C H A P T E R 2

Using VM-FEX

This chapter describes how to use the Cisco Virtual Machine Fabric Extender (VM-FEX).

This chapter includes the following sections:

Information About VM-FEX

The Cisco Virtual Machine Fabric Extender (VM-FEX) works with a Cisco virtual interface card (VIC) adapter to replace software-based switching on the server with external hardware-based switching. VM-FEX introduces many new components and concepts. It is important to understand these components and concepts in order to successfully deploy VM-FEX in your network.

VN-Link

A virtual network link (VN-Link) is a virtual network interface card (vNIC) that connects to a virtual Ethernet (vEthernet) interface or virtual host bus adapter (VHBA) that connects to a virtual Fibre Channel (vFC) interface using a Fibre Channel over Ethernet (FCoE) tunnel.

Static Interfaces

A static interface is an interface with parameters that are configured manually by you as the administrator. A static interface can be a vEthernet or Fibre Channel interface.

For statically created fixed and floating vEthernet interfaces, you as the device administrator can associate the configuration to the vEthernet interface before bringing the interface up. Follow these rules to resolve conflicts:

- Any configuration specified under the vEthernet interface overrides the configuration in the port profile.
- If a vEthernet interface is preconfigured with a port profile and the NIV adapter or vEthernet module signals a different port profile, the port profile specified in the protocol message takes precedence.
- vEthernet configuration or port profile configurations can be specified in a sync profile if configuration synchronization between two Cisco Nexus 5000 Series devices is enabled.

Static Virtual Adapters

A Static Virtual Adapter is a virtual network interface card (VNIC) or a virtual host bus adapter (vHBA).

Dynamic Interfaces or Dynamic VNICs

A dynamic interface or dynamic VNIC is a vEthernet interface that gets configured automatically when the adapter and device communicate. The dynamic VNIC inherits the parameters of the virtual network interface card (VNIC) port group on the virtual machine (VM).

A dynamic virtual adapter can be a virtual NIC but not a Virtual HBA. A dynamic virtual interface can be a vEthernet interface. Dynamic interfaces support hardware-based failovers in the, Unified Computing System (UCS), UCS VIC card.

Because dynamic fixed and floating vEthernet interfaces are created dynamically on the device, you cannot have a configuration that is associated with it. As a result, the device can only associate a configuration to this type of vEthernet using the specified port profile in either the VIF_CREATE or Virtual Ethernet Module (VEM) attach message.

Floating Virtual Interfaces

In a Cisco network interface virtualization (NIV) environment with a virtual computing, networking, server, and storage architecture, a VM migrates from one physical server to the other with the virtual network link and vEthernet interface on the physical device. A floating virtual interface is a virtual interface that migrates with its profile characteristics and parameters with the VM and virtual network.

For floating (static or dynamic) vEthernet interfaces, configuration changes (including shut or no shut) are allowed at anytime regardless of the attached (hypervisor requested for a vNIC) state except for the binding configuration, which cannot be changed while a vEthernet interface is attached. The binding configuration can be changed only if a vEthernet interfaced is not attached. To see the attachment state of a vEthernet interface, use the **show interface veth** command.

Fixed vEthernet Interfaces

A fixed vEthernet interface is a virtual interface that does not support migration across physical interfaces.

You can change the configuration of a fixed vEthernet interface (static or dynamic), including using the shut or no shut commands or the create or delete commands anytime. The vEthernet number to channel number binding is persistent unless you change it.

Dynamic Floating vEthernet Interfaces

The vEthernet interfaces are dynamically created when new VMs are brought up on the hypervisor. A dynamic floating vEthernet interface can move from one VEM to another as VMs are migrated using vMotion. Additionally, the statistics and policies of a dynamic vEthernet interfaces are retained when a VM moves from one host to another.

The following process shows how VMs and their associated vEthernet interfaces are brought up:

1. You create a port profile on the device, and it gets pushed to VMware vCenter (vCenter).

2. vCenter assigns a port group ID or the port profile and responds with the VMware port group ID.

3. When a server administrator brings up a VM on the hypervisor, you as the administrator chooses the port group for the VM and tells the hypervisor to bring up the VM.

4. If the port group that was picked is a port group that belongs to the Cisco Nexus 5000 Series Distributed Virtual Switch (DVS), VEM, or NIV adapter, the hypervisors are notified of the pending attach of the VM with the DV port id and the port group id.

5. One of the following actions occur:

   a. If the port group belongs to a VEM, the VEM sends a VM attach pending notification to the virtual supervisor module (VSM) that runs on the Cisco Nexus 5000 Series device. Otherwise, a new vEthernet is created for the DV port id.

   b. If the port group belongs to a NIV adapter, the NIV adapter detects the pending attach of the VM using pass-through switching (PTS) and sends a VIF create message to the device with the VM's DV port id and port group id number in the VMware provisioning type-length-value (TLV). The adapter also configures the dynamic VNIC that inherits the parameters from the port profile.

6. The device creates the appropriate vEthernet interface and also expands the port group id for the VM into a port profile and applies it to the vEthernet interface.

7. One of the following actions occur:

   a. For a VEM, the device control plane brings up the vEthernet interface like any regular Ethernet interface and the data plane programming is done on the VEM through the Message and Transaction Service (MTS) messages.

   b. For an NIV adapter, as the device brings up the vEthernet interface, a virtual network tag (VNTag) is allocated to the vEthernet interface and both the device forwarding path and adapter forwarding path are set up for the VNTag using the virtual interface card (VIC). The vEthernet interface parameters (for example, the default VLAN, default class of service (CoS), and so on) are sent to the adapter using the VIC. The vEthernet interface is declared up, and a VIC message is sent to the adapter instructing it to mark the vNIC to as up (operational).

Static Floating vEthernet Interfaces

You can create and configure floating vEthernet interfaces.

You can use the following command to select a vEthernet interface number that has been given a distributed virtual port and distributed virtual switch (DVS):

```
int veth10
   vmware dvport <dvport-number> dvsswitch uuid <uuid>
```

You can use the following command to select a vEthernet interface number for a MAC address of a VM:

```
int veth11
   vsi mac <mac-address>
```

The creation model is the same between static and dynamic floating vEthernet interfaces. When a VEM or NIV adapter signals the device to create vEthernet interface, the device first attempts to match for a configured static vEthernet interface. If a match is found, the specific vEthernet interface number is used. If there is a configuration specified for the static vEthernet interface, this configuration overrides the same coniguration specified, in the port profile.

Fixed vEthernet Interface Provisioning Model

A fixed vEthernet interface is not associated with VMs. Fixed vEthernet interfaces can be created and exposed to host operating systems (OSs) as regular NICs. The host OS can use these vEthernet interfaces and normal Ethernet interfaces as boot devices. If the server is VM-enabled, these fixed vEthernet interfaces present themselves as physical adapters because vCenter cannot migrate this type of vEthernet interface using vMotion.

The desired provisioning model for a fixed vEthernet interface is the dynamic model. In this provisioning model, there is no configuration on the device that represents the vEthernet interface. This model depends on the port profile information that is provided to the device through the VIC protocol.

The device advertises the list of port profile names that are configured on the device using the VIC control channel (VIF_SET of 0). The NIV adapter allows a way for you as the server administrator to choose from a list of port profiles to be attached to each fixed vEthernet interface.

This process describes how to bring up a fixed vEthernet interface:

1. The NIV adapter sends a VIF_CREATE message with the channel number and optional port profile name.

2. The device looks for a match from its list of static fixed vEthernet interface configurations and does one of the following:

    a. If there is a matching channel number, that vEthernet interface number is brought up.

    b. If there is no static fixed vEthernet interface with a matching channel number, a dynamic fixed vEthernet interface is created instead. A dynamic fixed vEthernet interface does not have any user configuration or port profile associated with it.

3. If a port profile name is in the VIF_CREATE message, and the fixed vEthernet interface is not already configured with a port profile in the device configuration, the port profile parameters within the VIF_CREATE message are configured for the vEthernet interface.

4. In the static fixed model, you as the administrator of the device must map the vEthernet interface that is being configured on the device to the channel number of the vNIC that it represents. Because this task requires that the device administrator gets the channel information from the server or adapter administrator and is prone to errors, the static configuration is considered to be less desirable.

5. For each static, fixed vNIC provisioned in an adapter, a corresponding vEthernet interface must be created and bound to an Ethernet interface and channel number where the Ethernet interface is the physical interface number that the adapter is connected to and channel number is the vNIC instance number provisioned in the adapter.

This example shows vEthernet interface 10 being created and bound to Ethernet interface 1/10, channel 4:

```
int veth10
bind interface ethernet 1/10 channel 4
inherit port profile xxx
```

A unique channel number given to each physical interface is required. You can remove the binding by using the no bind interface ethernet command, even after the vEthernet interface is attached to the VNIC in the host (that is, even after the device receives the **vif-create** command). This behavior is different from creating a floating vEthernet interface. With fixed vEthernet interfaces, the adapter internally maps the channel number to the OS-visible NIC interfaces and displays the mapping information in the adapter tool.

In a FEX Active-to-Active topology, there are at least two VIFs for each vEthernet interface and multiple bind interface commands are allowed under a vEthernet interface. For example, if an adapter has two uplinks and each uplink is connected to FEX port 100/1/10 and 100/1/11, then both bound interfaces must be configured as static fixed vEthernet interfaces.

```
int veth10
bind interface ethernet 100/1/10 channel 4
bind interface ethernet 101/1/11 channel 4
```

Static Fixed vEthernet Interface Configuration

In the Cisco Integrated Management Controller (CIMC), the port profile is not specified for the VNIC, and on the device, the static fixed vEthernet interface is manually configured in the running configuration.

```
interface veth 1
bind int e 100/1/1 channel 1
no shut
```

Dynamic Fixed vEthernet Interface Configuration

An alternative to creating a static, fixed vEthernet interface is to configure the device with the dynamic_vnic_profile port profile and configure a bound physical interface in the VNTag mode. The device pushes the dynamic_vnic_profile port profile to the Unified Computing System (UCS) VIC adapter.

You must go to CIMC and add or edit the fixed vNIC properties by changing the port profile to dynamic_vnic_profile. You can either manually type the string "dynamic_vnic_profile" or use the drop-down menu, because the device has pushed all port profiles to the adapter.

Then, you must reboot the server to make the changes take effect with the early firmware release. As a result, the dynamic fixed vEthernet interfaces are entered automatically in the device's running configuration.

```
interface veth 1
inherit port-profile xyz
bind int e 100/1/1 channel 1
```

NIV Host Discovery

Before NIV vEthernet interfaces can be created, the physical interface connected to the host must be configured with VNTag mode, and Data Center Bridging Exchange (DCBX) communication between the switchport and the host uplink must be converged with VNTag mode.

```
int e1/1 or e100/1/1
switchport mode vntag
```

<table>
<tr><td>✎
Note</td><td>Because the access or trunk keyword is a mandatory parameter, there is no way to force only VNTag encapsulation on an interface at all times.</td></tr>
</table>

<table>
<tr><td>✎
Note</td><td>VEM interface vEthernet does not need this configuration because frames from the VEM do not include a VNTag.</td></tr>
</table>

The host that runs softPTS can operate in classical Ethernet mode before transitioning to VNTag mode. The transition to VNTag mode is made when the host starts to include the NIV capability (that is, TLV) in its DCBX advertisement. Until then, the host runs softPTS operates in the classical Ethernet mode. While it operates in the classical Ethernet mode, the host has full connectivity to the network through the physical interface connected the device that is configured with VNTag mode. Because the port can be configured in access or trunk mode, you can use the following commands to configure the switchport behavior when the port is configured in VNTag mode:

```
switchport trunk allowed vlan [add|all|except|none|remove] <vlan-list>
switchport trunk native vlan <vlan>
switchport access vlan  <vlan>
```

You can use **switchport trunk allowed** *vlan* and **switchport trunk native vlan** *vlan* commands only when the port is configured to be a VNTag trunk device port that is operating in the classical Ethernet mode. The command configures the allowed VLAN list for the switchport and default VLAN for the trunk port.

You can use **switchport access vlan** only when the port is configured as a VNTag access device port that is operating in the classical Ethernet mode. This command allows you to configures the access VLAN for the device port.

Port Profile Inheritance

Because dynamic fixed and floating vEthernet interfaces are created dynamically on the device, there cannot be a configuration associated with it. As a result, the device can only configure this type of vEthernet interface using the port profile specified in either the VIF_CREATE or VEM attach message.

For statically created fixed and floating vEthernet interfaces, it is possible for you as the device administrator, to associate the configuration with the vEthernet interface before it is brought up. These rules are necessary to resolve conflicts:

- Any configuration specified under the vEthernet interface overrides the configuration in the port profile.
- If a vEthernet interface is preconfigured with a port profile, and the NIV adapter or VEM signals a different port profile, the port profile specified in the protocol message takes precedence.

vEthernet interface configurations or port profile configurations can be specified in a synchronization profile if configuration synchronization between two Cisco Nexus 5000 Series devices is enabled.

Distributed Virtual Switches

The DVS or virtual distributed switch (VDS) spans multiple VMware vSphere (vSphere) hosts and aggregating the network management of the virtual switching environment into a centralized management interface. Prior to DVS, each server needed to have a device configured and if VMware vMotion (vMotion) was desired, the configuration of these devices needed to be manually verified as consistent across the entire cluster. With DVS, a single device is created that spans all of the servers in the cluster. The port groups only need to be configured once, and they are automatically applied on all servers in the cluster.

The VEM for ESX systems always contains both the Local Switching module and the PTS module. The PTS module covers both the Soft-PTS and Hard-PTS mode of functions that can flexibly enable both VEM and NIV functionalities. This software package is also commonly known as VIB, where VIB is the format used by VMware.

Hardware and Software Requirements

Table 1 explains the VM-FEX hardware and software requirements.

Table 2-1   VM-FEX Hardware and Software Requirements

Component	Hardware and Software Combination 1	Hardware and Software Combination 2	Hardware and Software Combination 3
Server	Cisco C250-M2	OS =Wind2008 r2 sp1	ESX v4.1u1
CIMC	1.4(0.21)		
BIOS	1.4.1		
Cisco UCS P81E	VNIC drivers	cxxx-scu-drivers-w2k8r2x64-1.3.1.1	
Cisco Nexus 5548 device	5.1(3)N1(1)		Cisco NX-OS 5.. (0.223)
CNA VIC adapter firmware version	1.6 (0.72)		

FC storage array	—		
VCenter server/client	Version 4.1	Enterprise plus License	

Dynamic Fixed vEthernet Interface Configuration

The two possible configuration models for fixed vEthernet interfaces are static and dynamic. The preferred configuration model on the device is the dynamic fixed vEthernet interface because of the ease of configuration and the minimal interaction required between the device and the server administrator to set up the vEthernet configuration.

First, the device administrator configures several port profiles on the device. Each port profile defines a set of device port configurations that are applied to a vEthernet interface. The configuration is based on the various roles that a host could have in the network and is a template from which a server administrator can choose the configuration that is needed for the server port.

When a VIC session is established between the adapter and the device, as part of the protocol handshake, the device passes a list of port profiles that are relevant to the host. You can display this list of port profiles using a BIOS or adapter tool and choose a specific port profile from the list for a vNIC that is presented on the PCI bus. After choosing the port profile, the port profile ID is sent back to the device though the VIC protocol as part of this VIF bringup. The device creates a dynamic vEthernet interface during this VIF bringup and applies the port profile that it receives from the adapter.

The dynamic fixed vEthernet interfaces are entered automatically in the device running-configuration.

```
interface veth 1
inherit port-profile xyz
bind int e 100/1/1 channel 1
```

When the vEthernet handshake is complete and the interface is brought up, all the interface configuration that was part of the port profile is applied to the interface and the interface becomes operational.

If the physical link between the server and the device flaps or there is a disruption in the VIC session between the device and the server, the port profile list on the server is cleared so that there is no stale information that is retained on the host. When the VIC session is reestablished, a new port profile list is sent to the server.

Figure 2-1 shows a dynamic vEthernet configuration on a device.

Figure 2-1   Dynamic vEthernet Configured on a Device
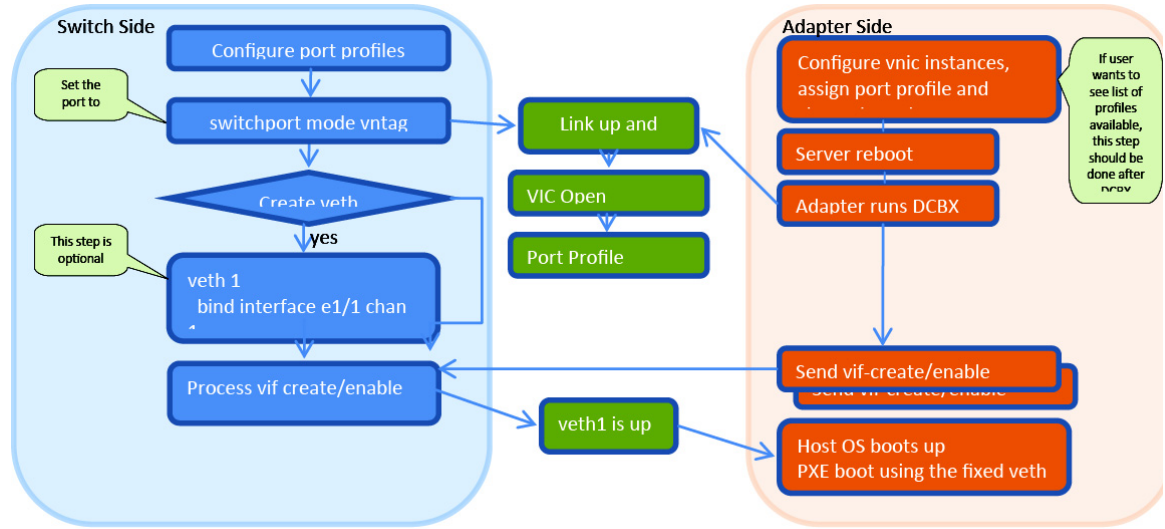
Static Fixed vEthernet Interface Configuration

In CIMC, the port profile is not specified for the VNIC. On the device, the static fixed vEthernet is manually entered in the running configuration such as:

```
interface vEthernet 1
bind int e 1/18 channel 1
no shut
```

Note   The UCS VIC static VNIC failover and failback is invisible to the host OS when performed in the hardware on the UCS VIC. That is, the OS sees the VNIC as up and running during a failover or failback.

