NSX Troubleshooting Guide

Update 8
Modified on 21 FEB 2020
VMware NSX Data Center for vSphere 6.3

vmware®

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

https://docs.vmware.com/

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

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

Contents

NSX Troubleshooting Guide

1

The *NSX Troubleshooting Guide* describes how to monitor and troubleshoot the VMware NSX® for vSphere® system by using the NSX Manager user interface, the vSphere Web Client, and other NSX components, as needed.

Intended Audience

This manual is intended for anyone who wants to use or troubleshoot any problem for NSX in a VMware vCenter environment. The information in this manual is written for experienced system administrators who are familiar with virtual machine technology and virtual datacenter operations. This manual assumes familiarity with VMware vSphere, including VMware ESXi, vCenter Server, and the vSphere Web Client.

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 http://www.vmware.com/support/pubs.

This chapter includes the following topics:

General Troubleshooting Guidelines

This topic describes the general guidelines that you can follow to troubleshoot any problem with NSX for vSphere.

1. Go to the Using the NSX Dashboard and see if there are any errors or warnings displayed for a component.
2. Go to **Monitor** tab of the primary NSX Manager, and see if there are any triggered system events. For more details on system events and alarms, refer to *NSX Logging and System Events*.
3. Use the `GET api/2.0/services/systemalarms` API to view alarms on NSX object. For more information on API, refer to *NSX API Guide*.
4. Resolve the problem as described in the *NSX Troubleshooting Guide*.

5 If your problem is not resolved, download the technical support logs and contact VMware support. See "How to file a Support Request in My VMware". For more information on how to download logs, refer *NSX Logging and System Events*.

Using the NSX Dashboard

The NSX dashboard provides visibility to the overall health of NSX components in one central view. NSX dashboard simplifies troubleshooting by displaying status of different NSX components such as NSX Manager, controllers, logical switches, host preparation, service deployment, backup as well as edge notifications.

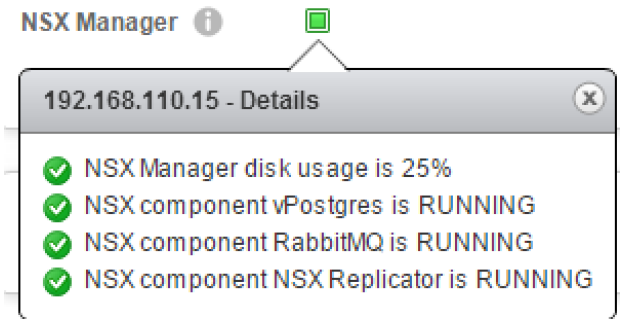
1 Log in to the vSphere Web Client.

2 Click **Networking & Security**, and then click **Dashboard**. The Dashboard page is displayed.

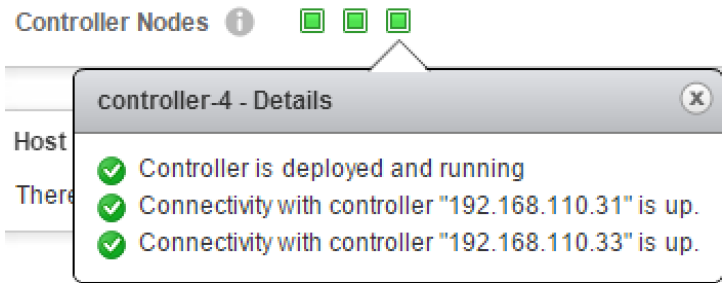
3 In a Cross-vCenter NSX environment, select the NSX Manager with primary role or secondary role.

Dashboard provides the following information:

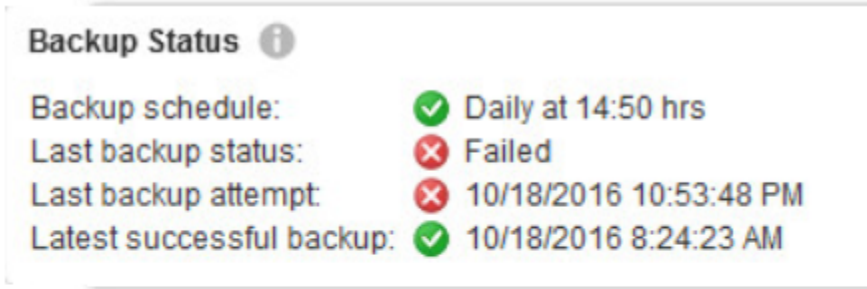
- NSX infrastructure— NSX Managercomponent status for following services is monitored:
  - Database service (vPostgres).
  - Message bus service (RabbitMQ).
  - Replicator service—Also monitors for replication errors (if Cross-vCenter NSX is enabled).
  - NSX Manager disk usage:
    - Yellow indicates disk usage >80%.
    - Red indicates disk usage >90%.

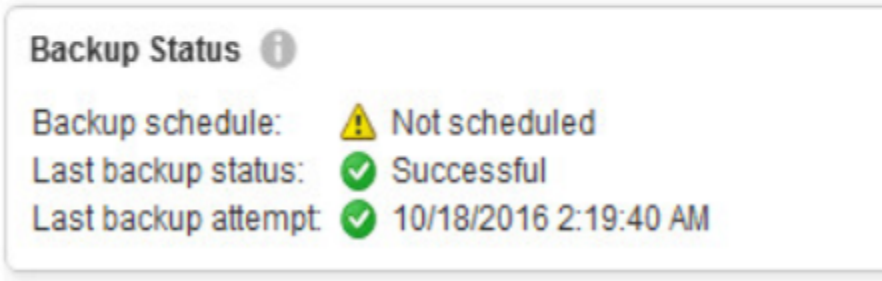


- NSX infrastructure—NSX Controller status:
  - Controller node status (up/down/running/deploying/removing/failed/unknown).
  - Controller peer connectivity status is displayed. If controller is down indicated as Red, then peer controllers are displayed as Yellow.
  - Controller VM status (powered off/deleted).
  - Controller disk latency alerts.



- NSX Manager backup status:
  - Backup schedule.
  - Last backup status (Failed/successful/not scheduled along with date and time).
  - Last backup attempt (date and time with details).
  - Last successful backup (date and time with details).





- NSX infrastructure—Host status for following services is monitored:
  - Deployment related:
    - Number of clusters with installation failed status.
    - Number of clusters that need upgrade.
    - Number of clusters where installation is in progress.
    - Number of unprepared clusters.
  - Firewall:
    - Number of clusters with firewall disabled.

        - Number of clusters where firewall status is yellow/red:
            - Yellow indicates that the distributed firewall is disabled on any of the clusters.
            - Red indicates that the distributed firewall was unable to get installed on any of the hosts/clusters.
    - VXLAN:
        - Number of clusters with VXLAN not configured.
        - Number of clusters where VXLAN status is green/yellow/red:
            - Green indicates that the feature was successfully configured.
            - Yellow means busy when VXLAN configuration is in-progress.
            - Red (error) indicates the state when VTEP creation failed, VTEP could not find the IP address, VTEP got *LinkLocal* IP address assigned, and so on.
- NSX infrastructure—Service deployment status
    - Deployment failures—installation status for the failed deployments.
    - Service status—for all the failed services.
- NSX infrastructure—NSX Edge notifications

  Edge notifications dashboard highlights active alarms for certain services. It monitors list of critical events that are listed below and tracks them till the issue is unresolved. Alarms are auto resolved when recovery event is reported, or edge is force synced, redeployed or upgraded.
    - Load balancer (edge load balancer server status):
        - Edge load balancer back end server is down.
        - Edge load balancer back end server warning status.
    - VPN (IPSec tunnel / IPSec channel status):
        - Edge IPSec channel is down.
        - Edge IPSec tunnel is down.
    - Appliance (edge VM, edge gateway, edge file system, NSX Manager, and edge services gateway reports status):
        - Edge services gateway missing health check pulse.
        - Edge VM got powered off.
        - Edge VM missing health check pulse.
        - NSX Edge reports bad state.
        - NSX Manager reports that edge services gateway is in bad state.
        - Edge VM is not present in VC inventory.

- HA split brain detected.

Note Load balancer and VPN alarms are not auto cleared on configuration update. Once the issue is resolved, you have to clear the alarms manually with API using the `alarm-id` command. Here is the example of API that you can use to clear the alarms. For details, refer to *NSX API Guide*.

```
GET https://<<NSX-IP>>/api/2.0/services/alarms/{source-Id}
POST https://<<NSX-IP>>/api/2.0/services/alarms?action=resolve

GET https://<<NSX-IP>>/api/2.0/services/systemalarms/<alarmId>
POST https://<<NSX-IP>>/api/2.0/services/systemalarms/<alarmId>?action=resolve
```

- NSX services—Firewall Publish status:
  - Number of hosts with Firewall Publish status as failed. Status is Red when any host do not successfully apply the published distributed firewall configuration.
- NSX services—Logical Networking status:
  - Number of logical switches with status `Error` or `Warning`.
  - Flags when backed distributed virtual port group is deleted from vCenter Server.

NSX Command Line Quick Reference

You can use the NSX Command Line Interface (CLI) to troubleshoot problems.

Table 1-1. Checking the NSX Installation on ESXi Host—Commands Run from NSX Manager

Description	Commands on NSX Manager	Notes
List all clusters to get the cluster IDs	`show cluster all`	View all cluster information
List all the hosts in the cluster to get the host IDs	`show cluster` *clusterID*	View the list of hosts in the cluster, the host-ids, and the host-prep installation status
List all the VMs on a host	`show host` *hostID*	View particular host information, VMs, VM IDs, and power status

Table 1-2. Names of VIBs and Modules Installed on Hosts to Use in Commands

NSX version	ESXi version	VIBs	Modules
Any 6.3.x	5.5	esx-vxlan and esx-vsip	vdl2, vdrb, vsip, dvfilter-switch-security, bfd, traceflow
6.3.2 and earlier	6.0 and later	esx-vxlan and esx-vsip	vdl2, vdrb, vsip, dvfilter-switch-security, bfd, traceflow
6.3.3 and later	6.0 and later	esx-nsxv	nsx-vdl2, nsx-vdrb, nsx-vsip, nsx-dvfilter-switch-security, nsx-core, nsx-bfd, nsx-traceflow

Table 1-3. Checking the NSX Installation on ESXi Host—Commands Run from Host

Description	Commands on Host	Notes
VIBs present depend on the NSX and ESXi versions. See table *Names of VIBs and Modules Installed on Hosts* for details on which modules to check on your installation.	`esxcli software vib get --vibname <name>`	Check the version/date installed `esxcli software vib list` displays a list of all VIBs on the system
List all the system modules currently loaded in the system	`esxcli system module list`	Older equivalent command: `vmkload_mod -l | grep -E vdl2|vdrb|vsip|dvfilter-switch-security`
Modules present depend on the NSX and ESXi versions. See table *Names of VIBs and Modules Installed on Hosts* for details on which modules to check on your installation.	`esxcli system module get -m <name>`	Run the command for each module
Two User World Agents (UWA) : control plane agent, firewall agent	`/etc/init.d/vShield-Stateful-Firewall status` `/etc/init.d/netcpad status`	
Check UWAs connection, port 1234 to controllers and 5671 to NSX Manager	`esxcli network ip connection list | grep 1234` `esxcli network ip connection list | grep 5671`	Controller TCP connection Message bus TCP connection
Check EAM status	vSphere Web Client, check **Administration > vSphere ESX Agent Manager**	

Table 1-4. Checking the NSX Installation on ESXi Host—Host Networking Commands

Description	Host Networking Commands	Notes
List physical NICs/vmnic	`esxcli network nic list`	Check the NIC type, driver type, link status, MTU
Physical NIC details	`esxcli network nic get -n vmnic#`	Check the driver and firmware versions along with other details

Table 1-4. Checking the NSX Installation on ESXi Host—Host Networking Commands (continued)

Description	Host Networking Commands	Notes
List vmk NICs with IP addresses/MAC/ MTU, and so on	`esxcli network ip interface ipv4 get`	To ensure VTEPs are correctly instantiated
Details of each vmk NIC, including vDS information	`esxcli network ip interface list`	To ensure VTEPs are correctly instantiated
Details of each vmk NIC, including vDS info for VXLAN vmks	`esxcli network ip interface list --netstack=vxlan`	To ensure VTEPs are correctly instantiated
Find the VDS name associated with this host's VTEP	`esxcli network vswitch dvs vmware vxlan list`	To ensure VTEPs are correctly instantiated
Ping from VXLAN-dedicated TCP/IP stack	`ping ++netstack=vxlan -I vmk1 x.x.x.x`	To troubleshoot VTEP communication issues: add option `-d -s 1572` to make sure that the MTU of transport network is correct for VXLAN
View routing table of VXLAN-dedicated TCP/IP stack	`esxcli network ip route ipv4 list -N vxlan`	To troubleshoot VTEP communication issues
View ARP table of VXLAN-dedicated TCP/IP stack	`esxcli network ip neighbor list -N vxlan`	To troubleshoot VTEP communication issues

Table 1-5. Checking the NSX Installation on ESXi Host—Host Log Files

Description	Log File	Notes
From NSX Manager	`show manager log follow`	Tails the NSX Manager logs For live troubleshooting
Any installation related logs for a host	`/var/log/esxupdate.log`	

Host related issues
VMkernel warning, messages, alerts, and availability report
	`/var/log/vmkernel.log`
	`/var/log/vmksummary.log`
	`/var/log/vmkwarning.log`	

Module load failure is captured	`/var/log/syslog`	IXGBE driver failure NSX modules dependency failure are key indicators
On vCenter, ESX Agent Manager is responsible for updates	In vCenter logs, `eam.log`	

Table 1-6. Checking Logical Switching—Commands Run from NSX Manager

Description	Command on NSX Manager	Notes
List all logical switches	`show logical-switch list all`	List all the logical switches, their UUIDs to be used in API, transport zone, and vdnscope

Table 1-7. Logical Switching—Commands Run from NSX Controller

Description	Commands on Controller	Notes
Find the controller that is the owner of the VNI	`show control-cluster logical-switches vni 5000`	Note the controller IP address in the output and SSH to it
Find all the hosts that are connected to this controller for this VNI	`show control-cluster logical-switch connection-table 5000`	The source IP address in output is the management interface of host, and the port number is the source port of TCP connection
Find the VTEPs registered to host this VNI	`show control-cluster logical-switches vtep-table 5002`	
List the MAC addresses learned for VMs on this VNI	`show control-cluster logical-switches mac-table 5002`	Map that the MAC address is actually on the VTEP reporting it
List the ARP cache populated by the VM IP updates	`show control-cluster logical-switches arp-table 5002`	ARP cache expires in 180 secs
For a specific host/controller pair, find out which VNIs host has joined	`show control-cluster logical-switches joined-vnis <host_mgmt_ip>`	

Table 1-8. Logical Switching—Commands Run from Hosts

Description	Command on Hosts	Notes
Check if the host VXLAN is in-sync or not	`esxcli network vswitch dvs vmware vxlan get`	Shows the sync state and port used for encapsulation
View VM attached and local switch port ID for datapath captures	`net-stats -l`	A nicer way to get vm switchport for a specific VM
Verify VXLAN kernel module vdl2 is loaded	`esxcli system module get -m vdl2`	Shows full detail of the specified module. Verify the version

Verify correct VXLAN VIB version is installed
See table *Names of VIBs and Modules Installed on Hosts* for details on which VIBs to check on your installation.
	`esxcli software vib get --vibname esx-vxlan`
	or
	`esxcli software vib get --vibname esx-nsxv`
		Shows full detail of the specified VIB
		Verify the version and date

Verify the host knows about other hosts in the logical switch	`esxcli network vswitch dvs vmware vxlan network vtep list --vxlan-id=5001 --vds-name=Compute_VDS`	Shows list of all the VTEPs that this host knows about that are hosting vtep 5001
Verify control plane is up and active for a Logical switch	`esxcli network vswitch dvs vmware vxlan network list --vds-name Compute_VDS`	Make sure the controller connection is up and the Port/Mac count matches the VMs on the LS on this host
Verify host has learnt MAC addresses of all VMs	`esxcli network vswitch dvs vmware vxlan network mac list --vds-name Compute_VDS --vxlan-id=5000`	This should list all the MACs for the VNI 5000 VMs on this host

Table 1-8. Logical Switching—Commands Run from Hosts (continued)

Description	Command on Hosts	Notes
Verify host has locally cached ARP entry for remote VM's	`esxcli network vswitch dvs vmware vxlan network arp list --vds-name Compute_VDS --vxlan-id=5000`	Verify host has locally cached ARP entry for remote VM's
Verify VM is connected to LS & mapped to a local VMKnic Also shows what vmknic ID a VM dvPort is mapped to	`esxcli network vswitch dvs vmware vxlan network port list --vds-name Compute_VDS --vxlan-id=5000`	the vdrport will always be listed as long as the VNI is attached to a router
View vmknic ID's and what switchport/ uplink they are mapped to	`esxcli network vswitch dvs vmware vxlan vmknic list --vds-name=DSwitch-Res01`	

Table 1-9. Checking Logical Switching—Log Files

Description	Log File	Notes
Hosts are always connected to controllers hosting their VNIs	`/etc/vmware/netcpa/config-by-vsm.xml`	This file should always have all the controllers in the environment listed The `config-by-vsm.xml` file is created by netcpa process
The `config-by-vsm.xml` file is pushed by NSX Manager using vsfwd If the `config-by-vsm.xml` file is not correct look at the vsfwd log	`/var/log/vsfwd.log`	Parse through this file looking for errors To restart process: `/etc/init.d/vShield-Stateful-Firewall stop|start`
Connection to controller is made using netcpa	`/var/log/netcpa.log`	Parse through this file looking for errors
Logical switching module logs are in *vmkernel.log*	`/var/log/vmkernel.log`	Check logical switching module logs in */var/log/vmkernel.log* "prefixed with VXLAN:"

Table 1-10. Checking Logical Routing—Commands Run from NSX Manager

Description	Commands on NSX Manager	Notes
Commands for ESG	`show edge`	CLI commands for Edge ServicesGateway (ESG) start with 'show edge'
Commands for DLR Control VM	`show edge`	CLI commands for Distributed Logical Router (DLR) Control VM start with 'show edge'
Commands for DLR	`show logical-router`	CLI commands for Distributed Logical Router (DLR) start with `show logical-router`
List all edges	`show edge all`	List all the edges that support the central CLI
List all the services and deployment details of an edge	`show edge edgeID`	View Edge Service Gateway Information

Table 1-10. Checking Logical Routing—Commands Run from NSX Manager (continued)

Description	Commands on NSX Manager	Notes
List the command options for edge	`show edge edgeID ?`	View details, such as version, log, NAT, routing table, firewall, configuration, interface, and services
View routing details	`show edge edgeID ip ?`	View routing info, BGP, OSPF and other details
View routing table	`show edge edgeID ip route`	View the routing table at Edge
View routing neighbor	`show edge edgeID ip ospf neighbor`	View routing neighbor relationship
View logical routers connection information	`show logical-router host hostID connection`	Verify that the number of LIFs connected are correct, the teaming policy is right and the appropriate vDS is being used
List all logical router instances running on the host	`show logical-router host hostID dlr all`	Verify the number of LIFs and routes Controller IP should be same on all hosts for a logical router Control Plane Active should be yes --brief gives a compact response
Check the routing table on the host	`show logical-router host hostID dlr dlrID route`	This is the routing table pushed by the controller to all the hosts in the transport zone This must be same across all the hosts If some of the routes are missing on few hosts, try the sync command from controller mentioned earlier The E flag means routes are learned via ECMP
Check the LIFs for a DLR on the host	`show logical-router host hostID dlr dlrID interface (all | intName) verbose`	The LIF information is pushed to hosts from the controller Use this command to ensure the host knows about all the LIFs it should

Table 1-11. Checking Logical Routing—Commands Run from NSX Controller

Description	Commands on NSX Controller	Notes
Find all the Logical Router Instances	`show control-cluster logical-routers instance all`	This should list the logical router instance and all the hosts in the transport zone which should have the logical router instance on them In addition, shows the Controller that servicing this logical router
View details of each logical router	`show control-cluster logical-routers instance 0x570d4555`	The IP column shows the vmk0 IP addresses of all hosts where this DLR exists
View all the interfaces CONNECTED to the logical router	`show control-cluster logical-routers interface-summary 0x570d4555`	The IP column shows the vmk0 IP addresses of all hosts where this DLR exists

Table 1-11. Checking Logical Routing—Commands Run from NSX Controller (continued)

Description	Commands on NSX Controller	Notes
View all the routes learned by this logical router	`show control-cluster logical-routers routes 0x570d4555`	Note that the IP column shows the vmk0 IP addresses of all hosts where this DLR exists
shows all the network connections established, like a net stat output	`show network connections of-type tcp`	Check if the host you are troubleshooting has netcpa connection Established to controller
Sync interfaces from controller to host	`sync control-cluster logical-routers interface-to-host <logical-router-id> <host-ip>`	Useful if new interface was connected to logical router but is not sync'd to all hosts
Sync routes from controller to host	`sync control-cluster logical-routers route-to-host <logical-router-id> <host-ip>`	Useful if some routes are missing on few hosts but are available on majority of hosts

Table 1-12. Checking Logical Routing—Commands Run from Edge

Description	Commands on Edge or Logical Router Control VM	Notes
View configuration	`show configuration <global | bgp | ospf | …>`	
View the routes learned	`show ip route`	Make sure the routing and forwarding tables are in sync
View the forwarding table	`show ip forwarding`	Make sure the routing and forwarding tables are in sync
View the distributed logical router interfaces	`show interface`	First NIC shown in the output is the distributed logical router interface The distributed logical router interface is not a real vNIC on that VM All the subnets attached to distributed logical router are of type INTERNAL
View the other interfaces (management)	`show interface`	Management/HA interface is a real vNIC on the logical router Control VM If HA was enabled without specifying an IP address, 169.254.x.x/ 30 is used If the management interface is given an IP address, it appears here
debug the protocol	`debug ip ospf` `debug ip bgp`	Useful to see issues with the configuration (such as mismatched OSPF areas, timers, and wrong ASN) Note: output is only seen on the Console of Edge (not via SSH session)

Table 1-12. Checking Logical Routing—Commands Run from Edge (continued)

Description	Commands on Edge or Logical Router Control VM	Notes

OSPF commands	`show configuration ospf`
	`show ip ospf interface`
	`show ip ospf neighbor`
	`show ip route ospf`
	`show ip ospf database`
	`show tech-support` (and look for strings “EXCEPTION” and “PROBLEM”)	

BGP commands	`show configuration bgp`
	`show ip bgp neighbor`
	`show ip bgp`
	`show ip route bgp`
	`show ip forwarding`
	`show tech-support` (look for strings “EXCEPTION” and “PROBLEM”)	

Table 1-13. Checking Logical Routing—Log Files from Hosts

Description	Log File	Notes
Distributed Logical Router instance information is pushed to hosts by vsfwd and saved in XML format	`/etc/vmware/netcpa/config-by-vsm.xml`	If distributed logical router instance is missing on the host, first look at this file to see if the instance is listed If not, restart vsfwd Also, use this file to ensure that all of the controllers are known to the host
The above file is pushed by NSX Manager using vsfwd If the `config-by-vsm.xml` file is not correct look at the vsfwd log	`/var/log/vsfwd.log`	Parse through this file looking for errors To restart process: /etc/init.d/vShield-Stateful-Firewall stop|start
Connection to controller is made using netcpa	`/var/log/netcpa.log`	Parse through this file looking for errors
Logical switching module logs are in vmkernel.log	`/var/log/vmkernel.log`	Check logical switching module logs in /var/log/vmkernel.log “prefixed with vxlan:”

Table 1-14. Controller Debugging—Command Run from NSX Manager

Descripction	Command (On NSX Manager)	Notes
List all controllers with state	`show controller list all`	Shows the list of all controllers and their running state

Table 1-15. Controller Debugging—Command Run from NSX Controller

Description	Command(On Controller)	Notes
Check controller cluster status	`show control-cluster status`	Should always show 'Join complete' and 'Connected to Cluster Majority'
Check the stats for flapping connections and messages	`show control-cluster core stats`	The dropped counter should not change
View the node's activity in relation to joining the cluster initially or after a restart	`show control-cluster history`	This is great for troubleshooting cluster join issues
View list of nodes in the cluster	`show control-cluster startup-nodes`	Note that the list doesn't have to have ONLY have active cluster nodes This should have a list of all the currently deployed controllers This list is used by starting controller to contact other controllers in the cluster
shows all the network connections established, like a net stat output	`show network connections of-type tcp`	Check if the host you are troubleshooting has netcpa connection Established to controller
To restart the controller process	`restart controller`	Only restarts the main controller process Forces a re-connection to the cluster
To reboot the controller node	`restart system`	Reboots the controller VM

Table 1-16. Controller Debugging—Log Files on NSX Controller

Description	Log File	Notes
View controller history and recent joins, restarts. and so on	`show control-cluster history`	Great troubleshooting tool for controller issues especially around clustering
Check for slow disk	`show log cloudnet/cloudnet_java-zookeeper<timestamp>.log filtered-by fsync`	A reliable way to check for slow disks is to look for "fsync" messages in the cloudnet_java-zookeeper log If sync takes more than 1 second, ZooKeeper prints this message, and it is a good indication that something else was utilizing the disk at that time
Check for slow/malfunctioning disk	`show log syslog filtered-by collectd`	Messages like the one in ample output about "collectd" tend to correlate with slow or malfunctioning disks
Check for diskspace usage	`show log syslog filtered-by freespace:`	There is a background job called "freespace" that periodically cleans up old logs and other files from the disk when the space usage reaches some threshold. In some cases, if the disk is small and/or filling up very fast, you'll see a lot of freespace messages. This could be an indication that the disk filled up

Table 1-16. Controller Debugging—Log Files on NSX Controller (continued)

Description	Log File	Notes
Find currently active cluster members	`show log syslog filtered-by Active cluster members`	Lists the node-id for currently active cluster members. May need to look in older syslogs as this message is not printed all the time.
View the core controller logs	`show log cloudnet/cloudnet_java-zookeeper.20150703-165223.3702.log`	There may be multiple zookeeper logs, look at the latest timestamped file This file has information about controller cluster master election and other information related to the distributed nature of controllers
View the core controller logs	`show log cloudnet/cloudnet.nsx-controller.root.log.INFO.20150703-165223.3668`	Main controller working logs, like LIF creation, connection listener on 1234, sharding

Table 1-17. Checking Distributed Firewall—Commands Run from NSX Manager

Description	Commands on NSX Manager	Notes
View a VMs Information	`show vm vmID`	Details such as DC, Cluster, Host, VM Name, vNICs, dvfilters installed
View particular virtual NIC information	`show vnic icID`	Details such as VNIC name, mac address, pg, applied filters
View all cluster information	`show dfw cluster all`	Cluster Name, Cluster Id, Datacenter Name, Firewall Status
View particular cluster information	`show dfw cluster clusterID`	Host Name, Host Id, Installation Status
View dfw related host information	`show dfw host hostID`	VM Name, VM Id, Power Status
View details within a dvfilter	`show dfw host hostID filter filterID <option>`	List rules, stats, address sets etc for each VNIC
View DFW information for a VM	`show dfw vm vmID`	View VM's name, VNIC ID, filters, and so on
View VNIC details	`show dfw vnic vnicID`	View VNIC name, ID, MAC address, portgroup, filter
List the filters installed per vNIC	`show dfw host hostID summarize-dvfilter`	Find the VM/vNIC of interest and get the name field to use in the next commands as filter
View rules for a specific filter/vNIC	`show dfw host hostID filter filterID rules` `show dfw vnic nicID`	
View details of an address set	`show dfw host hostID filter filterID addrsets`	The rules only display address sets, this command can be used to expand what is part of an address set
Spoofguard details per vNIC	`show dfw host hostID filter filterID spoofguard`	Check if SpoofGuard is enabled and what is the current IP/MAC

Table 1-17. Checking Distributed Firewall—Commands Run from NSX Manager (continued)

Description	Commands on NSX Manager	Notes
View details of flow records	`show dfw host hostID filter filterID flows`	If flow monitoring is enabled, host sends flow information periodically to NSX Manager Use this command to see flows per vNIC
View statistics for each rule for a vNIC	`show dfw host hostID filter filterID stats`	This is useful to see if rules are being hit

Table 1-18. Checking Distributed Firewall—Commands Run from Hosts

Description	Commands on Host	Notes

Lists VIBs downloaded on the host.
See table *Names of VIBs and Modules Installed on Hosts* for details on which VIBs to check on your installation.
	`esxcli software vib list | grep esx-vsip`
	or
	`esxcli software vib list | grep esx-nsxv`
		Check to make sure right vib version is downloaded

Details on system modules currently loaded
See table *Names of VIBs and Modules Installed on Hosts* for details on which modules to check on your installation.
	`esxcli system module get -m vsip`
	or
	`esxcli system module get -m nsx-vsip`
		Check to make sure that the module was installed/loaded

Process list	`ps | grep vsfwd`	View if the vsfwd process is running with several threads
Daemon command	`/etc/init.d/vShield-Stateful-Firewall {start|stop|status|restart}`	Check if the daemon is running and restart if needed
View network connection	`esxcli network ip connection list | grep 5671`	Check if the host has TCP connectivity to NSX Manager

Table 1-19. Checking Distributed Firewall—Log Files on Hosts

Description	Log	Notes
Process log	`/var/log/vsfwd.log`	vsfwd deamon log, useful for vsfwd process, NSX Manager connectivity, and RabbitMQ troubleshooting
Packet logs dedicated file	`/var/log/dfwpktlogs.log`	Dedicated log file for packet logs
Packet capture at the dvfilter	`pktcap-uw --dvfilter nic-1413082-eth0-vmware-sfw.2 --outfile test.pcap`	

NSX Host Health Check

From the NSX Manager central CLI, you can check the health status of each ESXi host.

The health status is reported as critical, unhealthy, or healthy.

For example:

```
nsxmgr> show host host-30 health-status
status: HEALTHY

nsxmgr> show host host-29 health-status
UNHEALTHY, Standard Switch vSwitch1 has no uplinks.
UNHEALTHY, Storage volume datastore1 has no enough free spaces: 19.% free.
status: UNHEALTHY

nsxmgr> show host host-28 health-status
CRITICAL, VXLAN VDS vds-site-a VNI 200000 multicast addr is not synchronized with VSM: 0.0.0.0.
CRITICAL, VXLAN VDS vds-site-a VNI 200003 multicast addr is not synchronized with VSM: 0.0.0.0.
CRITICAL, VXLAN VDS vds-site-a VNI 5000 multicast addr is not synchronized with VSM: 0.0.0.0.
Status: CRITICAL
```

The host-check command can also be invoked through the NSX Manager API.

Troubleshooting NSX Infrastructure

2

NSX preparation is a 4-step process.

1 Connect NSX Manager to vCenter Server. There is a one-to-one relationship between NSX Manager and vCenter Server.

   a Register with vCenter Server.

2 Deploy NSX Controllers (Only required for logical switching, distributed routing, or VXLAN in unicast or hybrid mode. If you are only using distributed firewall (DFW), controllers are not required).

3 Host Preparation: Installs VIBs for VXLAN, DFW, and DLR on all hosts in the cluster. Configures the Rabbit MQ-based messaging infrastructure. Enables firewall. Notifies controllers that hosts are ready for NSX.

4 Configure IP pool settings and configure VXLAN: Creates a VTEP port group and VMKNICs on all hosts in the cluster. During this step, you can set the transport VLAN ID, teaming policy, and MTU.

For more information about installation and configuration of each step, refer to *NSX Installation Guide* and *NSX Administration Guide*.

This chapter includes the following topics:

Host Preparation

vSphere ESX Agent Manager deploys vSphere installation bundles (VIBs) onto ESXi hosts.

The deployment on hosts requires that DNS be configured on the hosts, vCenter Server, and NSX Manager. Deployment does not require an ESXi host reboot, but any update or removal of VIBs requires an ESXi host reboot.

VIBs are hosted on NSX Manager and are also available as a zip file.

The file can be accessed from `https://<NSX-Manager-IP>/bin/vdn/nwfabric.properties`. The downloadable zip file differs based on NSX and ESXi version. For example, in NSX 6.3.0, vSphere 6.0 hosts use the file `https://<NSX-Manager-IP>/bin/vdn/vibs-6.3.0/6.0-`*`buildNumber`*`/vxlan.zip`.

```
# 5.5 VDN EAM Info
VDN_VIB_PATH.1=/bin/vdn/vibs-6.3.0/5.5-4744075/vxlan.zip
VDN_VIB_VERSION.1=4744075
VDN_HOST_PRODUCT_LINE.1=embeddedEsx
VDN_HOST_VERSION.1=5.5.*

# 6.0 VDN EAM Info
VDN_VIB_PATH.2=/bin/vdn/vibs-6.3.0/6.0-4744062/vxlan.zip
VDN_VIB_VERSION.2=4744062
VDN_HOST_PRODUCT_LINE.2=embeddedEsx
VDN_HOST_VERSION.2=6.0.*

# 6.5 VDN EAM Info
VDN_VIB_PATH.3=/bin/vdn/vibs-6.3.0/6.5-4744074/vxlan.zip
VDN_VIB_VERSION.3=4744074
VDN_HOST_PRODUCT_LINE.3=embeddedEsx
VDN_HOST_VERSION.3=6.5.*

# Single Version associated with all the VIBs pointed by above VDN_VIB_PATH(s)
VDN_VIB_VERSION=6.3.0.4744320

# Legacy vib location. Used by code to discover available legacy vibs.
LEGACY_VDN_VIB_PATH_FS=/common/em/components/vdn/vibs/legacy/
LEGACY_VDN_VIB_PATH_WEB_ROOT=/bin/vdn/vibs/legacy/
```

The VIBs installed on a host depends on the NSX and ESXi versions:

ESXi version	NSX version	VIBs installed
5.5	Any 6.3.x	■ esx-vsip ■ esx-vxlan
6.0 or later	6.3.2 or earlier	■ esx-vsip ■ esx-vxlan
6.0 or later	6.3.3 or later	■ esx-nsxv

You can view the installed VIBs using the `esxcli software vib list` command.

```
[root@esx-01a:~] esxcli software vib list | grep -e vsip -e vxlan
esx-vsip                       6.0.0-0.0.XXXXXXX                    VMware  VMwareCertified
2016-04-20
esx-vxlan                      6.0.0-0.0.XXXXXXX                    VMware  VMwareCertified
2016-04-20
```

or

```
esxcli software vib list | grep nsxv
esx-nsxv                       6.0.0-0.0.XXXXXXX                  VMware  VMwareCertified
2017-08-11
```

Common Issues During Host Preparation

During the preparation of hosts typical kinds of issues that can be encountered are as follows:

- EAM fails to deploy VIBs.
  - Might be due to incorrect configured DNS on hosts.
  - Might be due to a firewall blocking required ports between ESXi, NSX Manager, and vCenter Server.

  Most of the issues are resolved by clicking the **Resolve** option. Refer to Installation Status Is Not Ready .
- A previous VIB of an older version is already installed. This requires user intervention to reboot hosts.
- NSX Manager and vCenter Server experience communication issues. The **Host Preparation** tab in the Networking and Security Plug-in not showing all hosts properly:
  - Check if vCenter Server can enumerate all hosts and clusters.

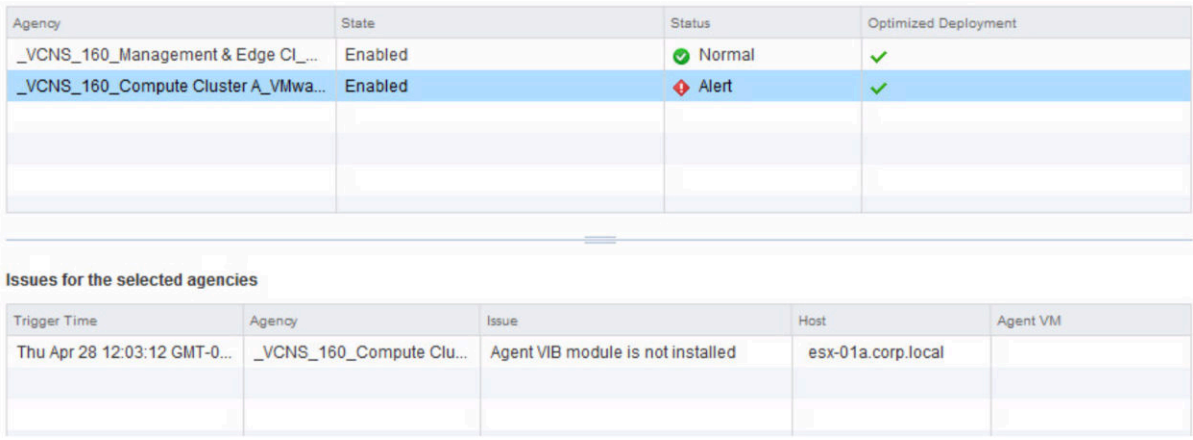
If problem is not fixed with the **Resolve** option, refer to Problem Not Fixed With the Resolve Option.

Host Preparation (VIBs) Troubleshooting

- Check communication channel health for the host. See Checking Communication Channel Health.
- Check vSphere ESX Agent Manager for errors.

  vCenter home > Administration > vCenter Server Extensions > vSphere ESX Agent Manager.

  On vSphere ESX Agent Manager, check the status of agencies that are prefixed with “VCNS160”. If an agency has a bad status, select the agency and view its issues.



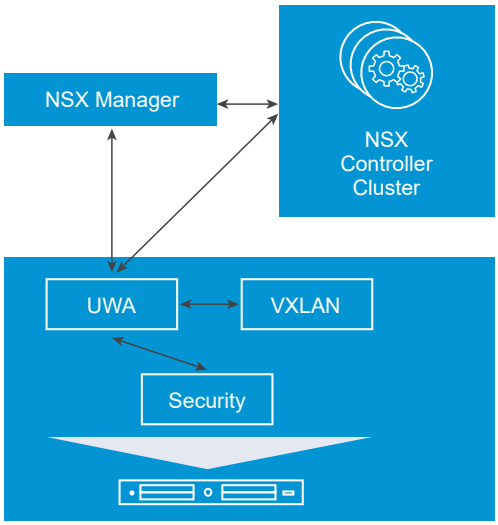
- On the host that is having an issue, run the `tail /var/log/esxupdate.log` command.

```
2016-04-28T19:02:52Z esxupdate: downloader: DEBUG: Downloading https://vcsa-(
o /tmp/tmpKT0wjN...
2016-04-28T19:03:12Z esxupdate: esxupdate: ERROR: An esxupdate error exceptic
2016-04-28T19:03:12Z esxupdate: esxupdate: ERROR: Traceback (most recent call
2016-04-28T19:03:12Z esxupdate: esxupdate: ERROR:   File "/usr/sbin/esxupdate
2016-04-28T19:03:12Z esxupdate: esxupdate: ERROR:     cmd.Run()
2016-04-28T19:03:12Z esxupdate: esxupdate: ERROR:   File "/build/mts/release/
site-packages/vmware/esx5update/Cmdline.py", line 106, in Run
2016-04-28T19:03:12Z esxupdate: esxupdate: ERROR:   File "/build/mts/release/
site-packages/vmware/esximage/Transaction.py", line 73, in DownloadMetadatas
2016-04-28T19:03:12Z esxupdate: esxupdate: ERROR: MetadataDownloadError: ('ht
fd3f37ad4c', None, "('https://vcsa-01a.corp.local:443/eam/vib?id=facdb160-216
rlopen error [Errno -3] Temporary failure in name resolution>')")
2016-04-28T19:03:12Z esxupdate: esxupdate: DEBUG: <<<
```

Host Preparation (UWA) Troubleshooting

NSX Manager configures two user world agents on all hosts in a cluster:

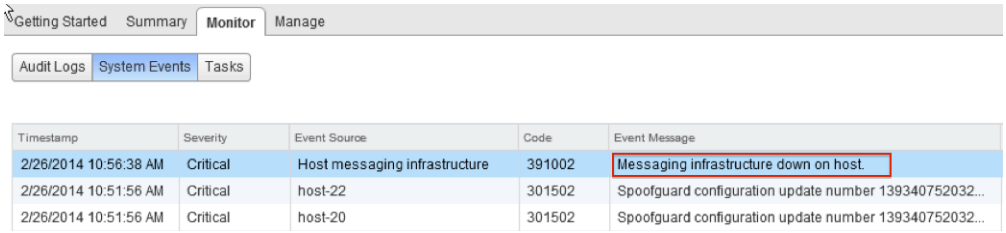
- Messaging bus UWA (vsfwd)
- Control plane UWA (netcpa)



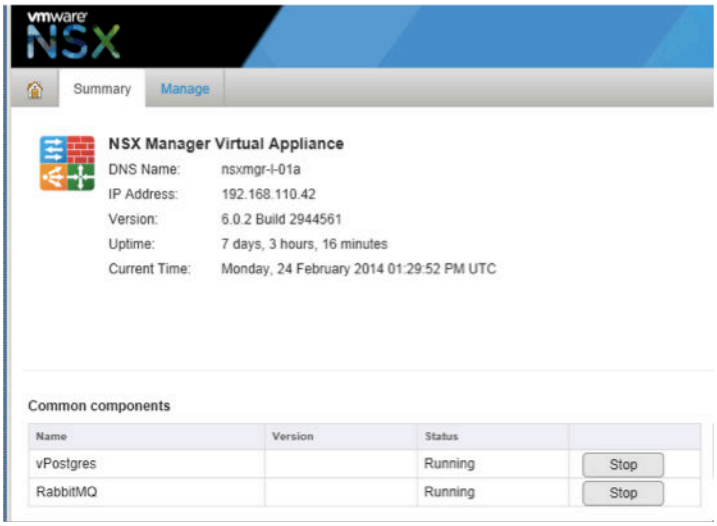
In rare cases, the installation of the VIBs succeeds but for some reason one or both of the user world agents is not functioning correctly. This could manifest itself as:

- The firewall showing a bad status.

- The control plane between hypervisors and the controllers being down. Check NSX Manager System Events. Refer to *NSX Logging and System Events*.



If more than one ESXi host is affected, check the status of message bus service on NSX Manager Appliance web UI under the **Summary** tab. If RabbitMQ is stopped, restart it.



If the message bus service is active on NSX Manager:

- Check the messaging bus user world agent status on the hosts by running the `/etc/init.d/vShield-Stateful-Firewall status` command on ESXi hosts.

```
[root@esx-01a:~] /etc/init.d/vShield-Stateful-Firewall status
vShield-Stateful-Firewall is running
```

- Check the message bus user world logs on hosts at `/var/log/vsfwd.log`.
- Run the `esxcfg-advcfg -l | grep Rmq` command on ESXi hosts to show all Rmq variables. There should be 16 Rmq variables.

```
[root@esx-01a:~] esxcfg-advcfg -l | grep Rmq
/UserVars/RmqIpAddress [String] : Connection info for RMQ Broker
/UserVars/RmqUsername [String] : RMQ Broker Username
/UserVars/RmqPassword [String] : RMQ Broker Password
/UserVars/RmqVHost [String] : RMQ Broker VHost
/UserVars/RmqVsmRequestQueue [String] : RMQ Broker VSM Request Queue
/UserVars/RmqPort [String] : RMQ Broker Port
/UserVars/RmqVsmExchange [String] : RMQ Broker VSM Exchange
/UserVars/RmqClientPeerName [String] : RMQ Broker Client Peer Name
/UserVars/RmqHostId [String] : RMQ Broker Client HostId
/UserVars/RmqHostVer [String] : RMQ Broker Client HostVer
```

```
/UserVars/RmqClientId [String] : RMQ Broker Client Id
/UserVars/RmqClientToken [String] : RMQ Broker Client Token
/UserVars/RmqClientRequestQueue [String] : RMQ Broker Client Request Queue
/UserVars/RmqClientResponseQueue [String] : RMQ Broker Client Response Queue
/UserVars/RmqClientExchange [String] : RMQ Broker Client Exchange
/UserVars/RmqSslCertSha1ThumbprintBase64 [String] : RMQ Broker Server Certificate base64 Encoded
Sha1 Hash
```

- Run the `esxcfg-advcfg -g /UserVars/RmqIpAddress` command on ESXi hosts. The output should display the NSX Manager IP address.

```
[root@esx-01a:~] esxcfg-advcfg -g /UserVars/RmqIpAddress
Value of RmqIpAddress is 192.168.110.15
```

- Run the `esxcli network ip connection list | grep 5671` command on ESXi hosts to check for active messaging bus connection.

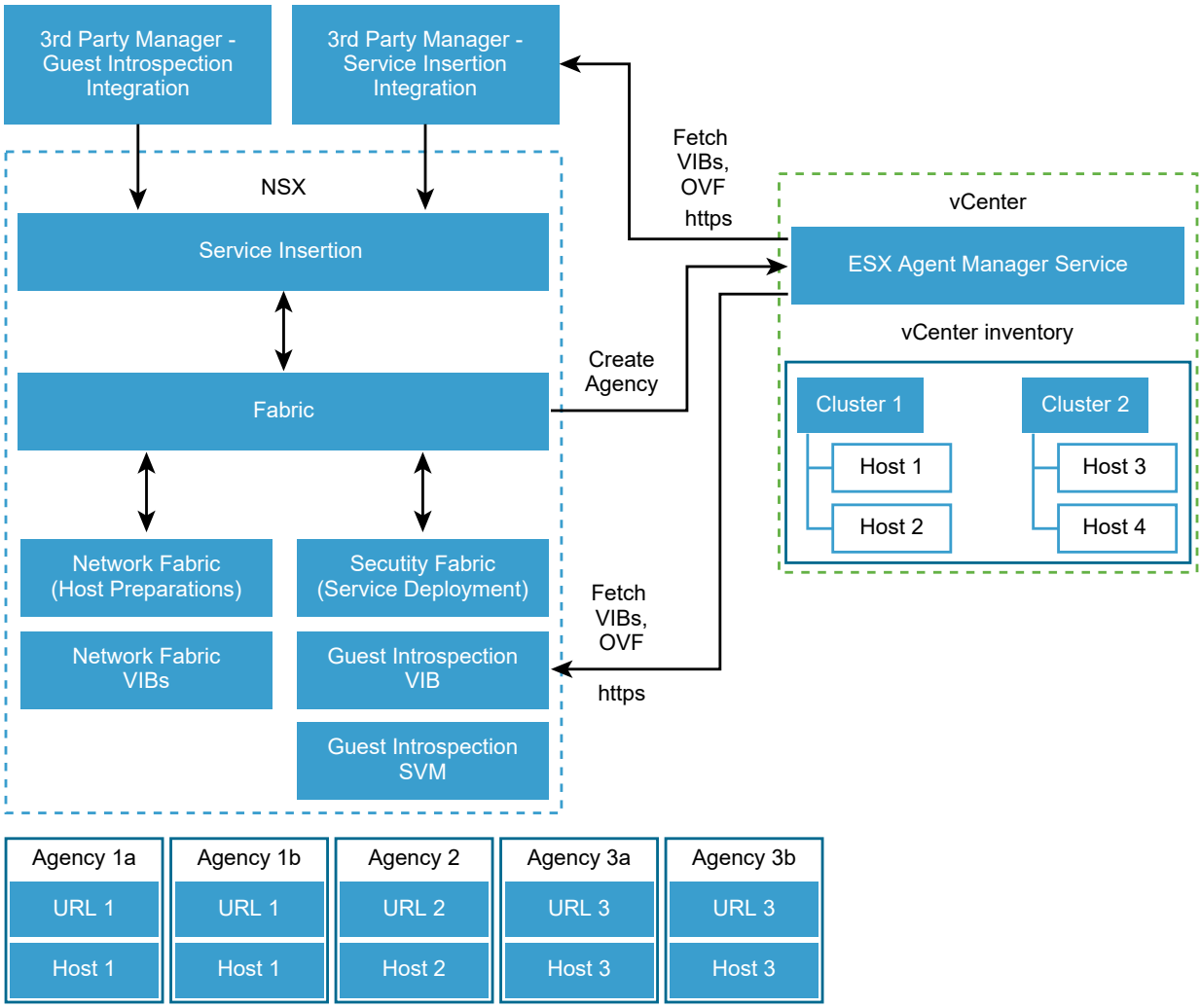
```
[root@esx-01a:~] esxcli network ip connection list | grep 5671
tcp         0       0  192.168.110.51:29969            192.168.110.15:5671   ESTABLISHED
35505  newreno  vsfwd
tcp         0       0  192.168.110.51:29968            192.168.110.15:5671   ESTABLISHED
35505  newreno  vsfwd
```

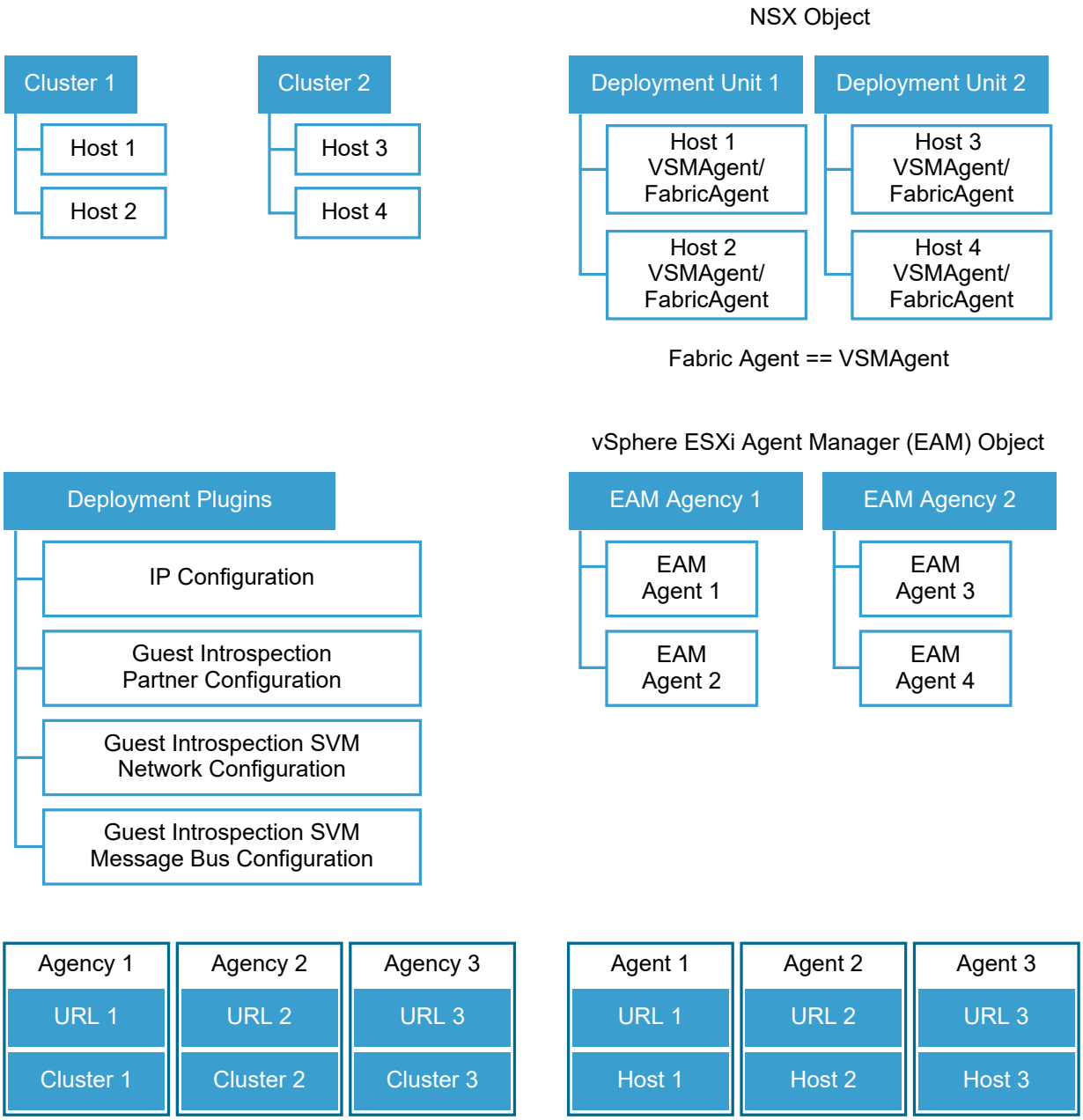
For problems related to control plane agent, refer to Control Plane Agent (netcpa) Issues.

Understanding Host Preparation Architecture

This topic explains the basic host preparation architecture.

- To deploy the network fabric, go to the **Host Preparation** tab.
- To deploy the security fabric, go to the **Service Deployment** tab.


3rd Party Manager - Guest Introspection Integration
3rd Party Manager - Service Insertion Integration
NSX
Service Insertion
Fabric
Network Fabric (Host Preparations)
Secutity Fabric (Service Deployment)
Network Fabric VIBs
Guest Introspection VIB
Guest Introspection SVM
Fetch VIBs, OVF
https
Create Agency
Fetch VIBs, OVF
https
vCenter
ESX Agent Manager Service
vCenter inventory
Cluster 1
Cluster 2
Host 1
Host 2
Host 3
Host 4
Agency 1a
URL 1
Host 1
Agency 1b
URL 1
Host 1
Agency 2
URL 2
Host 2
Agency 3a
URL 3
Host 3
Agency 3b
URL 3
Host 3


NSX Object
Cluster 1
Host 1
Host 2
Cluster 2
Host 3
Host 4
Deployment Unit 1
Host 1
VSMAgent/
FabricAgent
Host 2
VSMAgent/
FabricAgent
Deployment Unit 2
Host 3
VSMAgent/
FabricAgent
Host 4
VSMAgent/
FabricAgent
Fabric Agent == VSMAgent
vSphere ESXi Agent Manager (EAM) Object
Deployment Plugins
IP Configuration
Guest Introspection
Partner Configuration
Guest Introspection SVM
Network Configuration
Guest Introspection SVM
Message Bus Configuration
EAM Agency 1
EAM
Agent 1
EAM
Agent 2
EAM Agency 2
EAM
Agent 3
EAM
Agent 4
Agency 1
URL 1
Cluster 1
Agency 2
URL 2
Cluster 2
Agency 3
URL 3
Cluster 3
Agent 1
URL 1
Host 1
Agent 2
URL 2
Host 2
Agent 3
URL 3
Host 3

The following terms can help you to understand the host preparation architecture:

Fabric	Fabric is a software layer in NSX Manager which interacts with ESX Agent Manager to install network and security fabric services on hosts.
Network Fabric	Network fabric services are deployed on a cluster. Network fabric services include host preparation, VXLAN, distributed routing, distributed firewall, and message bus.
Security Fabric	Security fabric services are deployed on a cluster. Security fabric services include Guest Introspection and partner security solutions.
Fabric Agent	A fabric agent is a combination of a fabric service and a host in the NSX Manager database. One fabric agent is created per host for a cluster on which a networking or security fabric service is deployed. Also known as: VSM agent
Deployment Unit	A combination of a fabric service and a cluster in the NSX Manager database. A deployment unit must be created for networking and security services to get installed.
ESX Agent Manager Agent	An ESX Agent Manager Agent is a combination of a service specification and a host in the vCenter Server database. An ESX Agent Manager agent maps to an NSX Fabric Agent.
ESX Agent Manager Agency	An ESX Agent Manager Agency is a combination of a specification and a cluster in the vCenter Server database. The specification describes ESX Agent Manager agents and VIBs, OVFs and their configuration (such as datastore and network settings) that it manages. The NSX Manager creates an ESX Agent Manager agency for each of the clusters that are being prepared. An ESX Agent Manager agency maps to an NSX deployment unit. The NSX Manager database of deployment units and the vCenter ESX Agent Manager database of ESX Agent Manager agencies must be in sync. In rare cases, if the two databases are not in sync, then NSX triggers events and alarms to notify you of the condition. NSX Manager creates a Deployment Unit on its database for each ESX Agent Manager agency.

The NSX Manager creates an ESX Agent Manager agency for each of the clusters that are being prepared. NSX Manager creates a Deployment Unit on its database for each ESX Agent Manager agency. One ESX Agent Manager agency = One Deployment Unit .

You can view agencies in the following ways:

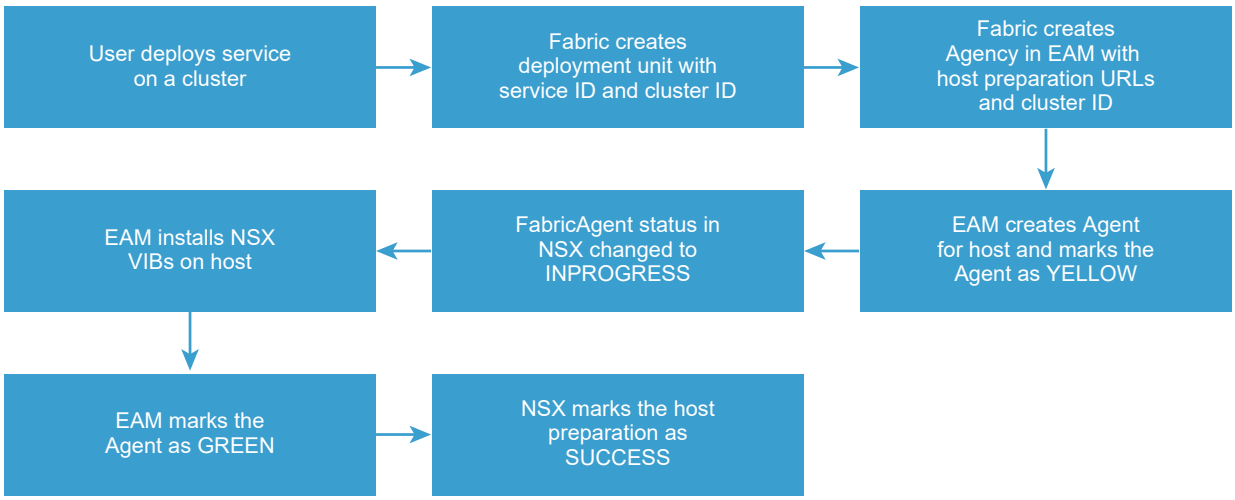
- From the EAM MOB *https://<VC-hostname/IP>/eam/mob/*.
- From the vSphere Web Client:
  - Go to **vCenter Solutions Manager > vSphere ESX Agent Manager > Manage**.
  - Under **ESX Agencies**, you can see the agencies (one per cluster that has been prepared for a host).

The lifecycle of a deployment unit is tied to that of the agency and removal of an agency from ESX Agent Manager results in removal of the corresponding deployment unit from the NSX.

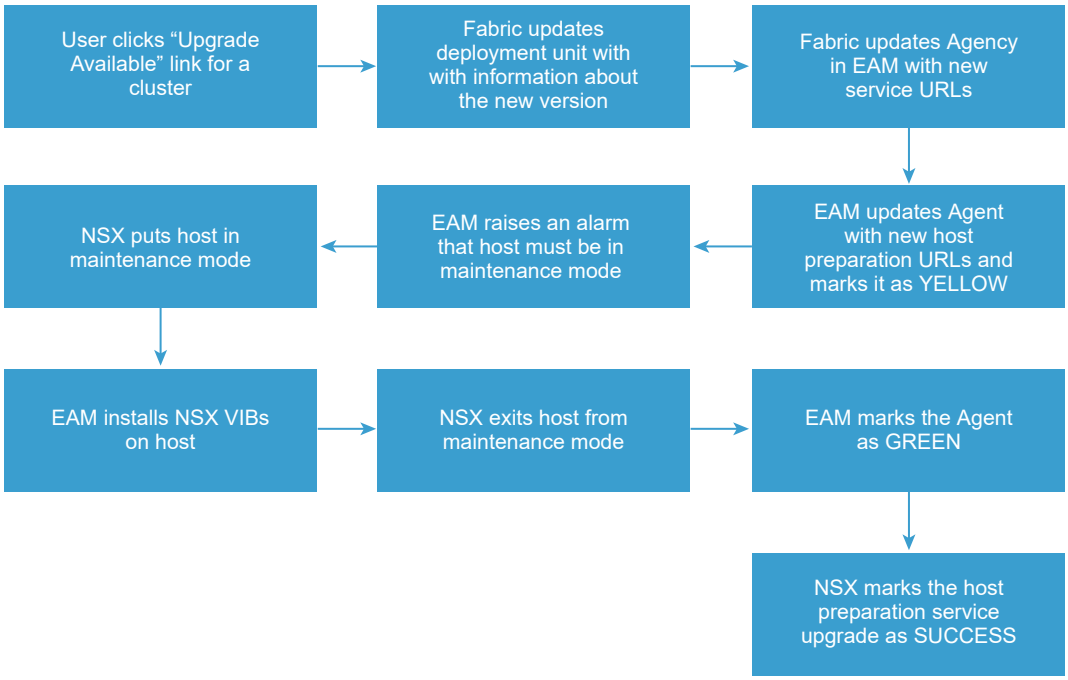
Service Deployment Workflow for Host Preparation

This topic displays the service deployment workflow (install and upgrade) for host preparation.

Install Workflow



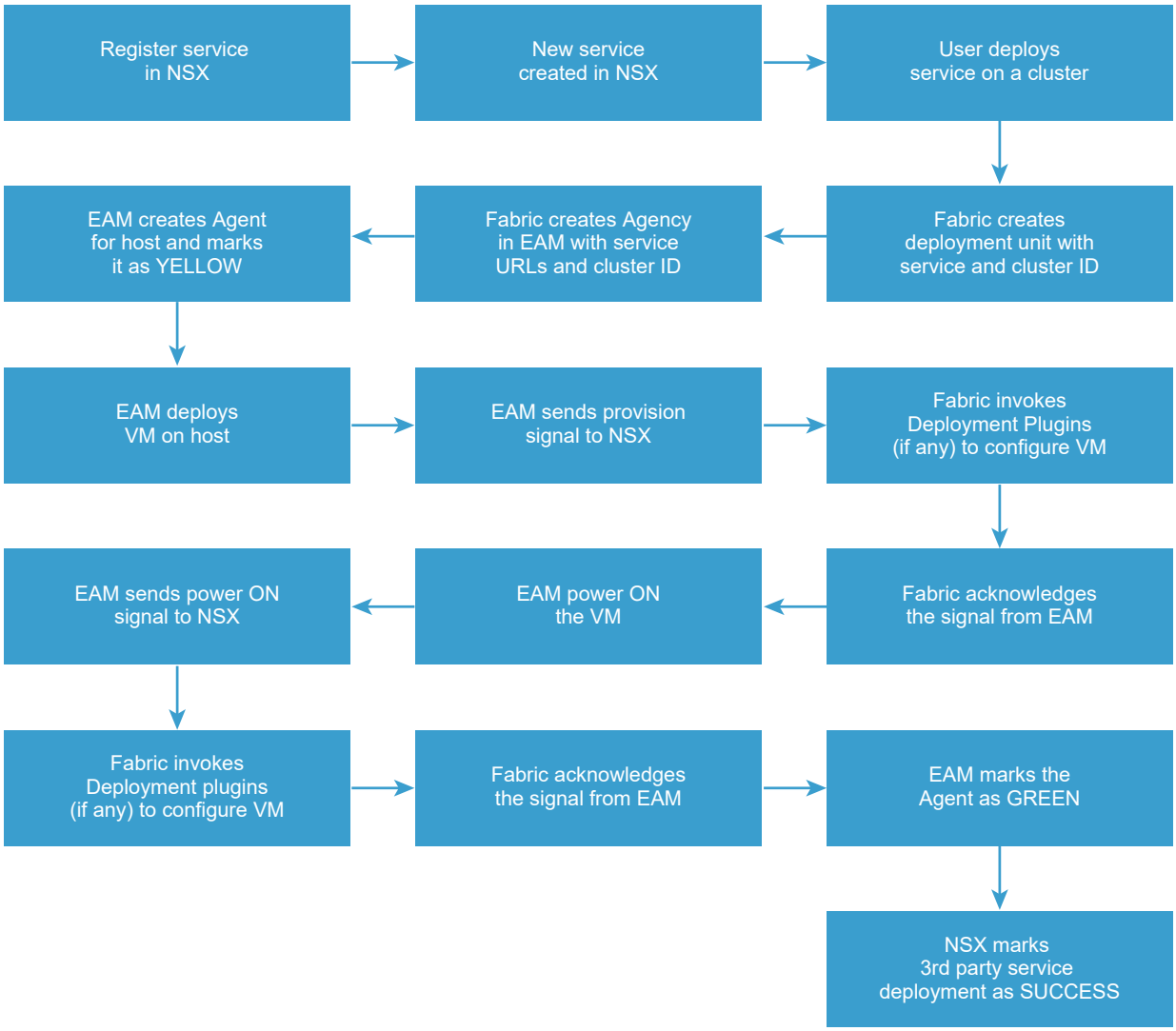
Upgrade Workflow



Service Deployment Workflow for Third Party Services

This topic displays the service deployment workflow (install and upgrade) for third party services.

Install Workflow



Upgrade Workflow

Checking Communication Channel Health

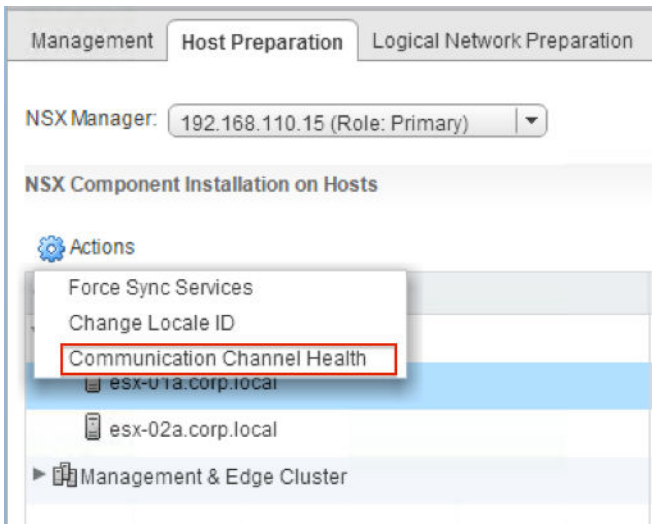
From vSphere Web Client, you can check the status of communication between various components.

