```

  Note: On Red Hat Enterprise Linux and its derivatives, SELinux is not supported. Disable SELinux to install the NSX agent.

6 You lose connection to your Linux VM after NSX agent installation begins. Messages such as the following appear on your screen: `Installation completed!!! Starting NSX Agent service. SSH connection will now be lost.`. Reconnect to your VM to complete the onboarding process.

Results

The NSX agent is installed on your workload VM(s).

Note

- After the NSX agent is successfully installed, port 8888 shows as open on the VM but it is blocked for VMs in the underlay mode and should be used only when required for advanced troubleshooting.
- The script uses `eth0` as the default interface.

What to do next

Manage Workload VMs

Uninstalling the NSX Agent

Use these OS-specific commands to uninstall the NSX Agent.

Uninstalling NSX agent from a Windows VM

Note To see other options available for the installation script, use `-help`.

1 Remote log in to the VM using RDP.

2 Run the installation script with the uninstall option:

```
\nsx_install.ps1  -operation uninstall
```

Uninstalling NSX agent from a Linux VM

Note To see other options available for the installation script, use `--help`.

1 Remote log in to the VM using SSH.

2 Run the installation script with the uninstall option:

```
sudo ./install_nsx_vm_agent.sh --uninstall
```

Install the NSX Agent Automatically

Currently only supported for Microsoft Azure.

In Microsoft Azure, if the following criteria are met, the NSX agent is installed automatically:

- Azure VM Extensions installed on the VMs in the VNet added into NSX Cloud. See Microsoft Azure documentation on VM Extensions for more details.
- The security group applied to VMs in Microsoft Azure must allow for NSX agent installation. If Quarantine Policy is enabled, apply the `vm-overrride-sg` to the workload VMs. If Quarantine Policy is disabled, apply the `vm_underlay_sg` to them.
- VMs tagged using the `nsx.network` and value `default`.

To enable this feature:

1 Go to **Clouds > Azure > VNets**.

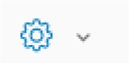
2 Select the VNet on whose VMs you want to auto-install the NSX agent.

3 Enable the option using any one of the following:

  - In the tile view, click on **ACTIONS > Edit Configuration**.

  - If you are in the grid view, select the checkbox next to the VNet and click **ACTIONS > Edit Configuration**.

  - If you are in the VNet's page, click the ACTIONS icon to go to **Edit Configurations**.



4 Move the slider next to **Auto Agent Installation** to the ON position.

Note If the NSX agent's installation fails, do the following:

1 Log in to the Microsoft Azure portal and navigate to the VM where the NSX agent installation failed.

2 Go to the VM's Extensions and uninstall the extension named `VMwareNsxAgentInstallCustomScriptExtension`.

3 Remove the `nsx.network` tag from this VM.

4 Add the `nsx.network` tag on this VM again.

Within about three minutes, the NSX agent gets installed on this VM.

Manage Workload VMs

After you have successfully onboarded workload VMs, you can use NSX-T Data Center to manage them.

DFW Rules for NSX-Managed Workload VMs

When you deploy the PCG on your Transit VPC/VNet or when you link a Compute VPC/VNet to a Transit, NSX Cloud creates default DFW rules for NSX-managed workload VMs that block all inbound connectivity to them.

The two stateless rules are for DHCP access and they do not affect access to your workload VMs.

The two stateful rules are as follows:

DFW Rules created by NSX Cloud under Policy: `cloud-stateful-cloud-<VPC/VNet ID>`	**Properties**
`cloud-<VPC/VNet ID>-managed`	Allows access to the VMs within the same VPC/VNet.
`cloud-<VPC/VNet ID>-inbound`	Blocks access to NSX-managed VMs from anywhere outside the VPC/VNet.

Note Do not edit any of the default rules.

You can create a copy of the existing inbound rule, adjust the sources and destinations, and set to **Allow**. Place the **Allow** rule above the default **Reject** rule. You can also add new policies and rules. See Add a Distributed Firewall for instructions.

Group VMs using NSX-T Data Center and Public Cloud Tags

NSX Cloud allows you to use the public cloud tags assigned to your workload VMs.

