NSX-T Installation Guide

VMware NSX-T Data Center 2.1

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-T Installation Guide

The *NSX-T Installation Guide* describes how to install the VMware NSX-T ® product. The information includes step-by-step configuration instructions, and suggested best practices.

Intended Audience

This information is intended for anyone who wants to install or use NSX-T. This information is written for experienced system administrators who are familiar with virtual machine technology and network virtualization concepts.

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.

Overview of NSX-T

1

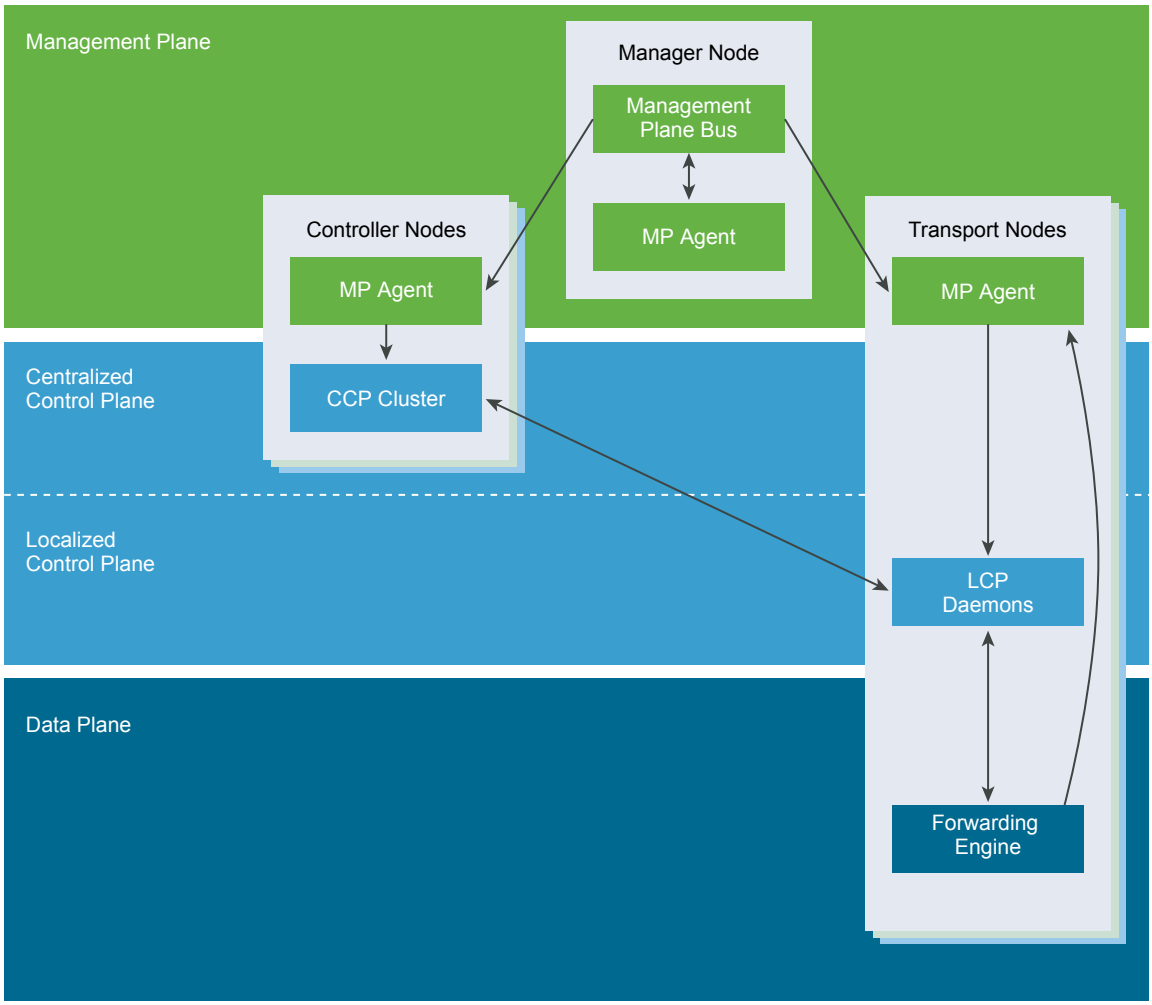
In much the same way that server virtualization programmatically creates, snapshots, deletes and restores software-based virtual machines (VMs), NSX-T network virtualization programmatically creates, deletes, and restores software-based virtual networks.

With network virtualization, the functional equivalent of a network hypervisor reproduces the complete set of Layer 2 through Layer 7 networking services (for example, switching, routing, access control, firewalling, QoS) in software. As a result, these services can be programmatically assembled in any arbitrary combination, to produce unique, isolated virtual networks in a matter of seconds.

NSX-T works by implementing three separate but integrated planes: management, control, and data. The three planes are implemented as a set of processes, modules, and agents residing on three types of nodes: manager, controller, and transport nodes.

- Every node hosts a management plane agent.
- The NSX Manager node hosts API services. Each NSX-T installation supports a single NSX Manager node.
- NSX Controller nodes host the central control plane cluster daemons.
- NSX Manager and NSX Controller nodes may be co-hosted on the same physical server.

- Transport nodes host local control plane daemons and forwarding engines.



This chapter includes the following topics:

- Data Plane
- Control Plane
- Management Plane
- NSX Manager
- NSX Policy Manager
- NSX Controller
- Logical Switches
- Logical Routers
- NSX Edge
- Transport Zones
- Key Concepts

Data Plane

Performs stateless forwarding/transformation of packets based on tables populated by the control plane and reports topology information to the control plane, and maintains packet level statistics.

The data plane is the source of truth for the physical topology and status for example, VIF location, tunnel status, and so on. If you are dealing with moving packets from one place to another, you are in the data plane. The data plane also maintains status of and handles failover between multiple links/tunnels. Per-packet performance is paramount with very strict latency or jitter requirements. Data plane is not necessarily fully contained in kernel, drivers, userspace, or even specific userspace processes. Data plane is constrained to totally stateless forwarding based on tables/rules populated by control plane.

The data plane also may have components that maintain some amount of state for features such as TCP termination. This is different from the control plane managed state such as MAC:IP tunnel mappings, because the state managed by the control plane is about how to forward the packets, whereas state managed by the data plane is limited to how to manipulate payload.

Control Plane

Computes all ephemeral runtime state based on configuration from the management plane, disseminates topology information reported by the data plane elements, and pushes stateless configuration to forwarding engines.

The control plane is sometimes described as the signaling for the network. If you are dealing with processing messages in order to maintain the data plane in the presence of static user configuration, you are in the control plane (for example, responding to a vMotion of a virtual machine (VM) is a control plane responsibility, but connecting the VM to the logical network is a management plane responsibility) Often the control plane is acting as a reflector for topological info from the data plane elements to one another for example, MAC/Tunnel mappings for VTEPs. In other cases, the control plane is acting on data received from some data plane elements to (re)configure some data plane elements such as, using VIF locators to compute and establish the correct subset mesh of tunnels.

The set of objects that the control plane deals with include VIFs, logical networks, logical ports, logical routers, IP addresses, and so on.

The control plane is split into two parts in NSX-T, the central control plane (CCP), which runs on the NSX Controller cluster nodes, and the local control plane (LCP), which runs on the transport nodes, adjacent to the data plane it controls. The Central Control Plane computes some ephemeral runtime state based on configuration from the management plane and disseminates information reported by the data plane elements via the local control plane. The Local Control Plane monitors local link status, computes most ephemeral runtime state based on updates from data plane and CCP, and pushes stateless configuration to forwarding engines. The LCP shares fate with the data plane element which hosts it.

Management Plane

The management plane provides a single API entry point to the system, persists user configuration, handles user queries, and performs operational tasks on all management, control, and data plane nodes in the system.

For NSX-T anything dealing with querying, modifying, and persisting user configuration is a management plane responsibility, while dissemination of that configuration down to the correct subset of data plane elements is a control plane responsibility. This means that some data belongs to multiple planes depending on what stage of its existence it is in. The management plane also handles querying recent status and statistics from the control plane, and sometimes directly from the data plane.

The management plane is the one and only source-of-truth for the configured (logical) system, as managed by the user via configuration. Changes are made using either a RESTful API or the NSX-T UI.

In NSX there is also a management plane agent (MPA) running on all cluster and transport nodes. Example use cases are bootstrapping configurations such as central management node address(es) credentials, packages, statistics, and status. The MPA can run relatively independently of the control plane and data plane, and to be restarted independently if its process crashes or wedges, however, there are scenarios where fate is shared because they run on the same host. The MPA is both locally accessible and remotely accessible. MPA runs on transport nodes, control nodes, and management nodes for node management. On transport nodes it may perform data plane related tasks as well.

Tasks that happen on the management plan include:

- Configuration persistence (desired logical state)
- Input validation
- User management -- role assignments
- Policy management
- Background task tracking

NSX Manager

NSX Manager provides the graphical user interface (GUI) and the REST APIs for creating, configuring, and monitoring NSX-T components, such as logical switches, and NSX Edge services gateways.

NSX Manager is the management plane for the NSX-T eco-system. NSX Manager provides an aggregated system view and is the centralized network management component of NSX-T. It provides a method for monitoring and troubleshooting workloads attached to virtual networks created by NSX-T. It provides configuration and orchestration of:

- Logical networking components – logical switching and routing
- Networking and Edge services
- Security services and distributed firewall

NSX Manager allows seamless orchestration of both built-in and external services. All security services, whether built-in or 3rd party, are deployed and configured by the NSX-T management plane. The management plane provides a single window for viewing services availability. It also facilitates policy based service chaining, context sharing, and inter-service events handling. This simplifies the auditing of the security posture, streamlining application of identity-based controls (for example, AD and mobility profiles).

NSX Manager also provides REST API entry-points to automate consumption. This flexible architecture allows for automation of all configuration and monitoring aspects via any cloud management platform, security vendor platform, or automation framework.

The NSX-T Management Plane Agent (MPA) is an NSX Manager component that lives on each and every node (hypervisor). The MPA is in charge of persisting the desired state of the system and for communicating non-flow-controlling (NFC) messages such as configuration, statistics, status and real time data between transport nodes and the management plane.

NSX Policy Manager

NSX Policy Manager provides an intent-based system to simplify the consumption of NSX-T services.

NSX Policy Manager provides a graphical user interface (GUI) and REST APIs to specify the intent related to networking, security, and availability.

NSX Policy Manager accepts the intent from the user in the form of a tree-based data model and configures the NSX Manager to realize that intent. The NSX Policy Manager supports communication intent specification that configures a distributed firewall on the NSX Manager.

NSX Controller

NSX Controller is an advanced distributed state management system that controls virtual networks and overlay transport tunnels.

NSX Controller is deployed as a cluster of highly available virtual appliances that are responsible for the programmatic deployment of virtual networks across the entire NSX-T architecture. The NSX-T Central Control Plane (CCP) is logically separated from all data plane traffic, meaning any failure in the control plane does not affect existing data plane operations. Traffic doesn’t pass through the controller; instead the controller is responsible for providing configuration to other NSX Controller components such as the logical switches, logical routers, and edge configuration. Stability and reliability of data transport are central concerns in networking. To further enhance high availability and scalability, the NSX Controller is deployed in a cluster of three instances.

Logical Switches

The logical switching capability in the NSX Edge platform provides the ability to spin up isolated logical L2 networks with the same flexibility and agility that exists for virtual machines.

A cloud deployment for a virtual data center has a variety of applications across multiple tenants. These applications and tenants require isolation from each other for security, fault isolation, and to avoid overlapping IP addressing issues. Endpoints, both virtual and physical, can connect to logical segments and establish connectivity independently from their physical location in the data center network. This is enabled through the decoupling of network infrastructure from logical network (i.e., underlay network from overlay network) provided by NSX-T network virtualization.

A logical switch provides a representation of Layer 2 switched connectivity across many hosts with Layer 3 IP reachability between them. If you plan to restrict some logical networks to a limited set of hosts or you have custom connectivity requirements, you may find it necessary to create additional logical switches.

Logical Routers

NSX-T logical routers provide North-South connectivity, thereby enabling tenants to access public networks, and East-West connectivity between different networks within the same tenants.

A logical router is a configured partition of a traditional network hardware router. It replicates the hardware's functionality, creating multiple routing domains within a single router. Logical routers perform a subset of the tasks that can be handled by the physical router, and each can contain multiple routing instances and routing tables. Using logical routers can be an effective way to maximize router usage, because a set of logical routers within a single physical router can perform the operations previously performed by several pieces of equipment.

With NSX-T it’s possible to create two-tier logical router topology: the top-tier logical router is Tier 0 and the bottom-tier logical router is Tier 1. This structure gives both provider administrator and tenant administrators complete control over their services and policies. Administrators control and configure Tier-0 routing and services, and tenant administrators control and configure Tier-1. The north end of Tier-0 interfaces with the physical network, and is where dynamic routing protocols can be configured to exchange routing information with physical routers. The south end of Tier-0 connects to multiple Tier-1 routing layer(s) and receives routing information from them. To optimize resource usage, the Tier-0 layer does not push all the routes coming from the physical network towards Tier-1, but does provide default information.

Southbound, the Tier-1 routing layer interfaces with the logical switches defined by the tenant administrators, and provides one-hop routing function between them. For Tier-1 attached subnets to be reachable from the physical network, route redistribution towards Tier-0 layer must the enabled. However, there isn’t a classical routing protocol (such as OSPF or BGP) running between Tier-1 layer and Tier-0 layer, and all the routes go through the NSX-T control plane. Note that the two-tier routing topology is not mandatory, if there is no need to separate provider and tenant, a single tier topology can be created and in this scenario the logical switches are connected directly to the Tier-0 layer and there is no Tier-1 layer.

A logical router consists of two optional parts: a distributed router (DR) and one or more service routers (SR).

A DR spans hypervisors whose VMs are connected to this logical router, as well as edge nodes the logical router is bound to. Functionally, the DR is responsible for one-hop distributed routing between logical switches and/or logical routers connected to this logical router. The SR is responsible for delivering services that are not currently implemented in a distributed fashion, such as stateful NAT.

A logical router always has a DR, and it has SRs if any of the following is true:

- The logical router is a Tier-0 router, even if no stateful services are configured
- The logical router is Tier-1 router linked to a Tier-0 router and has services configured that do not have a distributed implementation (such as NAT, LB, DHCP)

The NSX-T management plane (MP) is responsible for automatically creating the structure that connects the service router to the distributed router. The MP creates a transit logical switch and allocates it a VNI, then creates a port on each SR and DR, connects them to the transit logical switch, and allocates IP addresses for the SR and DR.

NSX Edge

NSX Edge provides routing services and connectivity to networks that are external to the NSX-T deployment.

With NSX Edge, virtual machines or workloads that reside on the same host on different subnets can communicate with one another without having to traverse a traditional routing interface.

NSX Edge is required for establishing external connectivity from the NSX-T domain, through a Tier-0 router via BGP or static routing. Additionally, an NSX Edge must be deployed if you require network address translation (NAT) services at either the Tier-0 or Tier-1 logical routers.

The NSX Edge gateway connects isolated, stub networks to shared (uplink) networks by providing common gateway services such as NAT, and dynamic routing. Common deployments of NSX Edge include in the DMZ and multi-tenant Cloud environments where the NSX Edge creates virtual boundaries for each tenant.

Transport Zones

A transport zone controls which hosts a logical switch can reach. It can span one or more host clusters. Transport zones dictate which hosts and, therefore, which VMs can participate in the use of a particular network.

A Transport Zone defines a collection of hosts that can communicate with each other across a physical network infrastructure. This communication happens over one or more interfaces defined as Virtual Tunnel Endpoints (VTEPs).

If two transport nodes are in the same transport zone, VMs hosted on those transport nodes can "see" and therefore be attached to NSX-T logical switches that are also in that transport zone. This attachment makes it possible for the VMs to communicate with each other, assuming that the VMs have Layer 2/Layer 3 reachability. If VMs are attached to switches that are in different transport zones, the VMs cannot communicate with each other. Transport zones do not replace Layer 2/Layer 3 reachability

requirements, but they place a limit on reachability. Put another way, belonging to the same transport zone is a prerequisite for connectivity. After that prerequisite is met, reachability is possible but not automatic. To achieve actual reachability, Layer 2 and (for different subnets) Layer 3 networking must be operational.

A host can serve as a transport node if it contains at least one NSX managed virtual distributed switch (N-VDS, previously known as hostswitch). When you create a host transport node and then add the node to a transport zone, NSX-T installs an N-VDS on the host. For each transport zone that the host belongs to, a separate N-VDS is installed. The N-VDS is used for attaching VMs to NSX-T logical switches and for creating NSX-T logical router uplinks and downlinks.

Key Concepts

The common NSX-T concepts that are used in the documentation and user interface.

Compute Manager	A compute manager is an application that manages resources such as hosts and VMs. One example is vCenter Server.
Control Plane	Computes runtime state based on configuration from the management plane. Control plane disseminates topology information reported by the data plane elements, and pushes stateless configuration to forwarding engines.
Data Plane	Performs stateless forwarding or transformation of packets based on tables populated by the control plane. Data plane reports topology information to the control plane and maintains packet level statistics.
External Network	A physical network or VLAN not managed by NSX-T. You can link your logical network or overlay network to an external network through an NSX Edge. For example, a physical network in a customer data center or a VLAN in a physical environment.
Fabric Node	Host that has been registered with the NSX-T management plane and has NSX-T modules installed. For a hypervisor host or NSX Edge to be part of the NSX-T overlay, it must be added to the NSX-T fabric.
DNE Key Manager	The component of the Distributed Network Encryption (DNE) feature that manages the keys used to provide encrypted and authenticated connections between two endpoints within a Software Defined Data Center (SDDC).
Logical Port Egress	Inbound network traffic to the VM or logical network is called egress because traffic is leaving the data center network and entering the virtual space.
Logical Port Ingress	Outbound network traffic from the VM to the data center network is called ingress because traffic is entering the physical network.
Logical Router	NSX-T routing entity.

Logical Router Port	Logical network port to which you can attach a logical switch port or an uplink port to a physical network.
Logical Switch	Entity that provides virtual Layer 2 switching for VM interfaces and Gateway interfaces. A logical switch gives tenant network administrators the logical equivalent of a physical Layer 2 switch, allowing them to connect a set of VMs to a common broadcast domain. A logical switch is a logical entity independent of the physical hypervisor infrastructure and spans many hypervisors, connecting VMs regardless of their physical location. In a multi-tenant cloud, many logical switches might exist side-by-side on the same hypervisor hardware, with each Layer 2 segment isolated from the others. Logical switches can be connected using logical routers, and logical routers can provide uplink ports connected to the external physical network.
Logical Switch Port	Logical switch attachment point to establish a connection to a virtual machine network interface or a logical router interface. The logical switch port reports applied switching profile, port state, and link status.
Management Plane	Provides single API entry point to the system, persists user configuration, handles user queries, and performs operational tasks on all of the management, control, and data plane nodes in the system. Management plane is also responsible for querying, modifying, and persisting use configuration.
NSX Controller Cluster	Deployed as a cluster of highly available virtual appliances that are responsible for the programmatic deployment of virtual networks across the entire NSX-T architecture.
NSX Edge Cluster	Collection of NSX Edge node appliances that have the same settings as protocols involved in high-availability monitoring.
NSX Edge Node	Component with the functional goal is to provide computational power to deliver the IP routing and the IP services functions.
NSX Managed Virtual Distributed Switch or KVM Open vSwitch	Software that runs on the hypervisor and provides physical traffic forwarding. The NSX managed virtual distributed switch (N-VDS, previously known as hostswitch) or OVS is invisible to the tenant network administrator and provides the underlying forwarding service that each logical switch relies on. To achieve network virtualization, a network controller must configure the hypervisor virtual switch with network flow tables that form the logical broadcast domains the tenant administrators defined when they created and configured their logical switches.

	Each logical broadcast domain is implemented by tunneling VM-to-VM traffic and VM-to-logical router traffic using the tunnel encapsulation mechanism Geneve. The network controller has the global view of the data center and ensures that the hypervisor virtual switch flow tables are updated as VMs are created, moved, or removed.
NSX Manager	Node that hosts the API services, the management plane, and the agent services.
Open vSwitch (OVS)	Open source software switch that acts as a virtual switch within XenServer, Xen, KVM, and other Linux-based hypervisors.
Overlay Logical Network	Logical network implemented using Layer 2-in-Layer 3 tunneling such that the topology seen by VMs is decoupled from that of the physical network.
Physical Interface (pNIC)	Network interface on a physical server that a hypervisor is installed on.
Tier-0 Logical Router	Provider logical router is also known as Tier-0 logical router interfaces with the physical network. Tier-0 logical router is a top-tier router and can be realized as active-active or active-standby cluster of services router. The logical router runs BGP and peers with physical routers. In active-standby mode the logical router can also provide stateful services.
Tier-1 Logical Router	Tier-1 logical router is the second tier router that connects to one Tier-0 logical router for northbound connectivity and one or more overlay networks for southbound connectivity. Tier-1 logical router can be an active-standby cluster of services router providing stateful services.
Transport Zone	Collection of transport nodes that defines the maximum span for logical switches. A transport zone represents a set of similarly provisioned hypervisors and the logical switches that connect VMs on those hypervisors.
Transport Node	A node capable of participating in an NSX-T overlay or NSX-T VLAN networking. For a KVM host, you can preconfigure the N-DVS, or you can have NSX Manager perform the configuration. For an ESXi host, NSX Manager always configures the N-VDs.
VM Interface (vNIC)	Network interface on a virtual machine that provides connectivity between the virtual guest operating system and the standard vSwitch or vSphere distributed switch. The vNIC can be attached to a logical port. You can identify a vNIC based on its Unique ID (UUID).

Virtual Tunnel Endpoint	Enable hypervisor hosts to participate in an NSX-T overlay. The NSX-T overlay deploys a Layer 2 network on top of an existing Layer 3 network fabric by encapsulating frames inside of packets and transferring the packets over an underlying transport network. The underlying transport network can be another Layer 2 networks or it can cross Layer 3 boundaries. The VTEP is the connection point at which the encapsulation and decapsulation takes place.
NSX-T Unified Appliance	NSX-T Unified Appliance is an appliance included in the NSX-T installation package. You can deploy the appliance in the role of NSX Manager and Policy Manager. In a proof-of-concept environment or production environment, the appliance must have only one role.

Preparing for Installation 2

Before installing NSX-T, make sure your environment is prepared.

This chapter includes the following topics:

- System Requirements
- Ports and Protocols
- Installation Checklist

System Requirements

NSX-T has specific requirements regarding hardware resources and software versions.

Hypervisor Requirements

Hypervisor	Version	CPU Cores	Memory
vSphere	Supported vSphere version	4	16 GB
RHEL KVM	7.4 and 7.3	4	16 GB
Ubuntu KVM	16.04.2 LTS	4	16 GB

For ESXi, NSX-T does not support the Host Profiles and Auto Deploy features.

Caution On RHEL, the `yum update` command might update the kernel version and break the compatibility with NSX-T. Be sure to disable the automatic kernel update when you run `yum update`. Also, after running `yum install`, verify that NSX-T supports the kernel version.

NSX Manager Resource Requirements

Thin virtual disk size is 3.1 GB and thick virtual disk size is 140 GB.

Appliance	Memory	vCPU	Storage	Hardware Version
NSX Manager Small VM	8 GB	2	140 GB	10 or higher
NSX Manager Medium VM	16 GB	4	140 GB	10 or higher
NSX Manager Large VM	32 GB	8	140 GB	10 or higher

The NSX Manager resource requirements apply to the NSX Policy Manager.

NSX Controller Resource Requirements

Appliance	Memory	vCPU	Disk Space
NSX Controller	16 GB	4	120 GB

NSX Edge VM Resource Requirements

Deployment Size	Memory	vCPU	Disk Space	VM Hardware Version
Small	4 GB	2	120 GB	10 or later (vSphere 5.5 or later)
Medium	8 GB	4	120 GB	10 or later (vSphere 5.5 or later)
Large	16 GB	8	120 GB	10 or later (vSphere 5.5 or later)

Note For NSX Manager and NSX Edge, the small appliance is for proof-of-concept deployments. The medium appliance is suitable for a typical production environment and can support up to 64 hypervisors. The large appliance is for large-scale deployments with more than 64 hypervisors.

Note VMXNET 3 vNIC is supported only for the NSX Edge VM.

NSX Edge VM and Bare-Metal NSX Edge CPU Requirements

Hardware	Type
CPU	■ Xeon 56xx (Westmere-EP) ■ Xeon E7-xxxx (Westmere-EX and later CPU generation) ■ Xeon E5-xxxx (Sandy Bridge and later CPU generation)

Bare-Metal NSX Edge Specific NIC Requirements

NIC Type	Description	PCI Device ID
Intel X520/Intel 82599	IXGBE_DEV_ID_82599_KX4	0x10F7
	IXGBE_DEV_ID_82599_KX4_MEZZ	0x1514
	IXGBE_DEV_ID_82599_KR	0x1517
	IXGBE_DEV_ID_82599_COMBO_BACK PLANE	0x10F8
		0x000C
	IXGBE_SUBDEV_ID_82599_KX4_KR_ MEZZ	0x10F9
		0x10FB
	IXGBE_DEV_ID_82599_CX4	0x11A9
	IXGBE_DEV_ID_82599_SFP	0x1F72
	IXGBE_SUBDEV_ID_82599_SFP	0x17D0
	IXGBE_SUBDEV_ID_82599_RNDC	0x0470
	IXGBE_SUBDEV_ID_82599_560FLR	0x1507
	IXGBE_SUBDEV_ID_82599_ECNA_DP	0x154D
	IXGBE_DEV_ID_82599_SFP_EM	0x154A
	IXGBE_DEV_ID_82599_SFP_SF2	0x1558
	IXGBE_DEV_ID_82599_SFP_SF_QP	0x1557
	IXGBE_DEV_ID_82599_QSFP_SF_QP	0x10FC
	IXGBE_DEV_ID_82599EN_SFP	0x151C
	IXGBE_DEV_ID_82599_XAUI_LOM	
	IXGBE_DEV_ID_82599_T3_LOM	
Intel X540	IXGBE_DEV_ID_X540T	0x1528
	IXGBE_DEV_ID_X540T1	0x1560
Intel X550	IXGBE_DEV_ID_X550T	0x1563
	IXGBE_DEV_ID_X550T1	0x15D1
Intel X710	I40E_DEV_ID_SFP_X710	0x1572
	I40E_DEV_ID_KX_C	0x1581
	I40E_DEV_ID_10G_BASE_T	0x1586
Intel XL710	I40E_DEV_ID_KX_B	0x1580
	I40E_DEV_ID_QSFP_A	0x1583
	I40E_DEV_ID_QSFP_B	0x1584
	I40E_DEV_ID_QSFP_C	0x1585

Bare-Metal NSX Edge Memory, CPU, and Disk Requirements

Memory	CPU Cores	Disk Space
32 GB	8	120 GB

NSX Manager Browser Support

Browser	Windows 10	Windows 8.1	Ubuntu 14.04	Mac OS X 10.11. 10.12
Internet Explorer 11	Yes	Yes		
Firefox 55			Yes	Yes
Chrome 60	Yes	Yes		Yes
Safari 10				Yes
Microsoft Edge 40	Yes			

Note Internet Explorer 11 in compatibility mode is not supported.

Supported Browser minimum resolution is 1280 x 800 px.

Distributed Network Encryption (DNE) Resource Requirements

Appliance	Memory	vCPU	Disk Space
DNE Key Manager	8 GB	2	20 GB

Additional requirements include:

- DNE is supported only on vSphere 6.5.
- DNE Key Manager is supported on vSphere with HA.
- For KVM, the integrity only option is supported only on >4.2 kernels.
- Deploy the NSX Edge node on a separate L3 subnet.
- If an encryption rule is applied on a hypervisor, the virtual tunnel endpoint (VTEP) interface minimum MTU size must be 1700. MTU size 2000 or later is preferred.

Ports and Protocols

Ports and protocols allow node-to-node communication paths in NSX-T, the paths must be secured and authenticated, and a storage location for the credentials must be used to establish mutual authentication.

By default, all certificates are self-signed certificates. The northbound GUI and API certificates and private keys can be replaced by CA signed certificates.