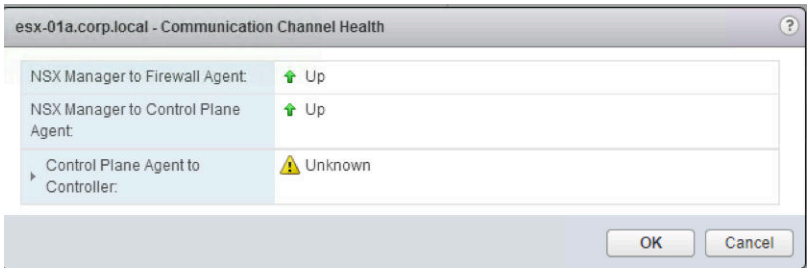
To check the communication channel health between NSX Manager and the firewall agent, NSX Manager and the control plane agent, and the control plane agent and controllers, perform the following steps:

1 In vSphere Web Client, navigate to **Networking & Security > Installation > Host Preparation**.

2 Select a cluster or expand a cluster and select a host. Click **Actions** ( ) then **Communication Channel Health**.

The communication channel health information is displayed.





If the status of any of the three connections for a host changes, a message is written to the NSX Manager log. In the log message, the status of a connection can be UP, DOWN, or NOT_AVAILABLE (displayed as Unknown in vSphere Web Client). If the status changes from UP to DOWN or NOT_AVAILABLE, a warning message is generated. For example:

```
2016-05-23 23:36:34.736 GMT+00:00  WARN TaskFrameworkExecutor-25 VdnInventoryFacadeImpl
$HostStatusChangedEventHandler:200 - Host Connection Status Changed: Event Code: 1941, Host:
esx-04a.corp.local (ID: host-46), NSX Manager - Firewall Agent: UP, NSX Manager - Control Plane
Agent: UP, Control Plane Agent - Controllers: DOWN.
```

If the status changes from DOWN or NOT_AVAILABLE to UP, an INFO message that is similar to the warning message is generated. For example:

```
2016-05-23 23:55:12.736 GMT+00:00  INFO TaskFrameworkExecutor-25 VdnInventoryFacadeImpl
$HostStatusChangedEventHandler:200 - Host Connection Status Changed: Event Code: 1938, Host:
esx-04a.corp.local (ID: host-46), NSX Manager - Firewall Agent: UP, NSX Manager - Control Plane
Agent: UP, Control Plane Agent - Controllers: UP.
```

If the control plane channel experiences a communication fault, a system event with one of the following granular failure reason is generated:

- 1255601: Incomplete Host Certificate
- 1255602: Incomplete Controller Certificate
- 1255603: SSL Handshake Failure

- 1255604: Connection Refused
- 1255605: Keep-alive Timeout
- 1255606: SSL Exception
- 1255607: Bad Message
- 1255620: Unknown Error

Also, heartbeat messages are generated from NSX Manager to hosts. A configuration full sync is triggered, if heartbeat between the NSX Manager and netcpa is lost.

For more information on how to download logs, refer to *NSX Administration Guide*.

Installation Status Is Not Ready

During host preparation, you may notice that the cluster status is displayed as `Not Ready`.

Problem

On the **Host Preparation** tab or **Service Deployment** tab, the installation status appears as `Not Ready`.

Solution

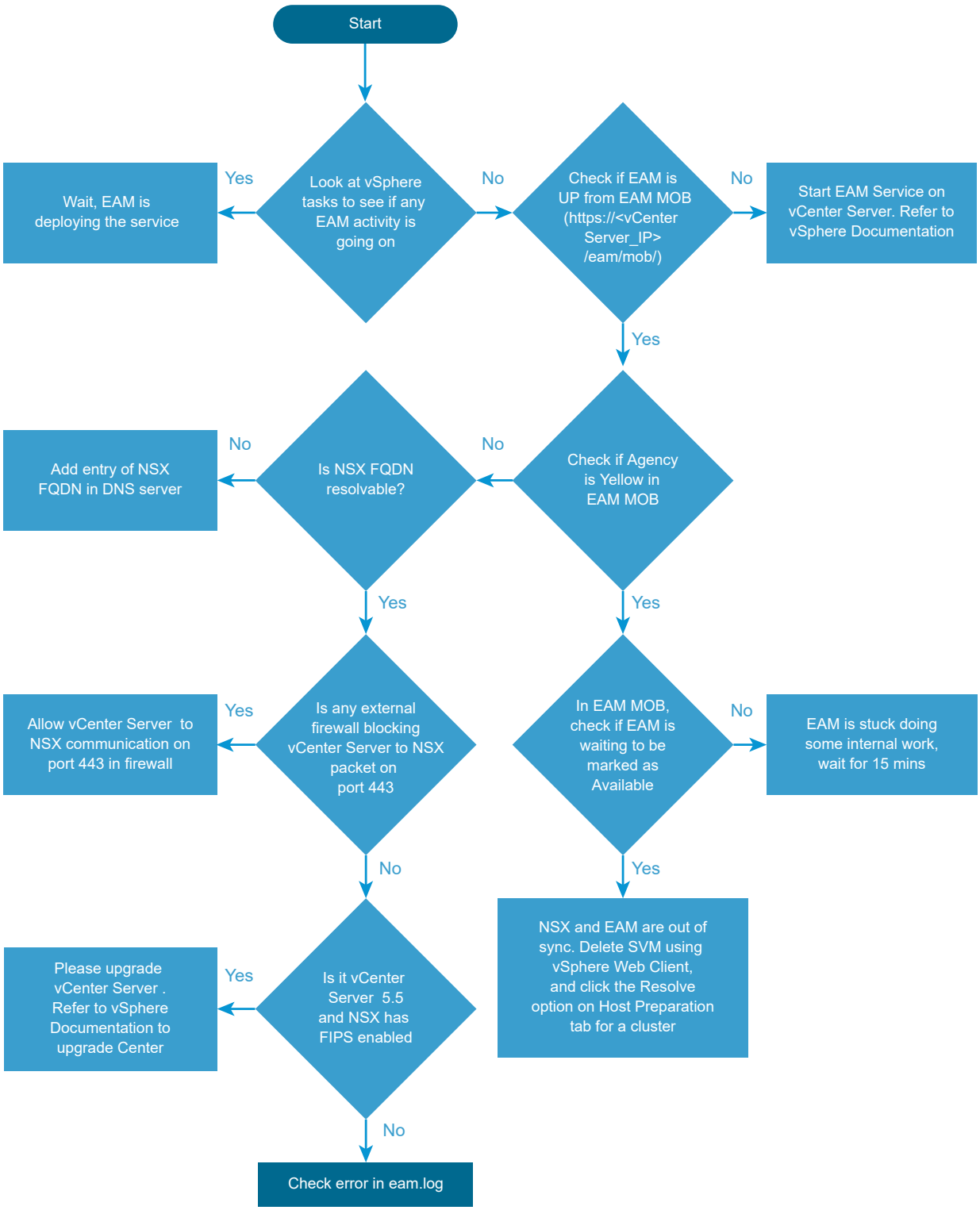
1. Go to the **Networking & Security > Installation**> **Host Preparation** tab or **Service Deployment** tab.
2. On the clusters and hosts, click `Not Ready`.

   You see error message.
3. Click the **Resolve** option.

   To see list of issues that are resolved by the **Resolve** option, refer to *NSX Logging and System Events*.
4. If you still see `Not Ready` and error is still not resolved, refer to Problem Not Fixed With the Resolve Option.

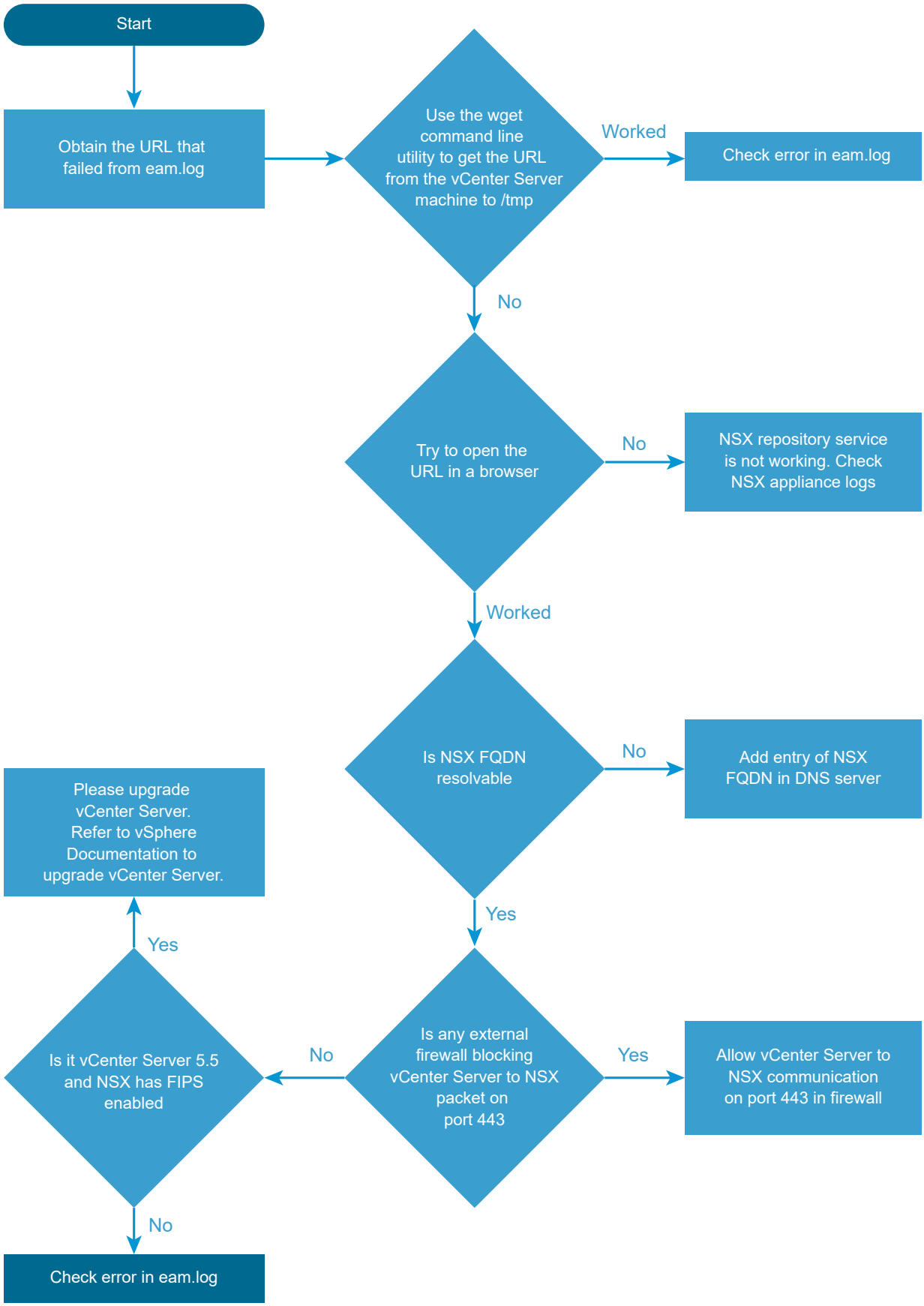
Service Not Responding

The flowchart is as an overview of the NSX host preparation process and what to do when the service is not responding for a long time or showing spinning icon for a long time.


Start
Look at vSphere tasks to see if any EAM activity is going on
Yes
Wait, EAM is deploying the service
No
Check if EAM is UP from EAM MOB (https://<vCenter Server_IP> /eam/mob/)
No
Start EAM Service on vCenter Server. Refer to vSphere Documentation
Yes
Check if Agency is Yellow in EAM MOB
No
Is NSX FQDN resolvable?
No
Add entry of NSX FQDN in DNS server
Yes
Is any external firewall blocking vCenter Server to NSX packet on port 443
Yes
Allow vCenter Server to NSX communication on port 443 in firewall
No
Is it vCenter Server 5.5 and NSX has FIPS enabled
Yes
Please upgrade vCenter Server . Refer to vSphere Documentation to upgrade Center
No
Check error in eam.log
Yes
In EAM MOB, check if EAM is waiting to be marked as Available
No
EAM is stuck doing some internal work, wait for 15 mins
Yes
NSX and EAM are out of sync. Delete SVM using vSphere Web Client, and click the Resolve option on Host Preparation tab for a cluster

Service Deployment Fails with OVF/VIB Not Accessible Error

The flowchart displays what to do when the service deployment fails with an `OVF/VIB not accessible` error.


Start
Obtain the URL that failed from eam.log
Use the wget command line utility to get the URL from the vCenter Server machine to /tmp
Worked
Check error in eam.log
No
Try to open the URL in a browser
No
NSX repository service is not working. Check NSX appliance logs
Worked
Is NSX FQDN resolvable
No
Add entry of NSX FQDN in DNS server
Yes
Is any external firewall blocking vCenter Server to NSX packet on port 443
Yes
Allow vCenter Server to NSX communication on port 443 in firewall
No
Is it vCenter Server 5.5 and NSX has FIPS enabled
Yes
Please upgrade vCenter Server. Refer to vSphere Documentation to upgrade vCenter Server.
No
Check error in eam.log

Problem Not Fixed With the Resolve Option

On the **Networking & Security > Installation** > **Host Preparation** tab or **Service Deployment** tab, the installation status appears as `Not Ready` on the clusters and hosts. Clicking the **Resolve** option does not fix the problem.

Problem

- Clicking the `Not Ready` link shows error as `VIB module for agent is not installed on the host`.
- ESXi host fails to access VIBs from the vCenter Server.
- While changing from vShield Endpoint to NSX Manager, you may see status as `Failed`.

Solution

1. Verify that the DNS is configured correctly on the vCenter Server, ESXi hosts and the NSX Manager. Ensure that the forward and reverse DNS resolution from the vCenter Server, ESXi hosts, NSX Manager and the vSphere Update Manager are working.
2. To determine if the problem is related to DNS, review the *esxupdate* logs and look for the message "`esxupdate: ERROR: MetadataDownloadError:IOError: <urlopen error [Errno –2] Name or service not known` in the *esxupdate.log* file.

   This message indicates that the ESXi host is unable to access the vCenter Server's Fully Qualified Domain Name (FQDN). For more information, see Verifying the VMware vCenter Server Managed IP Address (1008030).
3. Verify that Network Time Protocol (NTP) is configured correctly. VMware recommends configuring NTP. To determine whether NTP out of sync issues are impacting your environment, check the */etc/ntp.drift* file in the NSX Manager support bundles with version 6.2.4 and later.
4. Verify that all ports required for NSX for vSphere 6.x are not blocked by a firewall. For related information, refer to:
   - Network Port Requirements for VMware NSX for vSphere (2079386).
   - TCP and UDP Ports required to access VMware vCenter Server, VMware ESXi and ESX hosts, and other network components (1012382).

   Note VMware vSphere 6.x supports VIB downloads over port 443 (instead of port 80). This port is opened and closed dynamically. The intermediate devices between the ESXi hosts and vCenter Server must allow traffic using this port.
5. Verify that the vCenter Server Managed IP Address is configured correctly. For more information, see Verifying the VMware vCenter Server Managed IP Address (1008030).

6 Verify that the vSphere Update Manager is working correctly. Beginning with vCenter Server 6.0U3, NSX installation and upgrade procedures no longer leverage vSphere Update Manager with ESX Agent Manager. VMware strongly recommends running at least vCenter Server 6.0U3 or later. If you cannot upgrade, ensure that the vSphere Update Manager service is running. You can configure the vSphere Update Manager bypass option, as per KB 2053782.

7 If you specify non-default ports while deploying vCenter Server, ensure that these ports are not blocked by the ESXi host firewall.

8 Verify that vCenter Server *vpxd* process is listening on TCP port 8089. NSX Manager supports only the default port 8089.

About vSphere ESX Agent Manager (EAM)

vSphere ESX Agent Manager automates the process of deploying and managing NSX networking and security services, while extending the function of an ESXi host to provide additional services that a vSphere solution requires.

Logs and Services of the ESX Agent Manager

ESX Agent Manager logs are included as part of the vCenter log bundle.

- Windows—`C:\ProgramData\VMware\vCenterServer\logs\eam\eam.log`
- VCSA—`/var/log/vmware/vpx/eam.log`
- ESXi—`/var/log/esxupdate.log`

Monitoring ESX Agent Manager

Important Make sure to change the *bypassVumEnabled* flag to **`True`** before starting the NSX installation and change it back to **`False`** after the installation. See https://kb.vmware.com/kb/2053782.

To check the status of ESX Agent Manager:

1 Go to the vSphere Web Client.

2 Click **Administration > vCenter Server Extensions**, and then click the vSphere ESX Agent Manager.

   a Click the **Manage** tab.

   The **Manage** tab shows information about running agencies, lists any orphaned ESX agents, and logs information about the ESX agents that ESX Agent Manager manages.

   For more information about agents and agencies, see vSphere documentation.

   b Click the **Monitor** tab.

   The **Monitor > Events** tab shows information about the events associated with ESX Agent Manager.

Troubleshooting NSX Manager Issues

Validate that each troubleshooting step is true for your environment. Each step provides instructions to eliminate possible causes and take corrective action as necessary. The steps are ordered in the most appropriate sequence to isolate the issue and identify the proper resolution. Do not skip a step.

Problem

- Installing VMware NSX Manager fails.
- Upgrading VMware NSX Manager fails.
- Logging in to VMware NSX Manager fails.
- Accessing VMware NSX Manager fails.

Solution

1. Check the *NSX Release Notes* for current releases to see if the problem is resolved in a bug fix.
2. Ensure that the minimum system requirements are met when installing VMware NSX Manager.

   See the *NSX Installation Guide*.
3. Verify that all required ports are open in NSX Manager.

   See the *NSX Installation Guide*.
4. Installation issues:
   - If configuring the lookup service or vCenter Server fails, verify that the NSX Manager and lookup service appliances are in time sync. Use the same NTP server configurations on both NSX Manager and the lookup service. Also ensure that DNS is properly configured.
   - Verify that the OVA file is getting installed correctly. If an NSX OVA file cannot be installed, an error window in the vSphere client notes where the failure occurred. Also, verify and validate the MD5 checksum of the downloaded OVA/OVF file.
   - Verify that the time on the ESXi hosts is in sync with NSX Manager.
   - VMware recommends that you schedule a backup of the NSX Manager data immediately after installing NSX Manager.
5. Upgrade issues:
   - Before upgrading, see the latest interoperability information in the Product Interoperability Matrixes page.
   - VMware recommends that you back up your current configuration and download technical support logs before upgrading.
   - A force-resync with the vCenter Server may be required after the NSX Manager upgrade. To do this, log in to the NSX Manager Web Interface GUI. Then go to **Manage vCenter Registration > NSX Management Service > Edit** and re-enter the password for the administrative user.

6 Performance issues:

- Ensure that the minimum vCPU requirements are met.
- Verify that the root (/) partition has adequate space. You can verify this by logging in to the ESXi host and typing this command `df –h`.

  For example:

```
[root@esx-01a:~] df –h
Filesystem   Size   Used Available Use% Mounted on
NFS        111.4G  80.8G     30.5G  73% /vmfs/volumes/ds-site-a-nfs01
vfat       249.7M 172.2M     77.5M  69% /vmfs/volumes/68cb5875-d887b9c6-a805-65901f83f3d4
vfat       249.7M 167.7M     82.0M  67% /vmfs/volumes/fe84b77a-b2a8860f-38cf-168d5dfe66a5
vfat       285.8M 206.3M     79.6M  72% /vmfs/volumes/54de790f-05f8a633-2ad8-00505603302a
```

- Use the `esxtop` command to check which processes are using large amounts of CPU and memory.
- If the NSX Manager encounters any out-of-memory errors in the logs, verify that the `/common/dumps/java.hprof` file exists. If this file exists, create a copy of the file and include this with the NSX technical support log bundle.
- Verify that there are no storage latency issues in the environment.
- Attempt to migrate the NSX Manager to another ESXi host.

7 Connectivity issues:

- If NSX Manager is having connectivity issues either with vCenter Server or the ESXi host, log in to the NSX Manager CLI console, run the command: `debug connection IP_of_ESXi_or_VC`, and examine the output.
- Verify that the Virtual Center Web management services is started and the browser is not in an error state.
- If the NSX Manager Web User Interface (UI) is not updating, you can attempt to resolve the issue by disabling and then re-enabling the Web services. See https://kb.vmware.com/kb/2126701.
- Verify which port group and uplink NIC is used by the NSX Manager using the `esxtop` command on the ESXi host. For more information, see https://kb.vmware.com/kb/1003893.
- Attempt to migrate the NSX Manager to another ESXi host.
- Check the NSX Manager virtual machine appliance **Tasks and Events** tab from the vSphere Web Client under the **Monitor** tab.
- If the NSX Manager is having connectivity issues with vCenter Server, attempt to migrate the NSX Manager to the same ESXi host where the vCenter Server virtual machine is running to eliminate possible underlying physical network issues.

  Note that this only works if both virtual machines are on the same VLAN/port group.

Connecting NSX Manager to vCenter Server

A connection between the NSX Manager and the vCenter Server allows NSX Manager to use the vSphere API to perform functions such as deploy service VMs, prepare hosts, and create logical switch port groups. The connection process installs a web client plug-in for NSX on the Web Client Server.

For the connection to work, you must have DNS and NTP configured on NSX Manager, vCenter Server and the ESXi hosts. If you added ESXi hosts by name to the vSphere inventory, ensure that DNS servers have been configured on the NSX Manager and name resolution is working. Otherwise, NSX Manager cannot resolve the IP addresses. The NTP server must be specified so that the SSO server time and NSX Manager time are in sync. On NSX Manager, the drift file at `/etc/ntp.drift` is included in the tech Support bundle for NSX Manager.

The account you use to connect NSX Manager to vCenter Server must have the vCenter role "Administrator." Having the "Administrator" role enables NSX Manager to register itself with the Security Token Service server. When a particular user account is used to connect NSX Manager to vCenter, an “Enterprise Administrator" role for the user is also created on NSX Manager.

Common Issues Related to Connecting NSX Manager to vCenter Server

- DNS incorrectly configured on NSX Manager, vCenter Server, or an ESXi host.
- NTP incorrectly configured on NSX Manager, vCenter Server, or an ESXi host.
- User account without vCenter role of Administrator used to connect NSX Manager to vCenter.
- Network connectivity issues between NSX Manager and vCenter server.
- User logging into vCenter with an account that does not have a role on NSX Manager.

You need to initially log into vCenter with the account you used to link NSX Manager to vCenter Server. Then you can create additional users with roles on NSX Manager using the**Home > Networking & Security > NSX Managers > {IP of NSX Manager} > Manage > Users**.

The first login can take up to 4 minutes while vCenter loads and deploys NSX UI bundles.

Verify Connectivity from NSX Manager to vCenter Server

- Log in to the NSX Manager CLI console.
- To verify connectivity, view the ARP and routing tables.

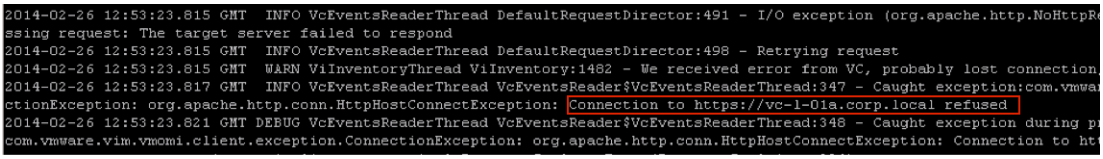
```
nsxmgr# show arp
IP address       HW type     Flags     HW address            Mask     Device
192.168.110.31   0x1         0x2       00:50:56:ae:ab:01     *        mgmt
192.168.110.2    0x1         0x2       00:50:56:01:20:a5     *        mgmt
192.168.110.1    0x1         0x2       00:50:56:01:20:a5     *        mgmt
192.168.110.33   0x1         0x2       00:50:56:ae:4f:7c     *        mgmt
192.168.110.32   0x1         0x2       00:50:56:ae:50:bf     *        mgmt
192.168.110.10   0x1         0x2       00:50:56:03:19:4e     *        mgmt
192.168.110.51   0x1         0x2       00:50:56:03:30:2a     *        mgmt
192.168.110.22   0x1         0x2       00:50:56:01:21:f9     *        mgmt
192.168.110.55   0x1         0x2       00:50:56:01:23:21     *        mgmt
```

```
192.168.110.26   0x1         0x2       00:50:56:01:21:ef     *         mgmt
192.168.110.54   0x1         0x2       00:50:56:01:22:ef     *         mgmt
192.168.110.52   0x1         0x2       00:50:56:03:30:16     *         mgmt
```

```
nsxmgr# show ip route
Codes: K – kernel route, C – connected, S – static,
       > – selected route, * – FIB route

S>* 0.0.0.0/0 [1/0] via 192.168.110.1, mgmt
C>* 192.168.110.0/24 is directly connected, mgmt
```

- Look for errors in the NSX Manager log to indicate the reason for not connecting to vCenter Server. The command to view the log is `show log manager follow`.



- Run the command: `debug connection IP_of_ESXi_or_VC`, and examine the output.

Perform Packet Capture on NSX Manager to View Connections

Use the debug packet command: `debug packet [capture|display] interface interface filter`

The interface name on NSX Manager is `mgmt`.

The filter syntax follows this form: "port_80_or_port_443"

The command runs in privileged mode only. To enter privileged mode, run the `enable` command and provide the admin password.

Packet capture example:

```
nsxmgr# en
nsxmgr# debug packet display interface mgmt port_80_or_port_443
tcpdump: verbose output suppressed, use –v or –vv for full protocol decode
listening on mgmt, link-type EN10MB (Ethernet), capture size 262144 bytes
23:40:25.321085 IP 192.168.210.15.54688 > 192.168.210.22.443: Flags [P.], seq 2645022162:2645022199,
ack 2668322748, win 244, options [nop,nop,TS val 1447550948 ecr 365097421], length 37
...
```

Verify Network Configuration on NSX Manager

The `show running-config` command shows the basic configuration of the management interface, NTP, and default route settings.

```
nsxmgr# show running-config
Building configuration...

Current configuration:
!
ntp server 192.168.110.1
```

```
!
ip name server 192.168.110.10
!
hostname nsxmgr
!
interface mgmt
 ip address 192.168.110.15/24
!
ip route 0.0.0.0/0 192.168.110.1
!
web-manager
```

NSX Manager Certificates

NSX Manager supports two ways to generate certificates.

- NSX Manager generated CSR: Limited functionality due to basic CSR
- PKCS#12: This is recommended for production

There is a known issue in which the CMS silently fails to make API calls.

This happens when the certificate issuer is not known to the caller because it is an untrusted root certificate authority or the certificate is self-signed. To resolve this issue, use a browser to navigate to the NSX Manager IP address or hostname and accept the certificate.

Secondary NSX Manager Stuck in Transit Mode

Use the solution described below if your secondary NSX Manager gets stuck in transit mode as described in the problem. The issue occurs when you restore the backup on primary NSX Manager when the secondary NSX Manager is in transit mode.

Problem

1 You have configured primary and secondary NSX Managers.

2 You take the backup of primary NSX Manager.

3 Later you remove the secondary NSX Manager. The secondary NSX Manager is in transit mode.

4 Now for some reasons, you restore the backup on primary NSX Manager.

5 In database, the transit NSX Manager gets updated as **Secondary**, but on UI it displays as **Transit**, and the sync fails.

6 You may not be able to remove the secondary NSX Manager, or promote it as a secondary again.

7 While promoting transit NSX Manager, an error message saying `NSX Manager node with IP address/hostname already exists` is displayed.

8 While removing transit NSX Manager, an error message saying `Incorrect user name or password` is displayed.

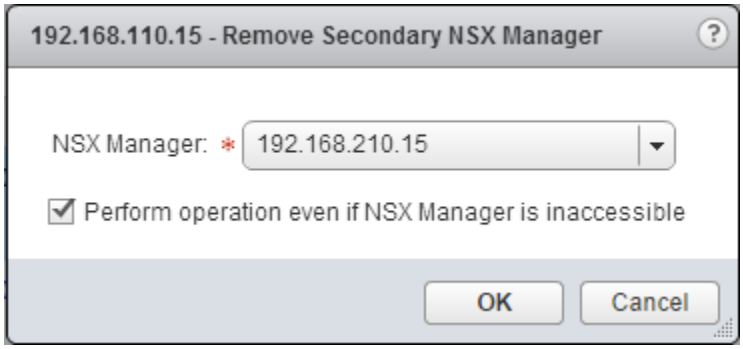
Solution

1 Log in to the vCenter linked to the primary NSX Manager using the vSphere Web Client.

2 Navigate to **Home > Networking & Security**> **Installation**, and then select **Management** tab.

3 Select the secondary NSX Manager that you want to delete and click **Actions**, and then click **Remove Secondary NSX Manager**.

A confirmation dialog box appears.



4 Select the **Perform operation even if NSX Manager is inaccessible** check box.

5 Click **OK**.

The secondary NSX Manager gets deleted from the primary database.

6 Add the secondary NSX Manager again.

What to do next

For more information about adding secondary NSX Manager, refer to *NSX Installation Guide*.

Configuring the NSX SSO Lookup Service Fails

Problem

- Registering NSX Manager to vCenter Server fails
- Configuring the SSO Lookup Service fails
- The following errors may appear:

```
nested exception is java.net.UnknownHostException: vc.local( vc.corp.local )
```

```
NSX Management Service operation failed.( Initialization of Admin Registration Service Provider failed. Root Cause: Error occurred while registration of lookup service, com.vmware.vim.sso.admin.exception.InternalError: General failure.
```

```
com.vmware.vshield.vsm.security.service.impl.SamlTokenSSOAuthenticator : SSO is not configured or initialized properly so cannot authenticate user.
```

Solution

1 Connectivity issues:

- If NSX Manager is having connectivity issues either with vCenter Server or the ESXi host, log in to the NSX Manager CLI console, run the command: `debug connection IP_of_ESXi_or_VC`, and examine the output.
- Ping from NSX Manager to the vCenter Server with the IP address and FQDN to check for routing, or static, or default route in NSX Manager, using this command:

  ```
  nsxmgr-l-01a# show ip route
  ```

  Codes:

  K – kernel route,

  C – connected,

  S – static

  > – selected route,

  * – FIB route

  ```
  S>* 0.0.0.0/0 [1/0] via 192.168.110.2, mgmt
  ```

  ```
  C>* 192.168.110.0/24 is directly connected, mgmt
  ```

2 DNS Issue

Ping from NSX Manager to vCenter Server with FQDN using this command:

```
nsx-mgr> ping vc-l-01a.corp.local
```

Output similar to the following example should appear:

```
nsx-mgr> ping vc-l-01a.corp.local
PING vc-1-01a.corp.local (192.168.110.51): 56 data bytes
64 bytes from 192.168.110.51: icmp_seq=0 ttl=64 time=1.749 ms
64 bytes from 192.168.110.51: icmp_seq=1 ttl=64 time=2.111 ms
64 bytes from 192.168.110.51: icmp_seq=2 ttl=64 time=8.082 ms
64 bytes from 192.168.110.51: icmp_seq=3 ttl=64 time=2.010 ms
64 bytes from 192.168.110.51: icmp_seq=4 ttl=64 time=0.857 ms
```

If this does not work, navigate to **Manage > Network > DNS Servers** in NSX Manager and ensure that DNS is properly configured.

3 Firewall Issue

If there is a firewall between NSX Manager and vCenter Server, verify that it allows SSL on `TCP/443`. Also, ping to check connectivity.

4 Verify that the following required ports are open in NSX Manager.

Table 2-1. NSX Manager Open Ports

Port	Required for
443/TCP	Downloading the OVA file on the ESXI host for deployment Using REST APIs Using the NSX Manager user interface
80/TCP	Initiating connection to the vSphere SDK Messaging between NSX Manager and NSX host modules
1234/TCP	Communication between NSX Controller and NSX Manager
5671	Rabbit MQ (messaging bus technology)
22/TCP	Console access (SSH) to CLI Note: By default, this port is closed

5 NTP Issues

Verify that time is synchronized between vCenter Server and NSX Manager. To achieve this, use the same NTP server configurations on the NSX Manager and vCenter Server.

To determine the time on the NSX Manager, run this command from the CLI:

```
nsxmgr-l-01a# show clock

Tue Nov 18 06:51:34 UTC 2014
```

To determine the time on the vCenter Server, run this command on the CLI:

vc-l-01a:~ # date

Output similar to the following should appear:

```
Tue Nov 18 06:51:31 UTC 2014
```

Note: After configuration of Time settings, restart the appliance.

6 User Permission Issues

Confirm that the user has **admin** privileges.

To register to vCenter Server or SSO Lookup Service, you must have administrative rights.

The default account is `administrator user: administrator@vsphere.local`

7 Reconnect to SSO by entering the credentials.

Logical Network Preparation: VXLAN Transport

NSX prepares the vSphere Distributed Switch that you select for VXLAN by creating a distributed virtual port group for the VTEP VMkernel NICs.

The teaming policy, load balancing method, MTU, and VLAN ID of the VTEPs are chosen during VXLAN configuration. The teaming and load balancing methods must match the configuration of the DVS selected for VXLAN.

The MTU must be set to be at least 1600 and not less than what is already configured on the DVS.

The number of VTEPs created depends on the teaming policy selected and the DVS configuration.

Common Issues During VXLAN Preparation

VXLAN preparation can fail for several reasons:

- Teaming method chosen for VXLAN does not match what can be supported by the DVS. To review supported methods, see the *VMware NSX for vSphere Network Virtualization Design Guide* at https://communities.vmware.com/docs/DOC-27683.
- Incorrect VLAN ID is chosen for the VTEPs.
- DHCP selected to assign VTEP IP addresses, but no DHCP server is available.
- A VMkernel NIC is missing. Resolve the error as described in VXLAN VMkernel NIC Out Of Sync.
- A VMkernel NIC has a bad IP address. Resolve the error as described in https://kb.vmware.com/kb/2137025.
- Incorrect MTU setting is chosen for the VTEPs. You should investigate if there is an MTU mismatch as described later in this topic.
- Incorrect VXLAN gateway is chosen. You should investigate if there is an error while configuring the VXLAN gateway as described later in this topic.

Important Port Numbers

The VXLAN UDP port is used for UDP encapsulation. Prior to NSX 6.2.3, the default VXLAN port number was 8472. In NSX 6.2.3 the default VXLAN port number changed to 4789 for new installations. In NSX 6.2 and later installations that use a hardware VTEP, you must use VXLAN port number 4789. For information on changing the VXLAN port configuration, see "Change VXLAN Port" in the *NSX Administration Guide*.

Control plane status displays as *disabled* if the host does not have any active VMs which need a controller connection

Use the `show logical-switch` commands to view VXLAN details on the host. For details, refer to *NSX Command Line Interface Reference*.

The `show logical-switch host hostID verbose` command will display status of control plane as *disabled* if the host has not been populated with any VMs which require a connection to the controller cluster for forwarding table information.

```
Network count:  18
VXLAN network:  32003
Multicast IP:   0.0.0.0
Control plane:  Disabled  <<========
MAC entry count:        0
ARP entry count:        0
Port count:     1
```

Error while configuring VXLAN gateway

When configuring VXLAN using a static IP pool at **Networking & Security > Installation > Host Preparation > Configure VXLAN** and the configuration fails to set an IP pool gateway on the VTEP, the VXLAN configuration status enters the Error (RED) state for the host cluster. The error message is "`VXLAN Gateway cannot be set on host`" and the error status is "`VXLAN_GATEWAY_SETUP_FAILURE`".

In the REST API call, `GET https://<nsxmgr-ip>/api/2.0/nwfabric/status?resource=<cluster-moid>`, the status of VXLAN is as follows:

```
<nwFabricFeatureStatus>
<featureId>com.vmware.vshield.nsxmgr.vxlan</featureId>
  <featureVersion>5.5</featureVersion>
  <updateAvailable>false</updateAvailable>
  <status>RED</status>
  <message>VXLAN Gateway cannot be set on host</message>
  <installed>true</installed>
  <enabled>true</enabled>
  <errorStatus>VXLAN_GATEWAY_SETUP_FAILURE</errorStatus>
</nwFabricFeatureStatus>
```

Workaround: To fix the error, there are two options.

- `Option 1`: Remove VXLAN configuration for the host cluster, fix the underlying gateway setup in the IP pool by making sure the gateway is properly configured and reachable, and then reconfigure VXLAN for the host cluster.
- `Option 2`: Perform the following steps.
  a. Fix the underlying gateway setup in the IP pool by making sure the gateway is properly configured and reachable.
  b. Put the host (or hosts) into maintenance mode to ensure no VM traffic is active on the host.
  c. Delete the VXLAN VTEPs from the host.
  d. Take the host out of maintenance mode. Taking hosts out of maintenance mode triggers the VXLAN VTEP creation process on NSX Manager. NSX Manager will try to re-create the required VTEPs on the host.

Investigate an MTU mismatch

- Run the following command to verify if the MTU is configured to 1600 or above:

  `ping ++netstack=vxlan -d -s 1572 -I` <vmkx hostname_or_IP>

  where *vmkx* is the ID of your VMkernel port and *hostname_or_IP* is the IP or hostname of the VMkernel port.

  This allows you to check the validity of all uplinks. If you are working in a multi-VTEP environment, you can validate all uplinks by running the ping command from each possible VTEP VMkernel source/destination interface to validate all the paths.

- Check the physical infrastructure. Many times issue gets resolved by a configuration change to the physical infrastructure.
- Determine whether the issue is confined to a single logical switch, or other logical switches are also affected. Verify if the issue affects all the logical switches.

For more information about the MTU check, see "Verify the NSX Working State" in the *NSX Upgrade Guide*.

VXLAN VMkernel NIC Out Of Sync

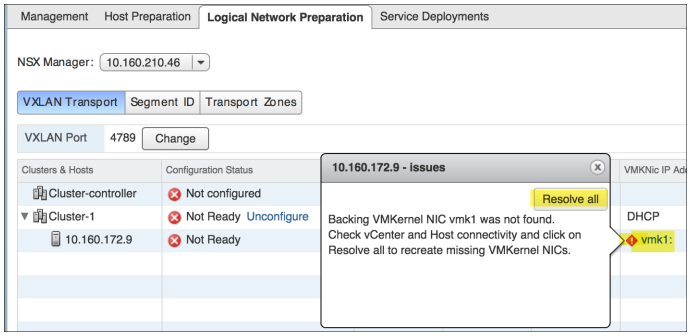
When the VMkernel NIC is deleted on the host, but the VMkernel NIC information is still available in NSX, then NSX Manager indicates the deleted VMkernel NIC with an **Error** icon.

Prerequisites

VMkernel NIC is deleted on the host.

Procedure

1. In the vSphere Web Client, navigate to **Networking & Security > Installation** > **Logical Network Preparation**.
2. On the **VXLAN Transport** tab, expand the Cluster and Hosts.



3. Click the **Error** icon to view with information of the VMkernel NIC which is deleted on the host.
4. Click the **Resolve All** button to recreate the deleted VMkernel NIC on the host.

Results

The deleted VMkernel NIC is recreated on the host.

Changing the VXLAN Teaming Policy and MTU Settings

The VXLAN teaming policy and MTU settings can be changed on prepared hosts and clusters, but the changes apply only when preparing new hosts and clusters for VXLAN. Existing virtual port groups for VTEP VMkernel can be changed only by manually preparing the hosts and clusters again. You can change the teaming policy and MTU settings using API.

Problem

Incorrect MTU setting is chosen for the VTEPs.

Solution

1. Retrieve information about all the VXLAN prepared switches using the `GET https://<NSX-Manager-IP-Address>/api/2.0/vdn/switches` API.

   In the output of the API , locate the switch that you would like to modify and note the name. For example, *dvs-35*.

2. Now query with the specific vSphere Distributed Switch that you noted earlier.

   For example, `GET https://<NSX-Manager-IP-Address>/api/2.0/vdn/switches/dvs-35` API.

   Output similar to the following example should appear:

```
<vdsContext>
<switch>
 <objectId>dvs-35</objectId>
 <objectTypeName>VmwareDistributedVirtualSwitch</objectTypeName>
 <vsmUuid>423A993F-BEE6-1285-58F1-54E48D508D90</vsmUuid>
 <nodeId>916287b3-761d-430b-8ab2-83878dfe3e7f</nodeId>
 <revision>6</revision>
 <type>
  <typeName>VmwareDistributedVirtualSwitch</typeName>
 </type>
 < name>vds-site-a</name>
 <scope>
  <id>datacenter-21</id>
  <objectTypeName>Datacenter</objectTypeName>
  < name>Datacenter Site A</name>
 </scope>
 <clientHandle/>
 <extendedAttributes/>
 <isUniversal>false</isUniversal>
 <universalRevision>0</universalRevision>
</switch>
<mtu>1600</mtu>
```

```
<teaming>FAILOVER_ORDER</teaming>
<uplinkPortName>Uplink 4</uplinkPortName>
<promiscuousMode>false</promiscuousMode
</vdsContext>
```

3 You can modify the parameters such as teaming policy and/or MTU on a vSphere Distributed Switch using the API call. The following example shows changing the teaming policy of *dvs-35* from *FAILOVER_ORDER* to *LOADBALANCE_SRCMAC* and MTU from *1600* to *9000* .

- For NSX: `PUT https://<NSX-Manager-IP-Address>/api/2.0/vdn/switches`

Output similar to the following example should appear:

```
<vdsContext>
<switch>
 <objectId>dvs-35</objectId>
 <objectTypeName>VmwareDistributedVirtualSwitch</objectTypeName>
 <vsmUuid>423A993F-BEE6-1285-58F1-54E48D508D90</vsmUuid>
 <nodeId>916287b3-761d-430b-8ab2-83878dfe3e7f</nodeId>
 <revision>6</revision>
 <type>
  <typeName>VmwareDistributedVirtualSwitch</typeName>
 </type>
 <name>vds-site-a</name>
 <scope>
 <id>datacenter-21</id>
 <objectTypeName>Datacenter</objectTypeName>
 <name>Datacenter Site A</name>
 </scope>
 <clientHandle/>
 <extendedAttributes/>
 <isUniversal>false</isUniversal>
 <universalRevision>0</universalRevision>
</switch>
 <mtu>9000</mtu>
```

```
  <teaming>LOADBALANCE_SRCMAC</teaming>
  <uplinkPortName>Uplink 4</uplinkPortName>
  <promiscuousMode>false</promiscuousMode>
</vdsContext>
```

Note Following is a list of valid teaming policy entries for the *<teaming>* parameter:

- FAILOVER_ORDER
- ETHER_CHANNEL
- LACP_ACTIVE
- LACP_PASSIVE
- LOADBALANCE_LOADBASED
- LOADBALANCE_SRCID
- LOADBALANCE_SRCMAC LACP_V2

4 Verify the syntax used is correct and the change is active for the vSphere Distributed Switch you are working with using the `GET` command. For example, `GET https://<NSX-Manager-IP-Address>/api/2.0/vdn/switches/dvs-35`.

5 Open the vSphere Web Client and confirm that the configuration changes are reflected.

Logical Switch Port Group Out Of Sync

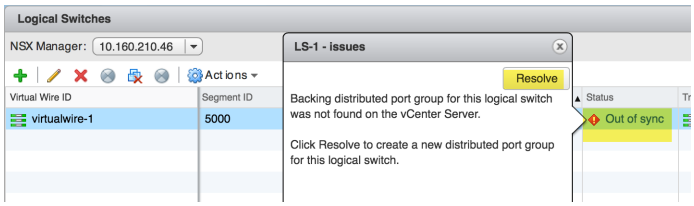
If the backup distributed virtual port group (DVPG) of the logical switch is deleted on the vCenter Server, then the Status column of the **Logical Switches** page displays **Out of sync** status.

Prerequisites

DVPG of the logical switch is deleted on vCenter Server.

Procedure

1 In the vSphere Web Client, navigate to **Home > Networking & Security** > **Logical Switches**.



2 In the Status column, click the **Out of sync** link to see the detailed reason for this out of sync state.

3 Click the **Resolve** button to resolve the issue.

Results

This invokes API to recreate the backup DVPG.

Troubleshooting NSX Routing

3

NSX has two types of routing subsystems, optimised for two key needs.

The NSX routing subsystems are:

- Routing within the logical space, also known as “East – West” routing, provided by the Distributed Logical Router (DLR);
- Routing between the physical and logical space, also known as “North – South” routing, provided by the Edge Services Gateways (ESG).

Both provide options for horizontal scaling.

You can scale-out distributed E-W routing via the DLR.

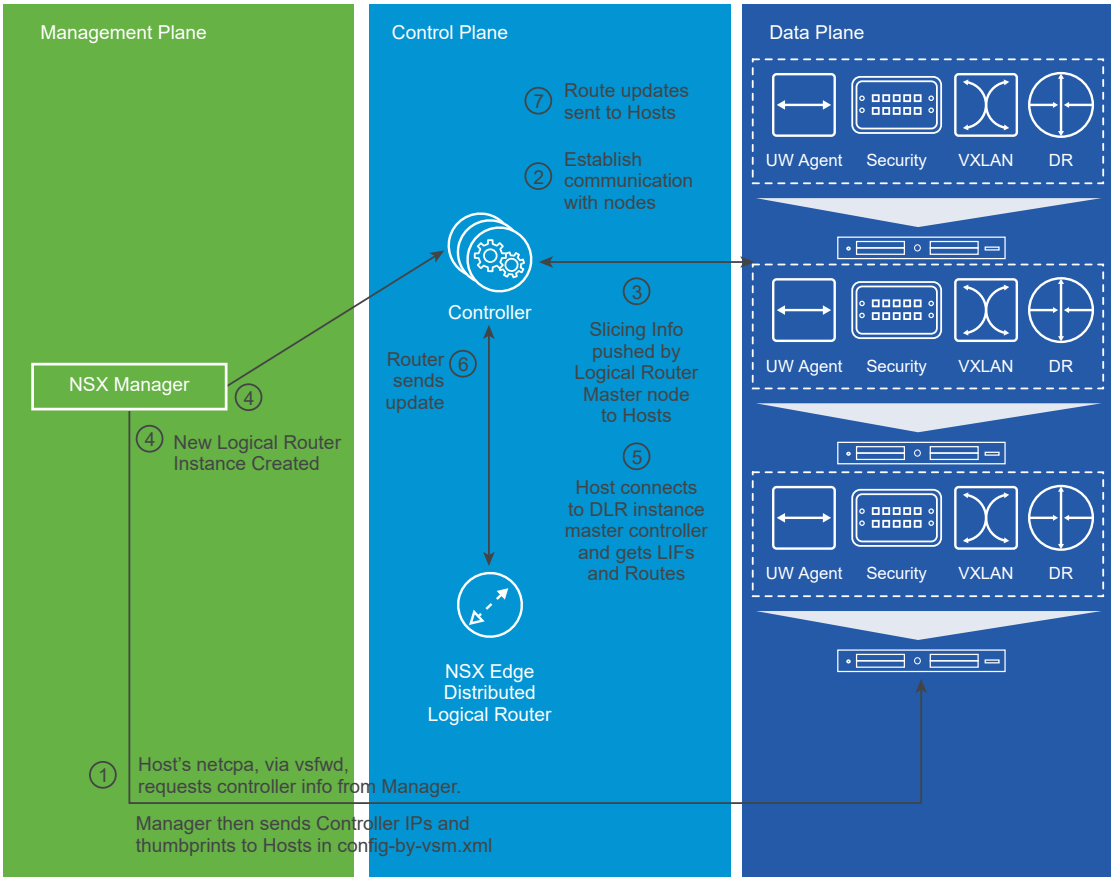
The DLR supports running a single dynamic routing protocol at a time (OSPF or BGP), while the ESG supports running both routing protocols at the same time. The reason for this is the DLR is designed to be s a “stub” router, with a single path out, which means more advanced routing configurations are typically not required.

Both the DLR and the ESG support having a combination of static and dynamic routes.

Both the DLR and the ESG support ECMP routes.

Both provide L3 domain separation, meaning that each instance of a Distributed Logical Router or an Edge Services Gateway has its own L3 configuration, similar to an L3VPN VRF.

Figure 3-1. The Creation of a DLR



This chapter includes the following topics:

- Understanding the Distributed Logical Router
- Understanding Routing Provided by the Edge Services Gateway
- ECMP Packet Flow
- NSX Routing: Prerequisites and Considerations
- DLR and ESG UIs
- New NSX Edge (DLR)
- Typical ESG and DLR UI Operations
- Troubleshooting NSX Routing

Understanding the Distributed Logical Router

The DLR is optimised for forwarding in the logical space between VMs, on VXLAN-backed or VLAN-backed portgroups.

The DLR has the following properties:

- High performance, low overhead first-hop routing:

- Scales linearly with the number of hosts
- Supports 8-way ECMP on uplink
- Up to 1,000 DLR instances per host
- Up to 999 logical interfaces (LIFs) on each DLR (8 x uplink + 991 internal) + 1 x management
- Up to 10,000 LIFs per host distributed across all DLR instances (not enforced by NSX Manager)

Keep in mind the following caveats:

- Cannot connect more than one DLR to any given VLAN or VXLAN.
- Cannot run more than one routing protocol on each DLR.
- If OSPF is used, cannot run it on more than one DLR uplink.
- To route between VXLAN and VLAN, the transport zone must span single DVS.

The DLR's design at a high level is analogous to a modular router chassis, in the following ways:

- ESXi hosts are like line cards:
  - They have ports with connected end stations (VMs).
  - This is where the forwarding decisions are made.
- The DLR Control VM is like a Route Processor Engine:
  - It runs dynamic routing protocols to exchange routing information with the rest of the network.
  - It computes forwarding tables for "line cards" based on the configuration of interfaces, static routes, and dynamic routing information.
  - It programs these forwarding tables into the "line cards" (via the Controller Cluster, to enable scale and resiliency).
- The physical network connecting ESXi hosts together is like a backplane:
  - It carries VLAN-encapsulated or VXLAN-encapsulated data between the "line cards."

High-Level DLR Packet Flow

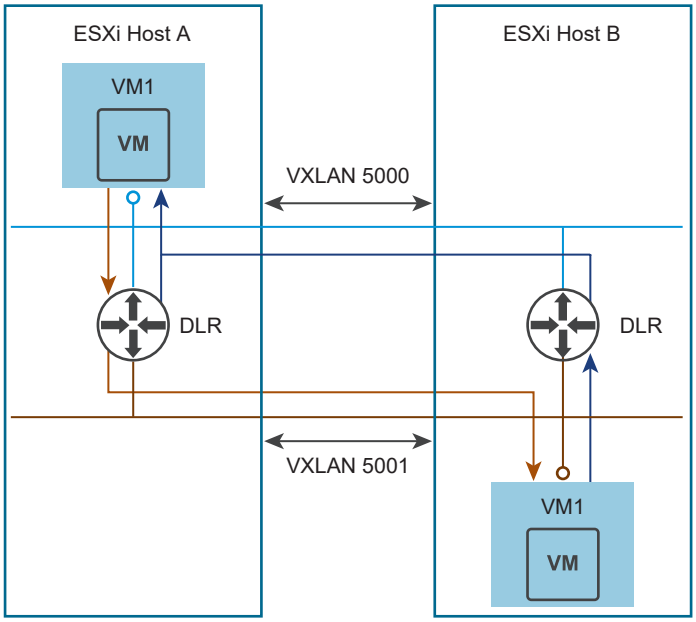
Each ESXi host has its own copy of each configured DLR instance. Each DLR instance has its own unique set of tables containing the information needed to forward packets. This information is synchronized across all hosts where this DLR instance exists. Instances of an individual DLR across different hosts have exactly the same information.

Routing is always handled by a DLR instance on the same host where the source VM is running. This means that when source and destination VMs are on different hosts, the DLR instance that provides routing between them sees packets only in one direction, from source VM to destination. Return traffic is only seen by the corresponding instance of the same DLR on the destination VM's host.

After the DLR has completed routing, delivery to the final destination is the responsibility of the DVS via L2 – VXLAN or VLAN if the source and destination VMs are on different hosts, or by the DVS locally if they are on the same host.

Figure 3-2. High-Level DLR Packet Flow illustrates data flow between two VMs, VM1 and VM2, running on different hosts and connected to two different Logical Switches, VXLAN 5000 and VXLAN 5001.

Figure 3-2. High-Level DLR Packet Flow



Packet flow (skipping ARP resolution):

1. VM1 sends a packet toward VM2, which is addressed to VM1's gateway for VM2's subnet (or default). This gateway is a VXLAN 5000 LIF on the DLR.
2. The DVS on ESXi Host A delivers the packet to the DLR on that host, where the lookup is performed, and the egress LIF is determined (in this case – VXLAN 5001 LIF).
3. The packet is then sent out of that destination LIF, which essentially returns the packet to the DVS, but on a different Logical Switch (5001).
4. The DVS then performs L2 delivery of that packet to the destination host (ESXi Host B), where the DVS will forward the packet to VM2.

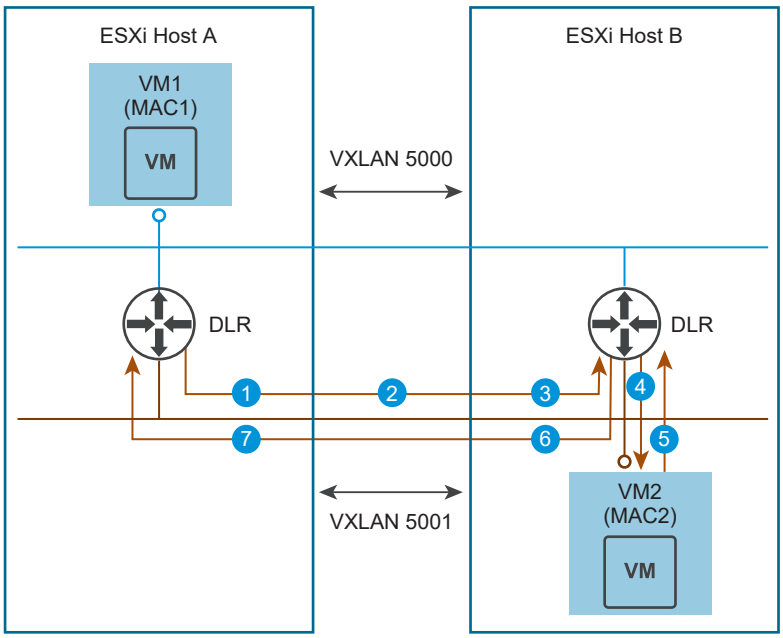
Return traffic will follow in the same order, where traffic from VM2 is forwarded to the DLR instance on ESXi Host B, and then delivered via L2 on VXLAN 5000.

DLR ARP Resolution Process

Before traffic from VM1 can reach VM2, the DLR needs to learn VM2's MAC address. After learning VM2's MAC address, the DLR can create the correct L2 headers for the outbound packets.

Figure 3-3. DLR ARP Process shows the DLR's ARP resolution process.

Figure 3-3. DLR ARP Process



To learn the MAC address, the DLR follows these steps:

1 The DLR instance on Host A generates an ARP request packet, with SRC MAC = vMAC, and DST MAC = Broadcast. The VXLAN module on Host A finds all VTEPs on the egress VXLAN 5001, and sends each one a copy of that broadcast frame.

2 As the frame leaves the host via the VXLAN encapsulation process, the SRC MAC is changed from vMAC to pMAC A, so that return traffic can find the originating DLR instance on Host A. Frame now is SRC MAC = pMAC A, and DST MAC = Broadcast.

3 As the frame is received and decapsulated on Host B, it is examined and found to be sourced from the IP address that matches the local DLR instance's LIF on VXLAN 5001. This flags the frame as abrequest to perform the proxy ARP function. The DST MAC is changed from Broadcast to vMAC so that the frame can reach the local DLR instance.

4 The local DLR instance on Host B receives the ARP Request frame, SRC MAC = pMAC A, DST MAC = vMAC, and sees its own LIF IP address requesting this. It saves the SRC MAC, and generates a new ARP Request packet, SRC MAC = vMAC, DST MAC = Broadcast. This frame is tagged as "DVS Local" to prevent it from being flooded via the dvUplink. The DVS delivers the frame to VM2.

5 VM2 sends an ARP Reply, SRC MAC = MAC2, DST MAC = vMAC. The DVS delivers it to the local DLR instance.

6 The DLR instance on Host B replaces DST MAC with the pMAC A saved at from step 4, and sends the packet back to the DVS for delivery back to Host A.

7 After the ARP Reply reaches Host A, DST MAC is changed to vMAC, and the ARP Reply frame with SRC MAC = MAC2 and DST MAC = vMAC reaches the DLR instance on Host A.

The ARP resolution process is complete, and the DLR instance on Host A can now start sending traffic to VM2.

DLR ARP Suppression

Address Resolution Protocol (ARP) suppression is a technique used to reduce the amount of ARP broadcast flooding within individual VXLAN segments, that is between VMs connected to the same logical switch.

When VM1 wants to know the MAC address for VM2, it sends an ARP request. This ARP request is intercepted by the logical switch and if logical switch already has an ARP entry for the target, it sends the ARP response to the VM.

If not, it sends an ARP query to the NSX Controller. If controller knows VM IP to MAC binding, controller replies with the binding and the logical switch sends the ARP response. If controller does not have the ARP entry, then the ARP request is re-broadcasted on the logical switch. NSX Controller learns the MAC address via Switch Security module which snoops on ARP requests/DHCP packets.

ARP suppression has been extended to include the Distributed logical router (DLR) as well.

- ARP requests from distributed logical router are treated the same way as ARP requests from other VMs and are subjected to suppression. When distributed logical router has to resolve ARP request of a destination IP, the ARP request is suppressed by the logical switch, preventing flooding when the IP to MAC binding is already known to the controller.
- When a LIF is created, distributed logical router adds the ARP entry for the LIF IP in the logical switch, so ARP requests for the LIF IP are also suppressed by the logical switch.

Understanding Routing Provided by the Edge Services Gateway

The second subsystem of NSX Routing is provided by Edge Services Gateway.

The ESG is essentially a router in a virtual machine. It is delivered in an appliance-like form factor with four sizes, with its complete lifecycle managed by the NSX Manager. The ESG's primary use case is as a perimeter router, where it is deployed between multiple DLRs and between the physical world and the virtualized network.

The ESG has the following properties:

- Each ESG can have up to 10 vNIC interfaces, or 200 trunk sub-interfaces.
- Each ESG supports 8-way ECMP for path redundancy and scalability.

ECMP Packet Flow

Suppose two ESGs are deployed to provide a DLR instance with 2-way ECMP uplinks with the physical environment.

Figure 3-4. High-Level ESG and DLR Packet Flow with ECMP shows the ESG and DLR packet flow when equal-cost multipath (ECMP) routing is enabled between two ESGs and the physical infrastructure.

VM1 thus has access to 2x bi-directional throughput compared with a deployment with a single ESG.

VM1 is connected to a Logical Switch with the VNI 5000.

The DLR has two LIFs – Internal on VNI 5000, and Uplink on VNI 5001.

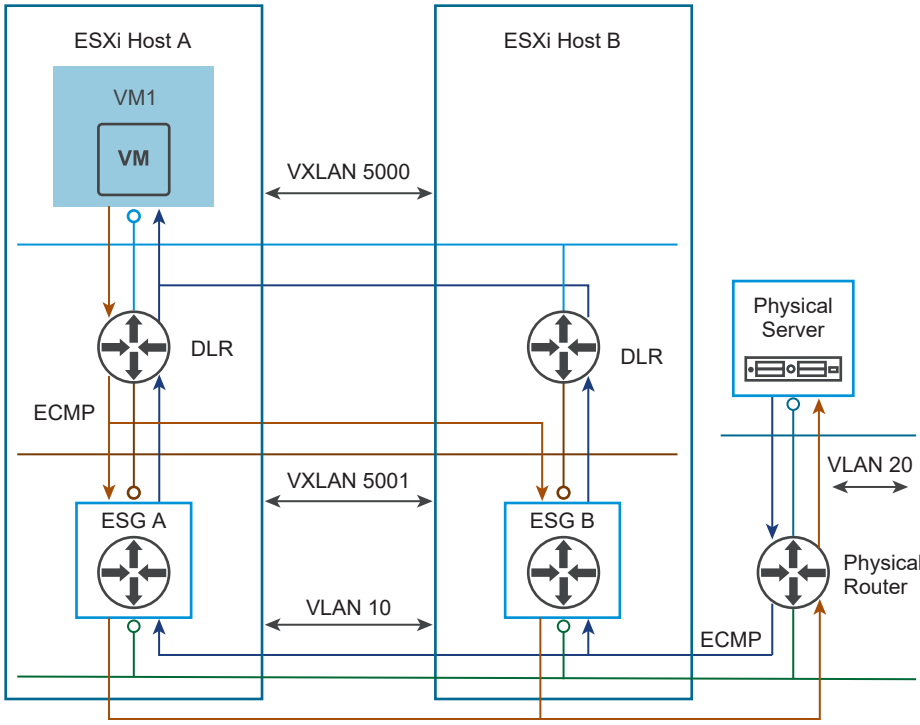
The DLR has ECMP enabled and is receiving equal cost routes toward the IP subnet of VLAN 20 from a pair of ESGs, ESG A and ESG B via a dynamic routing protocol (BGP or OSPF).

The two ESGs are connected to a VLAN-backed dvPortgroup associated with VLAN 10, where a physical router that provides connectivity to VLAN 20 is also connected.

The ESGs receive external routes for VLAN 20, via a dynamic routing protocol from the physical router.

The physical router in exchange learns about the IP subnet associated with VXLAN 5000 from both ESGs, and performs ECMP load balancing for the traffic toward VMs in that subnet.

Figure 3-4. High-Level ESG and DLR Packet Flow with ECMP



The DLR can receive up to eight equal-cost routes and balance traffic across the routes. ESG A and ESG B in the diagram provide two equal-cost routes.

ESGs can do ECMP routing toward the physical network, assuming multiple physical routers are present. For simplicity, the diagram shows a single physical router.

There is no need for ECMP to be configured on ESGs toward the DLR, because all DLR LIFs are “local” on the same host where ESG resides. There would be no additional benefit provided by configuring multiple uplink interfaces on a DLR.

In situations where more North-South bandwidth is required, multiple ESGs can be placed on different ESXi hosts to scale up to ~80Gbps with 8 x ESGs.

The ECMP packet flow (not including ARP resolution):

1 VM1 sends a packet to the physical server, which is sent to VM1's IP gateway (which is a DLR LIF) on ESXi Host A.

2 The DLR performs a route lookup for the IP of the physical server, and finds that it is not directly connected, but matches two ECMP routes received from ESG A and ESG B.

3 The DLR calculates an ECMP hash, and decides on a next hop, which could be either ESG A or ESG B, and sends the packet out the VXLAN 5001 LIF.

4 The DVS delivers the packet to the selected ESG.

5 The ESG performs the routing lookup and finds that the physical server's subnet is accessible via the physical router's IP address on VLAN 10, which is directly connected to one of ESG's interfaces.

6 The packet is sent out through the DVS, which passes it on to the physical network after tagging it with the correct 801.Q tag with VLAN ID 10.

7 The packet travels through the physical switching infrastructure to reach the physical router, which performs a lookup to find that the physical server is directly connected to an interface on VLAN 20.

8 The physical router sends the packet to the physical server.

On the way back:

1 The physical server sends the packet to VM1, with the physical router as the next hop.

2 The physical router performs a lookup for VM1's subnet, and sees two equal-cost paths to that subnet with the next hops, ESG A's and ESG B's VLAN 10 interface, respectively.

3 The physical router selects one of the paths and sends the packet toward the corresponding ESG.

4 The physical network delivers the packet to the ESXi host where the ESG resides, and delivers it to DVS, which decapsulates the packet and forwards it on the dvPortgroup associated with VLAN 10 to the ESG.

5 The ESG performs a routing lookup and finds that VM1's subnet is accessible via its interface associated with VXLAN 5001 with the next hop being DLR's uplink interface IP address.