Some packets might get lost due to the time needed for the reconvergence of the return traffic to take place.

The Cisco Nexus 5548 Series device sends the gratuitous ARP reply for the static MAC and IGMP join messages of the active vEthernet interface that has failed over or failed back for the reconvergence to take place. This action enables the return traffic path immediately in the direction from the network to the host.

In the FEX straight-through topology, the active Cisco Nexus 5000 Series device synchronizes continuously all unicast MAC addresses learned on the active vEthernet interface to the Cisco Nexus 5000 Series device with the standby vEthernet interface. The destination for these unicast MAC addresses point to them Multichassis EtherChannel Trunk (MCT) on the standby Cisco Nexus 5000 Series device. Unicast MAC addresses are synchronized using Cisco Fabric Services over Ethernet (CFSoE) as a transport. IGMP groups are not synchronized.

A future Cisco NX-OS version sends for all MAC dynamic addresses in addition to the static one earned behind the failed over/failed back vEthernet port and for all IGMP join messages collected for the active vEthernet interface, the IGMP join requests, and the Gratuitous ARP reply to the external network.

Fixed vEthernet Interface Failback and Failover with a vPC System

The fixed vEthernet interface failback and failover with a vPC system process is as follows:

1. The adapter is brought up.

2. If this interface is a fixed vEthernet interface, the NIV adapter also creates one vEthernet interface on the active Cisco Nexus 5000 Series device and one vEthernet interface on the standby Cisco Nexus 5000 Series device.

3. The NIV adapter uses the same channel number in both VIF_CREATE messages to the two Cisco Nexus 5000 Series devices. I

4. For static fixed vEthernet interfaces, you create the vEthernet interface on the two Cisco Nexus 5000 Series device with the same vEthernet interface number and channel number.

5. The adapter reaches the state to signal the VIF_ACTIVATE message on one of the uplinks, making one of the vEthernet interfaces active while the other one stays in the standby state.

6. The NIV adapter detects that a failover is needed either by a link down event or when a DCBX keep alive message times out.

   As a result of any of these events occurring, the NIV adapter activates the standby link to become the new forwarding path for the VIF. The NIV adapter sends the VIF_ACTIVATE message to the standby Cisco Nexus 5000 Series device.

7. When the standby Cisco Nexus 5000 Series device receives the VIF_ACTIVATE message, it does the following:

   a. Adjusts all unicast MAC address destinations to points to the VIF, instead of the MCT.

   b. Initiates an IGMP general query to the adapter. The IGMP joins from the host OS updates all the membership for the entire network path.

   c. Sends gratuitous ARP replies for all the unicast MAC addresses for this VIF to adjust the entire network path to the standby side.

   d. All other configuration-based features, such as a PVLAN or an ACL, are already pre programmed into the hardware for this VIF so no adjustment is necessary during the failover.

8. If the failed path comes back online, the devices do not take any failback action. The devices take action only if the NIV adapter sends the VIF_ACTIVATE message on the new link. As a result, the same pocess occurs again.

Fixed Static vEthernet Interface Failback and Failover with a vPC System

The process for a fixed static vEthernet interface fails back or fails over to a vPC system is as follows:

1. The adapter is brought up. If this interface is a fixed vEthernet interface, the NIV adapter also creates one vEthernet on the active Cisco Nexus 5000 Series device and one vEthernet on the standby Cisco Nexus 5000 Series device. The NIV adapter uses the same channel number in both VIF_CREATE messages to the two Cisco Nexus 5000 Series devices. For static fixed vEthernet interface, you must create the vEthernet interface on the two Cisco Nexus 5000 Series devices with the same vEthernet number and channel number.

2. The adapter reaches the state to signal the VIF_ACTIVATE message on one of the uplinks, making one of the vEthernet interface active while the other one stays in standby state.

3. The NIV adapter detects failover is needed either by a link down event or DCBX keep-alive times out. Let's enumerate all possible failures in the Cisco Nexus 5000 Series device and FEX fabric. As a result of any of the event occurring, the NIV adapter activates the standby link to become the new forwarding path for the VIF.

4. The NIV adapter sends the VIF_ACTIVATE message to the standby Cisco Nexus 5000 Series device. When the standby Cisco Nexus 5000 Series device receives the VIF_ACTIVATE message, it does the following:

   a. Adjusts all unicast MAC address destinations to point to the VIF, instead of the MCT.

   b. Initiates an IGMP general query to the adapter. The IGMP joins from the host OS updates all the membership for the entire network path.

   c. Sends gratuitous ARP replies for all the unicast MAC addresses for this VIF to adjust the entire network path to the standby side.

   d. All other configuration-based features such as a PVLAN or an ACL are already pre programmed into the hardware for this VIF so no adjustment is necessary during the failover.

5. If the failed path comes back online, the devices do not take any failback action. The devices take action only if the NIV adapter sends the VIF_ACTIVATE message on the new link. As a result, the same pocess occurs again.

VM-FEX Configuration Scenario

This scenario takes you through the steps to configure VM-FEX in a sample network.

The following are the high-level steps for configuring FM-FEX:

1. Bring up the server with the VEM-PTS bits installed.

2. Activate the two static vNICs on the MT card.

3. Configure the device with a static vEthernet port and bind it to the physical port.

4. Activate the two static vNICs on the Cisco UCS P81E Virtual Interface Card.

5. Define the port profile for dynamic vNICs on the Cisco UCS P81E Virtual Interface Card.

6. Reboot the server.

7. Verify that the two vEthernet interfaces are active and connected to the two static vNICs.

8. Verify that the dynamic vEthernet interfaces are active and connected to the dynamic vNIC.

9. Enable a vPC on the system.

10. Register the vPC system with the vCenter as a virtual distributed switch (VDS).

11. Create the VM-FEX data center and assign it to the vds-nx5k VDS folder.

    The port profile should propagate to the vds-nx5k VDS as a vmware port group.

12. Add the host to the vds-nx5k VDS and map the two static vNICs to it.

13. Activate the VM on the ESX host and map it to the port group.

14. Notice the dynamic vNIC has been activated, assigned to the VM in PTS and connected using the VN-Link to the vEthernet port on the Cisco Nexus 5000 Series device vPC.

Static VNIC Configuration
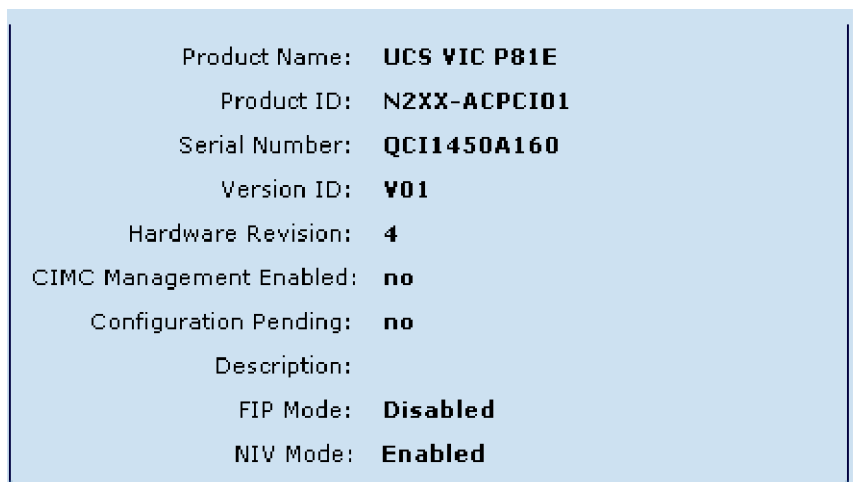
The static VNIC configuration is as follows:

✎
Note   Perform the following steps on both the EXS Host 1 and ESX Host 2.

Step 1   Connect to the CIMC controller of the ESX Host 1 and ESX Host 2.

Step 2   From the left navigation pane, choose **Server > Inventory**.

Step 3   From the Network Adapters menu, choose the UCS VIC adapter card and verify that NIV mode is enabled on the adapter as shown in Figure 2-2.

Figure 2-2   Verifying the NIV Mode


```
Product Name:          UCS VIC P81E
Product ID:            N2XX-ACPCI01
Serial Number:         QCI1450A160
Version ID:            V01
Hardware Revision:     4
CIMC Management Enabled:  no
Configuration Pending:    no
Description:
FIP Mode:              Disabled
NIV Mode:              Enabled
```

✎
Note   • The Cisco UCS P81E VIC can be enabled in FEX mode or CNA mode.

• The Cisco UCS P81E VIC in FEX mode supports hardware-based failover provided by the CIMC or active-standby VN-Link from the VNIC to the vEthernet port on the Cisco Nexus 5548 device.

• VNIC is primarily connected to the vEthernet port on the Cisco Nexus 5548 device and has a standby VN-Link connected to the standby interface on the Cisco Nexus 5548 device.

• The Cisco UCS P81E VIC in CNA mode supports Active/Active mode.

Step 4   From the Adapter Card Menu, choose the VNICs tab and click the **Add** button to add two static VNICs (A-FEXs) on each server's Ethernet0 and Ethernet1 VNIC.

Step 5   Define each VNIC's properties in order to activate two static VNICs on ESX Host 1. These static VNICs are needed by the VMware VDS (VDS) for the VNIC proxy path as shown in Figure 2-3.

Send comments to nexus5k-docfeedback@cisco.com

Figure 2-3   VNIC's Properties
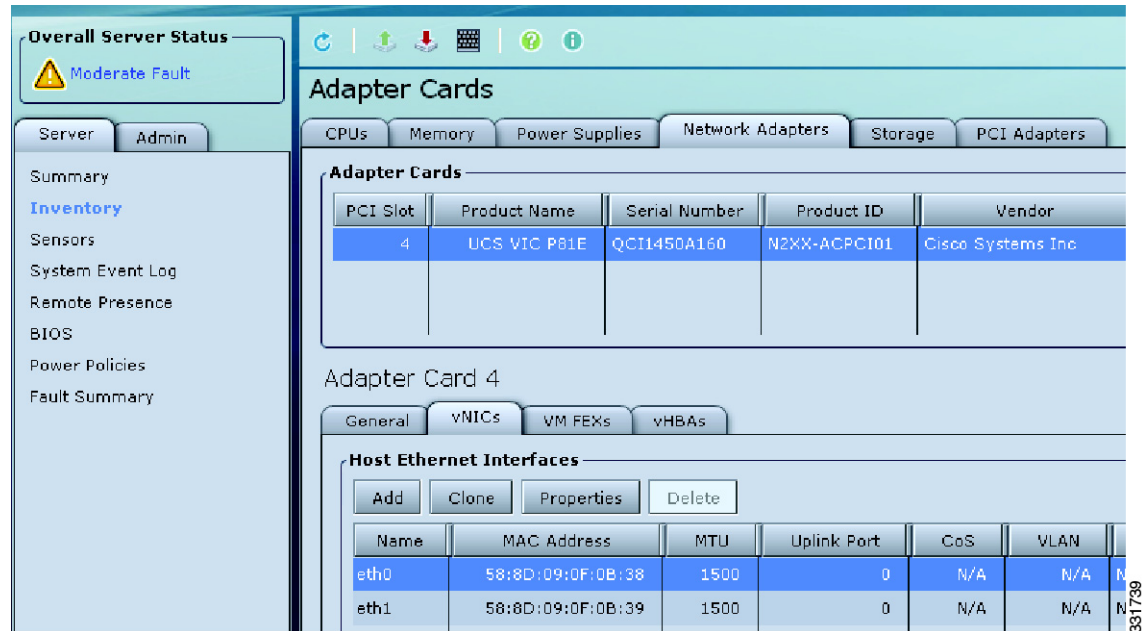
Step 6   Enter the following parameters for the two Static VNICs on the UCS VIC for ESX Host 1:

- ESX Host 1

- Name : stateth 0

- Uplink Port : 0

- MAC Address : already associated

- Channel Num : 10

- Port Profile : not associated

- Enable Uplink Failover (Enable to)

- Failback Timeout: 5

- Name : stateth 1

- Uplink Port : 0

- MAC Address : already associated

- Channel Number : 11

- Port Profile : not associated

- Enable Uplink Failover (Enabled)

- Failback Timeout : 5

- Leave other values set to default

The parameters for the static VNICs on the UCS VIC for ESX Host 1 are shown in Figure 2-4.

Figure 2-4   *VNICs on the UCS VIC for ESX Host 1*

The vNIC properties for **stateht1** is shown in Figure 2-5.

Figure 2-5   VNICs on the UCS VIC for ESX Host 2

Step 7   Enter the following parameters for the two static VNICs on the UCS VIC for ESX Host 2:

- ESX Host 2
- Name : stateth 0
- Uplink Port : 0
- MAC Address : already associated
- Channel Number: 10
- Port Profile : not associated
- Enable Uplinnk Failover (Enabled)
- Failback Timeout : 5
- Name : stateth 1
- Uplink Port : 0
- MAC Address : already associated
- Channel Num : 11
- Port Profile : not associated

- Enable Uplink Failover (Enable to)
- Leave other values set to default.

The parameters for the static VNICs on the UCS VIC for ESX Host 1 are shown in Figure 2-6.

Figure 2-6   vNIC Properties for ESX Host 2

Adding vNIC for **stateht1** is shown in Figure 2-7.

Figure 2-7   *Adding vNIC*

Step 8   From the Server tab, choose the **Summary** folder.

Step 9   Reboot the server by choosing the **Power Cycle Server** option in the right panel.

Note   A server reboot is needed with the early version of the Cisco UCS P81E VIC firmware in order to activate the changes on the Cisco UCS P81E VIC that are related to the static VNIC configuration.

Figure 2-8 shows the screen after the reboot.

Figure 2-8   *Cisco Integrated Management Controller*
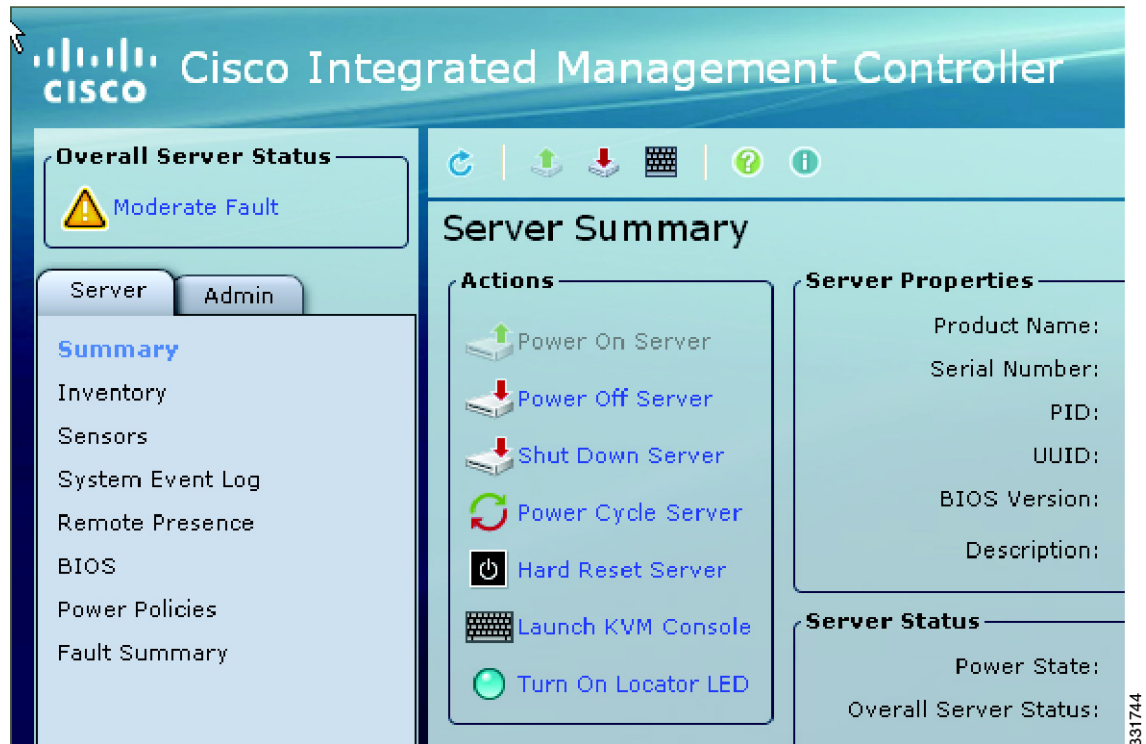
Step 10   Connect to the Cisco Nexus 5548-1 device and Cisco Nexus 5548-2 Series device and configure them for a static VNIC (A-FEX) connection as follows:

   a. On both devices, make sure that the following features are enabled by entering the following commands:

   – **install feature-set virtualization**

   – **feature-set virtualization**

   – **feature fex**

   – **feature vmfex**

   – **feature lldp**

   – **feature vpc**

   – **vethernet auto-create**

   – **feature fcoe**

   – **copy run start**

   – **reload**

   b.   On both devices, enable the physical interfaces connected to both ESX servers:

```
interface Ethernet1/17
switchport mode vntag
        shut
interface Ethernet1/18
switchport mode vntag
        shut
```

   c. On both devices, enable the virtual interfaces needed to connect to the two static VNICs enabled on the Cisco UCS VIC.

On the Cisco Nexus 5548-1 device enter these commands:

```
pod1-5548-1(config)# interface vethernet1
pod1-5548-1(config-if)# bind interface eth 1/17 channel 10
pod1-5548-1(config-if)# no shut
pod1-5548-1(config-if)# interface vethernet 3
pod1-5548-1(config-if)# bind interface eth 1/17 channel 11
pod1-5548-1(config-if)# no shut

pod1-5548-1(config-if)# interface  vethernet 2
pod1-5548-1(config-if)# bind interface eth 1/18 channel 10
pod1-5548-1(config-if)# no shut
pod1-5548-1(config-if)# interface vethernet 4
pod1-5548-1(config-if)# bind interface eth 1/18  channel 11
pod1-5548-1(config-if)# no shut
```

On the Cisco Nexus 5548-2 device enter these commands:

```
pod1-5548-2(config)# interface vethernet 1
pod1-5548-2(config-if)# bind interface eth 1/17 channel 10
pod1-5548-2(config-if)# no shut
pod1-5548-2(config-if)# interface  vethernet 3
pod1-5548-2(config-if)# bind interface eth 1/17 channel 11
pod1-5548-2(config-if)# no shut

pod1-5548-2(config-if)# interface  vethernet 2
pod1-5548-2(config-if)# bind interface eth 1/18 channel 10
pod1-5548-2(config-if)# no shut
pod1-5548-2(config-if)# interface vethernet 4
pod1-5548-2(config-if)# bind interface eth 1/18 channel 11
pod1-5548-2(config-if)# no shut
```

d. On both devices, enable the interface connected to both ESX servers.

```
interface Ethernet1/17
no  shutdown
nterface Ethernet1/18
no  shutdown
```

e. Verify that the two static VNICs are connected with VN-Link to each static fixed vEthernet interface on the Cisco NX5548 device on both ESX hosts using the following verification commands.