There are internal daemons that communicate over the loopback or UNIX domain sockets:

- KVM: MPA, netcpa, nsx-agent, OVS
- ESX: netcpa, ESX-DP (in the kernel)

In the RMQ user database (db), passwords are hashed with a non-reversible hash function. So h(p1) is the hash of password p1.

CCP	Central control plane
LCP	Local control plane
MP	Management plane
MPA	Management plane agent

Note You must enable SSH to access NSX-T nodes.

TCP and UDP Ports Used by NSX Manager

NSX Manager uses certain TCP and UDP ports to communicate with other components and products. These ports must be open in the firewall.

You can use an API call or CLI command to specify custom ports for transferring files (22 is the default) and for exporting Syslog data (514 and 6514 are the defaults). If you do, you will need to configure the firewall accordingly.

Table 2-1. TCP and UDP Ports Used by NSX Manager

Source	Target	Port	Protocol	Description
Management Clients	NSX Manager	22	TCP	SSH (Disabled by default)
NTP Servers	NSX Manager	123	UDP	NTP
Management Clients	NSX Manager	443	TCP	NSX API server
SNMP Servers	NSX Manager	161	UDP	SNMP
NSX Controllers, NSX Edge nodes, Transport Nodes	NSX Manager	8080	TCP	Install-upgrade HTTP repository
NSX Controllers, NSX Edge nodes, Transport Nodes	NSX Manager	5671	TCP	NSX messaging
NSX Manager	Management SCP Servers	22	TCP	SSH (upload support bundle, backups, etc.)
NSX Manager	DNS Servers	53	TCP	DNS
NSX Manager	DNS Servers	53	UDP	DNS
NSX Manager	NTP Servers	123	UDP	NTP
NSX Manager	SNMP Servers	161, 162	TCP	SNMP
NSX Manager	SNMP Servers	161, 162	UDP	SNMP
NSX Manager	Syslog Servers	514	TCP	Syslog
NSX Manager	Syslog Servers	514	UDP	Syslog
NSX Manager	Syslog Servers	6514	TCP	Syslog

Table 2-1. TCP and UDP Ports Used by NSX Manager (Continued)

Source	Target	Port	Protocol	Description
NSX Manager	Syslog Servers	6514	UDP	Syslog
NSX Manager	LogInsight Server	9000	TCP	Log Insight agent
NSX Manager	Traceroute Destination	33434 - 33523	UDP	Traceroute
NSX Manager	vCenter Server	80	TCP	NSX Manager to compute manager (vCenter Server) communication, when configured.
NSX Manager	vCenter Server	443	TCP	NSX Manager to compute manager (vCenter Server) communication, when configured.

TCP and UDP Ports Used by NSX Controller

NSX Controller uses certain TCP and UDP ports to communicate with other components and products. These ports must be open in the firewall.

You can use an API call or CLI command to specify custom ports for transferring files (22 is the default) and for exporting Syslog data (514 and 6514 are the defaults). If you do, you will need to configure the firewall accordingly.

Table 2-2. TCP and UDP Ports Used by NSX Controller

Source	Target	Port	Protocol	Description
Management Clients	NSX Controller	22	TCP	SSH (Disabled by default)
DNS Servers	NSX Controller	53	UDP	DNS
NTP Servers	NSX Controller	123	UDP	NTP
SNMP Servers	NSX Controller	161	UDP	SNMP
NSX Controllers	NSX Controller	1100	TCP	Zookeeper quorum
NSX Controllers	NSX Controller	1200	TCP	Zookeeper leader election
NSX Controllers	NSX Controller	1300	TCP	Zookeeper server
NSX Edge nodes, Transport Nodes	NSX Controller	1234	TCP	CCP-netcpa communication
NSX Controllers	NSX Controller	7777	TCP	Moot RPC
NSX Controllers	NSX Controller	11000 - 11004	UDP	Tunnels to other cluster nodes. You must open more ports if the cluster has more than 5 nodes.
Traceroute Destination	NSX Controller	33434 - 33523	UDP	Traceroute

Table 2-2. TCP and UDP Ports Used by NSX Controller (Continued)

Source	Target	Port	Protocol	Description
NSX Controllers	SSH Destination	22	TCP	SSH (Disabled by default)
NSX Controllers	DNS Servers	53	UDP	DNS
NSX Controllers	DNS Servers	53	TCP	DNS
NSX Controllers	Any	80	TCP	HTTP
NSX Controllers	NTP Servers	123	UDP	NTP
NSX Controllers	NSX Manager	5671	TCP	NSX messaging
NSX Controllers	LogInsight Server	9000	TCP	Log Insight agent
NSX Controllers	NSX Controller	11000 - 11004	TCP	Tunnels to other cluster nodes. You must open more ports if the cluster has more than 5 nodes.
NSX Controllers	NSX Manager	8080	TCP	NSX upgrade
NSX Controllers	Traceroute Destination	33434 - 33523	UDP	Traceroute
NSX Controllers	Syslog Servers	514	UDP	Syslog
NSX Controllers	Syslog Servers	514	TCP	Syslog
NSX Controllers	Syslog Servers	6514	TCP	Syslog

TCP and UDP Ports Used by NSX Edge

NSX Edge uses certain TCP and UDP ports to communicate with other components and products. These ports must be open in the firewall.

You can use an API call or CLI command to specify custom ports for transferring files (22 is the default) and for exporting Syslog data (514 and 6514 are the defaults). If you do, you will need to configure the firewall accordingly.

Table 2-3. TCP and UDP Ports Used by NSX Edge

Source	Target	Port	Protocol	Description
Management Clients	NSX Edge	22	TCP	SSH (Disabled by default)
NTP Servers	NSX Edge	123	UDP	NTP
SNMP Servers	NSX Edge	161	UDP	SNMP
Tenant workload	NSX Edge	67, 68	UDP	DHCP service to hosted tenant workload
Any	NSX Edge	1167	TCP	DHCP backend
NSX Edge nodes, Transport Nodes	NSX Edge	3784, 3785	UDP	BFD between the Transport Node TEP IP address in the data.
NSX Agent	NSX Edge	5555	TCP	NSX Cloud - Agent on instance communicates to NSX Cloud Gateway.
NSX Edge nodes	NSX Edge	6666	TCP	NSX Cloud - NSX Edge local communication.
NSX Edge nodes	NSX Manager	8080	TCP	NAPI, NSX-T upgrade

Table 2-3. TCP and UDP Ports Used by NSX Edge (Continued)

Source	Target	Port	Protocol	Description
NSX Edge nodes	NSX Edge	2480	TCP	Nestdb
NSX Edge nodes	Management SCP or SSH Servers	22	TCP	SSH
NSX Edge nodes	DNS Servers	53	UDP	DNS
NSX Edge nodes	Any	80	TCP	HTTP
NSX Edge nodes	NTP Servers	123	UDP	NTP
NSX Edge nodes	SNMP Servers	161, 162	UDP	SNMP
NSX Edge nodes	SNMP Servers	161, 162	TCP	SNMP
NSX Edge nodes	External Router	179	TCP	BGP between T-0 logical router and the external physical router.
NSX Edge nodes	NSX Manager	443	TCP	HTTPS
NSX Edge nodes	Syslog Servers	514	TCP	Syslog
NSX Edge nodes	Syslog Servers	514	UDP	Syslog
NSX Edge nodes	Any	1167	TCP	DHCP backend
NSX Edge nodes	NSX Controllers	1234	TCP	netcpa
NSX Edge nodes	Any	3000 - 9000	TCP	Metadata proxy
NSX Edge nodes	NSX Manager	5671	TCP	NSX messaging
NSX Edge nodes	Syslog Servers	6514	TCP	Syslog over TLS
NSX Edge nodes	Traceroute Destination	33434 - 33523	UDP	Traceroute
NSX Edge nodes	NSX Edge nodes	50263	UDP	High-Availability

TCP Ports Used by Key Manager

Key Manager uses certain TCP ports to communicate with other components and products. These ports must be open in the firewall.

Table 2-4. TCP Ports Used by Key Manager

Source	Target	Port	Protocol	Description
Any	Key Manager	22	TCP	SSH
MP	Key Manager	8992	TCP	Management plane to Key Manager communication

Table 2-4. TCP Ports Used by Key Manager (Continued)

Source	Target	Port	Protocol	Description
Hypervisor	Key Manager	8443	TCP	Hypervisor to Key Manager communication
Key Manager	Any	22	TCP	SSH

Installation Checklist

Typically, for the initial installation, the order of procedures is as follows:

1 Install NSX Manager see, Chapter 4 NSX Manager Installation.

2 Install NSX Controllers see, Chapter 5 NSX Controller Installation and Clustering.

3 Join NSX Controllers with the management plane, see Join NSX Controllers with the NSX Manager.

4 Create a master NSX Controller to initialize the control cluster, see Initialize the Control Cluster to Create a Control Cluster Master.

5 Join NSX Controllers into a control cluster, see Join Additional NSX Controllers with the Cluster Master.

   NSX Manager installs NSX-T modules after the hypervisor hosts are added.

   Note Certificates are created on hypervisor hosts when NSX-T modules are installed.

6 Join hypervisor hosts with the management plane, see Join the Hypervisor Hosts with the Management Plane.

   The host sends its host certificate to the management plane.

7 Install NSX Edges, see Chapter 6 NSX Edge Installation.

8 Join NSX Edges with the management plane, see Join NSX Edge with the Management Plane.

9 Create transport zones and transport nodes, see Chapter 9 Transport Zones and Transport Nodes.

   A virtual switch is created on each host. The management plane sends the host certificates to the control plane, and the management plane pushes control plane information to the hosts. Each host connects to the control plane over SSL presenting its certificate. The control plane validates the certificate against the host certificate provided by the management plane. The controllers accept the connection upon successful validation.

The typical installation order.

NSX Manager is installed first.

NSX Controller can be installed and join the management plane.

NSX-T modules can be installed on a hypervisor host before it joins the management plane, or you can perform both procedures at the same time using the **Fabric > Hosts > Add** UI or the `POST fabric/nodes` API.

NSX Controller, NSX Edges, and hosts with NSX-T modules can join the management plane at any time.

Post-Installation

When the hosts are transport nodes, you can create transport zones, logical switches, logical routers, and other network components through the NSX Manager UI or API at any time. When NSX Controllers, NSX Edges, and hosts join the management plane, the NSX-T logical entities and configuration state are pushed to the NSX Controllers, NSX Edges, and hosts automatically.

For more information, see the *NSX-T Administration Guide*.

Working with KVM

3

NSX-T supports KVM in two ways: 1) as a host transport node and 2) as a host for NSX Manager and NSX Controller.

Table 3-1. Supported KVM Versions

Requirements	Description
Supported platforms	■ RHEL 7.4 and 7.3. ■ Ubuntu 16.04.2 LTS

This chapter includes the following topics:

Set Up KVM

If you plan to use KVM as a transport node or as a host for NSX Manager and NSX Controller guest VMs, but you do not already have KVM setup, you can use the procedure described here.

Note The Geneve encapsulation protocol uses UDP port 6081. You must allow this port access in the firewall on the KVM host.

Procedure

1 Open the `/etc/yum.conf` file.

2 Search for the line `exclude`.

3 Add the line "`kernel* redhat-release*`" to configure yum to avoid any unsupported RHEL upgrades.

```
exclude=[existing list] kernel* redhat-release*
```

The supported RHEL version is 7.4 and 7.3.

4 Install KVM and bridge utilities.

Linux Distribution	Commands
Ubuntu	`apt-get install -y qemu-kvm libvirt-bin ubuntu-vm-builder bridge-utils virtinst virt-manager virt-viewer libguestfs-tools`

RHEL	`yum groupinstall "Virtualization Hypervisor"`
	`yum groupinstall "Virtualization Client"`
	`yum groupinstall "Virtualization Platform"`
	`yum groupinstall "Virtualization Tools"`

5 Check the hardware virtualization capability.

```
cat /proc/cpuinfo | egrep "vmx|svm"
```

The output should contain vmx.

6 Verify that the KVM module is installed.

Linux Distribution	Commands

Ubuntu	`kvm-ok`
	
	`INFO: /dev/kvm exists`
	`KVM acceleration can be used`

RHEL	`lsmod | grep kvm`
	
	`kvm_intel  53484  6`
	`kvm  316506  1 kvm_intel`

7 For KVM to be used as a host for NSX Manager or NSX Controller, prepare the bridge network.

In the following example, the first Ethernet interface (eth0 or ens32) is used for connectivity to the Linux machine itself. Depending on your deployment environment, this interface can use DHCP or static IP settings.

Note Interface names might vary in different environments.

Linux Distribution	Network Configuration

Ubuntu	Edit `/etc/network/interfaces`:
	`auto lo`
	`iface lo inet loopback`
	
	`auto eth0`
	`iface eth0 inet manual`
	
	`auto br0`
	`iface br0 inet static`
	`address 192.168.110.51`
	`netmask 255.255.255.0`
	`network 192.168.110.0`
	`broadcast 192.168.110.255`
	`gateway 192.168.110.1`
	`dns-nameservers 192.168.3.45`
	`dns-search example.com`
	`bridge_ports eth0`
	`bridge_stp off`
	`bridge_fd 0`
	`bridge_maxwait 0`
	
	Create a network definition xml file for the bridge. For example, create `/tmp/bridge.xml` with the following lines:
	`<network>`
	`<name>bridge</name>`
	`<forward mode='bridge'/>`
	`<bridge name='br0'/>`
	`</network>`
	
	Define and start the bridge network with the following commands:
	`virsh net-define`
	`bridge.xml`
	`virsh net-start bridge`
	`virsh net-autostart bridge`

Linux Distribution	Network Configuration

	You can check the status of the bridge network with the following command:
	`virsh net-list --all`
	
	`Name State Autostart Persistent`
	`----------------------------------------------------------`
	`bridge active yes yes`
	`default active yes yes`

RHEL	Edit `/etc/sysconfig/network-scripts/ifcfg-`*`management_interface`*:
	`DEVICE="ens32"`
	`TYPE="Ethernet"`
	`NAME="ens32"`
	`UUID="<something>"`
	`BOOTPROTO="none"`
	`HWADDR="<something>"`
	`ONBOOT="yes"`
	`NM_CONTROLLED="no"`
	`BRIDGE="br0"`
	
	Edit `/etc/sysconfig/network-scripts/ifcfg-br0`:
	`DEVICE="br0"`
	`BOOTPROTO="dhcp"`
	`NM_CONTROLLED="no"`
	`ONBOOT="yes"`
	`TYPE="Bridge"`

8 For KVM to be used as a transport node, prepare the network bridge.

In the following example, the first Ethernet interface (eth0 or ens32) is used for connectivity to the Linux machine itself. Depending on your deployment environment, this interface can use DHCP or static IP settings.

Configure one more interface than in the previous step.

Note Interface names may vary in different environments.

Linux Distribution	Network Configuration

Ubuntu	Edit `/etc/network/interfaces`:
	`auto lo`
	`iface lo inet loopback`
	
	`auto eth0`
	`iface eth0 inet manual`
	
	`auto eth1`
	`iface eth1 inet manual`
	
	`auto br0`
	`iface br0 inet dhcp`
	`  bridge_ports eth0`

RHEL	Edit `/etc/sysconfig/network-scripts/ifcfg-ens32`:
	`DEVICE="ens32"`
	`  TYPE="Ethernet"`
	`  NAME="ens32"`
	`  UUID="<something>"`
	`  BOOTPROTO="none"`
	`  HWADDR="<something>"`
	`  ONBOOT="yes"`
	`  NM_CONTROLLED="no"`
	`  BRIDGE="br0"`
	
	Edit `/etc/sysconfig/network-scripts/ifcfg-ens33`:
	`DEVICE="ens33"`
	`  TYPE="Ethernet"`
	`  NAME="ens33"`
	`  UUID="<something>"`
	`  BOOTPROTO="none"`
	`  HWADDR="<something>"`
	`  ONBOOT="yes"`
	`  NM_CONTROLLED="no"`
	
	Edit `/etc/sysconfig/network-scripts/ifcfg-br0`:
	`DEVICE="br0"`
	`  BOOTPROTO="dhcp"`
	`  NM_CONTROLLED="no"`
	`  ONBOOT="yes"`
	`  TYPE="Bridge"`

Important For Ubuntu, all network configurations must be specified in `/etc/network/interfaces`. Do not create individual network configuration files such as `/etc/network/ifcfg-eth1`, which can lead to transport node creation failure.

Once the KVM host is configured as a transport node, the bridge interface "nsx-vtep0.0" is created. In Ubuntu, /etc/network/interfaces has entries such as the following:

```
iface nsx-vtep0.0 inet static
pre-up ip addr flush dev nsx-vtep0.0
address <IP_pool_address>
netmask <subnet_mask>
mtu 1600
down ifconfig nsx-vtep0.0 down
up ifconfig nsx-vtep0.0 up
```

In RHEL, nsxa creates a configuration file called ifcfg-nsx-vtep0.0, which has entries such as the following:

```
DEVICE=nsx-vtep0.0
BOOTPROTO=static
NETMASK=<IP address>
IPADDR=<subnet mask>
MTU=1600
ONBOOT=yes
USERCTL=no
NM_CONTROLLED=no
```

9 To make the networking changes take effect, restart networking service `systemctl restart network` or reboot the Linux server.

Manage Your Guest VMs in the KVM CLI

NSX Manager and NSX Controller can be installed as KVM VMs. In addition, KVM can be used as the hypervisor for NSX-T transport nodes.

KVM guest VM management is beyond the scope of this guide. However, here are some simple KVM CLI commands to get your started.

To manage your guest VMs in the KVM CLI, you can use `virsh` commands. Following are some common `virsh` commands. Refer to KVM documentation for additional information.

```
# List running
virsh list

# List all
virsh list --all

# Control instances
virsh start <instance>
virsh shutdown <instance>
virsh destroy <instance>
virsh undefine <instance>
virsh suspend <instance>
```

```
virsh resume <instance>

# Access an instance's CLI
virsh console <instance>
```

In the Linux CLI, the `ifconfig` command shows the vnetX interface, which represents the interface created for the guest VM. If you add additional guest VMs, additional vnetX interfaces are added.

```
ifconfig
...

vnet0     Link encap:Ethernet  HWaddr fe:54:00:b0:a0:6d
          inet6 addr: fe80::fc54:ff:feb0:a06d/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:13183 errors:0 dropped:0 overruns:0 frame:0
          TX packets:181524 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:500
          RX bytes:4984832 (4.9 MB)  TX bytes:29498709 (29.4 MB)
```

NSX Manager Installation

4

NSX Manager provides a graphical user interface (GUI) and REST APIs for creating, configuring, and monitoring NSX-T components such as logical switches, logical routers, and firewalls.

NSX Manager provides a system view and is the management component of NSX-T.

You can install only one instance of NSX Manager.

Table 4-1. NSX Manager Deployment, Platform, and Installation Requirements

Requirements	Description
Supported deployment methods	■ OVA/OVF ■ QCOW2
Supported platforms	■ Supported vSphere version ■ RHEL 7.4 and 7.3. ■ Ubuntu 16.04.2 LTS You can use the vSphere high availability (HA) feature to ensure the availability of NSX Manager when NSX Manager is deployed on ESXi. On ESXi, it is recommended that the NSX Manager appliance be installed on shared storage. vSphere HA requires shared storage so that VMs can be restarted on another host if the original host fails.
IP address	An NSX Manager must have a static IP address. You cannot change the IP address after installation.
NSX-T appliance password	■ At least eight characters ■ At least one lower-case letter ■ At least one upper-case letter ■ At least one digit ■ At least one special character ■ At least five different characters ■ No dictionary words ■ No palindromes

Table 4-1. NSX Manager Deployment, Platform, and Installation Requirements (Continued)

Requirements	Description
Hostname	When installing NSX Manager, specify a hostname that does not contain invalid characters such as an underscore. If the hostname contains any invalid character, after deployment the hostname will be set to **nsx-manager**. For more information about hostname restrictions, see https://tools.ietf.org/html/rfc952 and https://tools.ietf.org/html/rfc1123.
VMware Tools	The NSX Manager VM running on ESXi has VMTools installed. Do not remove or upgrade VMTools.

Note On an NSX Manager fresh install, reboot, or after an **admin** password change when prompted on first login, it might take several minutes for the NSX Manager to start.

NSX Manager Installation Scenarios

Important When you install NSX Manager from an OVA or OVF file, either from vSphere Web Client or the command line, OVA/OVF property values such as user names, passwords, or IP addresses are not validated before the VM is powered on.

- If you specify a user name for the **admin** or **audit** user, the name must be unique. If you specify the same name, it is ignored and the default names (**admin** and **audit**) is used.
- If the password for the **admin** user does not meet the complexity requirements, you must log in to NSX Manager through SSH or at the console as the **admin** user. You are prompted to change the password.
- If the password for the **audit** user does not meet the complexity requirements, the user account is disabled. To enable the account, log in to NSX Manager through SSH or at the console as the **admin** user and run the command **set user audit** to set the **audit** user's password (the current password is an empty string).
- If the password for the **root** user does not meet the complexity requirements, you must log in to NSX Manager through SSH or at the console as **root** with the password **vmware**. You are prompted to change the password.

Note The core services on the appliance do not start until a password with sufficient complexity is set.

After you deploy NSX Manager from an OVA file, you cannot change the VM's IP settings by powering off the VM and modifying the OVA settings from vCenter Server.

This chapter includes the following topics:

Install NSX Manager on ESXi Using vSphere Web Client

You can use vSphere Web Client to deploy NSX Manager as a virtual appliance. You can also configure the NSX Manager installation to install the NSX Policy Manager.

The NSX Policy Manager is a virtual appliance that lets you manage NSX policies. You can configure NSX policies to specify rules for NSX-T components such as logical ports, IP addresses, and VMs. NSX policy rules allow you to set high-level usage and resource access rules that are enforced without specifying the exact details.

Note It is recommended that you use vSphere Web Client instead of vSphere Client. If you do not have vCenter Server in your environment, use `ovftool` to deploy NSX Manager. See Install NSX Manager on ESXi Using the Command-Line OVF Tool.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.

Procedure

1. Locate the NSX-T Unified Appliance OVA or OVF file.

   Either copy the download URL or download the OVA file onto your computer.
2. In vSphere Web Client, launch the **Deploy OVF template** wizard and navigate or link to the .ova file.
3. Enter a name for the NSX Manager, and select a folder or datacenter.

   The name you type appears in the inventory.

   The folder you select will be used to apply permissions to the NSX Manager.
4. Select a datastore to store the NSX Manager virtual appliance files.
5. If you are installing in vCenter, select a host or cluster on which to deploy the NSX Manager appliance.

   Normally, you would place the NSX Manager in a cluster that provides network management utilities.
6. Select the port group or destination network for the NSX Manager.
7. Specify the NSX Manager passwords and IP settings.

8 Type the **nsx-manager** role.

You can also type **nsx-policy-manager** to install the NSX Policy Manager.

9 (Optional) For optimal performance, reserve memory for the NSX-T component.

A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

10 Open the console of the NSX-T component to track the boot process.

11 After the NSX-T component boots, log in to the CLI as admin and run the `get interface eth0` command to verify that the IP address was applied as expected.

```
nsx-component> get interface eth0
Interface: eth0
  Address: 192.168.110.25/24
  MAC address: 00:50:56:86:7b:1b
  MTU: 1500
  Default gateway: 192.168.110.1
  Broadcast address: 192.168.110.255
  ...
```

12 Verify that your NSX-T component has the required connectivity.

Make sure that you can perform the following tasks.

- Ping your NSX-T component from another machine.
- The NSX-T component can ping its default gateway.
- The NSX-T component can ping the hypervisor hosts that are in the same network as the NSX-T component using the management interface.
- The NSX-T component can ping its DNS server and its NTP server.
- If you enabled SSH, make sure that you can SSH to your NSX-T component.

If connectivity is not established, make sure the network adapter of the virtual appliance is in the proper network or VLAN.

What to do next

Connect to the NSX Manager GUI by from a supported web browser.

The URL is https://<IP address of NSX Manager>. For example, https://10.16.176.10.

Note You must use HTTPS. HTTP is not supported.

Install NSX Manager on ESXi Using the Command-Line OVF Tool

If you prefer to automate or use CLI for the NSX Manager installation, you can use the VMware OVF Tool, which is a command-line utility.

By default, nsx_isSSHEnabled and nsx_allowSSHRootLogin are both disabled for security reasons. When they are disabled, you cannot SSH or log in to the NSX Manager command line. If you enable nsx_isSSHEnabled but not nsx_allowSSHRootLogin, you can SSH to NSX Manager but you cannot log in as root.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.

Procedure

- For a standalone host, run the `ovftool` command with the appropriate parameters.

```
C:\Users\Administrator\Downloads>ovftool
--name=nsx-manager
--X:injectOvfEnv
--X:logFile=ovftool.log
 --allowExtraConfig
--datastore=ds1
--net="management"
--acceptAllEulas
--noSSLVerify
--diskMode=thin
--powerOn
--prop:nsx_role=nsx-manager
--prop:nsx_ip_0=192.168.110.75
--prop:nsx_netmask_0=255.255.255.0
--prop:nsx_gateway_0=192.168.110.1
--prop:nsx_dns1_0=192.168.110.10
--prop:nsx_domain_0=corp.local
--prop:nsx_ntp_0=192.168.110.10
--prop:nsx_isSSHEnabled=<True|False>
--prop:nsx_allowSSHRootLogin=<True|False>
--prop:nsx_passwd_0=<password>
```

```
--prop:nsx_cli_passwd_0=<password>
--prop:nsx_hostname=nsx-manager
nsx-<component>.ova
vi://root:<password>@192.168.110.51
```

```
Opening OVA source: nsx-<component>.ova
The manifest validates
Source is signed and the certificate validates
Opening VI target: vi://root:<password>@192.168.110.51
Deploying to VI: vi://root:<password>@192.168.110.51
Transfer Completed
Powering on VM: nsx-manager
Task Completed
Completed successfully
```

- For a host managed by vCenter Server, run the `ovftool` command with the appropriate parameters. For example,

```
C:\Users\Administrator\Downloads>ovftool
--name=nsx-manager
--X:injectOvfEnv
--X:logFile=ovftool.log
 --allowExtraConfig
--datastore=ds1
--network="management"
--acceptAllEulas
--noSSLVerify
--diskMode=thin
--powerOn
--prop:nsx_role=nsx-manager
--prop:nsx_ip_0=192.168.110.75
--prop:nsx_netmask_0=255.255.255.0
--prop:nsx_gateway_0=192.168.110.1
--prop:nsx_dns1_0=192.168.110.10
--prop:nsx_domain_0=corp.local
--prop:nsx_ntp_0=192.168.110.10
--prop:nsx_isSSHEnabled=<True|False>
--prop:nsx_allowSSHRootLogin=<True|False>
--prop:nsx_passwd_0=<password>
--prop:nsx_cli_passwd_0=<password>
--prop:nsx_hostname=nsx-manager
nsx-<component>.ova
vi://administrator@vsphere.local:<password>@192.168.110.24/?ip=192.168.110.51
```

```
Opening OVA source: nsx-<component>.ova
The manifest validates
Source is signed and the certificate validates
Opening VI target: vi://administrator@vsphere.local@192.168.110.24:443/
Deploying to VI: vi://administrator@vsphere.local@192.168.110.24:443/
```

```
Transfer Completed
Powering on VM: nsx-manager
Task Completed
Completed successfully
```

- (Optional) For optimal performance, reserve memory for the NSX-T component.

  A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

- Open the console of the NSX-T component to track the boot process.
- After the NSX-T component boots, log in to the CLI as admin and run the `get interface eth0` command to verify that the IP address was applied as expected.

```
nsx-component> get interface eth0
Interface: eth0
  Address: 192.168.110.25/24
  MAC address: 00:50:56:86:7b:1b
  MTU: 1500
  Default gateway: 192.168.110.1
  Broadcast address: 192.168.110.255
  ...
```

- Verify that your NSX-T component has the required connectivity.

  Make sure that you can perform the following tasks.

  - Ping your NSX-T component from another machine.
  - The NSX-T component can ping its default gateway.
  - The NSX-T component can ping the hypervisor hosts that are in the same network as the NSX-T component using the management interface.
  - The NSX-T component can ping its DNS server and its NTP server.
  - If you enabled SSH, make sure that you can SSH to your NSX-T component.

  If connectivity is not established, make sure the network adapter of the virtual appliance is in the proper network or VLAN.