6 The ESG sends the packet to the DLR instance on the same host as the ESG.

7 The DLR performs a routing lookup to find that VM1 is available via its VXLAN 5000 LIF.

8 The DLR sends the packet out its VXLAN 5000 LIF to the DVS, which performs the final delivery.

NSX Routing: Prerequisites and Considerations

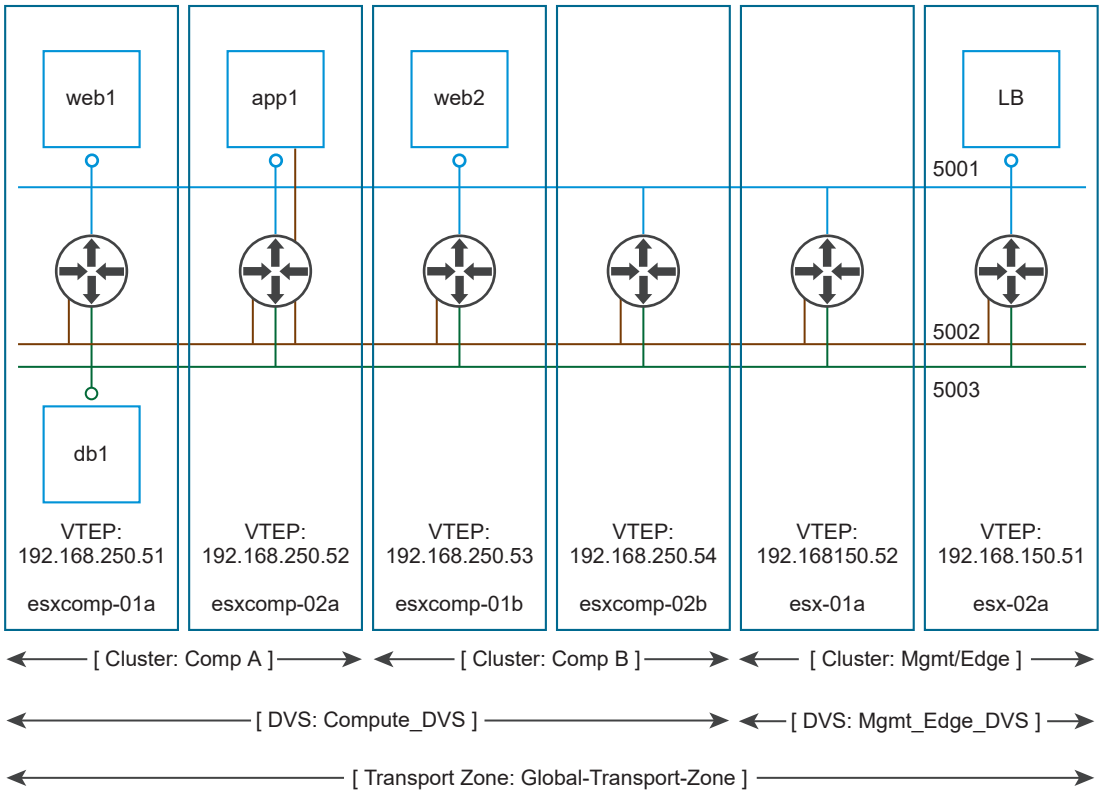
The DLR and the ESG rely on the DVS to provide L2 forwarding services for dvPortgroups (both VXLAN and VLAN based) for end-to end connectivity to work.

This means L2 that forwarding services that are connected to DLR or ESG must be configured and operational. In the NSX installation process, these services are provided by "Host Preparation" and "Logical Network Preparation."

When creating transport zones on multi-cluster DVS configurations, make sure that all clusters in the selected DVS are included under the transport zone. This ensures that the DLR is available on all clusters where DVS dvPortgroups are available.

When a transport zone is aligned with DVS boundary, the DLR instance is created correctly.

Figure 3-5. Transport Zone Correctly Aligned to DVS Boundary



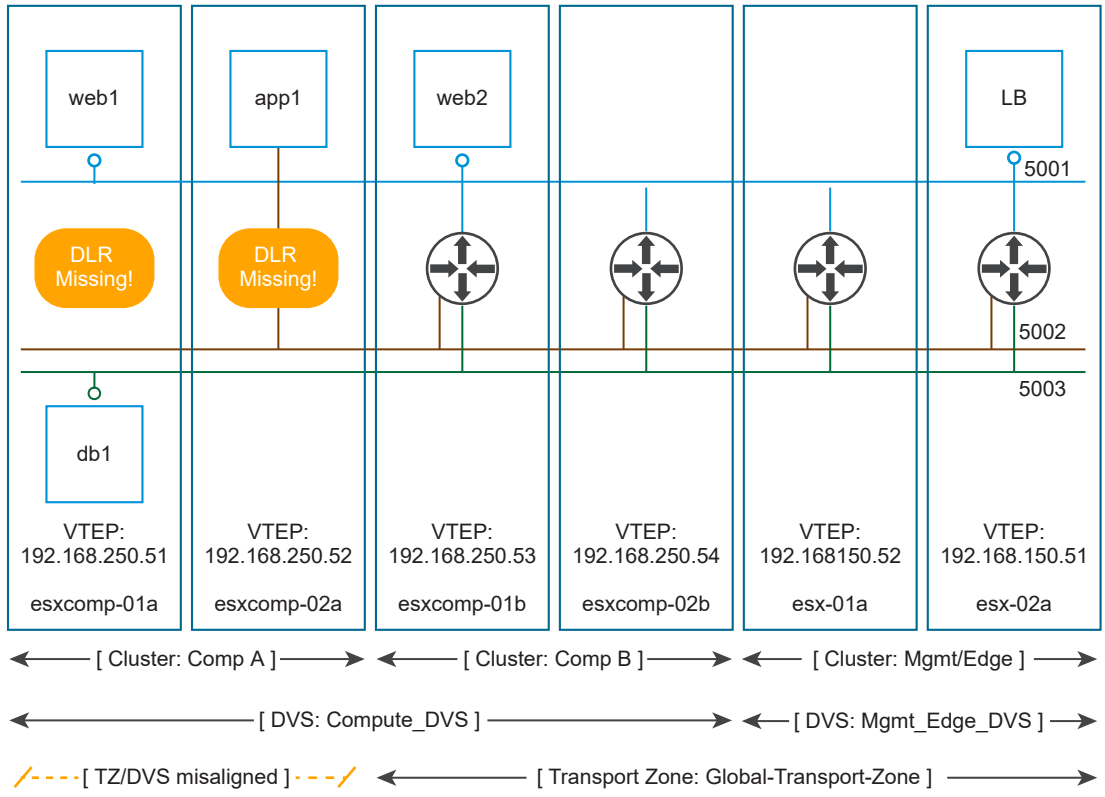
When a transport zone is not aligned to the DVS boundary, the scope of logical switches (5001, 5002 and 5003) and the DLR instances that these logical switches are connected to becomes disjointed, causing VMs in cluster Comp A to have no access to DLR LIFs.

In the diagram above, DVS “Compute_DVS” covers two clusters, “Comp A” and “Comp B”. The “Global-Transport-Zone” includes both “Comp A” and “Comp B.”

This results in correct alignment between the scope of Logical Switches (5001, 5002, and 5003), and the DLR instance created on all hosts in all clusters where these Logical Switches are present.

Now, let’s look at an alternative situation, where the Transport Zone was not configured to include cluster “Comp A”:

Figure 3-6. Transport Zone Misaligned with DVS Boundary



In this case, VMs running on cluster “Comp A” have full access to all logical switches. This is because logical switches are represented by dvPortgoups on hosts, and dvProtgroups are a DVS-wide construct. In our sample environment, “Compute_DVS” covers both “Comp A” an “Comp B."

DLR instances, however, are created in strict alignment with the transport zone scope, which means no DLR instance will be created on hosts in “Comp A."

As the result, VM “web1” will be able to reach VMs “web2” and “LB” because they are on the same logical switch, but VMs “app1” and “db1” will not be able to communicate with anything.

The DLR relies on the Controller Cluster to function, while the ESG does not. Make sure that the Controller Cluster is up and available before creating or changing a DLR configuration.

If the DLR is to be connected to VLAN dvPortgroups, ensure that ESXi hosts with the DLR configured can reach each other on UDP/6999 for DLR VLAN-based ARP proxy to work.

Considerations:

- A given DLR instance cannot be connected to logical switches that exist in different transport zones. This is to ensure all logical switches and DLR instances are aligned.
- The DLR cannot be connected to VLAN-backed portgroups, if that DLR is connected to logical switches spanning more than one DVS. As above, this is to ensure correct alignment of DLR instances with logical switches and dvPortgroups across hosts.

- When selecting placement of the DLR Control VM, avoid placing it on the same host as one or more of its upstream ESGs by using DRS anti-affinity rules if they are in the same cluster. This is to reduce the impact of host failure on DLR forwarding.
- OSPF can be enabled only on a single Uplink (but supports multiple adjacencies). BGP, on other hand, can be enabled on multiple Uplink interfaces, where it is necessary.

DLR and ESG UIs

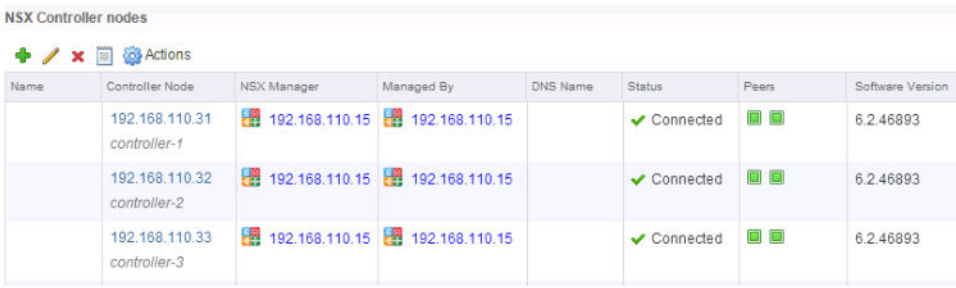
The DLR and ESG UIs provide indicators of the system working state.

NSX Routing UI

The vSphere Web Client UI provides two major sections relevant to NSX routing.

These include the L2 and control-plane infrastructure dependencies and the routing subsystem configuration.

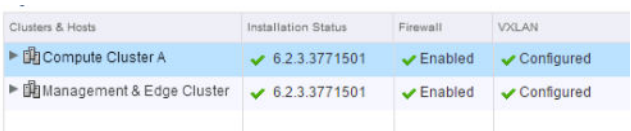
NSX distributed routing requires functions that are provided by the Controller Cluster. The following screen shot shows a Controller Cluster in a healthy state.



Things to note:

- There are three controllers deployed.
- The “Status” for all controllers is “Connected”.
- The software version for all controllers is the same.
- Each controller node has two peers.

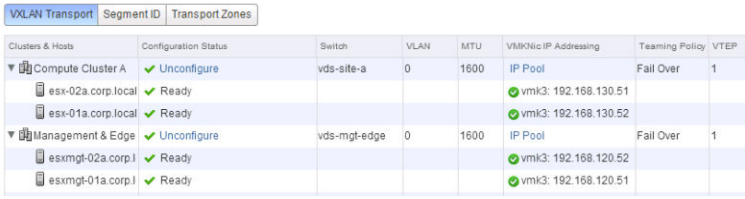
Host kernel modules for distributed routing are installed and configured as part of VXLAN configuration on the host. This means distributed routing requires that ESXi hosts are prepared and VXLAN is configured on them.



Things to note:

- “Installation Status” is green.
- “VXLAN” is “Configured."

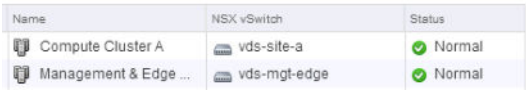
Makes sure that VXLAN transport components are correctly configured.



Things to note:

- The VLAN ID must be correct for the VTEP transport VLAN. Note that in the screen shot above it is “0." In most real-world deployments this would not the case.
- MTU is configured to be 1600 or larger. Make sure that the MTU is not set to 9000 with the expectation that the MTU on VMs would be also set to 9000. The DVS maximum MTU is 9000, and if VMs are also at 9000, there is no space for VXLAN headers.
- VMKNics must have the correct addresses. Make sure that they are not set to 169.254.x.x addresses, indicating that nodes have failed to get addresses from DHCP.
- The teaming policy must be consistent for all cluster members of the same DVS.
- The number of VTEPs must be the same as the number of dvUplinks. Make sure that valid/expected IP addresses are listed.

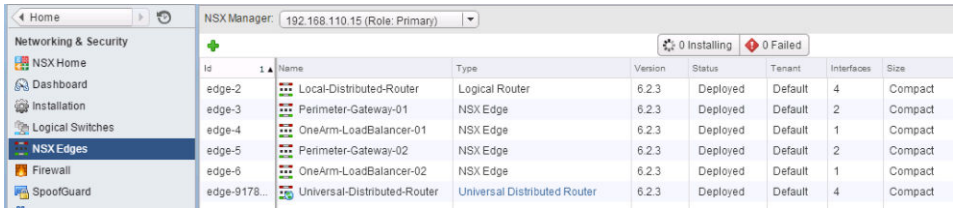
Transport Zones have to be correctly aligned to DVS boundaries, to avoid the situation in which the DLR is missing on some clusters.



NSX Edges UI

The NSX routing subsystem is configured and managed in the “NSX Edges” section of the UI.

When this part of the UI is selected, the following view appears.



All currently deployed DLRs and ESGs are shown, with the following information displayed for each:

- “Id” shows the ESG or DLR Edge appliance ID, which can be used for any API calls referring to that ESG or DLR
- “Tenant” + “Id” forms the DLR instance name. This name is visible and used in the NSX CLI.
- “Size” is always “Compact” for DLR, and the size that was selected by the operator for ESG.

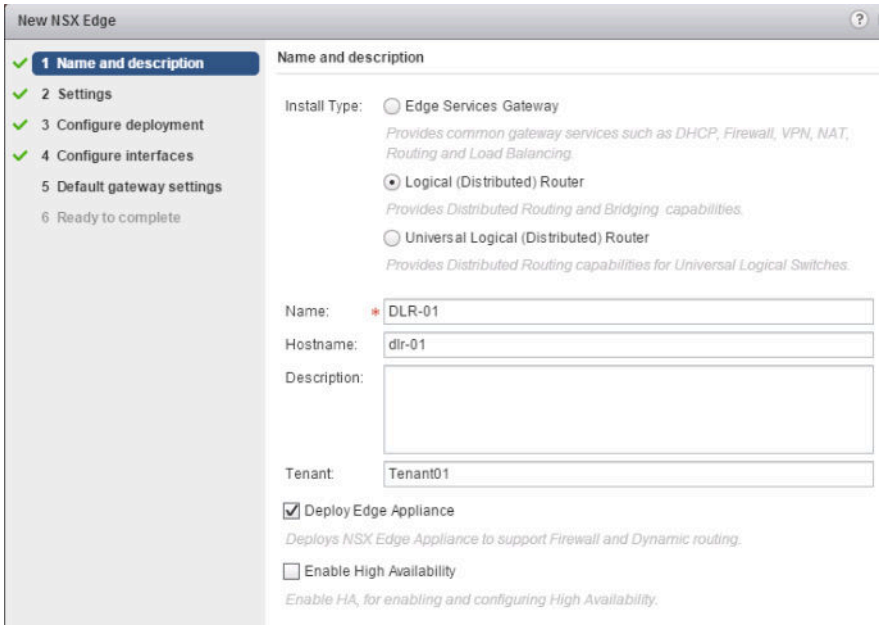
In addition to the information in the table, there is a context menu, accessible either via buttons or via “Actions."

Table 3-1. NSX Edge Context Menu

Icon	Action
	“Force Sync” operation clears the ESG’s or the DLR's Control VM’s configuration, reboots it, and re-pushes the configuration.
	“Redeploy” tears down the ESG or DLR, and creates is a new ESG or DLR with the same configuration. The existing ID is preserved.
	“Change Auto Rule Configuration” applies to the ESG’s built-in firewall rules, created when services are enabled on the ESG (for example, BGP which needs TCP/179).
	“Download tech support logs” creates a log bundle from the ESG or DLR Control VM For the DLR, host logs are not included in the tech support bundle and need to be collected separately.
	“Change appliance size” is only applicable to ESGs. This will perform a “redeploy” with a new appliance (vNIC MAC addresses will change).
	“Change CLI credentials” allows the operator to force-update the CLI credentials. If the CLI is locked-out on an ESG or DLR Control VM after 5 failed logins, this will not lift the lock-out. You will need to wait 5 minutes, or “Redeploy” your ESG/DLR to get back in with the correct credentials.
	“Change Log Level” changes the level of detail to be sent to ESG/DLR syslog.
	“Configure Advanced Debugging” re-deploys the ESG or DLR with core-dump enabled and additional virtual disk attached for storing core dump files.
	“Deploy” becomes available when an ESG has been created without deploying it. This option simply executes the deployment steps (deploys OVF, configures Interfaces, pushes configuration to the created appliance.
	If the version of DLR/ESG is older than NSX Manager, the “Upgrade Version” option becomes available.
	“Filter” can search for ESGs/DLRs by “Name.”

New NSX Edge (DLR)

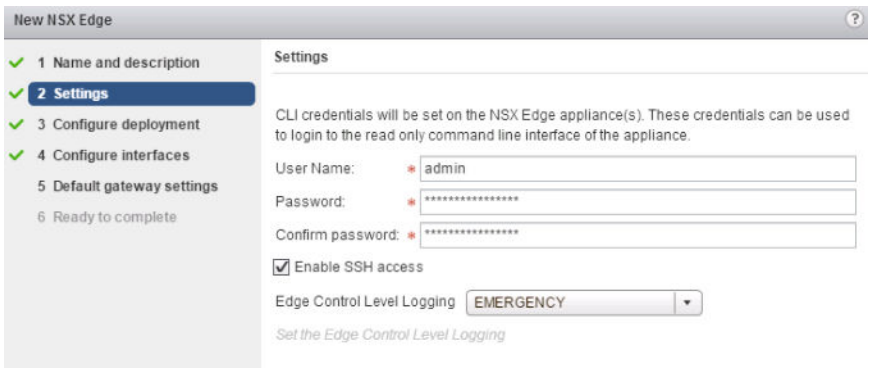
When an operator creates a new DLR, the following wizard is used to collect the necessary information.



On the “Name and Description” screen, the following information is collected:

- “Name” will appear in the “NSX Edges” UI.
- “Hostname” will be used to set the DNS name of the ESG or DLR Control VM, visible on SSH/ Console session, in syslog messages, and in the vCenter “Summary” page for the ESG/DLR VM under “DNS Name.”
- “Description” is in the UI showing the list of NSX Edges.
- “Tenant” will be used to form the DLR Instance Name, used by the NSX CLI. It can be also be used by external cloud management platform.

On the “Settings” screen:



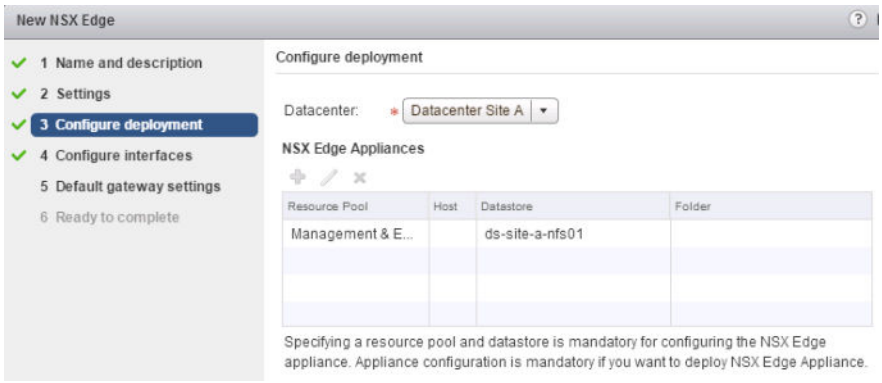
- “User Name” and “Password” set the CLI/VM console credentials to access the DLR Control VM. NSX does not support AAA on ESG or DLR Control VMs. This account has full rights to ESG/DLR Control VMs; however, the ESG/DLR configuration cannot be changed via the CLI/VMconsole.
- “Enable SSH access” enables the SSH daemon on the DLR Control VM to start.
  - The control VM Firewall rules need to be adjusted to allow SSH network access.

- The operator can connect to the DLR Control VM from either a host on the subnet of the Control VM's management Interface, or without such restriction on the OSPF/BGP "Protocol Address," if a protocol address is configured.

  Note It is not possible to have network connectivity between the DLR Control VM and any IP address that falls into any subnet configured on any of that DLR's "Internal" interfaces. This is because the egress interface for these subnets on DLR Control VM points to the pseudo-interface "VDR," which is not connected to the data plane.

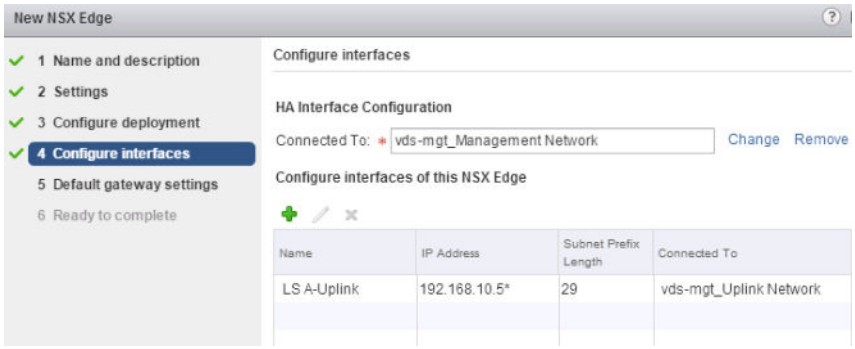
- "Enable HA" deploys Control VM as an Active/Standby HA pair.
- "Edge Control Level Logging" sets the syslog level on the Edge appliance.

On the "Configure deployment" screen:



- "Datacenter" selects the vCenter datacenter in which to deploy the Control VM.
- "NSX Edge Appliances" refers to the DLR Control VM and allows definition of exactly one (as shown).
  - If "HA" is enabled, the Standby Edge will be deployed on the same cluster, host, and datastore. A DRS "Separate Virtual Machines" rule will be created for the Active and Standby DLR Control VMs.

On the "Configure Interfaces" screen:

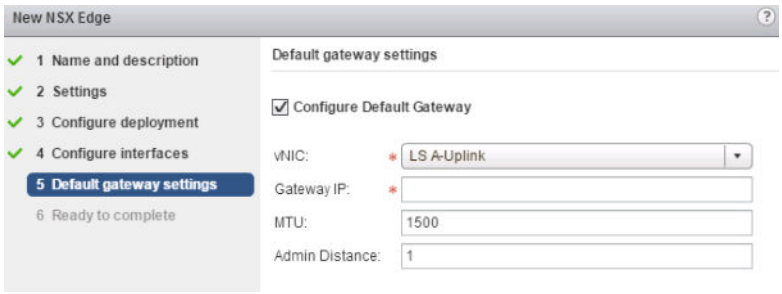


- "HA Interface"
  - Is not created as a DLR logical interface capable of routing. It is only a vNIC on the Control VM.

- This interface does not require an IP address, because NSX manages the DLR configuration via VMCI.
- This interface is used for HA heartbeat if the DLR "Enable High Availability" is checked on the "Name and description" screen.

- "Interfaces of this NSX Edge" refer to DLR Logical Interfaces (LIFs)
  - The DLR provides L3 gateway services to VMs on the "Connected To" dvPortgroup or logical switch with IP addresses from corresponding subnets.
  - "Uplink" type LIFs are created as vNICs on the Control VM, so, up to eight are supported; the last two available vNICs are allocated to the HA interface and one reserved vNIC.
  - An "Uplink" type LIF is required for dynamic routing to work on the DLR.
  - And "Internal" type LIFs are created as pseudo-vNICs on the Control VM, and it is possible to have up to 991 of them.

On the "Default gateway settings" screen:



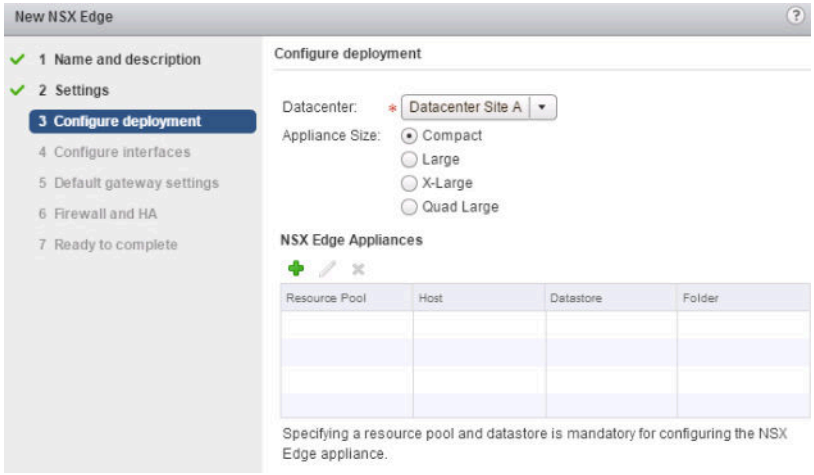
- Configure Default Gateway, if selected, will create a static default route on the DLR. This option is available if an "Uplink" type LIF is created in the previous screen.
- If ECMP is used on the uplink, the recommendation is to leave this option disabled, to prevent dataplane outage in case of next-hop failure.

Note The double right-arrow in the top right corner allows for "suspending" the wizard in progress so that it can be resumed at a later time.

ESG and DLR Differences

There are some differences between the wizard screens when an ESG is deployed, compared to a DLR.

The first one is on the "Configure deployment" screen:



For an ESG, “Configure Deployment” allows selection of the Edge size. If an ESG is used only for routing, “Large” is a typical size that is suitable in most scenarios. Selecting a larger size will not provide more CPU resources to the ESG’s routing processes, and will not lead to more throughput.

It is also possible to create an ESG without deploying it, which still requires configuration of an Edge Appliance.

A “Non-deployed” Edge can be later deployed via an API call or with the “Deploy” UI action.

If Edge HA is selected, you must create at least one “Internal” interface, or HA will fail silently, leading to the “split-brain” scenario.

The NSX UI and API allow an operator to remove the last “Internal” interface, which will cause HA to silently fail.

Typical ESG and DLR UI Operations

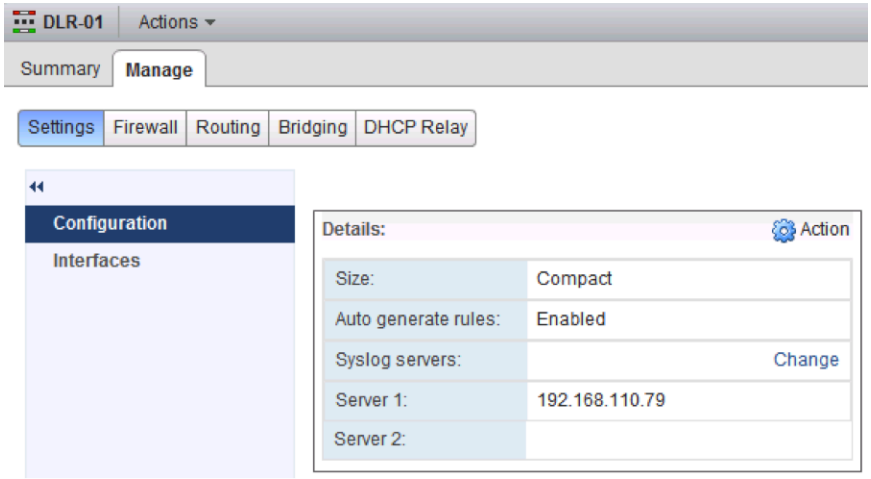
In addition to creation, there are several configuration operations that are typically executed after initial deployment.

These include:

- Syslog configuration
- Management of static routes
- Configuration of routing protocols and route redistribution

Syslog Configuration

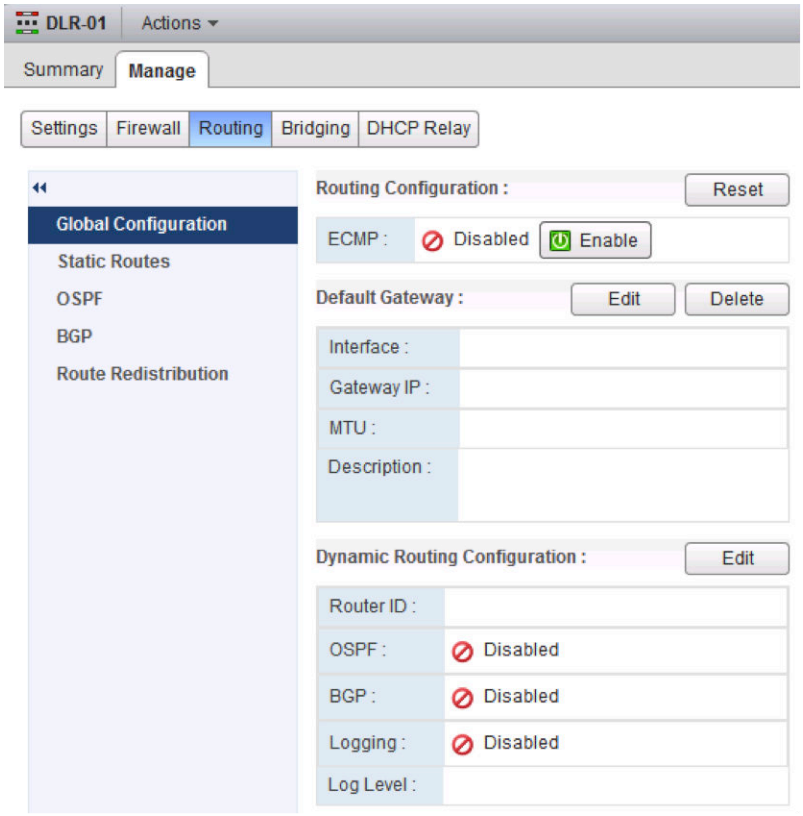
Configure the ESG or DLR Control VM to send log entries to a remote syslog server.



Notes:

- The syslog server must be configured as an IP address, because the ESG/DLR Control VM does not get configured with a DNS resolver.
  - In the ESG's case, it is possible to "Enable DNS Service" (DNS proxy) that ESG itself will be able to use to resolve DNS names, but generally specifying syslog server as an IP address in a more reliable method with fewer dependencies.
- There is no way to specify a syslog port in the UI (it is always 514), but protocol (UDP/TCP) can be specified.
- Syslog messages originate from the IP address of the Edge's interface that is selected as egress for the syslog server's IP by the Edge's forwarding table.
  - For the DLR, the syslog server's IP address cannot be on any subnets configured on any of the DLR's "Internal" interfaces. This is because the egress interface for these subnets on the DLR Control VM points to the pseudo-interface "VDR," which is not connected to the data plane.

By default, logging for the ESG/DLR routing engine is disabled. If required, enable it via UI by clicking "Edit" for the "Dynamic Routing Configuration."

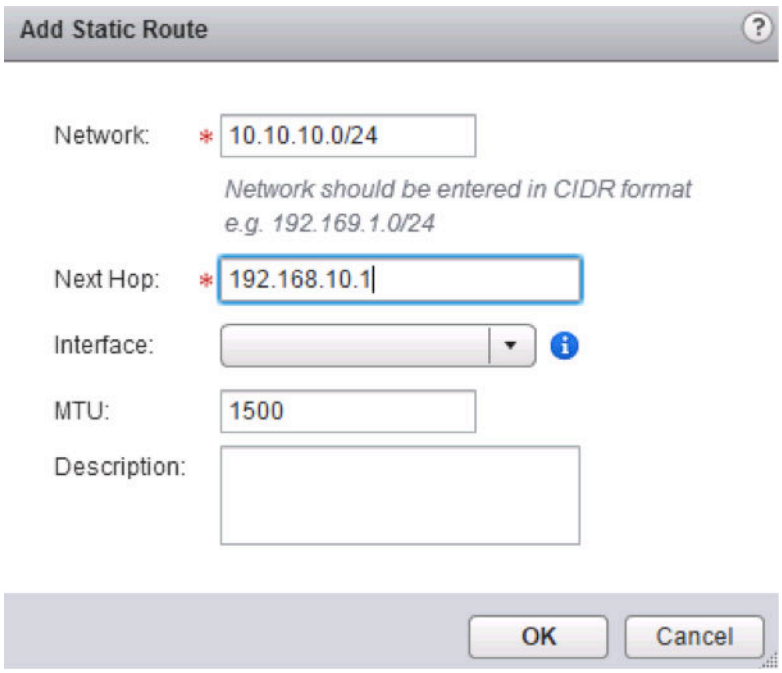


You must also configure the Router ID, which will typically be the IP address of the Uplink interface.

Static Routes

Static routes must have the next hop set to an IPaddress on a subnet associated with one of DLR's LIFs or ESG's Interfaces. Otherwise, configuration fails.

"Interface," if not selected, is set automatically by matching the next hop to one of directly connected subnets.

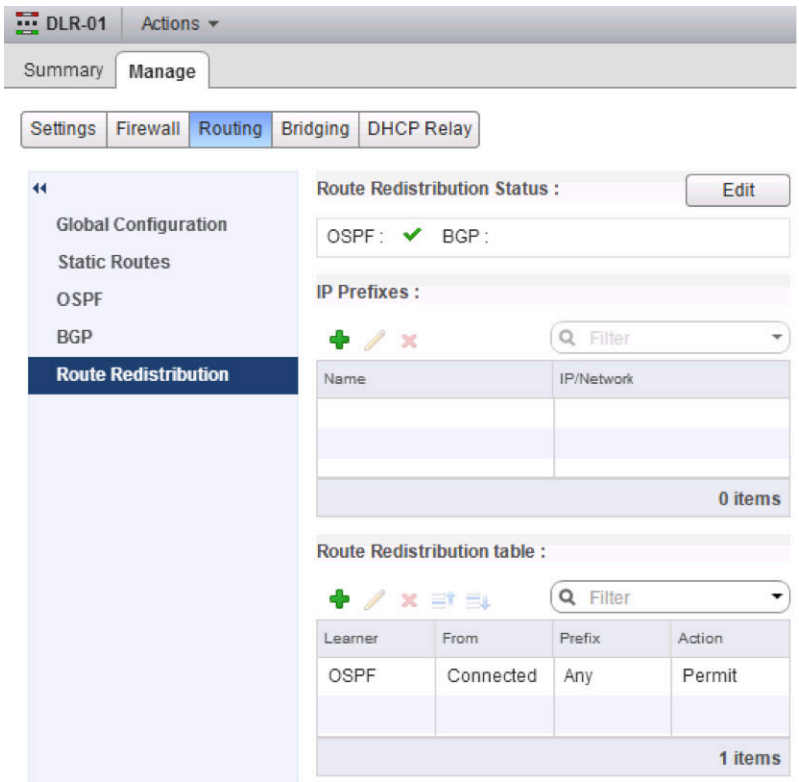


Route Redistribution

Adding an entry into the “Route Redistribution table” does not automatically enable redistribution for the selected “Learner Protocol." This must be done explicitly via “Edit” for “Route Redistribution Status."

The DLR is configured with redistribution of connected routes into OSPF by default, while ESG is not.

The “Route Redistribution table” is processed in top-to-bottom order, and processing is stopped after the first match. To exclude some prefixes from redistribution, include more specific entries at the top.



Troubleshooting NSX Routing

NSX provides multiple tools for making sure that routing is working.

NSX Routing CLI

There is a collection of CLI commands that allow an operator to examine the running state of various parts of the NSX routing subsystem.

Due to the distributed nature of the the NSX routing subsystem, there are a number of CLIs available, accessible on various components of NSX. Starting in NSX version 6.2, NSX also has a centralized CLI that helps reduce the “travel time” required to access and log in to various distributed components. It provides access to most of the information from a single location: the NSX Manager shell.

Checking the Prerequisites

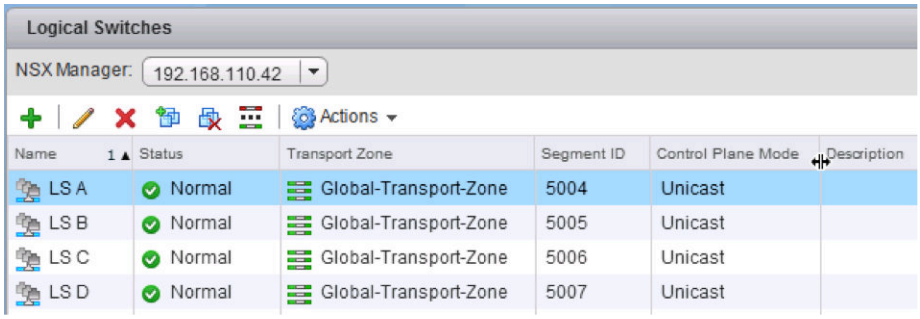
There are two major prerequisites that must be satisfied for each ESXi host:

- Any logical switches connected to the DLR are healthy.
- The ESXi host has been successfully prepared for VXLAN.

Logical Switch Health Check

NSX Routing works in conjunction with NSX logical switching. To verify that the logical switches connected to a DLR are healthy:

- Find the segment ID (VXLAN VNI) for each logical switch connected to the DLR in question (for example, 5004..5007).



- On the ESXi hosts where VMs served by this DLR are running, check the state of the VXLAN control plane for the logical switches connected to this DLR.

```
# esxcli network vswitch dvs vmware vxlan network list --vds-name=Compute_VDS
VXLAN ID  Multicast IP               Control Plane                       Controller Connection  Port
Count  MAC Entry Count  ARP Entry Count
--------  ------------------------  ----------------------------------  ---------------------
----------  ----------------  ----------------
    5004  N/A (headend replication)  Enabled (multicast proxy,ARP proxy)  192.168.110.201
(up)            2                 2                 0
    5005  N/A (headend replication)  Enabled (multicast proxy,ARP proxy)  192.168.110.202
(up)            1                 0                 0
    5006  N/A (headend replication)  Enabled (multicast proxy,ARP proxy)  192.168.110.203
(up)            1                 1                 0
    5007  N/A (headend replication)  Enabled (multicast proxy,ARP proxy)  192.168.110.202
(up)            1                 0                 0
```

Check the following for each relevant VXLAN:

- For logical switches in hybrid or unicast mode:
  - Control Plane is “Enabled.”
  - “multicast proxy” and “ARP proxy” are listed; “ARP proxy” will be listed even if you disabled IP Discovery.
  - A valid Controller IP address is listed under “Controller,” and “Connection” is “up.”
- “Port Count” looks right – there will be at least 1, even if there are no VMs on that host connected to the logical switch in question. This one port is the vdrPort, a special dvPort connected to the DLR kernel module on the ESXi host.

- Run the following command to make sure that the vdrPort is connected to each of the relevant VXLANs.

```
~ # esxcli network vswitch dvs vmware vxlan network port list --vds-name=Compute_VDS --vxlan-id=5004
Switch Port ID  VDS Port ID  VMKNIC ID
--------------  -----------  ---------
      50331656  53                   0
      50331650  vdrPort              0


~ # esxcli network vswitch dvs vmware vxlan network port list --vds-name=Compute_VDS --vxlan-id=5005
Switch Port ID  VDS Port ID  VMKNIC ID
--------------  -----------  ---------
      50331650  vdrPort              0
```

- In the example above, VXLAN 5004 has one VM and one DLR connection, while VXLAN 5005 only has a DLR connection.
- Check whether the appropriate VMs have been properly wired to their corresponding VXLANs, for example web-sv-01a on VXLAN 5004.

```
~ # esxcfg-vswitch -l
DVS Name         Num Ports   Used Ports  Configured Ports  MTU     Uplinks
Compute_VDS      1536        10          512               1600    vmnic0

  DVPort ID           In Use      Client
[..skipped..]
  53                  1           web-sv-01a.eth0
```

VXLAN Preparation Check

As part of VXLAN configuration of an ESXi host, the DLR kernel module is also installed, configured, and connected to a dvPort on a DVS prepared for VXLAN.

1 Run `show cluster all` to get the cluster ID.

2 Run `show cluster cluster-id` to get the host ID.

3 Run `show logical-router host hostID connection` to get the status information.

```
nsxmgr-01a# show logical-router host <hostID> connection

Connection Information:
-----------------------

DvsName           VdrPort           NumLifs  VdrVmac
--------          --------          -------  --------
Compute_VDS       vdrPort           4        02:50:56:56:44:52
    Teaming Policy: Default Teaming
    Uplink   : dvUplink1(50331650): 00:50:56:eb:41:d7(Team member)
```

```
    Stats : Pkt Dropped      Pkt Replaced     Pkt Skipped
    Input : 0                0                1968734458
   Output : 303              7799             31891126
```

- A DVS enabled with VXLAN will have one vdrPort created, shared by all DLR instances on that ESXi host.
- “NumLifs” refers to the number that is the sum of LIFs from all DLR instances that exist on this host.
- “VdrVmac” is the vMAC that the DLR uses on all LIFs across all instances. This MAC is the same on all hosts. It is never seen in any frames that travel the physical network outside of ESXi hosts.
- For each dvUplink of DVS enabled with VXLAN, there is a matching VTEP; except in cases where LACP / Etherchannel teaming mode is used, when only one VTEP is created irrespective of the number of dvUplinks.
  - Traffic routed by the DLR (SRC MAC = vMAC) when leaving the host will get the SRC MAC changed to pMAC of a corresponding dvUplink.
  - Note that the original VM’s source port or source MAC is used to determine the dvUplink (it is preserved for each packet in its DVS's metadata).
  - When there are multiple VTEPs on the host and one of dvUplinks fails, the VTEP associated with the failed dvUplink will be moved to one of the remaining dvUplinks, along with all VMs that were pinned to that VTEP. This is done to avoid flooding control plane changes that would be associated with moving VMs to a different VTEP.
- The number in “()” next to each “dvUplinkX” is the dvPort number. It is useful for packet capture on the individual uplink.
- The MAC address shown for each “dvUplinkX” is a “pMAC” associated with that dvUplink. This MAC address is used for traffic sourced from the DLR, such as ARP queries generated by the DLR and any packets that have been routed by the DLR when these packets leave the ESXi host. This MAC address can be seen on the physical network (directly, if DLR LIF is VLAN type, or inside VXLAN packets for VXLAN LIFs).
- Pkt Dropped / Replaced / Skipped refer to counters related to internal implementation details of the DLR, and are not typically used for troubleshooting or monitoring.

Brief Recap of Routing

To effectively troubleshoot routing issues, it is helpful to review how routing works and the related information tables.

1 Receive a packet to send to a destination IP address.

2 Check the routing table and determine the IP address of the next hop.

3 Determine which of your network interfaces can reach it.

4 Get a MAC address of that next hop (via ARP).

5 Build an L2 frame.

6 Send the frame out the interface.

So to do routing, you need:

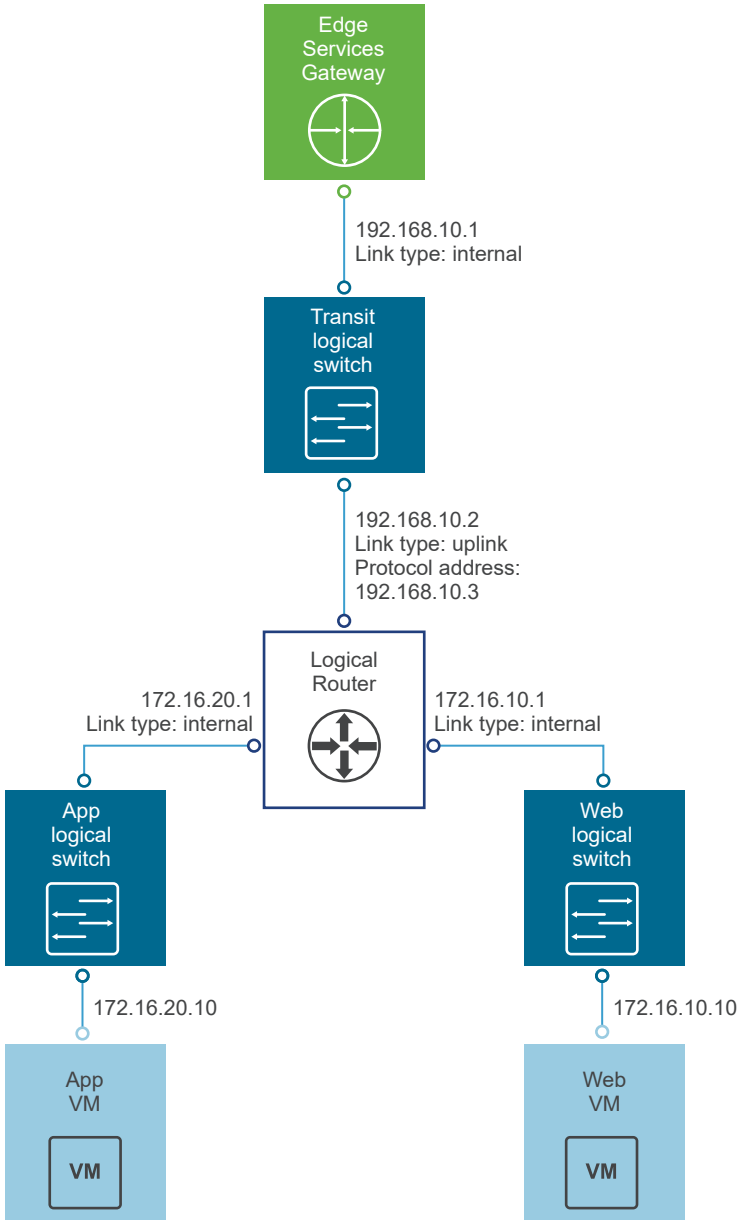
- An interface table (with interface IP addresses and netmasks)
- A routing table
- An ARP table

Verifying the DLR State Using a Sample Routed Topology

This section discusses how to verify the information that the DLR requires to route packets.

Let’s take a sample routed topology and create a set of logical switches and a DLR to create it in NSX.

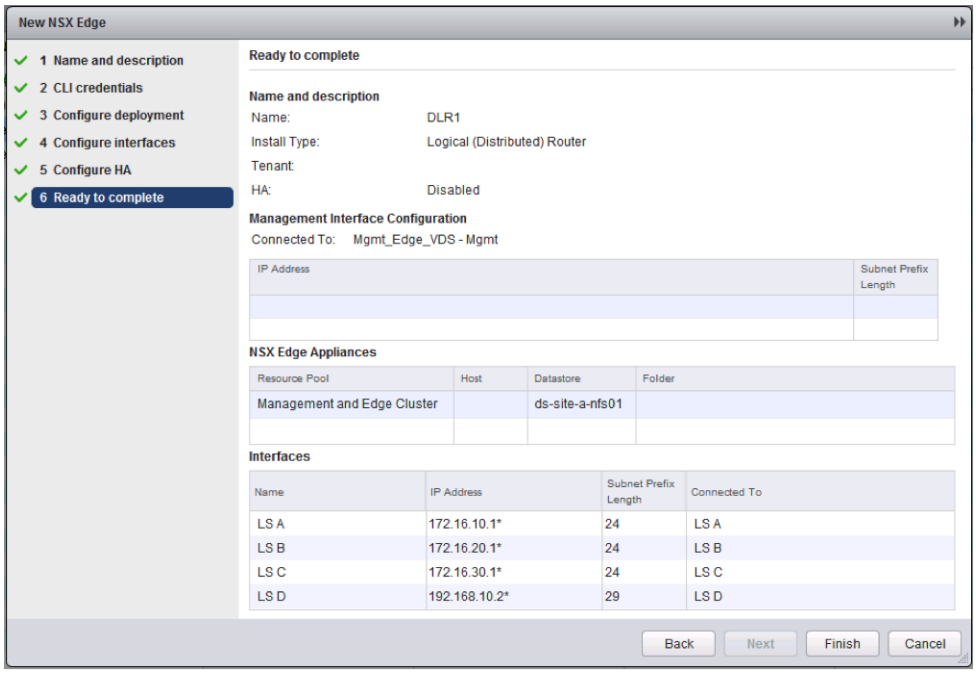
Figure 3-7. Sample Routed Topology



The diagram shows:

- 4 x Logical Switches, each with its own subnet
- 3 x VMs, connected one per logical switch
  - Each with its own IP address and IP gateway
  - Each with a MAC address (last two octets are shown)
- One DLR connected to the 4 logical switches; one logical switch is for the “Uplink," while the rest are Internal
- An external gateway, which could be an ESG, serving as an upstream gateway for the DLR.

The “Ready to complete” wizard screen shows for the DLR above.



After the deployment of the DLR finishes, ESXi CLI commands can be used to view and validate the distributed state of the DLR in question on the participating hosts.

Confirming DLR Instances

The first thing to confirm is whether the DLR instance has been created and whether its control plane is active.

1 From the NSX Manager shell, run `show cluster all` to get the cluster ID.

2 Run `show cluster cluster-id` to get the host ID.

3 Run `show logical-router host hostID dlr all verbose` to get the status information.

```
nsxmgr# show logical-router host host-id dlr all verbose

VDR Instance Information :
--------------------------

Vdr Name:                   default+edge-1
Vdr Id:                     1460487509
Number of Lifs:             4
Number of Routes:           5
State:                      Enabled
Controller IP:              192.168.110.201
Control Plane Active:       Yes
Control Plane IP:           192.168.210.51
Edge Active:                No
```

The points to note:

- This command displays all DLR instances that exist on the given ESXi host.
- “Vdr Name” consists of “Tenant” “+ “Edge Id." In the example, “Tenant” was not specified, so the word “default” is used. The “Edge Id” is “edge-1," which can be seen in the NSX UI.
  - In cases where there are many DLR instances on a host, a method for finding the right instance is to look for the “Edge ID” displayed in the UI “NSX Edges."
- “Vdr Id” is useful for further lookups, including logs.
- “Number of Lifs” refers to the LIFs that exist on this individual DLR instance.
- “Number of Routes” is in this case 5, which consists of 4 x directly connected routes (one for each LIF), and a default route.
- “State,” “Controller IP,” and “Control Plane Active” refer to the state of the DLR’s control plane and must list the correct Controller IP, with Control Plane Active: Yes. Remember, the DLR function requires working Controllers; the output above shows what is expected for a healthy DLR instance.
- “Control Plane IP” refers to the IP address that the ESXi host uses to talk to the Controller. This IP is always the one associated with the ESXi host’s Management vmknic, which in most cases is vmk0.
- “Edge Active” shows whether or not this host is the one where the Control VM for this DLR instance is running and in Active state.
  - The placement of the Active DLR Control VM determines which ESXi host is used to perform NSX L2 bridging, if it is enabled.
- There is also a “brief” version of the above command that produces a compressed output useful for a quick overview. Note that “Vdr Id” is displayed in hexadecimal format here:

```
nsxmgr# show logical-router host host-id dlr all brief

VDR Instance Information :
---------------------------

State Legend: [A: Active], [D: Deleting], [X: Deleted], [I: Init]
State Legend: [SF-R: Soft Flush Route], [SF-L: Soft Flush LIF]

Vdr Name             Vdr Id     #Lifs   #Routes State      Controller Ip    CP Ip
--------             -------    -----   ------- -----      -------------    ------
default+edge-1       0x570d4555 4       5       A          192.168.110.201  192.168.210.51
```

The “Soft Flush” states refer to short-lived transient states of the LIF lifecycle and is not normally seen in a healthy DLR.

DLR’s Logical Interfaces

After establishing the that the DLR has been created, make sure that all of the DLR's logical interfaces are present and have the correct configuration.

1 From the NSX Manager shell, run `show cluster all` to get the cluster ID.

2 Run `show cluster cluster-id` to get the host ID.

3 Run `show logical-router host hostID dlr all brief` to get the dlrID (Vdr Name).

4 Run `show logical-router host hostID dlr dlrID interface all brief` to get summarized status information for all interfaces.

5 Run `show logical-router host hostID dlr dlrID interface (all | intName) verbose` to get the status information for all interfaces or for a specific interface.

```
nsxmgr# show logical-router host hostID dlr dlrID interface all verbose

VDR default+edge-1:1460487509 LIF Information :

Name:                570d45550000000a
Mode:                Routing, Distributed, Internal
Id:                  Vxlan:5000
Ip(Mask):            172.16.10.1(255.255.255.0)
Connected Dvs:       Compute_VDS
VXLAN Control Plane: Enabled
VXLAN Multicast IP:  0.0.0.1
State:               Enabled
Flags:               0x2388
DHCP Relay:          Not enabled

Name:                570d45550000000c
Mode:                Routing, Distributed, Internal
Id:                  Vxlan:5002
Ip(Mask):            172.16.30.1(255.255.255.0)
Connected Dvs:       Compute_VDS
VXLAN Control Plane: Enabled
VXLAN Multicast IP:  0.0.0.1
State:               Enabled
Flags:               0x2288
DHCP Relay:          Not enabled

Name:                570d45550000000b
Mode:                Routing, Distributed, Internal
Id:                  Vxlan:5001
Ip(Mask):            172.16.20.1(255.255.255.0)
Connected Dvs:       Compute_VDS
VXLAN Control Plane: Enabled
VXLAN Multicast IP:  0.0.0.1
State:               Enabled
Flags:               0x2388
DHCP Relay:          Not enabled

Name:                570d455500000002
Mode:                Routing, Distributed, Uplink
Id:                  Vxlan:5003
Ip(Mask):            192.168.10.2(255.255.255.248)
Connected Dvs:       Compute_VDS
VXLAN Control Plane: Enabled
```

```
VXLAN Multicast IP:  0.0.0.1
State:               Enabled
Flags:               0x2208
DHCP Relay:          Not enabled
```

The points to note:

- LIF "Name" is unique across all DLR instances on the host. It is the same on hosts and on the DLR's master Controller node.
- LIF's "Mode" shows whether the LIF is routing or bridging, and whether it is internal or uplink.
- "Id" shows the LIF type and the corresponding service ID (VXLAN and VNI, or VLAN and VID).
- "Ip(Mask)" is shown for "Routing" LIFs.
- If a LIF is connected to a VXLAN in hybrid or unicast mode, "VXLAN Control Plane" is "Enabled."
- For VXLAN LIFs where VXLAN is in unicast mode, "VXLAN Multicast IP" is shown as "0.0.0.1"; otherwise the actual multicast IP address is displayed.
- "State" should be "Enabled" for routed LIFs. For bridging LIFs, it is "Enabled" on the host that is performing bridging and "Init" on all other hosts.
- "Flags" is a summary representation of the LIF's state and shows whether the LIF is:
  - Routed or Bridged
  - Whether the VLAN LIF is a DI
  - Whether it has DHCP relay enabled
  - Of note is the flag 0x0100, which is set when a VXLAN VNI join was caused by the DLR (as opposed to a host having a VM on that VXLAN)
  - Flags are displayed in a more readable format in "brief" mode.

```
nsxmgr# show logical-router host hostID dlr dlrID interface all brief

VDR default+edge-1 LIF Information :

State Legend: [A:Active], [d:Deleting], [X:Deleted], [I:Init],[SF-L:Soft Flush LIF]
Modes Legend: [B:Bridging],[E: Empty], [R:Routing],[S:Sedimented],[D:Distributed]
Modes Legend: [In:Internal],[Up:Uplink]

Lif Name             Id              Mode     State    Ip(Mask)
--------             --              -----    -----    --------
570d45550000000a     Vxlan:5001      R,D,In   A        172.16.10.1(255.255.255.0)
570d45550000000c     Vxlan:5003      R,D,In   A        172.16.30.1(255.255.255.0)
570d45550000000b     Vxlan:5002      R,D,In   A        172.16.20.1(255.255.255.0)
570d455500000002     Vxlan:5000      R,D,Up   A        192.168.10.5(255.255.255.248)
```

DLR's Routes

After you have established that a DLR is present and healthy and it has all the LIFs, the next thing to check is the routing table.

1 From the NSX Manager shell, run `show cluster all` to get the cluster ID.

2 Run `show cluster cluster-id` to get the host ID.

3 Run `show logical-router host hostID dlr all brief` to get the dlrID (Vdr Name).

4 Run `show logical-router host hostID dlr dlrID route` to get the status information for all interfaces.

```
nsxmgr# show logical-router host hostID dlr dlrID route

VDR default+edge-1:1460487509 Route Table
Legend: [U: Up], [G: Gateway], [C: Connected], [I: Interface]
Legend: [H: Host], [F: Soft Flush] [!: Reject] [E: ECMP]

Destination      GenMask          Gateway          Flags    Ref Origin   UpTime     Interface
-----------      -------          -------          -----    --- ------   ------     ---------
0.0.0.0          0.0.0.0          192.168.10.1     UG       1   AUTO     10068944   570d455500000002
172.16.10.0      255.255.255.0    0.0.0.0          UCI      1   MANUAL   10068944   570d45550000000a
172.16.20.0      255.255.255.0    0.0.0.0          UCI      1   MANUAL   10068944   570d45550000000b
172.16.30.0      255.255.255.0    0.0.0.0          UCI      1   MANUAL   10068944   570d45550000000c
192.168.10.0     255.255.255.248  0.0.0.0          UCI      1   MANUAL   10068944   570d455500000002
```

Points to note:

- "Interface" shows the egress LIF that will be selected for the corresponding "Destination." It is set to the "Lif Name" of one of the DLR's LIFs.
- For ECMP routes, there will be more than one route with the same Destination, GenMask, and Interface, but a different Gateway. Flags will also include "E" to reflect the ECMP nature of these routes.

DLR's ARP table

For packets it forwards, the DLR must be able to resolve ARP requests for the next hop's IP address. The results of this resolution process are stored locally on the individual hosts' DLR instances.

Controllers play no role in this process and are not used to distribute resulting ARP entries to other hosts.

Inactive cached entries are kept for 600 seconds, then removed. For more information about the DLR ARP resolution process, see DLR ARP Resolution Process.

1 From the NSX Manager shell, run `show cluster all` to get the cluster ID.

2 Run `show cluster cluster-id` to get the host ID.

3 Run `show logical-router host hostID dlr all brief` to get the dlrID (Vdr Name).

4 Run `show logical-router host hostID dlr dlrID arp` to get the status information for all interfaces.

```
nsxmgr# show logical-router host hostID dlr dlrID arp

VDR default+edge-1:1460487509 ARP Information :
Legend: [S: Static], [V: Valid], [P: Proxy], [I: Interface]
Legend: [N: Nascent], [L: Local], [D: Deleted]

Network          Mac                Flags      Expiry     SrcPort    Interface Refcnt
-------          ---                -----      -------    ---------  --------- ------
172.16.10.1      02:50:56:56:44:52  VI         permanent  0          570d45550000000a 1
172.16.10.11     00:50:56:a6:7a:a2  VL         147        50331657   570d45550000000a 2
172.16.30.1      02:50:56:56:44:52  VI         permanent  0          570d45550000000c 1
172.16.30.11     00:50:56:a6:ba:09  V          583        50331650   570d45550000000c 2
172.16.20.11     00:50:56:a6:84:52  VL         568        50331658   570d45550000000b 2
172.16.20.1      02:50:56:56:44:52  VI         permanent  0          570d45550000000b 1
192.168.10.2     02:50:56:56:44:52  VI         permanent  0          570d455500000002 1
192.168.10.1     00:50:56:8e:ee:ce  V          147        50331650   570d455500000002 1
```

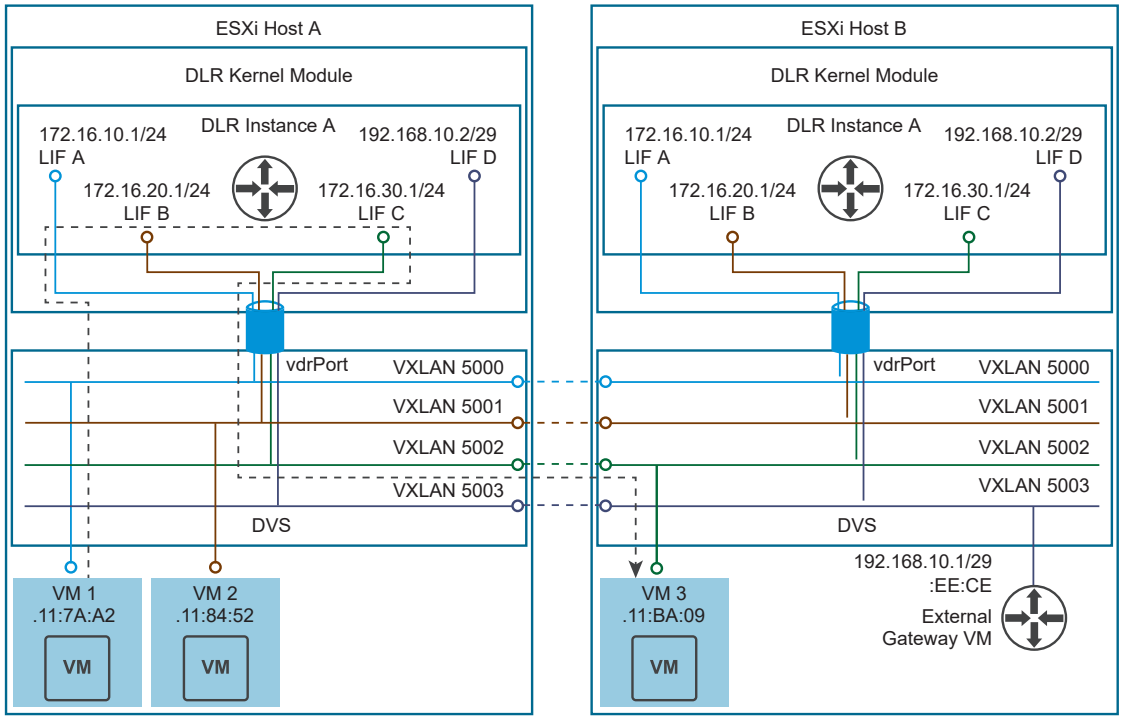
Things to note:

- All ARP entries for the DLR's own LIFs ("I" Flag) are the same and show the same vMAC that was discussed in VXLAN Preparation Check.
- ARP entries with the "L" Flag correspond to the VMs running on the host where the CLI command is run.
- "SrcPort" shows the dvPort ID where the ARP entry was originated. In cases where an ARP entry was originated on another host, the dvUplink's dvPort ID is shown. This dvPort ID can be cross-referenced with the dvUplink dvPort ID discussed in VXLAN Preparation Check.
- The "Nascent" flag is not normally observed. It is set while the DLR is waiting for the ARP reply to arrive. Any entries with that flag set might indicate that there is a problem with ARP resolution.

DLR and Its Related Host Components Visualized

The following diagram shows two hosts, ESXi Host A and ESXi Host B, where our example "DLR Instance A" is configured and connected to the four VXLAN LIFs.

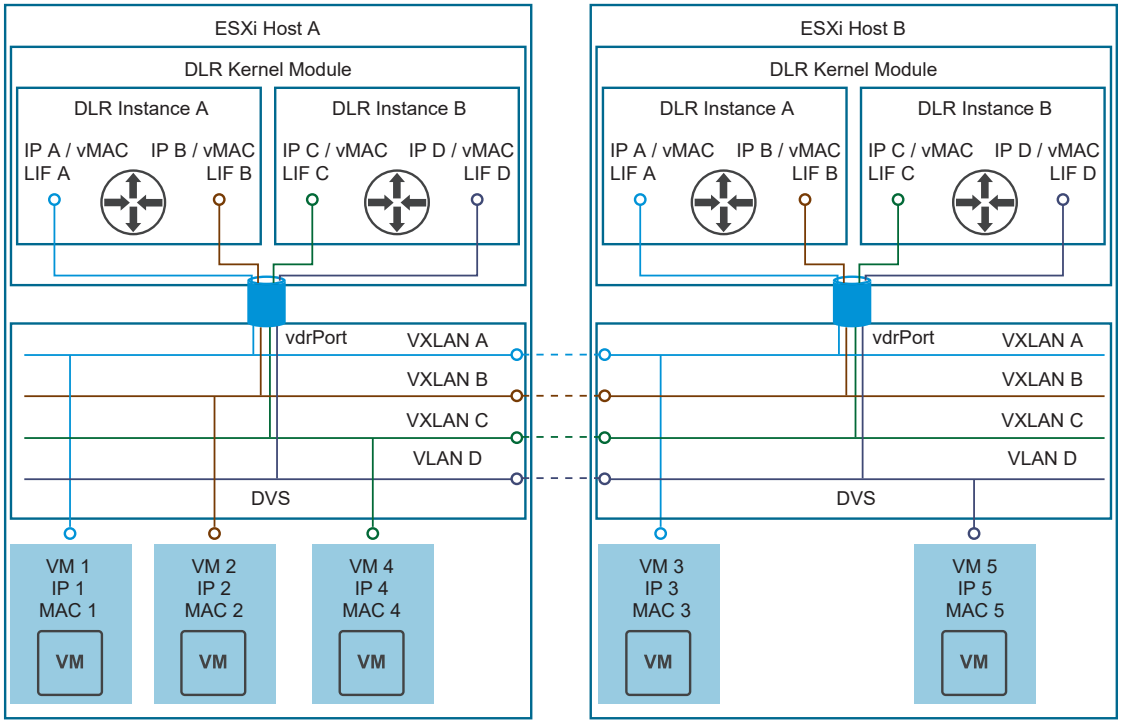
Figure 3-8. Two Hosts with a Single DLR Instance



- Each host has an “L2 Switch” (DVS), and a “Router on a stick” (DLR kernel module), connected to that “switch” via a “trunk” interface (vdrPort).
  - Note that this “trunk” can carry both VLANs and VXLANs; however, there are no 801.Q or UDP/VXLAN headers present in the packets that traverse the vdrPort. Instead, the DVS uses an internal metadata tagging method to communicate that information to the DLR kernel module.
- When the DVS sees a frame with Destination MAC = vMAC, it knows that it is for the DLR, and forwards that frame to the vdrPort.
- After packets arrive in the DLR kernel module via the vdrPort, their metadata is examined to determine the VXLAN VNI or VLAN ID that they belong to. This information is then used to determine which LIF of which DLR instance that packet belongs to.
  - The side effect of this system is that no more than one DLR instance can be connected to a given VLAN or VXLAN.

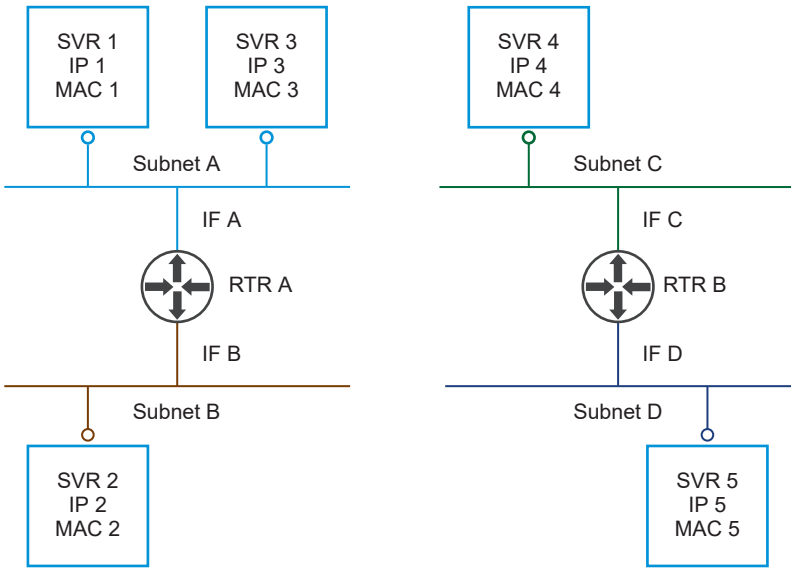
In cases where more than one DLR instance exists, the diagram above would look like this:

Figure 3-9. Two Hosts with Two DLR Instances



This would correspond to a network topology with two independent routing domains, operating in complete separation from each other, potentially with overlapping IP addresses.