NSX Manager uses tags to group VMs, as do public clouds. Therefore, to facilitate grouping VMs, NSX Cloud pulls in the public cloud tags applied to your workload VMs provided they meet predefined size and reserved-words criteria, into NSX Manager.

Note DFW rules depend on the tags assigned to VMs. Since these tags can be modified by anyone with the appropriate public cloud permissions, NSX-T Data Center assumes that such users are trustworthy and the responsibility of ensuring and auditing that VMs are correctly tagged at all times lies with the public cloud network administrator.

Tags terminology

A **tag** in NSX Manager refers to what is known as **value** in a public cloud context. The **key** of a public cloud tag, is referred to as **scope** in NSX Manager.

Components of tags in NSX Manager	Equivalent components of tags in the public cloud
Scope	Key
Tag	Value

Tag Types and Limitations

NSX Cloud allows three types of tags for NSX-managed public cloud VMs.

- **System Tags**: These tags are system-defined and you cannot add, edit, or delete them. NSX Cloud uses the following system tags:
  - azure:subscription_id
  - azure:region
  - azure:vm_rg
  - azure:vnet_name
  - azure:vnet_rg
  - azure:transit_vnet_name
  - azure:transit_vnet_rg
  - aws:account
  - aws:availabilityzone
  - aws:region
  - aws:vpc
  - aws:subnet
  - aws:transit_vpc

- **Discovered Tags**: Tags that you have added to your VMs in the public cloud are automatically discovered by NSX Cloud and displayed for your workload VMs in NSX Manager inventory. These tags are not editable from within NSX Manager. There is no limit to the number of discovered tags. These tags are prefixed with `dis:azure:` to denote they are discovered from Microsoft Azure and `dis:aws` from AWS.

  When you make any changes to the tags in the public cloud, the changes are reflected in NSX Manager within three minutes.

  By default this feature is enabled. You can enable or disable the discovery of Microsoft Azure or AWS tags at the time of adding the Microsoft Azure subscription or AWS account.
- **User Tags**: You can create up to 25 user tags. You have add, edit, delete privileges for user tags. For information on managing user tags, see Manage Tags for a VM.

Table 22-6. Summary of Tag Types and Limitations

Tag type	Tag scope or predetermined prefix	Limitations	Enterprise Administrator Privileges	Auditor Privileges
System-defined	Complete system tags: ■ azure:subscription_id ■ azure:region ■ azure:vm_rg ■ azure:vnet_name ■ azure:vnet_rg ■ aws:vpc ■ aws:availability zone	Scope (key): 20 characters Tag (value): 65 characters Maximum possible: 5	Read only	Read only
Discovered	Prefix for Microsoft Azure tags that are imported from your VNet: **dis:azure:** Prefix for AWS tags that are imported from your VPC: **dis:aws:**	Scope (key): 20 characters Tag (value): 65 characters Maximum allowed: unlimited **Note** The limits on characters excludes the prefix **dis:<public cloud name>**. Tags that exceed these limits are not reflected in NSX Manager. Tags with the prefix **nsx** are ignored.	Read only	Read only
User	User tags can have any scope (key) and value within the allowed number of characters, except: ■ the scope (key) prefix **dis:azure:** or **dis:aws:** ■ the same scope (key) as system tags	Scope (key): 30 characters Tag (value): 65 characters Maximum allowed: 25	Add/Edit/Delete	Read only

Examples of Discovered Tags

Note Tags are in the format `key=value` for the public cloud and `scope=tag` in NSX Manager.

Table 22-7.

Public Cloud tag for the workload VM	Discovered by NSX Cloud?	Equivalent NSX Manager tag for the workload VM
Name=Developer	Yes	dis:azure:Name=Developer
ValidDisTagKeyLength=ValidDisTagValue	Yes	dis:azure:ValidDisTagKeyLength=ValidDisTagValue
Abcdefghijklmnopqrstuvwxyz=value2	No (key exceeds 20 chars)	none
tag3=AbcdefghijklmnopqrstuvwxyzAb23690hgjgjuytreswqacvbcdefghijklmnopqrstuvwxyz	No (value exceeds 65 characters)	none
nsx.name=Tester	No (key has the prefix **nsx**)	none

How to use Tags in NSX Manager

- See Manage Tags for a VM.