What to do next

Connect to the NSX Manager GUI by from a supported web browser.

The URL is https://<IP address of NSX Manager>. For example, https://10.16.176.10.

Note You must use HTTPS. HTTP is not supported.

Install NSX Manager on KVM

NSX Manager can be installed as a virtual appliance on a KVM host.

The QCOW2 installation procedure uses guestfish, a Linux command-line tool to write virtual machine settings into the QCOW2 file.

Prerequisites

- KVM set up. See Set Up KVM.
- Privileges to deploy a QCOW2 image on the KVM host.
- Verify that the password in the guestinfo adheres to the password complexity requirements so that you can log in after installation. See Chapter 4 NSX Manager Installation.
- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- It is recommended to place NSX-T appliances on a management VM network.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.

Procedure

1. Download the NSX Manager QCOW2 image and then copy it to the KVM machine that runs the NSX Manager using SCP or sync.
2. (Ubuntu only) Add the currently logged in user as a libvirtd user:

   ```
   adduser $USER libvirtd
   ```

3. In the same directory where you saved the QCOW2 image, create a file called guestinfo (with no file extension) and populate it with the NSX Manager VM's properties.

   For example:

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <Environment
        xmlns="http://schemas.dmtf.org/ovf/environment/1"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:oe="http://schemas.dmtf.org/ovf/environment/1">
      <PropertySection>
            <Property oe:key="nsx_role" oe:value="nsx-manager"/>
            <Property oe:key="nsx_allowSSHRootLogin" oe:value="True"/>
            <Property oe:key="nsx_cli_passwd_0" oe:value="<password>"/>
            <Property oe:key="nsx_dns1_0" oe:value="192.168.110.10"/>
            <Property oe:key="nsx_domain_0" oe:value="corp.local"/>
            <Property oe:key="nsx_gateway_0" oe:value="192.168.110.1"/>
            <Property oe:key="nsx_hostname" oe:value="nsx-manager1"/>
            <Property oe:key="nsx_ip_0" oe:value="192.168.110.19"/>
            <Property oe:key="nsx_isSSHEnabled" oe:value="True"/>
            <Property oe:key="nsx_netmask_0" oe:value="255.255.255.0"/>
            <Property oe:key="nsx_ntp_0" oe:value="192.168.110.10"/>
   ```

```
          <Property oe:key="nsx_passwd_0" oe:value="<password>"/>
   </PropertySection>
</Environment>
```

In the example, nsx_isSSHEnabled and nsx_allowSSHRootLogin are both enabled. When they are disabled, you cannot SSH or log in to the NSX Manager command line. If you enable nsx_isSSHEnabled but not nsx_allowSSHRootLogin, you can SSH to NSX Manager but you cannot log in as root.

4 Use guestfish to write the `guestinfo` file into the QCOW2 image.

After the `guestinfo` information is written into a QCOW2 image, the information cannot be overwritten.

```
sudo guestfish --rw -i -a nsx-manager1-build.qcow2 upload guestinfo /config/guestinfo
```

5 Deploy the QCOW2 image with the `virt-install` command.

```
user@ubuntu1604:/var/lib/libvirt/images$ sudo virt-install --import --name nsx-manager1 --ram
16348 --vcpus 4 --network=bridge:br0,model=e1000 --disk path=/var/lib/libvirt/images/nsx-
manager-1.1.0.0.0.4446302.qcow2,format=qcow2 --nographics

Starting install...
Creating domain...       |    0 B     00:01
Connected to domain nsx-manager1
Escape character is ^]

nsx-manager1 login:
```

After the NSX Manager boots up, the NSX Manager console appears.

6 (Optional) For optimal performance, reserve memory for the NSX-T component.

A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

7 Open the console of the NSX-T component to track the boot process.

8 After the NSX-T component boots, log in to the CLI as admin and run the `get interface eth0` command to verify that the IP address was applied as expected.

```
nsx-component> get interface eth0
Interface: eth0
  Address: 192.168.110.25/24
  MAC address: 00:50:56:86:7b:1b
```

```
MTU: 1500
Default gateway: 192.168.110.1
Broadcast address: 192.168.110.255
...
```

9 Verify that your NSX-T component has the required connectivity.

Make sure that you can perform the following tasks.

- Ping your NSX-T component from another machine.
- The NSX-T component can ping its default gateway.
- The NSX-T component can ping the hypervisor hosts that are in the same network as the NSX-T component using the management interface.
- The NSX-T component can ping its DNS server and its NTP server.
- If you enabled SSH, make sure that you can SSH to your NSX-T component.

If connectivity is not established, make sure the network adapter of the virtual appliance is in the proper network or VLAN.

10 Exit the KVM console.

```
control-]
```

What to do next

Connect to the NSX Manager GUI by from a supported web browser.

The URL is https://<IP address of NSX Manager>. For example, https://10.16.176.10.

Note You must use HTTPS. HTTP is not supported.

NSX Controller Installation and Clustering

5

NSX Controller is an advanced distributed state management system that provides control plane functions for NSX-T logical switching and routing functions.

NSX Controllers serve as the central control point for all logical switches within a network and maintains information about all hosts, logical switches, and logical routers. NSX Controllers control the devices that perform packet forwarding. These forwarding devices are known as virtual switches.

Virtual switches, such as NSX managed virtual distributed switch (N-VDS, previously known as hostswitch) and Open vSwitch (OVS), reside on ESXi and other hypervisors such as KVM.

In a production environment, you must have an NSX Controller cluster with three members to avoid any outage to the NSX control plane. Each controller should be placed on a unique hypervisor host, three physical hypervisor hosts in total, to avoid a single physical hypervisor host failure impacting the NSX control plane. For lab and proof-of-concept deployments where there are no production workloads, it is acceptable to run a single controller to save resources.

Table 5-1. NSX Controller Deployment, Platform, and Installation Requirements

Requirements	Description
Supported deployment methods	■ OVA/OVF ■ QCOW2
Supported platforms	■ Supported vSphere version ■ RHEL 7.4 and 7.3. ■ Ubuntu 16.04.2 LTS NSX Controller is supported on ESXi as a VM and KVM. **Note** Installation via PXE boot is not supported.
IP address	An NSX Controller must have a static IP address. You cannot change the IP address after installation.
NSX-T appliance password	■ At least eight characters ■ At least one lower-case letter ■ At least one upper-case letter ■ At least one digit ■ At least one special character ■ At least five different characters ■ No dictionary words ■ No palindromes

Table 5-1. NSX Controller Deployment, Platform, and Installation Requirements (Continued)

Requirements	Description
Hostname	When installing NSX Controller, specify a hostname that does not contain invalid characters such as an underscore. If the hostname contains any invalid character, after deployment the hostname will be set to **localhost**. For more information about hostname restrictions, see https://tools.ietf.org/html/rfc952 and https://tools.ietf.org/html/rfc1123.
VMware Tools	The NSX Controller VM running on ESXi has VMTools installed. Do not remove or upgrade VMTools.

NSX Controller Installation Scenarios

Important When you install NSX Controller from an OVA or OVF file, either from vSphere Web Client or the command line, OVA/OVF property values such as user names, passwords, or IP addresses are not validated before the VM is powered on.

- If you specify a user name for the **admin** or **audit** user, the name must be unique. If you specify the same name, it is ignored and the default names (**admin** and **audit**) are used.
- If the password for the **admin** user does not meet the complexity requirements, you must log in to NSX Controller through SSH or at the console as the **admin** user. You are prompted to change the password.
- If the password for the **audit** user does not meet the complexity requirements, the user account is disabled. To enable the account, log in to NSX Controller through SSH or at the console as the **admin** user and run the command **set user audit** to set the **audit** user's password (the current password is an empty string).
- If the password for the **root** user does not meet the complexity requirements, you must log in to NSX Controller through SSH or at the console as **root** with the password **vmware**. You are prompted to change the password.
- Do not use root privileges to install daemons or applications. Using the root privileges to install daemons or applications can void your support contract. Use root privileges only when requested by the VMware Support team.

Note The core services on the appliance do not start until a password with sufficient complexity has been set.

After you deploy an NSX Controller from an OVA file, you cannot change the VM's IP settings by powering off the VM and modifying the OVA settings from vCenter Server.

This chapter includes the following topics:

- Install NSX Controller on KVM
- Join NSX Controllers with the NSX Manager
- Initialize the Control Cluster to Create a Control Cluster Master
- Join Additional NSX Controllers with the Cluster Master

Install NSX Controller on ESXi Using a GUI

If you prefer an interactive NSX Controller installation, you can use a UI-based VM management tool, such as the vSphere Client connected to vCenter Server.

To support backup and restore, the NSX Controller appliances must have static management IP addresses. Using DHCP to assign management IP addresses is not supported. Changing management IP addresses is not supported. See the *NSX-T Administration Guide* for backup and restore information.

Your passwords must comply with the password strength restrictions. NSX-T appliances enforce the following complexity rules:

- At least eight characters
- At least one lower-case letter
- At least one upper-case letter
- At least one digit
- At least one special character
- At least five different characters
- No dictionary words
- No palindromes

The installation succeeds if the password does not meet the requirements. However, when you log in for the first time, you are prompted to change the password.

Important The core services on the appliance do not start until a password with sufficient complexity has been set.

Important The NSX-T component virtual machine installations include VMware Tools. Removal or upgrade of VMware Tools is not supported for NSX-T appliances.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.

- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- Verify that you have adequate privileges to deploy an OVF template on the ESXi host.
- Verify that hostnames do not include underscores. Otherwise, the hostname is set to *nsx-controller*.
- A management tool that can deploy OVF templates, such as vCenter Server or the vSphere Client.

  The OVF deployment tool must support configuration options to allow for manual configuration.
- The Client Integration Plug-in must be installed.

Procedure

1. Locate the NSX Controller OVA or OVF file.

   Either copy the download URL or download the OVA file onto your computer.
2. In the management tool, launch the **Deploy OVF template** wizard and navigate or link to the .ova file.
3. Enter a name for the NSX Controller, and select a folder or datacenter.

   The name you type will appear in the inventory.

   The folder you select will be used to apply permissions to the NSX Controller.
4. Select a datastore to store the NSX Controller virtual appliance files.
5. If you are using vCenter, select a host or cluster on which to deploy the NSX Controller appliance.

   Normally, you would place the NSX Controller in a cluster that provides network management utilities.
6. Select the port group or destination network for the NSX Controller.
7. Specify the NSX Controller passwords and IP settings.
8. (Optional) For optimal performance, reserve memory for the NSX-T component.

   A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.
9. Open the console of the NSX-T component to track the boot process.
10. After the NSX-T component boots, log in to the CLI as admin and run the `get interface eth0` command to verify that the IP address was applied as expected.

```
nsx-component> get interface eth0
Interface: eth0
  Address: 192.168.110.25/24
  MAC address: 00:50:56:86:7b:1b
  MTU: 1500
  Default gateway: 192.168.110.1
  Broadcast address: 192.168.110.255
  ...
```

11 Verify that your NSX-T component has the required connectivity.

   Make sure that you can perform the following tasks.

   - Ping your NSX-T component from another machine.
   - The NSX-T component can ping its default gateway.
   - The NSX-T component can ping the hypervisor hosts that are in the same network as the NSX-T component using the management interface.
   - The NSX-T component can ping its DNS server and its NTP server.
   - If you enabled SSH, make sure that you can SSH to your NSX-T component.

   If connectivity is not established, make sure the network adapter of the virtual appliance is in the proper network or VLAN.

What to do next

Join the NSX Controller with the management plane. See Join NSX Controllers with the NSX Manager.

Install NSX Controller on ESXi Using the Command-Line OVF Tool

If you prefer to automate NSX Controller installation, you can use the VMware OVF Tool, which is a command-line utility.

By default, nsx_isSSHEnabled and nsx_allowSSHRootLogin are both disabled for security reasons. When they are disabled, you cannot SSH or log in to the NSX Controller command line. If you enable nsx_isSSHEnabled but not nsx_allowSSHRootLogin, you can SSH to NSX Controller but you cannot log in as root.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- OVF Tool version 4.0 or later.

Procedure

- For a standalone host, run the `ovftool` command with the appropriate parameters.

```
C:\Users\Administrator\Downloads>ovftool
--name=nsx-controller
--X:injectOvfEnv
--X:logFile=ovftool.log
--allowExtraConfig
--datastore=ds1
--network="management"
--noSSLVerify
--diskMode=thin
--powerOn
--prop:nsx_ip_0=192.168.110.210
--prop:nsx_netmask_0=255.255.255.0
--prop:nsx_gateway_0=192.168.110.1
--prop:nsx_dns1_0=192.168.110.10
--prop:nsx_domain_0=corp.local
--prop:nsx_ntp_0=192.168.110.10
--prop:nsx_isSSHEnabled=<True|False>
--prop:nsx_allowSSHRootLogin=<True|False>
--prop:nsx_passwd_0=<password>
--prop:nsx_cli_passwd_0=<password>
--prop:nsx_cli_audit_passwd_0=<password>
--prop:nsx_hostname=nsx-controller
<path/url to nsx component ova>
vi://root:<password>@192.168.110.51
```

- For a host managed by vCenter Server, run the `ovftool` command with the appropriate parameters.

```
C:\Users\Administrator\Downloads>ovftool
--name=nsx-controller
--X:injectOvfEnv
--X:logFile=ovftool.log
--allowExtraConfig
--datastore=ds1
--network="management"
--noSSLVerify
--diskMode=thin
--powerOn
--prop:nsx_ip_0=192.168.110.210
--prop:nsx_netmask_0=255.255.255.0
--prop:nsx_gateway_0=192.168.110.1
--prop:nsx_dns1_0=192.168.110.10
--prop:nsx_domain_0=corp.local
--prop:nsx_ntp_0=192.168.110.10
--prop:nsx_isSSHEnabled=<True|False>
--prop:nsx_allowSSHRootLogin=<True|False>
--prop:nsx_passwd_0=<password>
--prop:nsx_cli_passwd_0=<password>
```

```
--prop:nsx_cli_audit_passwd_0=<password>
--prop:nsx_hostname=nsx-controller
<path/url to nsx component ova>
vi://administrator@vsphere.local:<vcenter_password>@192.168.110.24/?ip=192.168.110.51
```

- (Optional) For optimal performance, reserve memory for the NSX-T component.

  A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

- Open the console of the NSX-T component to track the boot process.
- After the NSX-T component boots, log in to the CLI as admin and run the `get interface eth0` command to verify that the IP address was applied as expected.

  ```
  nsx-component> get interface eth0
  Interface: eth0
    Address: 192.168.110.25/24
    MAC address: 00:50:56:86:7b:1b
    MTU: 1500
    Default gateway: 192.168.110.1
    Broadcast address: 192.168.110.255
    ...
  ```

- Verify that your NSX-T component has the required connectivity.

  Make sure that you can perform the following tasks.

  - Ping your NSX-T component from another machine.
  - The NSX-T component can ping its default gateway.
  - The NSX-T component can ping the hypervisor hosts that are in the same network as the NSX-T component using the management interface.
  - The NSX-T component can ping its DNS server and its NTP server.
  - If you enabled SSH, make sure that you can SSH to your NSX-T component.

  If connectivity is not established, make sure the network adapter of the virtual appliance is in the proper network or VLAN.

What to do next

Join the NSX Controller with the management plane. See Join NSX Controllers with the NSX Manager.

Install NSX Controller on KVM

NSX Controller serves as the central control point for all logical switches within a network and maintains information about all hosts, logical switches, and distributed logical routers.

The QCOW2 installation procedure uses guestfish, a Linux command-line tool to write virtual machine settings into the QCOW2 file.

Prerequisites

- KVM set up. See Set Up KVM.
- Privileges to deploy a QCOW2 image on the KVM host.
- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.

Procedure

1. Download the NSX Controller QCOW2 image to the `/var/lib/libvirt/images` directory.
2. (Ubuntu only) Add the currently logged in user as a libvirtd user:

   ```
   adduser $USER libvirtd
   ```

3. In the same directory where you saved the QCOW2 image, create a file called `guestinfo` (with no file extension) and populate it with the NSX Controller VM's properties.

   For example:

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <Environment
        xmlns="http://schemas.dmtf.org/ovf/environment/1"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:oe="http://schemas.dmtf.org/ovf/environment/1">
      <PropertySection>
            <Property oe:key="nsx_allowSSHRootLogin" oe:value="True"/>
                                    <Property oe:key="nsx_cli_audit_passwd_0" oe:value="<password>"/>
            <Property oe:key="nsx_cli_passwd_0" oe:value="<password>"/>
            <Property oe:key="nsx_dns1_0" oe:value="192.168.110.10"/>
            <Property oe:key="nsx_domain_0" oe:value="corp.local"/>
            <Property oe:key="nsx_gateway_0" oe:value="192.168.110.1"/>
            <Property oe:key="nsx_hostname" oe:value="nsx-Controller1"/>
            <Property oe:key="nsx_ip_0" oe:value="192.168.110.34"/>
            <Property oe:key="nsx_isSSHEnabled" oe:value="True"/>
            <Property oe:key="nsx_netmask_0" oe:value="255.255.255.0"/>
            <Property oe:key="nsx_ntp_0" oe:value="192.168.110.10"/>
            <Property oe:key="nsx_passwd_0" oe:value="<password>"/>
      </PropertySection>
   </Environment>
   ```

In the example, nsx_isSSHEnabled and nsx_allowSSHRootLogin are both enabled. When they are disabled, you cannot SSH or log in to the NSX Controller command line. If you enable nsx_isSSHEnabled but not nsx_allowSSHRootLogin, you can SSH to NSX Controller but you cannot log in as root.

4 Use guestfish to write the `guestinfo` file into the QCOW2 image.

If you are making multiple NSX Controllers, make a separate copy of the QCOW2 image for each controller. After the `guestinfo` information is written into a QCOW2 image, the information cannot be overwritten.

```
sudo guestfish --rw -i -a nsx-controller1-build.qcow2 upload guestinfo /config/guestinfo
```

5 Deploy the QCOW2 image with the `virt-install` command.

```
user@ubuntu1604:/var/lib/libvirt/images$ sudo virt-install --import --name nsx-controller1 --ram
16384 --vcpus 2 --network=bridge:br0,model=e1000 --disk path=/var/lib/libvirt/images/nsx-
controller-release_version_number.qcow2,format=qcow2 --nographics --noautoconsole
```

After the NSX Controller boots up, the NSX Controller console appears.

6 (Optional) For optimal performance, reserve memory for the NSX-T component.

A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

7 Open the console of the NSX-T component to track the boot process.

8 After the NSX-T component boots, log in to the CLI as admin and run the `get interface eth0` command to verify that the IP address was applied as expected.

```
nsx-component> get interface eth0
Interface: eth0
  Address: 192.168.110.25/24
  MAC address: 00:50:56:86:7b:1b
  MTU: 1500
  Default gateway: 192.168.110.1
  Broadcast address: 192.168.110.255
  ...
```

9 Verify that your NSX-T component has the required connectivity.

Make sure that you can perform the following tasks.

- Ping your NSX-T component from another machine.
- The NSX-T component can ping its default gateway.
- The NSX-T component can ping the hypervisor hosts that are in the same network as the NSX-T component using the management interface.

- The NSX-T component can ping its DNS server and its NTP server.
- If you enabled SSH, make sure that you can SSH to your NSX-T component.

If connectivity is not established, make sure the network adapter of the virtual appliance is in the proper network or VLAN.

What to do next

Join the NSX Controller with the management plane. See Join NSX Controllers with the NSX Manager.

Join NSX Controller s with the NSX Manager

Joining NSX Controllers with the NSX Manager ensures that the NSX Manager and NSX Controllers can communicate with each other.

Prerequisites

- Verify that NSX Manager is installed.
- Verify that you have admin privileges to log in to the NSX Manager and NSX Controller appliances.

Procedure

1. Open an SSH session to NSX Manager.
2. Open an SSH session to each of the NSX Controller appliances.

   For example, NSX-Controller1, NSX-Controller2, NSX-Controller3.
3. On NSX Manager, run the `get certificate api thumbprint` command.

   ```
   NSX-Manager> get certificate api thumbprint
   ...
   ```
4. On each of the NSX Controller appliances, run the **join management-plane** command.

   ```
   NSX-Controller1> join management-plane NSX-Manager-IP-address username admin thumbprint <NSX-
   Manager-thumbprint>

   Password for API user: <NSX-Manager-password>
   Node successfully registered and controller restarted
   ```

   Run this command on each deployed NSX Controller node.

   Provide the following information:

   - IP address of the NSX Manager with an optional port number
   - Username of the NSX Manager
   - Certificate thumbprint of the NSX Manager
   - Password of the NSX Manager

5 Verify the result by running the `get managers` command on your NSX Controllers.

```
NSX-Controller1> get managers
- 192.168.110.47   Connected
```

6 On the NSX Manager appliance, run the `get management-cluster status` command and make sure the NSX Controllers are listed.

```
NSX-Manager> get management-cluster status
Number of nodes in management cluster: 1
- 192.168.110.47 (UUID 45A8869B-BB90-495D-8A01-69B5FCC56086) Online

Management cluster status: STABLE

Number of nodes in control cluster: 3
- 192.168.110.201 (UUID 45A8869B-BB90-495D-8A01-69B5FCC56086)
- 192.168.110.202 (UUID 45A8869B-BB90-495D-8A01-69B5FCC56086)
- 192.168.110.203 (UUID 45A8869B-BB90-495D-8A01-69B5FCC56086)
```

What to do next

Initialize the control cluster. See Initialize the Control Cluster to Create a Control Cluster Master.

Initialize the Control Cluster to Create a Control Cluster Master

After installing the first NSX Controller in your NSX-T deployment, you can initialize the control cluster. Initializing the control cluster is required even if you are setting up a small proof-of-concept environment with only one controller node. If you do not initialize the control cluster, the controller is not able to communicate with the hypervisor hosts.

Prerequisites

- Install at least one NSX Controller.
- Join the NSX Controller with the management plane.
- Verify that you have admin privileges to log in to the NSX Controller appliance.
- Assign a shared secret password. A shared secret password is a user-defined shared secret password (for example, "secret123").

Procedure

1 Open an SSH session for your NSX Controller.

2 Run the `set control-cluster security-model shared-secret secret <secret>` command and type a shared secret when prompted.

3 Run the `initialize control-cluster` command.

This command makes this controller the control cluster master.

For example:

```
NSX-Controller1> initialize control-cluster
Control cluster initialization successful.
```

4 Run the `get control-cluster status verbose` command.

Verify that `is master` and `in majority` are `true`, the status is `active`, and the `Zookeeper Server IP` is `reachable, ok`.

```
nsx-controller1> get control-cluster status verbose
NSX Controller Status:

uuid: 78d5b561-4f66-488d-9e53-089735eac1c1
is master: true
in majority: true
uuid                                   address              status
78d5b561-4f66-488d-9e53-089735eac1c1 192.168.110.34       active

Cluster Management Server Status:

uuid                                   rpc address          rpc port             global
id            vpn address          status
557a911f-41fd-4977-9c58-f3ef55b3efe7 192.168.110.34       7777
1                    169.254.1.1             connected

Zookeeper Ensemble Status:

Zookeeper Server IP: 10.0.0.1, reachable, ok
Zookeeper version: 3.5.1-alpha--1, built on 03/08/2016 01:18 GMT
Latency min/avg/max: 0/0/1841
Received: 212095
Sent: 212125
Connections: 5
Outstanding: 0
Zxid: 0x10000017a
Mode: leader
Node count: 33
Connections: /10.0.0.1:51726[1]
(queued=0,recved=60324,sent=60324,sid=0x100000f14a10003,lop=PING,est=1459376913497,to=30000,lcxid=0
x8,lzxid=0x10000017a,lresp=604617273,llat=0,minlat=0,avglat=0,maxlat=1088)
 /10.0.0.1:35462[0](queued=0,recved=1,sent=0)
 /10.0.0.1:51724[1]
(queued=0,recved=45786,sent=45803,sid=0x100000f14a10001,lop=GETC,est=1459376911226,to=40000,lcxid=0
x21e,lzxid=0x10000017a,lresp=604620658,llat=0,minlat=0,avglat=0,maxlat=1841)
 /10.0.0.1:51725[1]
(queued=0,recved=60328,sent=60333,sid=0x100000f14a10002,lop=PING,est=1459376913455,to=30000,lcxid=0
```

```
xc,lzxid=0x10000017a,lresp=604618294,llat=0,minlat=0,avglat=0,maxlat=1356)
 /10.0.0.1:51730[1]
(queued=0,recved=45315,sent=45324,sid=0x100000f14a10006,lop=PING,est=1459376914516,to=40000,lcxid=0
x49,lzxid=0x10000017a,lresp=604623243,llat=0,minlat=0,avglat=0,maxlat=1630)
```

What to do next

Add additional NSX Controllers to the control cluster. See Join Additional NSX Controllers with the Cluster Master.

Join Additional NSX Controllers with the Cluster Master

Having a multi-node cluster of NSX Controllers helps ensure that at least one NSX Controller is always available.

Prerequisites

- Install three NSX Controller appliances.
- Verify that you have admin privileges to log in to the NSX Controller appliances.
- Make sure the NSX Controller nodes have joined the management plane. See Join NSX Controllers with the NSX Manager.
- Initialize the control cluster to create a control cluster master.
- In the `join control-cluster` command, you must use an IP address, not a domain name.
- If you are using vCenter and you are deploying NSX-T components to the same cluster, make sure to configure DRS anti-affinity rules. Anti-affinity rules prevent DRS from migrating more than one node to a single host.

Procedure

1 Open an SSH session for each of your NSX Controller appliances.

  For example, NSX-Controller1, NSX-Controller2, and NSX-Controller3. In this example, NSX-Controller1 has already initialized the control cluster and is the control cluster master.

2 On the non-master NSX Controllers, run the `set control-cluster security-model` command with a shared secret password. The shared-secret password entered for NSX-Controller2 and NSX-Controller3 must match the shared-secret password entered on NSX-Controller1.

For example:

```
NSX-Controller2> set control-cluster security-model shared-secret secret <NSX-Controller1's-shared-secret-password>

Security secret successfully set on the node.
```

```
NSX-Controller3> set control-cluster security-model shared-secret secret <NSX-Controller1's-shared-secret-password>

Security secret successfully set on the node.
```

3 On the non-master NSX Controllers, run the `get control-cluster certificate thumbprint` command.

The command output is a string of numbers that is unique to each NSX Controller.

For example:

```
NSX-Controller2> get control-cluster certificate thumbprint
...
```

```
NSX-Controller3> get control-cluster certificate thumbprint
...
```

4 On the master NSX Controller, run the **join control-cluster** command.

Provide the following information:

- IP address with an optional port number of the non-master NSX Controllers (NSX-Controller2 and NSX-Controller3 in the example)
- Certificate thumbprint of the non-master NSX Controllers

Do not run the `join` commands on multiple controllers in parallel. Make sure the each join is complete before joining another controller.

```
NSX-Controller1> join control-cluster <NSX-Controller2-IP> thumbprint <nsx-controller2's-thumbprint>
Node 192.168.210.48 has successfully joined the control cluster.
Please run 'activate control-cluster' command on the new node.
```

Make sure that NSX-Controller2 has joined the cluster by running the `get control-cluster status` command.

```
NSX-Controller1> join control-cluster <NSX-Controller3-IP> thumbprint <nsx-controller3's-
thumbprint>
Node 192.168.210.49 has successfully joined the control cluster.
Please run 'activate control-cluster' command on the new node.
```

Make sure that NSX-Controller3 has joined the cluster by running the `get control-cluster status` command.

5 On the two NSX Controller nodes that have joined the control cluster master, run the `activate control-cluster` command.

Note Do not run the `activate` commands on multiple NSX Controllers in parallel. Make sure each activation is complete before activating another controller.

For example:

```
NSX-Controller2> activate control-cluster
Control cluster activation successful.
```

On NSX-Controller2, run the `get control-cluster status verbose` command, and make sure that the `Zookeeper Server IP` is `reachable, ok`.

```
NSX-Controller3> activate control-cluster
Control cluster activation successful.
```

On NSX-Controller3, run the `get control-cluster status verbose` command, and make sure that the `Zookeeper Server IP` is `reachable, ok`.