Figure 3-10. Network Topology Corresponding with Two Hosts and Two DLR Instances



Distributed Routing Subsystem Architecture

DLR instances on ESXi hosts have access to all information needed to perform L3 routing.

- Networks are directly connected (learned from the interfaces' configuration)

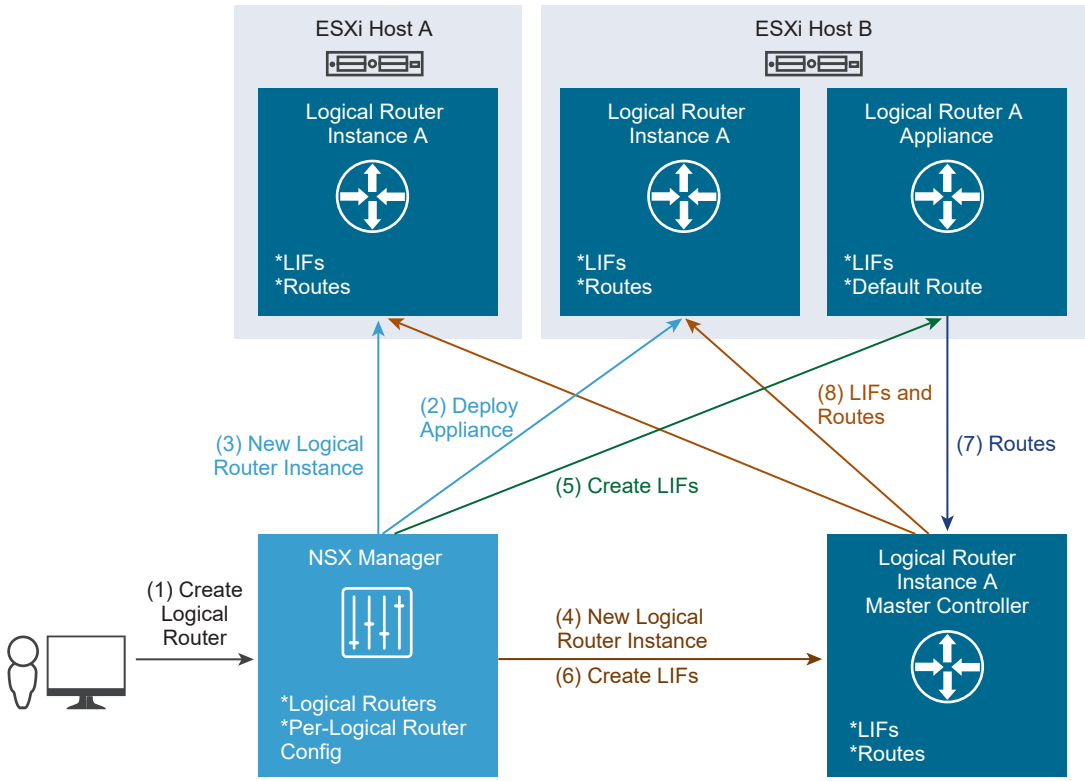
- Next hops for each subnet (looked up in routing table)
- MAC address to insert into egress frames to reach the next hops (ARP table)

This information is delivered to the instances distributed across multiple ESXi hosts.

DLR Creation Process

The following diagram is a high-level illustration for the process NSX follows to create a new DLR.

Figure 3-11. DLR Creation Process



When a UI wizard is submitted with the “Finish” button or an API call is made to deploy a new DLR, the system processes through the following steps:

1 NSX Manager receives an API call to deploy a new DLR (directly or from vSphere Web Client, invoked by the UI wizard).

2 NSX Manager calls its linked vCenter Server to deploy a DLR Control VM (or a pair, if HA was requested).

   a DLR Control VM is powered on and connects back to the NSX Manager, ready to receive configuration.

   b If an HA pair was deployed, NSX Manager configures an anti-affinity rule that will keep the HA pair running on different hosts. DRS then takes action to move them apart.

3 NSX Manager creates DLR instance on hosts:

   a NSX Manager looks up the logical switches that are to be connected to the new DLR to determine which transport zone they belong to.

   b It then looks up a list of clusters that are configured in this transport zone and creates the new DLR on each host in these clusters.

   c At this point, hosts only know the new DLR ID, but they do not have any corresponding information (LIFs or routes).

4 NSX Manager creates a new DLR instance on the Controller Cluster.

   a Controller Cluster allocates one of the Controller nodes to be the master for this DLR instance.

5 NSX Manager sends the configuration, including LIFs, to the DLR Control VM.

   a ESXi hosts (including the one where the DLR Control VM is running) receive slicing information from the Controller Cluster, determine which Controller node is responsible for the new DLR instance, and connect to the Controller node (if there was no existing connection).

6 After LIF creation on DLR Control VM, the NSX Manager creates the new DLR's LIFs on the Controller Cluster.

7 DLR Control VM connects to the new DLR instance's Controller node, and sends the Controller node the routes:

   a First the DLR translates its routing table into the forwarding table (by resolving prefixes to LIFs).

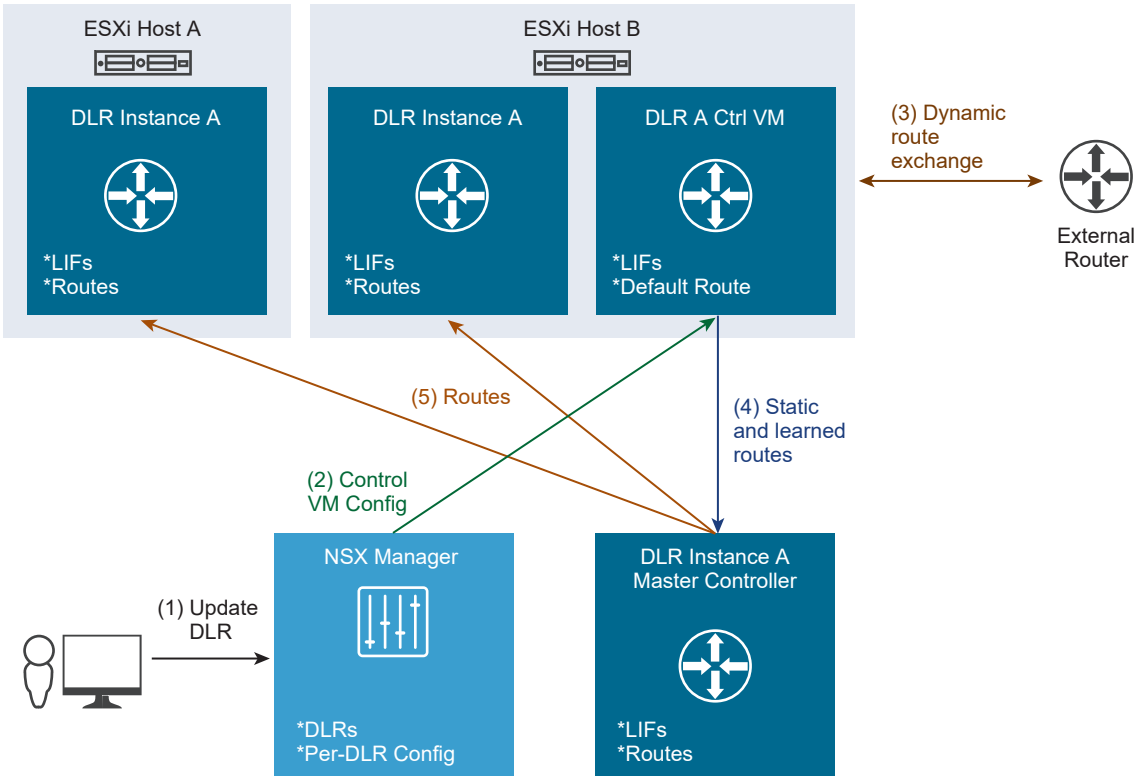
   b Then The DLR sends the resulting table to the Controller node.

8 Controller node pushes LIFs and routes to the other hosts where the new DLR instance exists, via the connection established in step 5.a.

Adding Dynamic Routing to a DLR

When the DLR is created via a "direct" API call (as opposed to using the vSphere Web Client UI), it is possible to supply it with a complete configuration that includes dynamic routing(1).

Figure 3-12. Dynamic Routing on the DLR



1 The NSX Manager receives an API call to change the existing DLR's configuration, in this case – add dynamic routing.

2 The NSX Manager sends the new configuration to the DLR Control VM.

3 The DLR Control VM applies the configuration and goes through the process of establishing routing adjacencies, exchanging routing information, and so on.

4 After the routing exchange, the DLR Control VM calculates the forwarding table and sends it to the DLR's master Controller node.

5 The DLR's master Controller node then distributes the updated routes to the ESXi hosts where the DLR instance exists.

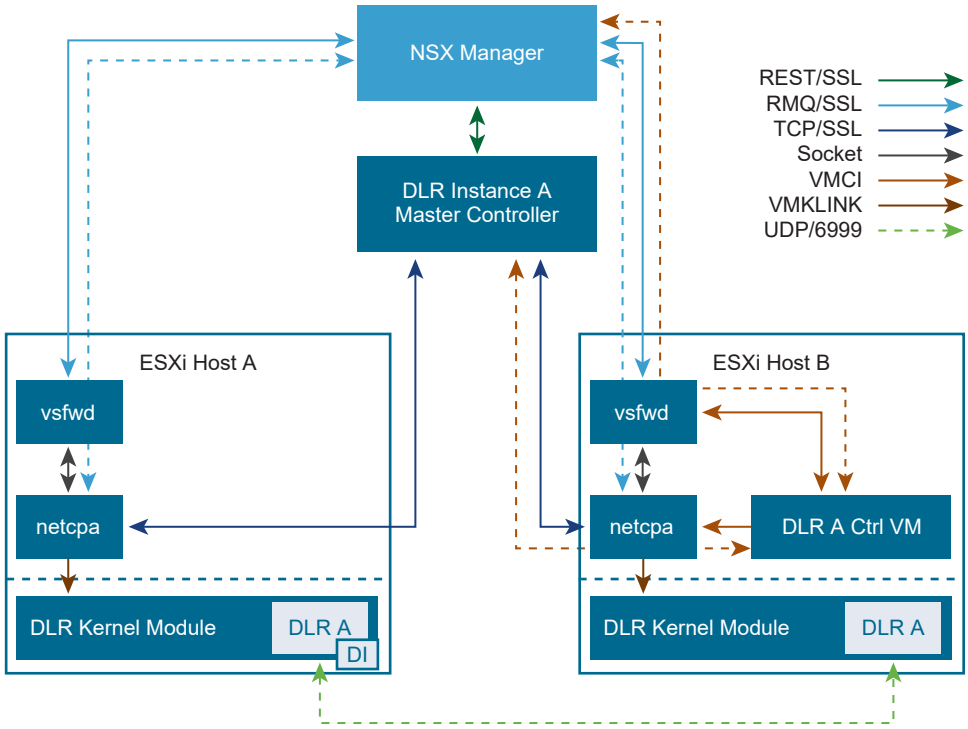
Note that the DLR instance on the ESXi host where the DLR Control VM is running receives its LIFs and routes only from the DLR's master Controller node, never directly from the DLR Control VM or the NSX Manager.

DLR Control and Management Plane Components and Communications

This section provides a brief overview of the components of the DLR control and management planes.

The figure shows the components and the corresponding communication channels between them.

Figure 3-13. DLR Control and Management Plane Components



- NSX Manager:
  - Has direct communications with the Controller Cluster
  - Has a direct permanent connection with the message bus client (vsfwd) process running on each host prepared for NSX
- For each DLR instance, one Controller node (out of the available 3) is elected as master
  - The master function can move to a different Controller node, if the original Controller node fails
- Each ESXi host runs two User World Agents (UWA): message bus client (vsfwd) and control plane agent (netcpa)
  - netcpa requires information from the NSX Manager to function (for example, where to find Controllers and how to authenticate to them); this information is accessed via the message bus connection provided by vsfwd
  - netcpa also communicates with the DLR kernel module to program it with the relevant information it receives from Controllers
- For each DLR instance, there is a DLR Control VM, which is running on one of the ESXi hosts; the DLR Control VM has two communication channels:
  - VMCI channel to the NSX Manager via vsfwd, which is used for configuring the Control VM
  - VMCI channel to the DLR master Controller via netcpa, which is used to send the DLR's routing table to the Controller

- In cases where the DLR has a VLAN LIF, one of the participating ESXi hosts is nominated by the Controller as a designated instance (DI). The DLR kernel module on other ESXi hosts requests that the DI perform proxy ARP queries on the associated VLAN.

NSX Routing Subsystem Components

The NSX routing subsystem is enabled by multiple components.

- NSX Manager
- Cluster of Controllers
- ESXi host modules (kernel and UWA)
- DLR Control VMs
- ESGs

NSX Manager

NSX Manager provides the following functions relevant to NSX routing:

- Acts as a centralized management plane, providing the unified API access point for all NSX management operations
- Installs the Distributed Routing Kernel Module and User World Agents on hosts to prepare them for NSX functions
- Creates/destroys DLRs and DLR LIFs
- Deploys/deletes DLR Control VM and ESG via vCenter
- Configures the Controller Cluster via a REST API and hosts via a message bus:
  - Provides host Control Plane agents with the IP addresses of Controllers
  - Generates and distributes to hosts and controllers the certificates to secure control plane communications
- Configures ESGs and DLR Control VMs via the message bus
  - Note that ESGs can be deployed on unprepared hosts, in which case VIX will be used in lieu of the message bus

Cluster of Controllers

NSX distributed routing requires Controllers, clustered for scale and availability, which provide the following functions:

- Support VXLAN and distributed routing control plane
- Provide CLI interface for statistics and runtime states
- Elect a master controller node for each DLR instance
  - Master node receives routing information from the DLR Control VM and distributes it to the hosts
  - Sends LIF table to the hosts

- Keeps track of the host where DLR Control VM resides
- Selects designated instance for VLAN LIFs and communicates this information to hosts; monitors DI host via control plane keepalives (timeout is 30 seconds, and detection time can be 20-40 seconds), sends hosts an update if the selected DI host disappears

ESXi host modules

NSX routing directly utilizes two User World Agents (UWA) and a routing kernel module and also relies on the VXLAN kernel module for VXLAN connectivity.

Here is a summary of what each of these components does:

- Control Plane Agent (netcpa) is a TCP (SSL) client that communicates with the Controller using the control plane protocol. It might connect to multiple controllers. netcpa communicates with the Message Bus Client (vsfwd) to retrieve control plane related information from NSX Manager.
- netcpa packaging and deployment:
  - The agent is packaged into the VXLAN VIB (vSphere installation bundle)
  - Installed by NSX Manager via EAM (ESX Agency Manager) during host preparation
  - Runs as a service daemon on ESXi netcpa
  - Can be started / stopped / queried via its startup script /etc/init.d/netcpad
  - Can be restarted remotely via Networking and Security UI Installation -> Host Preparation -> Installation Status, on individual hosts or on a whole cluster
- DLR Kernel Module (vdrb) integrates with DVS to enable L3 forwarding
  - Configured by netcpa
  - Installed as part of the VXLAN VIB deployment
  - Connects to DVS via the special trunk called “vdrPort," which supports both VLANs and VXLANs
  - Holds information about DLR instances, with per-instance:
    - LIF and Route tables
    - host-local ARP cache
- Message Bus Client (vsfwd) is used by netcpa, ESGs, and DLR Control VMs to communicate with the NSX Manager
  - vsfwd obtains NSX Manager’s IP address from /UserVars/RmqIpAddress set by vCenter via vpxa/hosd and logs into the Message Bus server using per-host credentials stored in other /UserVars/Rmq* variables
- netcpa running on an ESXi host relies on vsfwd to do the following:
  - Obtain host’s control plane SSL private key and certificate from NSX Manager. These are then stored in /etc/vmware/ssl/rui-for-netcpa.*

    - Get IP addresses and SSL thumbprints of Controllers from NSX Manager. These are then stored in /etc/vmware/netcpa/config-by-vsm.xml.
    - Create and delete DLR instances on its host on instruction from NSX Manager
- Packaging and deployment
    - Same as netcpa, it's a part of the VXLAN VIB
    - Runs as a service daemon on ESXi vsfwd
    - Can be started / stopped / queried via its startup script /etc/init.d/ vShield-Stateful-Firewall
- ESGs and DLR Control VMs use VMCI channel to vsfwd to receive configuration from NSX Manager

DLR Control VMs and ESGs

- DLR Control VM is a "route processor" for its DLR instance
    - Has a "placeholder" or a "real vNIC" interfaces for each DLR LIF, along with IP configuration
    - Can run one of two available dynamic routing protocol (BGP or OSPF) and/or use static routes
    - Requires at least one "Uplink" LIF to be able to run OSPF or BGP
    - Computes forwarding table from directly connected (LIF) subnets, static, and dynamic routes, and sends it via its VMCI link to netcpa to the DLR instance's master Controller
    - Supports HA in Active/Standby VM pair configuration
- ESG is a self-contained router in a VM
    - Completely independent from the NSX DLR routing subsystem (no NSX control plane integration)
    - Typically used as an upstream gateway for one or more DLRs
    - Supports more than one concurrently running dynamic routing protocol

NSX Routing Control Plane CLI

In addition to the host components, NSX Routing employs services of the Controller Cluster and DLR Control VMs, each of which is a source of the DLR control plane information and has its own CLI used to examine it.

DLR Instance Master Controller

Each DLR Instance is served by one of the Controller nodes. The following CLI commands can be used to view information that this Controller node has for the DLR Instance for which it is the master:

```
nsx-controller # show control-cluster logical-routers instance 1460487509
LR-Id      LR-Name           Hosts[]         Edge-Connection Service-Controller
1460487509 default+edge-1    192.168.210.57                  192.168.110.201
                             192.168.210.51
                             192.168.210.52
                             192.168.210.56
                             192.168.110.51
```

```
                              192.168.110.52

nsx-controller # show control-cluster logical-routers interface-summary 1460487509
Interface                       Type   Id          IP[]
570d455500000002                vxlan  5003        192.168.10.2/29
570d45550000000b                vxlan  5001        172.16.20.1/24
570d45550000000c                vxlan  5002        172.16.30.1/24
570d45550000000a                vxlan  5000        172.16.10.1/24

nsx-controller # show control-cluster logical-routers routes 1460487509
LR-Id       Destination        Next-Hop
1460487509  0.0.0.0/0          192.168.10.1
```

- The “instance” sub-command of the “show control-cluster logical-routers” command displays list of hosts that are connected to this Controller for this DLR Instance. In a correctly functioning environment, this list would include all hosts from all clusters where the DLR exists.
- The “interface-summary” displays the LIFs that the Controller learned from the NSX Manager. This information is sent to the hosts.
- The “routes” shows the routing table sent to this Controller by this DLR’s Control VM. Note that unlike on the ESXi hosts, this table does not include any directly connected subnets because this information is provided by the LIF configuration.

DLR Control VM

DLR Control VM has LIFs and routing/forwarding tables. The major output of DLR Control VM’s lifecycle is the DLR routing table, which is a product of Interfaces and Routes.

```
edge-1-0> show ip route

Codes: O - OSPF derived, i - IS-IS derived, B - BGP derived,
C - connected, S - static, L1 - IS-IS level-1, L2 - IS-IS level-2,
IA - OSPF inter area, E1 - OSPF external type 1, E2 - OSPF external type 2

Total number of routes: 5

S       0.0.0.0/0            [1/1]         via 192.168.10.1
C       172.16.10.0/24       [0/0]         via 172.16.10.1
C       172.16.20.0/24       [0/0]         via 172.16.20.1
C       172.16.30.0/24       [0/0]         via 172.16.30.1
C       192.168.10.0/29      [0/0]         via 192.168.10.2

edge-1-0> show ip forwarding
Codes: C - connected, R - remote,
       > - selected route, * - FIB route
R>* 0.0.0.0/0 via 192.168.10.1, vNic_2
```

```
C>* 172.16.10.0/24 is directly connected, VDR
C>* 172.16.20.0/24 is directly connected, VDR
C>* 172.16.30.0/24 is directly connected, VDR
C>* 192.168.10.0/29 is directly connected, vNic_2
```

- The purpose of the Forwarding Table is to show which DLR interface is chosen as the egress for a given destination subnet.
  - The “VDR” interface is displayed for all LIFs of “Internal” type. The “VDR” interface is a pseudo-interface that does not correspond to a vNIC.

The DLR Control VM’s interfaces can be displayed as follows:

```
edge-1-0> show interface
Interface VDR is up, line protocol is up
  index 2 metric 1 mtu 1500 <UP,BROADCAST,RUNNING,NOARP>
  HWaddr: be:3d:a1:52:90:f4
  inet6 fe80::bc3d:a1ff:fe52:90f4/64
  inet 172.16.10.1/24
  inet 172.16.20.1/24
  inet 172.16.30.1/24
  proxy_arp: disabled
  Auto-duplex (Full), Auto-speed (2460Mb/s)
    input packets 0, bytes 0, dropped 0, multicast packets 0
    input errors 0, length 0, overrun 0, CRC 0, frame 0, fifo 0, missed 0
    output packets 0, bytes 0, dropped 0
    output errors 0, aborted 0, carrier 0, fifo 0, heartbeat 0, window 0
    collisions 0

Interface vNic_0 is up, line protocol is up
  index 3 metric 1 mtu 1500 <UP,BROADCAST,RUNNING,MULTICAST>
  HWaddr: 00:50:56:8e:1c:fb
  inet6 fe80::250:56ff:fe8e:1cfb/64
  inet 169.254.1.1/30
  inet 10.10.10.1/24
  proxy_arp: disabled
  Auto-duplex (Full), Auto-speed (2460Mb/s)
    input packets 582249, bytes 37339072, dropped 49, multicast packets 0
    input errors 0, length 0, overrun 0, CRC 0, frame 0, fifo 0, missed 0
    output packets 4726382, bytes 461202852, dropped 0
    output errors 0, aborted 0, carrier 0, fifo 0, heartbeat 0, window 0
    collisions 0

Interface vNic_2 is up, line protocol is up
  index 9 metric 1 mtu 1500 <UP,BROADCAST,RUNNING,MULTICAST>
  HWaddr: 00:50:56:8e:ae:08
  inet 192.168.10.2/29
  inet6 fe80::250:56ff:fe8e:ae08/64
  proxy_arp: disabled
  Auto-duplex (Full), Auto-speed (2460Mb/s)
    input packets 361446, bytes 30167226, dropped 0, multicast packets 361168
```

```
    input errors 0, length 0, overrun 0, CRC 0, frame 0, fifo 0, missed 0
    output packets 361413, bytes 30287912, dropped 0
    output errors 0, aborted 0, carrier 0, fifo 0, heartbeat 0, window 0
    collisions 0
```

Notes of interest:

- Interface “VDR” does not have a VM NIC (vNIC) associated with it. It is a single “pseudo-interface” that is configured with all IP addresses for all DLR’s “Internal” LIFs.
- Interface vNic_0 in this example is the HA interface.
  - The output above was taken from a DLR deployed with HA enabled, and the HA interface is assigned an IP address. This appears as two IP addresses, 169.254.1.1/30 (auto-assigned for HA), and 10.10.10.1/24, manually assigned to the HA interface.
  - On an ESG, the operator can manually assign one of its vNICs as HA, or leave it as default for the system to choose automatically from available “Internal” interfaces. Having the “Internal” type is a requirement, or HA will fail.
- Interface vNic_2 is an Uplink type; therefore, it is represented as a “real” vNIC.
  - Note that the IP address seen on this interface is the same as the DLR’s LIF; however, the DLR Control VM will not answer ARP queries for the LIF IP address (in this case, 192.168.10.2/29). There is an ARP filter applied for this vNIC’s MAC address that makes it happen.
  - The point above holds true until a dynamic routing protocol is configured on the DLR, when the IP address will be removed along with the ARP filter and replaced with the “Protocol IP” address specified during the dynamic routing protocol configuration.
  - This vNIC is used by the dynamic routing protocol running on the DLR Control VM to communicate with the other routers to advertise and learn routes.
- After edge is disconnected and post HA failover, the disconnected edge interface IP address is removed from the active edge routing information base (RIB)/forwarding information base (FIB). But the standby edge FIB table or the `show ip forwarding` command still shows the IP and is not removed from the FIB table. This is expected behavior.

NSX Routing Subsystem Failure Modes and Effects

This chapter reviews the typical failure scenarios that might affect components of NSX routing subsystem and outlines the effects of these failures.

NSX Manager

Table 3-2. NSX Manager Faiure Modes and Effects

Failure Mode	Failure Effects
Loss of network connectivity to NSX Manager VM	■ Total outage of all NSX Manager functions, including CRUD for NSX routing/bridging ■ No configuration data loss ■ No data or control-plane outage
Loss of network connectivity between NSX Manager and ESXi hosts or RabbitMQ server failure	■ If DLR Control VM or ESG is running on affected hosts, CRUD operations on them fail ■ Creation and deletion of DLR instances on affected hosts fail ■ No configuration data loss ■ No data or control-plane outage ■ Any dynamic routing updates continue to work
Loss of network connectivity between NSX Manager and Controllers	■ Create, update, and delete operations for NSX distributed routing and bridging fail ■ No configuration data loss ■ No data or control-plane outage
NSX Manager VM is destroyed (datastore failure)	■ Total outage of all NSX Manager functions, including CRUD for NSX routing/bridging ■ Risk of subset of routing/bridging instances becoming orphaned if NSX Manager restored to an older configuration, requiring manual clean-up and reconciliation ■ No data or control-plane outage, unless reconciliation is required

Controller Cluster

Table 3-3. NSX Controller Faiure Modes and Effects

Failure Mode	Failure Effects
Controller cluster loses network connectivity with ESXi hosts	■ Total outage for DLR Control Plane functions (Create, update, and delete routes, including dynamic) ■ Outage for DLR Management Plane functions (Create, update, and delete LIFs on hosts) ■ VXLAN forwarding is affected, which may cause end to end (L2+L3) forwarding process to also fail ■ Data plane continues working based on the last-known state
One or two Controllers lose connectivity with ESXi hosts	■ If affected Controller can still reach other Controllers in the cluster, any DLR instances mastered by this Controller experience the same effects as described above. Other Controllers do not automatically take over
One Controller loses network connectivity with other Controllers (or completely)	■ Two remaining Controllers take over VXLANs and DLRs handled by the isolated Controller ■ Affected Controller goes into Read-Only mode, drop its sessions to hosts, and refuse new ones

Table 3-3. NSX Controller Faiure Modes and Effects (continued)

Failure Mode	Failure Effects
Controllers lose connectivity with each other	■ All Controllers will go into Read-Only mode, close connections to hosts, and refuse new ones ■ Create, update, and delete operations for all DLRs' LIFs and routes (including dynamic) fail ■ NSX routing configuration (LIFs) might get out of sync between the NSX Manager and Controller Cluster, requiring manual intervention to resync ■ Hosts will continue operating on last known control plane state
One Controller VM is lost	■ Controller Cluster loses redundancy ■ Management/Control plane continues to operate as normal
Two Controller VMs are lost	■ Remaining Controller will go into read-only mode; effect is the same as when Controllers lose connectivity with each other (above). Likely to require manual cluster recovery

Host Modules

netcpa relies on host SSL key and certificate, plus SSL thumbprints, to establish secure communications with the Controllers. These are obtained from NSX Manager via the message bus (provided by vsfwd).

If certificate exchange process fails, netcpa will not be able to successfully connect to Controllers.

Note: This section doesn't cover failure of kernel modules, as the effect of this is severe (PSOD) and is a rare occurrence.

Table 3-4. Host Module Faiure Modes and Effects

Failure Mode	Failure Effects
vsfwd uses username/password authentication to access message bus server, which can expire	■ If a vsfwd on a freshly prepared ESXi host cannot reach NSX Manager within two hours, the temporary login/password supplied during installation expires, and message bus on this host becomes inoperable
Effects of failure of the Message Bus Client (vsfwd) depend on the timing.	
If it fails before other parts of NSX control plane had a chance to reach steady running state	■ Distributed routing on the host stops functioning, because the host is not be able to talk to Controllers ■ Host do not learn DLR instances from NSX Manager
If it fails after host has reached steady state	■ ESGs and DLR Control VMs running on the host won't be able to receive configuration updates ■ Host do not learn of new DLRs, and are not able to delete existing DLRs ■ Host datapath will continue operating based on the configuration host had at the time of failure

Table 3-5. netcpa Faiure Modes and Effects

Failure Mode	Failure Effects
Effects of failure of the Control Plane Agent (netcpa) depend on the timing	
If it fails before NSX datapath kernel modules had a chance to reach steady running state	■ Distributed routing on the host stops functioning
If it fails after host has reached steady state	■ DLR Control VM(s) running on the host will not be able to send their forwarding table updates to Controller(s) ■ Distributed routing datapath will not receive any LIF or route updates from Controller(s), but will continue operating based on the state it had before the failure

DLR Control VM

Table 3-6. DLR Control VM Faiure Modes and Effects

Failure mode	Failure Effects
DLR Control VM is lost or powered off	■ Create, update, and delete operations for this DLR's LIFs and routes fail ■ Any dynamic route updates will not be sent to hosts (including withdrawal of prefixes received via now broken adjacencies)
DLR Control VM loses connectivity with the NSX Manager and Controllers	■ Same effects as above, except if DLR Control VM and its routing adjacencies are still up, traffic to and from previously learned prefixes will not be affected
DLR Control VM loses connection with the NSX Manager	■ NSX Manager's Create, update, and delete operations for this DLR's LIFs and routes fail and are not re-tried ■ Dynamic routing updates continue to propagate
DLR Control VM loses connection with the Controllers	■ Any routing changes (static or dynamic) for this DLR do not propagate to hosts

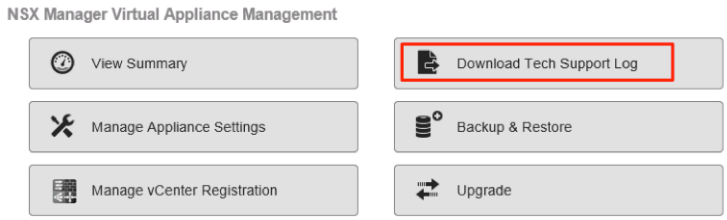
NSX Logs Relevant to Routing

The best practice is to configure all components of NSX to send their logs to a centralized collector, where they can be examined in one place.

If necessary, you can change the log level of NSX components. For more information, see "Setting the Logging Level of NSX Components" topic in *NSX Logging and System Events*.

NSX Manager Logs

- `show log` in the NSX Manager CLI
- Tech Support Log bundle, collected via the NSX Manager UI



The NSX Manager log contains information related to the management plane, which covers create, read, update, and delete (CRUD) operations.

Controller Logs

Controllers contain multiple modules, many with their own log files. Controller logs can be accessed using the `show log <log file> [ filtered-by <string> ]` command. The log files relevant to routing are as follows:

- `cloudnet/cloudnet_java-vnet-controller.<start-time-stamp>.log`: This log manages configuration and internal API server.
- `cloudnet/cloudnet.nsx-controller.log`: This is controller main process log.
- `cloudnet/cloudnet_cpp.log.nsx-controller.log`: This log manages clustering and bootstrap.
- `cloudnet/cloudnet_cpp.log.ERROR`: This file is present if any error occurs.

Controller logs are verbose and in most cases are only required when the VMware engineering team is brought in to assist with troubleshooting in more difficult cases.

In addition to the `show log` CLI, individual log files can be observed in real time as they are being updated, using the `watch log <logfile> [ filtered-by <string> ]` command.

The logs are included in the Controller support bundle that can be generated and downloaded by selecting a Controller node in the NSX UI and clicking the **Download tech support logs** icon.

ESXi Host Logs

NSX components running on ESXi hosts write several log files:

- VMkernel logs: `/var/log/vmkernel.log`
- Control Plane Agent logs: `/var/log/netcpa.log`
- Message Bus Client logs: `/var/log/vsfwd.log`

The logs can also be collected as part of the VM support bundle generated from vCenter Server. The log files are accessible only to the users or user groups having the *root* privilege.

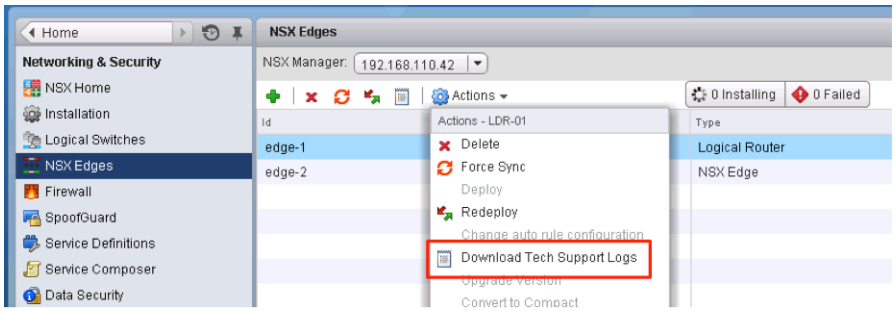
ESG/DLR Control VM Logs

There are two ways to access log files on the ESG and DLR Control VMs—to display them using a CLI or to download the tech support bundle, using the CLI or UI.

The CLI command to display logs is `show log [ follow | reverse ]`.

To download tech-support bundle:

- From the CLI, enter `enable` mode, then run the `export tech-support <[ scp | ftp ]> <URI>` command.
- From the vSphere Web Client, select the **Download Tech Support Logs** option in the **Actions** menu.



Other Useful Files and Their Locations

While not strictly logs, there are a number of files that can be helpful in understanding and troubleshooting NSX routing.

- The control plane agent configuration, `/etc/vmware/netcpa/config-by-vsm.xml` contains the information about the following components:
  - Controllers, IP addresses, TCP ports, certificate thumbprints, SSL enable/disable
  - dvUplinks on the DVS enabled with VXLAN (teaming policy, names, UUID)
  - DLR instances the host knows about (DLR ID, name)
- The control plane agent configuration, `/etc/vmware/netcpa/netcpa.xml` contains various configuration options for netcpa, including logging level (which by default is **info**).
- Control plane certificate files: `/etc/vmware/ssl/rui-for-netcpa.*`
  - Two files: host certificate and host private key
  - Used for authenticating host connections to Controllers

All of these files are created by control plane agent using information it receives from NSX Manager via the message bus connection provided by vsfwd.

Common Failure Scenarios and Fixes

The most common failure scenarios fall into two categories.

They are configuration and control-plane issues. Management plane issues, while possible, are not common.

Configuration Issues and Fixes

Common configuration issues and their effects are described in Table 3-7. Common Configuration Issues and Effects.

Table 3-7. Common Configuration Issues and Effects

Issues	Effects
Protocol and forwarding IP addresses are reversed for dynamic routing	Dynamic protocol adjacency won't come up
Transport zone is not aligned to the DVS boundary	Distributed routing does not work on a subset of ESXi hosts (those missing from the transport zone)
Dynamic routing protocol configuration mismatch (timers, MTU, BGP ASN, passwords, interface to OSPF area mapping)	Dynamic protocol adjacency does not come up
DLR HA interface is assigned an IP address and redistribution of connected routes is enabled	DLR Control VM might attract traffic for the HA interface subnet and blackhole the traffic

To resolve these issues, review the configuration and correct it as needed.

When necessary, use the `debug ip ospf` or `debug ip bgp` CLI commands and observe logs on the DLR Control VM or on the ESG console (not via SSH session) to detect protocol configuration issues.

Control-Plane Issues and Fixes

Control plane issues seen are often caused by the following issues:

- Host Control Plane Agent (netcpa) being unable to connect to NSX Manager through the message bus channel provided by vsfwd
- Controller cluster having issues with handling the master role for DLR/VXLAN instances

Controller cluster issues related to handling of master roles can often be resolved by restarting one of the NSX Controllers (`restart controller` on the Controller's CLI).

For more information about troubleshooting control-pane issues, see http://kb.vmware.com/kb/2125767.

Gathering Troubleshooting Data

This section provides a summary of the CLI commands that are commonly used for troubleshooting NSX routing.

NSX Manager

Starting in NSX 6.2, commands that were formerly run from the NSX Controller and other NSX components to troubleshoot NSX routing are now run directly from the NSX Manager.

- List of DLR instances
- List of LIFs for each DLR instance
- List of Routes for each DLR instance
- List of MAC addresses for each DLR bridging instance
- Interfaces
- Routing and forwarding tables
- State of dynamic routing protocols (OSPF or BGP)

- Configuration sent to the DLR Control VM or ESG by the NSX Manager

DLR Control VM and ESG

The DLR Control VM and ESG provide functionality to capture packets on their interfaces. Packet capture can assist with troubleshooting routing protocol problems.

1 Run `show interfaces` to list the interface names.

2 Run `debug packet [ display | capture ] interface <interface name>`.

   - If using capture, packets are saved into a `.pcap` file.

3 Run `debug show files` to list saved capture files.

4 Run `debug copy [ scp | ftp ] ...` to download captures for offline analysis.

```
dlr-01-0> debug packet capture interface vNic_2
tcpdump: listening on vNic_2, link-type EN10MB (Ethernet), capture size 65535 bytes
43 packets captured
48 packets received by filter
0 packets dropped by kernel
```

```
dlr-01-0> debug show files
total 4.0K
-rw------- 1 3.6K Mar 30 23:49 tcpdump_vNic_2.0
```

```
dlr-01-0> debug copy
  scp  use scp to copy
  ftp  use ftp to copy
```

```
dlr-01-0> debug copy scp
  URL  user@<remote-host>:<path-to>
```

The `debug packet` command uses tcpdump in the background and can accept filtering modifiers formatted in like tcpdump filtering modifiers on UNIX. The only consideration is that any white spaces in the filter expression need to be replaced with underscores ("_").

For example, the following command displays all traffic through vNic_0 except SSH, to avoid looking at the traffic belonging to the interactive session itself.

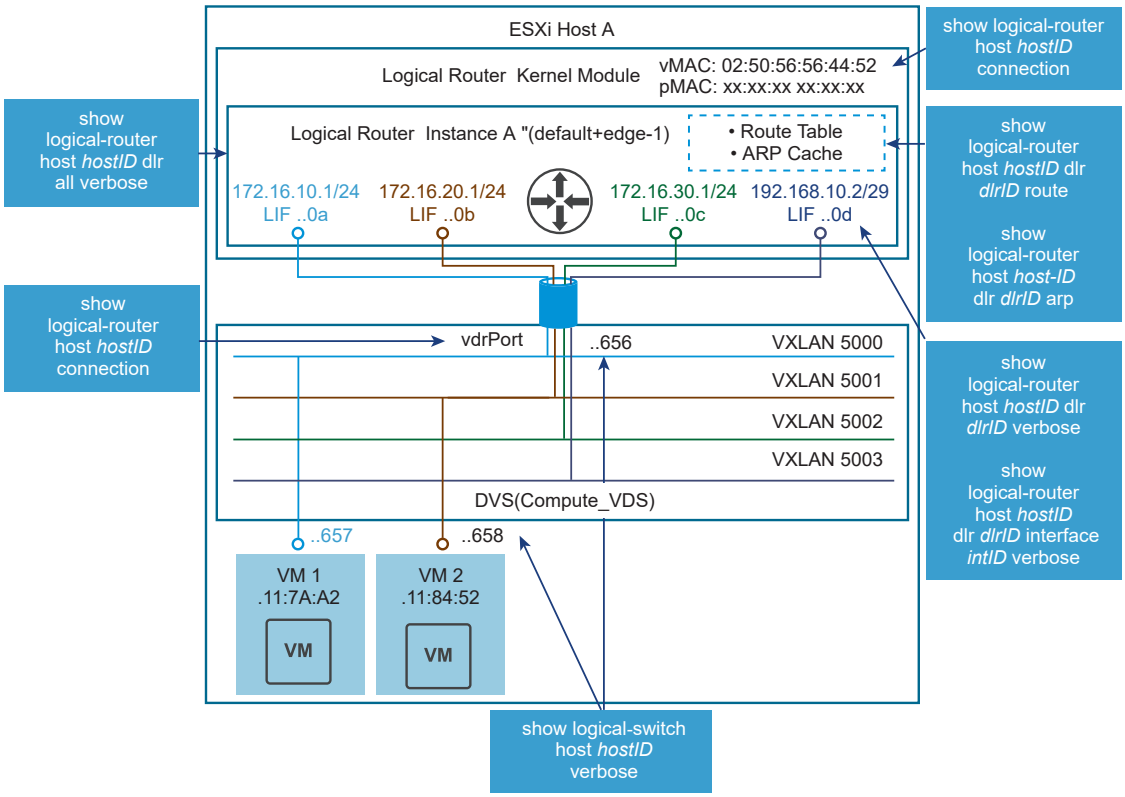
```
plr-02-0> debug packet display interface vNic_0 port_not_22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on vNic_0, link-type EN10MB (Ethernet), capture size 65535 bytes
04:10:48.197768 IP 192.168.101.3.179 > 192.168.101.2.25698: Flags [P.], seq 4191398894:4191398913,
ack 2824012766, win 913, length 19: BGP, length: 19
04:10:48.199230 IP 192.168.101.2.25698 > 192.168.101.3.179: Flags [.], ack 19, win 2623, length 0
04:10:48.299804 IP 192.168.101.2.25698 > 192.168.101.3.179: Flags [P.], seq 1:20, ack 19, win 2623,
```

```
length 19: BGP, length: 19
04:10:48.299849 IP 192.168.101.3.179 > 192.168.101.2.25698: Flags [.], ack 20, win 913, length 0
04:10:49.205347 IP 192.168.101.3.179 > 192.168.101.2.25698: Flags [P.], seq 19:38, ack 20, win 913,
length 19: BGP, length: 19
```

ESXi Hosts

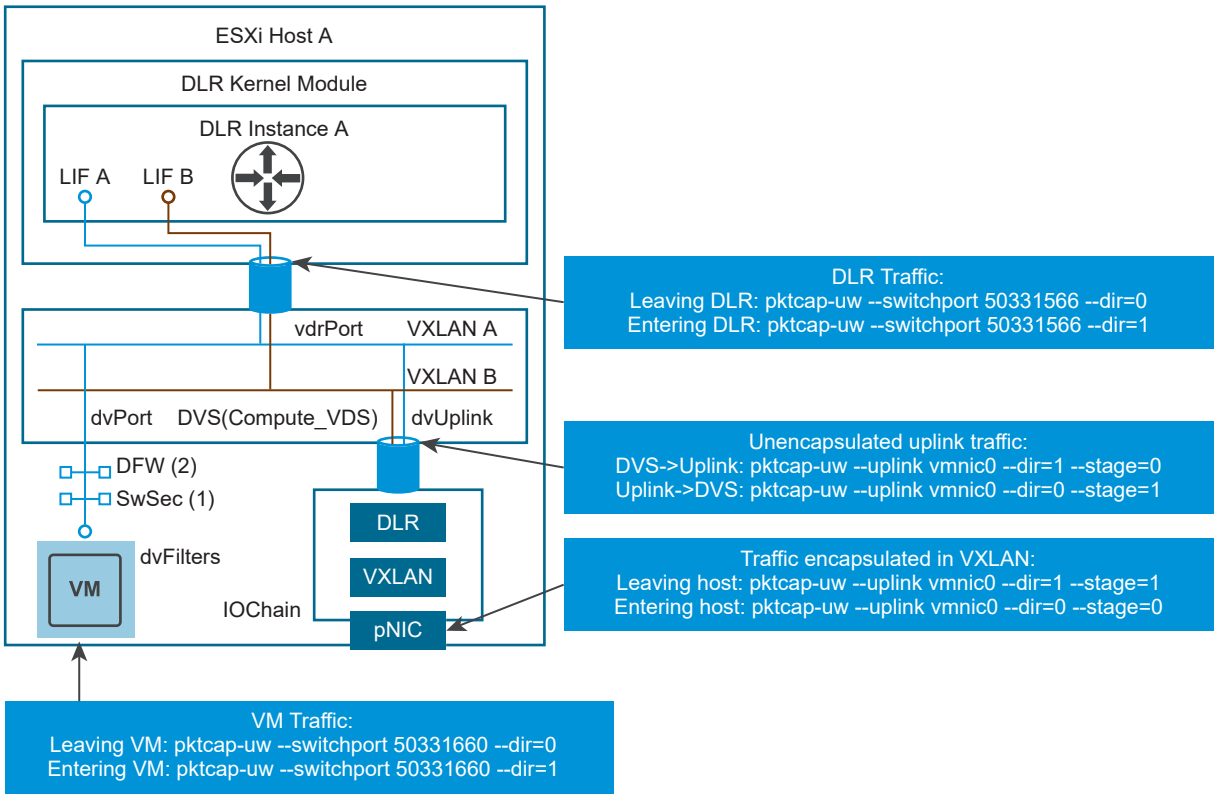
Hosts are closely connected to NSX Routing. Figure 3-14. Host Components Related to Troubleshooting NSX Routing shows visually the components participating in the routing subsystem and the NSX Manager CLI commands used to display information about them:

Figure 3-14. Host Components Related to Troubleshooting NSX Routing



Packets captured in the datapath can assist with identifying problems at various stages of packet forwarding. Figure 3-15. Capture Points and Related CLI Commands covers the major capture points and respective CLI command to use.

Figure 3-15. Capture Points and Related CLI Commands



Troubleshooting NSX Edge 4

This topic provides information for understanding and troubleshooting the VMware NSX Edge.

To troubleshoot issues with an NSX Edge appliance, validate that each troubleshooting step below is true for your environment. Each step provides instructions or a link to a document, to eliminate possible causes and take corrective action as necessary. The steps are ordered in the most appropriate sequence to isolate the issue and identify the proper resolution. Do not skip a step.

Check the release notes for current releases to see if the problem is resolved.

Ensure that the minimum system requirements are met when installing VMware NSX Edge. See the *NSX Installation Guide*.

Installation and Upgrade issues

- Verify that the issue you are encountering is not related to the "Would Block" issue. For more information, see https://kb.vmware.com/kb/2107951.
- If the upgrade or redeploy succeeds but there is no connectivity for the Edge interface, verify connectivity on the back-end Layer 2 switch. See https://kb.vmware.com/kb/2135285.
- If deployment or upgrade of the Edge fails with the error:

  ```
  /sbin/ifconfig vNic_1 up failed : SIOCSIFFLAGS: Invalid argument
  ```

  OR
- If the deployment or upgrade succeeds, but there is no connectivity on the Edge interfaces:
- Running the `show interface` command as well as Edge Support logs displays entries similar to:

  ```
  vNic_0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN qlen 1000
      link/ether 00:50:56:32:05:03 brd ff:ff:ff:ff:ff:ff
      inet 21.12.227.244/23 scope global vNic_0
      inet6 fe80::250:56ff:fe32:503/64 scope link tentative dadfailed
         valid_lft forever preferred_lft forever
  ```

  In both cases, the host switch is not ready or has some issues. To resolve, investigate the host switch.

Configuration Issues

- Collect the NSX Edge diagnostic information. See https://kb.vmware.com/kb/2079380.

  Filter the NSX Edge logs by searching for the string `vse_die`. The logs near this string might provide information about the configuration error.

High CPU Utilization

If you are experiencing high CPU utilization on the NSX Edge, verify the performance of the appliance using the `esxtop` command on the ESXi host. Review the following Knowledge Base articles:

- https://kb.vmware.com/kb/1008205
- https://kb.vmware.com/kb/1008014
- https://kb.vmware.com/kb/1010071
- https://kb.vmware.com/kb/2096171

Also see https://communities.vmware.com/docs/DOC-9279.

A high value for the `ksoftirqd` process indicates a high incoming packet rate. Check whether logging is enabled on the data path, such as for firewall rules. Run the `show log follow` command to determine whether a large number of log hits are being recorded.

Displaying Packet Drop Statistics

Starting with NSX for vSphere 6.2.3, you can use the command `show packet drops` to display packet drop statistics for the following:

- Interface
- Driver
- L2
- L3
- Firewall

To run the command, log in to the NSX Edge CLI and enter basic mode. For more information, see the *NSX Command Line Interface Reference*. For example:

```
show packet drops

vShield Edge Packet Drop Stats:

Driver Errors
=============
          TX      TX    TX   RX   RX      RX
Interface Dropped Error Ring Full Dropped Error Out Of Buf
vNic_0    0       0     0    0    0       0
vNic_1    0       0     0    0    0       0
```

```
vNic_2    0       0     0    0    0       2
vNic_3    0       0     0    0    0       0
vNic_4    0       0     0    0    0       0
vNic_5    0       0     0    0    0       0

Interface Drops
===============
Interface RX Dropped TX Dropped
vNic_0             4          0
vNic_1          2710          0
vNic_2             0          0
vNic_3             2          0
vNic_4             2          0
vNic_5             2          0

L2 RX Errors
============
Interface length crc frame fifo missed
vNic_0         0   0     0    0      0
vNic_1         0   0     0    0      0
vNic_2         0   0     0    0      0
vNic_3         0   0     0    0      0
vNic_4         0   0     0    0      0
vNic_5         0   0     0    0      0

L2 TX Errors
============
Interface aborted fifo window heartbeat
vNic_0          0    0      0         0
vNic_1          0    0      0         0
vNic_2          0    0      0         0
vNic_3          0    0      0         0
vNic_4          0    0      0         0
vNic_5          0    0      0         0

L3 Errors
=========
IP:
 ReasmFails : 0
 InHdrErrors : 0
 InDiscards : 0
 FragFails : 0
 InAddrErrors : 0
 OutDiscards : 0
 OutNoRoutes : 0
 ReasmTimeout : 0
ICMP:
 InTimeExcds : 0
 InErrors : 227
 OutTimeExcds : 0
 OutDestUnreachs : 152
 OutParmProbs : 0
 InSrcQuenchs : 0
 InRedirects : 0
 OutSrcQuenchs : 0
```

```
 InDestUnreachs : 151
 OutErrors : 0
 InParmProbs : 0

Firewall Drop Counters
======================

Ipv4 Rules
==========
Chain – INPUT
rid pkts bytes target prot opt in out source    destination
0    119 30517 DROP   all  --   *   * 0.0.0.0/0 0.0.0.0/0    state INVALID
0      0     0 DROP   all  --   *   * 0.0.0.0/0 0.0.0.0/0
Chain – POSTROUTING
rid pkts bytes target prot opt in out source    destination
0    101 4040  DROP   all   --  *   * 0.0.0.0/0 0.0.0.0/0    state INVALID
0      0    0  DROP   all   --  *   * 0.0.0.0/0 0.0.0.0/0

Ipv6 Rules
==========
Chain – INPUT
rid pkts bytes target prot opt in out source destination
0      0     0   DROP  all      *   * ::/0   ::/0            state INVALID
0      0     0   DROP  all      *   * ::/0   ::/0
Chain – POSTROUTING
rid pkts bytes target prot opt in out source destination
0      0     0   DROP  all       *   * ::/0   ::/0           state INVALID
0      0     0   DROP  all       *   * ::/0   ::/0
```

Expected Behavior When Managing NSX Edge

- In vSphere Web Client, when you configure L2 VPN on an ESX Edge and add, remove, or modify **Site Configuration Details**, this action will cause all existing connections to be disconnected and reconnected. This behavior is expected.
- NSX Edge is a virtual machine (VM) and consists of several files that are stored on a storage device. The key files are the configuration file, virtual disk file(s), NVRAM setting file, swap file, and log file. Based upon the VM Storage Profile applied or manual placement, the virtual machine configuration files, virtual disk file, swap file can be placed in the same location, or in separate locations on different datastores. In the case where the virtual machine files are present in different locations, NSX Manager shows and uses the datastore which has the VMX file for the VM deployment. During redeployment or upgrade operations, NSX Manager deploys the NSX Edge VM(s) on the configured datastore or the live datastore which hosts the VMX files. The *datastore name* and the *datastore ID* (which hosts VMX file of the VM) are returned as part of the `Appliance` parameter, and is displayed on the UI or provided as response to REST API. You must refer to vCenter Server for details on the exact layout each of the NSX Manager VM files and one or more datastores where the files are placed. For more information, refer to the following documentation:
  - *vSphere Virtual Machine Administrator* .
  - *vSphere Resource Management* .

- *vCenter Server and Host Management* .

This chapter includes the following topics:

Edge Firewall Packet Drop Issues

Display Firewall Packet Drop Statistics

Starting with NSX for vSphere 6.2.3, you can use the command `show packet drops` to display packet drop statistics for the firewall.

To run the command, log in to the NSX Edge CLI and enter basic mode. For more information, see the *NSX Command Line Interface Reference*. For example:

```
show packet drops

vShield Edge Packet Drop Stats:


Firewall Drop Counters
======================

Ipv4 Rules
==========
Chain - INPUT
rid pkts bytes target prot opt in out source    destination
0    119 30517 DROP   all  --   *   * 0.0.0.0/0 0.0.0.0/0    state INVALID
0      0     0 DROP   all  --   *   * 0.0.0.0/0 0.0.0.0/0
Chain - POSTROUTING
rid pkts bytes target prot opt in out source    destination
0    101 4040  DROP   all   --  *   * 0.0.0.0/0 0.0.0.0/0    state INVALID
0      0    0  DROP   all   --  *   * 0.0.0.0/0 0.0.0.0/0

Ipv6 Rules
==========
Chain - INPUT
rid pkts bytes target prot opt in out source destination
0      0     0   DROP  all      *   * ::/0   ::/0            state INVALID
0      0     0   DROP  all      *   * ::/0   ::/0
Chain - POSTROUTING
rid pkts bytes target prot opt in out source destination
0      0     0   DROP  all       *   * ::/0   ::/0           state INVALID
0      0     0   DROP  all       *   * ::/0   ::/0
```

Edge Packet Firewall Issues

To run a command, log in to the NSX Edge CLI and enter basic mode. For more information, see the *NSX Command Line Interface Reference*.

1 Check the firewall rules table with the `show firewall` command. The `usr_rules` table displays the configured rules.

```
nsxedge> show firewall
Chain PREROUTING (policy ACCEPT 3146M packets, 4098G bytes)
rid    pkts bytes target     prot opt in     out     source               destination

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
rid    pkts bytes target     prot opt in     out     source               destination
0     78903   16M ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
0         0     0 DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0
state INVALID
0      140K 9558K block_in   all  --  *      *       0.0.0.0/0            0.0.0.0/0
0     23789 1184K ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
state RELATED,ESTABLISHED
0      116K 8374K usr_rules  all  --  *      *       0.0.0.0/0            0.0.0.0/0
0         0     0 DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT 3146M packets, 4098G bytes)
rid    pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 173K packets, 22M bytes)
rid    pkts bytes target     prot opt in     out     source               destination

Chain POSTROUTING (policy ACCEPT 0 packets, 0 bytes)
rid    pkts bytes target     prot opt in     out     source               destination
0     78903   16M ACCEPT     all  --  *      lo      0.0.0.0/0            0.0.0.0/0
0      679K   41M DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0
state INVALID
0     3146M 4098G block_out  all  --  *      *       0.0.0.0/0            0.0.0.0/0
0         0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
PHYSDEV match --physdev-in tap0 --physdev-out vNic_+
0         0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
PHYSDEV match --physdev-in vNic_+ --physdev-out tap0
0         0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
PHYSDEV match --physdev-in na+ --physdev-out vNic_+
0         0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
PHYSDEV match --physdev-in vNic_+ --physdev-out na+
0     3145M 4098G ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
state RELATED,ESTABLISHED
0      221K   13M usr_rules  all  --  *      *       0.0.0.0/0            0.0.0.0/0
0         0     0 DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain block_in (1 references)
rid    pkts bytes target     prot opt in     out     source               destination

Chain block_out (1 references)
rid    pkts bytes target     prot opt in     out     source               destination
```

```
Chain usr_rules (2 references)
rid    pkts bytes target     prot opt in     out     source               destination
131074 70104 5086K ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
match-set 0_131074-os-v4-1 src
131075  116K 8370K ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
match-set 1_131075-ov-v4-1 dst
131073  151K 7844K ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
```

Check for an incrementing value of a `DROP invalid` rule in the `POST_ROUTING` section of the `show firewall` command. Typical reasons include:

- Asymmetric routing issues
- TCP-based applications that have been inactive for more than one hour. If there are inactivity time-out issues and applications are idle for an unusually long time, increase inactivity-timeout settings using the REST API. See https://kb.vmware.com/kb/2101275

2 Collect the `show ipset` command output.

```
nsxedge> show ipset
Name: 0_131074-os-v4-1
Type: bitmap:if (Interface Match)
Revision: 3
Header: range 0-64000
Size in memory: 8116
References: 1
Number of entries: 1
Members:
vse (vShield Edge Device)

Name: 0_131074-os-v6-1
Type: bitmap:if (Interface Match)
Revision: 3
Header: range 0-64000
Size in memory: 8116
References: 1
Number of entries: 1
Members:
vse (vShield Edge Device)

Name: 1_131075-ov-v4-1
Type: hash:oservice (Match un-translated Ports)
Revision: 2
Header: family inet hashsize 64 maxelem 65536
Size in memory: 704
References: 1
Number of entries: 2
Members:
Proto=6, DestPort=179, SrcPort=Any    (encoded: 0.6.0.179,0.6.0.0/16)
Proto=89, DestPort=Any, SrcPort=Any    (encoded: 0.89.0.0/16,0.89.0.0/16)

Name: 1_131075-ov-v6-1
Type: hash:oservice (Match un-translated Ports)
```

```
Revision: 2
Header: family inet hashsize 64 maxelem 65536
Size in memory: 704
References: 1
Number of entries: 2
Members:
Proto=89, DestPort=Any, SrcPort=Any    (encoded: 0.89.0.0/16,0.89.0.0/16)
Proto=6, DestPort=179, SrcPort=Any    (encoded: 0.6.0.179,0.6.0.0/16)
```

3 Enable logging on a particular firewall rule using the REST API or the Edge user interface, and monitor the logs with the `show log follow` command.

If logs are not seen, enable logging on the `DROP Invalid` rule using the following REST API.

```
URL : https://NSX_Manager_IP/api/4.0/edges/{edgeId}/firewall/config/global

PUT Method
Input representation
<globalConfig>   <!-- Optional -->
<tcpPickOngoingConnections>false</tcpPickOngoingConnections>   <!-- Optional. Defaults to false --
>
<tcpAllowOutOfWindowPackets>false</tcpAllowOutOfWindowPackets>    <!-- Optional. Defaults to
false -->
<tcpSendResetForClosedVsePorts>true</tcpSendResetForClosedVsePorts>    <!-- Optional. Defaults to
true -->
<dropInvalidTraffic>true</dropInvalidTraffic>    <!-- Optional. Defaults to true -->
<logInvalidTraffic>true</logInvalidTraffic>     <!-- Optional. Defaults to false -->
<tcpTimeoutOpen>30</tcpTimeoutOpen>       <!-- Optional. Defaults to 30 -->
<tcpTimeoutEstablished>3600</tcpTimeoutEstablished>   <!-- Optional. Defaults to 3600 -->
<tcpTimeoutClose>30</tcpTimeoutClose>   <!-- Optional. Defaults to 30 -->
<udpTimeout>60</udpTimeout>             <!-- Optional. Defaults to 60 -->
<icmpTimeout>10</icmpTimeout>           <!-- Optional. Defaults to 10 -->
<icmp6Timeout>10</icmp6Timeout>           <!-- Optional. Defaults to 10 -->
<ipGenericTimeout>120</ipGenericTimeout>    <!-- Optional. Defaults to 120 -->
</globalConfig>
Output representation
No payload
```

Use the `show log follow` command to look for logs similar to:

```
2016-04-18T20:53:31+00:00 edge-0 kernel: nf_ct_tcp: invalid TCP flag combination IN= OUT=
SRC=172.16.1.4 DST=192.168.1.4 LEN=40 TOS=0x00 PREC=0x00 TTL=64 ID=43343 PROTO=TCP
SPT=5050 DPT=80 SEQ=0 ACK=1572141176 WINDOW=512 RES=0x00 URG PSH FIN URGP=0
2016-04-18T20:53:31+00:00 edge-0 kernel: INVALID IN= OUT=vNic_1 SRC=172.16.1.4
DST=192.168.1.4 LEN=40 TOS=0x00 PREC=0x00 TTL=63 ID=43343 PROTO=TCP SPT=5050 DPT=80
WINDOW=512 RES=0x00 URG PSH FIN URGP=0
```

4 Check for matching connections in the Edge firewall state table with the `show flowtable rule_id` command:

```
nsxedge> show flowtable
```

```
1: tcp  6 21554 ESTABLISHED src=192.168.110.10 dst=192.168.5.3 sport=25981
d port=22 pkts=52 bytes=5432 src=192.168.5.3 dst=192.168.110.10 sport=22 dport=259
81 pkts=44 bytes=7201 [ASSURED] mark=0 rid=131073 use=1
2: tcp  6 21595 ESTABLISHED src=127.0.0.1 dst=127.0.0.1 sport=53194
dport=10 001 pkts=33334 bytes=11284650 src=127.0.0.1 dst=127.0.0.1 sport=10001 dport=5319
4 pkts=33324 bytes=1394146 [ASSURED] mark=0 rid=0 use=1
```

Compare the active connection count and the maximum allowed count with the `show flowstats` command:

```
nsxedge> show flowstats
Total Flow Capacity: 65536
Current Statistics :
cpu=0 searched=3280373 found=3034890571 new=52678 invalid=659946 ignore=77605
delete=52667 delete_list=49778 insert=49789 insert_failed=0 drop=0 early_drop=0
error=0 search_restart=0
```

5 Check the Edge logs with the `show log follow` command, and look for any ALG drops. Search for strings similar to `tftp_alg`, `msrpc_alg`, or `oracle_tns`. For additional information, see:

- https://kb.vmware.com/kb/2126674
- https://kb.vmware.com/kb/2137751

Edge Routing Connectivity Issues

1 Initiate controlled traffic from a client using the `ping <destination_IP_address>` command.

2 Capture traffic simultaneously on both interfaces, write the output to a file, and export it using SCP.

For example:

Capture the traffic on the ingress interface with this command:

```
debug packet display interface vNic_0 -n_src_host_1.1.1.1
```

Capture the traffic on the egress interface with this command:

```
debug packet display interface vNic_1 -n_src_host_1.1.1.1
```

For simultaneous packet capture, use the ESXi packet capture utility `pktcap-uw` tool in ESXi. See https://kb.vmware.com/kb/2051814.

If the packet drops are consistent, check for configuration errors related to:

- IP addresses and routes
- Firewall rules or NAT rules
- Asymmetric routing
- RP filter checks

a Check interface IP/subnets with the `show interface` command.

b If there are missing routes at the data plane, run these commands:

- `show ip route`
- `show ip route static`
- `show ip route bgp`
- `show ip route ospf`

c Check the routing table for needed routes by running the `show ip forwarding` command.

d If you have multiple paths, run the `show rpfilter` command.

```
nsxedge> show rpfilter
net.ipv4.conf.VDR.rp_filter = 0
net.ipv4.conf.all.rp_filter = 0
net.ipv4.conf.br-sub.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.lo.rp_filter = 0
net.ipv4.conf.vNic_0.rp_filter = 1
net.ipv4.conf.vNic_1.rp_filter = 1
net.ipv4.conf.vNic_2.rp_filter = 1
net.ipv4.conf.vNic_3.rp_filter = 1
net.ipv4.conf.vNic_4.rp_filter = 1
net.ipv4.conf.vNic_5.rp_filter = 1
net.ipv4.conf.vNic_6.rp_filter = 1
net.ipv4.conf.vNic_7.rp_filter = 1
net.ipv4.conf.vNic_8.rp_filter = 1
net.ipv4.conf.vNic_9.rp_filter = 1

nsxedge> show rpfstats
RPF drop packet count: 484
```

To check for RPF statistics, run the `show rpfstats` command.

```
nsxedge> show rpfstats
RPF drop packet count: 484
```

If the packet drops appear randomly, check for resource limitations:

a For CPU or memory usage, run these commands:

- `show system cpu`
- `show system memory`
- `show system storage`
- `show process monitor`
- `top`

  For ESXi, run the `esxtop n` command.

```
PCPU USED(%): 2.5 5.0 3.7  77 AVG:  22
PCPU UTIL(%): 0.5 2.7 3.3  92 AVG:  24

      ID      GID NAME             NWLD   %USED    %RUN    %SYS   %WAIT
98255269 98255269 esxtop.11224149     1   67.04   69.86    0.00    6.26
       2        2 system             139    3.03    4.61    0.00 12053.58
   86329    86329 app-01a              6    0.69    0.57    0.00  466.09
   78730    78730 db-01a               6    0.48    0.67    0.00  441.44
   90486    90486 app-02a              6    0.38    0.32    0.00  463.42

 %VMWAIT    %RDY    %IDLE    %OVRLP    %CSTP   %MLMTD    %SWPWT
 11.01       -    0.39    0.00    0.09    0.00    0.00    0.00
 600.00   53.81    0.10   93.13    0.00    0.00    0.00    0.00
 13900.00       -   28.68    0.00    2.69    0.00    0.00    0.00
 600.00   53.81    0.10   93.13    0.00    0.00    0.00    0.00
 600.00    0.00    0.19  151.92    0.00    0.00    0.00    0.00
```

NSX Manager and Edge Communication Issues

The NSX Manager communicates with NSX Edge through the VIX or Message Bus. It is chosen by the NSX Manager when the Edge is deployed and never changes.