- See Search for Objects.
- See Set up Micro-segmentation for Workload VMs.

Set up Micro-segmentation for Workload VMs

You can set up micro-segmentation for managed workload VMs.

Do the following to apply distributed firewall rules to NSX-managed workload VMs:

1 Create groups using VM names or tags or other membership criteria, for example, for **web**, **app**, **DB** tiers. For instructions, see Add a Group.

   Note You can use any of the following tags for membership criteria. See Group VMs using NSX-T Data Center and Public Cloud Tags for details.

   - system-defined tags
   - tags from your VPC or VNet that are discovered by NSX Cloud
   - or your own custom tags

   Note DFW rules depend on the tags assigned to VMs. Since these tags can be modified by anyone with the appropriate public cloud permissions, NSX-T Data Center assumes that such users are trustworthy and the responsibility of ensuring and auditing that VMs are correctly tagged at all times lies with the public cloud network administrator.

2 Create an East-West distributed firewall policy and rule and apply to the group you created. See Add a Distributed Firewall .

This micro-segmentation takes effect when the inventory is either manually re-synchronized from CSM, or within about three minutes when the changes are pulled into CSM from your public cloud.

How to use NSX-T Data Center Features with the Public Cloud

NSX Cloud creates a network topology for your public cloud and you must not edit or delete the auto-generated NSX-T Data Center logical entities.

Use this list as a quick reference for what is auto-generated and how you should use NSX-T Data Center features as they apply to the public cloud.

NSX Manager Configurations

See **Auto-created NSX-T Logical Entities** in the *NSX-T Data Center Installation Guide* for details on the logical entities created after a PCG is successfully deployed.

Important Do not edit or delete any of these auto-created entities.

Note If you are not able to access some features on Windows workload VMs ensure that the Windows firewall settings are correctly configured.

Logical Segments FAQs

Table 22-8.

Question	Answer
Where can I find details on logical segments?	See Chapter 4 Segments
Where can I find detailed information on logical switches?	See Chapter 13 Logical Switches.

Logical Routers FAQs

Table 22-9.

Question	Answer
Does NSX Cloud auto-create a logical router when a PCG is deployed?	Yes. When PCG is deployed on a Transit VPC or VNet, a tier-0 logical router is auto-created by NSX Cloud. A tier-1 router is created for each Compute VPC/VNet when it's linked to a Transit VPC/VNet.
Where can I find more information on logical routers?	See Chapter 2 Tier-0 Gateways and Chapter 3 Tier-1 Gateway.

IPFIX FAQs

Table 22-10.

Question	Answer
Are any specific configurations required for IPFIX to work in the public cloud?	Yes: - IPFIX is supported in NSX Cloud only on UDP port 4739. - **Switch and DFW IPFIX**: If the collector is in the same subnet as the Windows VM on which IPFIX profile has been applied, a static ARP entry for the collector on the Windows VM is needed because Windows silently discards UDP packets when no ARP entry is found.
Where can I find more information on IPFIX?	See Configure IPFIX.

Port Mirroring FAQs

Table 22-11.

Question	Answer
Are any specific configurations required for Port Mirroring in the public cloud?	Port Mirroring is supported only in AWS in the current release. - For NSX Cloud, configure Port Mirroring from **Tools > Port Mirroring Session**. - Only L3SPAN Port Mirroring is supported. - The collector must be in the same VPC as the source workload VM.
Where can I find more information on Port Mirroring?	See Monitor Port Mirroring Sessions .

Other FAQs

Table 22-12.

Question	Answer
Are the tags that I apply to my workload VMs in the public cloud available in NSX-T Data Center?	Yes. See Group VMs using NSX-T Data Center and Public Cloud Tags for details.
How do I set up micro-segmentation for my workload VMs that are managed by NSX-T Data Center?	See Set up Micro-segmentation for Workload VMs.

Using Advanced NSX Cloud Features

Verify NSX Cloud Components

It is a best practice to verify that all components are up and running, before deploying in a production environment.