6 Verify the result by running the `get control-cluster status` command.

```
NSX-Controller1> get control-cluster status
uuid: db4aa77a-4397-4d65-ad33-9fde79ac3c5c
is master: true
in majority: true
uuid                                   address              status
0cfe232e-6c28-4fea-8aa4-b3518baef00d 192.168.210.47       active
bd257108-b94e-4e6d-8b19-7fa6c012961d 192.168.210.48       active
538be554-1240-40e4-8e94-1497e963a2aa 192.168.210.49       active
```

The first UUID listed is for the control cluster as a whole. Each NSX Controller node has a UUID as well.

If you try to join a controller to a cluster and the command `set control-cluster security-model` or `join control-cluster` fails, the cluster configuration files might be in an inconsistent state.

To resolve the issue, perform the following steps:

- On the NSX Controller that you try to join to the cluster, run the command `deactivate control-cluster`.
- On the master controller, if the command `get control-cluster status` or `get control-cluster status verbose` displays information about the failed controller, run the command `detach control-cluster <IP address of failed controller>`.

What to do next

Deploy the NSX Edge. See Chapter 6 NSX Edge Installation.

NSX Edge Installation

6

The NSX Edge provides routing services and connectivity to networks that are external to the NSX-T deployment. An NSX Edge is required if you want to deploy a tier-0 router or a tier-1 router with stateful services such as network address translation (NAT), VPN and so on.

Table 6-1. NSX Edge Deployment, Platforms, and Installation Requirements

Requirements	Description
Supported deployment methods	■ OVA/OVF ■ ISO with PXE ■ ISO without PXE
Supported platforms	NSX Edge is supported only on ESXi or on bare metal. NSX Edge is not supported on KVM.
PXE installation	The Password string must be encrypted with sha-512 algorithm for the root and admin user password.
NSX-T appliance password	■ At least eight characters ■ At least one lower-case letter ■ At least one upper-case letter ■ At least one digit ■ At least one special character ■ At least five different characters ■ No dictionary words ■ No palindromes
Hostname	When installing NSX Edge, specify a hostname that does not contain invalid characters such as an underscore. If the hostname contains any invalid character, after deployment the hostname will be set to **`localhost`**. For more information about hostname restrictions, see https://tools.ietf.org/html/rfc952 and https://tools.ietf.org/html/rfc1123.
VMware Tools	The NSX Edge VM running on ESXi has VMTools installed. Do not remove or upgrade VMTools.
System	
NSX Ports	If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

Table 6-1. NSX Edge Deployment, Platforms, and Installation Requirements (Continued)

Requirements	Description
IP Addresses	If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance. IPv6 format is not supported.
OVF Template	■ Verify that you have adequate privileges to deploy an OVF template on the ESXi host. ■ Verify that hostnames do not include underscores. Otherwise, the hostname is set to *nsx-manager*. ■ A management tool that can deploy OVF templates, such as vCenter Server or the vSphere Client. The OVF deployment tool must support configuration options to allow for manual configuration. ■ The Client Integration Plug-in must be installed.
NTP Server	The same NTP server must be configured on all NSX Edge servers in an Edge cluster.

NSX Edge Installation Scenarios

Important When you install NSX Edge from an OVA or OVF file, either from vSphere Web Client or the command line, OVA/OVF property values such as user names, passwords, or IP addresses are not validated before the VM is powered on.

- If you specify a user name for the **admin** or **audit** user, the name must be unique. If you specify the same name, it is ignored and the default names (**admin** and **audit**) is used.
- If the password for the **admin** user does not meet the complexity requirements, you must log in to NSX Edge through SSH or at the console as the **admin** user with the password **vmware**. You are prompted to change the password.
- If the password for the **audit** user does not meet the complexity requirements, the user account is disabled. To enable the account, log in to NSX Edge through SSH or at the console as the **admin** user and run the command **set user audit** to set the **audit** user's password (the current password is an empty string).
- If the password for the **root** user does not meet the complexity requirements, you must log in to NSX Edge through SSH or at the console as **root** with the password **vmware**. You are prompted to change the password.

- Do not use **root** user credentials to perform operations on the product. You must use this access only when requested by the VMware Support team. Using the **root** user credentials to install daemons or applications will your support contract.

Note The core services on the appliance do not start until a password with sufficient complexity has been set.

After you deploy NSX Edge from an OVA file, you cannot change the VM's IP settings by powering off the VM and modifying the OVA settings from vCenter Server.

This chapter includes the following topics:

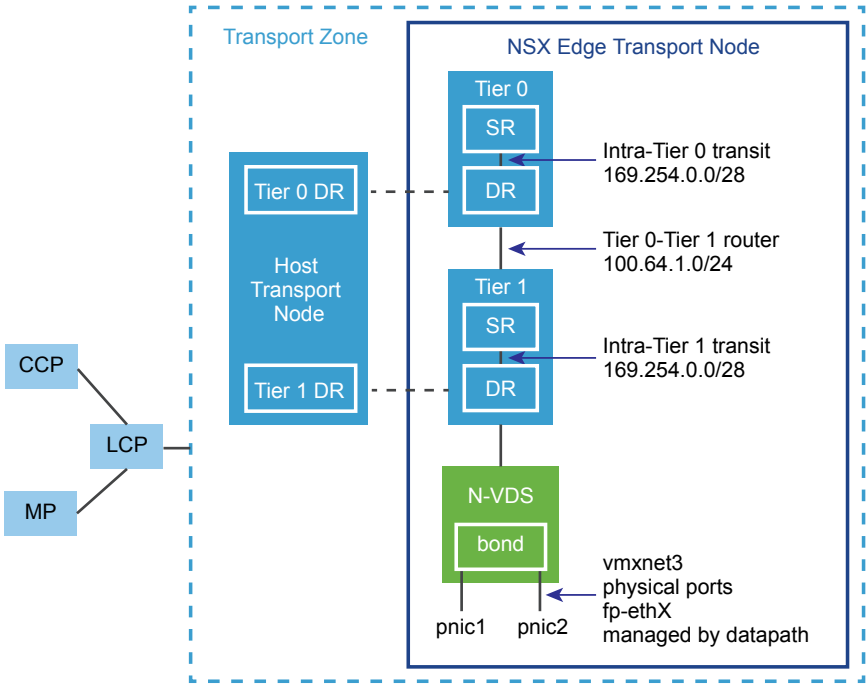
NSX Edge Networking Setup

NSX Edge can be installed via ISO, OVA/OVF, or PXE boot. Regardless of the installation method, make sure the host networking is prepared before you install NSX Edge.

High-Level View of NSX Edge Within a Transport Zone

The high-level view of NSX-T shows two transport nodes in a transport zone. One transport node is a host. The other is an NSX Edge.

Figure 6-1. High-Level Overview of NSX Edge



When you first deploy an NSX Edge, you can think of it as an empty container. The NSX Edge does not do anything until you create logical routers. The NSX Edge provides the compute backing for tier-0 and tier-1 logical routers. Each logical router contains a services router (SR) and a distributed router (DR). When we say that a router is distributed, we mean that it is replicated on all transport nodes that belong to the same transport zone. In the figure, the host transport node contains the same DRs contained on the tier-0 and tier-1 routers. A services router is required if the logical router is going to be configured to perform services, such as NAT. All tier-0 logical routers have a services router. A tier-1 router can have a services router if needed based on your design considerations.

By default, the links between the SR and the DR use the 169.254.0.0/28 subnet. These intra-router transit links are created automatically when you deploy a tier-0 or tier-1 logical router. You do not need to configure or modify the link configuration unless the 169.254.0.0/28 subnet is already in use in your deployment. Note that on a tier-1 logical router, the SR is present only if you select an NSX Edge cluster when creating the tier-1 logical router.

The default address space assigned for the tier-0-to-tier-1 connections is 100.64.0.0/10. Each tier-0-to-tier-1 peer connection is provided a /31 subnet within the 100.64.0.0/10 address space. This link is created automatically when you create a tier-1 router and connect it to a tier-0 router. You do not need to configure or modify the interfaces on this link unless the 100.64.0.0/10 subnet is already in use in your deployment.

Each NSX-T deployment has a management plane cluster (MP) and a control plane cluster (CCP). The MP and the CCP push configurations to each transport zone's local control plane (LCP). When a host or NSX Edge joins the management plane, the management plane agent (MPA) establishes connectivity with the host or NSX Edge, and the host or NSX Edge becomes an NSX-T fabric node. When the fabric node is then added as a transport node, LCP connectivity is established with the host or NSX Edge.

Lastly, the figure shows an example of two physical NICs (pnic1 and pnic2) that are bonded to provide high availability. These physical NICs are managed by the datapath. They can serve as either VLAN uplinks to an external network or as tunnel endpoint links to internal NSX-T-managed VM networks.

It is a best practice to allocate at least two physical links to each NSX Edge. Optionally, you can overlap the port groups on the same physical NIC using different VLAN IDs. The first network link found is used for management. For example, on an NSX Edge VM, the first link found might be vnic1. On a bare-metal installation, the first link found might be eth0 or em0. The remaining links are used for the uplinks and tunnels. For example, one might be for a tunnel endpoint used by NSX-T-managed VMs. The other might be used for an NSX Edge-to-external TOR uplink.

You can view the physical link information in the NSX Edge CLI by running the `get interfaces` and `get physical-ports` commands. In the API, you can use the `GET fabric/nodes/<edge-node-id>/network/interfaces` API call. Physical links are discussed in more detail in the next section.

Whether you install NSX Edge as a VM appliance or on bare metal, you have multiple options for the network configuration, depending on your deployment.

Transport Zones and N-VDS

To understand NSX Edge networking, you must know something about transport zones and N-VDS. Transport zones control the reach of Layer 2 networks in NSX-T. N-VDS is a software switch that gets created on a transport node. The purpose of N-VDS is to bind logical router uplinks and downlinks to physical NICs. For each transport zone that an NSX Edge belongs to, a single N-VDS gets installed on the NSX Edge.

There are two types of transport zones:

- Overlay for internal NSX-T tunneling between transport nodes—The NSX Edge can belong to only one overlay transport zone.
- VLAN for uplinks external to NSX-T—There is no restriction on the number of VLAN transport zones that an NSX Edge can belong to.

  An NSX Edge can belong to zero VLAN transport zones or many. In the case of zero VLAN transport zones, the NSX Edge can still have uplinks because the NSX Edge uplinks can use the same N-VDS installed for the overlay transport zone. You would do this if you want each NSX Edge to have only one N-VDS. Another design option is for the NSX Edge to belong to multiple VLAN transport zones, one for each uplink.

The most common design choice is three transport zones: One overlay and two VLAN transport zones for redundant uplinks.

Note that if you need to use the same VLAN ID for a transport network for overlay traffic and for other VLAN traffic, such as for a VLAN uplink, you must configure these on two different N-VDS, one for VLAN and the other for overlay.

For more information about transport zones, see About Transport Zones.

Virtual-Appliance/VM NSX Edge Networking

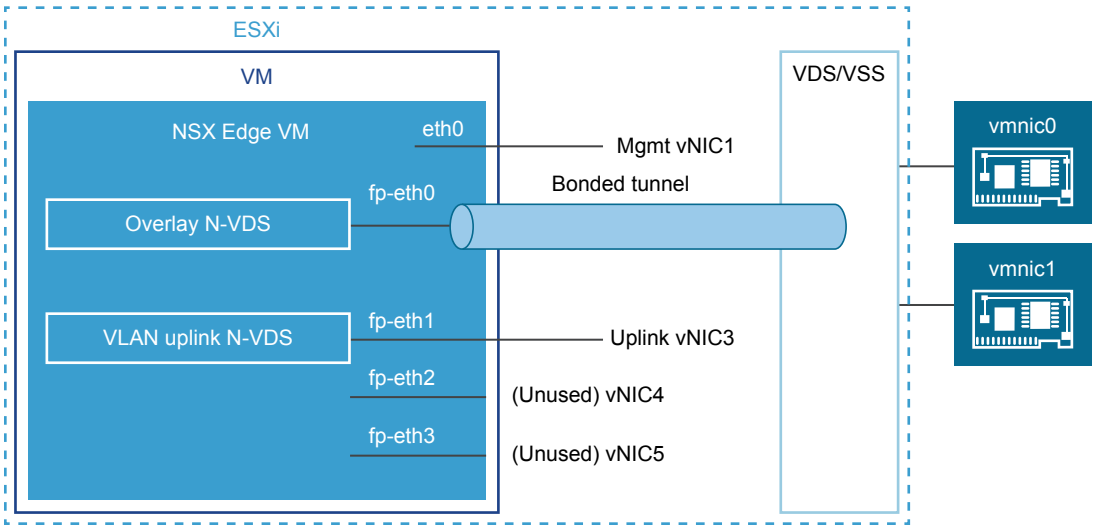
When you install NSX Edge as a virtual appliance or VM, internal interfaces are created, called fp-ethX, where X is 0, 1, 2, and 3. These interfaces are allocated for uplinks to a top-of-rack (ToR) switches and for NSX-T overlay tunneling.

When you create the NSX Edge transport node, you can select fp-ethX interfaces to associate with the uplinks and the overlay tunnel. You can decide how to use the fp-ethX interfaces.

On the vSphere distributed switch or vSphere standard switch, you should allocate at least two vmnics to the NSX Edge: One for NSX Edge management and one for uplinks and tunnels.

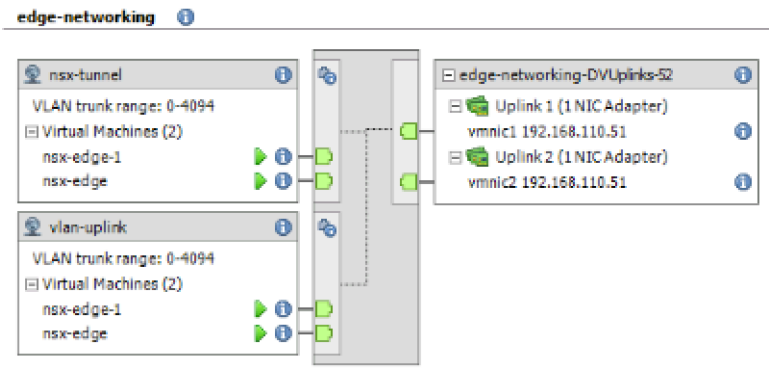
In the following sample physical topology, fp-eth0 is used for the NSX-T overlay tunnel. fp-eth1 is used for the VLAN uplink. fp-eth2 and fp-eth3 are not used.

Figure 6-2. One Suggested Link Setup for NSX Edge VM Networking



The NSX Edge shown in this example belongs to two transport zones (one overlay and one VLAN) and therefore has two N-VDS, one for tunnel and one for uplink traffic.

This screen shot shows the virtual machine port groups, nsx-tunnel and vlan-uplink.



During deployment, you must specify the network names that match the names configured on your VM port groups. For example, to match the VM port groups in the example, your network ovftool settings would be as follows if you were using the ovftool to deploy NSX Edge:

```
--net:"Network 0-Mgmt" --net:"Network 1-nsx-tunnel" --net:"Network 2=vlan-uplink"
```

The example shown here uses the VM port group names Mgmt, nsx-tunnel and vlan-uplink. This is just an example. You can use any names for your VM port groups.

The tunnel and uplink VM port groups configured for the NSX Edge do not need to be associated with VMkernel ports or given IP addresses. This is because they are used at Layer 2 only. If your deployment will use DHCP to provide an address to the management interface, make sure that only one NIC is assigned to the management network.

Notice that the VLAN and tunnel port groups are configured as trunk ports. This is required. For example, on a standard vSwitch, you configure trunk ports as follows: **Host > Configuration > Networking > Add Networking > Virtual Machine > VLAN ID All (4095)**.

If you are using an appliance-based or VM NSX Edge, you can use standard vSwitches or vSphere distributed switches.

It is possible to deploy an NSX Edge and a host transport node on the same hypervisor.

Optionally, you can install multiple NSX Edge appliances/VMs on a single host, and the same management, VLAN, and tunnel endpoint port groups can be used by all installed NSX Edges.

With the underlying physical links up and the VM port groups configured, you can install the NSX Edge.

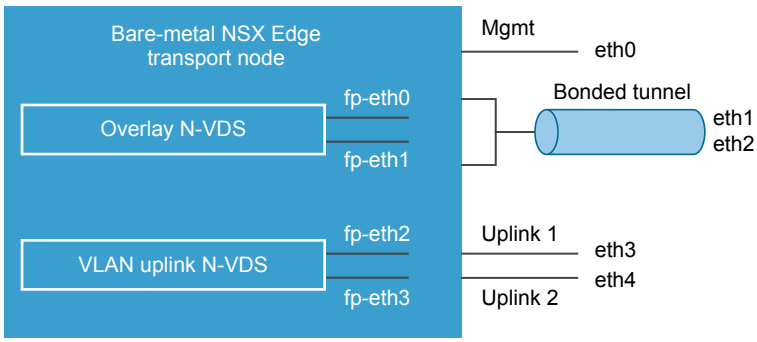
Bare-Metal NSX Edge Networking

The bare-metal NSX Edge contains internal interfaces called fp-ethX, where X is 0, 1, 2, 3, or 4. The number of fp-ethX interfaces created depends on how many physical NICs your bare-metal NSX Edge has. Up to four of these interfaces can be allocated for uplinks to top-of-rack (ToR) switches and NSX-T overlay tunneling.

When you create the NSX Edge transport node, you can select fp-ethX interfaces to associate with the uplinks and the overlay tunnel.

You can decide how to use the fp-ethX interfaces. In the following sample physical topology, fp-eth0 and fp-eth1 are bonded and used for the NSX-T overlay tunnel. fp-eth2 and fp-eth3 are used as redundant VLAN uplinks to TORs.

Figure 6-3. One Suggested Link Setup for Bare-Metal NSX Edge Networking



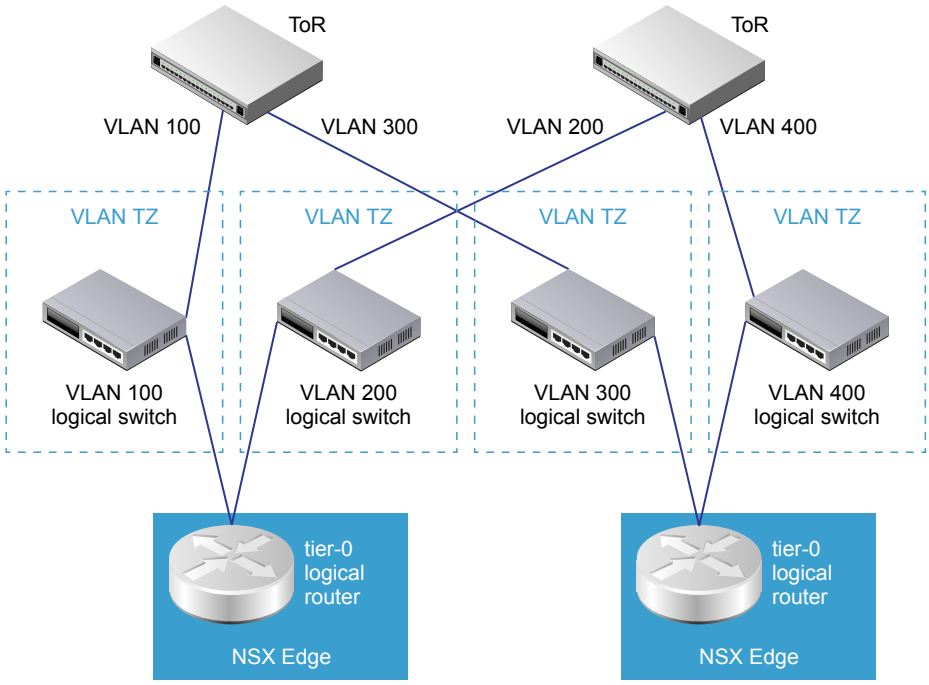
NSX Edge Uplink Redundancy

NSX Edge uplink redundancy allows two VLAN equal-cost multipath (ECMP) uplinks to be used on the NSX Edge-to-external TOR network connection.

When you have two ECMP VLAN uplinks, you should also have two TOR switches for high availability and fully meshed connectivity. Each VLAN logical switch has an associated VLAN ID.

When you add an NSX Edge to a VLAN transport zone, a new N-VDS is installed. For example, if you add an NSX Edge node to four VLAN transport zones, as shown in the figure, four N-VDS get installed on the NSX Edge.

Figure 6-4. One Suggested ECMP VLAN Setup for NSX Edge s to TORs



Create an NSX Edge VM on ESXi Host

You can configure an NSX Edge in the NSX Manager UI and automatically deploy the NSX Edge in vCenter Server.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- Verify that hostnames do not include underscores. Otherwise, the hostname is set to *localhost*.
- See NSX Edge network requirements in NSX Edge Networking Setup.
- Verify that the vCenter Server is registered as a compute manager.
- Verify that the vCenter Server datastore on which the NSX Edge is being installed has a minimum of 120GB available.
- Verify that the vCenter Server Cluster or Host has access to the specified networks and datastore in the configuration.

Procedure

1. From a browser, log in with admin privileges to an NSX Manager at `https://<nsx-manager-ip-address>`.
2. Select **Fabric > Nodes > Edges > Add Edge VM**.
3. Type a name for the NSX Edge.
4. Type the Host name or FQDN from vCenter Server.
5. Select a configuration size: small, medium, or large.

   The system requirements vary depending on the configuration size.
6. Specify the CLI and the root passwords for the systems.

   The restrictions on the root and CLI admin passwords also apply for automatic deployment.
7. Select the Compute Manager from the drop-down menu.

   The Compute Manager is the vCenter Server registered in the Management Plane.
8. For the Compute Manager, select a cluster from the drop-down menu or assign a resource pool.
9. Select a datastore to store the NSX Edge virtual machine files.

10 Select the cluster on which to deploy the NSX Edge VM.

   It is recommended to add the NSX Edge in a cluster that provides network management utilities.

11 Select the IP address and type the management network IP addresses and paths on which to place the NSX Edge interfaces.

   The management network must be able to access the NSX Manager. You can change the networks after the NSX Edge is deployed.

12 Add a default gateway if the management network IP address does not belong to same Layer 2 as the NSX Manager network.

   Verify that Layer 3 connectivity is available between NSX Manager and NSX Edge management network.

The NSX Edge deployment takes a 1-2 minutes to complete. You can track the real-time status of the deployment in the UI.

What to do next

If the NSX Edge deployment fails, navigate to `/var/log/cm-inventory/cm-inventory.log` and `/var/log/proton/nsxapi.log` files to troubleshoot the problem.

Before you add the NSX Edge to an NSX Edge cluster or configure as a transport node, make sure that the newly created NSX Edge node appears as Node Ready.

Install an NSX Edge on ESXi Using a GUI

If you prefer an interactive NSX Edge installation, you can use a UI-based VM management tool, such as the vSphere Client connected to vCenter Server.

In this release of NSX-T, IPv6 is not supported.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- Verify that you have adequate privileges to deploy an OVF template on the ESXi host.
- Verify that hostnames do not include underscores. Otherwise, the hostname is set to *localhost*.
- A management tool that can deploy OVF templates, such as vCenter Server or the vSphere Client.

  The OVF deployment tool must support configuration options to allow for manual configuration.

- The Client Integration Plug-in must be installed.
- See NSX Edge network requirements in NSX Edge Networking Setup.

Procedure

1. Locate the NSX Edge OVA or OVF file.

   Either copy the download URL or download the OVA file onto your computer.
2. In the management tool, launch the **Deploy OVF template** wizard and navigate or link to the .ova file.
3. Enter a name for the NSX Edge, and select a folder or vCenter Server datacenter.

   The name you type appears in the inventory.

   The folder you select is used to apply permissions to the NSX Edge.
4. Select a configuration size: small, medium, or large.

   The system requirements vary depending on the configuration size. See the *NSX-T Release Notes*.
5. Select a datastore to store the NSX Edge virtual appliance files.
6. If you are installing in vCenter Server, select a host or cluster on which to deploy the NSX Edge appliance.

   Normally, you would place the NSX Edge in a cluster that provides network management utilities.
7. Select the networks on which to place the NSX Edge interfaces.

   You can change the networks after the NSX Edge is deployed.
8. Specify the NSX Edge password and IP settings.
9. (Optional) For optimal performance, reserve memory for the NSX-T component.

   A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.
10. Open the console of the NSX Edge to track the boot process.

    If the console window does not open, make sure that pop-ups are allowed.
11. After the NSX Edge is completely booted, log in to the CLI with admin privileges.
12. Run the `get interface eth0` command to verify that the IP address was applied as expected

```
nsx-edge-1> get interface eth0

Interface: eth0
  Address: 192.168.110.37/24
  MAC address: 00:50:56:86:62:4d
```

```
MTU: 1500
Default gateway: 192.168.110.1
Broadcast address: 192.168.110.255
...
```

If needed, run the `set interface eth0 ip <CIDR> gateway <gateway-ip> plane mgmt` command to update the management interface. Optionally, you can start the SSH service with the `start service ssh` command.

13 Verify that the NSX Edge appliance has the required connectivity.

If you enabled SSH, make sure that you can SSH to your NSX Edge.

- You can ping your NSX Edge.
- NSX Edge can ping its default gateway.
- NSX Edge can ping the hypervisor hosts that are in the same network as the NSX Edge.
- NSX Edge can ping its DNS server and its NTP server.

14 Troubleshoot connectivity problems.

Note If connectivity is not established, make sure the VM network adapter is in the proper network or VLAN.

By default, the NSX Edge datapath claims all virtual machine NICs except the management NIC (the one that has an IP address and a default route). If DHCP assigns the wrong NIC as management, complete the tasks to correct the problem.

a Log in CLI and type the **`stop service dataplane`** command.

b Type the **`set interface eth0 dhcp plane mgmt`** command.

c Place eth0 into the DHCP network and wait for an IP address to be assigned to eth0.

d Type the **`start service dataplane`** command.

The datapath fp-ethX ports used for the VLAN uplink and the tunnel overlay are shown in the **get interfaces** and **get physical-port** commands on the NSX Edge.

What to do next

Join the NSX Edge with the management plane. See Join NSX Edge with the Management Plane.

Install NSX Edge on ESXi Using the Command-Line OVF Tool

If you prefer to automate NSX Edge installation, you can use the VMware OVF Tool, which is a command-line utility.

In this release of NSX-T, IPv6 is not supported.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- See NSX Edge network requirements in NSX Edge Networking Setup.
- Verify that you have adequate privileges to deploy an OVF template on the ESXi host.
- Verify that hostnames do not include underscores. Otherwise, the hostname is set to *localhost*.
- OVF Tool version 4.0 or later.

Procedure

- For a standalone host, run the `ovftool` command with the appropriate parameters.

```
C:\Users\Administrator\Downloads>ovftool
--name=nsx-edge-1
--deploymentOption=medium
--X:injectOvfEnv
--X:logFile=ovftool.log
--allowExtraConfig
--datastore=ds1
--net:"Network 0=Mgmt"
--net:"Network 1=nsx-tunnel"
--net:"Network 2=vlan-uplink"
--net:"Network 3=vlan-uplink"
--acceptAllEulas
--noSSLVerify
--diskMode=thin
--powerOn
--prop:nsx_ip_0=192.168.110.37
--prop:nsx_netmask_0=255.255.255.0
--prop:nsx_gateway_0=192.168.110.1
--prop:nsx_dns1_0=192.168.110.10
--prop:nsx_domain_0=corp.local
--prop:nsx_ntp_0=192.168.110.10
--prop:nsx_isSSHEnabled=True
--prop:nsx_allowSSHRootLogin=True
--prop:nsx_passwd_0=<password>
```

```
--prop:nsx_cli_passwd_0=<password>
--prop:nsx_hostname=nsx-edge
<path/url to nsx component ova>
vi://root:<password>@192.168.110.51
```

```
Opening OVA source: nsx-<component>.ova
The manifest validates
Source is signed and the certificate validates
Opening VI target: vi://root@192.168.110.24
Deploying to VI: vi://root@192.168.110.24
Transfer Completed
Powering on VM: nsx-edge-1
Task Completed
Completed successfully
```

- For a host managed by vCenter Server, run the `ovftool` command with the appropriate parameters.

```
C:\Users\Administrator\Downloads>ovftool
--name=nsx-edge-1
--deploymentOption=medium
--X:injectOvfEnv
--X:logFile=ovftool.log
--allowExtraConfig
--datastore=ds1
--net:"Network 0=Mgmt"
--net:"Network 1=nsx-tunnel"
--net:"Network 2=vlan-uplink"
--net:"Network 3=vlan-uplink"
--acceptAllEulas
--noSSLVerify
--diskMode=thin
--powerOn
--prop:nsx_ip_0=192.168.110.37
--prop:nsx_netmask_0=255.255.255.0
--prop:nsx_gateway_0=192.168.110.1
--prop:nsx_dns1_0=192.168.110.10
--prop:nsx_domain_0=corp.local
--prop:nsx_ntp_0=192.168.110.10
--prop:nsx_isSSHEnabled=True
--prop:nsx_allowSSHRootLogin=True
--prop:nsx_passwd_0=<password>
--prop:nsx_cli_passwd_0=<password>
--prop:nsx_hostname=nsx-edge
<path/url to nsx component ova>
vi://administrator@vsphere.local:<password>@192.168.110.24/?ip=192.168.210.53
```

```
Opening OVA source: nsx-<component>.ova
The manifest validates
Source is signed and the certificate validates
Opening VI target: vi://administrator@vsphere.local@192.168.110.24:443/
Deploying to VI: vi://administrator@vsphere.local@192.168.110.24:443/
```

```
Transfer Completed
Powering on VM: nsx-edge-1
Task Completed
Completed successfully
```

- (Optional) For optimal performance, reserve memory for the NSX-T component.