Note VIX is not supported in NSX 6.3.0 and later.

VIX

- VIX is used for NSX Edge if the ESXi host is not prepared.
- The NSX Manager gets host credentials from the vCenter Server to connect to the ESXi host first.
- The NSX Manager uses the Edge credentials to log in to the Edge appliance.
- The `vmtoolsd` process on the Edge handles the VIX communication.

VIX failures occur because of:

- The NSX Manager cannot communicate with the vCenter Server.
- The NSX Manager cannot communicate with the ESXi hosts.
- There are NSX Manager internal issues.
- There are Edge internal issues.

VIX Debugging

Check for VIX errors VIX_E_<error> in the NSX Manager logs to narrow down the cause. Look for errors similar to:

```
Vix Command 1126400 failed, reason com.vmware.vshield.edge.exception.VixException: vShield
Edge:10013:Error code 'VIX_E_FILE_NOT_FOUND' was returned by VIX API.:null

Health check failed for edge  edge-13 VM vm-5025 reason:
com.vmware.vshield.edge.exception.VixException: vShield Edge:10013:Error code
'VIX_E_VM_NOT_RUNNING' was returned by VIX API.:null
```

In general, if the same failure occurs for many Edges at the same time, the issue is not on the Edge side.

Message Bus Debugging

The Message Bus is used for NSX Edge communication when ESXi hosts are prepared.

When you encounter issues, the NSX Manager logs might contain entries similar to:

```
GMT ERROR taskScheduler-6 PublishTask:963 - Failed to configure VSE-vm index 0, vm-id vm-117,
edge edge-5. Error: RPC request timed out
```

This issue occurs if:

- Edge is in a bad state
- Message Bus connection is broken

To diagnose the issue on the Edge:

- To check rmq connectivity, run this command:

  ```
  nsxedge> show messagebus messages
  ----------------------
  Message bus is enabled
  cmd conn state : listening
  init_req       : 1
  init_resp      : 1
  init_req_err   : 0
  ...
  ```

- To check vmci connectivity, run this command:

  ```
  nsxedge> show messagebus forwarder
  ----------------------
  Forwarder Command Channel
  vmci_conn          : up
  app_client_conn    : up
  vmci_rx            : 3649
  vmci_tx            : 3648
  ```

```
vmci_rx_err        : 0
vmci_tx_err        : 0
vmci_closed_by_peer: 8
vmci_tx_no_socket  : 0
app_rx             : 3648
app_tx             : 3649
app_rx_err         : 0
app_tx_err         : 0
app_conn_req       : 1
app_closed_by_peer : 0
app_tx_no_socket   : 0
-----------------------
Forwarder Event Channel
vmci_conn          : up
app_client_conn    : up
vmci_rx            : 1143
vmci_tx            : 13924
vmci_rx_err        : 0
vmci_tx_err        : 0
vmci_closed_by_peer: 0
vmci_tx_no_socket  : 0
app_rx             : 13924
app_tx             : 1143
app_rx_err         : 0
app_tx_err         : 0
app_conn_req       : 1
app_closed_by_peer : 0
app_tx_no_socket   : 0
-----------------------
cli_rx             : 1
cli_tx             : 1
cli_tx_err         : 0
counters_reset     : 0
```

In the example, the output `vmci_closed_by_peer: 8` indicates the number of times the connection has been closed by the host agent. If this number is increasing and `vmci conn` is down, the host agent cannot connect to the RMQ broker. In `show log follow`, look for repeated errors in the Edge logs: `VmciProxy: [daemon.debug] VMCI Socket is closed by peer`

To diagnose the issue on the ESXi host:

- To check if the ESXi host connects to the RMQ broker, run this command:

```
esxcli network ip connection list | grep 5671

tcp   0   0  10.32.43.4:43329  10.32.43.230:5671    ESTABLISHED     35854  newreno
vsfwd
tcp   0   0  10.32.43.4:52667  10.32.43.230:5671    ESTABLISHED     35854  newreno
vsfwd
tcp   0   0  10.32.43.4:20808  10.32.43.230:5671    ESTABLISHED     35847  newreno
vsfwd
tcp   0   0  10.32.43.4:12486  10.32.43.230:5671    ESTABLISHED     35847  newreno  vsfwd
```

Edge Diagnosis and Recovery

Edge Diagnosis

- Check if `vmtoolsd` is running with this command:

```
nsxedge> show process list
Perimeter-Gateway-01-0> show process list
%CPU %MEM    VSZ   RSZ STAT  STARTED     TIME COMMAND
 0.0  0.1   4244   720 Ss     May 16 00:00:15 init [3]
...
 0.0  0.1   4240   640 S      May 16 00:00:00 logger -p daemon debug -t vserrdd
 0.2  0.9  57192  4668 S      May 16 00:23:07 /usr/local/bin/vmtoolsd --plugin-pa
 0.0  0.4   4304  2260 SLs    May 16 00:01:54 /usr/sbin/watchdog
 ...
```

- Check if Edge is in a good state by running this command:

```
nsxedge> show eventmgr
------------------------
messagebus     : enabled
debug          : 0
profiling      : 0
cfg_rx         : 1
cfg_rx_msgbus  : 0
...
```

  Use the `show eventmgr` command to verify that the query command is received and processed.

```
nsxedge> show eventmgr
------------------------
messagebus     : enabled
debug          : 0
profiling      : 0
cfg_rx         : 1
cfg_rx_msgbus  : 0
cfg_rx_err     : 0
cfg_exec_err   : 0
cfg_resp       : 0
cfg_resp_err   : 0
cfg_resp_ln_err: 0
fastquery_rx   : 0
fastquery_err  : 0
clearcmd_rx    : 0
clearcmd_err   : 0
ha_rx          : 0
ha_rx_err      : 0
ha_exec_err    : 0
status_rx      : 16
status_rx_err  : 0
status_svr     : 10
```

```
status_evt     : 0
status_evt_push: 0
status_ha      : 0
status_ver     : 1
status_sys     : 5
status_cmd     : 0
status_svr_err : 0
status_evt_err : 0
status_sys_err : 0
status_ha_err  : 0
status_ver_err : 0
status_cmd_err : 0
evt_report     : 1
evt_report_err : 0
hc_report      : 10962
hc_report_err  : 0
cli_rx         : 2
cli_resp       : 1
cli_resp_err   : 0
counter_reset  : 0
---------- Health Status -------------
system status  : good
ha state       : active
cfg version    : 7
generation     : 0
server status  : 1
syslog-ng      : 1
haproxy        : 0
ipsec          : 0
sslvpn         : 0
l2vpn          : 0
dns            : 0
dhcp           : 0
heartbeat      : 0
monitor        : 0
gslb           : 0
---------- System Events -------------
```

Edge Recovery

If the `vmtoolsd` is not running or the NSX Edge is in a bad state, reboot the edge.

To recover from a crash, a reboot should be sufficient. A redeploy should not be required.

Note Note down all logging information from the old edge when a redeploy is done.

To debug a kernel crash, you need to obtain:

- Either the vmss (VM suspend) or vmsn (VM snapshot) file for the edge VM while it is still in the crashed state. If there is a vmem file, this is also needed. This can be use to extract a kernel core dump file, which VMware Support can analyze.
- The Edge support log, generated right after the crashed edge has been rebooted (but not redeployed). You can also check the edge logs. See https://kb.vmware.com/kb/2079380.

- A screen shot of the Edge console is also helpful, although this does not usually contain the complete crash report.

Troubleshooting Firewall

5

This section provides information about troubleshooting firewall issues.

This chapter includes the following topics:

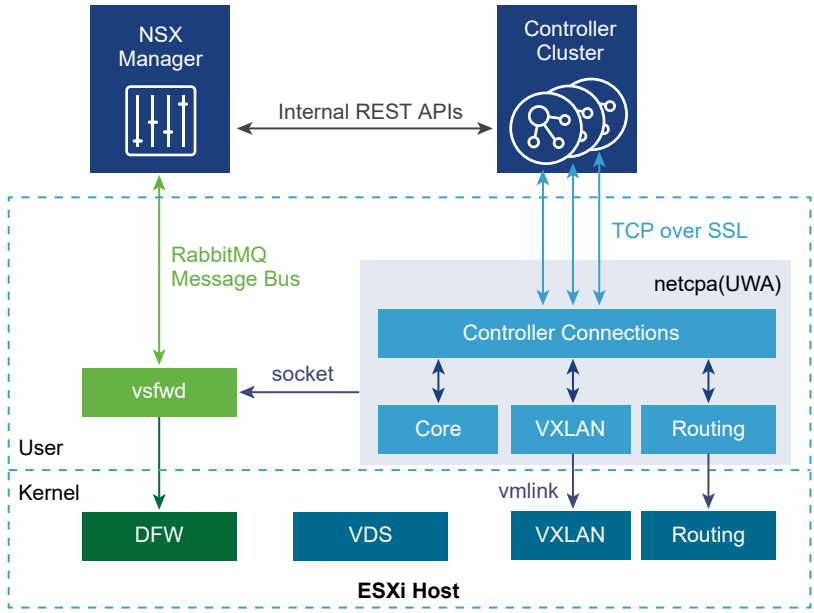
About Distributed Firewall

A RabbitMQ message bus is leveraged for communication between the vsfwd (RMQ client) and RMQ server process hosted on the NSX manager. The message bus is used by the NSX manager to send various information to the ESXi hosts, including policy rules that need to be programmed on the distributed firewall in the kernel.

NSX Distributed Firewall is a hypervisor kernel-embedded firewall that provides visibility and control for virtualized workloads and networks. You can create access control policies based on VMware vCenter objects like datacenters and clusters, virtual machine names and tags, network constructs such as IP/VLAN/VXLAN addresses, as well as user group identity from Active Directory. Consistent access control policy is now enforced when a virtual machine gets vMotioned across physical hosts without the need to rewrite firewall rules. Since Distributed Firewall is hypervisor-embedded, it delivers close to line rate throughput to enable higher workload consolidation on physical servers. The distributed nature of the firewall provides a scale-out architecture that automatically extends firewall capacity when additional hosts are added to a datacenter.

The NSX Manager web application and NSX components on ESXi hosts communicate with each other through a RabbitMQ broker process that runs on the same virtual machine as the NSX Manager web application. The communication protocol that is used is AMQP (Advanced Message Queueing Protocol) and the channel is secured using SSL. On an ESXi host, the VSFWD (vShield Firewall Daemon) process establishes and maintains the SSL connection to the broker and sends and receives messages on behalf of other components, which talks to it through IPC.

Figure 5-1. ESXi Host User and Kernel Space Diagram



CLI Commands for DFW

You can get most information about distributed firewalls on the NSX Manager central CLI.

Using the `Show dfw` Central CLI Commands

The path to drill down to the desired information is as follows:

1 Log in to the NSX Manager central CLI using the *admin* credentials.

2 Run the following commands:

a Run the `show cluster all`command to show all clusters.

```
nsxmgr>show cluster all
No.  Cluster Name                Cluster Id     Datacenter Name     Firewall Status
1    Compute Cluster A           domain-c33     Datacenter Site A   Enabled
2    Management & Edge Cluster   domain-c41     Datacenter Site A   Enabled
```

b Run the `show cluster <clusterID>` command to show hosts in a specific cluster.

```
nsxmgr> show cluster domain-c33
Datacenter: Datacenter Site A
Cluster: Compute Cluster A
No.  Host Name            Host Id         Installation Status
1    esx-02a.corp.local   host-32          Enabled
2    esx-01a.corp.local   host-28          Enabled
```

c Run the `show host <hostID>` to show all VMs on a host.

```
nsxmgr> show host host-28
Datacenter: Datacenter Site A
Cluster: Compute Cluster A
Host: esx-01a.corp.local
No.  VM Name    VM Id     Power Status
1    web-02a    vm-219    on
2    web-01a    vm-216    on
3    win8-01a   vm-206    off
4    app-02a    vm-264    on
```

d Run the `show vm <vmID>` command to show information for a VM, which includes filter names and vNIC IDs:

```
nsxmgr> show vm vm-264
Datacenter: Datacenter Site A
Cluster: Compute Cluster A
Host: esx-01a.corp.local
Host-ID: host-28
VM: app-02a
Virtual Nics List:
1.
Vnic Name      app-02a - Network adapter 1
Vnic Id        502ef2fa-62cf-d178-cb1b-c825fb300c84.000
Filters        nic-79396-eth0-vmware-sfw.2
```

e Note the vNIC ID and run further commands like `show dfw vnic <vnicID>` and `show dfw host <hostID> filter <filter ID> rules`:

```
nsxmgr> show dfw vnic 502ef2fa-62cf-d178-cb1b-c825fb300c84.000
Vnic Name      app-02a - Network adapter 1
Vnic Id        502ef2fa-62cf-d178-cb1b-c825fb300c84.000
Mac Address    00:50:56:ae:6c:6b
Port Group Id  dvportgroup-385
Filters        nic-79396-eth0-vmware-sfw.2

nsxmgr> show dfw host host-28 filter nic-79396-eth0-vmware-sfw.2 rules
ruleset domain-c33 {
  # Filter rules
  rule 1012 at 1 inout protocol any from addrset ip-securitygroup-10 to addrset ip-
securitygroup-10 drop with log;
  rule 1013 at 2 inout protocol any from addrset src1013 to addrset src1013 drop;
  rule 1011 at 3 inout protocol tcp from any to addrset dst1011 port 443 accept;
  rule 1011 at 4 inout protocol icmp icmptype 8 from any to addrset dst1011 accept;
  rule 1010 at 5 inout protocol tcp from addrset ip-securitygroup-10 to addrset ip-
securitygroup-11 port 8443 accept;
  rule 1010 at 6 inout protocol icmp icmptype 8 from addrset ip-securitygroup-10 to addrset
```

```
ip-securitygroup-11 accept;
  rule 1009 at 7 inout protocol tcp from addrset ip-securitygroup-11 to addrset ip-
securitygroup-12 port 3306 accept;
  rule 1009 at 8 inout protocol icmp icmptype 8 from addrset ip-securitygroup-11 to addrset
ip-securitygroup-12 accept;
  rule 1003 at 9 inout protocol ipv6-icmp icmptype 136 from any to any accept;
  rule 1003 at 10 inout protocol ipv6-icmp icmptype 135 from any to any accept;
  rule 1002 at 11 inout protocol udp from any to any port 67 accept;
  rule 1002 at 12 inout protocol udp from any to any port 68 accept;
  rule 1001 at 13 inout protocol any from any to any accept;
}

ruleset domain-c33_L2 {
  # Filter rules
  rule 1004 at 1 inout ethertype any from any to any accept;
}
```

Using the `export host-tech-support` Central CLI command

The `export host-tech-support` command allows you to export an ESXi host support bundle to a specified server. In addition, this command collects NSX related outputs and files (not limited to the following) on specified hosts such as:

- VMKernel and vsfwd log files
- List of filters
- List of DFW rules
- List of containers
- SpoofGuard details
- Host related information
- IP discovery related information
- RMQ command outputs
- Security group, services profile, and instance details
- ESX CLI related outputs

This command also removes any temporary files on the NSX Manager.

To collect NSX related outputs and files:

1 Log in to the NSX Manager central CLI using the *admin* credentials.

2 Run the following commands:

   a `show cluster all`- To find the required host ID.

   b `export host-tech-support host-id scp uid@ip:/path` - To generate the NSX technical support bundle and to copy it to a specified server.

For more information, refer to:

- NSX Command Line Quick Reference.
- *NSX Command Line Interface Reference*.

Troubleshooting Distributed Firewall

This topic provides information on understanding and troubleshooting VMware NSX 6.x Distributed Firewall (DFW).

Problem

- Publishing Distributed Firewall rules fails.
- Updating Distributed Firewall rules fails.

Cause

Validate that each troubleshooting step below is true for your environment. Each step provides instructions or a link to a document to eliminate possible causes and take corrective action as necessary. The steps are ordered in the most appropriate sequence to isolate the issue and identify the proper resolution. After each step, re-attempt to update/publish the Distributed Firewall rules.

Solution

1 Verify that the NSX VIBs are successfully installed on each of the ESXi hosts in the cluster. To do this, on each of the ESXi host that is on the cluster, run these commands.

```
# esxcli software vib list | grep vsip
esx-vsip                       6.0.0-0.0.4744062  VMware  VMwareCertified   2017-01-04

# esxcli software vib list | grep vxlan
esx-vxlan                      6.0.0-0.0.4744062  VMware  VMwareCertified   2017-01-04
```

NSX versions before NSX 6.2 have an additional VIB:

```
# esxcli software vib list | grep dvfilter

esx-dvfilter-switch-security   5.5.0-0.0.2318233  VMware  VMwareCertified   2015-01-24
```

Starting in NSX 6.3.3 with ESXi 6.0 or later, the esx-vxlan and esx-vsip VIBs are replaced with esx-nsxv.

```
# esxcli software vib list | grep nsxv
esx-nsxv                       6.0.0-0.0.6216823  VMware  VMwareCertified   2017-08-10
```

2 On the ESXi hosts, verify the vShield-Stateful-Firewall service is in a running state.

For example:

```
# /etc/init.d/vShield-Stateful-Firewall status

vShield-Stateful-Firewall is running
```

3 Verify that the Message Bus is communicating properly with the NSX Manager.

The process is automatically launched by the watchdog script and restarts the process if it terminates for an unknown reason. Run this command on each of the ESXi hosts on the cluster.

For example:

```
# ps | grep vsfwd

107557 107557 vsfwd /usr/lib/vmware/vsfw/vsfwd
```

There should be at least 12 or more vsfwd processes running in the command output. If there are less (most likely only 2) processes running, vsfwd is not running correctly.

4 Verify that port 5671 is opened for communication in the firewall configuration.

This command shows the VSFWD connectivity to the RabbitMQ broker. Run this command on ESXi hosts to see a list of connections from the vsfwd process on the ESXi host to the NSX Manager. Ensure that the port 5671 is open for communication in any of the external firewall on the environment. Also, there should be at least two connections on port 5671. There can be more connections on port 5671 as there are NSX Edge virtual machines deployed on the ESXi host which also establish connections to the RMQ broker.

For example:

```
# esxcli network ip connection list |grep 5671

tcp         0       0  192.168.110.51:30133            192.168.110.15:5671   ESTABLISHED
10949155  newreno  vsfwd
tcp         0       0  192.168.110.51:39156            192.168.110.15:5671   ESTABLISHED
10949155  newreno  vsfwd
```

5 Verify that VSFWD is configured.

This command should display the NSX Manager IP address.

```
# esxcfg-advcfg -g /UserVars/RmqIpAddress
```

6 If you are using a host-profile for this ESXi host, verify that the RabbitMQ configuration is not set in the host profile.

  See:

  - https://kb.vmware.com/kb/2092871
  - https://kb.vmware.com/kb/2125901

7 Verify if the RabbitMQ credentials of the ESXi host are out of sync with the NSX Manager. Download the NSX Manager Tech Support Logs. After gathering all the NSX Manager Tech Support logs, search all the logs for entries similar to:

  Replace host-420 with the mo-id of the suspect host.

```
PLAIN login refused: user 'uw-host-420' - invalid credentials.
```

8 If such entries are found on the logs for the suspected ESXi host, resynchronize the message bus.

  To resynchronize the message bus, use REST API. To better understand the issue, collect the logs immediately after the Message Bus is resynchronized.

```
HTTP Method : POST
Headers ,
Authorization : base64encoded value of username password
Accept : application/xml
Content-Type : application/xml
Request:

POST https://NSX_Manager_IP/api/2.0/nwfabric/configure?action=synchronize

Request Body:

<nwFabricFeatureConfig>
<featureId>com.vmware.vshield.vsm.messagingInfra</featureId>
<resourceConfig>
<resourceId>{HOST/CLUSTER MOID}</resourceId>
</resourceConfig>
</nwFabricFeatureConfig>
```

9 Use the `export host-tech-support <host-id> scp <uid@ip:/path>` command to gather host-specific firewall logs.

  For example:

```
nsxmgr# export host-tech-support host-28 scp Administrator@192.168.110.10
Generating logs for Host: host-28...
```

10 Use the `show dfw host host-id summarize-dvfilter` command to verify that the firewall rules are deployed on a host and are applied to virtual machines.

In the output, `module: vsip` shows that the DFW module is loaded and running. The `name` shows the firewall that is running on each vNic.

You can get the host IDs by running the `show dfw cluster all` command to get the cluster domain IDs, followed by the `show dfw cluster domain-id` to get the host IDs.

For example:

```
# show dfw host host-28 summarize-dvfilter

Fastpaths:
agent: dvfilter-faulter, refCount: 1, rev: 0x1010000, apiRev: 0x1010000, module: dvfilter
agent: ESXi-Firewall, refCount: 5, rev: 0x1010000, apiRev: 0x1010000, module: esxfw
agent: dvfilter-generic-vmware, refCount: 1, rev: 0x1010000, apiRev: 0x1010000, module: dvfilter-
generic-fastpath
agent: dvfilter-generic-vmware-swsec, refCount: 4, rev: 0x1010000, apiRev: 0x1010000, module:
dvfilter-switch-security
agent: bridgelearningfilter, refCount: 1, rev: 0x1010000, apiRev: 0x1010000, module: vdrb
agent: dvfg-igmp, refCount: 1, rev: 0x1010000, apiRev: 0x1010000, module: dvfg-igmp
agent: vmware-sfw, refCount: 4, rev: 0x1010000, apiRev: 0x1010000, module: vsip

Slowpaths:

Filters:
world 342296 vmm0:2-vm_RHEL63_srv_64-shared-846-3f435476-8f54-4e5a-8d01-59654a4e9979 vcUuid:'3f
43 54 76 8f 54 4e 5a-8d 01 59 65 4a 4e 99 79'
 port 50331660 2-vm_RHEL63_srv_64-shared-846-3f435476-8f54-4e5a-8d01-59654a4e9979.eth1
  vNic slot 2
   name: nic-342296-eth1-vmware-sfw.2
   agentName: vmware-sfw
   state: IOChain Attached
   vmState: Detached
   failurePolicy: failClosed
   slowPathID: none
   filter source: Dynamic Filter Creation
  vNic slot 1
   name: nic-342296-eth1-dvfilter-generic-vmware-swsec.1
   agentName: dvfilter-generic-vmware-swsec
   state: IOChain Attached
   vmState: Detached
   failurePolicy: failClosed
   slowPathID: none
   filter source: Alternate Opaque Channel
 port 50331661 (disconnected)
  vNic slot 2
   name: nic-342296-eth2-vmware-sfw.2 <======= DFW filter
   agentName: vmware-sfw
   state: IOChain Detached
   vmState: Detached
   failurePolicy: failClosed
   slowPathID: none
```

```
   filter source: Dynamic Filter Creation
 port 33554441 2-vm_RHEL63_srv_64-shared-846-3f435476-8f54-4e5a-8d01-59654a4e9979
  vNic slot 2
   name: nic-342296-eth0-vmware-sfw.2<========= DFW filter
   agentName: vmware-sfw
   state: IOChain Attached
   vmState: Detached
   failurePolicy: failClosed
   slowPathID: none
   filter source: Dynamic Filter Creation
```

11 Run the `show dfw host hostID filter filterID rules` command.

For example:

```
# show dfw host host-28 filter nic-79396-eth0-vmware-sfw.2 rules

ruleset domain-c33 {
  # Filter rules
  rule 1012 at 1 inout protocol any from addrset ip-securitygroup-10 to addrset ip-
securitygroup-10 drop with log;
  rule 1013 at 2 inout protocol any from addrset src1013 to addrset src1013 drop;
  rule 1011 at 3 inout protocol tcp from any to addrset dst1011 port 443 accept;
  rule 1011 at 4 inout protocol icmp icmptype 8 from any to addrset dst1011 accept;
  rule 1010 at 5 inout protocol tcp from addrset ip-securitygroup-10 to addrset ip-
securitygroup-11 port 8443 accept;
  rule 1010 at 6 inout protocol icmp icmptype 8 from addrset ip-securitygroup-10 to addrset ip-
securitygroup-11 accept;
  rule 1009 at 7 inout protocol tcp from addrset ip-securitygroup-11 to addrset ip-
securitygroup-12 port 3306 accept;
  rule 1009 at 8 inout protocol icmp icmptype 8 from addrset ip-securitygroup-11 to addrset ip-
securitygroup-12 accept;
  rule 1003 at 9 inout protocol ipv6-icmp icmptype 136 from any to any accept;
  rule 1003 at 10 inout protocol ipv6-icmp icmptype 135 from any to any accept;
  rule 1002 at 11 inout protocol udp from any to any port 67 accept;
  rule 1002 at 12 inout protocol udp from any to any port 68 accept;
  rule 1001 at 13 inout protocol any from any to any accept;
}

ruleset domain-c33_L2 {
  # Filter rules
  rule 1004 at 1 inout ethertype any from any to any accept;
```

12 Run the `show dfw host hostID filter filterID addrsets`command.

For example:

```
# show dfw host host-28 filter nic-342296-eth2-vmware-sfw.2 addrsets

addrset dst1011 {
ip 172.16.10.10,
ip 172.16.10.11,
ip 172.16.10.12,
```

```
ip fe80::250:56ff:feae:3e3d,
ip fe80::250:56ff:feae:f86b,
}
addrset ip-securitygroup-10 {
ip 172.16.10.11,
ip 172.16.10.12,
ip fe80::250:56ff:feae:3e3d,
ip fe80::250:56ff:feae:f86b,
}
addrset ip-securitygroup-11 {
ip 172.16.20.11,
ip fe80::250:56ff:feae:23b9,
}
addrset ip-securitygroup-12 {
ip 172.16.30.11,
ip fe80::250:56ff:feae:d42b,
}
addrset src1013 {
ip 172.16.10.12,
ip 172.17.10.11,
ip fe80::250:56ff:feae:cf88,
ip fe80::250:56ff:feae:f86b,
}
```

13 If you have validated each of the above troubleshooting steps and cannot publish firewall rules to the host virtual machines, execute a host-level force synchronization via the NSX Manager UI or via the following REST API call.

```
URL : [https:]https://<nsx-mgr-ip>/api/4.0/firewall/forceSync/<host-id>
HTTP Method : POST
Headers ,
Authorization : base64encoded value of username password
Accept : application/xml
Content-Type : application/xml
```

Solution

Notes:

- Ensure that VMware Tools is running on the virtual machines if firewall rules do not use IP addresses. For more information, see https://kb.vmware.com/kb/2084048.

  VMware NSX 6.2.0 introduced the option to discover the virtual machine IP address using DHCP snooping or ARP snooping. These new discovery mechanisms enable NSX to enforce IP address-based security rules on virtual machines that do not have VMware Tools installed. For more information, see the NSX 6.2.0 Release Notes.

DFW is activated as soon as the host preparation process is completed. If a virtual machine needs no DFW service at all, it can be added in the exclusion list functionality (by default, NSX Manager, NSX Controllers and Edge Services Gateways are automatically excluded from DFW function). There is a possibility that the vCenter Server access gets blocked after creating a Deny All rule in DFW. For more information, see https://kb.vmware.com/kb/2079620.

- When troubleshooting VMware NSX 6.x Distributed Firewall (DFW) with VMware Technical Support, these are required:
  - Output of the command `show dfw host hostID summarize-dvfilter` on each of the ESXi host on the cluster.
  - Distributed Firewall Configuration from the **Networking and Security > Firewall > General** tab and click **Export Configuration**. This exports the Distributed Firewall configuration to an XML format.
  - NSX Manager logs. For more information, see https://kb.vmware.com/kb/2074678.
  - vCenter Server logs. For more information, see https://kb.vmware.com/kb/1011641.

Identity Firewall

Problem

Publishing or updating Identity firewall rules fail.

Cause

Identity Firewall (IDFW) allows user-based distributed firewall rules (DFW).

User-based distributed firewall rules are determined by membership in an Active Directory (AD) group membership. IDFW monitors where Active Directory users are logged into and maps the login to an IP Address, which is used by DFW to apply firewall rules. IDFW requires either Guest Introspection framework, and/or Active Directory event log scraping.

Solution

1. Make sure that the Active Directory server full/delta sync is working on the NSX Manager.
   a. In the vSphere Web Client, log in to the vCenter linked to the NSX Manager.
   b. Navigate to **Home > Networking & Security> NSX Managers**, and then select your NSX Manager from the list.
   c. Choose the **Manage** tab, then the **Domains** tab. Select your domain from the list. Verify that the **Last Synchronization Status** column displays SUCCESS and the **Last Synchronization Time** is current.

2 If your firewall environment uses the event log scraping method of login detection, follow these steps to verify that you have configured an event log server for your domain:

   a In the vSphere Web Client, log in to the vCenter linked to the NSX Manager.

   b Navigate to **Home > Networking & Security> NSX Managers**, and then select your NSX Manager from the list.

   c Choose the **Manage** tab and then the **Domains** tab. Select your domain from the list. Here you can view and edit the detailed domain configuration.

   d Select **Event Log Servers** from the domain details and verify that your Event Log Server is added.

   e Select your Event Log Server, and verify that the **Last Sync Status** column displays SUCCESS and the **Last Sync Time** is current.

3 If your firewall environment uses Guest Introspection, the framework must be deployed to the compute clusters where your IDFW protected VMs will reside. The Service Health Status on the UI should be green. Guest Introspection diagnostic information is found in the following the Knowledge Base articles: Troubleshooting vShield Endpoint / NSX Guest Introspection https://kb.vmware.com/kb/2094261 and Collecting logs in VMware NSX for vSphere 6.x Guest Introspection Universal Service Virtual Machine https://kb.vmware.com/kb/2144624.

4 After verifying the correct configuration of your logon detection method, ensure that the NSX Manager is receiving logon events;

   a Log in an Active Directory user.

   b Run the following command to query for login events. Verify your user is returned in the results. `GET https://<nsxmgr-ip>/1.0/identity/userIpMapping`.

```
Example output:
<UserIpMappings>
    <UserIpMapping>
        <ip>50.1.111.192</ip>
        <userName>user1_group20</userName>
        <displayName>user1_group20</displayName>
        <domainName>cd.ad1.db.com</domainName>
        <startTime class="sql-timestamp">2017-05-11 22:30:51.0</startTime>
        <startType>EVENTLOG</startType>
        <lastSeenTime class="sql-timestamp">2017-05-11 22:30:52.0</lastSeenTime>
        <lastSeenType>EVENTLOG</lastSeenType>
    </UserIpMapping>
</UserIpMappings>
```

5 Verify that your security group is used in a firewall rule, or has an assigned security policy. Security group processing in IDFW will not take place unless one of these conditions is true.

6 After verifying that IDFW is detecting logons correctly, verify that the ESXi host where your desktop VM resides is receiving the correct configuration. These steps will use the NSX Manager central CLI. To check the desktop VM IP address populated in the **ip-securitygroup** list:

   a See CLI Commands for DFW to retrieve the filter name applied on the desktop VM.

   b Run the `show dfw host hostID filter filterID rules` command to view the locate DFW rules items.

   c Run the `show dfw host hostID filter filterID addrsets` command to view the IP address populated in the `ip-securitygroup` list. Verify that your IP is displayed in the list.

Solution

Note: When troubleshooting Identity IDFW with VMware Technical Support, this data is helpful:

- If using event log scraping Active Directory scale data:
  - # of Domains for a single NSX Manager

    # of Forests

    # of Users / Forest

    # of Users / Domain

    # of Active Directory groups per Domain

    # of Users / Active Directory Group

    # of Active Directory / User

    # of Domain Controllers

    # of Active Directory Log Servers
- User login scale data:
  - Average # of users per min
- Deployment Details using IDFW with VDI:
  - # of VDI desktops / VC

    # of hosts / VC

    # VDI desktops / host
- If using Guest Introspection:
  - Version of VMTools (Guest Introspection Drivers)

    Version of Windows Guest OS

Troubleshooting Load Balancing 6

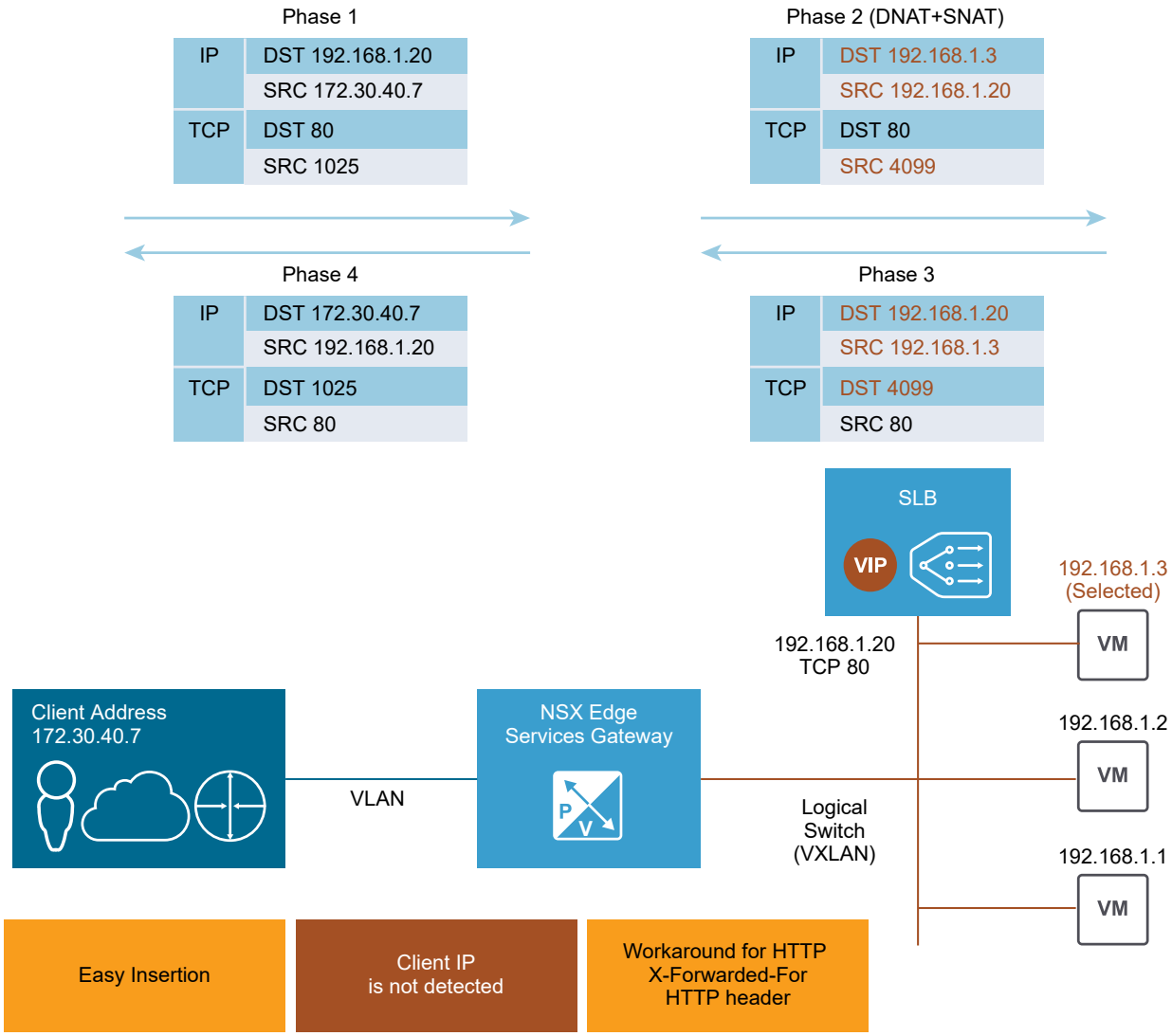
The NSX Edge load balancer enables network traffic to follow multiple paths to a specific destination. It distributes incoming service requests evenly among multiple servers in such a way that the load distribution is transparent to users. There are two types of load balancing services to configure in NSX, a one-armed mode, also known as a proxy mode, or the Inline mode, otherwise known as the transparent mode. For more information, refer to *NSX Administration Guide*.

Prior to beginning troubleshooting and configuration verification, get an accurate description of the error, create a topology map in relation to the client, virtual server and backend server, and understand the application requirements. For example a client cannot connect, is different than random session errors after connection. While troubleshooting load balancer, always starts by verifying connectivity error.

This chapter includes the following topics:

Configure a One-Armed Load Balancer

The Edge Services Gateway (ESG) can be thought of as a proxy for incoming client traffic.


Phase 1
IP
DST 192.168.1.20
SRC 172.30.40.7
TCP
DST 80
SRC 1025
Phase 2 (DNAT+SNAT)
IP
DST 192.168.1.3
SRC 192.168.1.20
TCP
DST 80
SRC 4099
Phase 4
IP
DST 172.30.40.7
SRC 192.168.1.20
TCP
DST 1025
SRC 80
Phase 3
IP
DST 192.168.1.20
SRC 192.168.1.3
TCP
DST 4099
SRC 80
SLB
VIP
192.168.1.3
(Selected)
VM
192.168.1.20
TCP 80
192.168.1.2
VM
192.168.1.1
VM
Client Address
172.30.40.7
VLAN
NSX Edge
Services Gateway
P
V
Logical
Switch
(VXLAN)
Easy Insertion
Client IP
is not detected
Workaround for HTTP
X-Forwarded-For
HTTP header

In proxy mode, the load balancer uses its own IP address as the source address to send requests to a backend server. The backend server views all traffic as being sent from the load balancer and responds to the load balancer directly. This mode is also called SNAT mode or non-transparent mode. For more information, refer to *NSX Administration Guide*.

A typical NSX one-armed load balancer is deployed on the same subnet with its backend servers, apart from the logical router. The NSX load balancer virtual server listens on a virtual IP for incoming requests from client and dispatches the requests to backend servers. For the return traffic, reverse NAT is required to change the source IP address from the backend server to a virtual IP (VIP) address and then send the virtual IP address to the client. Without this operation, the connection to the client would break.

After the ESG receives the traffic, it performs two operations: Destination Network Address Translation (DNAT) to change the VIP address to the IP address of one of the load balanced machines, and Source Network Address Translation (SNAT) to exchange the client IP address with the ESG IP address.

Then the ESG server sends the traffic to the load balanced server and the load balanced server sends the response back to the ESG then back to the client. This option is much easier to configure than the Inline mode, but has two potentials caveats. The first is that this mode requires a dedicated ESG server, and the second is that the load balancer servers are not aware of the original client IP address. One workaround for HTTP/HTTPS applications is to enable Insert X-Forwarded-For in the HTTP application profile so that the client IP address will be carried in the X-Forwarded-For HTTP header in the request sent to the backend server.

If client IP address visibility is required on the backend server for applications other than HTTP/HTTPS, you can configure the IP pool to be transparent. In case clients are not on the same subnet as the backend server, inline mode is recommended. Otherwise, you must use the load balancer IP address as the default gateway of the backend server.

Note Usually, there are three methods to guarantee connection integrity:

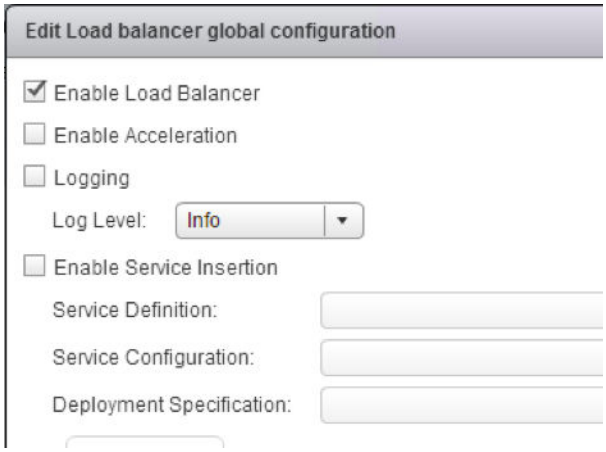
- Inline/transparent mode
- SNAT/proxy/non-transparent mode (discussed above)
- Direct server return (DSR) - Currently, this is unsupported

In DSR mode, the backend server responds directly to the client. Currently, NSX load balancer does not support DSR.

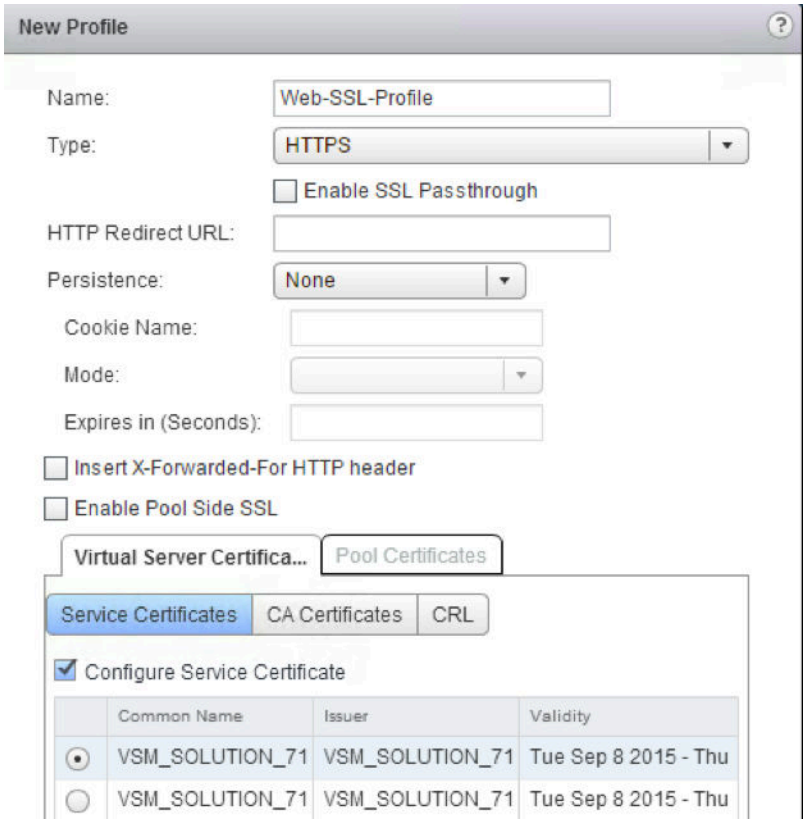
Procedure

1 As an example, lets configure a one-armed virtual server with SSL offload. Create a certificate by double-clicking the Edge and then selecting **Manage > Settings > Certificate**.

2 Enable the load balancer service by selecting **Manage > Load Balancer > Global Configuration > Edit**.



3 Create an HTTPS application profile by selecting **Manage > Load Balancer > Application Profiles**.

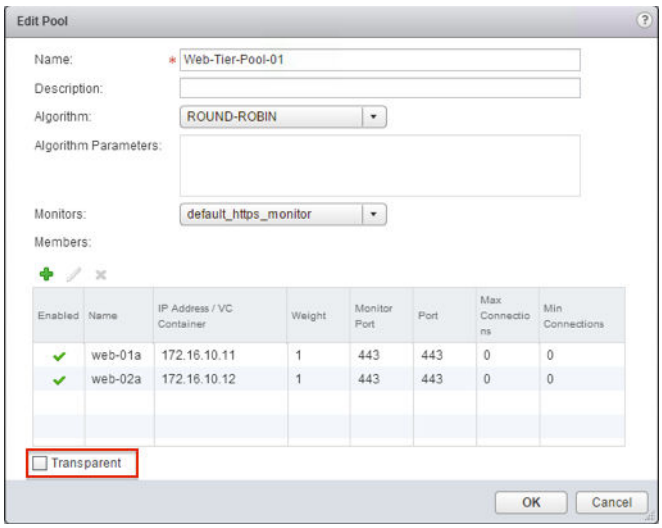


Note The screenshot above uses self-signed certificates for documentation-purposes only.

4 Optionally, click **Manage > Load Balancer > Service Monitoring** and edit the default service monitoring to change it from basic HTTP/HTTPS to specific URL/URIs, as required.

5 Create server pools by selecting **Manage > Load Balancer > Pools**.

To use SNAT mode, leave the **Transparent** check box unchecked in the pool configuration.



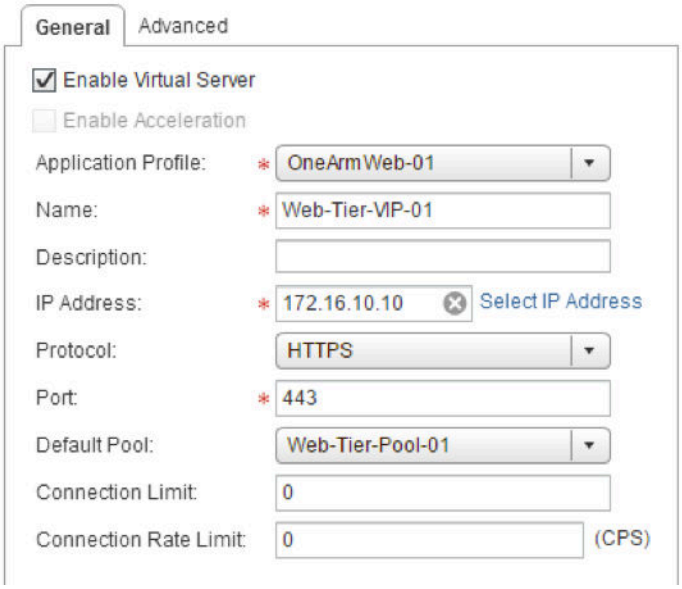
Ensure that the VMs are listed and enabled.

6 Optionally, click **Manage > Load Balancer > Pools > Show Pool Statistics** to check the status.

Make sure that the member status is UP.

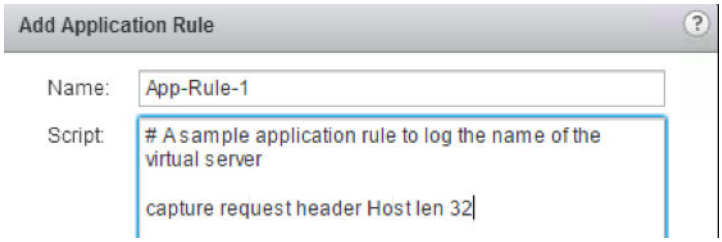
7 Create a virtual server by selecting **Manage > Load Balancer > Virtual Servers**.

If you would like to use the L4 load balancer for UDP or higher-performance TCP, check **Enable Acceleration**. If you check **Enable Acceleration**, make sure that the firewall status is **Enabled** on the load balancer NSX Edge, because a firewall is required for L4 SNAT.



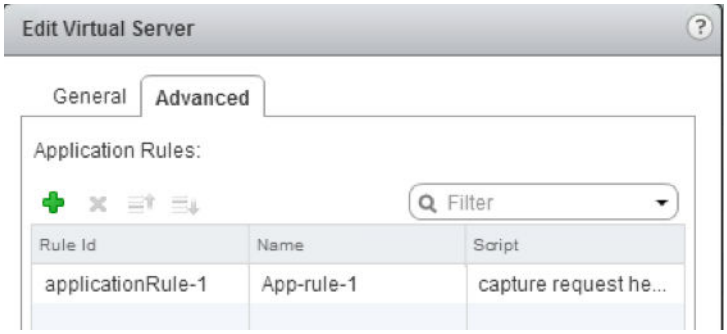
Ensure that the IP address is tied to the server pool.

8 Optionally, if using an application rule, check the configuration in **Manage > Load Balancer > Application Rules**.

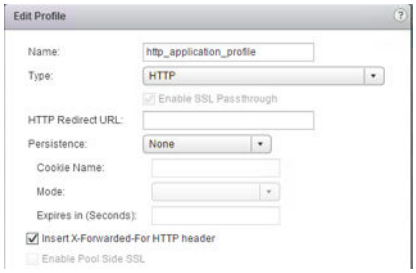


9 If using an application rule, ensure that the application rule is associated with the virtual server in **Manage > Load Balancer > Virtual Servers > Advanced**.

For supported examples, see: https://communities.vmware.com/docs/DOC-31772.

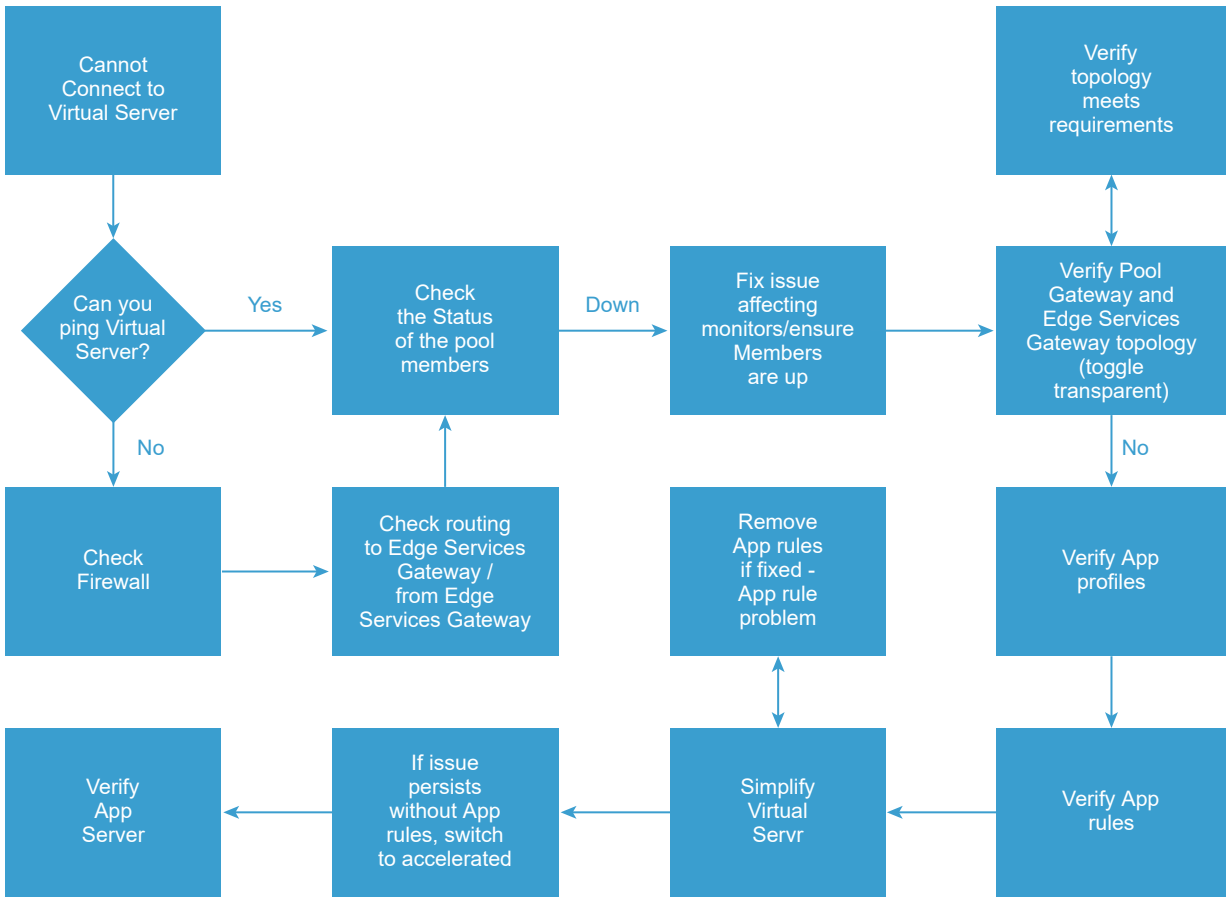


In non-transparent mode, the backend server cannot see the client IP, but can see the load balancer internal IP address. As a workaround for HTTP/HTTPS traffic, check **Insert X-Forwarded-For HTTP header**. With this option checked, the Edge load balancer adds the header "X-Forwarded-For" with the value of the client source IP address.



Troubleshooting Flowchart for Load Balancer

The following flowchart is as an overview on how to troubleshoot load balancer issues.



Load Balancer Configuration Verification and Troubleshooting Using the UI

You can verify the load balancer configuration through the vSphere Web Client. You can use the UI to do some load balancer troubleshooting.

After understanding what should be functioning and defining a problem, verify the configuration through the UI as follows.

Prerequisites

Note down the following details:

- The IP, protocol, and port of the virtual server.
- The IP, and port of the backend application servers.
- The topology that was intended - inline or one-armed. For details, refer to the Logical Load Balancer topic in *NSX Administration Guide*.
- Verify the trace route and use other network connectivity tools to see that the packets are going to the correct location (edge services gateway).
- Verify any upstream firewalls are allowing the traffic correctly.

- Define the problem that you are facing. For example, DNS records for the virtual server are correct, but you are not getting back any content, or incorrect content, and so on.

Problem

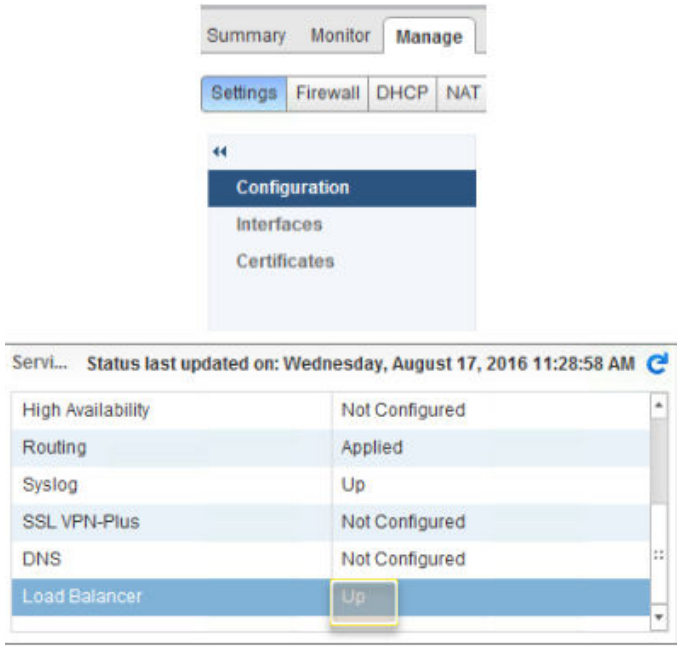
Load balancer is not working as expected.

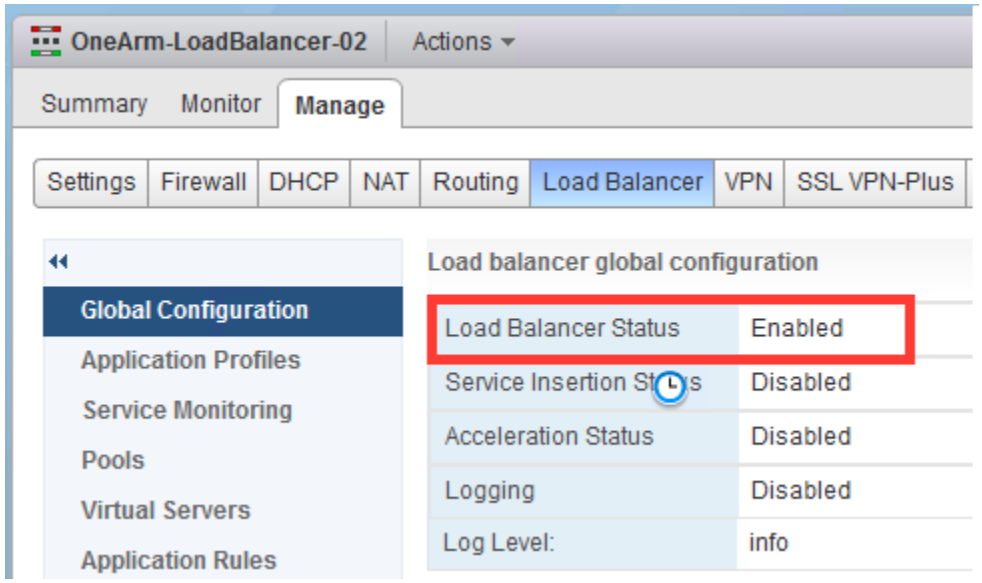
Solution

1. Verify the following application requirements - Protocols required to be supported on the load balancer (TCP, UDP, HTTP, HTTPs), ports, persistence requirements, and pool members.
   - Is the load balancer and firewall enabled and does the edge services gateway have proper routes?
   - What IP address, port and protocol should the virtual server be listening to?
   - Is SSL offload being used? Do you need to use SSL when communicating with the backend servers?
   - Are you using application rules?
   - What is the topology? The NSX load balancer needs to parse all the traffic from the client and the server.
   - Is the NSX load balancer inline or is the client source address translated to ensure return traffic travels back to the load balancer?

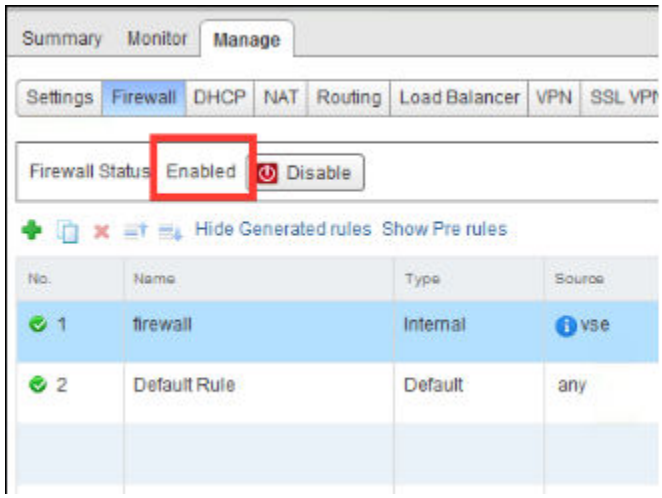
2 Navigate to the NSX Edge, and verify the configurations that are required to enable load balancing and allow traffic to flow as follows:

   a Verify the load balancer is listed as **Up**.





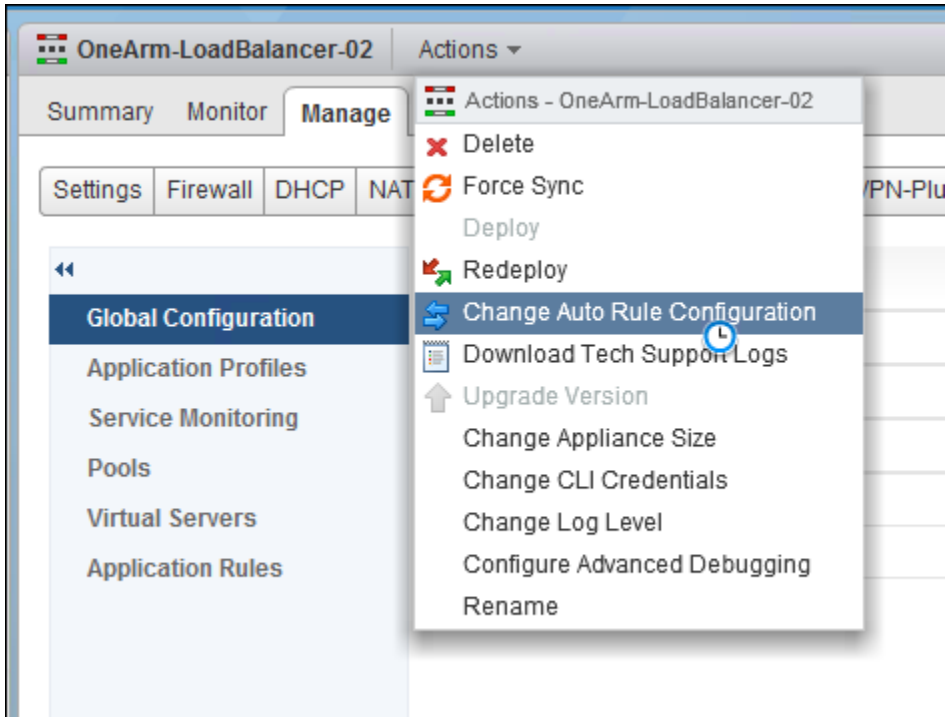
b Verify the firewall is **Enabled**. The firewall MUST be enabled for accelerated virtual servers. Non Accelerated TCP and L7 HTTP/HTTPS VIPs must have a policy that allows traffic. Note that the firewall filters will not impact accelerated virtual servers.



c Verify that the NAT rules are created for the virtual server. On the **NAT** tab, click the **Hide internal rules** or **Unhide internal rules** link to verify.

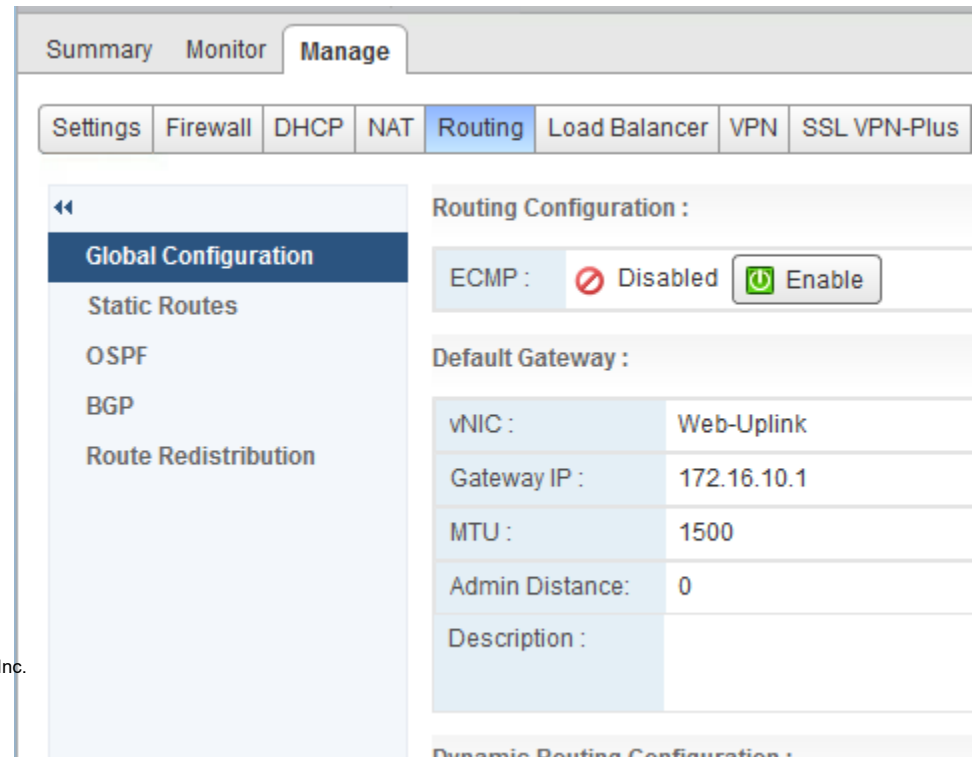
Note If you have load balancing enabled and services configured, but have not configured any NAT rules, it means that the auto rule configuration was not enabled.

d You can change the auto rule configurations. For details, refer to Change Auto Rule Configuration topic in the *NSX Administration Guide*. When an NSX edge services gateway is deployed, you have the option to configure auto rule configuration. If this option was not selected while deploying the edge services gateway, you must enable it for the load balancer to function correctly. Check the pool member status through the UI.

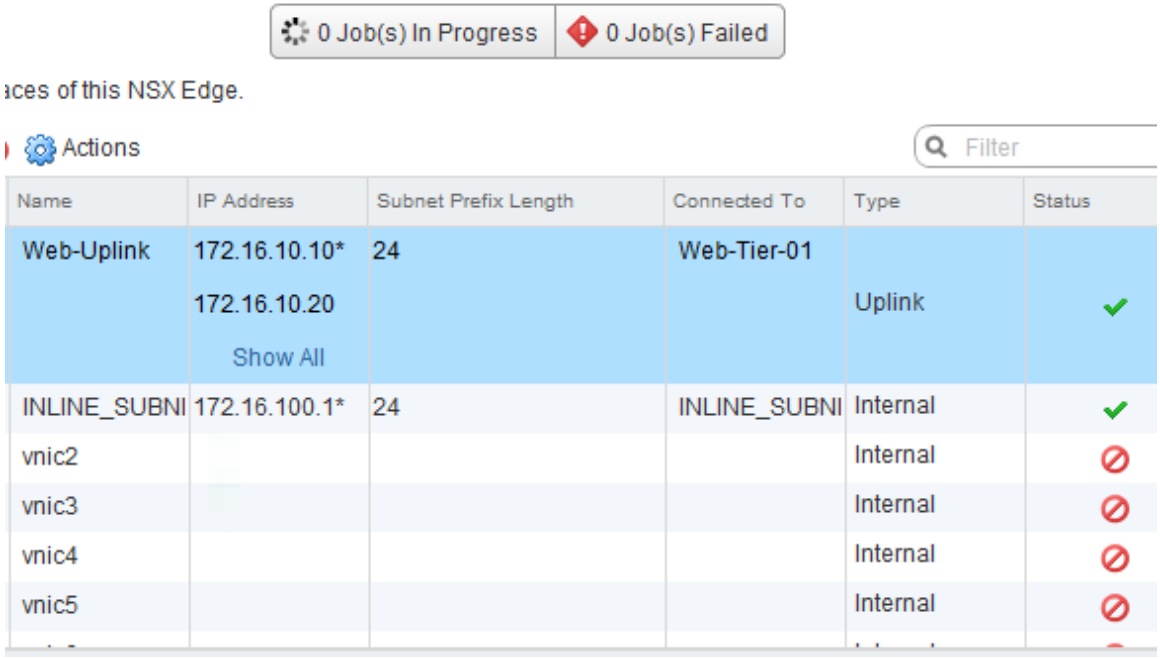


e Verify routing, and verify that the edge services gateway has a default route or a static route to your client systems and the backend servers. If there is no route to the servers, health check will not pass. If you are using a dynamic routing protocol you may have to use the CLI. For more information, refer to NSX Routing CLI.

a Verify default route.