Verify whether NSX Agent is connected to PCG

To verify that the NSX Agent on your workload VM is connected to PCG, do the following:

1 Type the `nsxcli` command to open NSX-T Data Center CLI.

2 Type the following command to get the gateway connection status, for example:

```
get gateway connection status
Public Cloud Gateway  : nsx–gw.vmware.com:5555
Connection Status     : ESTABLISHED
```

Verify VM Interface Tag in AWS or Microsoft Azure

The workload VMs must have the correct tags to connect to PCG.

1 Log in to the AWS console or the Microsoft Azure portal.

2 Verify the VM's eth0 or interface tag.

The `nsx.network` key must have the value `default`.

Enable NAT on NSX-managed VMs

NSX Cloud supports enabling NAT on NSX-managed VMs.

You can enable North-South traffic on VMs in NSX-managed VMs using public cloud tags.

On the NSX-managed VM for which you want to enable NAT, apply the following tag:

Table 22-13.

Key	Value
`nsx.publicip`	`public IP address from your public cloud`, for example, 50.1.2.3

Note The public IP address you provide here must be free to use and must not be assigned to any VM, even the workload VM you want to enable NAT for. If you assign a public IP address that was previously associated with any other instance or private IP address, NAT does not work. In that case, unassign the public IP address.

After this tag is applied, the workload VM can access internet traffic.

Generate Replicable Images

You can generate an AMI in AWS or a Managed Image in Microsoft Azure of a VM with the NSX agent installed on it.

With this feature, you can launch multiple VMs with with the agent configured and running.

There are two ways in which you can generate an AMI/Managed Image (image in the rest of this topic) of a VM with the NSX agent installed on it:

- **Generate image with an unconfigured NSX agent**: You can generate an image from a VM that has the NSX agent installed on it but not configured by using the `–noStart` option. This option allows the NSX agent package to be fetched and installed but the NSX services are not started. Also, no NSX configurations such as certificate generation, are made.
- **Generate image after removing existing NSX agent configurations**: You can remove configurations from an existing NSX-managed VM and use it for generating an image.

Generating AMI with an unconfigured NSX agent

You can generate an AMI of a VM with the NSX agent installed on it and not configured.

To generate an image from a VM that has the NSX agent installed on it using the **`–noStart`** option, do the following:

Procedure

1. Copy paste the NSX agent Installation Command from CSM. See instructions at Install NSX Agent
   a. Edit the command for Windows as follows:

      ```
      c:\> powershell –file ‘nsx_install.ps1” –operation install –dnsSuffix <> -noStart true
      ```

   b. Edit the command for Linux as follows:

      ```
      $ chmod +x install_nsx_vm_agent.sh && sudo ./install_nsx_vm_agent.sh ––no–start
      ```

2. Go to this VM in your public cloud and create an image.

Generating an Image After Removing Existing NSX Agent Configurations

You can generate an image of a VM that has a configured NSX agent.

To remove configurations from an existing NSX-managed VM and use it for generating images, do the following:

Procedure

1 Removing NSX agent configurations from a Windows or Linux VM:

   a Log in to the workload VM using preferably using a jumphost.

   b Open the NSX-T CLI:

   ```
   sudo nsxcli
   ```

   c Enter the following commands:

   ```
   hostname> set debug
   hostname> clear nsx-vm-agent state
   ```

2 Locate this VM in your public cloud and create an image.

Service Insertion for your Public Cloud

NSX Cloud supports the use of third-party services in your public cloud for NSX-managed workload VMs.

To utilize service insertion for your public cloud workload VMs, you must host the service appliance in the public cloud, not in NSX-T Data Center. It is recommended to host the service appliance in a Transit VPC/VNet.

Before you can enable service insertion, you must have the PCG deployed in a Transit VPC or VNet.

Here is an overview of the one-time configurations to allow service insertion for your NSX-managed workload VMs.

Table 22-14. Overview of configurations required for service insertion for NSX-managed workload VMs in the public cloud

How often?	Task	Instructions
Once for the initial setup	Set up the service appliance in your public cloud preferably in a Transit VPC or VNet (where you have deployed the PCG.	See instructions specific to the third-party service appliance and the public cloud.