  A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

- Open the console of the NSX Edge to track the boot process.
- After the NSX Edge is completely booted, log in to the CLI with admin privileges.
- Run the `get interface eth0` command to verify that the IP address was applied as expected

```
nsx-edge-1> get interface eth0

Interface: eth0
  Address: 192.168.110.37/24
  MAC address: 00:50:56:86:62:4d
  MTU: 1500
  Default gateway: 192.168.110.1
  Broadcast address: 192.168.110.255
  ...
```

  If needed, run the `set interface eth0 ip <CIDR> gateway <gateway-ip> plane mgmt` command to update the management interface. Optionally, you can start the SSH service with the `start service ssh` command.

- Verify that the NSX Edge appliance has the required connectivity.

  If you enabled SSH, make sure that you can SSH to your NSX Edge.

  - You can ping your NSX Edge.
  - NSX Edge can ping its default gateway.
  - NSX Edge can ping the hypervisor hosts that are in the same network as the NSX Edge.
  - NSX Edge can ping its DNS server and its NTP server.

- Troubleshoot connectivity problems.

  Note If connectivity is not established, make sure the VM network adapter is in the proper network or VLAN.

By default, the NSX Edge datapath claims all virtual machine NICs except the management NIC (the one that has an IP address and a default route). If DHCP assigns the wrong NIC as management, complete the tasks to correct the problem.

a Log in CLI and type the **`stop service dataplane`** command.

b Type the **`set interface eth0 dhcp plane mgmt`** command.

c Place eth0 into the DHCP network and wait for an IP address to be assigned to eth0.

d Type the **`start service dataplane`** command.

The datapath fp-ethX ports used for the VLAN uplink and the tunnel overlay are shown in the **get interfaces** and **get physical-port** commands on the NSX Edge.

What to do next

Join the NSX Edge with the management plane. See Join NSX Edge with the Management Plane.

Install NSX Edge via ISO File With a PXE Server

You can install NSX Edge devices in an automated fashion on bare metal or as a VM using PXE. Note that PXE boot installation is not supported for NSX Manager and NSX Controller. This includes automatically configuring networking settings, such as IP address, gateway, network mask, NTP, and DNS.

This procedure demonstrates how to set up a PXE server on Ubuntu. PXE is made up of several components: DHCP, HTTP, and TFTP.

DHCP dynamically distributes IP settings to NSX-T components, such as NSX Edge. In a PXE environment, the DHCP server allows NSX Edge to request and receive an IP address automatically.

TFTP is a file-transfer protocol. The TFTP server is always listening for PXE clients on the network. When it detects any network PXE client asking for PXE services, it provides the NSX-T component ISO file and the installation settings contained in a preseed file.

After the PXE server is ready, the procedure shows how to install NSX Edge with a preseeded configuration file.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- A PXE server must be available in your deployment environment. The PXE server can be set up on any Linux distribution. The PXE server must have two interfaces, one for external communication and another for providing DHCP IP and TFTP services.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- See NSX Edge network requirements in NSX Edge Networking Setup.

Procedure

1 (Optional) Create a kickstart file to set up a new TFTP or DHCP services on an Ubuntu server.

A kickstart file is a text file that contains CLI commands that you run on the appliance after the first boot.

Name the kickstart file based on the PXE server it is pointing to. For example:

```
nsxcli.install
```

and must be copied to your web server, for example at `/var/www/html/nsx-edge/nsxcli.install`.

In the kickstart file, you can add CLI commands.

For example:

To configure the IP address of the management interface:

```
stop dataplane
set interface eth0 <ip-cidr-format> plane mgmt
start dataplane
```

To change the admin user password:

```
set user admin password <new_password> old-password <old-password>
```

Note that if you specify a password in the preseed.cfg file, use the same password in the kickstart file. Otherwise, use the default password, which is "default".

To join the NSX Edge with the management plane:

```
join management-plane <mgr-ip> thumbprint <mgr-thumbprint> username <mgr-username> password <mgr
password>
```

2 Create two interfaces, one for management and another for DHCP and TFTP services.

Make sure that the DHCP/TFTP interface is in the same subnet that the NSX Edge will reside in.

For example, if the NSX Edge management interfaces are going to be in the 192.168.210.0/24 subnet, place eth1 in that same subnet.

```
# The loopback network interface
auto lo
iface lo inet loopback

# PXE server's management interface
auto eth0
iface eth0 inet static
  address 192.168.110.81
  gateway 192.168.110.1
  netmask 255.255.255.0
```

```
  dns-nameservers 192.168.110.10

# PXE server's DHCP/TFTP interface
auto eth1
iface eth1 inet static
  address 192.168.210.82
  gateway 192.168.210.1
  netmask 255.255.255.0
  dns-nameservers 192.168.110.10
```

3 Install DHCP server software.

```
sudo apt-get install isc-dhcp-server -y
```

4 Edit the `/etc/default/isc-dhcp-server` file, and add the interface that provides DHCP service.

```
INTERFACES="eth1"
```

5 (Optional) If you want this DHCP server to be the official DHCP server for the local network, uncomment the **authoritative;** line in the `/etc/dhcp/dhcpd.conf` file.

```
...
authoritative;
...
```

6 In `/etc/dhcp/dhcpd.conf`, define the DHCP settings for the PXE network.

For example:

```
subnet 192.168.210.0 netmask 255.255.255.0 {
   range 192.168.210.90 192.168.210.95;
   option subnet-mask 255.255.255.0;
   option domain-name-servers 192.168.110.10;
   option routers 192.168.210.1;
   option broadcast-address 192.168.210.255;
   default-lease-time 600;
   max-lease-time 7200;
}
```

7 Start the DHCP service.

```
sudo service isc-dhcp-server start
```

8 Verify that the DHCP service is running.

```
service --status-all | grep dhcp
```

9 Install Apache, TFTP, and other components that are required for PXE booting.

```
sudo apt-get install apache2 tftpd-hpa inetutils-inetd
```

10 Verify that TFTP and Apache are running.

```
service --status-all | grep tftpd-hpa
service --status-all | grep apache2
```

11 Add the following lines to the `/etc/default/tftpd-hpa` file.

```
RUN_DAEMON="yes"
OPTIONS="-l -s /var/lib/tftpboot"
```

12 Add the following line to the `/etc/inetd.conf` file.

```
tftp    dgram   udp    wait    root    /usr/sbin/in.tftpd /usr/sbin/in.tftpd -s /var/lib/tftpboot
```

13 Restart the TFTP service.

```
sudo /etc/init.d/tftpd-hpa restart
```

14 Copy or download the NSX Edge installer ISO file to where it needs to be.

15 Mount the ISO file and copy the install components to the TFTP server and the Apache server.

```
sudo mount -o loop ~/nsx-edge.<build>.iso /mnt
cd /mnt
sudo cp -fr install/netboot/* /var/lib/tftpboot/
sudo mkdir /var/www/html/nsx-edge
sudo cp -fr /mnt/* /var/www/html/nsx-edge/
```

16 (Optional) Edit the `/var/www/html/nsx-edge/preseed.cfg` file to modify the encrypted passwords.

You can use a Linux tool such as mkpasswd to create a password hash.

```
sudo apt-get install whois
sudo mkpasswd -m sha-512

Password:
$6$SUFGqs[...]FcoHLijOuFD
```

a To modify the root password, edit `/var/www/html/nsx-edge/preseed.cfg` and search for the following line:

```
d-i passwd/root-password-crypted password $6$tgmLNLMp$9BuAHhN...
```

b Replace the hash string.

You do not need to escape any special character such as $, ', ", or \.

c Add the `usermod` command to `preseed.cfg` to set the password for root, admin, or both.

For example, search for the `echo 'VMware NSX Edge'` line and add the following command.

```
usermod --password '\$6\$VS3exId0aKmzW\$U3g0V7BF0DXlmRI.LR0v/VgloxVotEDp00bO2hUF8u/' root; \
usermod --password '\$6\$VS3exId0aKmzW\$U3g0V7BF0DXlmRI.LR0v/VgloxVotEDp00bO2hUF8u/' admin; \
```

The hash string is an example. You must escape all special characters. The root password in the first `usermod` command replaces the password that is set in `d-i passwd/root-password-crypted password $6$tgm...`.

If you use the `usermod` command to set the password, the user is not prompted to change the password at the first login. Otherwise, the user must change the password at the first login.

17 Add the following lines to the `/var/lib/tftpboot/pxelinux.cfg/default` file.

Be sure to replace 192.168.210.82 with the IP address of your TFTP server.

```
label nsxedge
    kernel ubuntu-installer/amd64/linux
    ipappend 2
    append netcfg/dhcp_timeout=60 auto=true priority=critical vga=normal partman-
lvm/device_remove_lvm=true netcfg/choose_interface=auto debian-
installer/allow_unauthenticated=true preseed/url=http://192.168.210.82/nsx-edge/preseed.cfg
mirror/country=manual mirror/http/hostname=192.168.210.82 nsx-
kickstart/url=http://192.168.210.82/nsx-edge/nsxcli.install mirror/http/directory=/nsx-edge
initrd=ubuntu-installer/amd64/initrd.gz mirror/suite=xenial --
```

18 Add the following lines to the `/etc/dhcp/dhcpd.conf` file.

Be sure to replace 192.168.210.82 with the IP address of your DHCP server.

```
allow booting;
allow bootp;

next-server 192.168.210.82; #Replace this IP address
filename "pxelinux.0";
```

19 Restart the DHCP service.

```
sudo service isc-dhcp-server restart
```

Note If an error is returned, for example: "stop: Unknown instance: start: Job failed to start", run `sudo /etc/init.d/isc-dhcp-server stop` and then `sudo /etc/init.d/isc-dhcp-server start`. The `sudo /etc/init.d/isc-dhcp-server start` command returns information about the source of the error.

20 Use the bare-metal install instructions or the ISO install instructions to complete the installation.

- Install NSX Edge on Bare Metal
- Install NSX Edge via ISO File as a Virtual Appliance

21 Power on the NSX Bare Metal Host.

22 At the boot menu, select **nsxedge**.

The network is configured, partitions are created, and the NSX Edge components are installed.

When the NSX Edge login prompt appears, you can log in as admin or root.

By default, the root login password is **vmware**, and the admin login password is **default**.

23 (Optional) For optimal performance, reserve memory for the NSX-T component.

A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

24 Open the console of the NSX Edge to track the boot process.

If the console window does not open, make sure that pop-ups are allowed.

25 After the NSX Edge is completely booted, log in to the CLI with admin privileges.

26 Run the `get interface eth0` command to verify that the IP address was applied as expected

```
nsx-edge-1> get interface eth0

Interface: eth0
  Address: 192.168.110.37/24
  MAC address: 00:50:56:86:62:4d
```

```
MTU: 1500
Default gateway: 192.168.110.1
Broadcast address: 192.168.110.255
...
```

If needed, run the `set interface eth0 ip <CIDR> gateway <gateway-ip> plane mgmt` command to update the management interface. Optionally, you can start the SSH service with the `start service ssh` command.

27 Verify that the NSX Edge appliance has the required connectivity.

If you enabled SSH, make sure that you can SSH to your NSX Edge.

- You can ping your NSX Edge.
- NSX Edge can ping its default gateway.
- NSX Edge can ping the hypervisor hosts that are in the same network as the NSX Edge.
- NSX Edge can ping its DNS server and its NTP server.

28 Troubleshoot connectivity problems.

Note If connectivity is not established, make sure the VM network adapter is in the proper network or VLAN.

By default, the NSX Edge datapath claims all virtual machine NICs except the management NIC (the one that has an IP address and a default route). If DHCP assigns the wrong NIC as management, complete the tasks to correct the problem.

a Log in CLI and type the **`stop service dataplane`** command.

b Type the **`set interface eth0 dhcp plane mgmt`** command.

c Place eth0 into the DHCP network and wait for an IP address to be assigned to eth0.

d Type the **`start service dataplane`** command.

The datapath fp-ethX ports used for the VLAN uplink and the tunnel overlay are shown in the **get interfaces** and **get physical-port** commands on the NSX Edge.

What to do next

Join the NSX Edge with the management plane. See Join NSX Edge with the Management Plane.

Install NSX Edge on Bare Metal

You can install NSX Edge devices in a manual fashion on bare metal using an ISO file. This includes configuring networking settings, such as IP address, gateway, network mask, NTP, and DNS.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.

- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- See NSX Edge network requirements in NSX Edge Networking Setup.

Procedure

1. Create a bootable disk with the NSX Edge ISO file on it.
2. Boot the physical machine from the disk.
3. Choose **Automated installation**.

   There might be a pause of 10 seconds after you press Enter.

   During power-on, the installer requests a network configuration via DHCP. If DHCP is not available in your environment, the installer prompts you for IP settings.

   By default, the root login password is **vmware**, and the admin login password is **default**.
4. Open the console of the NSX Edge to track the boot process.

   If the console window does not open, make sure that pop-ups are allowed.
5. After the NSX Edge is completely booted, log in to the CLI with admin privileges.
6. Run the `get interface eth0` command to verify that the IP address was applied as expected

   ```
   nsx-edge-1> get interface eth0

   Interface: eth0
     Address: 192.168.110.37/24
     MAC address: 00:50:56:86:62:4d
     MTU: 1500
     Default gateway: 192.168.110.1
     Broadcast address: 192.168.110.255
     ...
   ```

   If needed, run the `set interface eth0 ip <CIDR> gateway <gateway-ip> plane mgmt` command to update the management interface. Optionally, you can start the SSH service with the `start service ssh` command.
7. Verify that the NSX Edge appliance has the required connectivity.

   If you enabled SSH, make sure that you can SSH to your NSX Edge.

   - You can ping your NSX Edge.
   - NSX Edge can ping its default gateway.
   - NSX Edge can ping the hypervisor hosts that are in the same network as the NSX Edge.
   - NSX Edge can ping its DNS server and its NTP server.

8 Troubleshoot connectivity problems.

Note If connectivity is not established, make sure the VM network adapter is in the proper network or VLAN.

By default, the NSX Edge datapath claims all virtual machine NICs except the management NIC (the one that has an IP address and a default route). If DHCP assigns the wrong NIC as management, complete the tasks to correct the problem.

a Log in CLI and type the **`stop service dataplane`** command.

b Type the **`set interface eth0 dhcp plane mgmt`** command.

c Place eth0 into the DHCP network and wait for an IP address to be assigned to eth0.

d Type the **`start service dataplane`** command.

The datapath fp-ethX ports used for the VLAN uplink and the tunnel overlay are shown in the **get interfaces** and **get physical-port** commands on the NSX Edge.

What to do next

Join the NSX Edge with the management plane. See Join NSX Edge with the Management Plane.

Install NSX Edge via ISO File as a Virtual Appliance

You can install NSX Edge devices in a manual fashion using an ISO file.

Important The NSX-T component virtual machine installations include VMware Tools. Removal or upgrade of VMware Tools is not supported for NSX-T appliances.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- If you do not already have one, create the target VM port group network. It is recommended to place NSX-T appliances on a management VM network.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- See NSX Edge network requirements in NSX Edge Networking Setup.

Procedure

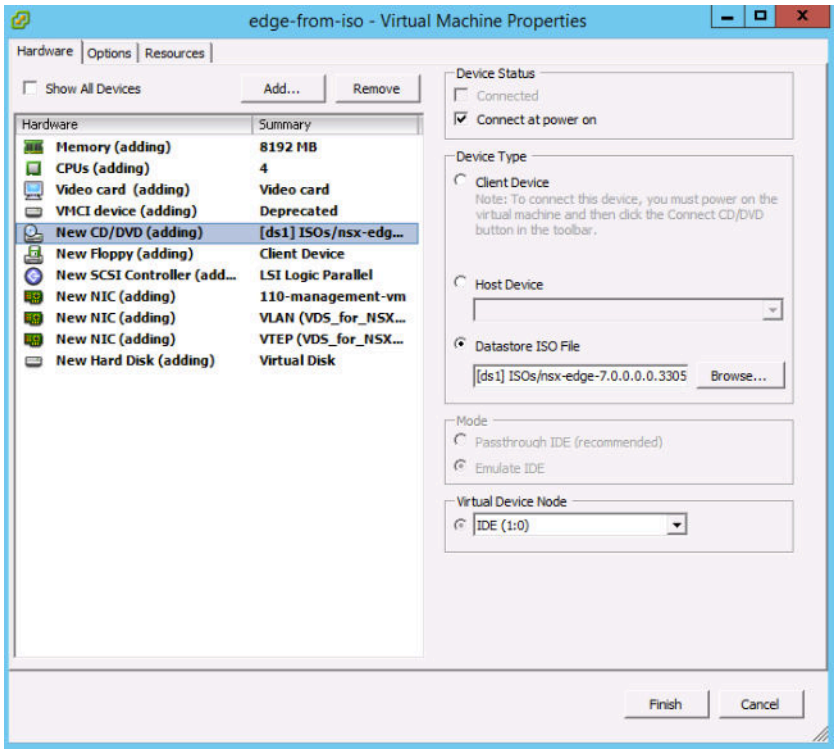
1 On a standalone host or in the vCenter Web client, create a VM and allocate the following resources:

- Guest operating system: Other (64-bit).
- 3 VMXNET3 NICs. NSX Edge does not support the e1000 NIC driver.

- The appropriate system resources required for your NSX-T deployment.

2 Bind the NSX Edge ISO file to the VM.

Make sure the CD/DVD drive device status is set to **Connect at power on**.



3 During ISO boot, open the VM console and choose **Automated installation**.

There might be a pause of 10 seconds after you press Enter.

During power-on, the VM requests a network configuration via DHCP. If DHCP is not available in your environment, the installer prompts you for IP settings.

By default, the root login password is **vmware**, and the admin login password is **default**.

When you log in for the first time, you are prompted to change the password. This password change method has strict complexity rules, including the following:

- At least eight characters
- At least one lower-case letter
- At least one upper-case letter
- At least one digit
- At least one special character
- At least five different characters
- No dictionary words

- No palindromes

Important The core services on the appliance do not start until a password with sufficient complexity has been set.

4 (Optional) For optimal performance, reserve memory for the NSX-T component.

A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

5 Open the console of the NSX Edge to track the boot process.

If the console window does not open, make sure that pop-ups are allowed.

6 After the NSX Edge is completely booted, log in to the CLI with admin privileges.

7 Run the `get interface eth0` command to verify that the IP address was applied as expected

```
nsx-edge-1> get interface eth0

Interface: eth0
  Address: 192.168.110.37/24
  MAC address: 00:50:56:86:62:4d
  MTU: 1500
  Default gateway: 192.168.110.1
  Broadcast address: 192.168.110.255
  ...
```

If needed, run the `set interface eth0 ip <CIDR> gateway <gateway-ip> plane mgmt` command to update the management interface. Optionally, you can start the SSH service with the `start service ssh` command.

8 Verify that the NSX Edge appliance has the required connectivity.

If you enabled SSH, make sure that you can SSH to your NSX Edge.

- You can ping your NSX Edge.
- NSX Edge can ping its default gateway.
- NSX Edge can ping the hypervisor hosts that are in the same network as the NSX Edge.
- NSX Edge can ping its DNS server and its NTP server.

9 Troubleshoot connectivity problems.

Note If connectivity is not established, make sure the VM network adapter is in the proper network or VLAN.

By default, the NSX Edge datapath claims all virtual machine NICs except the management NIC (the one that has an IP address and a default route). If DHCP assigns the wrong NIC as management, complete the tasks to correct the problem.

a Log in CLI and type the **`stop service dataplane`** command.

b Type the **`set interface eth0 dhcp plane mgmt`** command.

c Place eth0 into the DHCP network and wait for an IP address to be assigned to eth0.

d Type the **`start service dataplane`** command.

The datapath fp-ethX ports used for the VLAN uplink and the tunnel overlay are shown in the **get interfaces** and **get physical-port** commands on the NSX Edge.

What to do next

Join the NSX Edge with the management plane. See Join NSX Edge with the Management Plane.

Join NSX Edge with the Management Plane

Joining NSX Edges with the management plane ensures that the NSX Manager and NSX Edges can communicate with each other.

Prerequisites

Verify that you have admin privileges to log in to the NSX Edges and NSX Manager appliance.

Procedure

1 Open an SSH session to the NSX Manager appliance.

2 Open an SSH session to the NSX Edge.

3 On the NSX Manager appliance, run the `get certificate api thumbprint` command.

The command output is a string of numbers that is unique to this NSX Manager.

For example:

```
NSX-Manager1> get certificate api thumbprint
...
```

4 On the NSX Edge, run the **join management-plane** command.

Provide the following information:

- Hostname or IP address of the NSX Manager with an optional port number
- Username of the NSX Manager
- Certificate thumbprint of the NSX Manager

- Password of the NSX Manager

```
NSX-Edge1> join management-plane NSX-Manager1 username admin thumbprint <NSX-Manager1's-thumbprint>
Password for API user: <NSX-Manager1's-password>
Node successfully registered and Edge restarted
```

Repeat this command on each NSX Edge node.

Verify the result by running the `get managers` command on your NSX Edges.

```
nsx-edge-1> get managers
- 192.168.110.47   Connected
```

In the NSX Manager UI, the NSX Edge appears on the **Fabric > Edges** page. MPA connectivity should be Up. If MPA connectivity is not Up, try refreshing the browser screen.

What to do next

Add the NSX Edge as a transport node. See Create an NSX Edge Transport Node.

DNE Key Manager Installation

7

The DNE Key Manager is the component of the Distributed Network Encryption (DNE) feature that manages the keys used to provide encrypted and authenticated connections between two endpoints within a Software Defined Data Center (SDDC).

To use DNE, you must download and install the DNE Key Manager separately. The DNE Key Manager supports the OVA/OVF deployment method on ESXi.

Note You must have the NSX-T Enterprise license to use DNE.

The DNE Key Manager communicates with the NSX Manager and Hypervisors across SSL/TLS connections. DNE authenticates and encrypts network traffic for both vSphere and KVM hosts.

Table 7-1. DNE Key Manager Deployment, Platform, and Installation Requirements

Requirements	Description
Supported platforms	You can use the vSphere high availability (HA) feature to ensure the availability of the DNE Key Manager is deployed on ESXi.
Password	■ At least eight characters ■ At least one lower-case letter ■ At least one upper-case letter ■ At least one digit ■ At least one special character ■ At least five different characters ■ No dictionary words ■ No palindromes
Port number	8992 for NSX Manager 443 for vSphere or KVM Hosts **Note** Port configuration is handled automatically after a successful installation.
VMware Tools	The DNE Key Manager VM running on ESXi has VMTools installed. Do not remove or upgrade VMTools.

NSX Manager Installation Scenarios

- After you deploy DNE Key Manager from an OVA/OVF file, you cannot change the VM's IP settings by powering off the VM and modifying the OVA/OVF settings from vCenter Server. If you change the DNE Key Manager IP address, you must re-register it with the management plane.

  Note The core services on the appliance do not start until a password with sufficient complexity is set.

- When you install DNE Key Manager from an OVA or OVF file, either from vSphere Web Client or the command line, OVA/OVF property values such as usernames, passwords, or IP addresses are not validated before the VM is powered on. Make sure that the root and admin user passwords meet the password complexity requirements.
- Your passwords must comply with the password strength restrictions. NSX-T appliances enforce the complexity rules described in the password requirements.
- The installation succeeds if the password does not meet the requirements. However, when you log in for the first time, you are prompted to change the password.
- To support backup and restore, the DNE Key Manager must have static management IP addresses. Using DHCP to assign management IP addresses is not supported. Changing management IP addresses is not supported. See *NSX-T Administration Guide* for backup and restore information.

This chapter includes the following topics:

Download the DNE Key Manager on ESXi

You can install the DNE Key Manager as a virtual appliance on ESXi in your NSX-T environment. If your installation involves vSphere HA, you deploy the DNE Key Manager on one or more ESXi hosts.

Note It is recommended that you install the DNE Key Manager in the same cluster where the NSX Manager and NSX Controller are installed.

Prerequisites

- Verify that the NSX Manager is installed. See Chapter 4 NSX Manager Installation.
- Verify that at least one NSX Controller is configured. See Chapter 5 NSX Controller Installation and Clustering.

- Verify that your environment has the supported platforms, ports, and protocols. See Ports and Protocols.

Procedure

1. Download the DNE Key Manager OVA/OVF from the NSX-T downloads page.
2. Copy this file to the location where you plan to install DNE Key Manager.
3. Deploy the DNE Key Manager OVF from either the vSphere Client connected to vSphere Server or the CLI.

   Some topologies might require the DNE Key Manager to connect using some other interface other than eth0, in which case the join command can use the option interface and provide the interface of choice.

Install DNE Key Manager on ESXi Using a GUI

If you prefer an interactive DNE Key Manager installation, you can use a UI-based VM management tool, such as the vSphere Client connected to the vCenter Server.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- Most deployments place NSX-T appliances on a management VM network. You can also create a new VM port group for the DNE Key Manager appliance.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- Verify that you have adequate privileges to deploy an OVF template on the ESXi host.
- Verify that hostnames do not include underscores. Otherwise, the hostname is set to *localhost*.
- A management tool that can deploy OVF templates, such as vCenter Server or the vSphere Client.

  The OVF deployment tool must support configuration options to allow for manual configuration.
- The Client Integration Plug-in must be installed.

Procedure

1. Locate the DNE Key Manager OVA or OVF file.

   In the vSphere client, launch the **Deploy OVF template** wizard and navigate or link to the .ova or ovf file.
2. Enter a name for the DNE Key Manager, and select a folder or vCenter Server datacenter.

   The name you type appears in the inventory.

   The folder you select is used to apply permissions to the DNE Key Manager.

3 Select a datastore to store the DNE Key Manager virtual appliance files.

4 If you are installing in vCenter Server, select a host or cluster on which to deploy the DNE Key Manager appliance.

5 Select the networks on which to place the NSX Edge interfaces.

   You can change the networks after the NSX Edge is deployed.

6 Select the port group or destination network for the DNE Key Manager.

   For example, if you are using vSphere distributed switches, you might place DNE Key Manager on a port group called Mgmt_VDS - Mgmt.

7 Specify the DNE Key Manager password and IP settings.

8 (Optional) For optimal performance, reserve memory for the NSX-T component.

   A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.

9 Open the console of the NSX Edge to track the boot process.

10 After the DNE Key Manager is completely booted, log in to the CLI as root and run the ifconfig command.

   For example, run `ifconfig eth0` or the interface you use to connect to the management switch to verify that the IP address was applied as expected.

11 Verify that the NSX Edge appliance has the required connectivity.

   If you enabled SSH, make sure that you can SSH to your NSX Edge.

   - You can ping your NSX Edge.
   - NSX Edge can ping its default gateway.
   - NSX Edge can ping the hypervisor hosts that are in the same network as the NSX Edge.
   - NSX Edge can ping its DNS server and its NTP server.

What to do next

Join the DNE Key Manager with the management plane. See Join DNE Key Manager with the Management Plane.

Install DNE Key Manager on ESXi Using the Command-Line OVF Tool

If you prefer to automate DNE Key Manager installation, you can use the VMware OVF Tool, which is a command-line utility.