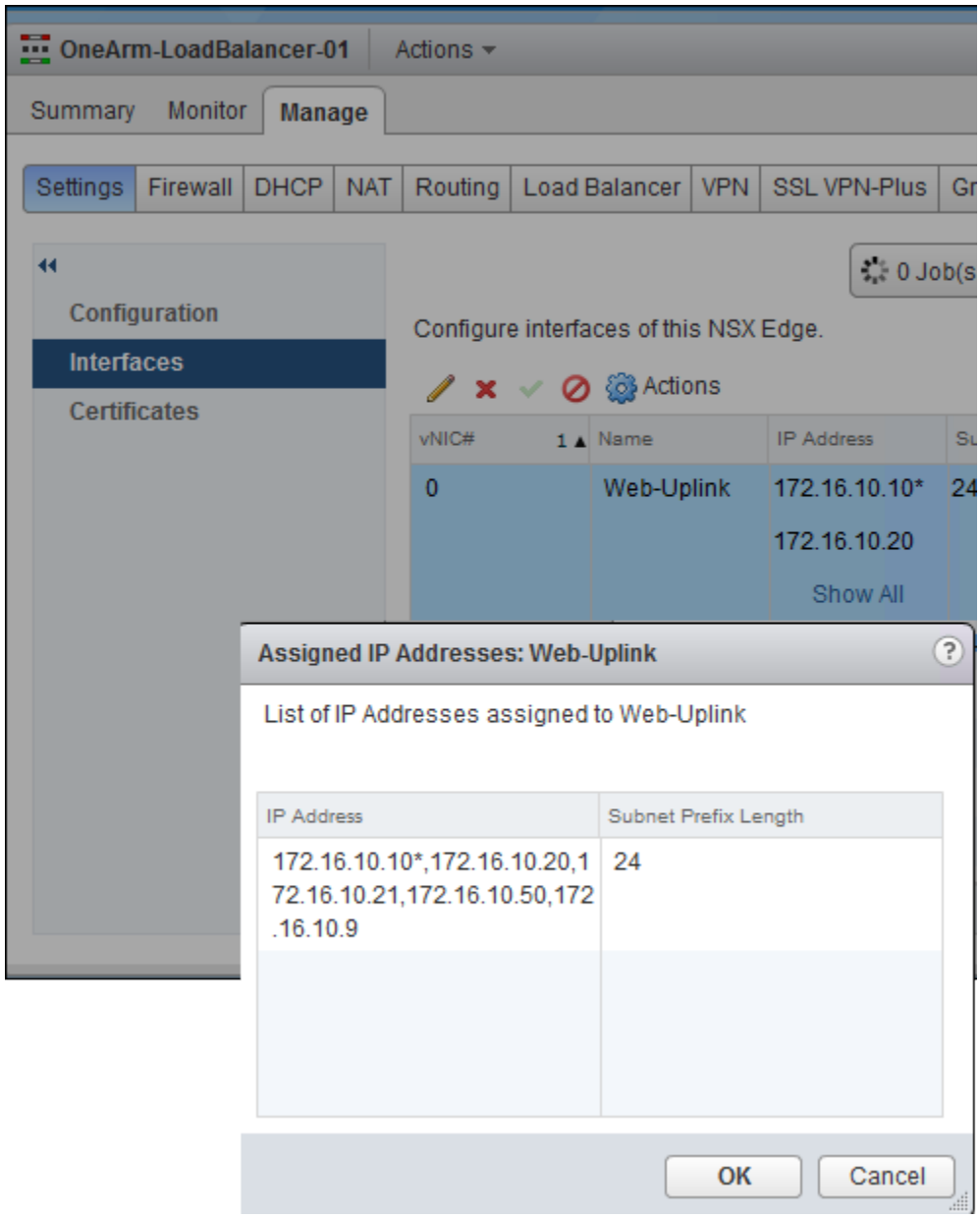
interface in the subnet. Many times the application servers are connected to these servers.



c Verify static routes from the **Routing** tab > **Static Routes**.

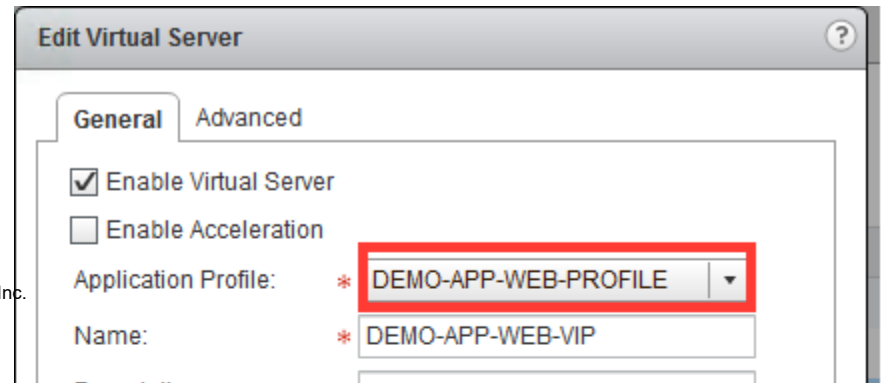
3 Verify the IP address, port and protocol of the virtual server.

a Double-click an NSX Edge and navigate to **Manage > Settings**> **Interfaces**. Verify that IP address for the virtual server is added to an interface.

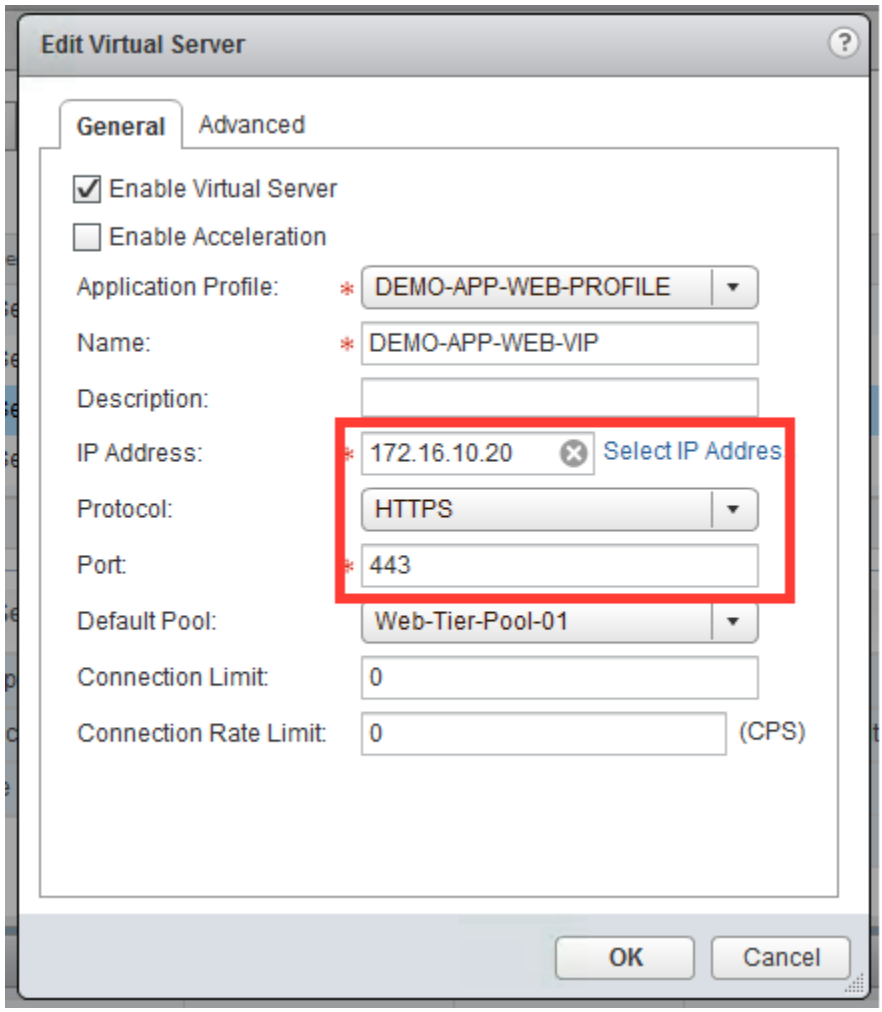


b Verify the virtual server has the proper IP address, port(s) and protocols configured to support the application.

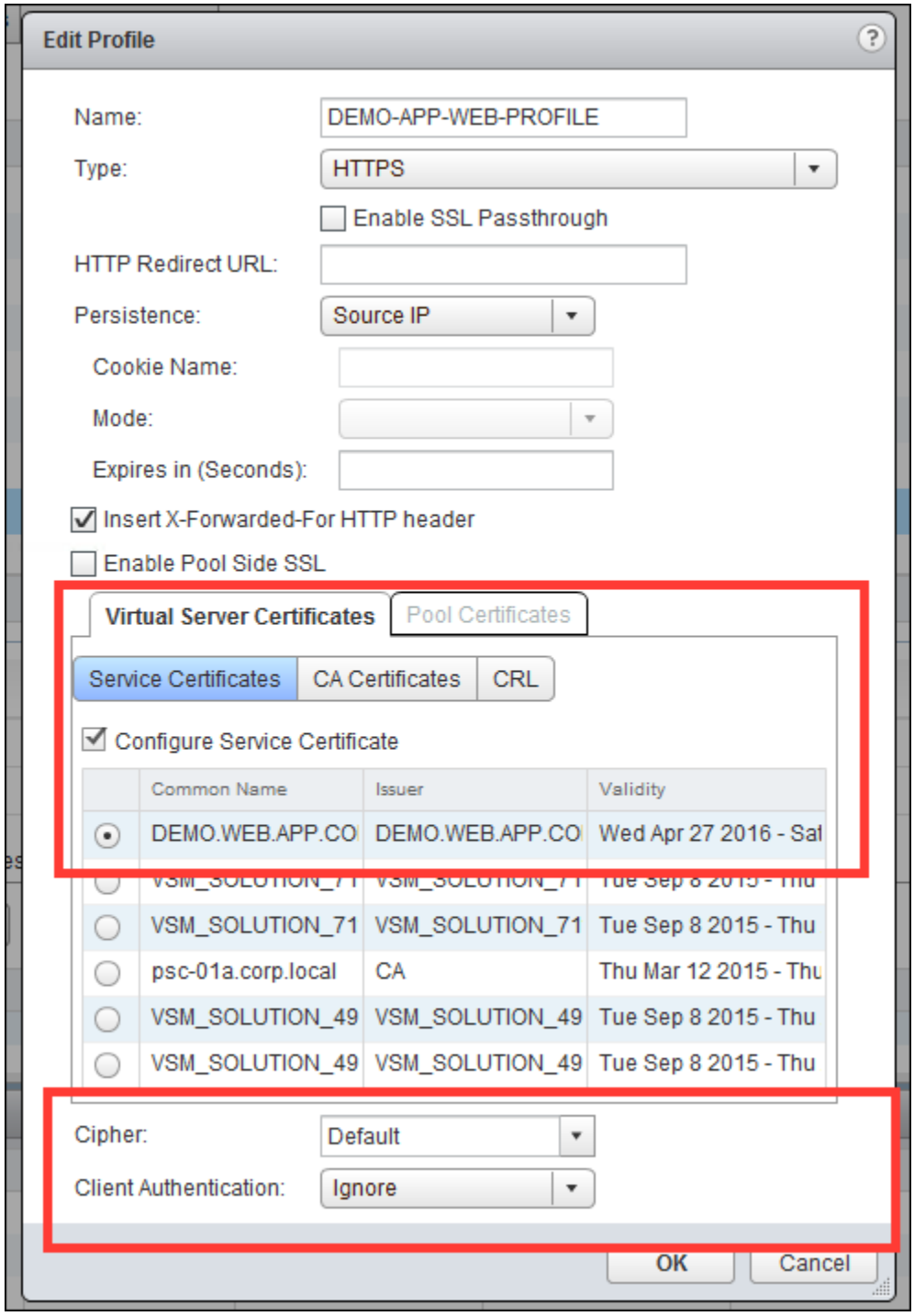
a Verify the application profile used by the virtual server.



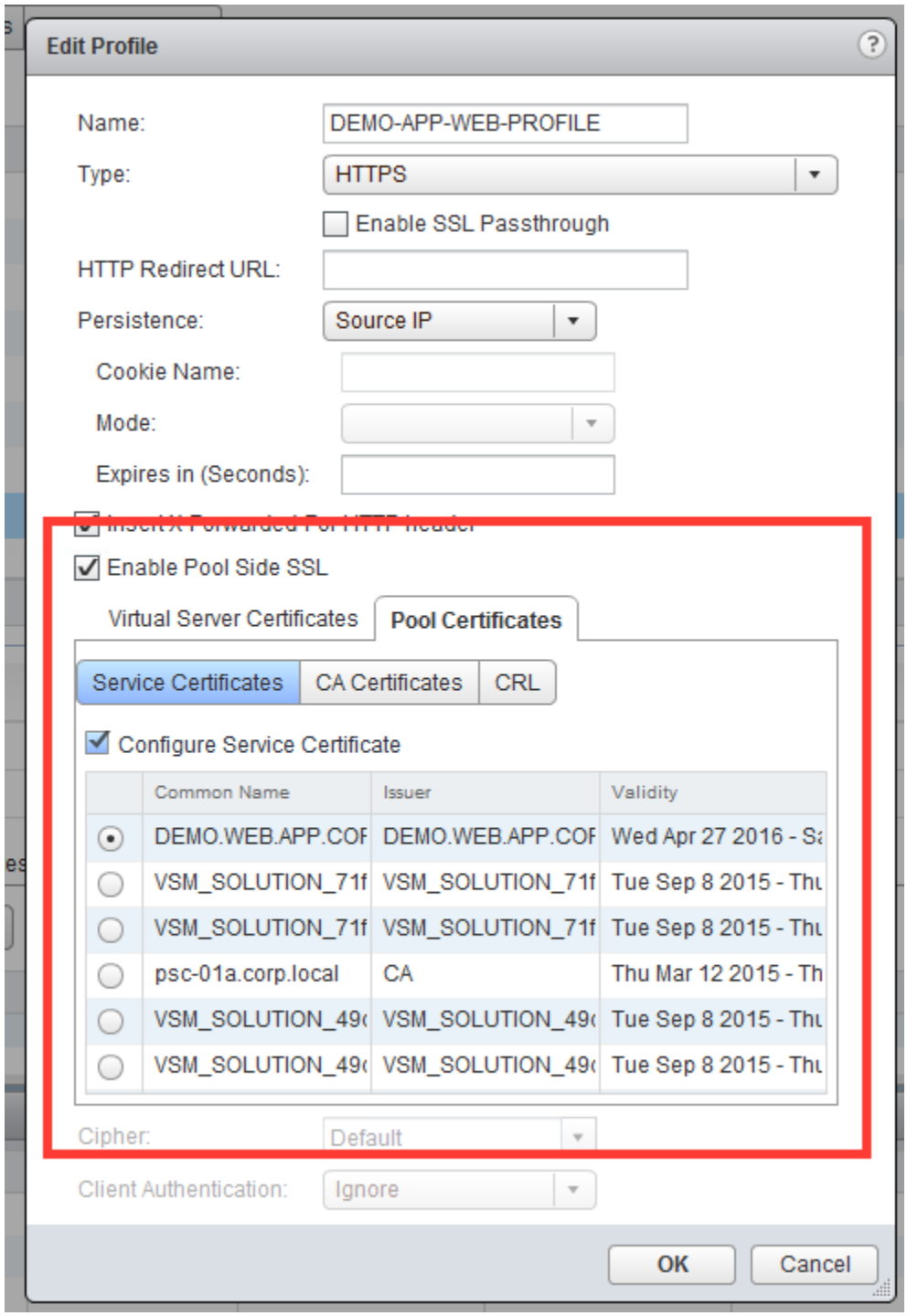
on the virtual server.



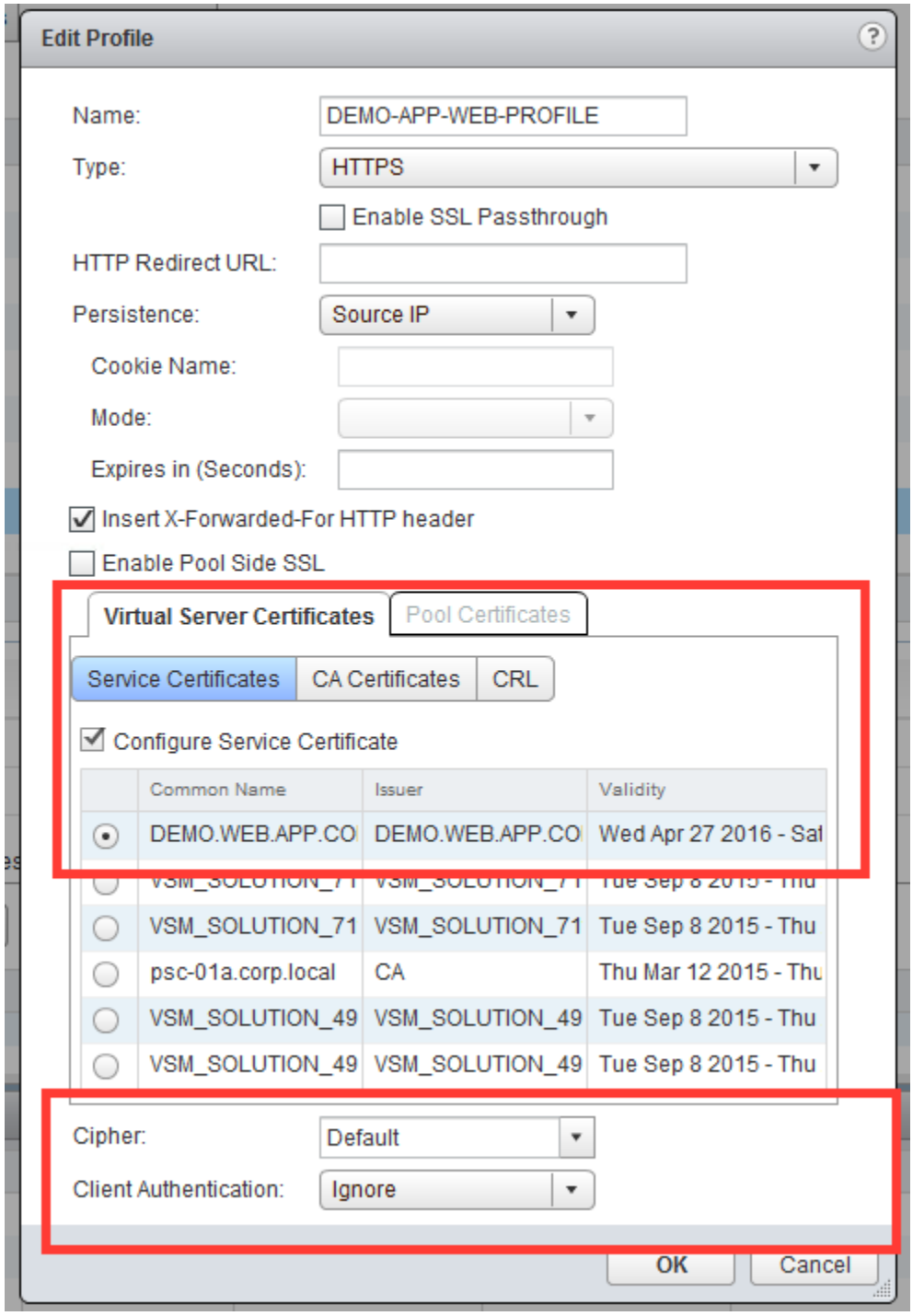
c Verify the application profile meets the persistent method supported, type (protocol), and SSL (if necessary). If using SSL, ensure you are using a certificate with the correct name and expiration date.



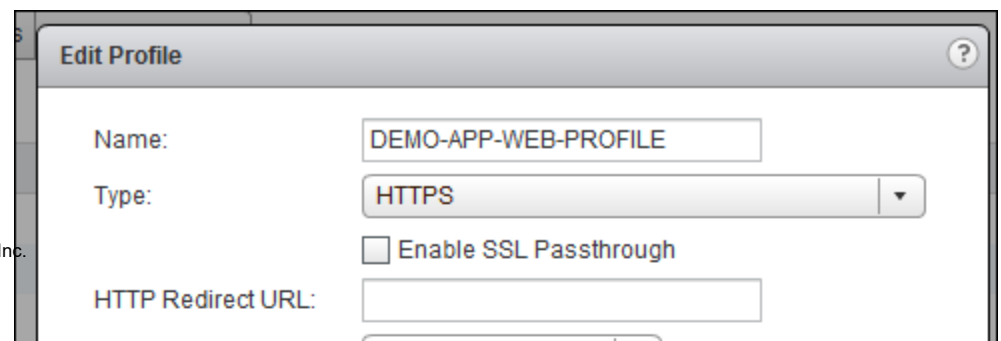
d Verify if the correct certificate is used for the clients to connect.



e Verify if you require a client certificate, but the clients are not configured. Also, verify if you have selected a narrow cipher list that is too narrow (for example, are clients using older browsers).

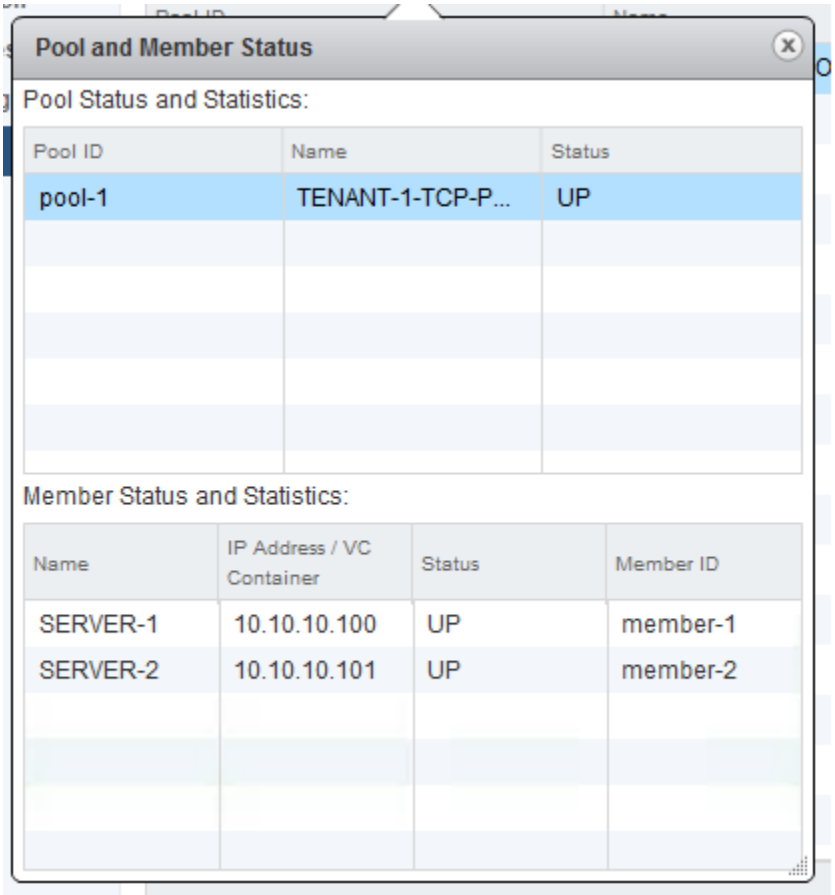


f Verify if you need SSL to the backend servers.

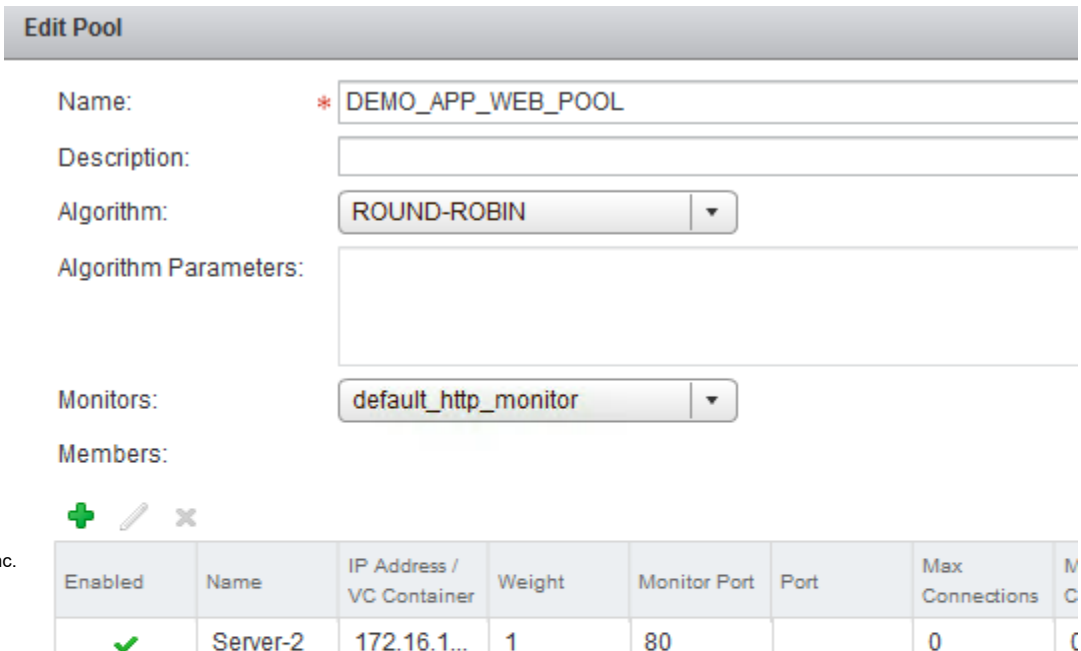


4 Check the pool status and configuration as follows:

a Verify the pool status, at least one member must be up to serve traffic, but one member may not be enough to serve all the traffic. If zero, or a limited member of pool members are up, try to rectify the problem as described in next steps.

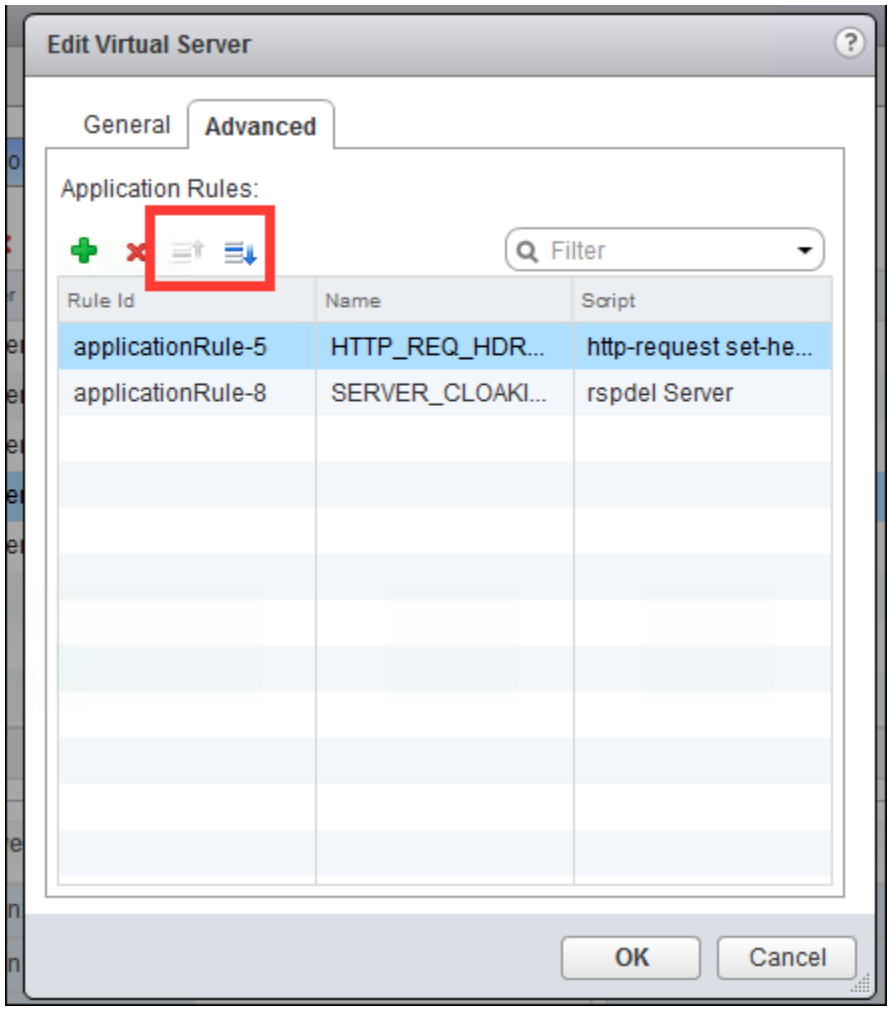


b Verify if the topology is correct. SNAT client traffic is controlled in the pool configuration. If the edge services gateway hosting the load balancer function is not inline to see all the traffic, then it will fail. To preserve the IP of the client source, select the **Transparent** mode. For information, refer to the*NSX Administration Guide*.



5 If you are using application rules, verify the rules. Remove the rules if necessary to see if traffic flows.

a Reorder the rules to see if the order of the rules is causing the logic to interrupt the traffic flow. For information on how to add an application rule and view application rule examples, see the Add an Application Rule topic in *NSX Administration Guide*.



What to do next

If you could not find the problem, you may need to use the CLI (Command Line Interface) to find out what is happening. For more information, refer to Load Balancer Troubleshooting Using the CLI.

Load Balancer Troubleshooting Using the CLI

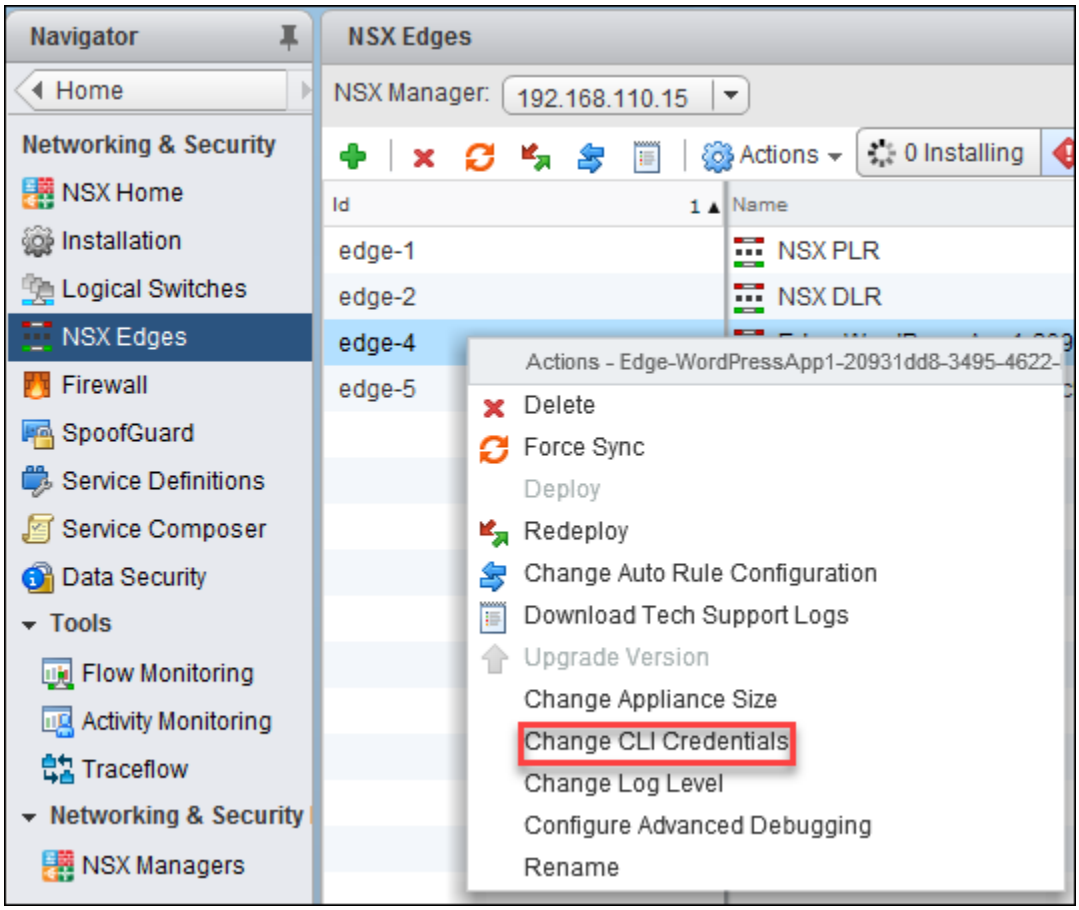
The NSX CLI can be used to get detailed tail logs, take packet captures, and look at the metrics for troubleshooting the load balancer.

Problem

Load balancing is not working as expected.

Solution

1 Enable or verify you can SSH to the virtual appliance. The edge services gateway is a virtual appliance that has the option to enable SSH while deploying. If you need to enable SSH, select the required appliance, and in the **Actions** menu, click **Change CLI Credentials**.



2 The edge services gateway has multiple show commands to look at the run time state, and the configuration state. Use the commands to show configuration and statistics information.

```
nsxedge> show configuration loadbalancer
nsxedge> show configuration loadbalancer virtual [virtual-server-name]
nsxedge> show configuration loadbalancer pool [pool-name]
nsxedge> show configuration loadbalancer monitor [monitor-name]
nsxedge> show configuration loadbalancer profile [profile-name]
nsxedge> show configuration loadbalancer rule [rule-name]
```

3 For load balancing and NAT to function correctly the firewall should be enabled. Use the `#show firewall` command. If you do not see any meaningful output using the command, refer to the Load Balancer Configuration Verification and Troubleshooting Using the UI section.

```
NSX-edge-8-0> show firewall
Chain PREROUTING (policy ACCEPT 21947 packets, 7809K bytes)
rid    pkts bytes target     prot opt in     out     source               destination

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
rid    pkts bytes target     prot opt in     out     source               destination
)       348 67915 ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
)       134  5360 DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0           state INVALID
)     21482 7736K block_in   all  --  *      *       0.0.0.0/0            0.0.0.0/0
)     20545 7671K ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           state RELATED
)       937 65139 usr_rules  all  --  *      *       0.0.0.0/0            0.0.0.0/0
)         0     0 DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
rid    pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 20673 packets, 1248K bytes)
rid    pkts bytes target     prot opt in     out     source               destination

Chain POSTROUTING (policy ACCEPT 0 packets, 0 bytes)
rid    pkts bytes target     prot opt in     out     source               destination
)       348 67915 ACCEPT     all  --  *      lo      0.0.0.0/0            0.0.0.0/0
)        34  1360 DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0           state INVALID
)     20295 1179K block_out  all  --  *      *       0.0.0.0/0            0.0.0.0/0
)         0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           PHYSDEV match
)         0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           PHYSDEV match
)         0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           PHYSDEV match
)         0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           PHYSDEV match
)     14599  802K ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           state RELATED
)      5696  377K usr_rules  all  --  *      *       0.0.0.0/0            0.0.0.0/0
)         0     0 DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain block_in (1 references)
rid    pkts bytes target     prot opt in     out     source               destination

Chain block_out (1 references)
rid    pkts bytes target     prot opt in     out     source               destination

Chain usr_rules (2 references)
rid    pkts bytes target     prot opt in     out     source               destination
l33137  4861  333K ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           match-set 0_
l33138     0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           match-set 1_
l33139   936 65099 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           match-set 2_
l33141   835 43459 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0           match-set 3_
l33131     1    40 LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0           LOG flags 0
l33131     1    40 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0
```

4 Load balancer requires NAT to function correctly. Use the `show nat` command. If you do not see any meaningful output using the command, refer to the Load Balancer Configuration Verification and Troubleshooting Using the UI section.

```
NSX-edge-8-0> show nat
Chain PREROUTING (policy ACCEPT 568 packets, 40044 bytes)
rid    pkts bytes target     prot opt in     out     source               destination
0       568 40044 int_dnat   all  --  *      *       0.0.0.0/0            0.0.0.0/0
0       568 40044 usr_dnat   all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain INPUT (policy ACCEPT 568 packets, 40044 bytes)
rid    pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 896 packets, 46706 bytes)
rid    pkts bytes target     prot opt in     out     source               destination
0       896 46706 int_dnat   all  --  *      *       0.0.0.0/0            0.0.0.0/0
0       896 46706 usr_dnat   all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain POSTROUTING (policy ACCEPT 896 packets, 46706 bytes)
rid    pkts bytes target     prot opt in     out     source               destination
0       896 46706 int_snat   all  --  *      *       0.0.0.0/0            0.0.0.0/0
0       896 46706 usr_snat   all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain int_dnat (2 references)
rid    pkts bytes target     prot opt in     out     source               destination

Chain int_snat (1 references)
rid    pkts bytes target     prot opt in     out     source               destination
0          0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain usr_dnat (2 references)
rid    pkts bytes target     prot opt in     out     source               destination
0          0     0 DNAT       tcp  --  vNic_2 *       0.0.0.0/0            192.168.8.20
0          0     0 LOG        all  --  vNic_2 *       0.0.0.0/0            192.168.8.11
0          0     0 DNAT       all  --  vNic_2 *       0.0.0.0/0            192.168.8.11

Chain usr_snat (1 references)
rid    pkts bytes target     prot opt in     out     source               destination
0          0     0 LOG        all  --  *      vNic_2  10.10.10.101         0.0.0.0/0
0          0     0 SNAT       all  --  *      vNic_2  10.10.10.101         0.0.0.0/0
0          0     0 LOG        all  --  *      vNic_2  10.10.10.0/24         0.0.0.0/0
0          0     0 SNAT       all  --  *      vNic_2  10.10.10.0/24         0.0.0.0/0
NSX-edge-8-0>
```

5 In addition to the firewall being enabled and the load balancer having NAT rules, you should also make sure the load balancing process is enabled. Use the `show service loadbalancer` command to check the load balancer engine status (L4/L7).

```
nsxedge> show service loadbalancer
haIndex:              0
------------------------------------------------------------------------
Loadbalancer Services Status:

L7 Loadbalancer     : running
------------------------------------------------------------------------
L7 Loadbalancer Statistics:
STATUS     PID        MAX_MEM_MB MAX_SOCK   MAX_CONN   MAX_PIPE   CUR_CONN   CONN_RATE
CONN_RATE_LIMIT MAX_CONN_RATE
running    1580       0          2081       1024       0          0          0
```

```
0               0
------------------------------------------------------------------------
L4 Loadbalancer Statistics:
MAX_CONN   ACT_CONN   INACT_CONN TOTAL_CONN
0          0          0          0

Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
```

a Use the `show service loadbalancer session` command to view the load balancer session table. You will see sessions if there is traffic on the system.

```
nsxedge> show service loadbalancer session
------------------------------------------------------------------------
L7 Loadbalancer Statistics:
STATUS     PID        MAX_MEM_MB MAX_SOCK   MAX_CONN   MAX_PIPE   CUR_CONN   CONN_RATE
CONN_RATE_LIMIT MAX_CONN_RATE
running    1580       0          2081       1024       0          0          0
0               0

------------------L7 Loadbalancer Current Sessions:

0x2192df1f300: proto=unix_stream src=unix:1 fe=GLOBAL be=<NONE> srv=<none> ts=09 age=0s
calls=2 rq[f=c08200h,
i=0,an=00h,rx=20s,wx=,ax=] rp[f=008000h,i=0,an=00h,rx=,wx=,ax=] s0=[7,8h,fd=1,ex=]
s1=[7,0h,fd=-1,ex=] exp=19s


------------------------------------------------------------------------
L4 Loadbalancer Statistics:
MAX_CONN   ACT_CONN   INACT_CONN TOTAL_CONN
0          0          0          0

L4 Loadbalancer Current Sessions:

pro expire state       source       virtual    destination
```

b Check the `show service loadbalancer` command to view the load balancer Layer 7 sticky-table status. Note that this table does not display information on accelerated virtual servers.

```
nsxedge> show service loadbalancer table
------------------------------------------------------------------------
L7 Loadbalancer Sticky Table Status:

TABLE    TYPE    SIZE(BYTE)   USED(BYTE)
```

6 If all the required services are running properly, look at the routing table and you need to have a route to the client and to the servers. Use the `show ip route` and `show ip forwarding` commands which maps routes to the interfaces.

```
NSX-edge-8-0> sh ip route

Codes: O - OSPF derived, i - IS-IS derived, B - BGP derived,
C - connected, S - static, L1 - IS-IS level-1, L2 - IS-IS level-2,
IA - OSPF inter area, E1 - OSPF external type 1, E2 - OSPF external type 2,
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

Total number of routes: 4

S       0.0.0.0/0             [1/1]         via 192.168.8.2
C       10.10.10.0/24         [0/0]         via 10.10.10.1
C       169.254.1.4/30        [0/0]         via 169.254.1.5
C       192.168.8.0/24        [0/0]         via 192.168.8.3
NSX-edge-8-0> sh ip forwarding
Codes: C - connected, R - remote,
       > - selected route, * - FIB route

R>* 0.0.0.0/0 via 192.168.8.2, vNic_2
C>* 10.10.10.0/24 is directly connected, vNic_0
C>* 169.254.1.4/30 is directly connected, vNic_0
C>* 192.168.8.0/24 is directly connected, vNic_2
NSX-edge-8-0>
```

7 Make sure that you have an ARP entry for the systems, such as the gateway or next hop, and the backend servers using the `show arp` command.

```
OneArm-LoadBalancer-01-0>   show arp
------------------------------------------------------------------------
vShield Edge ARP Cache:
IP Address                              Interface  MAC Address        State
fe80::250:56ff:feae:f86b                vNic_0     00:50:56:ae:f8:6b  STALE
fe80::250:56ff:feae:5066                vNic_1     00:50:56:ae:50:66  STALE
fe80::250:56ff:feae:3e3d                vNic_0     00:50:56:ae:3e:3d  STALE
172.16.100.11                           vNic_1     00:50:56:ae:50:66  REACHABLE
172.16.10.1                             vNic_0     02:50:56:56:44:52  REACHABLE
172.16.10.11                            vNic_0     00:50:56:ae:3e:3d  REACHABLE
OneArm-LoadBalancer-01-0>
```

8 The logs provide information to help find traffic which might help to diagnose issues. Use the `show log` or `show log follow` commands to tail the log that will help to find the traffic. Note that you must be running the load balancer with **Logging** enabled, and set to **Info** or **Debug**.

```
nsxedge> show log
2016-04-20T20:15:36+00:00 vShieldEdge kernel: Initializing cgroup subsys cpuset
2016-04-20T20:15:36+00:00 vShieldEdge kernel: Initializing cgroup subsys cpu
2016-04-20T20:15:36+00:00 vShieldEdge kernel: Initializing cgroup subsys cpuacct
...
```

9 After verifying that the basic services are running with proper paths to the clients, lets look at what is happening in the application layer. Use the `show service loadbalancer pool` command to view the load balancer pool status (L4/L7). One pool member must be up to serve content, and usually more than one is needed as the volume of requests exceeds the capacity of single workload. If health monitor is provided by built-in health check, the output displays `last state change time` and `failure reason` when health check fails. If health monitor is provided by monitor service, beside the above two outputs, `last check time` is also displayed.

```
nsxedge> show service loadbalancer pool
------------------------------------------------------------------------
Loadbalancer Pool Statistics:

POOL Web-Tier-Pool-01
|  LB METHOD round-robin
|  LB PROTOCOL L7
|  Transparent disabled
|  SESSION (cur, max, total) = (0, 0, 0)
|  BYTES in = (0), out = (0)
   +->POOL MEMBER: Web-Tier-Pool-01/web-01a, STATUS: UP
   |  |  HEALTH MONITOR = BUILT-IN, default_https_monitor:L7OK
   |  |  |  LAST STATE CHANGE: 2016-05-16 07:02:00
   |  |  SESSION (cur, max, total) = (0, 0, 0)
   |  |  BYTES in = (0), out = (0)
   +->POOL MEMBER: Web-Tier-Pool-01/web-02a, STATUS: UP
   |  |  HEALTH MONITOR = BUILT-IN, default_https_monitor:L7OK
   |  |  |  LAST STATE CHANGE: 2016-05-16 07:02:01
   |  |  SESSION (cur, max, total) = (0, 0, 0)
   |  |  BYTES in = (0), out = (0)
```

10 Check the service monitor status (OK, WARNING, CRITICAL) to see the health of all the configured backend servers.

```
nsxedge> show service loadbalancer monitor
------------------------------------------------------------------------
Loadbalancer Health Check Statistics:

MONITOR PROVIDER   POOL               MEMBER     HEALTH STATUS
built-in           Web-Tier-Pool-01   web-01a    default_https_monitor:L7OK
built-in           Web-Tier-Pool-01   web-02a    default_https_monitor:L7OK
```

For the `show service load balancer monitor` command, three types of health monitor values are displayed in the CLI output:

- Built-in: Health check is enabled and is performed by L7 engine (HA proxy).
- Monitor Service: Health check is enabled and is performed by monitor service engine (NAGIOS). The monitor service running status can be checked with `show service monitor` and `show service monitor service` CLI commands. The **Status** field should be `OK`, `WARNING` or `CRITICAL`.
- Not Defined: Health check is disabled.

The last column of the output is the health status of the pool member. Following status are displayed:

Table 6-1. Health status with description

Health Status	Description

Built-in	■ UNK: Unknown
	■ INI: Initializing
	■ SOCKERR: Socket error
	■ L4OK: Check passed on layer 4, no upper layers testing enabled
	■ L4TOUT: Layer 1-4 timeout
	■ L4CON: Layer 1-4 connection problem. For example, "Connection refused" (tcp rst) or "No route to host" (icmp)
	■ L6OK: Check passed on layer 6
	■ L6TOUT: Layer 6 (SSL) timeout
	■ L6RSP: Layer 6 invalid response - protocol error. May caused as the:
	  ■ Backend server only supports “SSLv3” or “TLSv1.0”, or
	  ■ Certificate of the backend server is invalid, or
	  ■ The cipher negotiation failed, and so on
	■ L7OK: Check passed on layer 7
	■ L7OKC: Check conditionally passed on layer 7. For example, 404 with disable-on-404
	■ L7TOUT: Layer 7 (HTTP/SMTP) timeout
	■ L7RSP: Layer 7 invalid response - protocol error
	■ L7STS: Layer 7 response error. For example, HTTP 5xx

CRITICAL	■ SSL protocol version 2 is not supported by your SSL library ■ Unsupported SSL protocol version ■ Cannot create SSL context ■ Cannot make SSL connection ■ Cannot initiate SSL handshake ■ Cannot retrieve server certificate ■ Cannot retrieve certificate subject ■ Wrong time format in certificate ■ Certificate '<cn>' expired on <expire time of certificate> ■ Certificate '<cn>' expired today <expire time of certificate>
WARNING/CRITICAL	Certificate ‘<cn>' expires in <days_left/expire time of certificate> day(s)

Table 6-1. Health status with description (continued)

Health Status	Description
ICMP	■ Net unreachable ■ Host unreachable ■ Protocol unreachable ■ Port unreachable ■ Source route failed ■ Source host isolated ■ Unknown network ■ Unknown host ■ Network denied ■ Host denied ■ Bad type of service (ToS) for network ■ Bad type of service (ToS) for host ■ Prohibited by filter ■ Host precedence violation ■ Precedence cutoff. Minimum level of precedence required for the operation ■ Invalid code
UDP/TCP	■ Socket creation failed ■ Connect to address xxxx and port xxx: [Refer to Linux error code] ■ No data received from host ■ Unexpected response from host/socket
HTTP/HTTPS	■ HTTP UNKNOWN: Memory allocation error ■ HTTP CRITICAL: Unable to open TCP socket (create socket or connect to server failed) ■ HTTP CRITICAL: Error while receiving data ■ HTTP CRITICAL: No data received from host ■ HTTP CRITICAL: Invalid HTTP response received from host: <status line> (Incorrect expected status line format) ■ HTTP CRITICAL: Invalid status Line <status line> (status code is not 3 digits: XXX) ■ HTTP CRITICAL: Invalid status <status line> (status code >= 600 or < 100) ■ HTTP CRITICAL: String not found ■ HTTP CRITICAL: Pattern not found ■ HTTP WARNING: Page size <page_length> too large ■ HTTP WARNING: Page size <page_length> too small

11 When the error code is L4TOUT/L4CON, it is usually connectivity issues on the underlying networking. `Duplicate IP` often happens as root cause with such reason. When this error happens, troubleshoot as follows:

   a Check the High Availability (HA) status of edges, when HA is enabled by using the `show service highavailability` command on both the edges. Check if the HA link is DOWN and all the edges are `Active`, so there are no duplicate edge IP on the network.

   b Check edge ARP table by `show arp` command, and verify if the backend server's ARP entry is changed between the two MAC addresses.

   c Check backend server ARP table or use the `arp-ping` command and check whether any other machine has the same IP similar to the edge IP.

12 Check the load balancer object statistics (VIPs, pools, members). Look at the specific pool and verify that the members are up and running. Check if the transparent mode is enabled. If yes, the edge services gateway should be inline between the client and the server. Verify if the servers are showing session counter increments.

```
nsxedge> show service loadbalancer pool Web-Tier-VIP-01
TIMESTAMP               SESSIONS       BYTESIN        BYTESOUT       SESSIONRATE    HTTPREQS
2016-04-27 19:56:40     00             00             00             00             00
2016-04-27 19:55:00     00             32             100            00             00
```

```
nsxedge> show service loadbalancer pool Web-Tier-VIP-01 | MEMBER
+-> POOL MEMBER: TENANT-1-TCP-POOL-80/SERVER-1, STATUS:  UP
+-> POOL MEMBER: TENANT-1-TCP-POOL-80/SERVER-2, STATUS:  UP
```

13 Now look at the virtual server and verify if there is a default pool, and see the pool is also bound to it. If you use pools via application rules, you need to look at the specific pools as shown in the `#show service loadbalancer pool` command. Specify the name of the virtual server.

```
nsxedge> show service loadbalancer virtual Web-Tier-VIP-01

----------------------------------------------------------------------
Loadbalancer VirtualServer Statistics:

VIRTUAL Web-Tier-VIP-01
|  ADDRESS [172.16.10.10]:443
|  SESSION (cur, max, total) = (0, 0, 0)
|  RATE (cur, max, limit) = (0, 0, 0)
|  BYTES in = (0), out = (0)
   +->POOL Web-Tier-Pool-01
   |  LB METHOD round-robin
   |  LB PROTOCOL L7
   |  Transparent disabled
   |  SESSION (cur, max, total) = (0, 0, 0)
   |  BYTES in = (0), out = (0)
      +->POOL MEMBER: Web-Tier-Pool-01/web-01a, STATUS: UP
      |  |  HEALTH MONITOR = BUILT-IN, default_https_monitor:L7OK
```

```
|  |  |  LAST STATE CHANGE: 2016-05-16 07:02:00
|  |  SESSION (cur, max, total) = (0, 0, 0)
|  |  BYTES in = (0), out = (0)
+->POOL MEMBER: Web-Tier-Pool-01/web-02a, STATUS: UP
|  |  HEALTH MONITOR = BUILT-IN, default_https_monitor:L7OK
|  |  |  LAST STATE CHANGE: 2016-05-16 07:02:01
|  |  SESSION (cur, max, total) = (0, 0, 0)
|  |  BYTES in = (0), out = (0)
```

14 If everything looks to be configured correctly and still you have an error, you should capture traffic to understand what is going on. There are two connections: the client to the virtual server, and the edge services gateway to the backend pool (with or without the transparent configuration at the pool level). The `#show ip forwarding` command listed the vNic interfaces, and you can use that data.

For example, assume the client computer is on *vNic_0* and the server on *vNic_1*. You use a client IP address of *192.168.1.2*, a VIP IP of *192.168.2.2* running on port 80. Load balancer interface IP *192.168.3.1* and a backend server IP of *192.168.3.3*. There are two different packet capture commands, one displays the packets, whereas the other captures the packets to file that you can download. Capture the packets to detect the load balancer abnormal failure. You can capture packets from two directions:

- Capture the packets from client.
- Capture the packets sent to backend server.

```
#debug packet capture interface interface-name [filter using _ for space]- creates a packet
capture file that you can download
#debug packet display interface interface-name [filter using _ for space]- outputs packet data to
the console
#debug show files - to see a list of packet capture
#debug copy scp user@url:path file-name/all - to download the packet capture
```

For example:

- Capture on vNIC_0: `debug packet display interface vNic_0`
- Capture on all interfaces: `debug packet display interface any`
- Capture on vNIC_0 with a filter: `debug packet display interface vNic_0 host_192.168.11.3_and_host_192.168.11.41`
- A packet capture of the client to virtual server traffic: `#debug packet display|capture interface vNic_0 host_192.168.1.2_and_host_192.168.2.2_and_port_80`
- A packet capture between the edge services gateway and the server where the pool is in transparent mode: `#debug packet display|capture interface vNic_1 host 192.168.1.2_and_host_192.168.3.3_and_port_80`
- A packet capture between the edge services gateway and the server where the pool is not in transparent mode: `#debug packet display|capture interface vNic_1 host 192.168.3.1_and_host_192.168.3.3_and_port_80`

Common Load Balancer Issues

This topic discusses several issues and how to resolve them.

The following issues are common when using NSX load balancing:

- Load balancing on TCP port (for example, port 443) does not work.
  - Verify the topology. For details, refer to *NSX Administration Guide*.
  - Verify the virtual server IP address is reachable with ping, or look at the upstream router to ensure the ARP table is populated.
  - Load Balancer Configuration Verification and Troubleshooting Using the UI.
  - Load Balancer Troubleshooting Using the CLI.
  - Capture packets.
- A member of the load balancing pool is not utilized.
  - Verify the server is in the pool, enabled, and monitor health status.
- Edge traffic is not load balanced.
  - Verify the pool and persistence configuration. If you have persistence configured and you are using a small number of clients, you may not see even distribution of connections to backend pool members.
- Layer 7 load balancing engine is stopped.
- Health monitor engine is stopped.
  - Enable load balancer service. Refer to the *NSX Administration Guide*.
- Pool member monitor status is WARNING/CRITICAL.
  - Verify the application server is reachable from the load balancer.
  - Verify the application server firewall or DFW is allowing traffic.
  - Ensure the application server is able to respond to the specified health probe.
- Pool member has the INACTIVE status.
  - Verify the pool member is enabled in the pool configuration.
- Layer 7 sticky table is not synchronized with the standby Edge.
  - Ensure that HA is configured.
- Client connections, but cannot complete an application transaction.
  - Verify that the proper persistence is configured in the application profile.
  - If the application works with only one server in the pool (and not two), it is most likely a persistence problem.

Basic Troubleshooting

1 Check the load balancer configuration status in the vSphere Web Client:

   a Click **Networking & Security > NSX Edges**.

   b Double-click an NSX Edge.

   c Click **Manage**, and then click the **Load Balancer** tab.

   d Check the load balancer status and logging level configured.

2 Before troubleshooting the load balancer service, run the following command on the NSX Manager to ensure that the service is up an running:

```
nsxmgr> show edge edge-4 service loadbalancer
haIndex:              0
------------------------------------------------------------------------
Loadbalancer Services Status:

L7 Loadbalancer     : running
------------------------------------------------------------------------
L7 Loadbalancer Statistics:
STATUS     PID        MAX_MEM_MB MAX_SOCK   MAX_CONN   MAX_PIPE   CUR_CONN   CONN_RATE
CONN_RATE_LIMIT MAX_CONN_RATE
running    1580       0          2081       1024       0          0          0
0               0
------------------------------------------------------------------------
L4 Loadbalancer Statistics:
MAX_CONN   ACT_CONN   INACT_CONN TOTAL_CONN
0          0          0          0

Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
```

Note You can run `show edge all` to look up the names of the NSX Edges.

Troubleshooting Configuration Issues

When the load balancer configuration operation is rejected by the NSX user interface or REST API call, this is classified as a configuration issue.

Troubleshooting Data Plane Issues

The load balancer configuration is accepted by NSX Manager, but there are connectivity or performance issues among the client-edge load-balance server. Data plane issues also include load balancer runtime CLI issues and load balancer system event issues.

1 Change the Edge logging level in NSX Manager from INFO to TRACE or DEBUG using this REST API call.

```
URL: https://NSX_Manager_IP/api/1.0/services/debug/loglevel/com.vmware.vshield.edge?level=TRACE
Method: POST
```

2 Check the pool member status in the vSphere Web Client.

   a Click **Networking & Security > NSX Edges**.

   b Double-click an NSX Edge.

   c Click **Manage**, and then click the **Load Balancer** tab.

   d Click **Pools** to see a summary of the configured load balancer pools.

   e Select your load balancer pool. click **Show Pool Statistics**, and verify that the pool state is UP.

3 You can get more detailed load balancer pool configuration statistics from the NSX Manager using the following REST API call:

```
URL: https://NSX_Manager_IP/api/4.0/edges/{edgeId}/loadbalancer/statistics
Method: GET

<?xml version="1.0" encoding="UTF-8"?>
<loadBalancerStatusAndStats>
    <timeStamp>1463507779</timeStamp>
    <pool>
        <poolId>pool-1</poolId>
        <name>Web-Tier-Pool-01</name>
        <member>
            <memberId>member-1</memberId>
            <name>web-01a</name>
            <ipAddress>172.16.10.11</ipAddress>
            <status>UP</status>
            <lastStateChangeTime>2016-05-16 07:02:00</lastStateChangeTime>
            <bytesIn>0</bytesIn>
            <bytesOut>0</bytesOut>
            <curSessions>0</curSessions>
            <httpReqTotal>0</httpReqTotal>
            <httpReqRate>0</httpReqRate>
            <httpReqRateMax>0</httpReqRateMax>
            <maxSessions>0</maxSessions>
            <rate>0</rate>
            <rateLimit>0</rateLimit>
            <rateMax>0</rateMax>
            <totalSessions>0</totalSessions>
```

```
        </member>
        <member>
            <memberId>member-2</memberId>
            <name>web-02a</name>
            <ipAddress>172.16.10.12</ipAddress>
            <status>UP</status>
            <lastStateChangeTime>2016-05-16 07:02:01</lastStateChangeTime>
            <bytesIn>0</bytesIn>
            <bytesOut>0</bytesOut>
            <curSessions>0</curSessions>
            <httpReqTotal>0</httpReqTotal>
            <httpReqRate>0</httpReqRate>
            <httpReqRateMax>0</httpReqRateMax>
            <maxSessions>0</maxSessions>
            <rate>0</rate>
            <rateLimit>0</rateLimit>
            <rateMax>0</rateMax>
            <totalSessions>0</totalSessions>
        </member>
        <status>UP</status>
        <bytesIn>0</bytesIn>
        <bytesOut>0</bytesOut>
        <curSessions>0</curSessions>
        <httpReqTotal>0</httpReqTotal>
        <httpReqRate>0</httpReqRate>
        <httpReqRateMax>0</httpReqRateMax>
        <maxSessions>0</maxSessions>
        <rate>0</rate>
        <rateLimit>0</rateLimit>
        <rateMax>0</rateMax>
        <totalSessions>0</totalSessions>
    </pool>
    <virtualServer>
        <virtualServerId>virtualServer-1</virtualServerId>
        <name>Web-Tier-VIP-01</name>
        <ipAddress>172.16.10.10</ipAddress>
        <status>OPEN</status>
        <bytesIn>0</bytesIn>
        <bytesOut>0</bytesOut>
        <curSessions>0</curSessions>
        <httpReqTotal>0</httpReqTotal>
        <httpReqRate>0</httpReqRate>
        <httpReqRateMax>0</httpReqRateMax>
        <maxSessions>0</maxSessions>
        <rate>0</rate>
        <rateLimit>0</rateLimit>
        <rateMax>0</rateMax>
        <totalSessions>0</totalSessions>
    </virtualServer>
</loadBalancerStatusAndStats>
```

4 To check load balancer statistics from the command line, run the following commands on the NSX Edge.

For a particular virtual server: First run `show service loadbalancer virtual` to get the virtual server name. Then run `show statistics loadbalancer virtual <virtual-server-name>`.

For a particular TCP pool: First run `show service loadbalancer pool` to get the pool name. Then run `show statistics loadbalancer pool <pool-name>`.

5 Review the load balancer statistics for signs of failure.

Troubleshooting Virtual Private Networks (VPN) 7

NSX Edge supports several types of VPNs. This troubleshooting section describes how to troubleshoot L2 VPN and SSL VPN issues.

This chapter includes the following topics:

L2 VPN

With L2 VPN, you can stretch multiple logical L2 networks (both VLAN and VXLAN) across L3 boundaries, tunneled within an SSL VPN. In addition, you can configure multiple sites on an L2 VPN server. Virtual machines remain on the same subnet when they are moved between sites and their IP addresses do not change. You also have the option to deploy a standalone edge on a remote site without that site being “NSX Enabled”. Egress optimization enables the edge to route any packets sent towards the Egress Optimization IP address locally, and bridge everything else.

L2 VPN thus allows enterprises to seamlessly migrate workloads backed by VXLAN or VLAN between physically separated locations. For cloud providers, L2 VPN provides a mechanism to on-board tenants without modifying existing IP addresses for workloads and applications.

L2 VPN Common Configuration Issues

This topic discusses common configuration issues related to L2 VPN.

Problem

Following are common configuration issues:

- L2 VPN client is configured, but internet-facing firewall does not allow traffic to flow the tunnel via destination port 443.
- L2 VPN client is configured to validate server certificate, but it is not configured with correct CA certificate or FQDN.
- L2 VPN server is configured, but NAT / firewall rule is not created on internet facing firewall.

- Trunk interface is not backed by either a distributed port group or a standard port group.

Note L2 VPN server listens on port 443 by default. This port is configurable from L2 VPN server settings.

L2 VPN client makes an outgoing connection to port 443 by default. This port is configurable from L2 VPN client settings.

Solution

1 Check if L2 VPN server process is running.
   a Log in to NSX Edge VM.
   b Run the `show process monitor` command, and verify if you can find a process with name *l2vpn*.
   c Run the `show service network-connections` command, and verify if *l2vpn* process is listening on port 443.
2 Check if L2 VPN client process is running.
   a Log in to NSX Edge VM.
   b Run the `show process monitor` command, and verify if you can find a process with name *naclientd*.
   c Run the `show service network-connections` command, and verify if *naclientd* process is listening on port 443.
3 Check if L2 VPN server is accessible from internet.
   a Open browser, and visit **`https://<l2vpn-public-ip>`**.
   b A portal login page should be displayed. If portal page is displayed, it means that L2 VPN server is reachable over internet.
4 Check if trunk interface is backed by a distributed port group or a standard port group.
   a If the trunk interface is backed by a distributed port group, a sink port is automatically set.
   b If the trunk interface is backed by a standard port group, you should manually configure the vSphere Distributed Switch as follows:
      - Set the port to **promiscuous** mode.
      - Set the **Forged Transmits** to **Accept**.
5 Mitigate L2 VPN looping issue.
   a Two major issues are observed if NIC teaming is not configured correctly — MAC flapping, and duplicate packets. Verify configuration as described in L2VPN Options to Mitigate Looping.

6 Check if VMs across L2 VPN can communicate with each other.

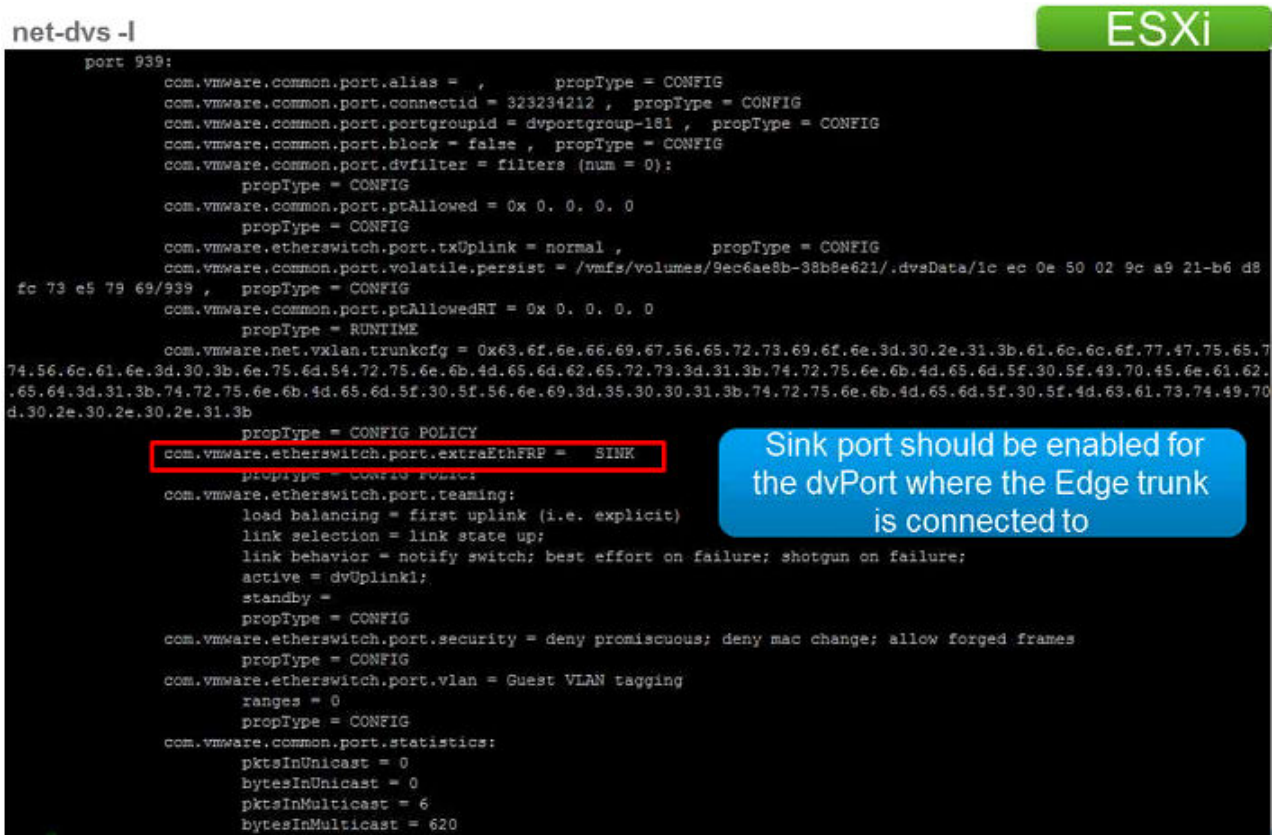
a Log in to L2 VPN server CLI, and capture packet on the corresponding tap interface `debug packet capture interface name`.

b Log in to L2 VPN client, and capture packet capture on the corresponding tap interface `debug packet capture interface name`

c Analyze these captures to check if ARP is getting resolved and data traffic flow.

d Check if `Allow Forged Transmits: dvSwitch` property is set to *L2 VPN trunk port*.

e Check if sink port is set to *L2 VPN trunk port*. To do so, log in to host and issue command `net-dvs -l`. Check for sink property set for L2 VPN edge internal port (`com.vmware.etherswitch.port.extraEthFRP = SINK`). Internal port refers to the *dvPort* where the NSX Edge trunk is connected to.