On the Cisco Nexus 5548-1 device enter these commands:

```
pod1-5548-1# show interface vethernet 1
Vethernet1 is up
Bound Interface is Ethernet1/17
Hardware is Virtual, address is 0005.73fc.24a0
Port mode is access
Speed is auto-speed
Duplex mode is auto
300 seconds input rate 0 bits/sec, 0 packets/sec
300 seconds output rate 0 bits/sec, 0 packets/sec
Rx
0 unicast packets  0 multicast packets  0 broadcast packets
0 input packets  0 bytes
0 input packet drops
Tx
0 unicast packets  0 multicast packets  0 broadcast packets
0 output packets  0 bytes
0 flood packets
0 output packet drops

POD1-5548-1# show interface vethernet 1 detail
vif_index: 20
```

```
-------------------------
veth is bound to interface Ethernet1/17 (0x1a010000)
  priority: 0
  vntag: 16
  status: active
  channel id: 10
  registered mac info:
      vlan 0 - mac 00:00:00:00:00:00
      vlan 0 - mac 58:8d:09:0f:0b:3c
      vlan 0 - mac ff:ff:ff:ff:ff:ff
```

On the Cisco Nexus 5548-2 device enter these commands:

```
POD1-5548-2# show interface vethernet 1 detail
vif_index: 20
-------------------------
  veth is bound to interface Ethernet1/17 (0x1a010000)
  priority: 0
  vntag: 16
  status: standby
  channel id: 10
  registered mac info:
      vlan 0 - mac 00:00:00:00:00:00
      vlan 0 - mac 58:8d:09:0f:0b:3c
      vlan 0 - mac ff:ff:ff:ff:ff:ff


POD1-5548-1# show mac address
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link
   VLAN     MAC Address      Type      age     Secure NTFY    Ports
---------+-----------------+--------+---------+------+----+------------------
* 1       588d.090f.0970    static   0            F    F  Veth2
* 1       588d.090f.0971    static   0            F    F  Veth4
* 1       588d.090f.0b3c    static   0            F    F  Veth1
* 1       588d.090f.0b3d    static   0            F    F  Veth3
* 4040    0005.73fc.1fbc    static   0            F    F  Router
```

Note Both vEthernet 1 interface active and standby are up on both Cisco Nexus 5548 devices, but the static MAC address has been learned only by the active vEthernet.

f. Shut down the Ethernet 1/17-18 interfaces and verify that the fixed vEthernet interfaces are in the failover state. The vEthernet 1 interface standby takes over the active vEthernet 1 role and its static MAC address moves with it:

```
POD1-5548-1(config-if-range)# inerface ethernet 1/17-18
POD1-5548-1(config-if-range)# no shut

POD1-5548-2# show interface vethernet 1 detail
vif_index: 20
-------------------------
  veth is bound to interface Ethernet1/17 (0x1a010000)
  priority: 0
  vntag: 16
  status: active
  channel id: 10
  registered mac info:
      vlan 0 - mac 00:00:00:00:00:00
      vlan 0 - mac 58:8d:09:0f:0b:3c
      vlan 0 - mac ff:ff:ff:ff:ff:ff

POD1-5548-2# show mac address
```

```
        Legend:
                * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
                age - seconds since last seen,+ - primary entry using vPC Peer-Link
           VLAN      MAC Address      Type       age      Secure NTFY    Ports
        ---------+----------------+--------+---------+------+----+------------------
        * 1         588d.090f.0970   static    0           F    F   Veth2
        * 1         588d.090f.0971   static    0           F    F   Veth4
        * 1         588d.090f.0b3c   static    0           F    F   Veth1
        * 1         588d.090f.0b3d   static    0           F    F   Veth3


    POD1-5548-1(config-if-range)# show interface veth 1 detail
      veth is not bound
      priority: n/a
      vlan: n/a
      vntag: n/a
      registered mac info:

    POD1-5548-1(config-if-range)# show mac address
    Legend:
            * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
            age - seconds since last seen,+ - primary entry using vPC Peer-Link
       VLAN      MAC Address      Type       age      Secure NTFY    Ports
    ---------+----------------+--------+---------+------+----+------------------
    * 4040     0005.73fc.1fbc   static    0           F    F   Router
    POD1-5548-1(config-if-range)#
```

g. Activate the physical Ethernet 1/17-18 interfaces on the Cisco Nexus 5548-1 device and verify that the failback of static fixed vEthernet interfaces is performed and that the active role and static MAC address moves back to the Cisco Nexus 5548-1 device:

```
    POD1-5548-1# show interface vethernet 1 detail
    vif_index: 20
    -------------------------
      veth is bound to interface Ethernet1/17 (0x1a010000)
      priority: 0
      vntag: 16
      status: active
      channel id: 10
      registered mac info:
          vlan 0 - mac 00:00:00:00:00:00
          vlan 0 - mac 58:8d:09:0f:0b:3c
          vlan 0 - mac ff:ff:ff:ff:ff:ff

    POD1-5548-1# show mac address
    Legend:
            * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
            age - seconds since last seen,+ - primary entry using vPC Peer-Link
       VLAN      MAC Address      Type       age      Secure NTFY    Ports
    ---------+----------------+--------+---------+------+----+------------------
    * 4040     0005.73fc.1fbc   static    0           F    F   Router
    * 1        588d.090f.0970   static    0           F    F   Veth2
    * 1        588d.090f.0971   static    0           F    F   Veth4
    * 1        588d.090f.0b3c   static    0           F    F   Veth1
    * 1        588d.090f.0b3d   static    0           F    F   Veth3
    POD1-5548-1#

    POD1-5548-2# show interface vethernet 1 detail
    vif_index: 20
    -------------------------
      veth is bound to interface Ethernet1/17 (0x1a010000)
      priority: 0
      vntag: 16
```

```
        status: standby
        channel id: 10
        registered mac info:
            vlan 0 - mac 00:00:00:00:00:00
            vlan 0 - mac 58:8d:09:0f:0b:3c
            vlan 0 - mac ff:ff:ff:ff:ff:ff

POD1-5548-2# show mac address
POD1-5548-2#
```
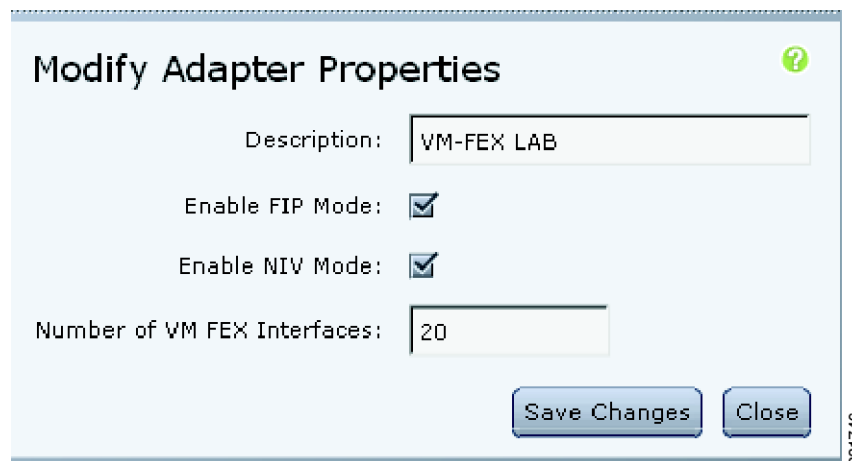
Dynamic Floating vEthernet Interfaces Configuration Scenario

The Dynamic Floating vEthernet Interfaces Configuration Scenario is as follows:

Step 1 Enable 20 dynamic VNICs on each server's Cisco UCS VIC card.

Step 2 From the navigation pane in CIMC, choose **Sever Menu > Inventory** folder.

Step 3 From the right window choose the **General** tab for the Cisco UCS VIC and click the **Modify Adapter Properties** to enable 20 Dynamic VNIC interfaces on both ESX hosts in NIV mode.

Step 4 Click **Save Changes**.

Step 5 Reboot the servers for the changes to take effect.

Modifying Adapter Properties is shown in Figure 2-9.

Figure 2-9   Modifying Adapter Properties

Step 6 On both Cisco Nexus 5548 Series devices, configure the port profile for the dynamic VNICs that connect to the automatically created dynamic vEthernet interfaces on the Cisco Nexus 5548 device. This port profile appears as a port group on the VDS.

The dynamic vEthernet interfaces should be entered into the running configuration and connected to the dynamic vNICs automatically when the VM connects to the dynamic port group in PTS mode as shown in Figure 2-10.

Figure 2-10   Virtual FEXs

Step 7   On both Cisco Nexus 5548 devices, configure the port profile that is exported to the vCenter VDS as port group VLAN 61.

Step 8   Enable VLAN 61.

```
port-profile type vethernet VLAN61
switchport mode access
switchport access vlan 61
dvs-name all
port-bindindg dynamic
state enabled
vlan 61
```

The **dvs-name all** command defines the dvs device in vCenter that this port profile is exported to as a port group as shown in Figure 2-11. Using the all option, the port group is exported to all DVSes in the data center.

These keywords are the vEthernet port binding types:

- The **dynamic** keyword specifies that the port is connected when a VM is powered on and disconnected when it is powered off. The maximum port limits are enforced.

- The **ephemeral** keyword specifies that the port is created when a VM is powered on and destroyed when it is powered off. The maximum port limits are not enforced.

- The **static** keyword specifies that the port is always connected. The maximum port limits are enforced.

Step 9   Connect to the vCenter server using the vSphere client.

Step 10   In the vSphere client, choose **Home > Inventory > Hosts and Clusters**.

Step 11 Right click the vSphere server to create a new datacenter name, for example, DCy (where y is the number of your pod).

Step 12 Ignore the certificate warning.

Figure 2-11   vSphere Client
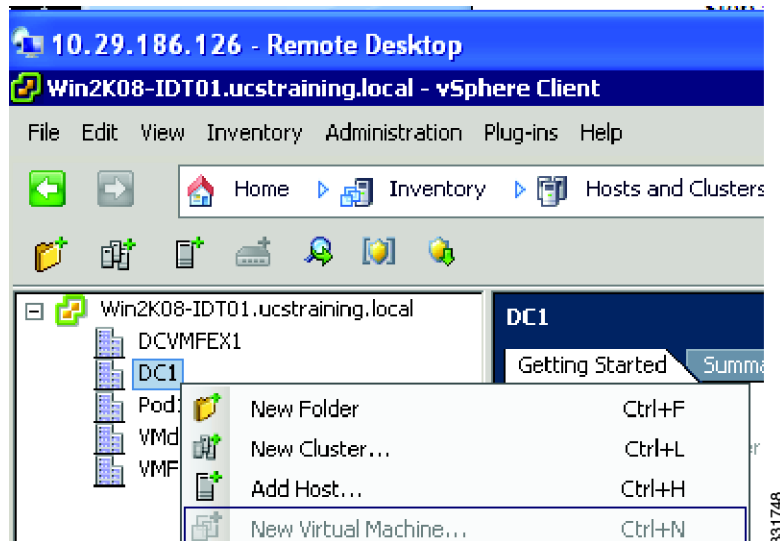
Step 13 Connect the vCenter vSphere server to the vPC system. The following configuration order establishes the SVS connection between the vCenter server and the vPC system as shown in Figure 2-12.

Figure 2-12   SVS connection between the vCenter server and the vPC system

Note A Cisco Nexus 5000 Series device or pair of Cisco Nexus 5000 Series devices in vPC mode can support one DVS cluster.

Each DVS is created with an extension key and a certificate. In the case where two Cisco Nexus 5000 Series devices are configured in a vPC topology, the primary vPC device pushes the DVS and extension key pair to a particular datacenter that is supported on a vCenter and port profiles. Both primary and secondary devices independently poll the vCenter to perform garbage collection by retrieving the list of DV port IDs in use. Any vEthernet interface that maps to an unused DV port ID is deleted. The vCenter never initiates communication to the devices.

Step 14   Register the Cisco Nexus 5548 device plug-in with the vSphere server as follows:

 a.   Download and save the Cisco_Nexus_5000 Series.xml extension file from both Cisco Nexus 5548 devices.

 b.   Download the plug-in by entering the IP address of the Cisco Nexus 5548 web server in your browser as shown in Figure 2-13.

Figure 2-13   Index of Downloads
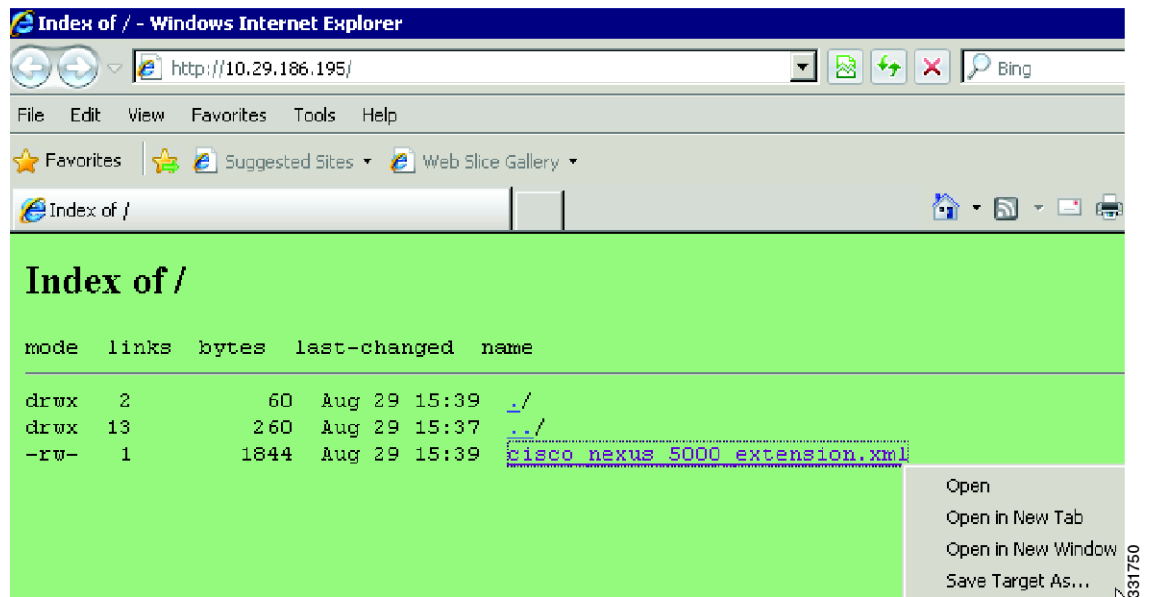
Step 15   Register both Cisco_nexus_5000_extension.xml files with the vSphere server:

 a.   From the Plug-In Menu of the vCenter server, choose the **Manage Plug-ins** option and right-click in the empty space above the available plug-in-in to create a new plug-in.

 b.   Ignore the certificate warning.

Figure 2-14   Plug-in Manager
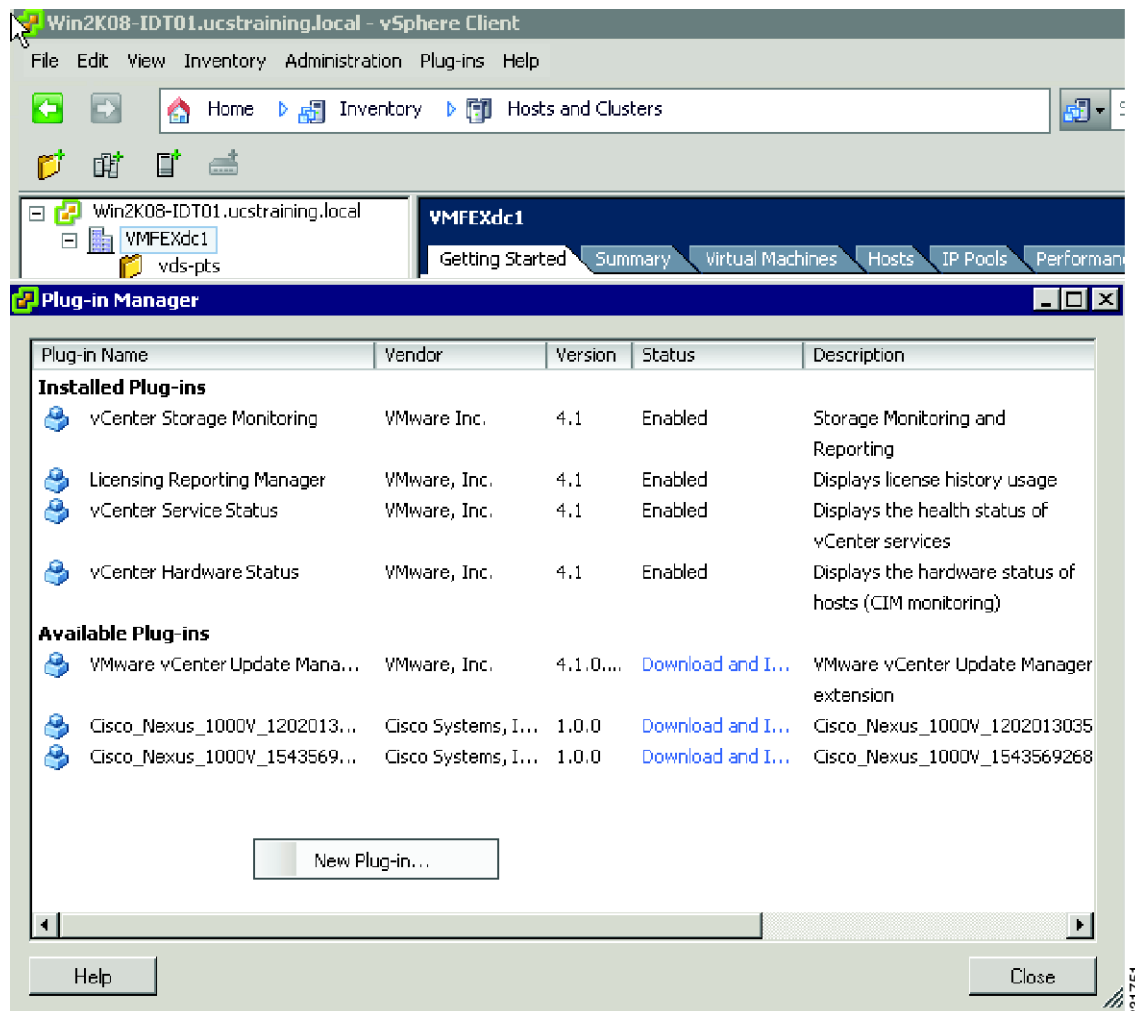
c. Browse for the new plug-in and click the **Register plug-in** button as shown in Figure 2-14.