By default, nsx_isSSHEnabled and nsx_allowSSHRootLogin are both disabled for security reasons. When they are disabled, you cannot SSH or log in to the DNE Key Manager command line. If you enable nsx_isSSHEnabled but not nsx_allowSSHRootLogin, you can SSH to DNE Key Manager but you cannot log in as root.

Prerequisites

- Verify that the system requirements are met. See System Requirements.
- Verify that the required ports are open. See Ports and Protocols.
- Most deployments place NSX-T appliances on a management VM network. You can also create a new VM port group for the DNE Key Manager appliance.

  If you have multiple management networks, you can add static routes to the other networks from the NSX-T appliance.
- Plan your IPv4 IP address scheme. In this release of NSX-T, IPv6 is not supported.
- Verify that you have adequate privileges to deploy an OVF template on the ESXi host.
- Verify that hostnames do not include underscores. Otherwise, the hostname is set to *localhost*.
- OVF Tool version 4.0 or later.
- For the nsx_hostname=nsx-keymanager property, enclose the root password (<password>) in single quotes.

  For example: `vi://root:'my_root_password'@10.112.202.150`.

Procedure

- For a standalone host, run the `ovftool` command with the appropriate parameters.

```
C:\Users\Administrator\Downloads>ovftool
--name=nsx-keymanager
--X:injectOvfEnv
--X:logFile=ovftool.log
--allowExtraConfig
--datastore=ds1
--network="management"
--acceptAllEulas
--noSSLVerify
--diskMode=thin
--powerOn
--prop:nsx_ip_0=192.168.110.210
--prop:nsx_netmask_0=255.255.255.0
--prop:nsx_gateway_0=192.168.110.1
--prop:nsx_dns1_0=192.168.110.10
--prop:nsx_domain_0=corp.local
--prop:nsx_ntp_0=192.168.110.10
--prop:nsx_isSSHEnabled=<True|False>
--prop:nsx_allowSSHRootLogin=<True|False>
--prop:nsx_passwd_0=<password>
```

```
--prop:nsx_cli_passwd_0=<password>
--prop:nsx_cli_audit_passwd_0=<password>
--prop:nsx_hostname=nsx-keymanager
<path/url to nsx component ova> vi://root:<password>@192.168.110.51
```

```
Opening OVA source: nsx-<component>.ova
The manifest validates
Source is signed and the certificate validates
Opening VI target: vi://root@192.168.110.24
Deploying to VI: vi://root@192.168.110.24
Transfer Completed
Powering on VM: nsx-edge-1
Task Completed
Completed successfully
```

- For a host managed by vCenter Server, run the `ovftool` command with the appropriate parameters.

```
C:\Users\Administrator\Downloads>ovftool
--name=nsx-keymanager
--X:injectOvfEnv
--X:logFile=ovftool.log
--allowExtraConfig
--datastore=ds1
--network="management"
--acceptAllEulas
--noSSLVerify
--diskMode=thin
--powerOn
--prop:nsx_ip_0=192.168.110.210
--prop:nsx_netmask_0=255.255.255.0
--prop:nsx_gateway_0=192.168.110.1
--prop:nsx_dns1_0=192.168.110.10
--prop:nsx_domain_0=corp.local
--prop:nsx_ntp_0=192.168.110.10
--prop:nsx_isSSHEnabled=<True|False>
--prop:nsx_allowSSHRootLogin=<True|False>
--prop:nsx_passwd_0=<password>
--prop:nsx_cli_passwd_0=<password>
--prop:nsx_cli_audit_passwd_0=<password>
--prop:nsx_hostname=nsx-keymanager
<path/url to nsx component ova> vi://administrator@vsphere.local:<password>@192.168.110.24/?
ip=192.168.110.51
```

```
Opening OVA source: nsx-<component>.ova
The manifest validates
Source is signed and the certificate validates
Opening VI target: vi://administrator@vsphere.local@192.168.110.24:443/
Deploying to VI: vi://administrator@vsphere.local@192.168.110.24:443/
Transfer Completed
```

```
Powering on VM: nsx-edge-1
Task Completed
Completed successfully
```

- (Optional) For optimal performance, reserve memory for the NSX-T component.

  A memory reservation is a guaranteed lower bound on the amount of physical memory that the host reserves for a virtual machine, even when memory is overcommitted. Set the reservation to a level that ensures the NSX-T component has sufficient memory to run efficiently. See System Requirements.
- Open the console of the NSX Edge to track the boot process.
- After the DNE Key Manager is completely booted, log in to the CLI as root and run the ifconfig command.

  For example, run `ifconfig eth0` or the interface you use to connect to the management switch to verify that the IP address was applied as expected.
- Verify that the NSX Edge appliance has the required connectivity.

  If you enabled SSH, make sure that you can SSH to your NSX Edge.
  - You can ping your NSX Edge.
  - NSX Edge can ping its default gateway.
  - NSX Edge can ping the hypervisor hosts that are in the same network as the NSX Edge.
  - NSX Edge can ping its DNS server and its NTP server.

What to do next

Join the DNE Key Manager with the management plane. See Join DNE Key Manager with the Management Plane.

Join DNE Key Manager with the Management Plane

Joining NSX Managerand the DNE Key Manager allows these components to communicate with each other.

Prerequisites

Verify that the NSX Manager is installed.

Procedure

1. Open an SSH session to the NSX Manager appliance as **admin** to log into the CLI.
2. Open an SSH session to the DNE Key Manager appliance as **admin** to log into the CLI.

3 On the NSX Manager appliance, run the `get certificate api thumbprint` command.

For example:

```
NSX-Manager1> get certificate api thumbprint
...
```

The command output is a string of numbers unique to this NSX Manager.

4 On the DNE Key Manager appliance, run the **join management-plane** command.

When prompted, provide the following information:

- Hostname or IP address of the NSX Manager with an optional port number
- Username of the NSX Manager
- Certificate thumbprint of the NSX Manager
- Password of the NSX Manager
- Interface name. The default interface is eth0.

```
NSX-Key-Manager1> join management-plane <NSX-Manager1-IP-Address> username admin thumbprint <NSX-
Manager1-thumbprint>
Password for API user: <NSX-Manager1-password>
Restarting the KeyManager service. This may take a while ...
Restart Done.
KeyManager node successfully registered and service restarted
```

5 Verify that the DNE Key Manager is configured properly using either a GUI or API call.

- From your browser, log in to NSX Manager https://nsx-manager-ip-address. Select **Encryption** and navigate to the **Keys** tab.

  Key Manager Status: Connected with a green dot appears.
- Invoke the API call, `/api/v1/network-encryption/key-managers` .

What to do next

Enable the DNE configuration. See Enabling and Disabling DNE.

Enabling and Disabling DNE

After installation, DNE is initially disabled by default (per licensing requirements). Installation does not depend on whether DNE is enabled or not.

Procedure

- To enable DNE using REST API.

  `POST /api/v1/network-encryption/status?action=update_status&status=ENABLE&context=ALL`

- To disable DNE using REST API.
  a Do a GET API call.

    ```
    GET /api/v1/network-encryption/status
    ```

  b From the GET results, issue a POST command with the changed data.

    ```
    POST /api/v1/network-encryption/status?
    action=update_status&status=DISABLE&context=ALL
    ```

    DNE immediately suspends all policy enforcement operations which includes authentication and encryption. While disabled, existing policy configurations are not enforced. The policy configurations are not deleted to let you can enable them when needed.

What to do next

You can enable or disable DNE after you install using either the NSX Manager console. See "Manage DNE Settings" in the *NSX-T Administration Guide*.

Host Preparation

8

When hypervisor hosts are prepared to operate with NSX-T, they are known as fabric nodes. Hosts that are fabric nodes have NSX-T modules installed and are registered with the NSX-T management plane.

This chapter includes the following topics:

Install Third-Party Packages on a KVM Host

To prepare a KVM host to be a fabric node, you must install some third-party packages.

Procedure

- For Ubuntu 16.04.2 LTS, run the following commands:

```
apt-get install libunwind8 libgflags2v5 libgoogle-perftools4 traceroute
apt-get install python-mako python-simplejson python-unittest2 python-yaml python-netaddr
apt-get install libboost-filesystem1.58.0 libboost-chrono1.58.0 libgoogle-glog0v5
apt-get install dkms
apt-get install libboost-date-time1.58.0 python-protobuf python-gevent libsnappy1v5 libleveldb1v5
```

- Verify that the Red Hat 7.4 and 7.3 hosts are registered and the Red Hat repository is accessible.

  If the Red Hat host is not registered, manually install the listed dependencies.

  - tcpdump
  - boost-filesystem
  - PyYAML
  - boost-iostreams
  - boost-chrono
  - python-mako
  - python-netaddr

- python-six
- gperftools-libs
- libunwind
- snappy
- boost-date-time
- c-ares
- redhat-lsb-core
- wget
- net-tools
- yum-utils

- For Red Hat, run the following commands:

```
yum groupinstall “Virtualization Hypervisor”
yum groupinstall “Virtualization Client”
yum groupinstall "Virtualization Platform"
yum groupinstall "Virtualization Tools"
```

  If you are not able to install the packages, you can manually install them with the command `yum install glibc.i686 nspr` on a new Red Hat installation.

Add a Hypervisor Host to the NSX-T Fabric

A fabric node is a node that has been registered with the NSX-T management plane and has NSX-T modules installed. For a hypervisor host to be part of the NSX-T overlay, it must first be added to the NSX-T fabric.

Note You can skip this procedure if you installed the modules on the hosts manually and joined the hosts to the management plane using the CLI.

Prerequisites

- For each host that you plan to add to the NSX-T fabric, first gather the following host information:
  - Hostname
  - Management IP address
  - Username
  - Password
  - (KVM) SHA-256 SSL thumbprint
  - (ESXi) SHA-256 SSL thumbprint

- Optionally, retrieve the hypervisor thumbprint so that you can provide it when adding the host to the fabric.
  - One method to gather the information is to run the following command in a Linux shell:

    ```
    # echo -n | openssl s_client -connect <esxi-ip-address>:443 2>/dev/null | openssl x509 -noout -fingerprint -sha256
    ```

  - Another method uses the ESXi CLI in the ESX host:

    ```
    [root@host:~] openssl x509 -in /etc/vmware/ssl/rui.crt -fingerprint -sha256 -noout
    SHA256 Fingerprint=49:73:F9:A6:0B:EA:51:2A:15:57:90:DE:C0:89:CA:7F:46:8E:30:15:CA:4D:5C:
    95:28:0A:9E:A2:4E:3C:C4:F4
    ```

  - To retrieve the SHA-256 thumbprint from a KVM hypervisor, run the command in the KVM host.

    ```
    # awk '{print $2}' /etc/ssh/ssh_host_rsa_key.pub | base64 -d | sha256sum -b | sed 's/ .*$//' |
    xxd -r -p | base64
    ```

- For Ubuntu, verify that the required third-party packages are installed. See Install Third-Party Packages on a KVM Host.

Procedure

1. In the NSX Manager CLI, verify that the install-upgrade service is running.

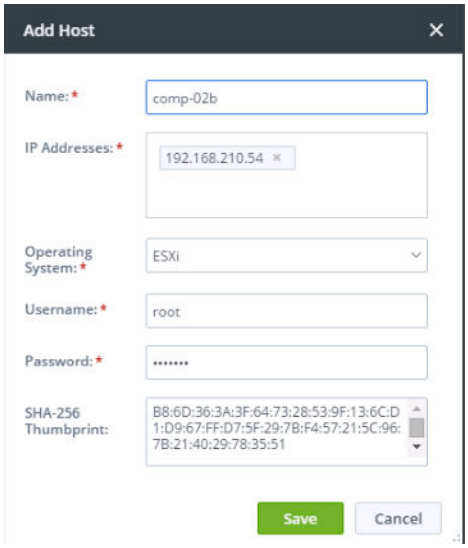
   ```
   nsx-manager-1> get service install-upgrade

   Service name: install-upgrade
   Service state: running
   Enabled: True
   ```

2. From a browser, log in to an NSX Manager at https://<nsx-manager-ip-address>.
3. Select **Fabric > Nodes > Hosts** and click **Add**.

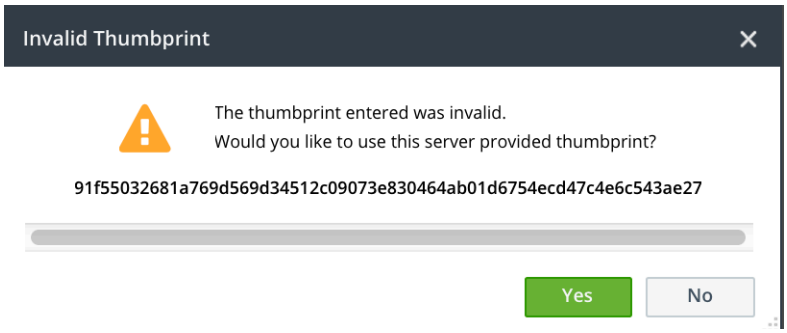
4 Enter the hostname, IP address, username, password, and the optional thumbprint.

For example:



If you do not enter the host thumbprint, the NSX-T UI prompts you to use the default thumbprint in the plain text format retrieved from the host.

For example:



When a host is successfully added to the NSX-T fabric, the NSX Manager **Fabric > Nodes > Hosts** UI displays **Deployment Status: Installation Successful** and **MPA Connectivity: Up**. **LCP Connectivity** remains unavailable until after you have made the fabric node into a transport node.

As a result of adding a host to the NSX-T fabric, a collection of NSX-T modules are installed on the host. On ESXi, the modules are packaged as VIBs. For KVM on RHEL, they are packaged as RPMs. For KVM on Ubuntu, they are packaged as DEBs.

To verify on ESXi, you can run the `esxcli software vib list | grep nsx` command, where the date is the day that you performed the installation.

To verify on RHEL, run the `yum list installed` or `rpm –qa` command.

To verify on Ubuntu, run the `dpkg ––get–selections` command.

You can view the fabric nodes with the GET https://<nsx-mgr>/api/v1/fabric/nodes/<node-id> API call:


```
{
"resource_type" : "HostNode",
"id" : "69b2a1d3-778d-4835-83c5-94cee99a213e",
"display_name" : "10.143.1.218",
"fqdn" : "w1-mvpcloud-218.eng.vmware.com",
"ip_addresses" : [ "10.143.1.218" ],
"external_id" : "69b2a1d3-778d-4835-83c5-94cee99a213e",
"discovered_ip_addresses" : [ "10.143.1.218" ],
"os_type" : "ESXI",
"os_version" : "6.5.0",
"managed_by_server" : "",
"_create_user" : "admin",
"_create_time" : 1498155416694,
"_last_modified_user" : "admin",
"_last_modified_time" : 1498155416694,
"_protection" : "NOT_PROTECTED",
"_revision" : 0
}
```


You can monitor the status in the API with the GET https://<nsx-mgr>/api/v1/fabric/nodes/<node-id>/status API call.


```
{
  "lcp_connectivity_status" : "UP",
  "mpa_connectivity_status" : "UP",
  "last_sync_time" : 1480370899198,
  "mpa_connectivity_status_details" : "Client is responding to heartbeats",
  "lcp_connectivity_status_details" : [ {
    "control_node_ip" : "10.143.1.47",
    "status" : "UP"
  } ],
  "inventory_sync_paused" : false,
  "last_heartbeat_timestamp" : 1480369333415,
  "system_status" : {
    "mem_used" : 2577732,
    "system_time" : 1480370897000,
    "file_systems" : [ {
      "file_system" : "root",
      "total" : 32768,
      "used" : 5440,
      "type" : "ramdisk",
      "mount" : "/"
    }, {
      "file_system" : "etc",
      "total" : 28672,
      "used" : 264,
      "type" : "ramdisk",
      "mount" : "/etc"
    }, {
      "file_system" : "opt",
```

```
      "total" : 32768,
      "used" : 20,
      "type" : "ramdisk",
      "mount" : "/opt"
    }, {
      "file_system" : "var",
      "total" : 49152,
      "used" : 2812,
      "type" : "ramdisk",
      "mount" : "/var"
    }, {
      "file_system" : "tmp",
      "total" : 262144,
      "used" : 21728,
      "type" : "ramdisk",
      "mount" : "/tmp"
    }, {
      "file_system" : "iofilters",
      "total" : 32768,
      "used" : 0,
      "type" : "ramdisk",
      "mount" : "/var/run/iofilters"
    }, {
      "file_system" : "hostdstats",
      "total" : 116736,
      "used" : 2024,
      "type" : "ramdisk",
      "mount" : "/var/lib/vmware/hostd/stats"
    } ],
    "load_average" : [ 0.03999999910593033, 0.03999999910593033, 0.05000000074505806 ],
    "swap_total" : 0,
    "mem_cache" : 0,
    "cpu_cores" : 2,
    "source" : "cached",
    "mem_total" : 8386740,
    "swap_used" : 0,
    "uptime" : 3983605000
  },
  "software_version" : "2.0.0.0.0.4649755",
  "host_node_deployment_status" : "INSTALL_SUCCESSFUL"
}
```


What to do next

If you have a large number of hypervisors (for example, 500 or more), NSX Manager might experience high CPU usage and performance problems. You can avoid the problem by running the script `aggsvc_change_intervals.py`, which is located in the NSX file store. (You can use the NSX CLI command `copy file` or the API `POST /api/v1/node/file-store/<file-name>?action=copy_to_remote_file` to copy the script to a host.) This script changes the polling intervals of certain processes. Run the script as follows:

```
python aggsvc_change_intervals.py -m '<NSX Manager IP address>' -u 'admin' -p '<password>' -i 900
```

To change the polling intervals back to their default values:

```
python aggsvc_change_intervals.py –m '<NSX Manager IP address>' –u 'admin' –p '<password>' –r
```

Create a transport zone. See About Transport Zones.

Manual Installation of NSX-T Kernel Modules

As an alternative to using the NSX-T **Fabric > Nodes > Hosts > Add** UI or the `POST /api/v1/fabric/nodes` API, you can install NSX-T kernel modules manually from the hypervisor command line.

Manually Install NSX-T Kernel Modules on ESXi Hypervisors

To prepare hosts to participate in NSX-T, you must install NSX-T kernel modules on ESXi hosts. This allows you to build the NSX-T control-plane and management-plane fabric. NSX-T kernel modules packaged in VIB files run within the hypervisor kernel and provide services such as distributed routing, distributed firewall, and bridging capabilities.

You can download the NSX-T VIBs manually and make them part of the host image. The download paths can change for each release of NSX-T. Always check the NSX-T downloads page to get the appropriate VIBs.

Procedure

1. Log in to the host as root or as a user with administrative privileges
2. Navigate to the /tmp directory.

   ```
   [root@host:~]: cd /tmp
   ```

3. Download and copy the nsx-lcp file into the /tmp directory.
4. Run the install command.

   ```
   [root@host:/tmp]: esxcli software vib install –d /tmp/nsx–lcp–<release>.zip
   Installation Result
      Message: Operation finished successfully.
      Reboot Required: false
      VIBs Installed: VMware_bootbank_nsx–aggservice_<release>, VMware_bootbank_nsx–da_<release>,
   VMware_bootbank_nsx–esx–datapath_<release>, VMware_bootbank_nsx–exporter_<release>,
   VMware_bootbank_nsx–host_<release>, VMware_bootbank_nsx–lldp_<release>, VMware_bootbank_nsx–
   mpa_<release>, VMware_bootbank_nsx–netcpa_<release>, VMware_bootbank_nsx–python–
   protobuf_<release>, VMware_bootbank_nsx–sfhc_<release>, VMware_bootbank_nsxa_<release>,
   ```

```
VMware_bootbank_nsxcli_<release>
   VIBs Removed:
   VIBs Skipped:
```

Depending on what was already installed on the host, some VIBs might be installed, some might be removed, and some might be skipped. A reboot is not required unless the command output says `Reboot Required: true`.

As a result of adding an ESXi host to the NSX-T fabric, the following VIBs get installed on the host.

- nsx-aggservice—Provides host-side libraries for NSX-T aggregation service. NSX-T aggregation service is a service that runs in the management-plane nodes and fetches runtime state from NSX-T components.
- nsx-da—Collects discovery agent (DA) data about the hypervisor OS version, virtual machines, and network interfaces. Provides the data to the management plane, to be used in troubleshooting tools.
- nsx-esx-datapath—Provides NSX-T data plane packet processing functionality.
- nsx-exporter—Provides host agents that report runtime state to the aggregation service running in the management plane.
- nsx-host— Provides metadata for the VIB bundle that is installed on the host.
- nsx-lldp—Provides support for the Link Layer Discovery Protocol (LLDP), which is a link layer protocol used by network devices for advertising their identity, capabilities, and neighbors on a LAN.
- nsx-mpa—Provides communication between NSX Manager and hypervisor hosts.
- nsx-netcpa—Provides communication between the central control plane and hypervisors. Receives logical networking state from the central control plane and programs this state in the data plane.
- nsx-python-protobuf—Provides Python bindings for protocol buffers.
- nsx-sfhc—Service fabric host component (SFHC). Provides a host agent for managing the lifecycle of the hypervisor as a fabric host in the management plane's inventory. This provides a channel for operations such as NSX-T upgrade and uninstall and monitoring of NSX-T modules on hypervisors.
- nsxa—Performs host-level configurations, such as N-VDS creation and uplink configuration.
- nsxcli—Provides the NSX-T CLI on hypervisor hosts.
- nsx-support-bundle-client - Provides the ability to collect support bundles.

To verify, you can run the **esxcli software vib list | grep nsx** or **esxcli software vib list | grep <yyyy-mm-dd>** command on the ESXi host, where the date is the day that you performed the installation.

What to do next

Add the host to the NSX-T management plane. See Join the Hypervisor Hosts with the Management Plane.

Manually Install NSX-T Kernel Modules on Ubuntu KVM Hypervisors

To prepare hosts to participate in NSX-T, you can manually install NSX-T kernel modules on Ubuntu KVM hosts. This allows you to build the NSX-T control-plane and management-plane fabric. NSX-T kernel modules packaged in DEB files run within the hypervisor kernel and provide services such as distributed routing, distributed firewall, and bridging capabilities.

You can download the NSX-T DEBs manually and make them part of the host image. Be aware that download paths can change for each release of NSX-T. Always check the NSX-T downloads page to get the appropriate DEBs.

Prerequisites

- Verify that the required third-party packages are installed. See Install Third-Party Packages on a KVM Host.

Procedure

1. Log in to the host as a user with administrative privileges.
2. (Optional) Navigate to the /tmp directory.

   ```
   cd /tmp
   ```

3. Download and copy the nsx-lcp file into the /tmp directory.
4. Untar the package.

   ```
   tar –xvf nsx-lcp-<release>-ubuntu-trusty_amd64.tar.gz
   ```

5. Navigate to the package directory.

   ```
   cd nsx-lcp-trusty_amd64/
   ```

6. Install the packages.

   ```
   sudo dpkg –i *.deb
   ```

7. Reload the OVS kernel module.

   ```
   /etc/init.d/openvswitch-switch force-reload-kmod
   ```

   If the hypervisor uses DHCP on OVS interfaces, restart the network interface on which DHCP is configured. You can manually stop the old dhclient process on the network interface and restart a new dhclient process on that interface.

8 To verify, you can run the `dpkg -l | grep nsx` command.

```
user@host:~$ dpkg -l | grep nsx

ii  nsx-agent                            <release>     amd64        NSX Agent
ii  nsx-aggservice                       <release>     all          NSX Aggregation Service Lib
ii  nsx-cli                              <release>     all          NSX CLI
ii  nsx-da                               <release>     amd64        NSX Inventory Discovery Agent
ii  nsx-host                             <release>     all          NSX host meta package
ii  nsx-host-node-status-reporter        <release>     amd64        NSX Host Status Reporter for
Aggregation Service
ii  nsx-lldp                             <release>     amd64        NSX LLDP Daemon
ii  nsx-logical-exporter                 <release>     amd64        NSX Logical Exporter
ii  nsx-mpa                              <release>     amd64        NSX Management Plane Agent Core
ii  nsx-netcpa                           <release>     amd64        NSX Netcpa
ii  nsx-sfhc                             <release>     amd64        NSX Service Fabric Host
Component
ii  nsx-transport-node-status-reporter   <release>     amd64        NSX Transport Node Status
Reporter
ii  nsxa                                 <release>     amd64        NSX L2 Agent
```

Any errors are most likely caused by incomplete dependencies. The `apt-get install -f` command can attempt to resolve dependencies and re-run the NSX-T installation.

What to do next

Add the host to the NSX-T management plane. See Join the Hypervisor Hosts with the Management Plane.

Verify Open vSwitch Version

You must have the supported Open vSwitch version before you manually install the NSX-T kernel modules on RHEL KVM hosts to avoid any errors.

The supported Open vSwitch version is 2.7.0.6814985-1.

Procedure

1 Verify the current version of your Open vSwitch.

If you have a Open vSwitch newer or older version, you must replace that Open vSwitch version with the supported one.

2 Open the Open vSwitch folder.

3 Remove the existing Open vSwitch packages.

- kmod-openvswitch
- nicira-ovs-hypervisor-node
- openvswitch
- openvswitch-selinux-policy

4 Replace existing Open vSwitch version with the supported one.

- For newer Open vSwitch version, use the `--nodeps` command.

  For example, rpm-Uvh kmod-openvswitch-<new version>.e17.x86_64.rpm --nodeps

  rpm-Uvh nicira-ovs-hypervisor-node-<new version>.x86_64.rpm --nodeps

  rpm-Uvh openvswitch-*.rpm --nodeps

- For older Open vSwitch version, use the `--force` command.

  For example, rpm-Uvh kmod-openvswitch-<new version>.e17.x86_64.rpm --nodeps --force

  rpm-Uvh nicira-ovs-hypervisor-node-<new version>.x86_64.rpm --nodeps --force

  rpm-Uvh openvswitch-*.rpm --nodeps --force

What to do next

Install the NSX-T kernel modules on RHEL KVM hosts. See Manually Install NSX-T Kernel Modules on RHEL KVM Hypervisors.

Manually Install NSX-T Kernel Modules on RHEL KVM Hypervisors

To prepare hosts to participate in NSX-T, you can manually install NSX-T kernel modules on RHEL KVM hosts. This allows you to build the NSX-T control-plane and management-plane fabric. NSX-T kernel modules packaged in RPM files run within the hypervisor kernel and provide services such as distributed routing, distributed firewall, and bridging capabilities.

You can download the NSX-T RPMs manually and make them part of the host image. Be aware that download paths can change for each release of NSX-T. Always check the NSX-T downloads page to get the appropriate RPMs.

Prerequisites

Ability to reach a RHEL repository.

Procedure

1 Log in to the host as an administrator.

2 Download and copy the nsx-lcp file into the /tmp directory.

3 Untar the package.

```
tar -zxvf nsx-lcp-<release>-rhel73_x86_64.tar.gz
```

4 Navigate to the package directory.

```
cd nsx-lcp-rhel73_x86_64/
```

5 Install the packages.

```
sudo yum install *.rpm
```

When you run the yum install command, any NSX-T dependencies are resolved, assuming the RHEL host can reach the RHEL repository.

6 Reload the OVS kernel module.

```
/etc/init.d/openvswitch force-reload-kmod
```

If the hypervisor uses DHCP on OVS interfaces, restart the network interface on which DHCP is configured. You can manually stop the old dhclient process on the network interface and restart a new dhclient process on that interface.

7 To verify, you can run the `rpm -qa | egrep 'nsx|openvswitch|nicira'` command.

The installed packages in the output must match the packages in the nsx-rhel73 directory.

What to do next

Add the host to the NSX-T management plane. See Join the Hypervisor Hosts with the Management Plane.

Join the Hypervisor Hosts with the Management Plane

Joining the hypervisor hosts with the management plane ensures that the NSX Manager and the hosts can communicate with each other.

Prerequisites

The installation of NSX-T modules must be complete.

Procedure

1 Open an SSH session to the NSX Manager appliance.

2 Log in with the Administrator credentials.

3 Open an SSH session to the hypervisor host.

4 On the NSX Manager appliance, run the `get certificate api thumbprint` cli command.

The command output is a string of numbers that is unique to this NSX Manager.

For example:

```
NSX-Manager1> get certificate api thumbprint
...
```

5 On the hypervisor host, run the **nsxcli** command to enter the NSX-T CLI.

Note For KVM, run the command as a superuser (sudo).

```
[user@host:~] nsxcli
host>
```

The prompt changes.