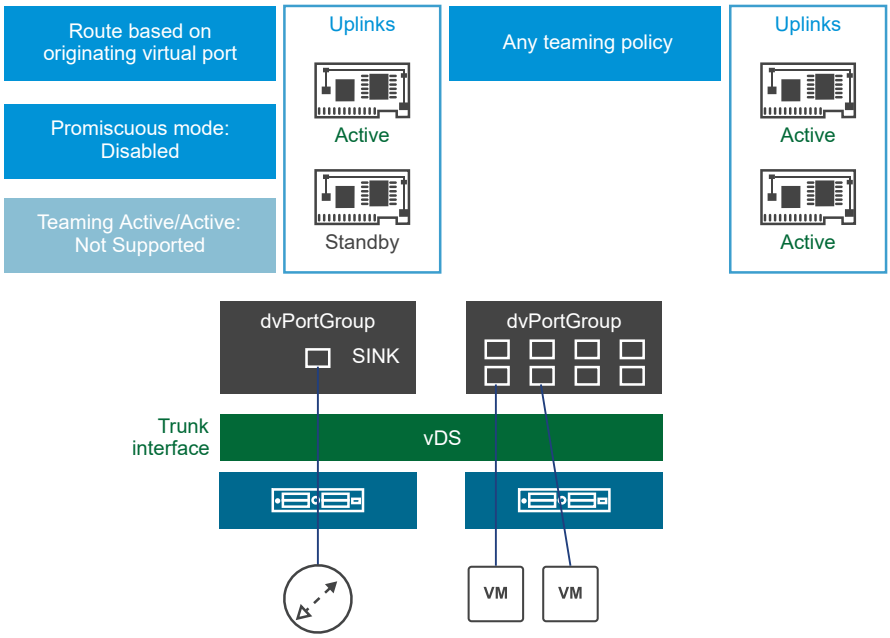


L2VPN Options to Mitigate Looping

There are two options to mitigate looping. Either the NSX Edges and VMs can be on different ESXi hosts, or the NSX Edges and VMs can be on the same ESXi host.

Option 1: Separate ESXi hosts for the L2VPN Edges and the VMs

1. Deploy L2VPN Edges and VMs on separate ESXi hosts



1 Deploy the Edges and the VMs on separate ESXi hosts.

2 Configure the Teaming and Failover Policy for the Distributed Port Group associated with the Edge's Trunk vNic as follows:

   a Load balancing as "Route based on originating virtual port."

   b Configure only one uplink as Active and the other uplink as Standby.

3 Configure the teaming and failover policy for the distributed port group associated with the VMs as follows:

   a Any teaming policy is okay.

   b Multiple active uplinks can be configured.

4 Configure Edges to use sink port mode and disable promiscuous mode on the trunk vNic.

Note

- Disable promiscuous mode: If you are using vSphere Distributed Switch.
- Enable promiscuous mode: If you are using virtual switch to configure trunk interface.

If a virtual switch has promiscuous mode enabled, some of the packets that come in from the uplinks that are not currently used by the promiscuous port, are not discarded. You should enable and then disable `ReversePathFwdCheckPromisc` that will explicitly discard all the packets coming in from the currently unused uplinks, for the promiscuous port.

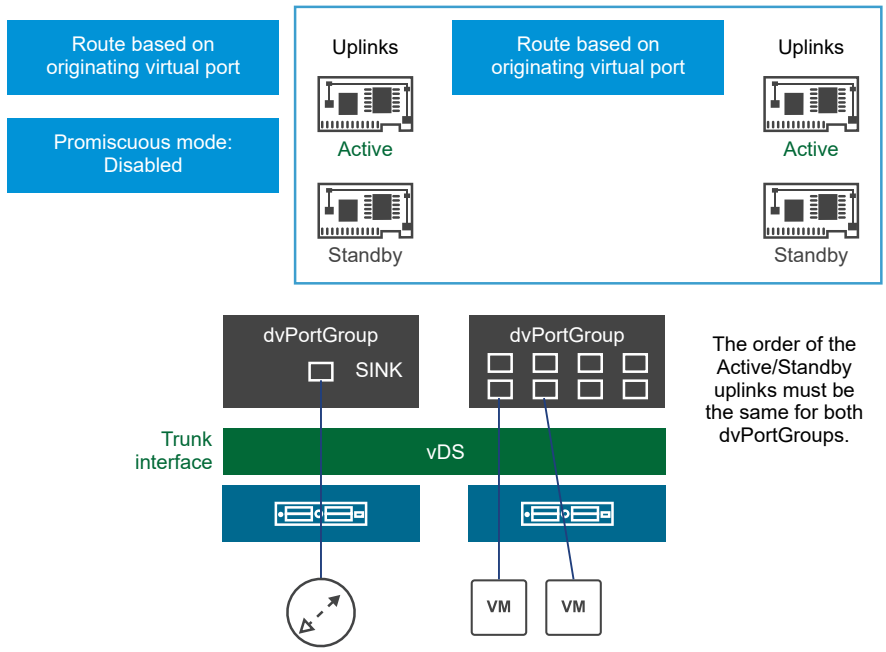
To block the duplicate packets, activate RPF check for the promiscuous mode from the ESXi CLI where NSX Edge is present:

```
esxcli system settings advanced set –o /Net/ReversePathFwdCheckPromisc –i 1
esxcli system settings advanced list –o /Net/ReversePathFwdCheckPromisc
Path: /Net/ReversePathFwdCheckPromisc
Type: integer
Int Value: 1
Default Int Value: 0
Max Value: 1
Min Value: 0
String Value:
Default String Value:
Valid Characters:
Description: Block duplicate packet in a teamed environment when the virtual switch is set to
Promiscuous mode.
```

In **PortGroup** security policy, set **PromiscousMode** from **Accept** to **Reject** and back to **Accept** to activate the configured change.

- Option 2: Edges and VMs on the same ESXi host

2. Deploy L2VPN Edges and VMs on the same host



a Configure the teaming and failover policy for the distributed port group associated with Edge's trunk vNic as follows:

1 Load balancing as "Route based on originating virtual port."

2 Configure one uplink as active and the other uplink as standby.

b Configure the teaming and failover policy for the distributed port group associated with the VMs as follows:

1 Any teaming policy is okay.

2 Only one uplink can be active.

3 The order of the active/standby uplinks must be the same for the VMs' distributed port group and the Edge's trunk vNic distributed port group.

c Configure the client-side standalone edge to use sink port mode and disable promiscuous mode on the trunk vNic.

Troubleshooting Using the CLI

You can use the NSX Command Line Interface (CLI) to do some L2 VPN troubleshooting.

Problem

L2 VPN is not working as expected.

Solution

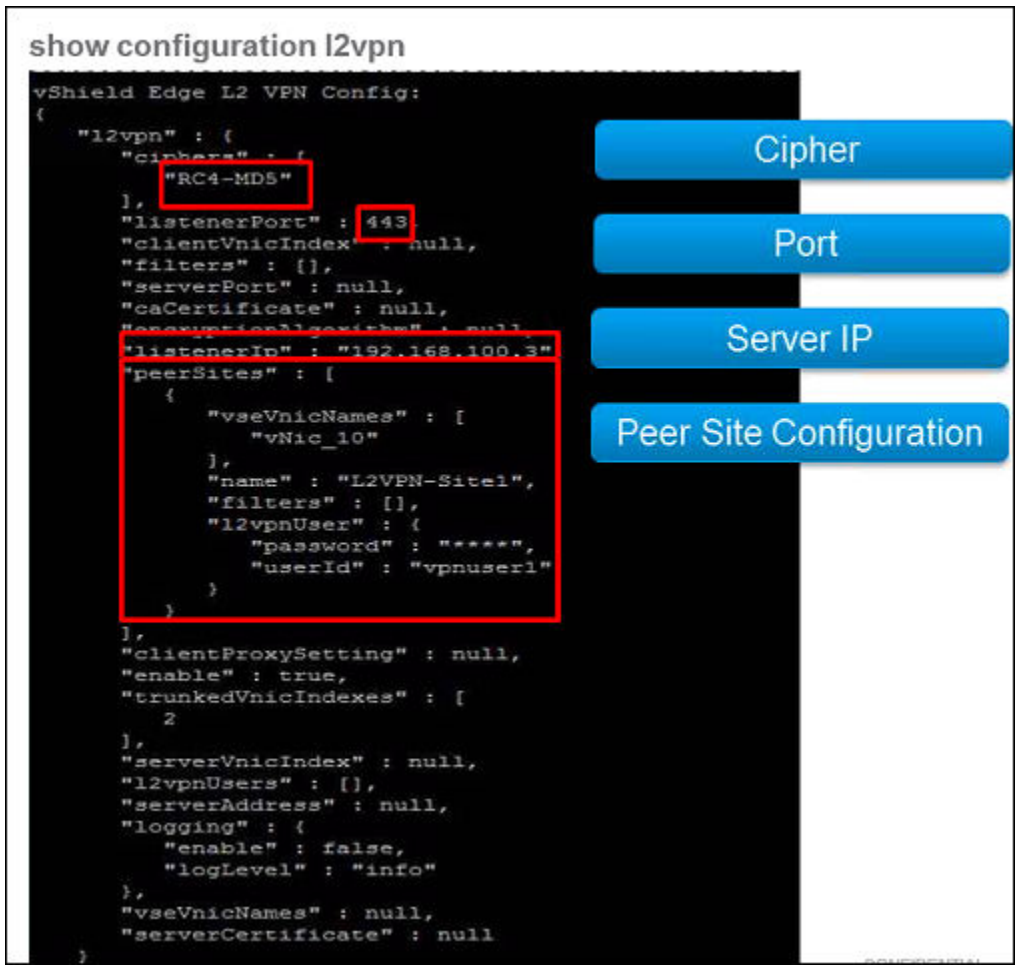
1 Use the following central CLI command to see configuration issues:

```
show edge <edgeID> configuration l2vpn.
```

For example, `show edge edge-1 configuration l2vpn`.

2 Use the following commands on both the client and server edge:

- `show configuration l2vpn` - Check the four following key values to verify the server.



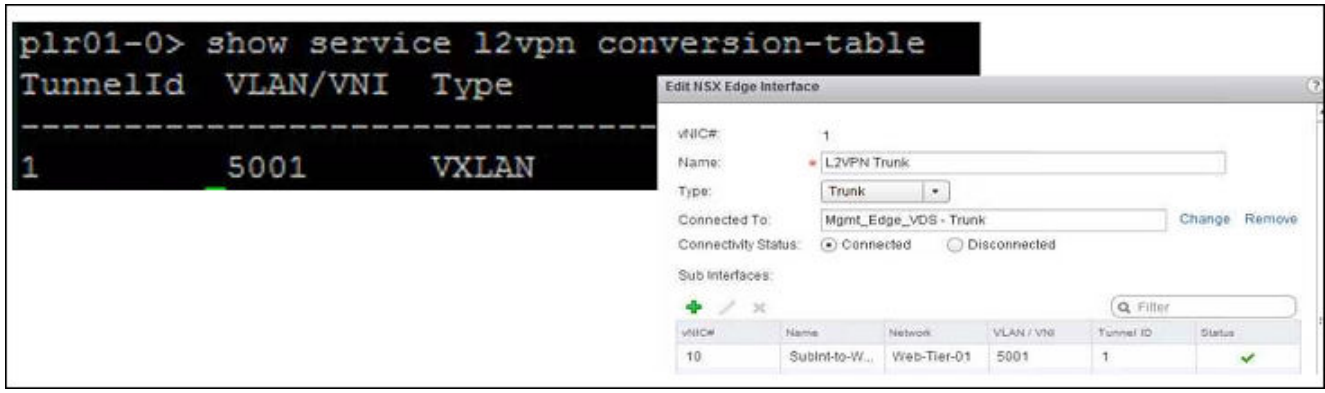
- `show service l2vpn bridge` - The number of interfaces depends on the number of L2 VPN clients. In below output, a single L2 VPN client (na1) is configured. Port1 refers to *vNic_2*. The MAC address of 02:50:56:56:44:52 has been learned on the *vNic_2* interface, and is not local to the edge (L2 VPN server). Row 3 in the following example refers to *na1* interface.

```
plr01-0> show service l2vpn bridge

bridge name     bridge id               STP enabled     interfaces
br-sub          8000.0050568e19fb       no              vNic_2
                                                        na1

List of learned MAC addresses for L2 VPN bridge br-sub
-----------------------------------------------------------------------
port no mac addr                is local?       vlanid          ageing timer
  1     00:50:56:8e:19:fb       yes             0                 0.00
  1     02:50:56:56:44:52       no              1                 0.87
  2     2a:56:30:31:7e:3b       yes             0                 0.00
```

- `show service l2vpn trunk table`
- `show service l2vpn conversion table` - In the following example, an Ethernet frame which arrives on tunnel #1 will have its VLAN ID #1 converted to VXLAN with a VLAN # of 5001 before the packet is passed to the VDS.



- `show process monitor` - Identify if the l2vpn (server) and naclientd (client) processes are running.
- `show service network-connections` - Identify if the l2vpn (server) and naclientd (client) processes are listening on port 443.

SSL VPN

You can use this information to troubleshoot problems with your setup.

SSL VPN Web Portal Does Not Open

SSL VPN users are unable to open the SSL VPN web portal login page to download and install the SSL VPN-Plus client installation package.

Problem

The SSL VPN Web portal login page does not open, or the page renders incorrectly in your system browser.

Cause

One of the following reasons can cause this problem:

- Your system is using an unsupported browser version.
- Cookies and JavaScript are not enabled in your browser.

Solution

1 Make sure that you open the SSL VPN web portal login page in any of the following supported browsers.

Browser	Minimum Supported Versions
Internet Explorer	9.0.8112.16421
Chrome	67.03396
Safari	10.x

2 Open your browser settings, and ensure that cookies and JavaScript are enabled.

3 If the browser language is not set to English, set the language to English, and see if the issue persists.

4 Check whether you have selected AES cipher on the SSL VPN server. Some browsers do not support AES encryption.

SSL VPN-Plus: Installation Failures

Use this topic to understand probable SSL VPN-Plus client-specific installation problems and how you can resolve them.

Problem

Common problems associated with SSL VPN-Plus client installation are as follows:

- SSL VPN-Plus client is installed successfully, but the client does not work.
- On Mac machines, kernel extension warning messages are displayed.
- On Mac OS High Sierra, the following installation error messages are displayed:

```
/opt/sslvpn-plus/naclient/signed_kext/tap.kext failed to load - (libkern/kext)system policy
prevents
loading; check the system/kernel logs for errors or try kextutil(8).
Error: Could not load /opt/sslvpn-plus/naclient/signed_kext/tap.kext
```

```
installer[4571] <Debug>: install:didFailWithError:Error Domain=
PKInstallErrorDomain Code=112 "An error occurred while running scripts from the package
“naclient.pkg”.
" UserInfo={NSFilePath=./postinstall,NSURL=file://<pathtofile>/
naclient.pkg,PKInstallPackageIdentifier=
com.vmware.sslvpn,NSLocalizedDescription=An error occurred while running scripts from the
package “naclient.pkg”.}

installer[4571] <Error>: Install failed: The Installer encountered an error that caused the
installation to fail. Contact the software manufacturer for assistance.
installer: The install failed (The Installer encountered an error that caused the installation to
fail.
Contact the software manufacturer for assistance.)
```

- On Windows machines, the following error message is displayed: `Driver installation failed for reason E000024B: please try rebooting the machine.`

Cause

One of the following reasons can cause the SSL VPN-Plus client to fail even after you have installed it successfully on your computer:

- Configuration file (naclient.cfg) is missing or the configuration file is invalid.
- Directory permissions or user permissions are incorrect.
- SSL VPN server is not reachable.
- On Mac and Linux machines, the tap driver is not loaded.

On Mac machines, kernel extension warning messages are displayed because your system blocks loading the kernel extension.

On Mac OS High Sierra, installation errors are displayed when your Mac machine does not allow kext, and neither does it prompt you to load the kext.

On Windows machines, driver installation failure (E000024B) is displayed because you have enabled the **Hide SSL client network adapter** option in the Edge SSL VPN-Plus Client installer.

Solution

1. Ensure that you install the SSL VPN-Plus client on supported operating systems. For information about supported operating systems, see the SSL VPN-Plus Overview topic in the *NSX Administration Guide*.
2. On Windows machines, make sure that users who install the SSL VPN-Plus client have **administrator** privileges. On Mac and Linux machines, users must have **root** privileges to install the SSL VPN-Plus client. In addition, for the SSL VPN-Plus client to start and run successfully on Mac machines, users must have **execute** permissions on the `usr/local/lib` directory.
3. On Linux machines, make sure that the following libraries are installed. These libraries are required for the UI to work.
   - TCL
   - TK
   - NSS
4. If the tap driver is not loaded on Mac and Linux machines, run the shell script to load the driver.

Operating System	Description
Mac	Run the `Naclient.sh` shell script from the `/opt/sslvpn-plus/naclient/` directory with **sudo** privileges.
Linux	Run the `naclient.sh` shell script with **sudo** privileges. You can find this script in the `linux_phat_client/linux_phat_client` directory.

5 To resolve the kernel extension warning messages on machines with macOS High Sierra or later, you must provide explicit user approval for loading a kernel extension (kext). Do the following steps:

   a On your Mac machine, open the **System Preferences > Security & Privacy** window.

   b At the bottom of the window, you can see a message similar to `"Some system software was blocked from loading." Click the "Allow" button`.

   c To proceed with the installation, click **Allow**.

   For detailed information about providing user approval for loading a kernel extension, see https://developer.apple.com/library/content/technotes/tn2459/_index.html.

   d While the kernel extension is being loaded, the SSL VPN-Plus client installation process continues to run in the background. The SSL VPN-Plus client gets installed, but you get the following error message: `The installation failed. The installer encountered an error that cause the installation to fail. Contact the software manufacturer for assistance.`

   e To resolve this error, uninstall the SSL VPN-Plus client, and reinstall it.

6 To resolve installation error messages on Mac OS High Sierra, do these steps.

   a Make sure that notifications are enabled. Go to **System Preferences > Security & Privacy > Allow Notifications**.

   Note When you install SSL VPN-Plus client for the first time on Mac OS High Sierra, a notification window prompts you to allow the installation. This notification usually lasts for 30 minutes. If the notification disappears before you clicked **Allow**, restart your machine and reinstall the SSL VPN-Plus client.

   If the installation still fails, it implies that your system does not allow kernel extension (kext), and neither does it prompt you to load the kext. Complete the remaining substeps to add `tuntap kext team id` to the pre-approved kext list.

   b Restart your Mac machine in recovery mode.

      1 Click the Apple logo at the top left of your screen.

      2 Click **Restart**.

      3 Immediately press the Command and R keys until you see an Apple logo or a spinning globe. A spinning globe appears when your Mac machine tries to start macOS recovery by connecting to the Internet because it is unable to start through the built-in recovery system. Mac is now started in recovery mode.

   c On the top bar, click **Utlities > Terminal**.

   d To add `tuntap kext team id` to the pre-approved kext list, run the `– spctl kext–consent add KS8XL6T9FZ` command.

   e Restart your Mac machine in normal mode.

f To verify whether the team-id is seen in the pre-approved kext list, run the `– spctl kext–consent list` command.

g Install the SSL VPN-Plus client package.

7 On Windows machines, if you see the driver installation failure error (E00024B), disable the **Hide SSL client network adapter** option in the Edge SSL VPN-Plus Client installer. For instructions about disabling this option, see the VMware knowledge base article at https://kb.vmware.com/s/article/2108766.

SSL VPN-Plus: Communication Issues

Use this topic to understand probable SSL VPN connectivity and data path issues and how you can resolve them.

Problem

Common problems associated with SSL VPN connectivity and data path are as follows:

- SSL VPN-Plus client is unable to connect to the SSL VPN server.
- SSL VPN-Plus client is installed, but the SSL VPN-Plus services are not running.
- Maximum count of logged-in users is reached. The SSL VPN web portal or the SSL VPN-Plus client displays the following message:

  `Maximum users reached/Maximum count of logged in user reached as per SSL VPN license. Please try after some time` or `SSL read has failed`.
- SSL VPN services are running, but the data path is not working.
- SSL VPN connection is established, but applications in the private network are not accessible.

Solution

1 If the SSL VPN-Plus client is unable to connect to the SSL VPN server, do the following:

- Make sure that the SSL VPN user is logging in with the correct user name and password.
- Check whether the SSL VPN user is valid.
- Verify whether the SSL VPN user can reach the SSL VPN server by using the web portal.

2 On the NSX Edge, do the following steps to verify whether the SSL VPN process is running.

   a Log in to the NSX Edge from the CLI. For more information about logging in to the Edge CLI, see the *NSX Command Line Interface Reference*.

   b Run the `show process monitor` command, and locate the *sslvpn* process.

   c Run the `show service network-connections` command, and check if the *sslvpn* process is listed on port 443.

   Note By default, your system uses port 443 for SSL traffic. However, if you have configured a different TCP port for SSL traffic, make sure that the *sslvpn* process is listed on that TCP port number.

3 On the SSL VPN-Plus client, verify whether the SSL VPN-Plus services are running.

Operating System	Description
Windows	Open the **Task Manager**, and check whether the SSL VPN-Plus Client service is started.
Mac	■ Make sure that the `naclientd` process is started for the daemon. ■ Make sure that the `naclient` process is started for the GUI. To check whether the processes are running, run the `ps –ef | grep "naclient"` command.
Linux	■ Make sure that the `naclientd` and `naclient_poll` processes are started. ■ To check whether the processes are running, run the `ps –ef | grep "naclient"` command.

   If the services are not running, run the following commands to start the services.

Operating System	Command
Mac	Run the `sudo launchctl load –w /Library/LaunchDaemons/com.vmware.naclientd.plist` command.
Linux	Run the `sudo service naclient start` command.

4 If the maximum count of logged-in SSL VPN users is reached, increase the number of concurrent users (CCU) by increasing the NSX Edge form factor.

   For more information, see the *NSX Administration Guide*. Note that the connected users get disconnected from VPN when you perform this operation.

5 If the SSL VPN services are running, but the data path is not working, do the following steps:

   a Check whether a virtual IP is assigned after a successful connection.

   b Verify whether the routes are added.

6 When applications in the private (back-end) network are not accessible, do the following steps to resolve the issue:

a Make sure that the private network and IP pool are not in the same subnet.

b If the administrator has not defined an IP pool, or if the IP pool is exhausted, do these steps.

1 Log in to the vSphere Web Client.

2 Click **Networking & Security**, and then click **NSX Edges**.

3 Double-click an NSX Edge, and then click the **SSL VPN-Plus** tab.

4 Add a static IP pool as explained in Add an IP Pool topic in the *NSX Administration Guide*. Make sure that you add the IP address in the **Gateway** text box. The gateway IP address is assigned to *na0* interface. All non-TCP traffic flows through the virtual adapter named as *na0* interface. You can create multiple IP pools with different gateway IP addresses, but assigned to the same *na0* interface.

5 Use the `show interface na0` command to verify the provided IP addresses, and check whether all the IP pools are assigned to the same *na0* interface.

6 Log in to the client machine, go to the **SSL VPN-Plus Client - Statistics** screen and verify the assigned virtual IP address.

c Log in to the NSX Edge Command Line Interface (CLI), and take a packet capture on na0 interface by running the `debug packet capture interface na0` command. You can also capture packets by using the **Packet Capture** tool. For details, see the *NSX Administration Guide*.

Note Packet capture continues to run in the background until you stop the capture by running the `no debug packet capture interface na0` command.

d If TCP Optimization is not enabled, verify firewall rules.

e For non-TCP traffic, make sure that the back-end network has the default gateway set as an internal interface of the Edge.

f For Mac and Linux clients, log in to the system on which the SSL VPN client is installed, and take packet capture on the tap0 interface or on the virtual adapter by running the `tcpdump –i tap0 –s 1500 –w filepath` command. On Windows clients, use a packet analyzer tool, such as Wireshark, and capture packets on the SSL VPN-Plus Client adapter.

7 If all the above steps do not resolve the issue, use the following NSX Edge CLI commands to troubleshoot further.

Purpose	Command
Check the SSL VPN status.	`show service sslvpn–plus`
Check the SSL VPN statistics.	`show service sslvpn–plus stats`

Purpose	Command
Check VPN clients that are connected.	`show service sslvpn-plus tunnels`
Check SSL VPN-Plus sessions.	`show service sslvpn-plus sessions`

SSL VPN-Plus: Authentication Issues

You experience problems with SSL VPN-Plus authentication.

Problem

SSL VPN-Plus authentication fails.

Solution

- For authentication issues, verify the following settings:
  a. Ensure that the external authentication server is reachable from the NSX Edge. From the NSX Edge, ping the authentication server and verify if the server is reachable.
  b. Check the external authentication server configuration using tools such as the LDAP browser and see if the configuration works. Only LDAP and AD authentication servers can be checked using the LDAP browser.
  c. Ensure that the local authentication server is set to lowest priority if configured in authentication process.
  d. If using Active Directory (AD), set it to `no-ssl` mode and take packet capture on the interface from which AD Server is reachable.
  e. If authentication is successful in the syslog server, you see a message similar to: `Log Output - SVP_LOG_NOTICE, 10-28-2013,09:28:39,Authentication,a,-,-,10.112.243.61,-,PHAT,,SUCCESS,,,10-28-2013,09:28:39,-,-,,,,,,,,,-,,-,`
  f. If authentication fails, in the syslog server, you see a message similar to: `Log Output - SVP_LOG_NOTICE, 10-28-2013,09:28:39,Authentication,a,-,-,10.112.243.61,-,PHAT,,FAILURE,,,10-28-2013,09:28:39,-,-,,,,,,,,,-,,-,`

SSL VPN-Plus Client Stops Responding

SSL VPN-Plus client stops responding when TCP optimization is enabled.

Problem

You have configured SSL VPN-Plus service to run on an NSX Edge and enabled TCP optimization for sending traffic through the tunnel. The SSL VPN-Plus client stops responding when you run any network performance measurement and tuning tool (for example, iperf3) on the SSL VPN-Plus client.

Cause

One of the following two scenarios can cause the tunnel read error to occur when data is sent from the SSL VPN-Plus client:

- The back-end server closes the TCP connection with the SSL VPN server by sending a TCP FIN sequence.
- Tunnels write operation fails while forwarding data to the back-end server.

The tunnel read error is `unknown protocol ID`. This error clears the tunnel between the SSL VPN server and the SSL VPN-Plus client, which in turn causes the SSL read/write operations to fail on the client, and the SSL VPN-Plus client stops responding.

Solution

- To resolve this issue, follow these steps in the vSphere Web Client to disable TCP optimization for the private network traffic through the SSL VPN tunnel.
  a. Double-click the NSX Edge VM on which you have configured the SSL VPN-Plus service.
  b. Click the **SSL VPN-Plus** tab, and then select the private network.
  c. Clear the **Enable TCP Optimization** check box.

Basic Log Analysis

SSL VPN-Plus gateway logs are sent to the syslog server configured on the NSX Edge appliance. SSL VPN-Plus client logs are stored in the following directory on the remote user's computer: `C:\Users\`*`username`*`\AppData\Local\VMware\vpn\svp_client.log`.

Basic Log Analysis - Authentication

Authentication Success

- The following log output shows that the user *a* is successfully authenticated with Network Access Client on *28th of October 2016* at *0928* hour.

```
SVP_LOG_NOTICE,10-28-2016,09:28:39,Authentication,a,-,-,10.112.243.61,-,PHAT,,SUCCESS,,,10-28-2016,09:28:39,-,-,,,,,,,,,,,-,,-,-
```

Authentication Failure

- The following log output shows that the user *a* failed to authenticate with Network Access Client on *28th of October 2016* at *0928* hour.

```
SVP_LOG_NOTICE,10-28-2016,09:28:39,Authentication,a,-,-,10.112.243.61,-,PHAT,,FAILURE,,,10-28-2016,09:28:39,-,-,,,,,,,,,,,-,,-,-
```

To resolve authentication issues, refer to SSL VPN-Plus: Installation Failures.

Basic Log Analysis - Data Path

Data Path Success

- The following log output shows that the user *a* is successfully connected with Network Access Client over TCP on *28th of October 2016* at *0941* hour to the back end web server *192.168.10.8* .

```
SVP_LOG_INFO,10-28-2016,09:41:03,TCP
Connect,a,-,-,10.112.243.61,-,PHAT,,SUCCESS,,,10-28-2013,09:41:03,-,-,192.168.10.8,8
0,,,,,,,,,,-,,-,-
```

Data Path Failure

- The following log output shows that the user *a* failed to connect with Network Access Client over TCP on *28th of October 2016* at *0941* hour to the back end web server *192.168.10.8* .

```
SVP_LOG_INFO,10-28-2016,09:41:03,TCP
Connect,a,-,-,10.112.243.61,-,PHAT,,FAILURE,,,10-28-2013,09:41:03,-,-,192.168.10.8,8
0,,,,,,,,,,-,,-,-
```

IPSec VPN

Use this information to help you troubleshoot negotiation problems with your setup.

Successful Negotiation (both Phase 1 and Phase 2)

The following examples display a successful negotiating result between NSX Edge and a Cisco device.

NSX Edge

From the NSX Edge command line interface (ipsec auto -status, part of show service ipsec command):

```
000 #2: "s1-c1":500 STATE_QUICK_I2 (sent QI2, IPsec SA established);
      EVENT_SA_REPLACE in 2430s; newest IPSEC; eroute owner; isakmp#1; idle;
      import:admin initiate
000 #2: "s1-c1" esp.f5f6877d@10.20.131.62 esp.7aaf335f@10.20.129.80
      tun.0@10.20.131.62 tun.0@10.20.129.80 ref=0 refhim=4294901761
000 #1: "s1-c1":500 STATE_MAIN_I4 (ISAKMP SA established); EVENT_SA_REPLACE in
      27623s; newest ISAKMP; lastdpd=0s(seq in:0 out:0); idle;
      import:admin initiate
```

Cisco

```
ciscoasa# show crypto isakmp sa detail

    Active SA: 1
    Rekey SA: 0 (A tunnel will report 1 Active and 1 Rekey SA during rekey)
Total IKE SA: 1

    IKE Peer: 10.20.129.80
    Type : L2L        Role    : responder
    Rekey : no        State   : MM_ACTIVE
    Encrypt : 3des    Hash    : SHA
```

```
    Auth : preshared  Lifetime: 28800
    Lifetime Remaining: 28379
```

Phase 1 Policy Not Matching

The following lists Phase 1 Policy Not Matching Error logs.

NSX Edge

NSX Edge hangs in STATE_MAIN_I1 state. Look in /var/log/messages for information showing that the peer sent back an IKE message with "NO_PROPOSAL_CHOSEN" set.

```
000 #1: "s1-c1":500 STATE_MAIN_I1 (sent MI1,
      expecting MR1); EVENT_RETRANSMIT in 7s; nodpd; idle;
      import:admin initiate
000 #1: pending Phase 2 for "s1-c1" replacing #0
Aug 26 12:31:25 weiqing-desktop ipsec[6569]:
      | got payload 0x800(ISAKMP_NEXT_N) needed: 0x0 opt: 0x0
Aug 26 12:31:25 weiqing-desktop ipsec[6569]:
      | ***parse ISAKMP Notification Payload:
Aug 26 12:31:25 weiqing-desktop ipsec[6569]:
      |    next payload type: ISAKMP_NEXT_NONE
Aug 26 12:31:25 weiqing-desktop ipsec[6569]: |    length: 96
Aug 26 12:31:25 weiqing-desktop ipsec[6569]:
      |    DOI: ISAKMP_DOI_IPSEC
Aug 26 12:31:25 weiqing-desktop ipsec[6569]: |    protocol ID: 0
Aug 26 12:31:25 weiqing-desktop ipsec[6569]: |    SPI size: 0
Aug 26 12:31:25 weiqing-desktop ipsec[6569]:
      |    Notify Message Type: NO_PROPOSAL_CHOSEN
Aug 26 12:31:25 weiqing-desktop ipsec[6569]:
      "s1-c1" #1: ignoring informational payload,
       type NO_PROPOSAL_CHOSEN msgid=00000000
```

Cisco

If debug crypto is enabled, an error message is printed to show that no proposals were accepted.

```
ciscoasa# Aug 26 18:17:27 [IKEv1]:
      IP = 10.20.129.80, IKE_DECODE RECEIVED
      Message (msgid=0) with payloads : HDR + SA (1)
      + VENDOR (13) + VENDOR (13) + NONE (0) total length : 148
Aug 26 18:17:27 [IKEv1 DEBUG]: IP = 10.20.129.80,
      processing SA payload
Aug 26 18:17:27 [IKEv1]: Phase 1 failure:  Mismatched attribute
      types for class Group Description:  Rcv'd: Group 5
      Cfg'd: Group 2
Aug 26 18:17:27 [IKEv1]: Phase 1 failure:  Mismatched attribute
      types for class Group Description:  Rcv'd: Group 5
      Cfg'd: Group 2
Aug 26 18:17:27 [IKEv1]: IP = 10.20.129.80, IKE_DECODE SENDING
      Message (msgid=0) with payloads : HDR + NOTIFY (11)
      + NONE (0) total length : 124
Aug 26 18:17:27 [IKEv1 DEBUG]: IP = 10.20.129.80,
      All SA proposals found unacceptable
```

```
Aug 26 18:17:27 [IKEv1]: IP = 10.20.129.80, Error processing
      payload: Payload ID: 1
Aug 26 18:17:27 [IKEv1 DEBUG]: IP = 10.20.129.80, IKE MM Responder
      FSM error history (struct &0xd8355a60)  <state>, <event>:
      MM_DONE, EV_ERROR-->MM_START, EV_RCV_MSG-->MM_START,
      EV_START_MM-->MM_START, EV_START_MM-->MM_START,
      EV_START_MM-->MM_START, EV_START_MM-->MM_START,
      EV_START_MM-->MM_START, EV_START_MM
Aug 26 18:17:27 [IKEv1 DEBUG]: IP = 10.20.129.80, IKE SA
      MM:9e0e4511 terminating:  flags 0x01000002, refcnt 0,
      tuncnt 0
Aug 26 18:17:27 [IKEv1 DEBUG]: IP = 10.20.129.80, sending
      delete/delete with reason message
```

Phase 2 Not Matching

The following lists Phase 2 Policy Not Matching Error logs.

NSX Edge

NSX Edge hangs at STATE_QUICK_I1. A log message shows that the peer sent a NO_PROPOSAL_CHOSEN message.

```
000 #2: "s1-c1":500 STATE_QUICK_I1 (sent QI1, expecting
      QR1); EVENT_RETRANSMIT in 11s; lastdpd=-1s(seq in:0 out:0);
      idle; import:admin initiate
Aug 26 12:33:54 weiqing-desktop ipsec[6933]: | got payload
      0x800(ISAKMP_NEXT_N) needed: 0x0 opt: 0x0
Aug 26 12:33:54 weiqing-desktop ipsec[6933]: | ***parse
      ISAKMP Notification Payload:
Aug 26 12:33:54 weiqing-desktop ipsec[6933]: |    next payload
      type: ISAKMP_NEXT_NONE
Aug 26 12:33:54 weiqing-desktop ipsec[6933]: |    length: 32
Aug 26 12:33:54 weiqing-desktop ipsec[6933]:
      |    DOI: ISAKMP_DOI_IPSEC
Aug 26 12:33:54 weiqing-desktop ipsec[6933]: |    protocol ID: 3
Aug 26 12:33:54 weiqing-desktop ipsec[6933]: |    SPI size: 16
Aug 26 12:33:54 weiqing-desktop ipsec[6933]: |    Notify Message
      Type: NO_PROPOSAL_CHOSEN
Aug 26 12:33:54 weiqing-desktop ipsec[6933]: "s1-c1" #3:
      ignoring informational payload, type NO_PROPOSAL_CHOSEN
      msgid=00000000
```

Cisco

Debug message show that Phase 1 is completed, but Phase 2 failed because of policy negotiation failure.

```
Aug 26 16:03:49 [IKEv1]: Group = 10.20.129.80,
      IP = 10.20.129.80, PHASE 1 COMPLETED
Aug 26 16:03:49 [IKEv1]: IP = 10.20.129.80, Keep-alive type
      for this connection: DPD
Aug 26 16:03:49 [IKEv1 DEBUG]: Group = 10.20.129.80,
      IP = 10.20.129.80, Starting P1 rekey timer: 21600 seconds
Aug 26 16:03:49 [IKEv1]: IP = 10.20.129.80, IKE_DECODE RECEIVED
```

```
      Message (msgid=b2cdcb13) with payloads : HDR + HASH (8)
      + SA (1) + NONCE (10) + KE (4) + ID (5) + ID (5) + NONE (0)
      total length : 288
.
.
.
Aug 26 16:03:49 [IKEv1]: Group = 10.20.129.80, IP = 10.20.129.80,
      Session is being torn down. Reason: Phase 2 Mismatch
```

PFS Mismatch

The following lists PFS Mismatch Error logs.

NSX Edge

PFS is negotiated as part of Phase 2. If PFS does not match, the behavior is similar to the failure case described in Phase 2 Not Matching.

```
000 #4: "s1-c1":500 STATE_QUICK_I1 (sent QI1, expecting
      QR1); EVENT_RETRANSMIT in 8s; lastdpd=-1s(seq in:0 out:0);
      idle; import:admin initiate
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: | got payload 0x800
      (ISAKMP_NEXT_N) needed: 0x0 opt: 0x0
Aug 26 12:35:52 weiqing-desktop ipsec[7312]:
      | ***parse ISAKMP Notification Payload:
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: |    next payload
      type: ISAKMP_NEXT_NONE
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: |    length: 32
Aug 26 12:35:52 weiqing-desktop ipsec[7312]:
      |    DOI: ISAKMP_DOI_IPSEC
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: |    protocol ID: 3
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: |    SPI size: 16
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: |    Notify Message
      Type: NO_PROPOSAL_CHOSEN
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: "s1-c1" #1: ignoring
      informational payload, type NO_PROPOSAL_CHOSEN
      msgid=00000000
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: | info:  fa 16 b3 e5
      91 a9 b0 02  a3 30 e1 d9  6e 5a 13 d4
Aug 26 12:35:52 weiqing-desktop ipsec[7312]: | info:  93 e5 e4 d7
Aug 26 12:35:52 weiqing-desktop ipsec[7312]:
      | processing informational NO_PROPOSAL_CHOSEN (14)
```

Cisco

```
<BS>Aug 26 19:00:26 [IKEv1 DEBUG]: Group = 10.20.129.80,
      IP = 10.20.129.80, sending delete/delete with
      reason message
Aug 26 19:00:26 [IKEv1 DEBUG]: Group = 10.20.129.80,
      IP = 10.20.129.80, constructing blank hash payload
Aug 26 19:00:26 [IKEv1 DEBUG]: Group = 10.20.129.80,
      IP = 10.20.129.80, constructing blank hash payload
```

```
Aug 26 19:00:26 [IKEv1 DEBUG]: Group = 10.20.129.80,
      IP = 10.20.129.80, constructing IKE delete payload
Aug 26 19:00:26 [IKEv1 DEBUG]: Group = 10.20.129.80,
      IP = 10.20.129.80, constructing qm hash payload
Aug 26 19:00:26 [IKEv1]: IP = 10.20.129.80, IKE_DECODE SENDING
      Message (msgid=19eb1e59) with payloads : HDR + HASH (8)
      + DELETE (12) + NONE (0) total length : 80
Aug 26 19:00:26 [IKEv1]: Group = 10.20.129.80, IP = 10.20.129.80,
      Session is being torn down. Reason: Phase 2 Mismatch
```

PSK not Matching

The following lists PSK Not Matching Error logs

NSX Edge

PSK is negotiated in the last round of Phase 1. If PSK negotiation fails, NSX Edge state is STATE_MAIN_I4. The peer sends a message containing INVALID_ID_INFORMATION.

```
Aug 26 11:55:55 weiqing-desktop ipsec[3855]:
      "s1-c1" #1: transition from state STATE_MAIN_I3 to
      state STATE_MAIN_I4
Aug 26 11:55:55 weiqing-desktop ipsec[3855]: "s1-c1" #1:
      STATE_MAIN_I4: ISAKMP SA established
      {auth=OAKLEY_PRESHARED_KEY
      cipher=oakley_3des_cbc_192 prf=oakley_sha group=modp1024}
Aug 26 11:55:55 weiqing-desktop ipsec[3855]: "s1-c1" #1: Dead Peer
      Detection (RFC 3706): enabled
Aug 26 11:55:55 weiqing-desktop ipsec[3855]: "s1-c1" #2:
      initiating Quick Mode PSK+ENCRYPT+TUNNEL+PFS+UP+SAREFTRACK
      {using isakmp#1 msgid:e8add10e proposal=3DES(3)_192-SHA1(2)_160
      pfsgroup=OAKLEY_GROUP_MODP1024}
Aug 26 11:55:55 weiqing-desktop ipsec[3855]: "s1-c1" #1:
      ignoring informational payload, type INVALID_ID_INFORMATION
      msgid=00000000
```

Cisco

```
Aug 26 15:27:07 [IKEv1]: IP = 10.115.199.191,
      IKE_DECODE SENDING Message (msgid=0) with payloads : HDR
      + KE (4) + NONCE (10) + VENDOR (13) + VENDOR (13)
      + VENDOR (13) + VENDOR (13) + NAT-D (130) + NAT-D (130)
      + NONE (0) total length : 304
Aug 26 15:27:07 [IKEv1]: Group = 10.115.199.191,
      IP = 10.115.199.191, Received encrypted Oakley Main Mode
      packet with invalid payloads, MessID = 0
Aug 26 15:27:07 [IKEv1]: IP = 10.115.199.191, IKE_DECODE SENDING
      Message (msgid=0) with payloads : HDR + NOTIFY (11)
      + NONE (0) total length : 80
Aug 26 15:27:07 [IKEv1]: Group = 10.115.199.191,
      IP = 10.115.199.191, ERROR, had problems decrypting
```

```
      packet, probably due to mismatched pre-shared key.
      Aborting
```

Packet Capture for a Successful Negotiation

The following lists a packet capture session for a successful negotiation between NSX Edge and a Cisco device.

```
No.     Time        Source        Destination   Protocol Info
9203    768.394800  10.20.129.80  10.20.131.62  ISAKMP  Identity Protection
                                                        (Main Mode)
Frame 9203 (190 bytes on wire, 190 bytes captured)
Ethernet II, Src: Vmware_9d:2c:dd (00:50:56:9d:2c:dd),
      Dst: Cisco_80:70:f5 (00:13:c4:80:70:f5)
Internet Protocol, Src: 10.20.129.80 (10.20.129.80),
      Dst: 10.20.131.62 (10.20.131.62)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 0000000000000000
  Next payload: Security Association (1)
  Version: 1.0
  Exchange type: Identity Protection (Main Mode) (2)
  Flags: 0x00
  Message ID: 0x00000000
  Length: 148
  Security Association payload
    Next payload: Vendor ID (13)
    Payload length: 84
    Domain of interpretation: IPSEC (1)
    Situation: IDENTITY (1)
    Proposal payload # 0
      Next payload: NONE (0)
      Payload length: 72
      Proposal number: 0
      Protocol ID: ISAKMP (1)
      SPI Size: 0
      Proposal transforms: 2
      Transform payload # 0
         Next payload: Transform (3)
         Payload length: 32
         Transform number: 0
         Transform ID: KEY_IKE (1)
         Life-Type (11): Seconds (1)
         Life-Duration (12): Duration-Value (28800)
         Encryption-Algorithm (1): 3DES-CBC (5)
         Hash-Algorithm (2): SHA (2)
         Authentication-Method (3): PSK (1)
         Group-Description (4): 1536 bit MODP group (5)
      Transform payload # 1
         Next payload: NONE (0)
         Payload length: 32
         Transform number: 1
         Transform ID: KEY_IKE (1)
```

```
          Life-Type (11): Seconds (1)
          Life-Duration (12): Duration-Value (28800)
          Encryption-Algorithm (1): 3DES-CBC (5)
          Hash-Algorithm (2): SHA (2)
          Authentication-Method (3): PSK (1)
          Group-Description (4): Alternate 1024-bit MODP group (2)
    Vendor ID: 4F456C6A405D72544D42754D
      Next payload: Vendor ID (13)
      Payload length: 16
      Vendor ID: 4F456C6A405D72544D42754D
    Vendor ID: RFC 3706 Detecting Dead IKE Peers (DPD)
      Next payload: NONE (0)
      Payload length: 20
      Vendor ID: RFC 3706 Detecting Dead IKE Peers (DPD)

No.     Time        Source        Destination   Protocol Info
9204    768.395550  10.20.131.62  10.20.129.80  ISAKMP Identity Protection
                                                          (Main Mode)

Frame 9204 (146 bytes on wire, 146 bytes captured)
Ethernet II, Src: Cisco_80:70:f5 (00:13:c4:80:70:f5),
      Dst: Vmware_9d:2c:dd (00:50:56:9d:2c:dd)
Internet Protocol, Src: 10.20.131.62 (10.20.131.62),
      Dst: 10.20.129.80 (10.20.129.80)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 34704CFC8C8DBD09
  Next payload: Security Association (1)
  Version: 1.0
  Exchange type: Identity Protection (Main Mode) (2)
  Flags: 0x00
  Message ID: 0x00000000
  Length: 104
  Security Association payload
    Next payload: Vendor ID (13)
    Payload length: 52
    Domain of interpretation: IPSEC (1)
    Situation: IDENTITY (1)
    Proposal payload # 1
      Next payload: NONE (0)
      Payload length: 40
      Proposal number: 1
      Protocol ID: ISAKMP (1)
      SPI Size: 0
      Proposal transforms: 1
      Transform payload # 1
         Next payload: NONE (0)
         Payload length: 32
         Transform number: 1
         Transform ID: KEY_IKE (1)
         Encryption-Algorithm (1): 3DES-CBC (5)
         Hash-Algorithm (2): SHA (2)
         Group-Description (4): Alternate 1024-bit MODP group (2)
         Authentication-Method (3): PSK (1)
```

```
         Life-Type (11): Seconds (1)
         Life-Duration (12): Duration-Value (28800)
  Vendor ID: Microsoft L2TP/IPSec VPN Client
    Next payload: NONE (0)
    Payload length: 24
    Vendor ID: Microsoft L2TP/IPSec VPN Client

No.     Time        Source        Destination   Protocol Info
9205    768.399599  10.20.129.80  10.20.131.62  ISAKMP Identity Protection
                                                      (Main Mode)

Frame 9205 (222 bytes on wire, 222 bytes captured)
Ethernet II, Src: Vmware_9d:2c:dd (00:50:56:9d:2c:dd),
      Dst: Cisco_80:70:f5 (00:13:c4:80:70:f5)
Internet Protocol, Src: 10.20.129.80 (10.20.129.80),
      Dst: 10.20.131.62 (10.20.131.62)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 34704CFC8C8DBD09
  Next payload: Key Exchange (4)
  Version: 1.0
  Exchange type: Identity Protection (Main Mode) (2)
  Flags: 0x00
  Message ID: 0x00000000
  Length: 180
  Key Exchange payload
    Next payload: Nonce (10)
    Payload length: 132
    Key Exchange Data (128 bytes / 1024 bits)
  Nonce payload
    Next payload: NONE (0)
    Payload length: 20
    Nonce Data


No.     Time        Source        Destination   Protocol Info
9206    768.401192  10.20.131.62  10.20.129.80  ISAKMP Identity Protection
                                                      (Main Mode)
Frame 9206 (298 bytes on wire, 298 bytes captured)
Ethernet II, Src: Cisco_80:70:f5 (00:13:c4:80:70:f5),
      Dst: Vmware_9d:2c:dd (00:50:56:9d:2c:dd)
Internet Protocol, Src: 10.20.131.62 (10.20.131.62),
      Dst: 10.20.129.80 (10.20.129.80)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 34704CFC8C8DBD09
  Next payload: Key Exchange (4)
  Version: 1.0
  Exchange type: Identity Protection (Main Mode) (2)
  Flags: 0x00
  Message ID: 0x00000000
  Length: 256
  Key Exchange payload
```

```
    Next payload: Nonce (10)
    Payload length: 132
    Key Exchange Data (128 bytes / 1024 bits)
  Nonce payload
    Next payload: Vendor ID (13)
    Payload length: 24
    Nonce Data
  Vendor ID: CISCO-UNITY-1.0
    Next payload: Vendor ID (13)
    Payload length: 20
    Vendor ID: CISCO-UNITY-1.0
  Vendor ID: draft-beaulieu-ike-xauth-02.txt
    Next payload: Vendor ID (13)
    Payload length: 12
    Vendor ID: draft-beaulieu-ike-xauth-02.txt
  Vendor ID: C1B7EBE18C8CBD099E89695E2CB16A4A
    Next payload: Vendor ID (13)
    Payload length: 20
    Vendor ID: C1B7EBE18C8CBD099E89695E2CB16A4A
  Vendor ID: CISCO-CONCENTRATOR
    Next payload: NONE (0)
    Payload length: 20
    Vendor ID: CISCO-CONCENTRATOR

No.     Time        Source        Destination   Protocol Info
9207    768.404990  10.20.129.80  10.20.131.62  ISAKMP Identity Protection
                                                          (Main Mode)

Frame 9207 (110 bytes on wire, 110 bytes captured)
Ethernet II, Src: Vmware_9d:2c:dd (00:50:56:9d:2c:dd),
      Dst: Cisco_80:70:f5 (00:13:c4:80:70:f5)
Internet Protocol, Src: 10.20.129.80 (10.20.129.80),
      Dst: 10.20.131.62 (10.20.131.62)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 34704CFC8C8DBD09
  Next payload: Identification (5)
  Version: 1.0
  Exchange type: Identity Protection (Main Mode) (2)
  Flags: 0x01
  Message ID: 0x00000000
  Length: 68
  Encrypted payload (40 bytes)

No.     Time        Source        Destination   Protocol Info
9208    768.405921  10.20.131.62  10.20.129.80  ISAKMP   Identity Protection
                                                          (Main Mode)
Frame 9208 (126 bytes on wire, 126 bytes captured)
Ethernet II, Src: Cisco_80:70:f5 (00:13:c4:80:70:f5),
      Dst: Vmware_9d:2c:dd (00:50:56:9d:2c:dd)
Internet Protocol, Src: 10.20.131.62 (10.20.131.62),
      Dst: 10.20.129.80 (10.20.129.80)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
```

```
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 34704CFC8C8DBD09
  Next payload: Identification (5)
  Version: 1.0
  Exchange type: Identity Protection (Main Mode) (2)
  Flags: 0x01
  Message ID: 0x00000000
  Length: 84
  Encrypted payload (56 bytes)

No.     Time        Source        Destination   Protocol Info
9209    768.409799  10.20.129.80  10.20.131.62  ISAKMP   Quick Mode

Frame 9209 (334 bytes on wire, 334 bytes captured)
Ethernet II, Src: Vmware_9d:2c:dd (00:50:56:9d:2c:dd),
      Dst: Cisco_80:70:f5 (00:13:c4:80:70:f5)
Internet Protocol, Src: 10.20.129.80 (10.20.129.80),
      Dst: 10.20.131.62 (10.20.131.62)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 34704CFC8C8DBD09
  Next payload: Hash (8)
  Version: 1.0
  Exchange type: Quick Mode (32)
  Flags: 0x01
  Message ID: 0x79a63fb1
  Length: 292
  Encrypted payload (264 bytes)

No.     Time        Source        Destination   Protocol Info
9210    768.411797  10.20.131.62  10.20.129.80  ISAKMP   Quick Mode

Frame 9210 (334 bytes on wire, 334 bytes captured)
Ethernet II, Src: Cisco_80:70:f5 (00:13:c4:80:70:f5),
      Dst: Vmware_9d:2c:dd (00:50:56:9d:2c:dd)
Internet Protocol, Src: 10.20.131.62 (10.20.131.62),
      Dst: 10.20.129.80 (10.20.129.80)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 34704CFC8C8DBD09
  Next payload: Hash (8)
  Version: 1.0
  Exchange type: Quick Mode (32)
  Flags: 0x01
  Message ID: 0x79a63fb1
  Length: 292
  Encrypted payload (264 bytes)

No.     Time        Source        Destination   Protocol Info
9211    768.437057  10.20.129.80  10.20.131.62  ISAKMP   Quick Mode

Frame 9211 (94 bytes on wire, 94 bytes captured)
Ethernet II, Src: Vmware_9d:2c:dd (00:50:56:9d:2c:dd),
```

```
      Dst: Cisco_80:70:f5 (00:13:c4:80:70:f5)
Internet Protocol, Src: 10.20.129.80 (10.20.129.80),
      Dst: 10.20.131.62 (10.20.131.62)
User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
Internet Security Association and Key Management Protocol
  Initiator cookie: 92585D2D797E9C52
  Responder cookie: 34704CFC8C8DBD09
  Next payload: Hash (8)
  Version: 1.0
  Exchange type: Quick Mode (32)
  Flags: 0x01
  Message ID: 0x79a63fb1
  Length: 52
  Encrypted payload (24 bytes)
```

Troubleshooting NSX Controller 8

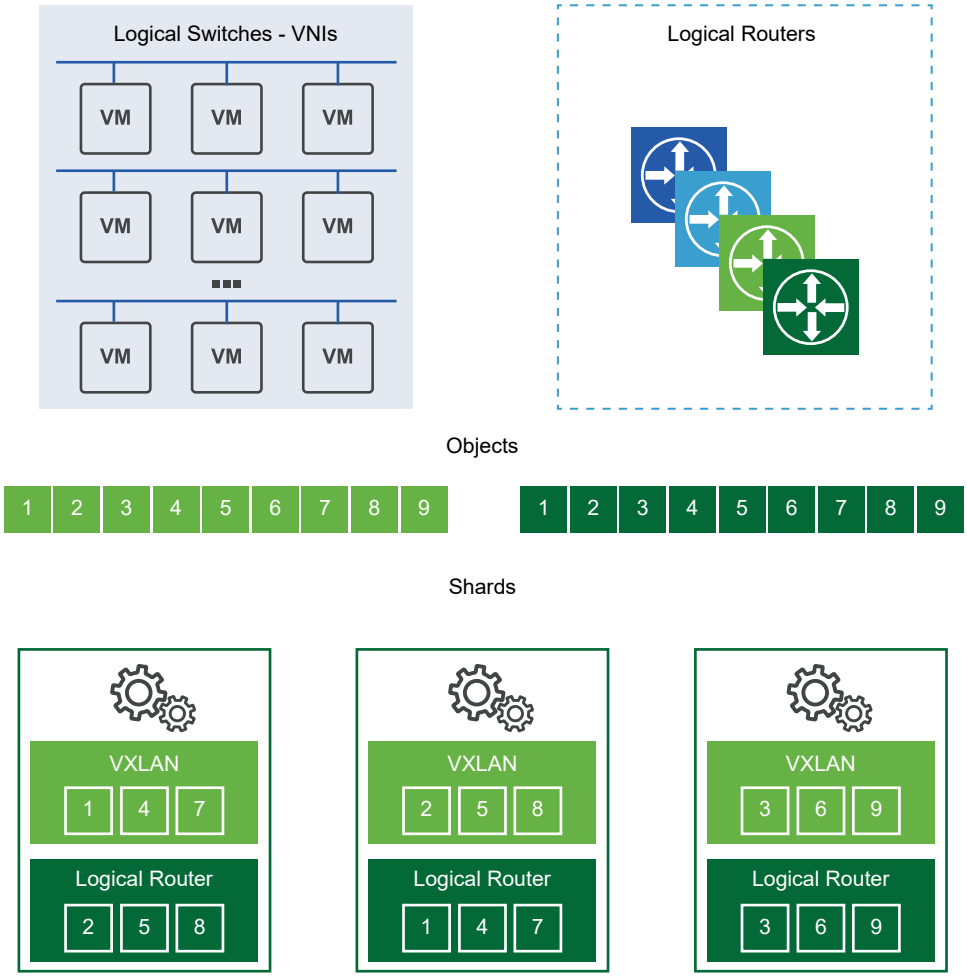
This section provides information on identifying cause for NSX Controller failure and troubleshooting controllers.

This chapter includes the following topics:

Understanding the Controller Cluster Architecture

The NSX Controller cluster represents a scale-out distributed system, where each controller node is assigned a set of roles that define the type of tasks the node can implement. For resiliency and performance, deployments of controller VM should be in three distinct hosts.

Sharding is used to distribute workloads across NSX Controller cluster nodes. Sharding is the action of dividing NSX Controller workloads into different shards so that each NSX Controller instance has an equal portion of the work.



This demonstrates how distinct controller nodes act as master for given entities such as logical switching, logical routing and other services. After a master NSX Controller instance is chosen for a role, that NSX Controller divides the different logical switches and routers among all available NSX Controller instances in a cluster.

Each numbered box on the shard represents shards that the master uses to divide the workloads. The logical switch master divides the logical switches into shards and assigns these shards to different NSX Controller instances. The master for the logical routers also divides the logical routers into shards and assigns these shards to different NSX Controller instances.

These shards are assigned to the different NSX Controller instances in that cluster. The master for a role decides which NSX Controller instances are assigned to which shard. If a request comes in on router shard 3, the shard is told to connect to the third NSX Controller instance. If a request comes in on logical switch shard 2, that request is processed by the second NSX Controller instance.

When one of the NSX Controller instances in a cluster fails, the masters for the roles redistribute the shards to the remaining available clusters. One of the controller nodes is elected as a master for each role. The master is responsible for allocating shards to individual controller nodes, determining when a node has failed, and reallocating the shards to the other nodes. The master also informs the ESXi hosts about the failure of the cluster node.

The election of the master for each role requires a majority vote of all active and inactive nodes in the cluster. This is the primary reason why a controller cluster must always be deployed with an odd number of nodes.

ZooKeeper

ZooKeeper is a client server architecture that is responsible for NSX Controller cluster mechanism. The controller cluster is discovered and created using Zookeeper. When cluster is coming up, it literally means ZooKeeper is coming up between all the nodes. ZooKeeper nodes goes through election process to form the control cluster. There must be a ZooKeeper master node in the cluster. This is done via inter-node election.

When a new controller node is created,NSX Manager propagates the node information to the current cluster, with node IP and ID. As such, each node knows the total number of nodes available for clustering. During ZooKeeper master election, each node casts one vote to elect a master node. The election is triggered again until one node has a majority of the votes. For example, in a three node cluster, the master must have received at least two of the votes.

Note To prevent scenario where a ZooKeeper master cannot be elected, the number of nodes in the cluster MUST be three.

- When the first controller is deployed, it's a special case and the first controller becomes master. As such, when deploying controllers, the first node must complete deployment before any other nodes are added.
- When adding the second controller, it's also a special case, because the number of nodes at this time is even.
- When the third node is added, the cluster reaches a supported stable state.

ZooKeeper can sustain only one failure at a time. This means that if one controller node goes down, it must be recovered before any other failures. Otherwise, there can be problems with the cluster breaking.

Central Control Plane (CCP) Domain Manager

This is the layer above ZooKeeper which provides configuration for ZooKeeper on all nodes to start. Domain manager updates the configuration between all nodes in the cluster, and then makes a remote procedure call for the ZooKeeper process to start.

Domain manager is responsible to start all domains. To join the cluster, CCP domain talks to CCP domain on other machines. The component of CCP domain that helps with cluster initialization is *zk-cluster-bootstrap*.

Controller Relation with Other Components

The controller cluster is responsible for maintaining and providing information about logical switches, logical routers, and VTEPs to the ESXi hosts.

When a logical switch is created, the controller nodes within the cluster determines which node will be *master* or *owner* for that logical switch. The same applies when a logical router is added.

Once ownership is established for a logical switch or logical router, the node sends that ownership to the ESXi hosts that belong to that switch or router's transport zone. The entire election of ownership and propagation of the ownership information to the hosts is called 'sharding'. Note that ownership means that node is responsible for all NSX related operations for that logical switch or logical router. The other nodes will not perform any operation for that logical switch.

Because only one owner must be the source of truth for a logical switch and logical router, any time the controller cluster breaks in such a way that two or more nodes are elected as owner for a logical switch or logical router, each host in the network may have a different information regarding the source of truth for that logical switch or logical router. If this happens, there will be network outage because network control and data plane operations can only have one source of truth.

If a controller node goes down, the remaining nodes in the cluster will rerun sharding to determine ownership of the logical switch and logical routing.

NSX Controller Deployment Issues

NSX Controllers are deployed by NSX Manager in OVA format. Having a controller cluster provides high availability. Deploying controllers requires that NSX Manager, vCenter Server, and ESXi hosts have DNS and NTP configured. A static IP pool must be used to assign IP addresses to each controller.

It is recommended that you implement DRS anti-affinity rules to keep NSX Controllers on separate hosts. You must deploy THREE NSX Controllers.

Common Issues with Controllers

During the deployment of NSX Controllers, the typical issues that can be encountered are as follows:

- Deployment of NSX Controller(s) fails.
- NSX Controller fails to join the cluster.
- Running the `show control-cluster status` command shows the `Majority status` flapping between `Connected to cluster majority` to `Interrupted connection to cluster majority`.
- NSX Dashboard displaying issue with the connectivity status.
    - The `show control-cluster status` command is the recommended command to view whether a controller has joined a control cluster. You need to run this on each controller to find out the overall cluster status.

```
controller # show control-cluster status
Type                Status                                          Since
--------------------------------------------------------------------------------
Join status:        Join complete                                   10/17 18:16:58
Majority status:    Connected to cluster majority                   10/17 18:16:46
Restart status:     This controller can be safely restarted         10/17 18:16:51
Cluster ID:         af2e9dec-19b9-4530-8e68-944188584268
Node UUID:          af2e9dec-19b9-4530-8e68-944188584268
Role                Configured status   Active status
```

```
--------------------------------------------------------------------------------
api_provider        enabled             activated
persistence_server  enabled             activated
switch_manager      enabled             activated
logical_manager     enabled             activated
dht_node            enabled             activated
```

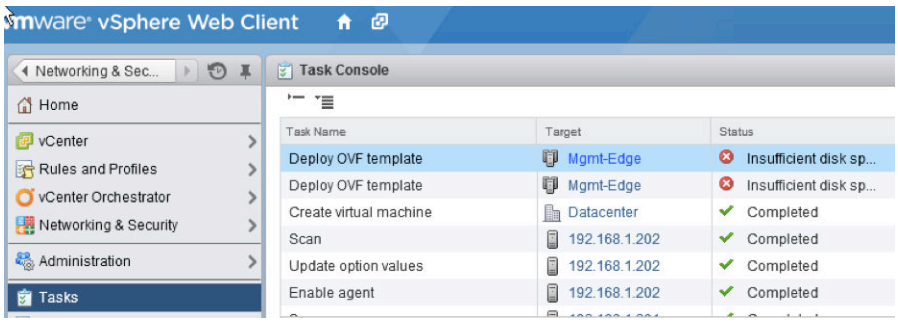
Note When you see controller node is disconnected, do NOT use `join cluster` or `force join` command. This command is not designed to join node to cluster. Doing this, cluster might enter in to a totally uncertain state.

Cluster startup nodes are just a hint to the cluster members on where to look when the members start up. Do not be alarmed if this list contains cluster members no longer in service. This will not impact cluster functionality.

All cluster members should have the same cluster ID. If they do not, then the cluster is in a broken status and you should work with VMware technical support to repair it.