Figure 2-15   Register plug-in
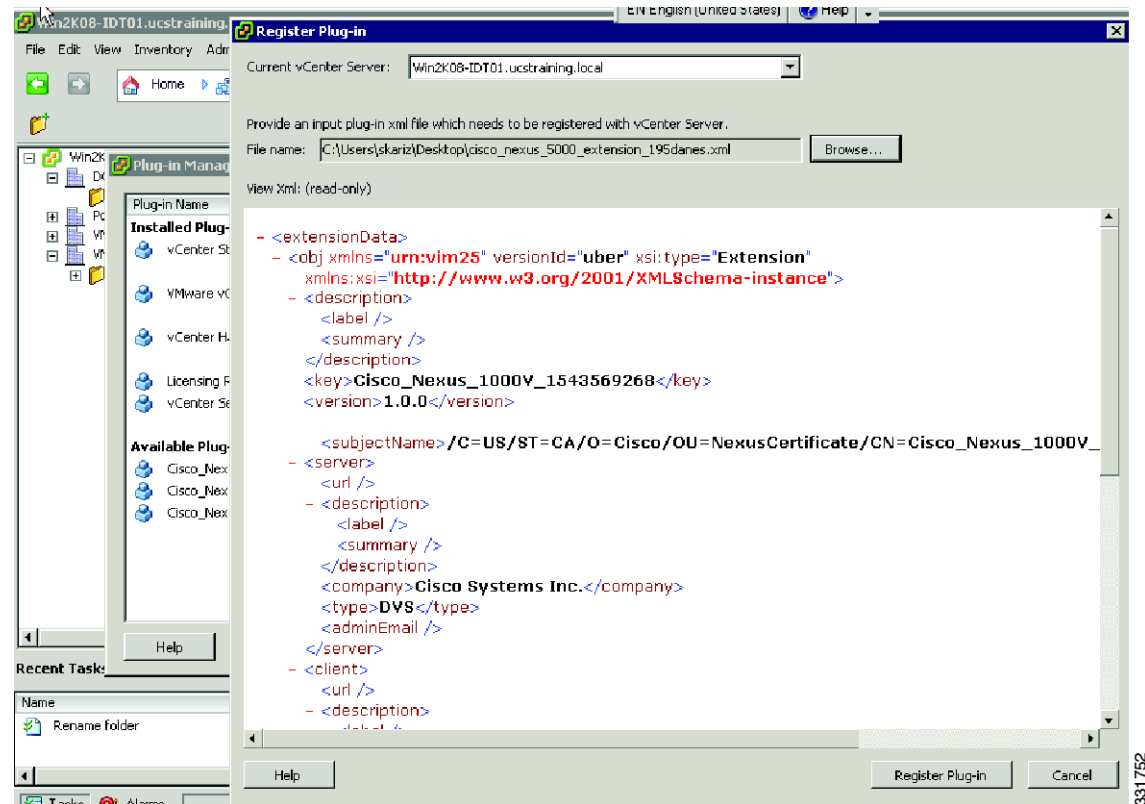
d. Ignore the security certificate warning and wait for the report about the plug-in being successfully installed as shown in Figure 2-15. As a result, the plug-in is listed under the Available Plug-ins.
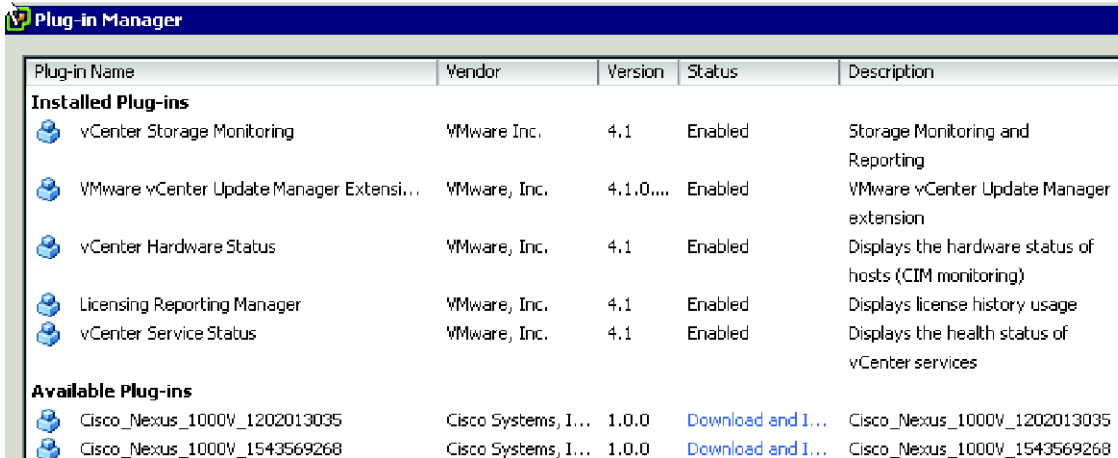
Note   If the process returns an error, your plug-in might have already been imported under the available plug-ins.

e. Verify in vCenter that the plug-in has been successfully installed for both Cisco Nexus 5548 devices as shown in Figure 2-16.

Figure 2-16   *Status of Plug-ins*

Step 16   On the Cisco Nexus 5548-1 and Cisco Nexus 5548-2 Series devices, enable the vPC domain and LACP feature.

On the Cisco Nexus 5548-2 Series device enter the following commands:

```
pod1-5548-2(config)# feature vpc
pod1-5548-2(config)# feature lacp

pod1-5548-2(config)# vpc domain 1
pod1-5548-2(config-vpc-domain)# role priority 100     /no priority  the other switch
pod1-5548-2(config-vpc-domain)# peer-keepalive destination 10.29.186.195 vrf management

pod1-5548-2(config)# int eth 1/27-30
pod1-5548-1(config-if-range)# channel-group 10
pod1-5548-1(config-if-range)# exit

pod1-5548-2(config)# int po10
pod1-5548-2(config-if)# vpc peer-link
pod1-5548-2(config-if)# spanning-tree port type network
pod 1-5548-2(config-if)#                              //  permit all vlans but but not FCOE
VLAN 200!
pod 1-5548-2(config-if)#  sw trunk allowed vlan 1-199, 201-3967, 4048-4093
pod1-5548-2(config-if)# no shut
```

On the Cisco Nexus 5548-1 Series device:

```
pod1-5548-1(config)# feature vpc
pod1-5548-1(config)# feature lacp

pod1-5548-1(config)# vpc domain 1
pod1-5548-1(config-vpc-domain)# role priority 90
pod1-5548-1(config-vpc-domain)# peer-keepalive destination 10.29.186.196 vrf management


pod1-5548-1(config)# int eth 1/27-30
pod1-5548-1(config-if-range)# channel-group 10
pod1-5548-1(config-if-range)# exit

pod1-5548-1(config)# int po10
pod1-5548-1(config-if)# vpc peer-link
pod1-5548-1(config-if)# spanning-tree port type network //  permit all vlans but but not
FCOE VLAN 200!
```

```
pod1-5548-1(config-if)#)  sw trunk allowed vlan 1-199, 201-3967, 4048-4093  //no fcoe vlan

pod1-5548-1(config-if)#no shut
```

Step 17   Verify which vPC member is the primary and which is the secondary vPC member.

```
pod1-5548-2# show vpc
Legend:
                 (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                   : 1
Peer status                     : peer adjacency formed ok
vPC keep-alive status           : peer is alive
Configuration consistency status: success
Per-vlan consistency status     : success
vPC role                        : secondary
Number of vPCs configured       : 0
Peer Gateway                    : Disabled
Dual-active excluded VLANs       : -
Graceful Consistency Check      : Enabled

vPC Peer-link status
---------------------------------------------------------------------
id   Port   Status Active vlans
--   ----   ------ --------------------------------------------------
1    Po10   up     1,61
POD1-5548-1# show vpc
Legend:
                 (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                   : 1
Peer status                     : peer adjacency formed ok
vPC keep-alive status           : peer is alive
Configuration consistency status: success
Per-vlan consistency status     : success
vPC role                        : primary
Number of vPCs configured       : 0
Peer Gateway                    : Disabled
Dual-active excluded VLANs       : -
Graceful Consistency Check      : Enabled
vPC Peer-link status
---------------------------------------------------------------------
id   Port   Status Active vlans
--   ----   ------ --------------------------------------------------
1    Po10   up     1,61
OD1-5548-2(config-if)# show vpc role

vPC Role status
--------------------------------------------------
vPC role                        : secondary
Dual Active Detection Status    : 0
vPC system-mac                  : 00:23:04:ee:be:01
vPC system-priority             : 32667
vPC local system-mac            : 00:05:73:fc:24:fc
vPC local role-priority         : 110


--------------------------------------------------------
------------------------------------------------------------

POD1-5548-1# show vpc role

vPC Role status
--------------------------------------------------
```

```
vPC role                        : primary
Dual Active Detection Status    : 0
vPC system-mac                  : 00:23:04:ee:be:01
vPC system-priority             : 32667
vPC local system-mac            : 00:05:73:fc:1f:bc
vPC local role-priority         : 90
POD1-5548-1#
```

Step 18   Configure the SVS connection on the primary and secondary vPC members.

Note   The secondary SVS member communicates with the primary SVS member to synchronize the extension key.

```
pod1-5548-1(config)# svs connection 2VC
pod1-5548-1(config-svs-conn)# protocol vmware-vim
pod1-5548-1(config-svs-conn)# vmware dvs datacenter-name DC1
pod1-5548-1(config-svs-conn)# dvs-name  Pod1PTS
pod1-5548-1(config-svs-conn)# remote ip  address 10.29.186.125 port 80 vrf management
 pod1-5548-1(config-svs-conn)# install certificate default
 pod1-5548-1(config-svs-conn)#  extension-key: Cisco_Nexus_1000V_1543569268  // only on
the primary vpc member
```

Note   Find the name of the extension-key Cisco_Nexus_1000V_1543569268 in the vCenter server under the installed plug-ins for the respective Cisco Nexus 5548 Series device. Use the install certificate default command to generate an extension key that you can import into the vCenter server as shown in Figure 2-17.

Figure 2-17   Available Plug-ins

Note   The configuration process is more consistent if the plug-in is imported after the install certificate command is issued on the device.

Step 19   Connect the primary vPC member to the vSphere center.
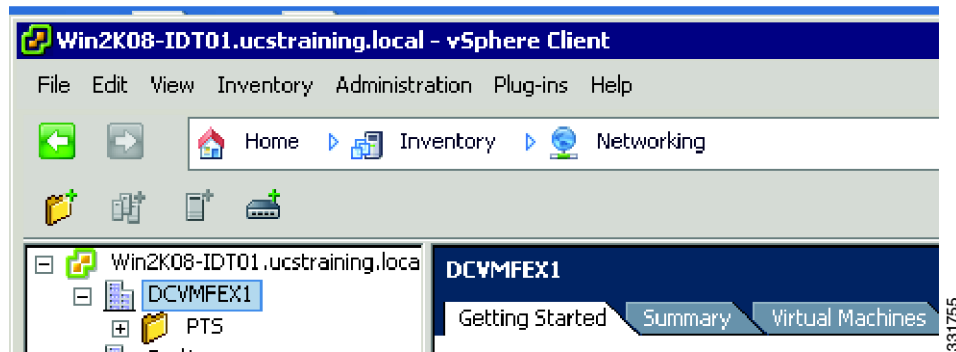
```
pod1-5548-1(config)# svs connection 2VC
pod1-5548-1(config-svs-conn)# connect
Note: Command execution in progress..please wait
pod1-5548-1(config-svs-conn)#
```

One of the results of the connect action is that the dvs-pts folder is created under your datacenter as shown in Figure 2-18.

Figure 2-18   vSphere - DCVMFEX1

Step 20   Manually configure the secondary vPC member and connect it to the vCenter.

> **Note**   The secondary vPC member does not connect even if the connect command is successful.

```
pod1-5548-2(config)# svs connection 2VC
pod1-5548-2(config-svs-conn)# protocol vmware-vim
pod1-5548-2(config-svs-conn)# vmware dvs datacenter-name DC1
pod1-5548-2(config-svs-conn)# dvs-name Pod1PTS
pod1-5548-2(config-svs-conn)# remote ip  address 10.29.186.125 port 80 vrf management
pod1-5548-2(config-svs-conn)# install certificate default
pod1-5548-2(config-svs-conn)#
```

> **Note**   The extension key is not configured manually. Instead, it is synchronized by the vPC.

The extension key is synchronized by the primary vPC member to the secondary vPC member. You can verify that the extension key is the same on both peers by using the **show svs connection** command. If the two Cisco Nexus 5548 devices were not vPC peers, the extension key would be different for each device, and each device would have to establish its own connection to the vCenter.

Step 21   Verify that the SVS connection between the vCenter and the vPC system is activated properly and that the new VDS device, Pod1PTS, and the port profile are imported to the vCenter as the port group of the new VDS device.

```
POD1-5548-1(config)# show svs connection

connection 2VC:
    ip address: 10.29.186.125
    remote port: 80
    vrf: amang
    protocol: vmware-vim https
    certificate: default
    datacenter name: DC1
    extension key: Cisco_Nexus_1000V_1543569268
    dvs name: Pod1PTS
    DVS uuid: 44 f1 04 50 47 cc 42 d6-ba ea 53 f9 f9 32 4b fb  // This is pushed by
VCenter to switch
    config status: Enabled
    operational status: Connected                                              // Only
Primary connects to VCenter
    sync status: in progress
    version: VMware vCenter Server 4.1.0 build-258902

Peer Info:
```
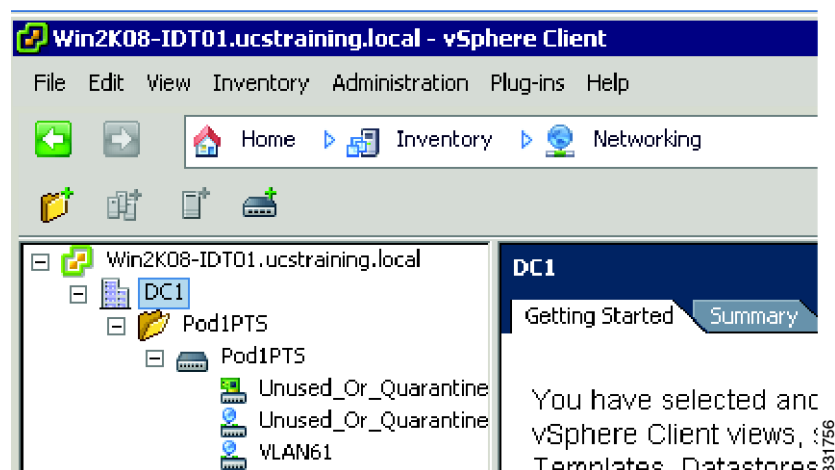
```
     ----------
    connection 2VC:
        ip address: 10.29.186.125
        remote port: 80
        vrf: management
        protocol: vmware-vim https
        extension key: Cisco_Nexus_1000V_1543569268 //This  extension key is result of  sync
        certificate: default //status of the VPC primary to second.
          certificate match: TRUE
        datacenter name: DC1
        dvs name: Pod1PTS
        DVS uuid: -
        config status: Disabled
        operational status: Disconnected
```

Step 22   After the SVS connection is activated, verify that the vPC system pushed the port profile VLAN61 and two other port profiles (that are not used) to the VDS. These port profiles become the VDS port groups under the DC1 datacenter. Also, verify that Po1PTS VDS was created by the vPC system in vCenter as shown in Figure 2-19.

Figure 2-19 ***vPC system in vCenter***


```
POD1-5548-2# show svs connection

connection 2VC:
    ip address: 10.29.186.125
    remote port: 80
    vrf: management
    protocol: vmware-vim https
    certificate: default
    datacenter name: DC1
    extension key: Cisco_Nexus_1000V_1543569268      // Received from the VPC Primary
    dvs name: pod1PTS
    DVS uuid: -
    config status: Enabled
    operational status: Disconnected
    sync status: -
    version: -


Peer Info:
----------
connection 2VC:
```

```
ip address: 10.29.186.125
remote port: 80
vrf: amang
protocol: vmware-vim https
extension key: Cisco_Nexus_1000V_1543569268
certificate: default
  certificate match: TRUE
datacenter name: DC1
dvs name: Pod1PTS
DVS uuid: 44 f1 04 50 47 cc 42 d6-ba ea 53 f9 f9 32 4b fb //Received from VCenter
config status: Enabled
operational status: Connected
```

Step 23   Verify that the VEM has been installed in PTS mode on both ESX hosts successfully. See the *VM-FEX/A-FEX Hardware/Software Installation Procedure* in case the VEM has not been installed correctly. (Currently the VDS device has no host or any VMNIC associated. It has only port groups available.)

⚠

Caution   Add the two ESX hosts to the DC1 datacenter in the vSphere server by choosing the **Home > Inventory > Hosts & Clusters** folder. Define the following characteristics for the two hosts:

ESX Host1:

```
10.29.186.197
Username= root
Password= Not4u2use!
Use the Enterprise License Plus entered
Select the shared datastore visible across the Qlogic VHBA and the datastore1 which is
local.
```

ESX Host 2:

```
10.29.186.198
Username= root
Password= Not4u2use!
se the Enterprise License Plus entered.
Select the shared datastore visible across the Qlogic VHBA and the datastore1 which is
local.
```

While adding the host, the wizard should show a screen for each host as shown in Figure 2-20.