	Register the third-party service in NSX-T Data Center.	See Create the Service Definition and a Corresponding Virtual Endpoint
	Create a virtual instance endpoint of the service using a /32 Virtual Service IP address (VSIP) to be used only for service insertion by the service appliance. The VSIP should not conflict with the CIDR range of VPCs or VNets. This VSIP is advertised over BGP to the PCG.	See Create the Service Definition and a Corresponding Virtual Endpoint
	Create an IPSec VPN tunnel between the service appliance and the PCG.	See Set up an IPSec VPN Session

Table 22-14. Overview of configurations required for service insertion for NSX-managed workload VMs in the public cloud (continued)

How often?	Task	Instructions
	Configure BGP between the PCG and the service appliance. **Note** Configure the service appliance to advertise the VSIP and the PCG to advertise the default route (0.0.0.0/0).	See Configure BGP and Route Redistribution
As and when required	After the one-time configurations are complete, set up redirection rules to reroute selective traffic from NSX-managed workload VMs to the VSIP. These rules are applied to the uplink port of the PCG.	See Set up Redirection Rules.

Procedure

1 Create the Service Definition and a Corresponding Virtual Endpoint

  You must use NSX Manager APIs to create a service definition and virtual endpoint for the service appliance in your public cloud.

2 Set up an IPSec VPN Session

  Set up an IPSec VPN session between the PCG and your service appliance.

3 Configure BGP and Route Redistribution

  Configure BGP between the PCG and the service appliance over the IPSec VPN tunnel.

4 Set up Redirection Rules

  Redirection rules can be adjusted according to your requirements.

Create the Service Definition and a Corresponding Virtual Endpoint

You must use NSX Manager APIs to create a service definition and virtual endpoint for the service appliance in your public cloud.

Prerequisites

Pick out a /32 reserved IP address to serve as the Virtual Endpoint for the service appliance in your public cloud, for example, `100.100.100.100/32`. This is referred to as the Virtual Service IP (VSIP).

Note If you deployed your service appliance in a High Availability pair, do not create another service definition but use the same VSIP when advertising it to the PCG during BGP configuration.

Procedure

1 To create a Service Definition for the service appliance, run the following API call using NSX Manager credentials for authorization:

```
POST https://{{NSX Manager-IP}}/policy/api/v1/enforcement-points/default/service-definitions
```

Example request:

```
{
     "resource_type":"ServiceDefinition",
     "description":"NS-Service",
     "display_name":"Service_Appliance1",
     "attachment_point":[
        "TIER0_LR"
     ],
     "transports":[
        "L3_ROUTED"
     ],
     "functionalities":[
        "NG_FW", "BYOD"
     ],
     "on_failure_policy":"ALLOW",
     "implementations":[
        "NORTH_SOUTH"
     ],
     "vendor_id" : "Vendor1"
}
```

Example response:

```
{
    "resource_type": "ServiceDefinition",
    "description": "NS-Service",
    "id": "33890153-6eea-4c9d-8e34-7b6532b9d65c",
    "display_name": "Service_Appliance1",
    "attachment_point": [
        "TIER0_LR"
    ],
    "transports": [
        "L3_ROUTED"
    ],
    "functionalities": [
        "NG_FW", "BYOD"
    ],
    "vendor_id": "Vendor1",
    "on_failure_policy": "ALLOW",
    "implementations": [
        "NORTH_SOUTH"
    ],
    "_create_time": 1540424262137,
    "_last_modified_user": "nsx_policy",
    "_system_owned": false,
    "_protection": "REQUIRE_OVERRIDE",
    "_last_modified_time": 1540424262137,
    "_create_user": "nsx_policy",
    "_revision": 0
}
```

2 To create a Virtual Endpoint for the service appliance, run the following API call using NSX Manager credentials for authorization:

```
PATCH https://{{NSX Manager-IP}}policy/api/v1/infra/tier-0s/<tier-0 router ID>/locale-services/
cloud/endpoints/virtual-endpoints/Service_Appliance1_Endpoint
```

Example request:

```
{
  "resource_type": "VirtualEndpoint",
  "display_name": "Service_Appliance1_Endpoint",
  "target_ips": [
    {
      "ip_addresses": [
        "100.100.100.100"
      ],
      "prefix_length": 32
    }
  ],
  "service_names": [
    "Service_Appliance1"
  ]
}
```

Example response:

```
200 OK
```

Note The `display_name` in step 1 must match the `service_names` in step 2.