6 On the hypervisor host, run the **join management-plane** command.

Provide the following information:

- Hostname or IP address of the NSX Manager with an optional port number
- Username of the NSX Manager
- Certificate thumbprint of the NSX Manager
- Password of the NSX Manager

```
host> join management-plane NSX-Manager1 username admin thumbprint <NSX-Manager1's-thumbprint>
Password for API user: <NSX-Manager1's-password>
Node successfully joined
```

Verify the result by running the `get managers` command on your hosts.

```
host> get managers
- 192.168.110.47   Connected
```

In the NSX Manager UI in **Fabric > Node > Hosts**, verify that the host's MPA connectivity is **Up**.

You can view the fabric host's state with the **GET https://<nsx-mgr>/api/v1/fabric/nodes/<fabric-node-id>/state** API call:

```
{
  "details": [],
  "state": "success"
}
```

The management plane sends the host certificates to the control plane, and the management plane pushes control plane information to the hosts.

You should see NSX Controller addresses in `/etc/vmware/nsx/controller-info.xml` on each ESXi host.

```
[root@host:~] cat /etc/vmware/nsx/controller-info.xml
<?xml version="1.0" encoding="utf-8"?>
<config>
  <connectionList>
    <connection id="0">
        <server>10.143.1.47</server>
        <port>1234</port>
```

```
        <sslEnabled>true</sslEnabled>
        <pemKey>-----BEGIN CERTIFICATE-----...-----END CERTIFICATE-----</pemKey>
    </connection>
    <connection id="1">
        <server>10.143.1.45</server>
        <port>1234</port>
        <sslEnabled>true</sslEnabled>
        <pemKey>-----BEGIN CERTIFICATE-----...-----END CERTIFICATE-----</pemKey>
    </connection>
    <connection id="2">
        <server>10.143.1.46</server>
        <port>1234</port>
        <sslEnabled>true</sslEnabled>
        <pemKey>-----BEGIN CERTIFICATE-----...-----END CERTIFICATE-----</pemKey>
    </connection>
  </connectionList>
</config>
```

The host connection to NSX-Ts is initiated and sits in "CLOSE_WAIT" status until the host is promoted to a transport node. You can see this with the **esxcli network ip connection list | grep 1234** command.

```
# esxcli network ip connection list | grep 1234
tcp         0       0  192.168.210.53:45823        192.168.110.34:1234  CLOSE_WAIT    37256  newreno
netcpa
```

For KVM, the command is `netstat –anp ––tcp | grep 1234`.

```
user@host:~$ netstat –anp ––tcp | grep 1234
tcp  0   0 192.168.210.54:57794  192.168.110.34:1234   CLOSE_WAIT –
```

What to do next

Create a transport zone. See About Transport Zones.

Transport Zones and Transport Nodes

9

Transport zones and transport nodes are important concepts in NSX-T.

This chapter includes the following topics:

About Transport Zones

A transport zone is a container that defines the potential reach of transport nodes. Transport nodes are hypervisor hosts and NSX Edges that will participate in an NSX-T overlay. For a hypervisor host, this means that it hosts VMs that will communicate over NSX-T logical switches. For NSX Edges, this means that it will have logical router uplinks and downlinks.

If two transport nodes are in the same transport zone, VMs hosted on those transport nodes can be attached to NSX-T logical switches that are also in that transport zone. This attachment makes it possible for the VMs to communicate with each other, assuming that the VMs have Layer 2/Layer 3 reachability. If VMs are attached to switches that are in different transport zones, the VMs cannot communicate with each other. Transport zones do not replace Layer 2/Layer 3 underlay reachability requirements, but they place a limit on reachability. Put another way, belonging to the same transport zone is a prerequisite for connectivity. After that prerequisite is met, reachability is possible but not automatic. To achieve actual reachability, Layer 2 and (for different subnets) Layer 3 underlay networking must be operational.

Transport nodes can be hypervisor hosts or NSX Edges. NSX Edges can belong to multiple transport zones. Hypervisor hosts (and NSX-T logical switches) can belong to only one transport zone.

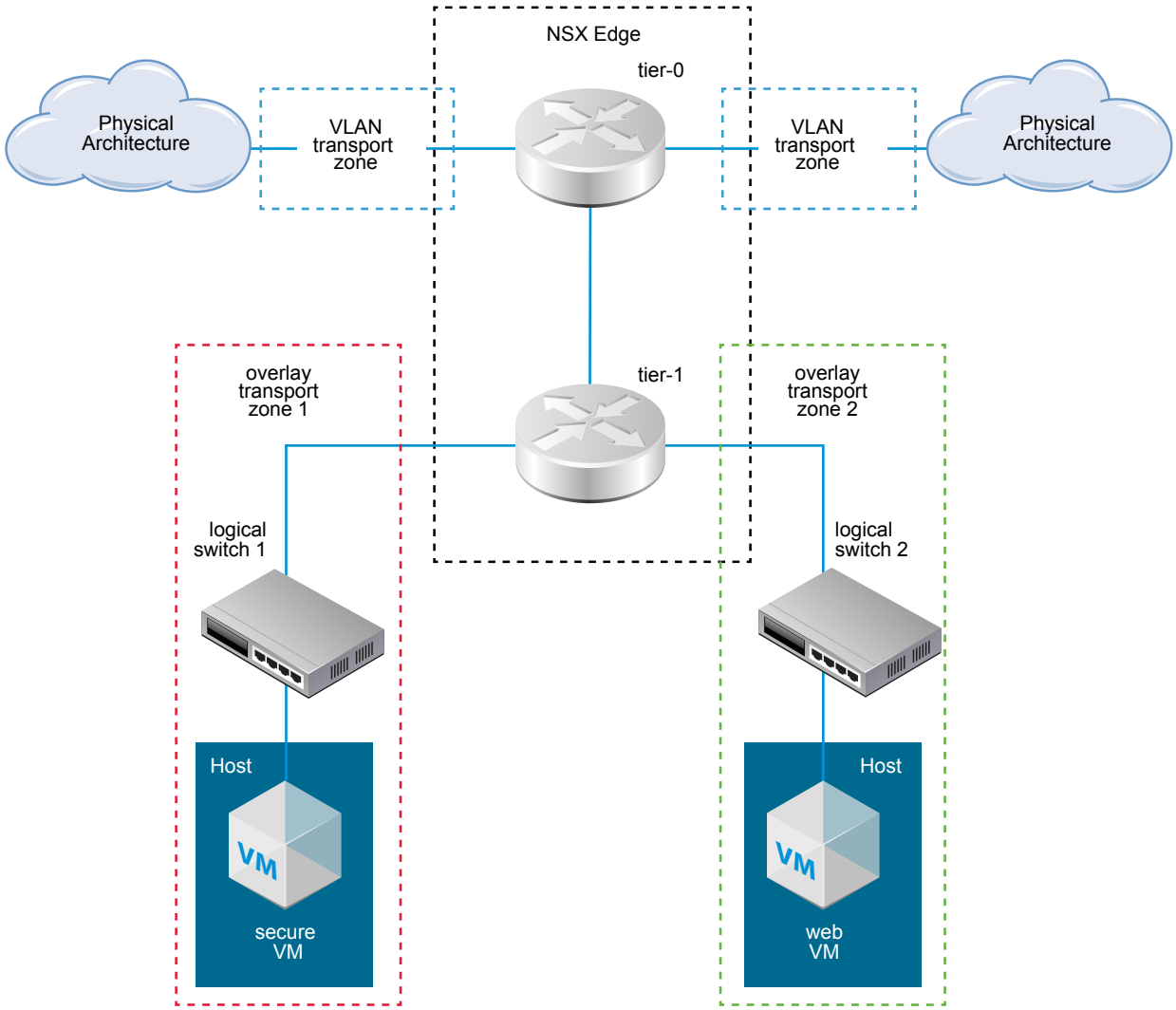
Suppose a single transport node contains both regular VMs and high-security VMs. In your network design, the regular VMs should be able to reach each other but should not be able to reach the high-security VMs. To accomplish this goal, you can place the secure VMs on hosts that belong to one transport zone named secure-tz. The regular VMs would then be on a different transport zone called

general-tz. The regular VMs attach to an NSX-T logical switch that is also in general-tz. The high-security VMs attach to an NSX-T logical switch that is in the secure-tz. The VMs in different transport zones, even if they are in the same subnet, cannot communicate with each other. The VM-to-logical switch connection is what ultimately controls VM reachability. Thus, because two logical switches are in separate transport zones, "web VM" and "secure VM" cannot reach each other.

An NSX Edge transport node can belong to multiple transport zones: One overlay transport zone and multiple VLAN transport zones. VLAN transport zones are for the VLAN uplinks to the outside world.

For example, the following figure shows an NSX Edge that belongs to three transport zones: two VLAN transport zones and overlay transport zone 2. Overlay transport zone 1 contains a host, an NSX-T logical switch, and a secure VM. Because the NSX Edge does not belong to overlay transport zone 1, the secure VM has no access to or from the physical architecture. In contrast, the Web VM in overlay transport zone 2 can communicate with the physical architecture because the NSX Edge belongs to overlay transport zone 2.

Figure 9-1. NSX-T Transport Zones



Create an IP Pool for Tunnel Endpoint IP Addresses

You can use an IP pool for the tunnel endpoints. Tunnel endpoints are the source and destination IP addresses used in the external IP header to uniquely identify the hypervisor hosts originating and terminating the NSX-T encapsulation of overlay frames. You can use either DHCP or manually configured IP pools for tunnel endpoint IP addresses.

If you are using both ESXi and KVM hosts, one design option would be to use two different subnets for the ESXi tunnel endpoint IP pool (sub_a) and the KVM tunnel endpoint IP Pool (sub_b). In this case, on the KVM hosts a static route to sub_a needs to be added with a dedicated default gateway.

This is an example of the resulting routing table on an Ubuntu host where sub_a = 192.168.140.0 and sub_b = 192.168.150.0. (The management subnet, for example, could be 192.168.130.0.)

Kernel IP routing table:

```
Destination          Gateway          Genmask          Iface
0.0.0.0              192.168.130.1    0.0.0.0          eth0
192.168.122.0        0.0.0.0          255.255.255.0    virbr0
192.168.130.0        0.0.0.0          255.255.255.0    eth0
192.168.140.0        192.168.150.1    255.255.255.0    nsx-vtep0.0
192.168.150.0        0.0.0.0          255.255.255.0    nsx-vtep0.0
```

The route can be added in at least two different ways.

```
route add -net 192.168.140.0 netmask 255.255.255.0 gw 192.168.150.1 dev nsx-vtep0.0
```

In /etc/network/interfaces before "up ifconfig nsx-vtep0.0 up" add this static route:

```
post-up route add -net 192.168.140.0 netmask 255.255.255.0 gw 192.168.150.1
```

Procedure

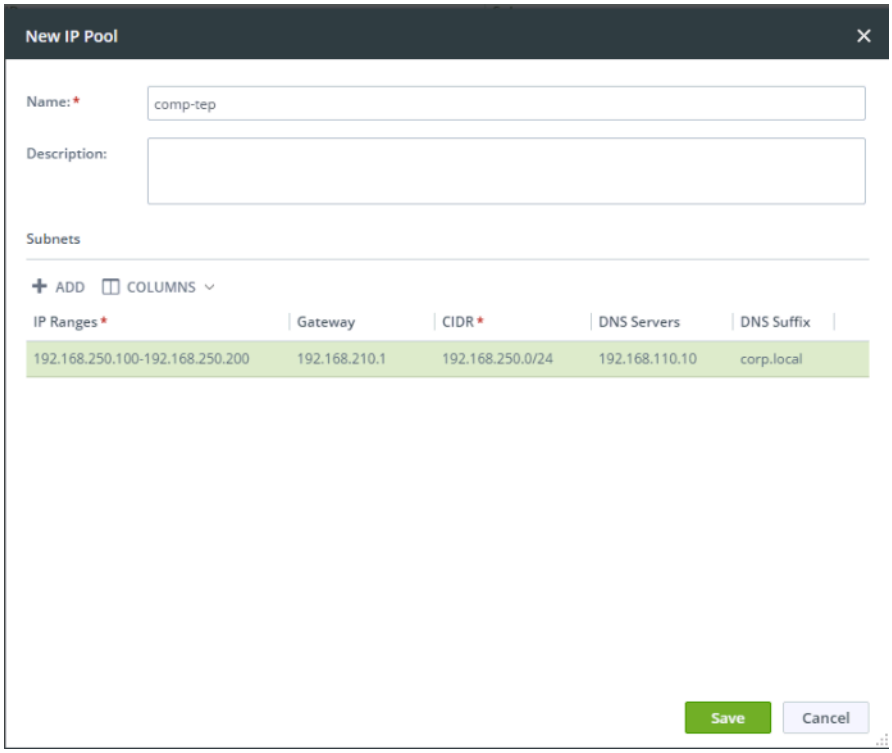
1. From a browser, log in with admin privileges to an NSX Manager at `https://<nsx-manager-ip-address>`.
2. Select **Inventory > IP Pools** and click **Add**.
3. Enter the name of the IP pool, an optional description, and the network settings.

   The network settings include:

   - Range of IP addresses
   - Gateway
   - Network address in CIDR notation
   - (optional) Comma-separated list of DNS servers

- (optional) DNS suffix

For example:



You can view the IP pools with the `GET https://<nsx-mgr>/api/v1/pools/ip-pools` API call:

```
{
  "cursor": "0036e2d8c2e8-f6d7-498e-821b-b7e44d2650a9ip-pool-1",
  "sort_by": "displayName",
  "sort_ascending": true,
  "result_count": 1,
  "results": [
    {
      "id": "e2d8c2e8-f6d7-498e-821b-b7e44d2650a9",
      "display_name": "comp-tep",
      "resource_type": "IpPool",
      "subnets": [
        {
          "dns_nameservers": [
            "192.168.110.10"
          ],
          "allocation_ranges": [
            {
              "start": "192.168.250.100",
              "end": "192.168.250.200"
            }
          ],
          "gateway_ip": "192.168.250.1",
          "cidr": "192.168.250.0/24",
          "dns_suffix": "corp.local"
```

```
        }
      ],
      "_last_modified_user": "admin",
      "_last_modified_time": 1443649891178,
      "_create_time": 1443649891178,
      "_system_owned": false,
      "_create_user": "admin",
      "_revision": 0
    }
  ]
}
```


What to do next

Create an uplink profile. See Create an Uplink Profile.

Create an Uplink Profile

An uplink profile defines policies for the links from hypervisor hosts to NSX-T logical switches or from NSX Edge nodes to top-of-rack switches.

The settings defined by uplink profiles may include teaming policies, active/standby links, the transport VLAN ID, and the MTU setting.

Uplink profiles allow you to consistently configure identical capabilities for network adapters across multiple hosts or nodes. Uplink profiles are containers for the properties or capabilities that you want your network adapters to have. Instead of configuring individual properties or capabilities for each network adapter, you can specify the capabilities in uplink profiles, which you can then apply when you create NSX-T transport nodes.

You can use the default uplink profile if the NSX Edge installed on bare metal has one active uplink and one passive standby uplink. You can also create custom uplink profile for NSX Edge installed on bare metal.

Standby uplinks are not supported with VM/appliance-based NSX Edge. When you install NSX Edge as a virtual appliance, you must create a custom uplink profile rather than use the default uplink profile. For each uplink profile created for a VM-based NSX Edge, the profile must specify only one active uplink and no standby uplink.

Note NSX Edge VMs do allow for multiple uplinks, if you create a separate N-VDS for each uplink, using a different VLAN for each. This is to support a single NSX Edge node that connects to multiple TOR switches.

Prerequisites

- Familiarize yourself with NSX Edge networking. See NSX Edge Networking Setup.
- Each uplink must correspond to an up and available physical link on your hypervisor host or on the NSX Edge node.

For example, your hypervisor host has two physical links that are up: vmnic0 and vmnic1. Suppose vmnic0 is used for management and storage networks, while vmnic1 is unused. This would mean that vmnic1 can be used as an NSX-T uplink, but vmnic0 cannot. To do link teaming, you must have two unused physical links available, such as vmnic1 and vmnic2.

For an NSX Edge, tunnel endpoint and VLAN uplinks can use the same physical link. For example, vmnic0/eth0/em0 could be used for your management network and vmnic1/eth1/em1 could be used for your fp-ethX links.

Procedure

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

2 Select **Fabric > Profiles > Uplink Profiles** and click **Add**.

3 Complete the uplink profile details.

Option	Description
Name	Enter an uplink profile name.
Description	Add an optional uplink profile description.
Teaming policy	Select Failover order or load balance source from the drop-down menu. The default setting is failover order. Failover order - From the list of active adapters, always use the highest order uplink that passes failover detection criteria. No actual load balancing is performed with this option. **Note** For KVM hosts, load balance source teaming policy is not supported. Failover order teaming policy is supported on KVM hosts. Load balance source - Select an uplink based on a hash of the source Ethernet MAC address.
LAGs	(Optional) Link aggregation groups (LAGs) using Link Aggregation Control Protocol (LACP) for the transport network. **Note** For LACP, multiple LAG is not supported on KVM hosts. Add a comma-separated list of active uplink names. Add a comma-separated list of standby uplink names. The active and standby uplink names you create can be any text to represent physical links. These uplink names are referenced later when you create transport nodes. The transport node UI/API allows you to specify which physical link corresponds to each named uplink.

4 Enter a Transport VLAN value.

5 Enter the MTU value.

The default value is 1600.

You can view the uplink profiles with the `GET /api/v1/host-switch-profiles` API call:


```
{
  "result_count": 2,
  "results": [
    {
      "resource_type": "UplinkHostSwitchProfile",
      "id": "16146a24-122b-4274-b5dd-98b635e4d52d",
      "display_name": "comp-uplink",
      "transport_vlan": 250,
      "teaming": {
        "active_list": [
          {
            "uplink_type": "PNIC",
            "uplink_name": "uplink-1"
          }
        ],
        "standby_list": [],
        "policy": "FAILOVER_ORDER"
      },
      "mtu": 1600,
      "_last_modified_time": 1457984399526,
      "_create_time": 1457984399526,
      "_last_modified_user": "admin",
      "_system_owned": false,
      "_create_user": "admin",
      "_revision": 0
    },
    {
      "resource_type": "UplinkHostSwitchProfile",
      "id": "c9e35cec-e9d9-4c51-b52e-17a5c1bd9a38",
      "display_name": "vlan-uplink",
      "transport_vlan": 100,
      "teaming": {
        "active_list": [
          {
            "uplink_type": "PNIC",
            "uplink_name": "uplink-1"
          }
        ],
        "standby_list": [],
        "policy": "FAILOVER_ORDER"
      },
      "mtu": 1600,
      "_last_modified_time": 1457984399574,
      "_create_time": 1457984399574,
      "_last_modified_user": "admin",
      "_system_owned": false,
      "_create_user": "admin",
      "_revision": 0
    }
  ]
}
```

What to do next

Create a transport zone. See Create Transport Zones.

Create Transport Zones

Transport zones dictate which hosts and, therefore, which VMs can participate in the use of a particular network. A transport zone does this by limiting the hosts that can "see" a logical switch—and, therefore, which VMs can be attached to the logical switch. A transport zone can span one or more host clusters.

An NSX-T environment can contain one or more transport zones based on your requirements. A host can belong to multiple transport zones. A logical switch can belong to only one transport zone.

NSX-T does not allow connection of VMs that are in different transport zones in the Layer 2 network. The span of a logical switch is limited to a transport zone, so virtual machines in different transport zones cannot be on the same Layer 2 network.

The overlay transport zone is used by both host transport nodes and NSX Edges. When a host or NSX Edge transport node is added to an overlay transport zone, an N-VDS is installed on the host or NSX Edge.

The VLAN transport zone is used by the NSX Edge for its VLAN uplinks. When an NSX Edge is added to a VLAN transport zone, a VLAN N-VDS is installed on the NSX Edge.

The N-VDS allow for virtual-to-physical packet flow by binding logical router uplinks and downlinks to physical NICs.

When you create a transport zone, you must provide a name for the N-VDS that will be installed on the transport nodes when they are later added to this transport zone. The N-VDS name can be whatever you want it to be.

Procedure

1. From a browser, log in with admin privileges to an NSX Manager at `https://<nsx-manager-ip-address>`.
2. Select **Fabric > Transport Zones > Add**.
3. Enter a name for the transport zone, a N-VDS name, and the traffic type (overlay or VLAN).
4. (Optional) View the new transport zone with the `GET https://<nsx-mgr>/api/v1/transport-zones` API call.

```
{
  "cursor": "00369b661aed-1eaa-4567-9408-ccbcfe50b416tz-vlan",
  "result_count": 2,
  "results": [
    {
      "resource_type": "TransportZone",
      "description": "comp overlay transport zone",
      "id": "efd7f38f-c5da-437d-af03-ac598f82a9ec",
      "display_name": "tz-overlay",
      "host_switch_name": "overlay-hostswitch",
```

```
      "transport_type": "OVERLAY",
      "transport_zone_profile_ids": [
        {
          "profile_id": "52035bb3-ab02-4a08-9884-18631312e50a",
          "resource_type": "BfdHealthMonitoringProfile"
        }
      ],
      "_create_time": 1459547126454,
      "_last_modified_user": "admin",
      "_system_owned": false,
      "_last_modified_time": 1459547126454,
      "_create_user": "admin",
      "_revision": 0,
      "_schema": "/v1/schema/TransportZone"
    },
    {
      "resource_type": "TransportZone",
      "description": "comp vlan transport zone",
      "id": "9b661aed-1eaa-4567-9408-ccbcfe50b416",
      "display_name": "tz-vlan",
      "host_switch_name": "vlan-uplink-hostwitch",
      "transport_type": "VLAN",
      "transport_zone_profile_ids": [
        {
          "profile_id": "52035bb3-ab02-4a08-9884-18631312e50a",
          "resource_type": "BfdHealthMonitoringProfile"
        }
      ],
      "_create_time": 1459547126505,
      "_last_modified_user": "admin",
      "_system_owned": false,
      "_last_modified_time": 1459547126505,
      "_create_user": "admin",
      "_revision": 0,
      "_schema": "/v1/schema/TransportZone"
    }
  ]
}
```


What to do next

Optionally, create a custom transport-zone profile and bind it to the transport zone. You can create custom transport-zone profiles using the `POST /api/v1/transportzone-profiles` API. There is no UI workflow for creating a transport-zone profile. After the transport-zone profile is created, you can find it to the transport zone with the `PUT /api/v1/transport-zones/<transport-zone-id>` API.

Create a transport node. See Create a Host Transport Node.

Create a Host Transport Node

A transport node is a node that is capable of participating in an NSX-T overlay or NSX-T VLAN networking.

For a KVM host, you can preconfigure the N-VDS, or you can have NSX Manager perform the configuration. For an ESXi host, NSX Manager always configures the N-VDS.

Note If you plan to create transport nodes from a template VM, make sure that there are no certificates on the host in `/etc/vmware/nsx/`. The netcpa agent does not create a certificate if a certificate already exists.

Prerequisites

- The host must be joined with the management plane, and MPA connectivity must be Up on the **Fabric > Hosts** page.
- A transport zone must be configured.
- An uplink profile must be configured, or you can use the default uplink profile.
- An IP pool must be configured, or DHCP must be available in the network deployment.
- At least one unused physical NIC must be available on the host node.

Procedure

1. From a browser, log in with admin privileges to an NSX Manager at `https://<nsx-manager-ip-address>`.
2. Select **Fabric > Nodes > Transport Nodes > Add**.
3. Enter a name for the transport node.
4. Select a node from the drop-down menu.
5. (Optional) From the Available column, select one or more transport zones and click the right-arrow to move the zones to the Selected column.
6. Click the **N-VDS** tab.
7. For a KVM node, select the N-VDS type.

Option	Description
Standard	NSX Manager creates the N-VDS. This option is selected by default.
Preconfigured	The N-VDS is already configured.

   For a non-KVM node, the N-VDS type is always **Standard**.

8. For a standard N-VDS, provide the following details.

Option	Description
N-VDS Name	Must be the same as the N-VDS name of the transport zone that this node belongs to.
NIOC Profile	Select the NIOC profile from the drop-down menu.
Uplink Profile	Select the uplink profile from the drop-down menu.

Option	Description
IP Assignment	Select **Use DHCP**, **Use IP Pool**, or **Use Static IP List**. If you select **Use Static IP List**, you must specify a list of comma-separated IP addresses, a gateway, and a subnet mask.
IP Pool	If you selected **Use IP Pool** for IP assignment, specify the IP pool name.
Physical NICs	Make sure that the physical NIC is not already in use (for example, by a standard vSwitch or a vSphere distributed switch). Otherwise, the transport node state will be **partial success**, and the fabric node LCP connectivity will fail to establish.

9 For a preconfigured N-VDS, provide the following details.

Option	Description
N-VDS External ID	Must be the same as the N-VDS name of the transport zone that this node belongs to.
VTEP	Virtual tunnel endpoint name.

After adding the host as a transport node, the host connection to NSX Controllers changes to the Up status.

10 View the connection status.

- For ESXi, type the `esxcli network ip connection list | grep 1234` command.

```
# esxcli network ip connection list | grep 1234
tcp   0   0  192.168.210.53:20514  192.168.110.34:1234   ESTABLISHED  1000144459  newreno
netcpa
```

- For KVM, type the command `netstat –anp ––tcp | grep 1234`.

```
user@host:~$ netstat –anp ––tcp | grep 1234
tcp  0   0 192.168.210.54:57794  192.168.110.34:1234   ESTABLISHED –
```

11 (Optional) View the transport node with the `GET https://<nsx–mgr>/api/v1/transport–nodes/<node–id>` API call.

```
{
  "resource_type": "TransportNode",
  "description": "",
  "id": "95c8ce77–f895–43de–adc4–03a3ae2565e2",
  "display_name": "node–comp–01b",
  "tags": [],
  "transport_zone_endpoints": [
    {
      "transport_zone_id": "efd7f38f–c5da–437d–af03–ac598f82a9ec",
      "transport_zone_profile_ids": [
        {
          "profile_id": "52035bb3–ab02–4a08–9884–18631312e50a",
          "resource_type": "BfdHealthMonitoringProfile"
        }
      ]
```

```
    }
  ],
  "host_switches": [
    {
      "host_switch_profile_ids": [
        {
          "value": "8abdb6c0-db83-4e69-8b99-6cd85bfcc61d",
          "key": "UplinkHostSwitchProfile"
        },
        {
          "value": "9e0b4d2d-d155-4b4b-8947-fbfe5b79f7cb",
          "key": "LldpHostSwitchProfile"
        }
      ],
      "host_switch_name": "overlay-hostswitch",
      "pnics": [
        {
          "device_name": "vmnic1",
          "uplink_name": "uplink-1"
        }
      ],
      "static_ip_pool_id": "c78ac522-2a50-43fe-816a-c459a210127e"
    }
  ],
  "node_id": "c551290a-f682-11e5-ae84-9f8726e1de65",
  "_create_time": 1460051753373,
  "_last_modified_user": "admin",
  "_system_owned": false,
  "_last_modified_time": 1460051753373,
  "_create_user": "admin",
  "_revision": 0
}
```


12 Add the newly created transport node to a transport zone.

a Select the transport node.

b Select **Actions > Add to Transport Zone**.

c Select the transport zone from the drop-down menu.

All other fields are populated.

Note For a standard N-VDS, after the transport node is created, if you want to change the configuration, such as IP assignment, you must do it through the NSX Manager GUI and not through the CLI on the host.

What to do next

Migrate network interfaces from a vSphere Standard Switch to an NSX-T Virtual Distributed Switch. See Migrating Network Interfaces from a vSphere Standard Switch to an NSX-T Virtual Distributed Switch.