Figure 2-20   Adding the Host Wizard Screen
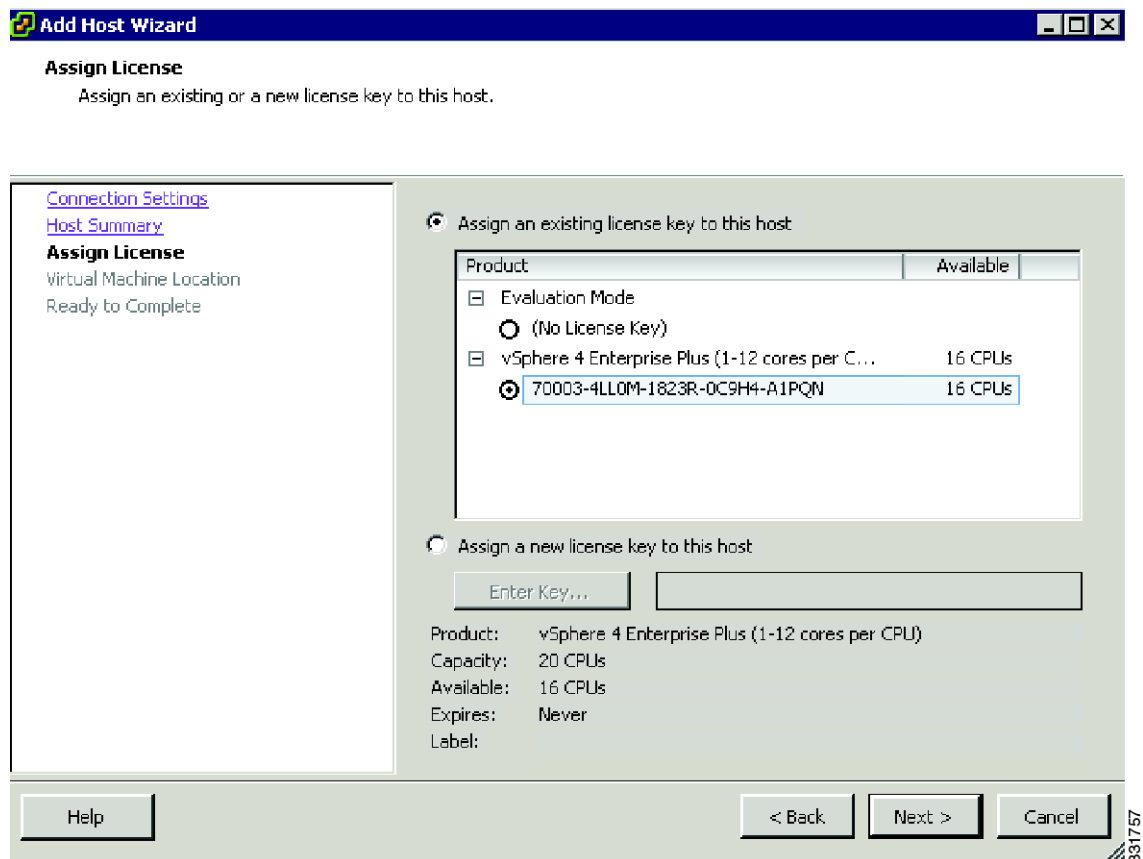
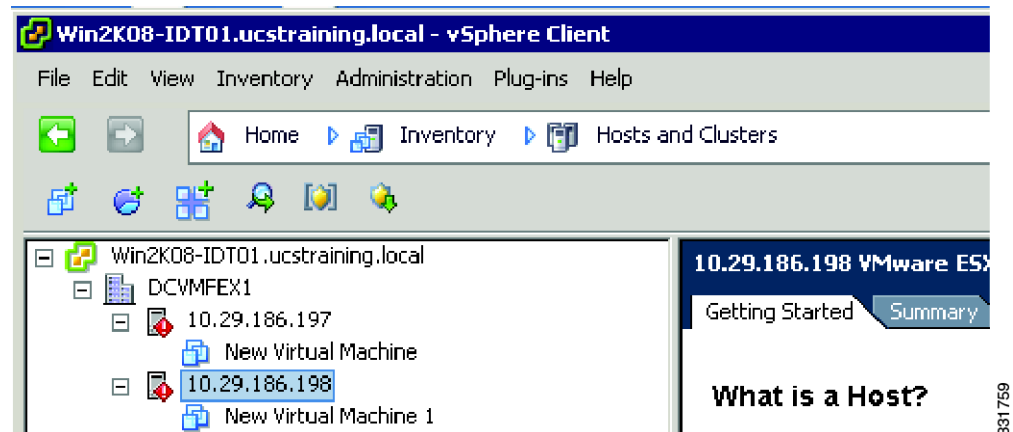The summary before finishing the wizard is shown in Figure 2-21.

Figure 2-21   Reviewing the Summary
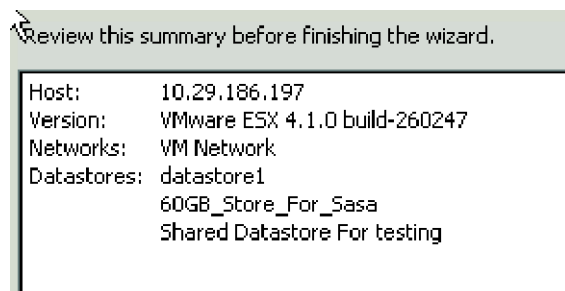
The New Virtual Machines on vSphere Client is shown in Figure 2-22.

Figure 2-22   New Virtual Machines on vSphere Client

Step 24   Add ESX host 1 and ESX host 2 to the PodYPTS VDS that was created. From the VSphere Server choose the **Home > Inventory > Networking** folder and right- click the data center DC*y*, where *y* is the number of your pod. See Figure 2-23.

Figure 2-23   Adding Hosts to the PodYPTS VDS

![Note icon]
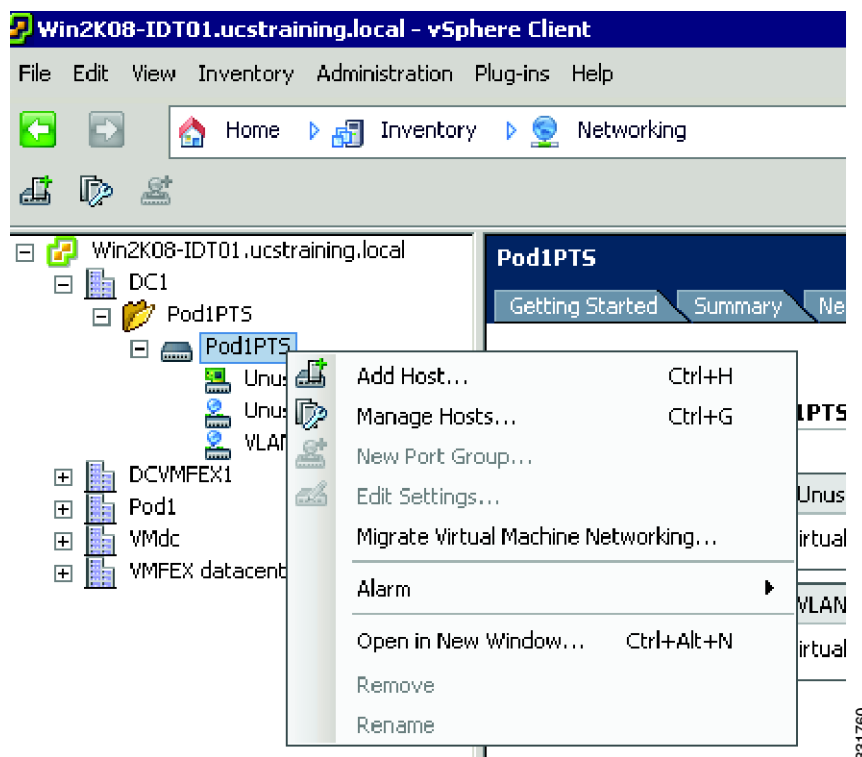
Note   If the ESX host cannot be added to the data center, verify that it is not managed by another vCenter server. Access the host remotely from the other vCenter server and try to add it to the correct vCenter server. Choose the vSphere Enterprise License Plus license.

Step 25 From the vCenter server, choose **Home** > **Inventory** > **Networking** for your selected PODy PTS VDS under PODy, where y is your pod number

Step 26 Add both hosts to the VDS and choose the two VMNICs (the static fixed VNICs needed by VMware for the VMware proxy path).

Step 27 Choose the two VMNICs for each ESX host that is recognized by the staticeth 0 and staticeth 1 interfaces on the Cisco UCS VIC VNIC and that are connected to the vEthernet 1, vEthernet 2, vEthernet 3 and vEthernet 4 interfaces. See Figure 2-24.

Figure 2-24   Viewing details of each VMNIC

Note Use the **View Details** option of each VMNIC to help you choose the proper static VMNICs. In this particular case, VMNIC16 and VMNIC17 are assigned to each host because they are connected to vEthernet 1, vEthernet 2, vEthernet 3 and vEthernet 4 interfaces. See Figure 2-25.

Figure 2-25   *Physical Adapter Details - vEthenet1*
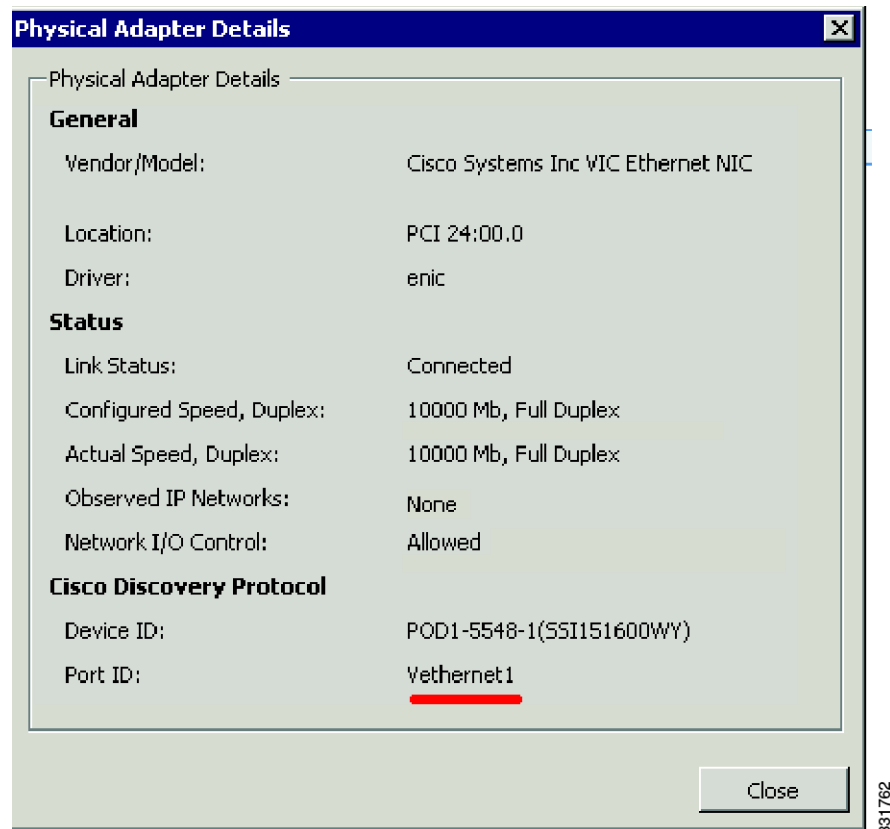
The Physical Adapter Details for the Port ID vEthenet3 is shown in Figure 2-26.

Figure 2-26   *Physical Adapter Details - vEthenet3*

The Physical Adapter Details for the Port ID vEthenet2 is shown Figure 2-27.

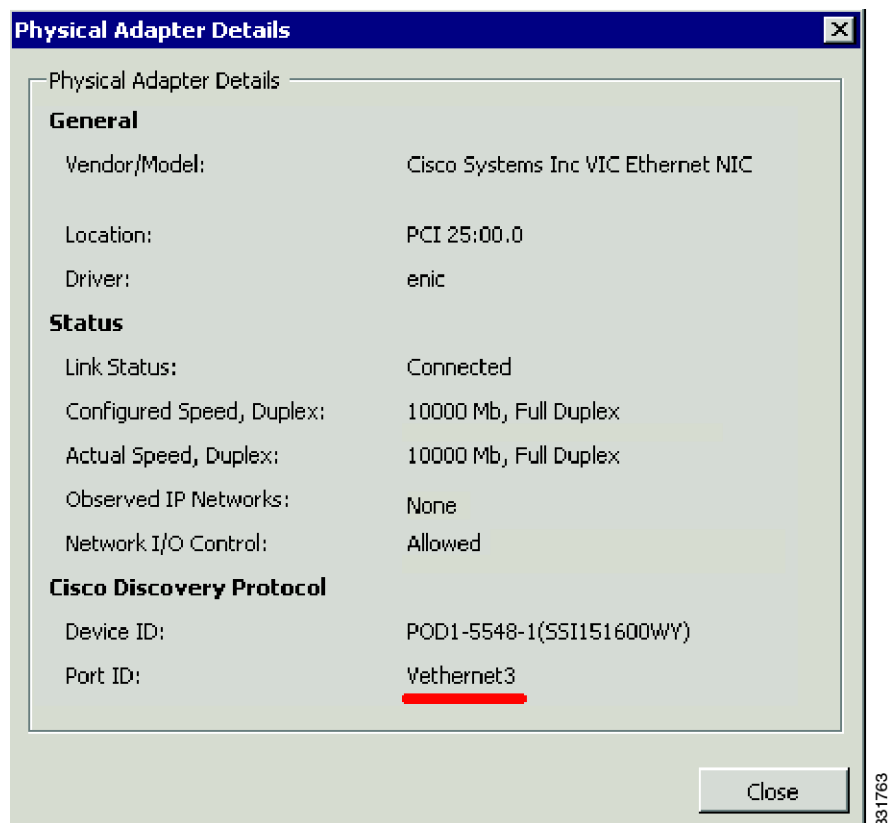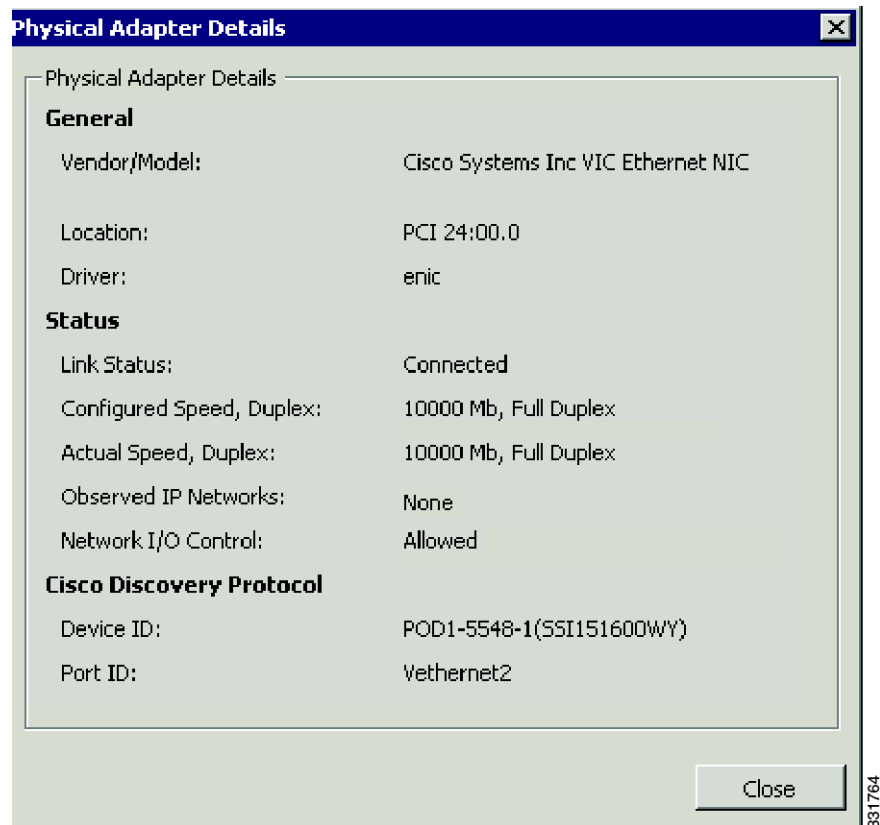*Figure 2-27*   *Physical Adapter Details - vEthenet2*

The Physical Adapter Details for the Port ID vEthenet4 is Figure 2-28.

Figure 2-28   Physical Adapter Details - vEthenet4
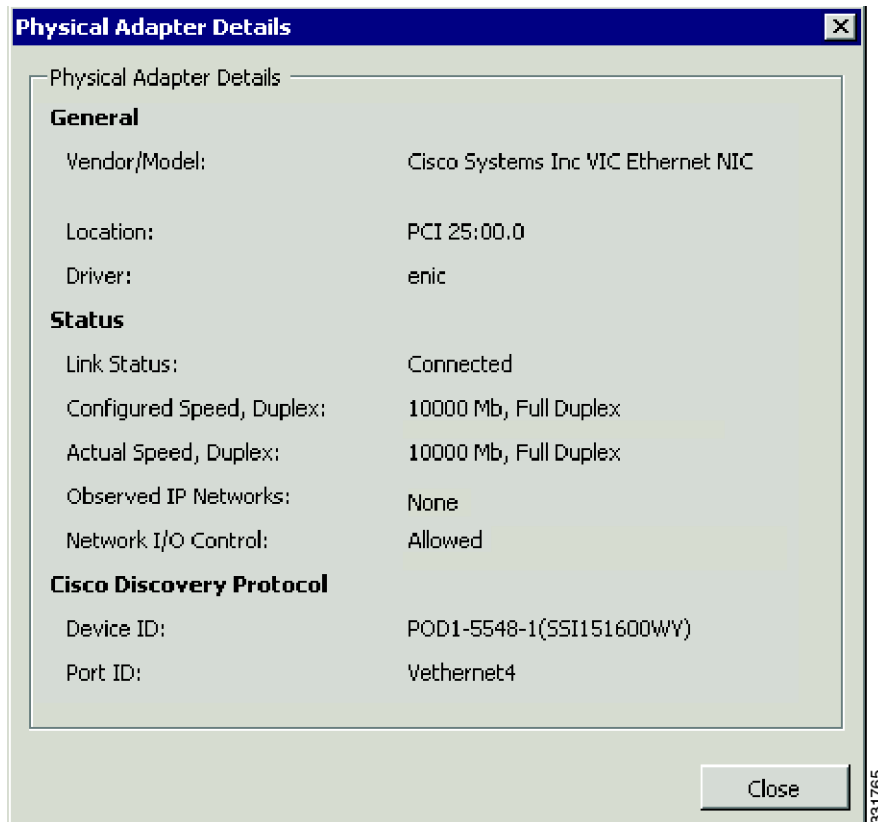
Step 28   Finish the association of the VMNIC with the VEM in the PTS process by leaving the rest of the parameters with their default settings.