What to do next

Set up an IPSec VPN Session

Set up an IPSec VPN Session

Set up an IPSec VPN session between the PCG and your service appliance.

Prerequisites

- One or an HA pair of PCGs must be deployed in a Transit VPC/VNet.
- The service appliance must be set up in your public cloud, preferably in the Transit VPC/VNet.

Procedure

1 Navigate to **Networking > VPN**

2 Add a **VPN service** of type IPSec and note the following configuration options specific to NSX Cloud. See Add an IPSec VPN Service for other details.

Option	Description
Name	The name of this VPN service is used to set up the local endpoint and the IPSec VPN sessions. Make a note of it.
Service Type	Confirm that this value is set to IPSec.
Tier-0 Gateway	Select the tier-0 gateway auto-created for your Transit VPC/VNet. Its name contains your VPC/VNet ID, for example, `cloud-t0-vpc-6bcd2c13`.

3 Add a **Local Endpoint** for your PCG. The IP address of the local endpoint is the value of the tag `nsx:local_endpoint_ip` for the PCG deployed in your Transit VPC/VNet. Log in to your Transit VPC/VNet for this value. Note the following configurations specific to NSX Cloud and see Add Local Endpoints for other details.

Option	Description
Name	The local endpoint name is used to set up the IPSec VPN sessions. Make a note of it.
VPN Service	Select the VPN Service you added in step 2.
IP Address	Find this value by logging in to the AWS console or the Microsoft Azure portal. It is the value of the tag `nsx:local_endpoint_ip` applied to the uplink interface of the PCG.

4 Create a **Route-Based IPSec session** between the PCG and the service appliance in your public cloud (preferably hosted in the Transit VPC/VNet).

Option	Description
Type	Confirm that this value is set to **Route Based**.
VPN Service	Select the VPN Service you added in step 2.
Local Endpoint	Select the local endpoint you created in step 3.
Remote IP	Enter the private IP address of the service appliance. **Note** If your service appliance is accessible using a public IP address, assign a public IP address to the local endpoint IP (also known as secondary IP) to the PCG's uplink interface.
Tunnel Interface	This subnet must match with the service appliance subnet for the VPN tunnel. Enter the subnet value you set up in the service appliance for the VPN tunnel or note the value you enter here and make sure the same subnet is used when setting up the VPN tunnel in the service appliance. **Note** You configure BGP on this tunnel interface. See Configure BGP and Route Redistribution .

Option	Description
Remote ID	Enter the private IP address of your service appliance in the public cloud.
IKE Profile	The IPSec VPN session must be associated with an IKE profile. If you created a profile, select it from the drop-down menu. You can also use the default profile.

What to do next

Configure BGP and Route Redistribution

Configure BGP and Route Redistribution

Configure BGP between the PCG and the service appliance over the IPSec VPN tunnel.

You set up BGP neighbors on the IPSec VPN tunnel interface that you established between PCG and the service appliance. See Configure BGP for more details.

You need to configure BGP similarly on your service appliance. See documentation for your specific service in the public cloud for details.

Next, set up route redistribution as follows:

- The PCG advertises its default route (0.0.0.0/0) to the service appliance.
- The service appliance advertises the VSIP to the PCG. This is the same IP address which is used when registering the service. See Create the Service Definition and a Corresponding Virtual Endpoint.

  Note If your service appliance is deployed in a High Availability pair, advertise the same VSIP from both service appliances.

Prerequisites

Procedure

1 Navigate to **Networking > Tier-0 Gateways**

2 Select the auto-created tier-0 gateway for your Transit VPC/VNet named like `cloud-t0-vpc-6bcd2c13` and click **Edit**.