Migrating Network Interfaces from a vSphere Standard Switch to an NSX-T Virtual Distributed Switch

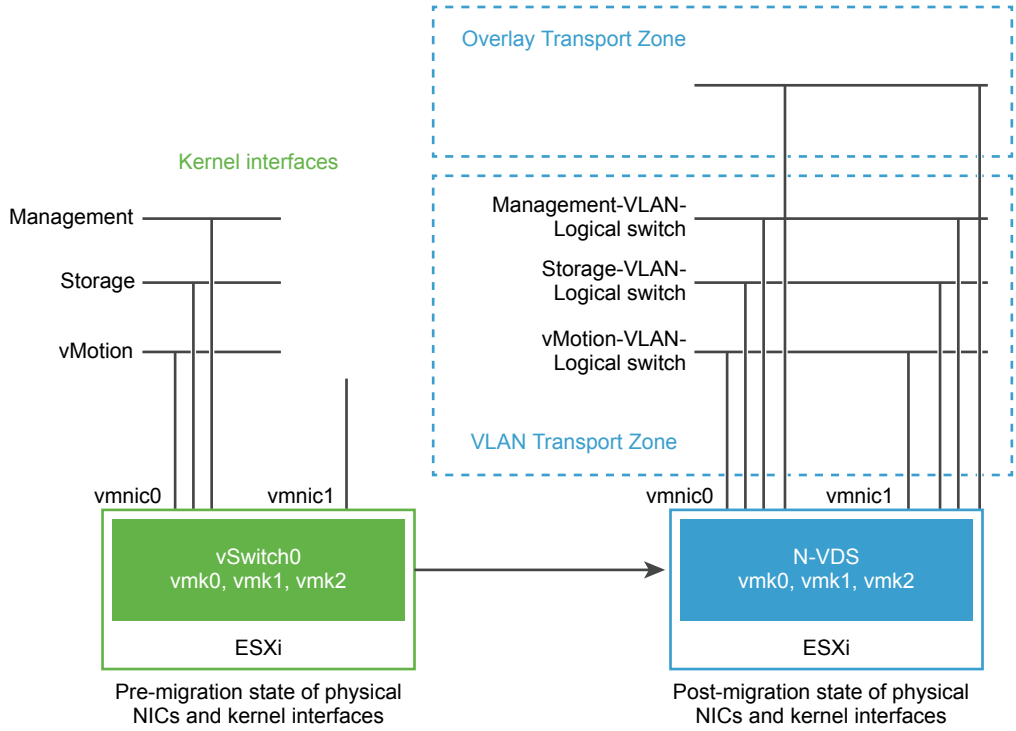
When you create a transport node, it might be necessary to migrate the physical NICs and kernel interfaces from a vSphere Standard Switch (VSS) to an NSX-T Virtual Distributed Switch (N-VDS).

A vSphere ESXi host includes single or multiple physical NICs. Kernel interfaces are defined on these hosts to provide connectivity to the management interface, storage, and other interfaces.

These physical NICs and their VMkernel interfaces (vmknics) are initially attached to a VSS.

For example, if a host only has two physical NICs, you might want to assign both those NICs to the N-VDS for redundancy. Then, the vmknic of that host must be migrated to this N-VDS.

Figure 9-2. Pre and Post Migration of Network Interfaces to an N-VDS



Before migration, the vSphere ESXi host has two uplinks derived from the two physical ports - vmnic0 and vmnic1. Here, vmnic0 is configured to be in an active state, attached to a VSS, whereas vmnic1 is unused. In addition, there are three vmknics: vmk0, vmk1, and vmk2.

You can use vSphere ESXi APIs to perform migration to N-VDS. See *NSX-T API Guide*.

Post migration, the vmnic0, vmnic1, and their vmknics are migrated to the N-VDS switch. Both vmnic0 and vmnic1 are connected over VLAN and the overlay transport zones.

Migrate VM Kernel Interfaces to an N-VDS

Configure the vSphere ESXi hypervisor as an NSX-T transport node with two uplinks. Assign only the vmnic1 configuration to the transport node at that stage. The vmnic0 configuration remains attached to the VSS.

Migrate the vmknics from the VSS to the N-VDS, using NSX-T API calls to the NSX Manager. See *NSX-T API Guide*. For example, you can migrate the storage kernel interface vmk1 to N-VDS.

Prerequisites

- Verify that the physical network infrastructure provides the same LAN connectivity to vmnic1 and vmnic0.
- Verify that the unused physical NIC, vmnic1, has Layer 2 connectivity with vmnic0.
- All vmknics involved in migration must belong to the same network. If you migrate vmknics to an uplink connected to a different network, the host might become unreachable or non-functional.

Procedure

1. Create a VLAN transport zone with the host_switch_name of the N-VDS used by the OVERLAY transport zone.
2. Create a VLAN-backed logical switch in the VLAN transport zone with a VLAN ID that matches the VLAN ID used by vmk1 on the VSS.
3. Add the vSphere ESXi transport node into the VLAN transport zone.
4. Retrieve the vSphere ESXi transport node configuration.

   ```
   GET /api/v1/transport-nodes/<transportnode-id>
   ```

   Where <*transportnode-id*> is the UUID of the transport node.
5. Migrate the storage kernel interface, vmk1 to N-VDS.

   ```
   PUT https://<NSXmgr>/api/v1/transport-nodes/<transportnode-id> ?
   if_id=<vmk>&esx_mgmt_if_migration_dest=<network>
   ```

   Where the <*transportnode-id*> is the UUID of the transport node. The <*vmk*> is the name of the vmknic, vmk1. The <*network*> is the UUID of the target logical switch.
6. Verify that the migration has finished successfully.

   ```
   GET /api/v1/transport-nodes/<transportnode-id>/state
   ```

   Wait until the migration state appears as `SUCCESS`. You can also verify the migration status of the vmknic in vCenter Server.

The VMkernel interface is migrated from VSS to N-VDS switch.

What to do next

You can migrate the remaining vmknics and the management kernel interface of the VSS to the N-VDS.

Migrate Management Kernel Interface from a VSS to an N-VDS

When you migrate the management kernel interface, you move vmnic0 and vmk0 from a VSS to an N-VDS . Before you migrate the VM management interface to N-VDS, perform connectivity checks.

Then you can migrate the physical uplink vmnic0 and vmk0 to the N-VDS together in one step. Modify the transport node configuration so that the vmnic0 is now configured as one of its uplinks.

Note : If you want to migrate the uplink vmnic0 and kernel interface vmk0 separately, first migrate vmk0 and then migrate vmnic0. If you first migrate vmnic0, then vmk0 remains on the VSS without any backing uplinks and you lose connectivity to the host.

Note You cannot revert the migration of the network interfaces from N-VDS to VSS.

Prerequisites

- Verify connectivity to the already migrated vmknics. See Migrate VM Kernel Interfaces to an N-VDS.
- Verify that an external device can reach interfaces vmk1 on storage VLAN-backed logical switch and vmk2 on the vMotion VLAN-backed logical switch.
- If vmk0 and vmk1 use different VLANs, trunk VLAN must be configured on the physical switch connected to PNICs vmnic0 and vmnic1 to support both VLANs.

Procedure

1. (Optional) Create a second management kernel interface on VSS and migrate this newly created interface to N-VDS.
2. (Optional) From an external device , verify connectivity to the test management interface.
3. If vmk0 uses a different VLAN than vmk1, create a VLAN-backed logical switch in the VLAN transport zone with a VLAN ID that matches the VLAN ID used by vmk0 on the VSS.
4. Retrieve the vSphere ESXi transport node configuration.

   `GET /api/v1/transport-nodes/<transportnode-id>`

   Where *<transportnode-id>* is the UUID of the transport node.
5. In the `host_switch_spec:host_switches` element of the configuration, add the vmnic0 to the pnics table and assign it to uplink-2.

```
"pnics": [      {
                    "device_name": "vmnic0",
                    "uplink_name": "uplink-2"
                },
                {
```

```
                    "device_name": "vmnic1",
                    "uplink_name": "uplink-1"
                }
        ],
```

6 Migrate the management kernel interface, vmk0 to N-VDS using the updated configuration.

`PUT api/v1/transport-nodes/< transportnode-id>if_id=<vmk>&esx_mgmt_if_migration_dest=<network>`

Where, the <*transportnode-id*> is the UUID of the transport node. The <*vmk*> is the name of the VMkernel management interface vmk0. The <network> is the UUID of the target logical switch.

7 Verify that the migration has finished successfully.

`GET /api/v1/transport-nodes/<transportnode-id>/state`

Wait until the migration state appears as `SUCCESS`. In vCenter Server, you can verify whether the kernel adapters are configured to display the new logical switch name.

What to do next

Verify the status of the transport node.

Verify the Transport Node Status

Make sure that the transport node creation process is working correctly.

After creating a host transport node, the N-VDS gets installed on the host.

Procedure

1 View the N-VDS on ESXi with the `esxcli network ip interface list` command.

On ESXi , the command output should include a vmk interface (for example, vmk10) with a VDS name that matches the name you used when you configured the transport zone and the transport node.

```
# esxcli network ip interface list
...

vmk10
   Name: vmk10
   MAC Address: 00:50:56:64:63:4c
   Enabled: true
   Portset: DvsPortset-1
   Portgroup: N/A
   Netstack Instance: vxlan
   VDS Name: overlay-hostswitch
   VDS UUID: 18 ae 54 04 2c 6f 46 21-b8 ae ef ff 01 0c aa c2
   VDS Port: 10
   VDS Connection: 10
   Opaque Network ID: N/A
   Opaque Network Type: N/A
   External ID: N/A
```

```
   MTU: 1600
   TSO MSS: 65535
   Port ID: 67108895

 ...
```

If you are using the vSphere Client, you can view the installed N-VDS in the UI by selecting host **Configuration > Network Adapters**.

The KVM command to verify the N-VDS installation is `ovs-vsctl show`. Note that on KVM, the N-VDS name is nsx-switch.0. It does not match the name in the transport node configuration. This is by design.

```
# ovs-vsctl show
...
    Bridge "nsx-switch.0"
        Port "nsx-uplink.0"
            Interface "em2"
        Port "nsx-vtep0.0"
            tag: 0
            Interface "nsx-vtep0.0"
                type: internal
        Port "nsx-switch.0"
            Interface "nsx-switch.0"
                type: internal
    ovs_version: "2.4.1.3340774"
```

2 Check the transport node's assigned tunnel endpoint address.

The vmk10 interface receives an IP address from the NSX-T IP pool or DHCP, as shown here:

```
# esxcli network ip interface ipv4 get
Name   IPv4 Address    IPv4 Netmask   IPv4 Broadcast   Address Type  DHCP DNS
-----  --------------  -------------  ---------------  ------------  --------
vmk0   192.168.210.53  255.255.255.0  192.168.210.255  STATIC           false
vmk1   10.20.20.53     255.255.255.0  10.20.20.255     STATIC           false
vmk10  192.168.250.3   255.255.255.0  192.168.250.255  STATIC           false
```

In KVM, you can verify the tunnel endpoint and IP allocation with the `ifconfig` command.

```
# ifconfig
...
nsx-vtep0.0 Link encap:Ethernet  HWaddr ba:30:ae:aa:26:53
          inet addr:192.168.250.4  Bcast:192.168.250.255  Mask:255.255.255.0
         ...
```

3 Check the API for state information.

Use the `GET https://<nsx-mgr>/api/v1/transport-nodes/<transport-node-id>/state` API call. For example:


```
{
  "state": "success",
  "host_switch_states": [
    {
      "endpoints": [
        {
          "default_gateway": "192.168.250.1",
          "device_name": "vmk10",
          "ip": "192.168.250.104",
          "subnet_mask": "255.255.255.0",
          "label": 69633
        }
      ],
      "transport_zone_ids": [
        "efd7f38f-c5da-437d-af03-ac598f82a9ec"
      ],
      "host_switch_name": "overlay-hostswitch",
      "host_switch_id": "18 ae 54 04 2c 6f 46 21-b8 ae ef ff 01 0c aa c2"
    }
  ],
  "transport_node_id": "2d030569-5769-4a13-8918-0c309c63fdb9"
}
```


Add a Compute Manager

A compute manager, for example, vCenter Server, is an application that manages resources such as hosts and VMs. NSX-T polls compute managers to find out about changes such as the addition or removal of hosts or VMs and updates its inventory accordingly.

In this release, this feature supports:

- vCenter Server versions 6.5 Update 1 and 6.5 GA only.
- IPv6 as well as IPv4 communication with vCenter Server.
- A maximum of 5 compute managers.

Procedure

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

2 Select **Fabric > Compute Managers** from the navigation panel.

3 Click **Add**.

4 Complete the compute manager details.

Option	Description
Name and Description	Type the name to identify the vCenter Server. You can optionally describe any special details such as, the number of clusters in the vCenter Server.
Domain Name/IP Address	Type the IP address of the vCenter Server.
Type	Keep the default option.
Username and Password	Type the vCenter Server login credentials.
Thumbprint	Type the vCenter Server SHA-256 thumbprint algorithm value.

If you left the thumbprint value blank, you are prompted to accept the server provided thumbprint.

After you accept the thumbprint, it takes a few seconds for NSX-T to discover and register the vCenter Server resources.

5 If the progress icon changes from **In progress** to **Not registered**, perform the following steps to resolve the error.

a Select the error message and click **Resolve**. One possible error message is the following:

```
Extension already registered at CM <vCenter Server name> with id <extension ID>
```

b Enter the vCenter Server credentials and click **Resolve**.

If an existing registration exists, it will be replaced.

The Compute Managers panel shows a list of compute managers. You can click the manager's name to see or edit details about the manager, or to manage tags that apply to the manager.

Configure Automated Transport Node Creation

If you have a vCenter Server cluster, you can automate the installation and creation of transport nodes on all the NSX-T hosts in single or multiple clusters instead of configuring manually.

Note Automated NSX-T transport node creation is supported only on vCenter Server 6.5 Update 1 and 6.5 GA.

If the transport node is already configured, then automated transport node creation is not applicable for that node.

Prerequisites

- The host must be part of a vCenter Server cluster.
- A transport zone must be configured.
- An uplink profile must be configured, or you can use the default uplink profile.
- An IP pool must be configured, or DHCP must be available in the network deployment.
- At least one unused physical NIC must be available on the host node.

- vCenter Server should have at least one cluster.
- A compute manager must be configured.

Procedure

1. From a browser, log in with admin privileges to an NSX Manager at `https://<nsx-manager-ip-address>`.
2. Select **Fabric > Nodes > Hosts**.
3. From the Managed by drop-down menu select an existing compute manager.
4. Select a cluster and click **Configure Cluster**.
5. Complete the configure cluster details.

Option	Description
Automatically Install NSX	Toggle the button to enable the installation of NSX-T on all the hosts in the vCenter Server cluster.
Automatically Create Transport Node	Toggle the button to enable the transport node creation on all the hosts in the vCenter Server cluster. **Note** This is a required setting.
Transport Zone	Select an existing transport node from the drop-down menu.
Uplink Profile	Select an existing uplink profile from the drop-down menu or create a custom uplink profile. **Note** The hosts in a cluster must have the same uplink profile. You can also use the default uplink profile.
IP Assignment	Select either **Use DHCP** or **Use IP Pool** from the drop-down menu. If you select **Use IP Pool**, you must allocate an existing IP pool in the network from the drop-down menu.
Physical NICs	Make sure that the physical NIC is not already in use for example, by a standard vSwitch or a vSphere distributed switch. Otherwise, the transport node state is partially successful, and the fabric node LCP connectivity fails to establish. You can use the default uplink or assign an existing uplink from the drop-down menu. Click **Add PNIC** to increase the number of NICs in the configuration.

   NSX-T installation and transport node creation on each host in the cluster starts in parallel. The entire process depends on the number of hosts in the cluster.

   When a new host is added to the vCenter Server cluster, NSX-T installation and transport node creation happens automatically.

6. (Optional) View the ESXi connection status.

```
# esxcli network ip connection list | grep 1234
tcp   0   0  192.168.210.53:20514  192.168.110.34:1234   ESTABLISHED  1000144459  newreno  netcpa
```

7 (Optional) Remove an NSX-T installation and transport node from a host in the cluster.

   a Select a cluster and click **Configure Cluster**.

   b Toggle the Automatically Install NSX button to disable the option.

   c Select one or more host and click **Uninstall NSX**.

   The uninstallation takes up to three minutes.

Create an NSX Edge Transport Node

A transport node is a node that is capable of participating in an NSX-T overlay or NSX-T VLAN networking. Any node can serve as a transport node if it contains an N-VDS. Such nodes include but are not limited to NSX Edges. This procedure demonstrates how to add an NSX Edge as a transport node.

An NSX Edge can belong to one overlay transport zone and multiple VLAN transport zones. If a VM requires access to the outside world, the NSX Edge must belong to the same transport zone that the VM's logical switch belongs to. Generally, the NSX Edge belongs to at least one VLAN transport zone to provide the uplink access.

Note If you plan to create transport nodes from a template VM, make sure that there are no certificates on the host in `/etc/vmware/nsx/`. The netcpa agent does not create a new certificate if a certificate already exists.

Prerequisites

- The NSX Edge must be joined with the management plane, and MPA connectivity must be Up on the **Fabric > Edges** page. See Join NSX Edge with the Management Plane.
- Transport zones must be configured.
- An uplink profile must be configured, or you can use the default uplink profile for bare-metal NSX Edge nodes.
- An IP pool must be configured, or DHCP must be available in the network deployment.
- At least one unused physical NIC must be available on the host or NSX Edge node.

Procedure

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

2 Select **Fabric > Nodes > Transport Nodes > Add**.

3 Type a name for the NSX Edge transport node

4 Select an NSX Edge fabric node from the drop-down list.

5 From the Available column, select a transport zone and click the right-arrow to move the zone to the Selected column.

An NSX Edge transport node belongs to at least two transport zones, an Overlay for NSX-T connectivity and VLAN for uplink connectivity.

6 Click **N-VDS** tab and provide the following details for a standard N-DVS.

Option	Description
N-VDS Name	Must match the names that you configured when you created the transport zones.
Uplink Profile	Select the uplink profile from the drop-down menu. The available uplinks depend on the configuration in the selected uplink profile.
IP Assignment	Select **Use DHCP**, **Use IP Pool**, or **Use Static IP List** for the overlay N-VDS. If you select **Use Static IP List**, you must specify a list of comma-separated IP addresses, a gateway, and a subnet mask. For VLAN N-DVS, leave the IP Pool field blank. No overlay tunnel endpoint IP address is needed because overlay N-DVS is for uplink VLAN traffic only.
Physical NICs	Unlike a host transport node, which uses vmnicX as the physical NIC, an NSX Edge transport node uses fp-ethX.

7 (Optional) View the transport node with the `GET https://<nsx-mgr>/api/v1/transport-nodes/<transport-node-id>` API call.

```
GET https://<nsx-mgr>/api/v1/transport-nodes/78a03020-a3db-44c4-a8fa-f68ad4be6a0c

{
  "resource_type": "TransportNode",
  "id": "78a03020-a3db-44c4-a8fa-f68ad4be6a0c",
  "display_name": "node-comp-01b",
  "transport_zone_endpoints": [
    {
      "transport_zone_id": "efd7f38f-c5da-437d-af03-ac598f82a9ec",
      "transport_zone_profile_ids": [
        {
          "profile_id": "52035bb3-ab02-4a08-9884-18631312e50a",
          "resource_type": "BfdHealthMonitoringProfile"
        }
      ]
    }
  ],
  "host_switches": [
    {
      "host_switch_profile_ids": [
        {
          "value": "8abdb6c0-db83-4e69-8b99-6cd85bfcc61d",
          "key": "UplinkHostSwitchProfile"
        },
        {
          "value": "9e0b4d2d-d155-4b4b-8947-fbfe5b79f7cb",
          "key": "LldpHostSwitchProfile"
        }
      ],
```

```
      "host_switch_name": "overlay-hostswitch",
      "pnics": [
        {
          "device_name": "vmnic1",
          "uplink_name": "uplink-1"
        }
      ],
      "static_ip_pool_id": "c78ac522-2a50-43fe-816a-c459a210127e"
    }
  ],
  "node_id": "c551290a-f682-11e5-ae84-9f8726e1de65",
  "_create_time": 1459547122893,
  "_last_modified_user": "admin",
  "_last_modified_time": 1459547126740,
  "_create_user": "admin",
  "_revision": 1
}
```


8 (Optional) For status information, use the `GET https://<nsx-mgr>/api/v1/transport-nodes/<transport-node-id>/status` API call.


```
{
  "control_connection_status": {
    "degraded_count": 0,
    "down_count": 0,
    "up_count": 1,
    "status": "UP"
  },
  "tunnel_status": {
    "down_count": 0,
    "up_count": 0,
    "status": "UNKNOWN",
    "bfd_status": {
      "bfd_admin_down_count": 0,
      "bfd_up_count": 0,
      "bfd_init_count": 0,
      "bfd_down_count": 0
    },
    "bfd_diagnostic": {
      "echo_function_failed_count": 0,
      "no_diagnostic_count": 0,
      "path_down_count": 0,
      "administratively_down_count": 0,
      "control_detection_time_expired_count": 0,
      "forwarding_plane_reset_count": 0,
      "reverse_concatenated_path_down_count": 0,
      "neighbor_signaled_session_down_count": 0,
      "concatenated_path_down_count": 0
    }
  },
  "pnic_status": {
    "degraded_count": 0,
```

```
    "down_count": 0,
    "up_count": 4,
    "status": "UP"
  },
  "mgmt_connection_status": "UP",
  "node_uuid": "cd4a8501-0ffc-44cf-99cd-55980d3d8aa6",
  "status": "UNKNOWN"
}
```


What to do next

Add the NSX Edge node to an edge cluster. See Create an NSX Edge Cluster.

Create an NSX Edge Cluster

Having a multi-node cluster of NSX Edges helps ensure that at least one NSX Edge is always available. In order to create a tier-0 logical router or a tier-1 router with NAT, you must associate it with an NSX Edge cluster. Therefore, even if you have only one NSX Edge, it must still belong to an NSX Edge cluster to be useful.

An NSX Edge transport node can be added to only one NSX Edge cluster.

An NSX Edge cluster can be used to back multiple logical routers.

After creating the NSX Edge cluster, you can later edit it to add additional NSX Edges.

Prerequisites

- Install at least one NSX Edge node.
- Join the NSX Edges with the management plane.
- Add the NSX Edges as transport nodes.
- Optionally, create an NSX Edge cluster profile for high availability (HA) at **Fabric > Profiles > Edge Cluster Profiles**. You can also use the default NSX Edge cluster profile.

Procedure

1. From a browser, log in with admin privileges to an NSX Manager at `https://<nsx-manager-ip-address>`.
2. Navigate to **Fabric > Nodes > Edge Clusters**.
3. Enter the NSX Edge cluster a name.
4. Select an NSX Edge cluster profile.
5. Click **Edit** and select either **Physical Machine** or **Virtual Machine**.

   Physical Machine refers to NSX Edges that are installed on bare metal. Virtual Machine refers to NSX Edges that are installed as virtual machines/appliances.

6 For Virtual Machine, select either NSX Edge Node or **Public Cloud Gateway Node** from the Member Type drop-down menu.

   If the virtual machine is deployed in a public cloud environment, select Public Cloud Gateway otherwise select NSX Edge Node.

7 From the **Available** column, select NSX Edges and click the right-arrow to move them to the **Selected** column.

What to do next

You can now build logical network topologies and configure services. See the *NSX-T Administration Guide*.

Uninstalling NSX-T

10

You can remove elements of an NSX-T overlay, remove a hypervisor host from NSX-T, or uninstall NSX-T completely.

This chapter includes the following topics:

Unconfigure an NSX-T Overlay

If you want to delete an overlay but keep your transport nodes in place, follow these steps.

Procedure

1. In your VM management tool, detach all VMs from any logical switches.
2. In the NSX Manager UI or API, delete all logical routers.
3. In the NSX Manager UI or API, delete all logical switch ports and then all logical switches.
4. In the NSX Manager UI or API, delete all NSX Edges and then all NSX Edge clusters.
5. Configure a new NSX-T overlay, as needed.

Remove a Host From NSX-T or Uninstall NSX-T Completely

If you want to uninstall NSX-T completely or just remove a hypervisor host from NSX-T so that the host can no longer take part in the NSX-T overlay, follow these steps.

The following procedure describes how to perform a clean uninstall of NSX-T.

Prerequisites

If the VM management tool is vCenter Server, put the vSphere host in maintenance mode.

Procedure

1. In your VM management tool, detach all VMs on the host from any NSX-T logical switches.

2 In the NSX Manager, delete the host transport node with the **Fabric > Nodes > Transport Nodes** UI or with the `DELETE /api/v1/transport-node/<node-id>` API.

Deleting the transport node causes the N-VDS to be removed from the host. You can confirm this by running the following command.

```
[root@host:~] esxcli network vswitch dvs vmware list
```

On KVM, the command is:

```
ovs-vsctl show
```

3 In the NSX Manager CLI, verify that the NSX-T install-upgrade service is running.

```
nsx-manager-1> get service install-upgrade
Service name: install-upgrade
Service state: running
Enabled: True
```

4 Uninstall the host from the management plane and remove the NSX-T modules.

It might take up to 5 minutes for all NSX-T modules to be removed.

There are several methods you can use to remove the NSX-T modules:

- In the NSX Manager, select **Fabric > Nodes > Hosts > Delete**.

  Make sure **Uninstall NSX Components** is checked. This causes the NSX-T modules to be uninstalled on the host.

  Remove the RHEL 7.4 and RHEL 7.3 dependency packages - json_spirit, python-greenlet, libev, protobuf, leveldb, python-gevent, python-simplejson, glog.

  Remove the Ubuntu 16.04.x dependency packages - nicira-ovs-hypervisor-node, openvswitch-switch, openvswitch-datapath-dkms, openvswitch-pki, python-openvswitch, openvswitch-common, libjson-spirit.

  Note that using **Fabric > Nodes > Hosts > Delete** with the **Uninstall NSX Components** option unchecked is not meant to be used to unregister a host. It is only meant as a workaround for hosts that are in a bad state.

- Use the `DELETE /api/v1/fabric/nodes/<node-id>` API.

  Note This API does not remove the dependency packages from the nsx-lcp bundle.

  Remove the RHEL 7.4 and RHEL 7.3 dependency packages - json_spirit, python-greenlet, libev, protobuf, leveldb, python-gevent, python-simplejson, glog.

  Remove the Ubuntu 16.04.x dependency packages - nicira-ovs-hypervisor-node, openvswitch-switch, openvswitch-datapath-dkms, openvswitch-pki, python-openvswitch, openvswitch-common, libjson-spirit.

- Use the CLI for vSphere.
  a Get the manager thumbprint.

```
manager> get certificate api thumbprint
```

  b On the host's NSX-T CLI, run the following command to detach the host from the management plane.

```
host> detach management-plane <MANAGER> username <ADMIN-USER> password <ADMIN-PASSWORD>
thumbprint <MANAGER-THUMBPRINT>
```

  c On the host, run the following command to remove filters.

```
[root@host:~] vsipioctl clearallfilters
```

  d On the host, run the following command to stop netcpa.

```
[root@host:~] /etc/init.d/netcpad stop
```

  e Power off the VMs on the host.

  f Manually uninstall the NSX-T modules from the host.

  Note that removing individual modules is not supported. You must remove all modules in one command.

```
esxcli software vib remove -n nsx-shared-libs -n nsx-common-libs -n nsx-metrics-libs -n
nsx-rpc-libs -n nsx-nestdb-libs -n nsxa -n nsx-lldp -n nsx-da -n nsx-exporter -n nsx-
aggservice -n nsxcli -n nsx-python-protobuf -n nsx-sfhc -n nsx-netcpa -n nsx-mpa -n nsx-
esx-datapath -n nsx-host -n nsx-support-bundle-client -n nsx-nestdb -n nsx-platform-client
-n nsx-hyperbus -n nsx-ctxteng -n nsx-python-gevent -n nsx-python-greenlet
```

- On RHEL 7.4 and RHEL 7.3, use the `sudo yum remove nsx* <package-name>` command.

  Remove the dependency packages, glog, json_spirit, kmod-openvsiwth, nicira-ovs-hypervisor-node, openvswitch, openvswitch-selinux-policy, python-simplejson

- On Ubuntu 16.04.x, use the `apt-get remove "nsx*" <package-name>` command.

  Remove the dependency packages, nicira-ovs-hypervisor-node, openvswitch-switch, openvswitch-datapath-dkms, openvswitch-pki, python-openvswitch, openvswitch-common, libjson-spirit

What to do next

After making this change, the host is removed from the management plane and can no longer take part in the NSX-T overlay.

If you are removing NSX-T completely, in your VM management tool, shut down NSX Manager, NSX Controllers, and NSX Edges and delete them from the disk.