Step 29   Choose **Next** until the process of adding the two hosts is finished.

The results of this process are as follows:

- A new VDS, named PTS, has been created.
- The new VDS is based on the two ESX hosts.
- The ESX hosts are each running a VEM in PTS mode.
- Each VEM is associated to the two static uplink VMNICs.
- The static uplink VMNICs are up and connected to the vEthernet 1, vEthernet 2, vEthernet 3 and vEthernet 4 interfaces.

All four VMNICs have to be up and running. Notice the red x next to VMNIC, which indicates that the VMNIC is not connected, and the two static VNICs that have been mapped to the VMNICs to the Pod1PTS VDS on each host as shown in Figure 2-29.

Figure 2-29   Pod1PTS VDS on each host
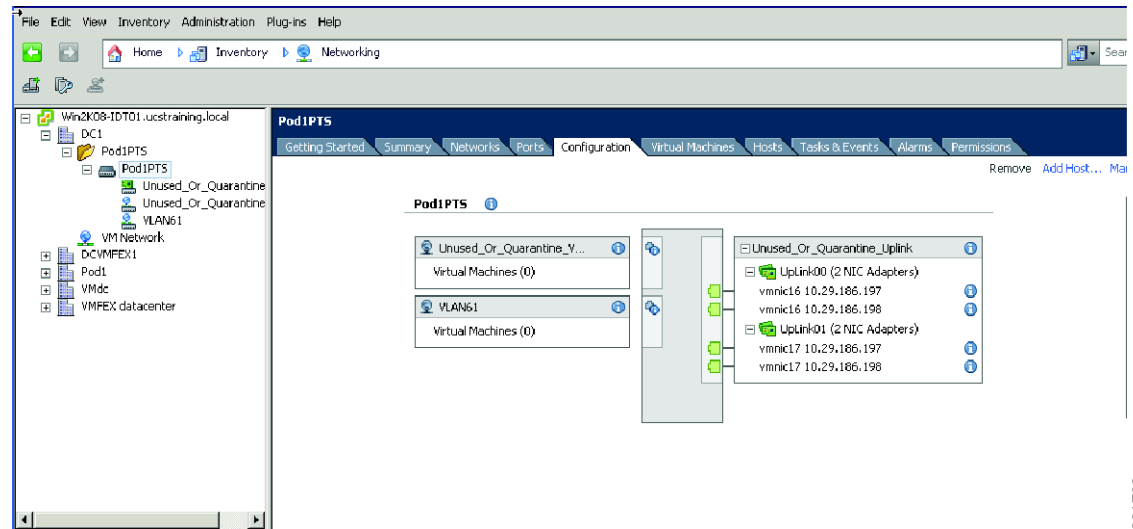
The adapter is first being brought up.

If this interface is a fixed vEthernet interface, the NIV adapter also creates one vEthernet interface on the active Cisco Nexus 5000 Series device and one vEthernet interface on the standby Cisco Nexus 5000 Series device. The NIV adapter uses the same channel number in both the VIF_CREATE messages to the two Cisco Nexus 5000 Series devices. For static fixed vEthernet interface, you must create the vEthernet interface on the two Cisco Nexus 5000 Series devices with the same vEthernet interface number and channel number. For dynamically created fixed vEthernet interface, the two Cisco Nexus 5000 Series devices coordinate to ensure the same channel number on the two Cisco Nexus 5000 Series device are mapped to the same vEthernet interface number.

If this is a floating (static or dynamic) vEthernet interface, the NIV adapter creates one vEthernet interface on the active Cisco Nexus 5000 Series device and one vEthernet interface on the standby Cisco Nexus 5000 Series device. Each Cisco Nexus 5000 Series device runs its own VIC protocol session to the NIV adapter. The NIV adapter uses the same dvport-id in both VIF_CREATE messages to the two Cisco Nexus 5000 Series devices. The two Cisco Nexus 5000 Series devices coordinate to ensure the same vEthernet number is allocated to the two vEthernet interfaces that are mapped to the same dvport_id.

VM-to-VM-FEX Connection Scenario

The VM-to-VM-FEX Connection Scenario is as follows:

Step 1   Create the VM on the ESX host and connect it using VM-FEX (VNLink in the hardware with PTS) to the dynamic vEthernet port on the vPC system using hardware-based redundancy.

Step 2   On ESX Host 1, create a new VM to be migrated from one host to the other host in order to show the floating vEthernet interface. The VM is connected in VM-FEX mode to the vPC system. VM storage needs to be available to both hosts. The storage is the VMFS-formatted Logical Unit Number (LUN) that resides on the FC Storage array and is available across the multihop Fiber Channel over Ethernet (FCoE) network (vfc 400 and vfc 500 interfaces on Qlogic adapter).

Send comments to nexus5k-docfeedback@cisco.com

Step 3   Right-click the host to add the new VM. See Figure 2-30.

Figure 2-30   *Adding the new Virtual Machine*
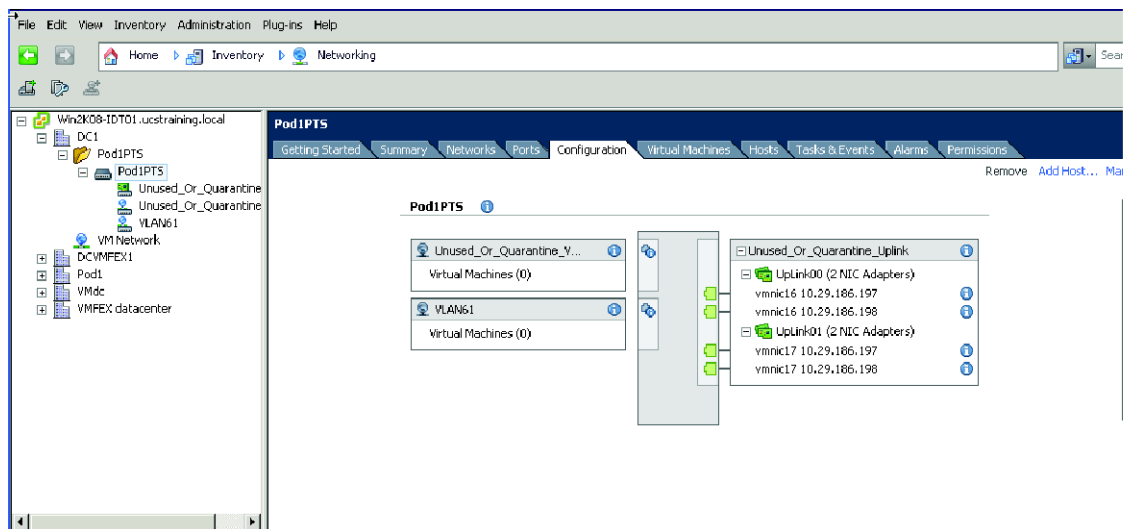
Step 4   Choose the Typical VM, associate the Name to the VM define your DCY (Y is the number of your pod) as the Inventory location and choose the 60GB_shared_datastore.

Step 5   Choose the Linux OS Red Hat Enterprise Linux 4 (64-bit) and click **Next**. See Figure 2-31.

Figure 2-31   Creating a new Virtual Machine
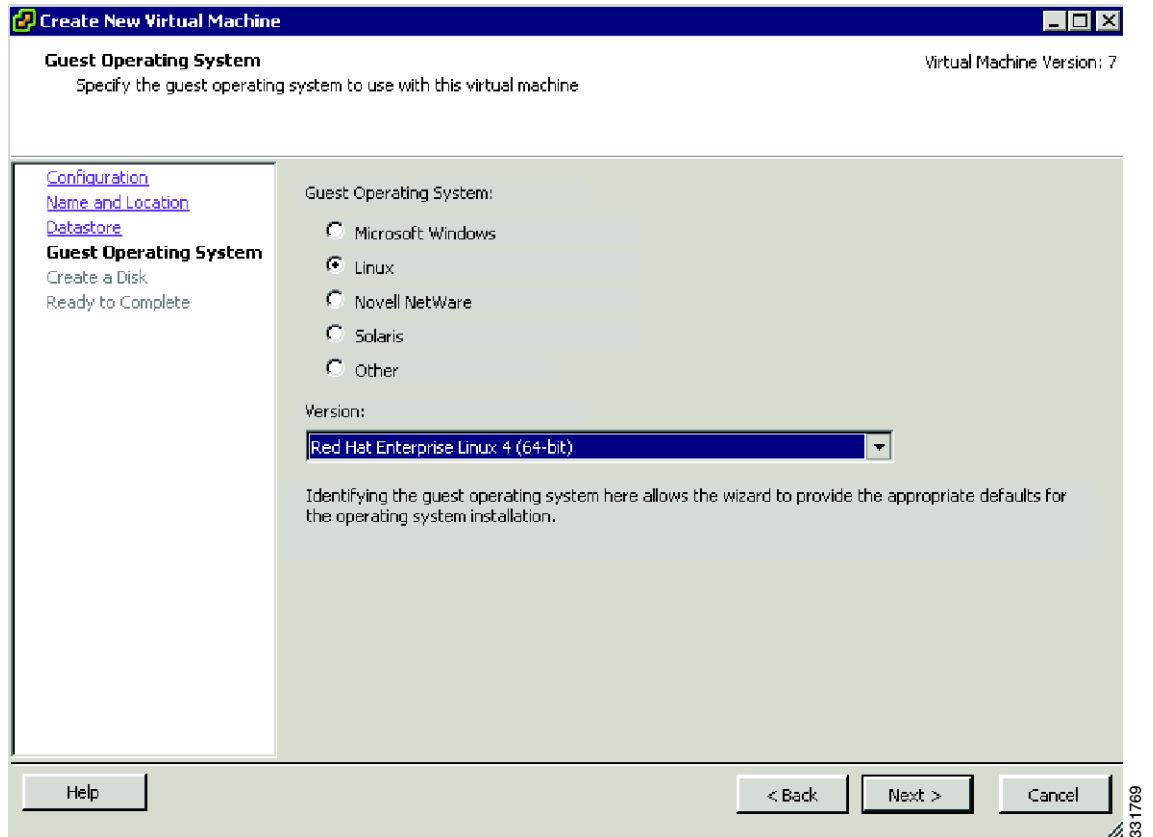
Step 6   Choose the 7-GB to 15GB virtual disk on the shared data store as shown in Figure 2-32.

Figure 2-32   *Shared Datastore*
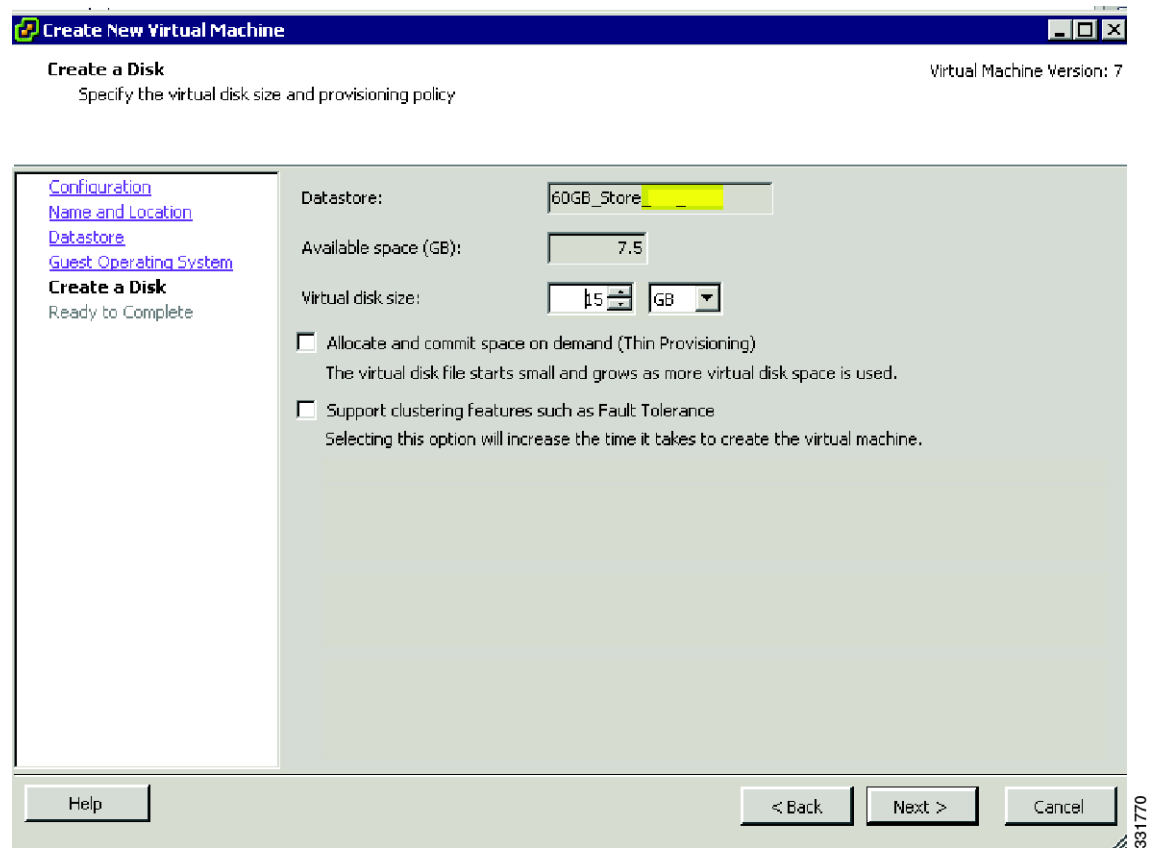
The Ready to Complete Screen is shown in Figure 2-33.

Send comments to nexus5k-docfeedback@cisco.com

Figure 2-33   Ready to Complete Screen
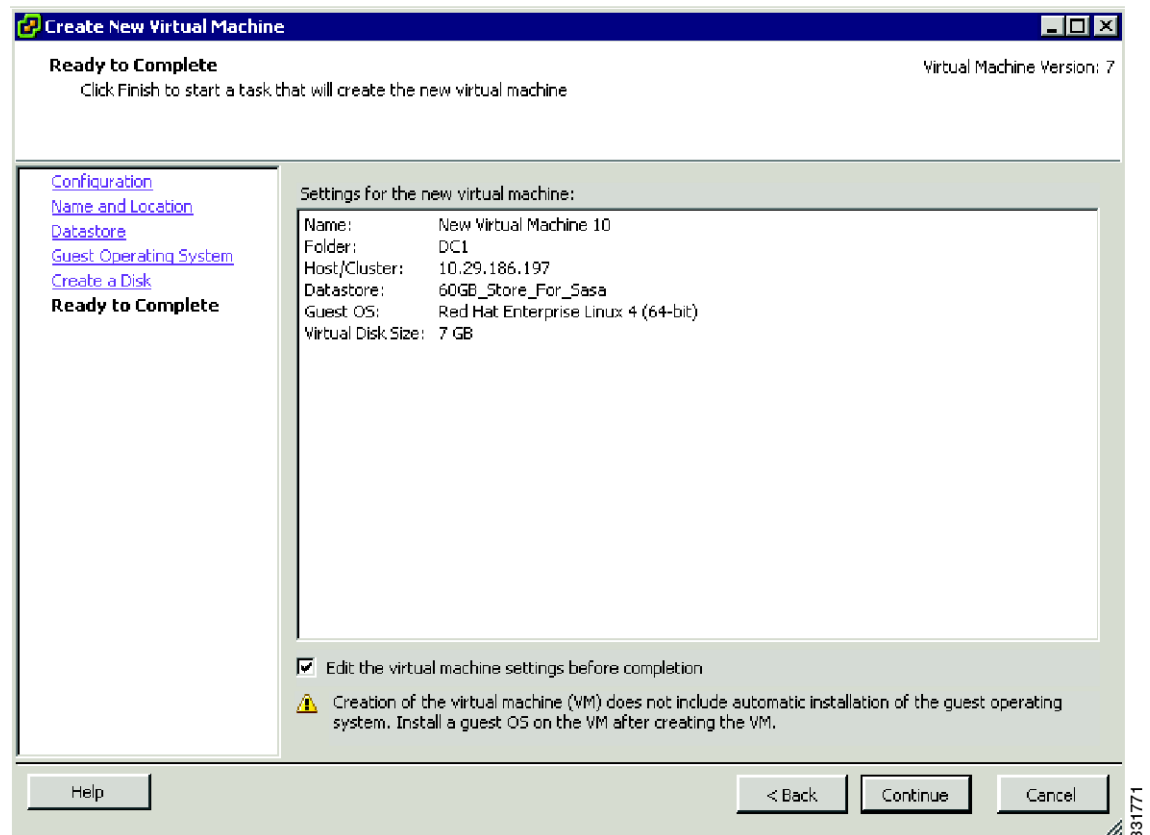
Step 7   Choose **New Nic** to map the new VM VNIC to the VLAN61 port group that was created on the vPC system as shown in Figure 2-34.