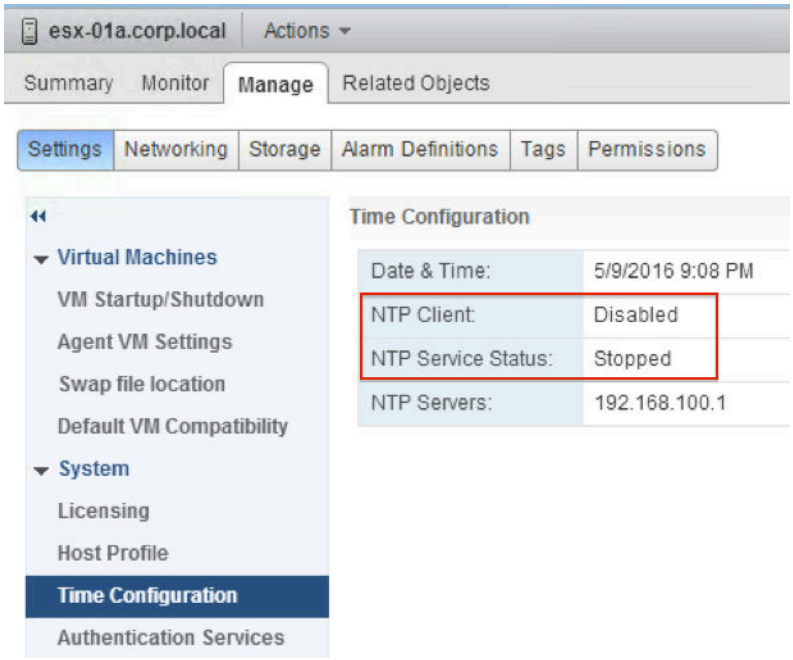
- The `show control-cluster startup-nodes` command was not designed to display all nodes currently in the cluster. Instead, it shows which other controller nodes are used by this node to bootstrap membership into the cluster when the controller process restarts. Accordingly, the command output may show some nodes which are shut down or have otherwise been pruned from the cluster.
- In addition, the `show control-cluster network ipsec status` command allows to inspect the Internet Protocol Security (IPsec) state. If you see that controllers are unable to communicate between themselves for a few minutes to hours, run the `cat /var/log/syslog | egrep "sending DPD request|IKE_SA"` command and see if the log messages indicate absence of traffic. You can also run the `ipsec statusall | egrep "bytes_i|bytes_o"` command and verify that there are no two IPsec tunnels established. Provide the output of these commands and the controller logs when reporting a suspected control cluster issue to your VMware technical support representative.

- IP connectivity issues between the NSX Manager and the NSX controllers. This is generally caused by physical network connectivity issues or a firewall blocking communication.
- Insufficient resources such as storage available on vSphere to host the controllers. Viewing the vCenter events and tasks log during controller deployment can identify such issues.



- A misbehaving "rogue" controller or an upgraded controllers in the **Disconnected** state.

- DNS on ESXi hosts and NSX Manager have not been configured properly.
- NTP on ESXi hosts and NSX Manager are not in sync.



- When newly connected VMs have no network access, this is likely caused by a control-plane issue. Check the controller status.

  Also try running the `esxcli network vswitch dvs vmware vxlan network list --vds-name <name>` command on ESXi hosts to check the control-plane status. Note that the Controller connection is down.

```
/etc/vmware/netcpa # esxcli network vswitch dvs vmware vxlan network list --vds-name Compute_VDS
VXLAN ID  Multicast IP               Control Plane                       Controller Connection
ARP Entry Count  MTEP Count
--------  -------------------------  ----------------------------------  ---------------------
--------------  ----------
    5000  N/A (headend replication)  Enabled (multicast proxy,ARP proxy)  192.168.110.203 (down)
               0           0
```

- Running the `show log manager follow` NSX Manager CLI command can identify any other reasons for a failure to deploy controllers.

```
2014-02-26 10:09:44.931 GMT  INFO taskScheduler-25 VcConnection$VimClient:1219 - Create stub for com.vmware.vim.bindin
28c5157-abf3-718e-88c5-42209f389211
2014-02-26 10:09:44.932 GMT DEBUG VcEventsReaderThread VcEventsReader$VcEventsReaderThread:301 - got prop collector up
ctReference: type = PropertyFilter, value = session[d46b86a2-7a10-c17e-6ebe-8ab252ee4efd]527420f2-bdd7-529b-8ab6-17d16
6E3-4A64-96D7-5833C287588F
2014-02-26 10:09:44.937 GMT ERROR taskScheduler-25 VCUtils:184 - Error while waiting for property collector updates.
com.vmware.vim.binding.vim.fault.NoDiskSpace:
datastore = datastore1 (1)
inherited from com.vmware.vim.binding.vim.fault.FileFault:
file = [datastore1 (1)] NSX_Controller_1c3dd18d-0cd3-4d7d-896b-51247176ae77/NSX_Controller_1c3dd18d-0cd3-4d7d-896b-512
inherited from com.vmware.vim.binding.vim.fault.VimFault:

inherited from com.vmware.vim.binding.vim.fault.NoDiskSpace: Insufficient disk space on datastore 'datastore1 (1)'.
```

Host Connectivity Issues

Check for host connectivity errors using the following commands. Run these commands on each of the controller nodes.

- Check for any abnormal error statistics using the `show log cloudnet/cloudnet_java-vnet-controller*.log filtered-by host_IP` command.
- Verify the logical switch/router message statistics or high message rate using the following commands:
  - `show control-cluster core stats`: overall stats
  - `show control-cluster core stats-sample`: latest stats samples
  - `show control-cluster core connection-stats ip`: per connection stats
  - `show control-cluster logical-switches stats`
  - `show control-cluster logical-routers stats`
  - `show control-cluster logical-switches stats-sample`
  - `show control-cluster logical-routers stats-sample`
  - `show control-cluster logical-switches vni-stats vni`
  - `show control-cluster logical-switches vni-stats-sample vni`
  - `show control-cluster logical-switches connection-stats ip`
  - `show control-cluster logical-routers connection-stats ip`
- You can use the `show host hostID health-status` command to check the health status of hosts in your prepared clusters. For controller troubleshooting, the following health checks are supported:
  - Check whether the `net-config-by-vsm.xml` is synchronized to controller list.
  - Check if there is a socket connection to controller.
  - Check whether the VXLAN Network Identifier (VNI) is created and whether the configuration is correct.
  - Check VNI connects to master controllers (if control plane is enabled).

Installation and Deployment Issues

- Verify that there are at least three controller nodes deployed in a cluster. VMware recommends to leverage the native vSphere anti-affinity rules to avoid deploying more than one controller node on the same ESXi host.
- Verify that all NSX Controllers display a `Connected` status. If any of the controller nodes display a `Disconnected` status, ensure that the following information is consistent by running the `show control-cluster status` command on all controller nodes:

Type	Status
Join status	Join complete
Majority status	Connected to cluster majority
Cluster ID	Same information on all controller nodes

- Ensure that all roles are consistent on all controller nodes:

Role	Configured status	Active status
`api_provider`	enabled	activated
`persistence_server`	enabled	activated
`switch_manager`	enabled	activated
`logical_manager`	enabled	activated
`directory_server`	enabled	activated

- Verify that `vnet-controller` process is running. Run the `show process` command on all controller nodes and ensure that `java-dir-server` service is running.
- Verify the cluster history and ensure there is no sign of host connection flapping, or VNI join failures and abnormal cluster membership change. To verify this, run the `show control-cluster history` command. The commands also shows if the node is frequently restarted. Verify that there are not many log files with zero (0) size and with different process IDs.
- Verify that VXLAN Network Identifier (VNI) is configured. For more information, see the VXLAN Preparation Steps section of the VMware VXLAN Deployment Guide.
- Verify that SSL is enabled on the controller cluster. Run the `show log cloudnet/cloudnet_java-vnet-controller*.log filtered-by sslEnabled` command on each of the controller nodes.

Troubleshooting Disk Latency

You can view disk latency alerts from the **Management** tab. NSX Controllers must operate on disks with low latency.

View Disk Latency Alerts

Disk latency alerts monitors and reports disk availability or latency issues. You can view disk latency details for each NSX Controller. The read latency and write latency calculations are inputted into a 5-second (by default) moving average, which in turn is used to trigger an alert upon breaching the latency limit. The alert is turned off after the average comes down to the low watermark. By default, the high watermark is set to 200 ms, and the low watermark is set to 100 ms. High latencies impacts the operation of the distributed clustering applications on each controller node.

To view the disk latency alerts for NSX Controller, perform the following procedure:

Prerequisites

Latency limit is reached.

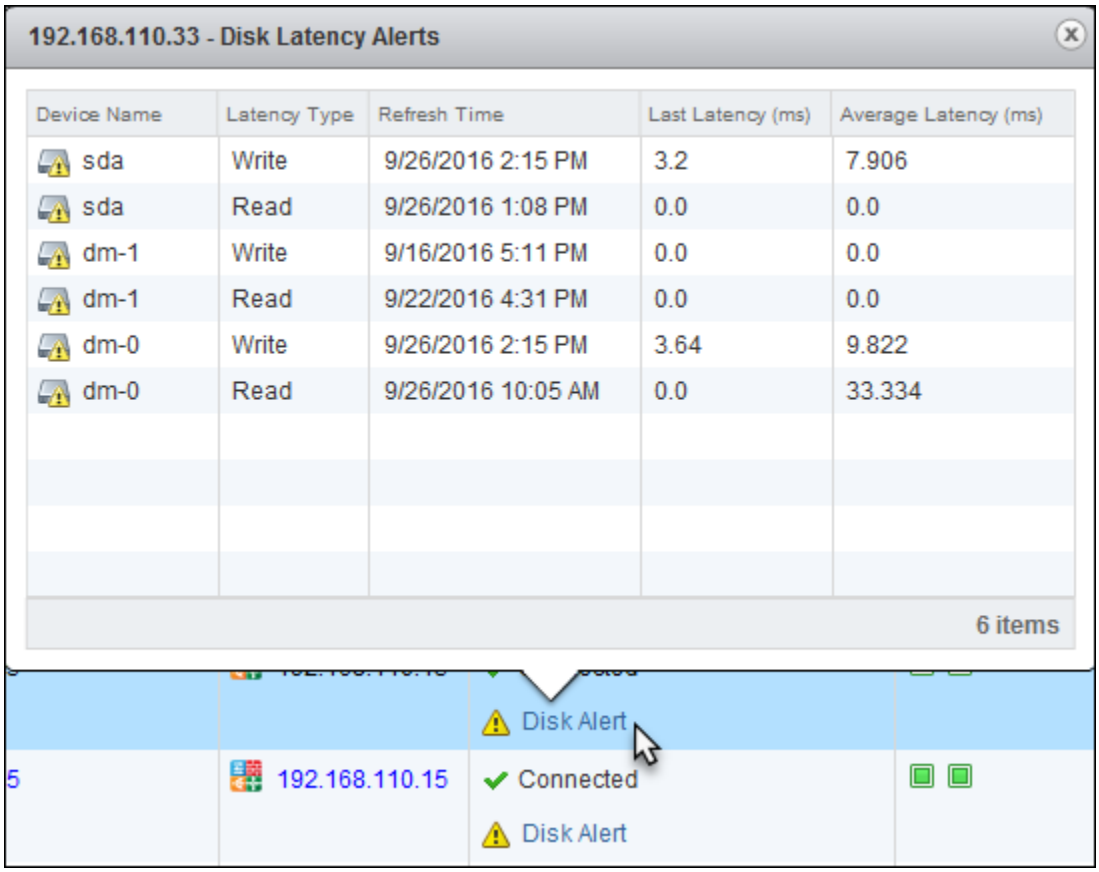
Procedure

1 Log in to the vSphere Web Client.

2 Click **Networking & Security**, and then click **Installation**.

3 Under **Management**, go to the required controller, and click the **Disk Alert** link.

The Disk Latency Alerts window appears.



Results

You can view the latency details for the selected controller. The alert logs are stored for seven days in the `cloudnet/run/iostat/iostat_alert.log` file. You can use the `show log cloudnet/run/iostat/iostat_alert.log` command to display the log file.

What to do next

For more troubleshooting information on disk latency, refer to Disk Latency Issues.

For more information about log messages, refer to *NSX Logging and System Events*.

Disk Latency Issues

The controllers must operate on disks with low latency. The cluster requires disk storage system for each node to have a peak write latency of less than 300 ms, and a mean write latency of less than 100 ms.

Problem

- A deployed NSX Controller is disconnected from a controller cluster.
- Unable to gather any controller logs as disk partition is full.
- If the storage system does not meet these requirements, the cluster can become unstable and cause system downtime.
- TCP listeners applicable to a functioning NSX Controller, no longer appear in the output of the `show network connections of-type tcp` command.
- The disconnected controller attempts to join the cluster using an all-zeroes UUID, which is not valid.
- The show control-cluster history command displays a message similar to:

  `INFO.20150530-000550.1774:D0530 13:25:29.452639 1983 zookeeper_client.cc:774] Zookeeper client disconnected!`

- Running the `show log cloudnet/cloudnet_java-zookeeper*.log` command in the NSX Controller console contains entries similar to:

```
cloudnet_java-zookeeper.20150530-000550.1806.log-2015-05-30
     13:25:07,382 47956539 [SyncThread:1] WARN
     org.apache.zookeeper.server.persistence.FileTxnLog - fsync-ing the write ahead
     log in SyncThread:1 took 3219ms which will adversely effect operation latency.
     See the ZooKeeper troubleshooting guide
```

- The NSX Controller logs contains entries similar to:

```
D0525 13:46:07.185200 31975
     rpc-broker.cc:369] Registering address resolution for: 20.5.1.11:7777
D0525 13:46:07.185246 31975
     rpc-tcp.cc:548] Handshake complete, both peers support the same
     protocol
D0525 13:46:07.197654 31975
     rpc-tcp.cc:1048] Rejecting a connection from peer
     20.5.1.11:42195/ef447643-f05d-4862-be2c-35630df39060, cluster
     9f7ea8ff-ab80-4c0c-825e-628e834aa8a5, which doesn't match our cluster
     (00000000-0000-0000-0000-000000000000)
D0525 13:46:07.222869 31975
     rpc-tcp.cc:1048] Rejecting a connection from peer
     20.5.1.11:42195/ef447643-f05d-4862-be2c-35630df39060, cluster
     9f7ea8ff-ab80-4c0c-825e-628e834aa8a5, which doesn't match our cluster
     (00000000-0000-0000-0000-000000000000)
```

Cause

This issue occurs due to slow disk performance, which adversely impacts the NSX Controller cluster.

- Check for slow disks by looking for *fsync* messages in the `/var/log/cloudnet/cloudnet_java-zookeeper log` file. If *fsync* takes more than one second, Zookeeper displays a *fsync* warning message, and it is a good indication that the disk is too slow. VMware recommends dedicating a Logical Unit Number (LUN) specifically for the control-cluster and/or moving the storage array closer to the control-cluster in terms of latencies.
- You can view the read latency and write latency calculations that are inputted into a 5-second (by default) moving average, which in turn is used to trigger an alert upon breaching the latency limit. The alert is turned off after the average comes down to the low watermark. By default, the high watermark is set to 200 ms, and the low watermark is set to 100 ms. You can use the `show disk-latency-alert config` command. The output is displayed as follows:

```
enabled=True   low-wm=51      high-wm=150
nsx-controller # set disk-latency-alert enabled yes
nsx-controller # set disk-latency-alert low-wm 100
nsx-controller # set disk-latency-alert high-wm 200
```

- Use the `GET /api/2.0/vdn/controller/<controller-id>/systemStats` REST API to fetch latency alert status of the controller nodes.
- Use the `GET /api/2.0/vdn/controller` REST API to indicate whether a disk latency alert is detected on a controller node.

Solution

1. Deploy NSX Controller on low-latency disks.
2. Each controller should use its own disk storage server. Do not share same disk storage server between two controllers.

What to do next

For more information on how to view alerts, refer to View Disk Latency Alerts .

NSX Controller Cluster Failures

When one of the NSX Controller nodes in the cluster fails, you still have two controllers that are working. The cluster majority is maintained, and the control plane continues to function.

Problem

NSX Controller cluster has failed.

Solution

1. Log in to the vSphere Web Client.
2. From **Networking & Security**, click **Installation > Management**.

3 In the NSX Controller nodes section, observe the Peers column. If the Peers column shows green boxes, it represents no error in the peer controller connectivity in the cluster. A red box indicates an error with a peer. Click the box to view details.

4 If the Peers column displays a problem in the controller cluster, log in to each NSX Controller CLI to perform a detailed diagnosis. Run the show `control-cluster status` command to diagnose the state of each controller. All controllers in the cluster must have the same cluster UUID, however cluster UUID might not be same as the UUID of the master controller. You can find information about deployment issues as described in NSX Controller Deployment Issues .

5 You can try the following steps to resolve the issue before redeploying the controller node or the controller cluster:

   a Check that the controller is powered on.

   b Try to ping to and from the affected controller to other nodes and manager to check network paths. If you find any network issues, address them as described in NSX Controller Deployment Issues .

   c Check the Internet Protocol Security (IPSec) status using the following CLI commands.

      - Verify if IPSec is enabled using the `show control-cluster network ipsec status` command.
      - Verify the status of the IPSec tunnels using the `show control-cluster network ipsec tunnels` command.

      You can also use the IPSec status information to open a ticket with the VMware technical support.

   d If the issue is not a network issue, you can choose whether to reboot or redeploy.

   If you want to reboot a node, ensure that only one controller is rebooted at a time. However, if the controller cluster is in a state where more than one controller node has failed, reboot all of them at the same time. When you are rebooting a node from a healthy cluster, always confirm that the cluster is reformed properly afterwards, and then confirm that the cluster resharding is done properly.

6 If you decide to redeploy controllers, use one of the following two approaches:

   - Approach 1: Delete the broken controller node and redeploy a new controller node.
   - Approach 2: Delete the controller cluster and redeploy a new controller cluster.

     VMware recommends the second approach.

What to do next

Choose any one approach:

- Approach 1: Delete Broken Controller and Redeploy New Controller
- Approach 2: Redeploy NSX Controller Cluster

Approach 1: Delete Broken Controller and Redeploy New Controller

You can first try resolving the issue without redeploying a new NSX Controller cluster. In this approach, you first delete the broken NSX Controller node, and then deploy a new NSX Controller node.

Procedure

1 Delete an NSX Controller

You can delete an NSX Controller forcefully or gracefully. Graceful removal procedure checks for the following conditions before removing the node:

2 Redeploy an NSX Controller

After deleting the broken controller node, deploy a new controller node.

Delete an NSX Controller

You can delete an NSX Controller forcefully or gracefully. Graceful removal procedure checks for the following conditions before removing the node:

- There is no current NSX Controller node upgrade operation.
- The controller cluster is healthy, and a controller cluster API request can be processed.
- The host state, as obtained from the vCenter Server inventory, shows connected and powered on.
- This is not the last controller node.

Forceful removal procedure does not check the above mentioned conditions before removing the controller node.

- Things to remember while deleting controllers:
  - Do not attempt to delete the controller VM before deleting it through the vSphere Web Client UI or API. When the UI is not usable, use the `DELETE /2.0/vdn/controller/{controllerId}` API to delete the controller.
  - After deletion of a node, ensure that the existing cluster stays stable.
  - When deleting all the nodes in a cluster, the last remaining node must be deleted using the **Forcefully remove the controller** option. Always verify that the controller VM is deleted successfully. If not, manually power down the VM and delete the controller VM using the UI.
  - If the delete operation fails, it means that the VM could not get deleted. In such case, invoke controller delete through UI with the **Forcefully remove the controller** option. For API, set the `forceRemoval` parameter to *true*. After forceful removal, manually power down the VM and delete the controller VM using the UI.
  - Since a multi-node cluster can only sustain one failure, deletion counts as a failure. The deleted node must be redeployed before another failure occurs.

- For Cross-vCenter NSX environment:
  - Deleting the controller VM or powering it off directly in vCenter Server is not a supported operation. The **Status** column displays **Out of sync** status.
  - If controller deletion succeeds only partially, and an entry is left behind in the NSX Manager database in a Cross-vCenter NSX environment, use the `DELETE api/2.0/vdn/controller/external` API.
  - If the controller was imported through the NSX Manager API, use the `removeExternalControllerReference` API with the `forceRemoval` option.
  - When deleting a controller, NSX requests to delete a controller VM via vCenter Server using the Managed Object ID (MOID) of the VM. If vCenter Server cannot find VM by its MOID, NSX reports failure for the controller delete request and aborts the operation.

    If the **Forcefully Delete** option is selected, NSX do not abort the controller delete operation and will clear the controller's information. NSX also update all the hosts to no longer trust the deleted controller. However, if the controller VM is still active and running with a different MOID, it still has credentials to participate as a member of the controller cluster. Under this scenario, any logical switch or router that is assigned to this controller node will not function properly because the ESXi hosts no longer trust the deleted controller.

To delete the NSX Controller, perform the following procedure:

Procedure

1. Log in to the vSphere Web Client.
2. Click **Networking & Security**, and then click **Installation**.
3. Under **Management**, select the controller that you want to delete.
4. Click the **Delete (x)** icon.
5. Select either **Delete** or **Forcefully Delete**.
   - When you select the **Forcefully Delete** option, the controller gets deleted forcefully and not gracefully. This option ignores any failures and clears the data from database. You should verify that any possible failures are taken care of manually. You must confirm that the controller VM is successfully deleted. If not, you must delete it through vCenter Server.

   Note If you are deleting the last controller in the cluster, you must select the **Forcefully Delete** option to remove the last controller node. When there are no controllers in the system, the hosts are operating in what is called "headless" mode. New VMs or vMotioned VMs will have networking issues until new controllers are deployed and the synchronization is completed.

   - If you do not select this , the controller gets deleted gracefully.
6. Click **Yes**. Graceful controller deletion uses the following sequence:
   a. Power off the node.
   b. Check the cluster health.

c If the cluster is not healthy, power on the controller, and fail the removal request.

d If the cluster is healthy, remove the controller VM, and release the IP address of the node.

e Remove the controller VM's identity from the cluster.

The selected controller is deleted.

7 Re-synchronize the controller state by clicking **Actions > Update Controller State**.

Redeploy an NSX Controller

After deleting the broken controller node, deploy a new controller node.

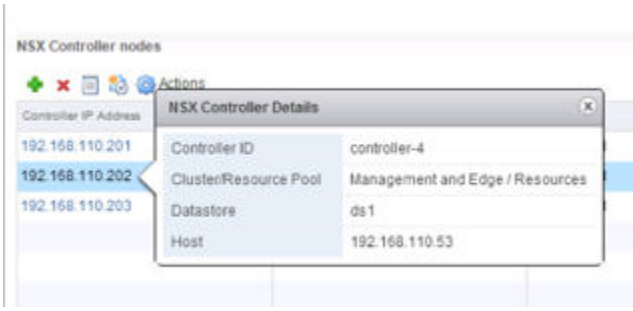
Procedure

1 Log in to the vSphere Web Client.

2 From **Networking & Security**, click **Installation > Management**.

3 In the **NSX Controller nodes** section, click the affected controller. Take screenshots or note down the configuration information in the **NSX Controller Details** screen for future reference.

For example:



4 Deploy a new NSX Controller node by clicking the **Add Node (+)** icon.

5 In the Add Controller dialog box, select the data center on which you are adding the nodes, and configure the controller settings.

a Select the appropriate cluster.

b Select a Host in the cluster and storage.

c Select the distributed port-group.

d Select the IP pool from which IP addresses are to be assigned to the node.

e Click **OK**, wait for the installation to complete, and ensure that the node has a **Normal** status.

For detailed information about adding a controller node, see "Deploy NSX Controller Cluster" in the *NSX Installation Guide*.

6 Resynchronize the controller state by clicking **Actions > Update Controller State**.

Update Controller State pushes the current VXLAN and Distributed Logical Router configuration (including Universal Objects in a Cross-vCenter NSX deployment) from NSX Manager to the controller cluster.

Approach 2: Redeploy NSX Controller Cluster

In this approach, delete all the three controller nodes, and add new controller nodes to maintain a fully functional three-node cluster.

VMware recommends deleting the NSX Controller cluster when any of the following conditions are true:

- One or more controller nodes face catastrophic or unrecoverable errors.
- Controller virtual machines are inaccessible and cannot be fixed.

In such cases, preferably delete all the controller nodes, even when some of the controller nodes seem healthy.

Redeploy a new controller cluster, and then update the controller state mechanism on the NSX Manager. Updating the controller state causes VXLAN to be resynchronized and the distributed logical routers to be redeployed.

Procedure

1. Log in to the vSphere Web Client.
2. Navigate to **Networking & Security > Installation > Management**.
3. In the **NSX Controller Nodes** section, delete all the three controller nodes. Select one node at a time and click the **Delete** ( ) icon.

   When no controllers exist in the system, the hosts operate in "headless" mode. New virtual machines or migrated virtual machines will have networking issues until new controllers are deployed and the synchronization is completed.
4. Deploy three new controller nodes to create a fully functional NSX Controller cluster.

   For detailed information about adding a controller cluster, see "Deploy NSX Controller Cluster" in the *NSX Installation Guide*.
5. Resynchronize the controller state by clicking **Actions > Update Controller State**.

Phantom Controller

A phantom controller can be a live controller virtual machine (VM) or non-existent VM that can be participating or not participating in the cluster. NSX Manager synchronises the list of all VMs from the vCenter Server inventory. A phantom controller is created when the vCenter Server or host deletes a controller VM without a request from NSX Manager, or when vCenter Server inventory changes the reference MOID of the controller VMs.

When controller is created from NSX, the configuration information is stored inside the NSX Manager. NSX Manager deploys the new controller VM through the vCenter Server.

NSX administrator provides configuration, including IP address pool to the NSX Manager to create a controller. NSX Manager removes an IP address from the pool, and pushes that IP with the rest of the controller configuration as a VM creation request to the vCenter Server. NSX Manager waits for vCenter Server to confirm the status of the request.

- `The controller creation process was successful`: If the controller VM is created successfully, vCenter Server starts the controller VM. NSX Manager stores the Managed Object ID (MOID) of the VM with the rest of the controller's configuration information. The MOID (or MO-REF) is a unique identifier that vCenter assigns to every object in its inventory. vCenter Server also use this MOID to track the VM if it remains part of the vCenter Server inventory.
- `The controller creation process was not successful`: If the IP and network connection configurations were incorrect, then NSX Manager might not be able to contact vCenter Server. NSX Manager waits for a preset amount of time to create a single node controller cluster (for the first one) or new controller to join the active cluster. If timer expires, NSX Manager requests vCenter Server to delete the VM. The IP address is returned back to the pool and NSX declares controller creation failure.

How Phantom Controller Gets Created

When NSX Manager requests to delete a controller, vCenter Server finds the controller VM using the MOID for deletion.

However, if any vCenter activities result in removal of the controller VM from the vCenter Server inventory, vCenter removes the MOID from its database. Note that the controller VM can still be alive and active on the NSX Manager even after getting removed from the vCenter inventory. But for the vCenter Server, controller VM no longer exists. Even though vCenter Server has removed the VM from its inventory, the VM may not be deleted. If the VM is still active, then it is still participating or attempting to participate in the NSX controller cluster.

Following are the most common example of how phantom controller gets created:

- The vCenter Server administrator removes the host that contains the controller VM from the inventory. Later adds the host back. When the host is removed, vCenter Server delete all the MOIDs associated with the host and the VMs within it. When the host is added back later, vCenter Server assigns brand new MOID to the host and the VMs. For the NSX users, the host and VM are still the same, but from the vCenter Server's perspective, the hosts and VMs are brand new objects. However, for all practical purposes, the hosts and VMs are still the same. The applications that run within the host and VMs do not change.
- The vCenter Server administrator deletes the controller VM through vCenter Server or using Host Management. The deletion was not initiated by NSX Manager.
- *Delete* in this case also includes any host/storage failures that result in the loss of the VM. In this case, the VM is lost to vCenter Server and also lost to the cluster and NSX Manager. But because the deletion was not initiated by NSX Manager, both NSX Manager and the controller cluster thinks that the controller is still valid. The controller status returned to the NSX Manager indicates that this controller node is down and not part of the cluster and displayed on the UI. NSX Manager also have logs indicating that the controller is no longer reachable.

What to Do When You See Phantom Controller

1 Synchronize controllers as described in NSX Controller Is Disconnected.

2 See the log entries. For cases where the controller VM got deleted accidentally or got corrupted, you must use the **Forcefully Delete** option to clear the entry from the NSX Manager database. For details, refer to Delete an NSX Controller.

3 After deleting the controller, confirm that:

- The controller VM is actually deleted.
- The `show controller-cluster startup-nodes` command shows only valid controllers.
- The syslog entries for the NSX Manager no longer shows an extra controller.

From NSX 6.2.7 or later, NSX Manager verifies with the vCenter inventory to ensure that the controller VM still exist in the inventory based on the original MOID. If NSX Manager cannot find controller VM in the inventory, NSX Manager searches the VM using the VM's instance UUID. The instance UUID is stored within the VM, so it does not change even when the VM is added back to the vCenter inventory. If NSX Manager is able to find the VM with the instance UUID, NSX Manager updates its database with the new MOID.

However, if you clone the controller VM, the cloned VM has same properties as the original VM along with a new instance UUID. NSX Manager cannot detect MOID for the cloned VM.

Log Entries for Phantom Controller

Following error level log entry is seen when a phantom controller is detected:

- `2017-07-31 22:15:05.844 UTC ERROR NVPStatusCheck ControllerServiceImpl:2146 - Controller <#> does not exist, might be deleted already. Skip saving its connectivity info.`
- `2017-07-31 22:15:05.769 UTC ERROR NVPStatusCheck ControllerServiceImpl:2580 - the node is created by this NSX Manager <#>, but database has no record and delete might be in progress.`

NSX Controller Is Disconnected

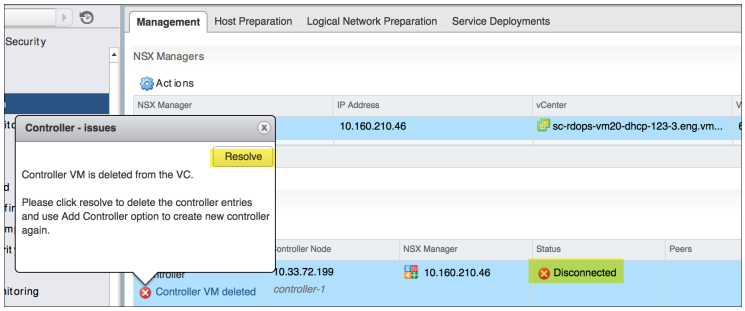
If the NSX Controller VM was powered off from vCenter Server or a controller VM was deleted from the vCenter Server, the **Status** column of the **Installation** > **Management** page displays **Out of sync** status.

Prerequisites

Controller VM powered off or controller VM deleted from the vCenter Server.

Procedure

1 In the vSphere Web Client, navigate to **Networking & Security > Installation** > **Management**.



2 Click the **Error** link to see the detailed reason for this out of sync state.

3 Click the **Resolve** button to resolve the issue.

Results

If the controller VM is powered off, Management Plane triggers a `power on` command for the controller.

If the controller VM is deleted, the entries of the controller are deleted from the Management Plane and Management Plane communicates the controller deletion to the Central Control Plane.

What to do next

Create a new controller using the **Add Node** option. For details, refer to the *NSX Administration Guide*.

Control Plane Agent (netcpa) Issues

On NSX for vSphere, control plane (netcpa) works as a local agent daemon, communicating with NSX Manager and with the controller cluster. **Communication Channel Health** feature is a proactive health check which periodically reports the central control plane to local control plane status to NSX Manager and is displayed at the NSX Manager UI. This report also serves as a heartbeat to detect the operational status of the NSX Manager to ESXi host netcpa channel. It provides error details during communication faults, generates an event when a channel goes into a wrong status, and also generates heartbeat messages from NSX Manager to hosts.

Problem

Connectivity issues between control plane agent and controller.

Cause

If there is any missing connection, then control plane agent may not be working properly.

Solution

1 Validate the connection status when the channel goes into a wrong state using the following command:

```
GET https://<NSX_Manager_IP>/api/2.0/vdn/inventory/host/{hostId}/connection/status
```

Following is the example of the return value:

```
<?xml version="1.0" encoding="UTF-8"?>
<hostConnStatus>
<hostName>10.161.246.20</hostName>
<hostId>host-21</hostId>
<nsxMgrToFirewallAgentConn>UP</nsxMgrToFirewallAgentConn>
<nsxMgrToControlPlaneAgentConn>UP</nsxMgrToControlPlaneAgentConn>
<hostToControllerConn>DOWN</hostToControllerConn>
<fullSyncCount>-1</fullSyncCount>
<hostToControllerConnectionErrors>
<hostToControllerConnectionError>
<controllerIp>10.160.203.236</controllerIp>
<errorCode>1255604</errorCode>
<errorMessage>Connection Refused</errorMessage>
</hostToControllerConnectionError>
<hostToControllerConnectionError>
<controllerIp>10.160.203.237</controllerIp>
<errorCode>1255603</errorCode>
<errorMessage>SSL Handshake Failure</errorMessage>
</hostToControllerConnectionError>
</hostToControllerConnectionErrors>
</hostConnStatus>
```

The following error codes are supported:

```
1255602: Incomplete Controller Certificate
1255603: SSL Handshake Failure
1255604: Connection Refused
1255605: Keep-alive Timeout
1255606: SSL Exception
1255607: Bad Message
1255620: Unknown Error
```

2 Determine the reason for the control plane agent being down as follows:

a Check the control plane agent status on hosts by running the `/etc/init.d/netcpad status` command on ESXi hosts.

```
[root@esx-01a:~] /etc/init.d/netcpad status
netCP agent service is running
```

b Check the control plane agent configurations using the `more /etc/vmware/netcpa/config-by-vsm.xml` command. The IP addresses of the NSX Controllers should be listed.

```
[root@esx-01a:~] more /etc/vmware/netcpa/config-by-vsm.xml
<config>
  <connectionList>
    <connection id="0000">
      <port>1234</port>
      <server>192.168.110.31</server>
      <sslEnabled>true</sslEnabled>
```

```
      <thumbprint>A5:C6:A2:B2:57:97:36:F0:7C:13:DB:64:9B:86:E6:EF:1A:7E:5C:36</thumbprint>
    </connection>
    <connection id="0001">
      <port>1234</port>
      <server>192.168.110.32</server>
      <sslEnabled>true</sslEnabled>
      <thumbprint>12:E0:25:B2:E0:35:D7:84:90:71:CF:C7:53:97:FD:96:EE:ED:7C:DD</thumbprint>
    </connection>
    <connection id="0002">
      <port>1234</port>
      <server>192.168.110.33</server>
      <sslEnabled>true</sslEnabled>
      <thumbprint>BD:DB:BA:B0:DC:61:AD:94:C6:0F:7E:F5:80:19:44:51:BA:90:2C:8D</thumbprint>
    </connection>
  </connectionList>
...
```

3 Validate connections to the controllers from the control plane agent using the following command. The output is one connection for each controller.

```
>[root@esx-01a:~] esxcli network ip connection list | grep 1234
tcp     0   0  192.168.110.51:16594     192.168.110.31:1234   ESTABLISHED     36754  newreno
netcpa-worker
tcp     0   0  192.168.110.51:46917     192.168.110.33:1234   ESTABLISHED     36754  newreno
netcpa-worker
tcp     0   0  192.168.110.51:47891     192.168.110.32:1234   ESTABLISHED     36752  newreno
netcpa-worker
```

4 Validate the connections to the controllers from the control plane agent to show `CLOSED` or `CLOSE_WAIT` status by running the following command:

```
esxcli network ip
        connection list |grep "1234.*netcpa*" | egrep "CLOSED|CLOSE_WAIT"
```

5 If the control plane agent has been down for a significantly long time, the connections may not be present at all. To validate this, run the following command. The output is one connection for each controller.

```
esxcli network ip
        connection list |grep "1234.*netcpa*" |grep ESTABLISHED
```

6 Control Plane Agent (netcpa) auto-recovery mechanism: The automatic control plane agent monitoring process detects the control plane agent in wrong status. When the control plane agent is in a wrong status, it stops responding and then automatically tries to recover.

   a When the control plane agent stops responding, live core file is generated. You can find the core file as follows:

```
ls /var/core
 netcpa-worker-zdump.000
```

   b Syslog error is reported in the *vmkwarning.log* file .

```
cat /var/run/log/vmkwarning.log | grep NETCPA
2017-08-11T06:32:17.994Z cpu1:1000044539)ALERT: Critical - NETCPA is hanged
Taking live-dump & restarting netcpa process!
```

Note If the control plane agent monitor experiences a temporary failure due to a delayed response to the status check, a warning message similar to the following may be reported in the VMKernel logs.

```
Warning - NETCPA getting netcpa status failed!
```

You can ignore this warning.

7 If the issue is not recovered automatically, restart the control plane agent as follows:

   a Log in as root to the ESXi host through SSH or through the console.

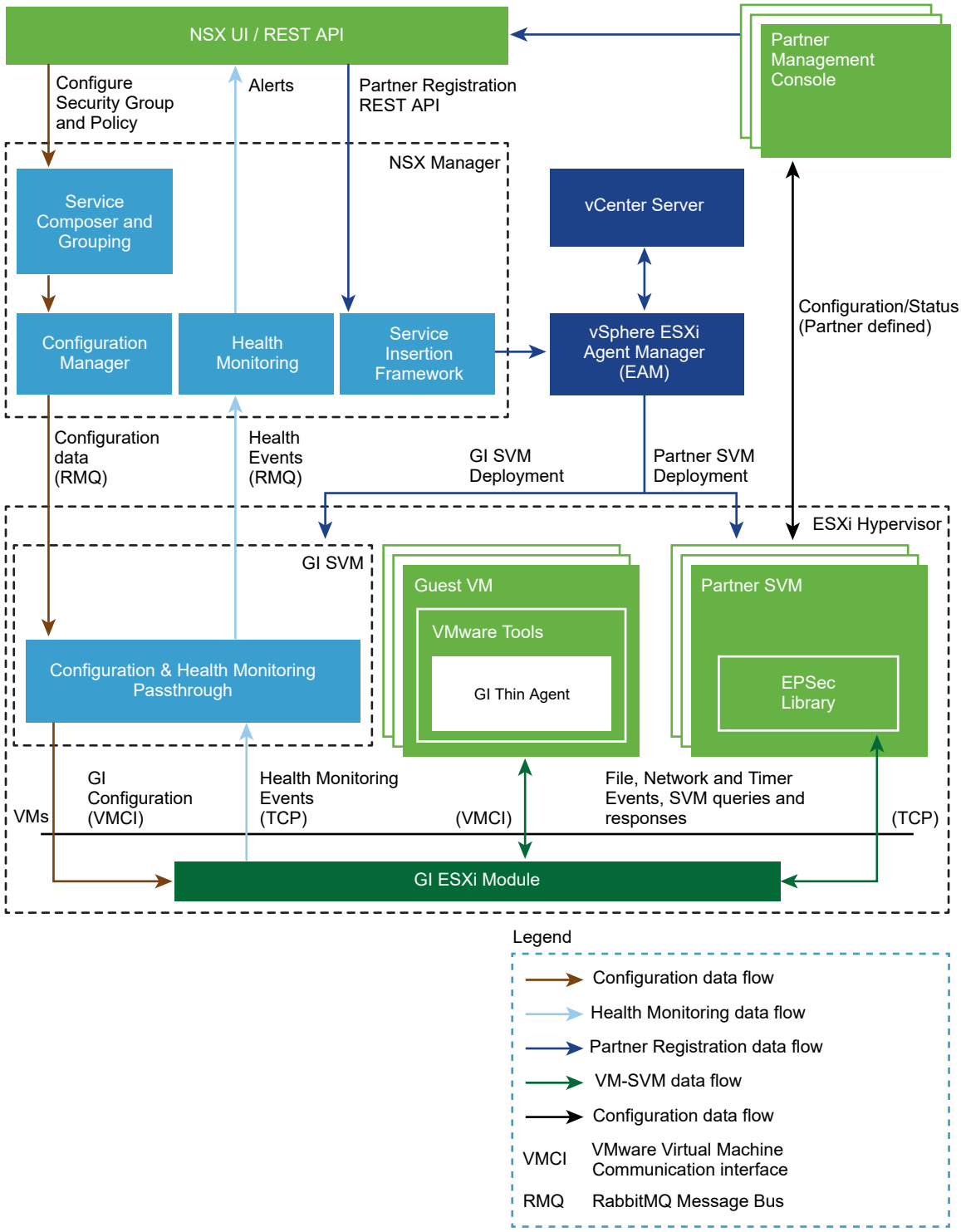
   b Run the `/etc/init.d/netcpad restart` command to restart the control plane agent on the ESXi host.

Troubleshooting Guest Introspection

9

This chapter includes the following topics:

Guest Introspection Architecture



Guest Introspection Logs

There are several different logs you can capture to use while troubleshooting Guest Introspection.

ESX GI Module (MUX) Logs

If virtual machines on an ESXi host are not working with Guest Introspection, or if there are alarms on a host regarding communication to the SVA, then it could be a problem with the ESX GI Module on the ESXi host.

Log Path and Sample Message

MUX Log path
/var/log/syslog
var/run/syslog.log

ESX GI Module (MUX) messages follow the format of <timestamp>EPSecMUX<[ThreadID]>: <message>

For example:

```
2017-07-16T05:44:49Z EPSecMux[38340669]: [ERROR] (EPSEC) [38340669]
Attempted to recv 4 bytes from sd 49, errno = 104 (Connection reset by peer)
```

In the above example

- [ERROR] is the type of message. Other types can be [DEBUG], [INFO]
- (EPSEC) represents that the messages are specific to Endpoint Security

Enabling and Viewing Log Files

To view the version of the ESX GI Module VIB installed on the host, run the `#esxcli software vib list | grep epsec-mux` command.

To turn on full logging, perform these steps on the ESXi host command shell:

1 Run the ps -c |grep Mux command to find the ESX GI Module processes that are currently running.

For example:

```
~ # ps -c | grep Mux
192223 192223 sh  /bin/sh /sbin/watchdog.sh -s vShield-Endpoint-Mux -q 100 -t 1000000 /usr/lib/
vmware/vShield-Endpoint-Mux 900 -c 910
192233 192233 vShield-Endpoint-Mux /usr/lib/vmware/vShield-Endpoint-Mux 900 -c 910
192236 192233 vShield-Endpoint-Mux /usr/lib/vmware/vShield-Endpoint-Mux 900 -c 910
```

2 If the service is not running, you can restart it with these commands:`/etc/init.d/vShield-Endpoint-Mux start` or `/etc//init.d/vShield-Endpoint-Mux restart`.

3 To stop the running ESX GI Module processes, including the watchdog.sh process, run the `~ # kill -9 192223 192233 192236` command.

Note that two ESX GI Module processes are spawned.

4 Start an ESX GI Module with a new -doption. Note that option -d does not exist for epsec-mux builds 5.1.0-01255202 and 5.1.0-01814505 `~ # /usr/lib/vmware/vShield-Endpoint-Mux -d 900 -c 910`

5 View the ESX GI Module log messages in the `/var/log/syslog.log` file on the ESXi host. Check that the entries corresponding to the global solutions, solution ID, and port number are specified correctly.

Example: Sample muxconfig.xml File

```
<?xml version="1.0" encoding="UTF-8"?>

<EndpointConfig>

  <InstalledSolutions>

    <Solution>

      <id>100</id>

      <ipAddress>xxx.xxx.xxx.xxx</ipAddress>

      <listenOn>ip</listenOn>

      <port>48655</port>

      <uuid>42383371-3630-47b0-8796-f1d9c52ab1d0</uuid>

      <vmxPath>/vmfs/volumes/7adf9e00-609186d9/EndpointService (216)/EndpointService (216).vmx</
vmxPath>

    </Solution>

    <Solution>

      <id>102</id>

      <ipAddress>xxx.xxx.xxx.xxx</ipAddress>

      <listenOn>ip</listenOn>

      <port>48651</port>

      <uuid>423839c4-c7d6-e92e-b552-79870da05291</uuid>

      <vmxPath>/vmfs/volumes/7adf9e00-609186d9/apoon/EndpointSVM-alpha-01/EndpointSVM-alpha-01.vmx</
vmxPath>

    </Solution>

    <Solution>

      <id>6341068275337723904</id>

      <ipAddress>xxx.xxx.xxx.xxx</ipAddress>

      <listenOn>ip</listenOn>
```

```
      <port>48655</port>
      <uuid>42388025-314f-829f-2770-a143b9cbd1ee</uuid>
      <vmxPath>/vmfs/volumes/7adf9e00-609186d9/DlpService (1)/DlpService (1).vmx</vmxPath>
    </Solution>
  </InstalledSolutions>
  <DefaultSolutions/>
  <GlobalSolutions>
    <solution>
      <id>100</id>
      <tag></tag>
      <order>0</order>
    </solution>
    <solution>
      <id>102</id>
      <tag></tag>
      <order>10000</order>
    </solution>
    <solution>
      <id>6341068275337723904</id>
      <tag></tag>
      <order>10001</order>
    </solution>
  </GlobalSolutions>
</EndpointConfig>
```

GI Thin Agent Logs

The thin agent is installed on the VM Guest OS and detects user logon details.

Log Path and Sample Message

The thin agent consists of GI drivers – vsepflt.sys, vnetflt.sys, vnetwfp.sys (Windows 10 and later).

The thin agent logs are on the ESXi host, as part of the VCenter Log Bundle. The log path is `/vmfs/volumes/<datastore>/<vmname>/vmware.log` For example: `/vmfs/volumes/5978d759-56c31014-53b6-1866abaace386/Windows10-(64-bit)/vmware.log`

Thin agent messages follow the format of <timestamp> <VM Name><Process Name><[PID]>: <message>.

In the log example below `Guest: vnet or Guest:vsep`, indicate log messages related to the respective GI drivers, followed by debug messages.

For example:

```
2017-10-17T14:25:19.877Z| vcpu-0| I125: Guest: vnet: AUDIT: DriverEntry :
 vnetFilter build-4325502 loaded
2017-10-17T14:25:20.282Z| vcpu-0| I125: Guest: vsep:
AUDIT: VFileSocketMgrConnectHelper : Mux is connected
2017-10-17T14:25:20.375Z| vcpu-0| I125:
Guest: vsep: AUDIT: DriverEntry : vfileFilter build-4286645 loaded

2017-10-17T18:22:35.924Z| vcpu-0| I125: Guest: vsep: AUDIT:
VFileSocketMgrConnectHelper : Mux is connected
2017-10-17T18:24:05.258Z| vcpu-0| I125: Guest: vsep: AUDIT:
VFileFltPostOpCreate : File (\Windows\System32\Tasks\Microsoft\Windows\
SoftwareProtectionPlatform\SvcRestartTask) in a transaction, ignore
```

Example: Enabling vShield Guest Introspection Thin Agent Driver Logging

Because the debug setting can flood the vmware.log file to the point that it throttles, we recommend you disable the debug mode as soon as you have collected all the required information.

This procedure requires you to modify the Windows registry. Before you modify the registry, ensure to take a backup of the registry. For more information on backing up and restoring the registry, see the Microsoft Knowledge Base article 136393.

To enable debug logging for the thin agent driver:

1. Click **Start > Run**. Enter regedit, and click **OK**. The Registry Editor window opens. For more information seen the Microsoft Knowledge Base article 256986.
2. Create this key using the registry editor: `HKEY_LOCAL_Machine\SYSTEM\CurrentControlSet\services\vsepflt\parameters`.
3. Under the newly created parameters key, create these DWORDs. Ensure that hexadecimal is selected when putting in these values:

   ```
   Name: log_dest
   Type: DWORD
   Value: 0x2
   ```

```
Name: log_level
Type: DWORD
Value: 0x10
```

Other values for `log level` parameter key:

```
Audit 0x1
Error 0x2
Warn 0x4
Info 0x8
Debug 0x10
```

4 Open a command prompt as an administrator. Run these commands to unload and reload the vShield Endpoint filesystem mini driver:

- `fltmc unload vsepflt`
- `fltmc load vsepflt`

You can find the log entries in the vmware.log file located in the virtual machine.

Enabling vShield GI Network Introspection Driver Logging

Because the debug setting can flood the vmware.log file to the point that it can make it to throttle, we recommend you disable the debug mode as soon as you have collected all the required information.

This procedure requires you to modify the Windows registry. Before you modify the registry, ensure to take a backup of the registry. For more information on backing up and restoring the registry, see the Microsoft Knowledge Base article 136393.

1 Click **Start > Run**. Enter regedit, and click **OK**. The Registry Editor window opens. For more information seen the Microsoft Knowledge Base article 256986.

2 Edit the registry:

```
Windows Registry Editor Version 5.0
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\vnetflt\Parameters]
"log_level" = DWORD: 0x0000001F
"log_dest"  = DWORD: 0x00000001
```

3 Reboot the virtual machine.

vsepflt.sys and vnetflt.sys Log File Location

With the log_dest registry settings DWORD: 0x00000001, the Endpoint thin agent driver logs into the debugger. Run the debugger (DbgView from SysInternals or windbg) to capture the debug output.

Alternatively you can set the log_dest registry setting to DWORD:0x000000002, in which case the driver logs will be printed to vmware.log file, which is located in the corresponding virtual machine folder on the ESXi Host.

Enabling UMC logging

The Guest Introspection user-mode component (UMC) runs within the VMware Tools service in the protected virtual machine.

1 On Windows XP and Windows Server 2003, create a `tools config` file if it doesn't exist in the following path: `C:\Documents and Settings\All Users\Application Data\VMware\VMware Tools\tools.conf`.

2 On Windows Vista, Windows 7 and Windows Server 2008, create a `tools config` file if it doesn't exist in the following path: `C:\ProgramData\VMWare\VMware Tools\tools.conf`

3 Add these lines in the `tools.conf` file to enable UMC component logging.

```
[logging]
log = true
vsep.level = debug
vsep.handler = vmx
```

With the `vsep.handler = vmx` setting, the UMC component logs into the `vmware.log` file, which is located in the corresponding virtual machine folder on the ESXi host.

With the following setting logs, the UMC component logs will be printed in the specified log file.

```
vsep.handler = file
vsep.data = c:/path/to/vsep.log
```

GI EPSecLib and SVM Logs

The EPSecLib receives events from the ESXi host ESX GI Module (MUX).

Log Path and Sample Message

EPSecLib Log Path
/var/log/syslog
var/run/syslog

EPSecLib messages follow the format of <timestamp> <VM Name><Process Name><[PID]>: <message>

In the example below [ERROR] is the type of message and (EPSEC) represents the messages that are specific to Guest Introspection.

For example:

```
Oct 17 14:26:00 endpoint-virtual-machine EPSecTester[7203]: [NOTICE] (EPSEC)
 [7203] Initializing EPSec library build: build-00000

Oct 17 14:37:41 endpoint-virtual-machine EPSecSample: [ERROR] (EPSEC) [7533] Event
terminated reading file. Ex: VFileGuestEventTerminated@tid=7533: Event id: 3554.
```

Collecting Logs

To enable debug logging for the EPSec library, which is a component inside GI SVM:

1 Log in to the GI SVM by obtaining the console password from NSX Manager.

2 Create `/etc/epseclib.conf` file and add:

`ENABLE_DEBUG=TRUE`

`ENABLE_SUPPORT=TRUE`

3 Change permissions by running the `chmod 644 /etc/epseclib.conf` command.

4 Restart the GI-SVM process by running the `/usr/local/sbin/rcusvm restart` command.

This enables debug logging for EPSecLib on the GI SVM and the debug logs can be found in /var/log/messages which are applicable for NSX for vSphere 6.2.x & 6.3.x. Because the debug setting can flood the vmware.log file to the point that it can make it to throttle, we recommend you disable the debug mode as soon as you have collected all the required information.

GI SVM Logs

Before you capture logs, determine the Host ID, or Host MOID:

- Run the `show cluster all` and `show cluster <cluster ID>` commands in the NSX Manager.

For example:

```
nsxmgr-01a> show cluster all

No.  Cluster Name       Cluster Id              Datacenter Name   Firewall Status
1    RegionA01-COMP01   domain-c26              RegionA01         Enabled
2    RegionA01-MGMT01   domain-c71              RegionA01         Enabled

nsxmgr-01a> show cluster  domain-c26

Datacenter: RegionA01
Cluster: RegionA01-COMP01
No.  Host Name            Host Id                 Installation Status
1    esx-01a.corp.local   host-29                 Ready
2    esx-02a.corp.local   host-31                 Ready
```

1 To determine the current logging state, run this command:

`GET https://nsxmanager/api/1.0/usvmlogging/host-##/com.vmware.vshield.usvm`

`GET https://nsxmanager/api/1.0/usvmlogging/host-##/root`

2 To change the current logging state run this command:

`POST https://nsxmanager/api/1.0/usvmlogging/host-##/changelevel`

```
## Example to change root logger ##

<?xml version="1.0" encoding="UTF-8" ?>
<logginglevel>
```

```
<loggerName>root</loggerName>
<level>DEBUG</level>
</logginglevel>

## Example to change com.vmware.vshield.usvm ##

<?xml version="1.0" encoding="UTF-8" ?>
<logginglevel>
<loggerName>com.vmware.vshield.usvm</loggerName>
<level>DEBUG</level>
</logginglevel>
```

3 To generate logs, run this command:

`GET https://NSXMGR_IP/api/1.0/hosts/host.###/techsupportlogs`

Select `Send` and `Download`.

Note that this command generates GI SVM logs and saves the file as `techsupportlogs.log.gz` file. Because the debug setting can flood the vmware.log file to the point that it can make it to throttle, we recommend you disable the debug mode as soon as you have collected all the required information.

Collecting Guest Introspection Environment and Work Details

Collecting environment details is useful when checking the compatibility of components.

1 Determine if NSX Guest Introspection is used in the customer environment. If it is not, remove the Guest Introspection service for the virtual machine, and confirm the issue is resolved.

2 Collect environment details:

a ESXi build version - Run the command`uname -a` on the ESXi host or click on a host in the vSphere Web Client and look for the build number at top of the right-hand pane.

b Linux product version and build number

c `/usr/sbin/vsep -v` will give the production version

```
Build number
------------------
Ubuntu
dpkg -l | grep vmware-nsx-gi-file
SLES12 and RHEL7
rpm -qa | grep vmware-nsx-gi-file
```

3 VMware NSX ® for vSphere ® version, and the following:

- Partner solution name and version number
- EPSec Library version number used by the partner solution: Log into the GI SVM and run `#strings` path to `EPSec library/libEPSec.so | grep BUILD`
- Guest operating system in the virtual machine

- Any other third-party applications or file system drivers

4 ESX GI Module (MUX) version - run the command esxcli software `vib list | grep epsec–mux`.

5 Collect workload details, such as the type of server.

6 Collect ESXi host logs. For more information, see Collecting diagnostic information for VMware ESX/ESXi (653).

7 Collect service virtual machine (GI SVM) logs from the partner solution. Reach out to your partner for more details on GI SVM log collection.

8 Collect a suspend state file while the problem is occurring, see Suspending a virtual machine on ESX/ESX (2005831) to collect diagnostic information.

9 After collecting date, compare the compatibility of the vSphere components. For more information, see the VMware Product Interoperability Matrices.

Troubleshooting the Thin Agent on Linux or Windows

The Guest Introspection thin agent is installed withVMware Tools ™ on each guest virtual machine.

Troubleshooting the Thin Agent on Linux

If a virtual machine is slow in reading and writing operations, and unzipping or saving files then there may be issues with the thin agent.

1 Check the compatibility of all the components involved. Compatibility is one of the main issues with Endpoint. You need the build numbers for ESXi, vCenter Server, NSX Manager, and which ever Security solution you have chosen (Trend Micro, McAfee, Kaspersky, Symantec etc). Once this data has been collected, compare the compatibility of the vSphere components. For more information, see the VMware Product Interoperability Matrices.

2 Ensure that File Introspection is installed on the system.

3 Verify that the thin agent is running by with the *service vsepd status* command. Once this command is executed you should see the vsep service in running state.

4 If you believe that the thin agent is causing a performance issue with the system, stop the service by running the *service vsepd stop* command.

5 Then perform a test to get a baseline. You can then start the vsep service and perform another test by running the *service vsepd start* command.

6 Enable debugging for the Linux thin agent:

   a Open the `/etc/vsep/vsep.conf` file

   b Change DEBUG_LEVEL=4 to DEBUG_LEVEL=7 for all logs

   c This can be set to DEBUG_LEVEL=6 for moderate logs

d The default log destination(DEBUG_DEST=2) is vmware.log (on host) to change it to guest (i.e /var/log/message or /var/log/syslog) set DEBUG_DEST=1

Note Enabling full logging may result in heavy log activity flooding the `vmware.log` file, causing it to potentially grow to be very large. Disable full logging as soon as possible.

Troubleshooting the Thin Agent on Windows

1 Check the compatibility of all the components involved. You need the build numbers for ESXi, vCenter Server, NSX Manager, and which ever Security solution you have chosen (Trend Micro, McAfee, Kaspersky, Symantec etc). Once all of this data has been collected, you can compare the compatibility of the vSphere components. For more information, see the VMware Product Interoperability Matrices.

2 Ensure that VMware Tools ™ is up-to-date. If you see that only a particular virtual machine is affected, see Installing and upgrading VMware Tools in vSphere (2004754).

3 Verify that the thin agent is loaded by running the Powershell command `fltmc`.

Once this command is executed, You should see the name vsepflt on the list of drivers. If the driver is not loaded, you should be able to load the driver with the `fltmc load vsepflt` command.

4 If t the thin agent is causing a performance issue with the system, unload the driver with this command: `fltmc unload vsepflt`.

Next, perform a test to get a baseline. You can then load the driver and perform another test by running this command:

`fltmc load vsepflt`.

If you do verify that there is a performance problem with the Thin agent, see Slow VMs after upgrading VMware tools in NSX and vCloud Networking and Security (2144236).

5 If you are not using Network Introspection, remove or disable this driver.

Network Introspection can also be removed through the Modify VMware Tools installer:

a Mount the VMware Tools installer.

b Navigate to **Control Panel > Programs and Features**.

c Right-click **VMware Tools > Modify**.

d Select **Complete install**.

e Find NSX File Introspection. There should be a sub folder just for Network Introspection.

f Disable **Network Introspection**.

g Reboot the VM to complete the uninstallation of the driver.

6 Enable debug logging for the thin agent. For more information, see Guest Introspection Logs. All debugging information is configured to log to the vmware.log file for that virtual machine.

7 Review the file scans of the thin agent by reviewing the procmon logs. For more information, see Troubleshooting vShield Endpoint performance issues with anti-virus software (2094239).

Collect Environment and Workload Details

1 Determine if NSX Guest Introspection is used in the customer environment. If it is not, remove the Guest Introspection service for the virtual machine, and confirm the issue is resolved. Troubleshoot a Guest Introspection issue only if Guest Inspection is required.

2 Collect environment details:

   a ESXi build version - Run the command `uname –a` on the ESXi host or click on a host in the vSphere Web Client and look for the build number at top of the right-hand pane.

   b Linux product version and build number

   c */usr/sbin/vsep -v* will give the production version

```
Build number
___________________
Ubuntu
dpkg –l | grep vmware–nsx–gi–file
SLES12 and RHEL7
rpm –qa | grep vmware–nsx–gi–file
```

3 VMware NSX ® for vSphere ® version, and the following:

   - Partner solution name and version number
   - EPSec Library version number used by the partner solution: Log into the SVM and run `#strings` path to `EPSec library/libEPSec.so | grep BUILD`
   - Guest operating system in the virtual machine
   - Any other third-party applications or file system drivers

4 ESX GI Module (MUX) version - run the command *esxcli software vib list | grep epsec-mux*.

5 Collect workload details, such as the type of server.

6 Collect ESXi host logs. For more information, see Collecting diagnostic information for VMware ESX/ESXi (653).

7 Collect service virtual machine (SVM) logs from the partner solution. Reach out to your partner for more details on SVM log collection.

8 Collect a suspend state file while the problem is occurring, see Suspending a virtual machine on ESX/ESX (2005831) to collect diagnostic information.

Troubleshooting Thin Agent crash

If the Thin Agent crashes, the core file is generated in the `/directory`. Collect the core dump file (core) from `location / directory`. Use the `file` command to check if core is generated by vsep. For example:

```
# file core
core: ELF 64-bit LSB  core file x86-64, version 1 (SYSV), SVR4-style, from '/usr/sbin/vsep'
```

Virtual machine hangs or freezes

Collect the VMware vmss file of the virtual machine in a suspended state, see Suspending a virtual machine on ESX/ESXi to collect diagnostic information (2005831) or crash the virtual machine and collect the full memory dump file. VMware offers a utility to convert an ESXi vmss file to a core dump file. See Vmss2core fling for more information.

Troubleshooting ESX GI Module (MUX)

ESX GI Module (MUX)

If all virtual machines on an ESXi host are not working with Guest Introspection, or there are alarms on a particular host regarding communication to the GI SVA, then it could be a problem with the ESX GI Module module on the ESXi host.

1 Check to see if the service is running on the ESXi host by running the *# /etc/init.d/vShield-Endpoint-Mux status* command:

   For example:

   ```
   # /etc/init.d/vShield-Endpoint-Mux status
            vShield-Endpoint-Mux is running
   ```

2 If you see that the service is not running, restart it or start it with this command:

   `/etc/init.d/vShield-Endpoint-Mux start`

   or

   `/etc/init.d/vShield-Endpoint-Mux restart`

   Note that it is safe to restart this service during production hours as it does not have any great impact, and restarts in a couple of seconds.

3 To get a better idea of what the ESX GI Module is doing or check the communication status, you can check the logs on the ESXi host. ESX GI Module logs are written to the host `/var/log/syslog` file. This is also included in the ESXi host support logs.

   For more information, see Collecting diagnostic information for ESX/ESXi hosts and vCenter Server using the vSphere Web Client (2032892)

4 The default logging option for ESX GI Module is info and can be raised to debug to gather more information:

   For more information, see Guest Introspection Logs.

5 Re-installing the ESX GI Module module can also fix many issues, especially if the wrong version is installed, or the ESXi host was brought into the environment which previously had Endpoint installed on it. This needs to be removed and re-installed.

   To remove the VIB, run this command: *esxcli software vib remove -n epsec-mux*

6 If you run into issues with the VIB installation, check the /var/log/esxupdate.log file on the ESXi host. This log shows the most clear information as to why the driver did not successfully get installed. This is a common issue for ESX GI Module installation issues. For more information, see Installing NSX Guest Introspection services (ESX GI Module VIB) on the ESXi host fails in VMware NSX for vSphere 6.x (2135278).

7 To check for a corrupt ESXi image look for a message similar to this:

```
esxupdate: esxupdate: ERROR: Installation Error:
 (None, 'No image profile is found on the host or image profile is empty.
 An image profile is required to install or remove VIBs. To install an image profile,
 use the esxcli image profile install command.')
```

8 To verify that the image is corrupted run the command *cd /vmfs/volumes* on the ESXi host.

   a Search for the imgdb.tgz file by running this command: *find * | grep imgdb.tgz*.

      This command normally results in two matches. For example:

      0ca01e7f-cc1ea1af-bda0-1fe646c5ceea/imgdb.tgz or *edbf587b-da2add08-3185-3113649d5262/imgdb.tgz*

   b On each match, run this command: *ls -l match_result*

      For example:

```
> ls –l 0ca01e7f–cc1ea1af–bda0–1fe646c5ceea/imgdb.tgz –rwx––––––
 1 root root  26393 Jul 20 19:28 0ca01e7f–cc1ea1af–bda0–1fe646c5ceea/imgdb.tgz
   > ls –l edbf587b–da2add08–3185–3113649d5262/imgdb.tgz –rwx––––––
  1 root root   93 Jul 19 17:32 edbf587b–da2add08–3185–3113649d5262/imgdb.tgz
```

      The default size for the imgdb.tgz file is far greater than the other file or if one of the files is only a couple of bytes, it indicates that the file is corrupt. The only supported way to resolve this is to re-install ESXi for that particular ESXi host.

Troubleshooting EPSecLib

The NSX Manager handles the deployment of this virtual machine.

EPSecLib

In the past (with vShield), the third-party SVA solution handles the deployment. That solution now connects to the NSX Manager. The NSX Manager handles the deployment of this SVA. If there are alarms on the SVAs in the environment, redeploy them through the NSX Manager.

- Any configuration is lost as this is all stored inside the NSX Manager.
- It is better to redeploy the SVA virtual machines, instead of rebooting them.
- NSX relies on EAM for deploying and monitoring VIBs and SVMs on host such as the SVA.
- EAM is the source of truth for determining the Install Status.
- The Install status in NSX User Interface (UI) can only tell if the VIBs are installed, or if the SVM is powered on.
- The Service status in NSX UI indicates if the functionality in the virtual machine is working

SVA deployment and relationship between NSX and vCenter Server Process

1. When the Cluster is selected to be prepared for Endpoint, an Agency is created on EAM to deploy the SVA.
2. EAM then deploys the ovf to the ESXi host with the agency info it created.
3. NSX Manager verifies if ovf was deployed by EAM.
4. NSX Manager verifies if virtual machine was powered on by EAM.
5. NSX Manager communicates to the Partner SVA Solution Manager that the virtual machine was powered on and registered.
6. EAM sends an event to NSX to indicate that installation was complete.
7. Partner SVA Solution Manager sends an event to NSX to indicate that the service inside the SVA virtual machine is up and running.
8. If you are having an issue with the SVA, there are two places you can look at the logs. You can check the EAM logs, as EAM handles the deployment of these virtual machines. For more information, see Collecting diagnostic information for VMware vCenter Server 4.x, 5.x and 6.0 (1011641). Alternatively, look at the SVA logs.

   For more information, see Guest Introspection Logs.
9. If there is a problem with the SVA deployment, it is possible that there is an issue with EAM and the communication to NSX Manager. You can check the EAM logs, and the simplest thing to do is to restart the EAM Service. For more information, see Host Preparation.
10. If all of the above seems to be working, but you want to test the Endpoint functionality, you can test this with an Eicar Test file:
    - Create a text file with any label. For example: eicar.test.
    - The contents of the file should only be the following string:

      ```
      X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
      ```

- Save the file. Upon saving, you should see that the file is deleted. This verifies that the Endpoint solution is working.