3 Click the number or icon next to **BGP Neighbors** under the **BGP** section.

4 Note these configurations:

Option	Description
IP Address	Use the IP address configured on the service appliance tunnel interface for the VPN between the PCG and the service appliance.
Remote AS Number	This number must match the AS number of the service appliance in your public cloud.

5 (Required) From the **Static Route** section, set up a static route to the PCG's default route (0.0.0.0/0).

6 From the **Route Redistribution** section, select the static route associated with the default route.

What to do next

Set up Redirection Rules

Set up Redirection Rules

Redirection rules can be adjusted according to your requirements.

After the initial setup is completed, you can create and edit redirection rules as required for rerouting different types of traffic for your NSX-managed workload VMs through the service appliance.

Prerequisites

You must have all the Service Insertion setup completed before you can create redirection rules.

Procedure

1 Navigate to **Security** > **North South Firewall** > **Network Introspection (N-S)**

2 Click **Add Policy**.

Option	Description
Domain	NSX-T Data Center2.4: Select the domain auto-created for the tier-0 gateway for this Transit VPC/VNet, for example, `cloud–vpc–6bcd2c13`. NSX-T Data Center 2.4.1: The Domain object is not visible in the user interface. No action required.
Redirect To:	Select the name of the Virtual Endpoint you created for this service appliance when registering the service. See Create the Service Definition and a Corresponding Virtual Endpoint.

3 Select the new policy and click **Add Rule**. Note the following values specific to service insertion:

Option	Description
Sources	Select a group of subnets whose traffic must be redirected, for example, a group of your NSX-managed workload VMs.
Destinations	Select a list of destination IP addresses or services. such as `Google`, that you want to route through the service appliance.
Applied To	Select the uplink port of the active and standby PCG.
Action	Select **Redirect**.

Enable Syslog Forwarding

NSX Cloud supports syslog forwarding.

You can enable syslog forwarding for Distributed Firewall (DFW) packets on managed VMs. See **Configure Remote Logging** in the *NSX-T Data Center Troubleshooting Guide* for further details.

Do the following:

Procedure

1. Log in to PCG using the jump host.
2. Type `nsxcli` to open the NSX-T Data Center CLI.
3. Type the following commands to enable DFW log forwarding:

   ```
   nsx-public-cloud-gateway> set gw-controller vm-log-forwarding enabled
   nsx-public-cloud-gateway> set logging-server <server-IP-address> proto udp level info messageid
   FIREWALL-PKTLOG
   ```

   After this is set, NSX agent DFW packet logs are available under `/var/log/syslog` on PCG.
4. To enable log forwarding per VM, enter the following command:

   ```
   nsx-public-cloud-gateway> set gw-controller vm-log-forwarding enabled <vm-id>
   ```

FAQ

This lists some frequently asked questions.

I tagged my VM correctly and installed the agent, but my VM is quarantined. What should I do?

If you encounter this problem, try the following:

- Check whether the NSX Cloud tag: `nsx.managed` and its value: `default` are correctly typed in. This is case-sensitive.
- Resync the AWS or Microsoft Azure account from CSM:
  - Log in to CSM.
  - Go to **Clouds > AWS/Azure > Accounts**.
  - Click on **Actions** from the public cloud account tile and click **Resync Account**.

What should I do if I cannot access my workload VM?

Under certain rare conditions, you may lose connectivity to your managed Linux or Windows workload VMs. Try the following steps:

From your Public Cloud (AWS or Microsoft Azure)

- Ensure that all ports on the VM, including those managed by NSX Cloud, the OS firewall (Microsoft Windows or IPTables), and NSX-T Data Center are properly configured in order to allow traffic,

  For example, to allow `ping` to a VM, the following needs to be properly configured:

  - Security Group on AWS or Microsoft Azure. See Manage Quarantine Policy for more information.
  - NSX-T Data Center DFW rules. See DFW Rules for NSX-Managed Workload VMs for details.
  - Windows Firewall or IPTables on Linux.
- Attempt resolving the issue by logging in to the VM using SSH or other methods, for example, the Serial Console in Microsoft Azure.
- You can reboot the locked out VM.
- If you still cannot access the VM, then attach a secondary NIC to the workload VM from which to access that workload VM.