Send comments to nexus5k-docfeedback@cisco.com

Figure 2-34   Virtual Machine Properties
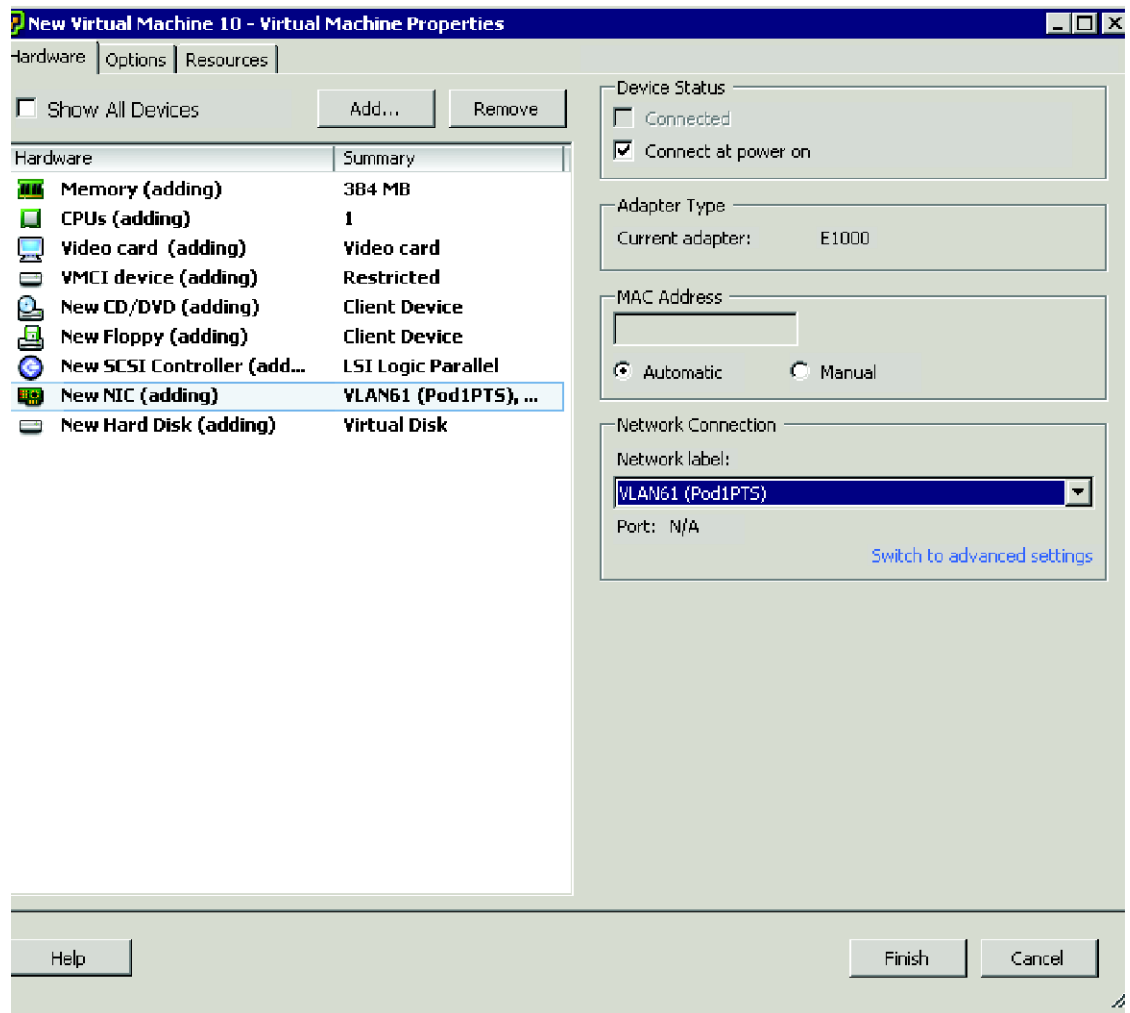
Step 8   Choose **New CD/DVD** drive to map the RedHat ISO image from the shared storage as a new virtual CD/DVD drive as shown in Figure 2-35.

Figure 2-35   *Mapping the RedHat ISO image as a new virtual CD/DVD drive*
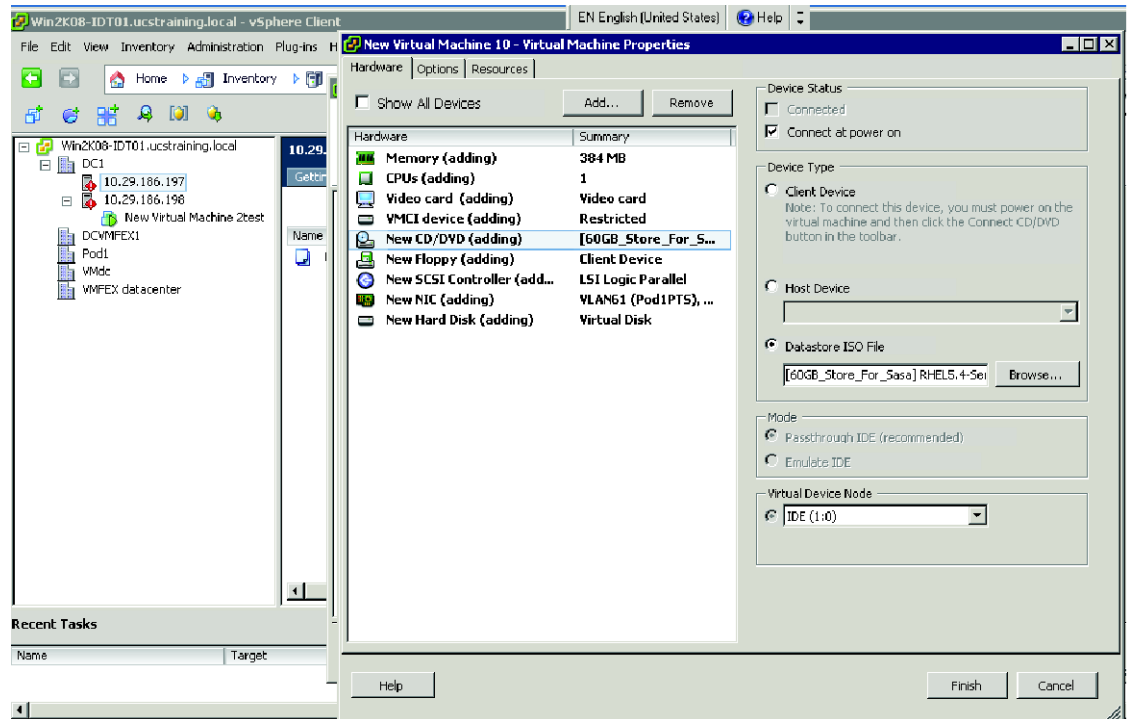
Step 9   Right click **New Virtual Machine** and choose **Power> Power on** the VM and observe the loading of the RedHat OS.

Figure 2-36   *Powering on the Virtual Machine*
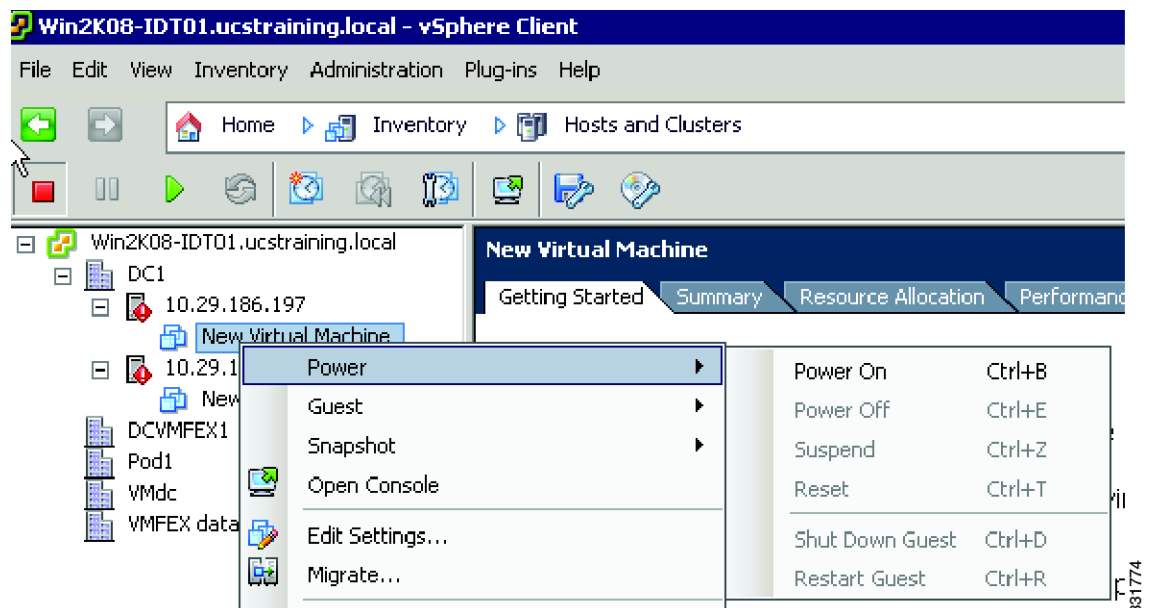
Step 10   From the VM console, notice that the OS has loaded as shown in Figure 2-37.

Figure 2-37   Red Hat Linux 5 Loaded Screen
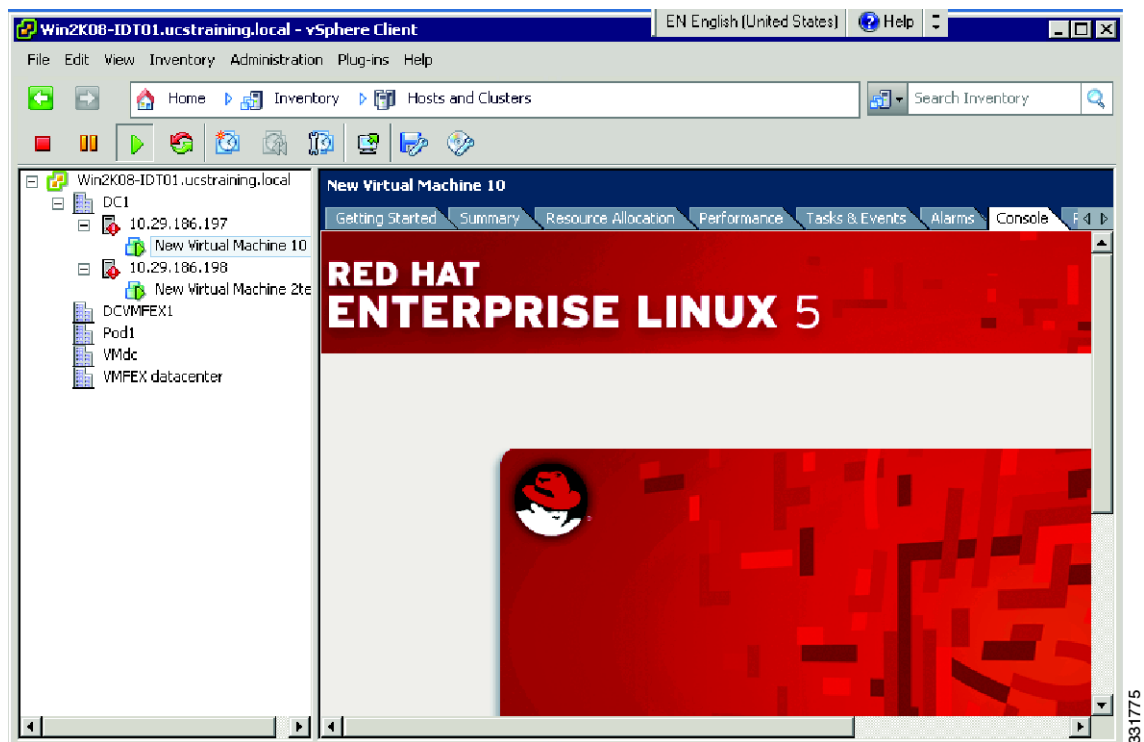
VM-FEX Connectivity Verification Scenario

The VM-FEX Connectivity Verification Scenario is as follows:

Step 1   Observe how the VM is mapped to the dynamic VNIC/VMNIC and connected across VM- FEX to the new automatically created dynamic vEthernet interface on the Cisco Nexus 5548 device.

The Cisco UCS VIC in NIV mode enables the dynamic VNIC when the VM is associated to port group VLAN61 and configures it with the parameters inherited from the port group. The Cisco UCS VIC sends a DCBX VIC message to the Cisco Nexus 5548 device vPC system. The vPC system enables a dynamic vEthernet interface in active/standby mode on the vPC members and connects the VNIC to the vPC system using a VN-Link in the hardware with PTS. The dynamic VNIC is one of the 20 provisioned dynamic VNICs on the UCS VIC with the in-hardware-based failover enabled.

```
POD1-5548-2# show running // this interface was not manually configured but automatically
that is why it is called dynamic
interface Vethernet5
inherit port-profile VLAN61
vsi mac 0050.5684.0000

…output omitted...

OD1-5548-1# show interface vethernet 5
Vethernet5 is up
    Bound Interface is Ethernet1/17
    Hardware is Virtual, address is 0023.04ee.be01
    Port mode is access
    Speed is auto-speed
```

```
      Duplex mode is auto
    300 seconds input rate 0 bits/sec, 0 packets/sec
    300 seconds output rate 0 bits/sec, 0 packets/sec
    Rx
      0 unicast packets  0 multicast packets  0 broadcast packets
      0 input packets  0 bytes
      0 input packet drops
    Tx
      0 unicast packets  5 multicast packets  0 broadcast packets
      5 output packets  1420 bytes
      0 flood packets
      0 output packet drops

  POD1-5548-1#


  POD1-5548-1# show running // this interface was not manually configured but automatically
  that is why it is called dynamic

  interface Vethernet5
    inherit port-profile VLAN61
    vsi mac 0050.5684.0000

  …output omitted...


  POD1-5548-2# show interface vethernet 5
  Vethernet5 is up
     Bound Interface is Ethernet1/17
     Hardware is Virtual, address is 0023.04ee.be01
     Port mode is access
     Speed is auto-speed
     Duplex mode is auto
    300 seconds input rate 0 bits/sec, 0 packets/sec
    300 seconds output rate 0 bits/sec, 0 packets/sec
    Rx
      0 unicast packets  0 multicast packets  0 broadcast packets


  POD1-5548-2# show interface veth 5 detail   // veth 5 is standby on switch NX5548-2

  vif_index: 25
  -------------------------
    veth is bound to interface Ethernet1/17 (0x1a010000)  // veth 5 is maped to phy. Interf.
  eth 1/17
    priority: 0
    vntag: 6
    status: standby
     registered mac info:

  POD1-5548-2#
```

Step 2   Verify that the VM VNIC MAC address is the same as the MAC address of the dynamic VNIC connected to dynamic vEthernet port number 5.

```
  POD1-5548-2# show mac address
  Legend:
          * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
          age - seconds since last seen,+ - primary entry using vPC Peer-Link
     VLAN     MAC Address     Type     age     Secure NTFY  Ports
  ---------+----------------+--------+---------+------+----+------------------
  + 61      0050.5684.0000   dynamic  0           F    F  Po10 // VM connects in VLAN 61
  + 1       588d.090f.0970   dynamic  0           F    F  Po10
```

```
+ 1        588d.090f.0971    dynamic   0           F    F   Po10
+ 1        588d.090f.0b3c    dynamic   0           F    F   Po10
+ 1        588d.090f.0b3d    dynamic   0           F    F   Po10
POD1-5548-2#
```

These interfaces are reported by vCenter Server and are available to the Pod1PTS VDS. Port 32 is in VLAN61 and is a dynamic VMNIC interface that is up and connected to the VM VNIC is shown in Figure 2-38.

Figure 2-38   Dynamic VMNIC Interface

You can list them using the ESX CIMC by opening a KVM session to the host. Alternatively, you can enter the following **ESX console** command to list the 20 dynamic VMNICs that are available to be activated once the VM connects to port group VLAN61:

ifconfig-a | more

The names of dynamic VNICs in the output of the command are vm_vmnic0 through vf_vmnic19 is show in Figure 2-39.

Figure 2-39   Names of Dynamic VNICs in the Output

Step 3   Power off the VM that is connected to the VLAN61 port group and observe how vEthernet 5 moves to a non existing state in the running configuration and the dynamic MAC address disappears from the MAC address table.

```
POD1-5548-2# show interface vethernet 5
```

The interface is no longer in the running configuration (even though port 32, the dynamic VNIC that was the VMNIC interface in the vCenter server in VLAN61, is also down.)

Step 4   Power on the VM again.

Configuration Summary

The VM is powered on and resides on the ESX Host 1 with all the information stored on the shared storage.

The VM was connected to the PODy (where y is the number of your POD) PTS VDS by associating it to port group VLAN61 that was created on the Cisco Nexus 5548 device. The VM has been connected to the vPC system automatically using a VN-Link in the hardware in PTS mode or in VM-FEX mode.

The VEM bits are used in PTS mode to connect the VM VNIC to the VMNIC interface.

In this case, the VMNIC interface is not a real VMNIC but a dynamic VNIC that is presented as an interface to the ESX OS. The dynamic VNIC is enabled when the Cisco UCS VIC creates and configures the VNIC parameters inherited from port group VLAN61.

Sample Configuration

```
!Command: show running-config
!Time: Thu Sep  1 09:29:07 2011

version 5.1(3)N1(1)
feature fcoe

install feature-set virtualization
feature-set virtualization
hostname POD1-5548-1

feature telnet
cfs eth distribute
feature interface-vlan
feature lacp
feature vmfex
feature vpc
feature lldp
feature fex

username adminbackup password 5 !  role network-operator
username admin password 5 $1$W40UpSMJ$RP8cRyIJjLhVYlwls9lcq1  role network-admin
no password strength-check

banner motd #Nexus 5000 Switch
#

ip domain-lookup
class-map type qos class-fcoe
class-map type queuing class-fcoe
  match qos-group 1
class-map type queuing class-all-flood
  match qos-group 2
class-map type queuing class-ip-multicast
  match qos-group 2
class-map type network-qos class-fcoe
  match qos-group 1
class-map type network-qos class-all-flood
  match qos-group 2
class-map type network-qos class-ip-multicast
  match qos-group 2

system qos
  service-policy type qos input fcoe-default-in-policy
```

```
      service-policy type queuing input fcoe-default-in-policy
      service-policy type queuing output fcoe-default-out-policy
      service-policy type network-qos fcoe-default-nq-policy
policy-map type control-plane copp-system-policy
control-plane
      service-policy input copp-system-policy
snmp-server user admin network-admin auth md5 0xb2ee7c583c00370672f6605f06113332
 priv 0xb2ee7c583c00370672f6605f06113332 localizedkey
snmp-server enable traps entity fru

vrf context management
  ip route 0.0.0.0/0 10.29.184.1
vlan 1,61
vlan 200
  fcoe vsan 200
vethernet auto-create
vpc domain 1
  role priority 90
  peer-keepalive destination 10.29.186.196
port-profile type vethernet VLAN61
  switchport access vlan 61
  state enabled
port-profile type ethernet Unused_Or_Quarantine_Uplink
  shutdown
  description Port-group created for Nexus1000V internal usage. Do not use.
  state enabled
port-profile type vethernet Unused_Or_Quarantine_Veth
  shutdown
  description Port-group created for Nexus1000V internal usage. Do not use.
  state enabled
port-profile type vethernet statfloat
      sw access vlan 1
    state enabled
vsan database
  vsan 200
fcdomain fcid database
  vsan 200 wwn 20:00:58:8d:09:0f:09:6e fcid 0x730000 dynamic
  vsan 200 wwn 21:00:00:c0:dd:15:21:7f fcid 0x730001 dynamic
  vsan 200 wwn 21:00:00:c0:dd:12:14:03 fcid 0x730002 dynamic


interface Vlan1

interface Vlan61
  no shutdown
  ip address 10.10.10.1/24

interface port-channel10
  switchport mode trunk
  spanning-tree port type network
  vpc peer-link

interface vfc300
  bind interface Ethernet1/26
  switchport mode E
  switchport trunk allowed vsan 1
  switchport trunk allowed vsan add 200
  no shutdown

interface vfc100
  bind interface Vethernet100

interface vfc400
  bind interface Ethernet1/5
```

```
      no shutdown

interface vfc200
  bind interface Vethernet200

interface vfc500
  bind interface Ethernet1/6
  no shutdown
vsan database
  vsan 200 interface vfc300
  vsan 200 interface vfc100
  vsan 200 interface vfc400
  vsan 200 interface vfc200
  vsan 200 interface vfc500


interface Ethernet1/1

interface Ethernet1/2

interface Ethernet1/3

interface Ethernet1/4

interface Ethernet1/5
  switchport mode trunk

interface Ethernet1/6
  switchport mode trunk

interface Ethernet1/7

interface Ethernet1/8

interface Ethernet1/9

interface Ethernet1/10

interface Ethernet1/11

interface Ethernet1/12

interface Ethernet1/13

interface Ethernet1/14

interface Ethernet1/15

interface Ethernet1/16

interface Ethernet1/17
  switchport mode vntag

interface Ethernet1/18
  switchport mode vntag

interface Ethernet1/19

interface Ethernet1/20

interface Ethernet1/21

interface Ethernet1/22
```

```
interface Ethernet1/23

interface Ethernet1/24
  shutdown

interface Ethernet1/25

interface Ethernet1/26
  switchport mode trunk

interface Ethernet1/27
  switchport mode trunk
  spanning-tree port type network
  channel-group 10

interface Ethernet1/28
  switchport mode trunk
  spanning-tree port type network
  channel-group 10

interface Ethernet1/29
  switchport mode trunk
  spanning-tree port type network
  channel-group 10

interface Ethernet1/30
  switchport mode trunk
  spanning-tree port type network
  channel-group 10

interface Ethernet1/31

interface Ethernet1/32

interface mgmt0
  ip address 10.29.186.195/22

interface Vethernet1
  bind interface Ethernet1/17 channel 10

interface Vethernet2
  bind interface Ethernet1/18 channel 10

interface Vethernet3
  bind interface Ethernet1/17 channel 11

interface Vethernet4
  bind interface Ethernet1/18 channel 11

interface Vethernet5
  inherit port-profile VLAN61
 vsi mac 0050.5684.0004

interface Vethernet100
  shutdown
  switchport mode trunk
  bind interface Ethernet1/17 channel 3

interface Vethernet145
  inherit port-profile statfloat
  vsi mac 0050.5601.0144

interface Vethernet200
  shutdown
```

```
    switchport mode trunk
    bind interface Ethernet1/18 channel 3

line console
line vty
boot kickstart bootflash:/n5000-uk9-kickstart.5.1.3.N1.0.223.gbin
boot system bootflash:/n5000-uk9.5.1.3.N1.0.223.gbin
svs connection 2VC
  protocol vmware-vim
  remote ip address 10.29.186.125 port 80 vrf managment
  vmware dvs uuid "8e c7 04 50 cb 46 d1 71-d1 98 68 74 a3 26 6d ff" datacenter-n
ame DC1
  connect
logging monitor 4
zone default-zone permit vsan 1
zone default-zone permit vsan 200




!Command: show running-config
!Time: Thu Sep  1 09:36:47 2011

version 5.1(3)N1(1)
feature fcoe

install feature-set virtualization
feature-set virtualization
hostname POD1-5548-2

feature telnet
cfs eth distribute
feature interface-vlan
feature lacp
feature vmfex
feature vpc
feature lldp
feature fex

username adminbackup password 5 !  role network-operator
username admin password 5 $1$2/VZLE3/$esbYsWazKUXTc/G3xDXd3/  role network-admin
no password strength-check

banner motd #Nexus 5000 Switch
#

ip domain-lookup
class-map type qos class-fcoe
class-map type queuing class-fcoe
  match qos-group 1
class-map type queuing class-all-flood
  match qos-group 2
class-map type queuing class-ip-multicast
  match qos-group 2
class-map type network-qos class-fcoe
  match qos-group 1
class-map type network-qos class-all-flood
  match qos-group 2
class-map type network-qos class-ip-multicast
  match qos-group 2

system qos
  service-policy type qos input fcoe-default-in-policy
  service-policy type queuing input fcoe-default-in-policy
```

Cisco Nexus 5000 Series NX-OS Operations Guide, Release 5.1(3)N1(1)

Send comments to nexus5k-docfeedback@cisco.com

```
    service-policy type queuing output fcoe-default-out-policy
    service-policy type network-qos fcoe-default-nq-policy
policy-map type control-plane copp-system-policy
control-plane
    service-policy input copp-system-policy
snmp-server user admin network-admin auth md5 0x659dc61aafac579799b0801bb80a7435
 priv 0x659dc61aafac579799b0801bb80a7435 localizedkey
snmp-server enable traps entity fru

vrf context management
    ip route 0.0.0.0/0 10.29.184.1
vlan 1,61
vlan 200
    fcoe vsan 200
vethernet auto-create
vpc domain 1
    role priority 100
    peer-keepalive destination 10.29.186.195
port-profile type vethernet VLAN61
    switchport access vlan 61
    state enabled
port-profile type ethernet Unused_Or_Quarantine_Uplink
    shutdown
    description Port-group created for Nexus1000V internal usage. Do not use.
    state enabled
port-profile type vethernet Unused_Or_Quarantine_Veth
    shutdown
    description Port-group created for Nexus1000V internal usage. Do not use.
    state enabled
port-profile type vethernet statfloat
    switchport access vlan 1
     state enabled
vlan 200
    fcoe vsan 200
vethernet auto-create
vpc domain 1
    role priority 100
    peer-keepalive destination 10.29.186.195
vsan database
    vsan 200
fcdomain fcid database
    vsan 200 wwn 21:00:00:c0:dd:15:21:7d fcid 0x080000 dynamic
    vsan 200 wwn 21:00:00:c0:dd:12:14:01 fcid 0x080001 dynamic
    vsan 200 wwn 20:00:58:8d:09:0f:09:6f fcid 0x080002 dynamic

interface port-channel10
    switchport mode trunk
    spanning-tree port type network
    vpc peer-link

interface vfc300
    bind interface Ethernet1/26
    switchport mode E
    switchport trunk allowed vsan 1
    switchport trunk allowed vsan add 200
    no shutdown

interface vfc100
    bind interface Vethernet100

interface vfc400
    bind interface Ethernet1/5
    no shutdown
```

```
interface vfc200
  bind interface Vethernet200

interface vfc500
  bind interface Ethernet1/6
  no shutdown
vsan database
  vsan 200 interface vfc100
  vsan 200 interface vfc400
  vsan 200 interface vfc200
  vsan 200 interface vfc500

interface Ethernet1/1

interface Ethernet1/2

interface Ethernet1/3

interface Ethernet1/4

interface Ethernet1/5
  switchport mode trunk

interface Ethernet1/6
  switchport mode trunk

interface Ethernet1/7

interface Ethernet1/8

interface Ethernet1/9

interface Ethernet1/10

interface Ethernet1/11

interface Ethernet1/12

interface Ethernet1/13

interface Ethernet1/14

interface Ethernet1/15

interface Ethernet1/16

interface Ethernet1/17
  switchport mode vntag

interface Ethernet1/18
  switchport mode vntag

interface Ethernet1/19
  shutdown

interface Ethernet1/20
  shutdown

interface Ethernet1/21
  shutdown

interface Ethernet1/22
  shutdown
```

```
interface Ethernet1/23
  shutdown

interface Ethernet1/24
  shutdown

interface Ethernet1/25
  shutdown

interface Ethernet1/26
  switchport mode trunk

interface Ethernet1/27
  switchport mode trunk
  channel-group 10

interface Ethernet1/28
  switchport mode trunk
  channel-group 10

interface Ethernet1/29
  switchport mode trunk
  channel-group 10

interface Ethernet1/30
  switchport mode trunk
  channel-group 10

interface Ethernet1/31

interface Ethernet1/32

interface Ethernet2/1

interface Ethernet2/2

interface Ethernet2/3

interface Ethernet2/4

interface Ethernet2/5

interface Ethernet2/6

interface Ethernet2/7

interface Ethernet2/8

interface mgmt0
  ip address 10.29.186.196/22

interface Vethernet1
  bind interface Ethernet1/17 channel 10

interface Vethernet2
  bind interface Ethernet1/18 channel 10

interface Vethernet3
  bind interface Ethernet1/17 channel 11

interface Vethernet4
  bind interface Ethernet1/18 channel 11

interface Vethernet5
```

```
    inherit port-profile VLAN61
    vsi mac 0050.5684.0004


interface Vethernet100
  shutdown
  switchport mode trunk
  bind interface Ethernet1/17 channel 4

interface Vethernet145
  inherit port-profile statfloat
  vsi mac 0050.5601.0144

interface Vethernet200
  shutdown
  switchport mode trunk
  bind interface Ethernet1/18 channel 4
line console
line vty
boot kickstart bootflash:/n5000-uk9-kickstart.5.1.3.N1.0.223.gbin
boot system bootflash:/n5000-uk9.5.1.3.N1.0.223.gbin
svs connection 2VC
  protocol vmware-vim
  remote ip address 10.29.186.125 port 80 vrf management
zone default-zone permit vsan 1
zone default-zone permit vsan 200


pod1-5596-2# sh run

!Command: show running-config
!Time: Thu Sep  1 12:54:08 2011

version 5.1(3)N1(1)
feature fcoe

install feature-set virtualization
feature-set virtualization
switchname pod1-5596-2

feature telnet
feature lacp
feature vmfex
feature lldp
feature fex

username admin password 5 $1$sdMFwb78$3J2cgv7YbedIzxXlrm66H0  role network-admin
no password strength-check

banner motd #Nexus 5000 Switch
#

ip domain-lookup
class-map type qos class-fcoe
class-map type queuing class-fcoe
  match qos-group 1
class-map type queuing class-all-flood
  match qos-group 2
class-map type queuing class-ip-multicast
  match qos-group 2
class-map type network-qos class-fcoe
  match qos-group 1
class-map type network-qos class-all-flood
  match qos-group 2
```

```
class-map type network-qos class-ip-multicast
  match qos-group 2
system qos
  service-policy type qos input fcoe-default-in-policy
  service-policy type network-qos fcoe-default-nq-policy
  service-policy type queuing input fcoe-default-in-policy
  service-policy type queuing output fcoe-default-out-policy
class-map type control-plane match-any copp-system-class-arp
  match protocol arp
class-map type control-plane match-any copp-system-class-bgp
  match protocol bgp
class-map type control-plane match-any copp-system-class-bridging
  match protocol bridging
class-map type control-plane match-any copp-system-class-cdp
  match protocol cdp
class-map type control-plane match-any copp-system-class-default
  match protocol default
class-map type control-plane match-any copp-system-class-dhcp
  match protocol dhcp
class-map type control-plane match-any copp-system-class-eigrp
  match protocol eigrp
class-map type control-plane match-any copp-system-class-exception
  match protocol exception
class-map type control-plane match-any copp-system-class-glean
  match protocol glean
class-map type control-plane match-any copp-system-class-hsrp-vrrp
  match protocol hsrp_vrrp
class-map type control-plane match-any copp-system-class-icmp-echo
  match protocol icmp_echo
class-map type control-plane match-any copp-system-class-igmp
  match protocol igmp
class-map type control-plane match-any copp-system-class-ip-frag
  match protocol ip_frag
class-map type control-plane match-any copp-system-class-isis-dce
  match protocol isis_dce
class-map type control-plane match-any copp-system-class-lacp
  match protocol lacp
class-map type control-plane match-any copp-system-class-lldp
  match protocol lldp_dcbx
class-map type control-plane match-any copp-system-class-mgmt
  match protocol mgmt
class-map type control-plane match-any copp-system-class-msdp
  match protocol msdp
class-map type control-plane match-any copp-system-class-multicast
  match protocol multicast
class-map type control-plane match-any copp-system-class-ospf
  match protocol ospf
class-map type control-plane match-any copp-system-class-pim
  match protocol pim
class-map type control-plane match-any copp-system-class-reg
  match protocol reg
class-map type control-plane match-any copp-system-class-rip
  match protocol rip
class-map type control-plane match-any copp-system-class-same-if
  match protocol same_if
class-map type control-plane match-any copp-system-class-ttl
  match protocol ttl
control-plane
  service-policy input copp-system-policy-default
slot 1
  port 39-48 type fc
snmp-server user admin network-admin auth md5 0x338f7b1376b2f1548acf1f8295c0e734
 priv 0x338f7b1376b2f1548acf1f8295c0e734 localizedkey
snmp-server enable traps entity fru
```

```
snmp-server community labpublic group network-operator

vrf context management
  ip route 0.0.0.0/0 10.29.184.1
vlan 1
vlan 200
  fcoe vsan 200
vsan database
  vsan 200
fcdomain fcid database
  vsan 1 wwn 26:02:00:01:55:35:0c:d3 fcid 0xc90000 dynamic
  vsan 1 wwn 21:00:00:c0:dd:15:21:7f fcid 0xc90001 dynamic
  vsan 1 wwn 21:00:00:c0:dd:12:14:01 fcid 0xc90002 dynamic
  vsan 1 wwn 21:00:00:c0:dd:15:21:7d fcid 0xc90003 dynamic
  vsan 200 wwn 26:02:00:01:55:35:0c:d3 fcid 0x340000 dynamic


interface vfc300
  bind interface Ethernet1/32
  switchport mode E
  switchport trunk allowed vsan 1
  switchport trunk allowed vsan add 200
  no shutdown
vsan database
  vsan 200 interface vfc300
  vsan 200 interface fc1/48


interface fc1/39

interface fc1/40
  no shutdown

interface fc1/41

interface fc1/42

interface fc1/43

interface fc1/44

interface fc1/45

interface fc1/46

interface fc1/47

interface fc1/48
  no shutdown

interface Ethernet1/1
  switchport trunk allowed vlan none
  spanning-tree port type edge trunk

interface Ethernet1/2
  switchport mode trunk
  switchport trunk allowed vlan 1,1000

interface Ethernet1/3

interface Ethernet1/4

interface Ethernet1/5
```

```
interface Ethernet1/6

interface Ethernet1/7

interface Ethernet1/8

interface Ethernet1/9

interface Ethernet1/10

interface fc1/41

interface fc1/42

interface fc1/43

interface fc1/44

interface fc1/45

interface fc1/46

interface fc1/47

interface fc1/48
  no shutdown

interface Ethernet1/1
  switchport trunk allowed vlan none
  spanning-tree port type edge trunk

interface Ethernet1/2
  switchport mode trunk
  switchport trunk allowed vlan 1,1000

interface Ethernet1/3

interface Ethernet1/4

interface Ethernet1/5

interface Ethernet1/6

interface Ethernet1/7

interface Ethernet1/8

interface Ethernet1/9

interface Ethernet1/10

interface Ethernet1/11

interface Ethernet1/12

interface Ethernet1/13

interface Ethernet1/14

interface Ethernet1/15

interface Ethernet1/16

interface Ethernet1/17
```

Send comments to nexus5k-docfeedback@cisco.com

```
interface Ethernet1/18

interface Ethernet1/19

interface Ethernet1/20

interface Ethernet1/21

interface Ethernet1/22

interface Ethernet1/23

interface Ethernet1/24

interface Ethernet1/25

interface Ethernet1/26
  shutdown

interface Ethernet1/27

interface Ethernet1/28

interface Ethernet1/29

interface Ethernet1/30

interface Ethernet1/31

interface Ethernet1/32
  switchport mode trunk

interface Ethernet1/33

interface Ethernet1/34

interface Ethernet1/35

interface Ethernet1/36

interface Ethernet1/37

interface Ethernet1/38

interface mgmt0
  ip address 10.29.186.193/22
line console
line vty
boot kickstart bootflash:/n5000-uk9-kickstart.5.1.3.N1.0.212.bin
boot system bootflash:/n5000-uk9.5.1.3.N1.0.212.bin
interface fc1/39
interface fc1/40
interface fc1/41
interface fc1/42
interface fc1/43
interface fc1/44
interface fc1/45
interface fc1/46
interface fc1/47
interface fc1/48
  switchport mode F
logging monitor 6
zone default-zone permit vsan 1
```

Send comments to nexus5k-docfeedback@cisco.com

```
      zone default-zone permit vsan 200


      !Command: show running-config
      !Time: Thu Sep  1 09:42:50 2011

      version 5.1(3)N1(1)
      feature fcoe

      install feature-set virtualization
      feature-set virtualization
      hostname pod1-5596-1

      feature telnet
      feature vmfex
      feature lldp
      feature fex

      username admin password 5 $1$aFZpnyOJ$kKzqLw8Ftu2I.T8mS.w1O0  role network-admin
      no password strength-check

      banner motd #Nexus 5000 Switch
      #

      ip domain-lookup
      class-map type qos class-fcoe
      class-map type queuing class-fcoe
        match qos-group 1
      class-map type queuing class-all-flood
        match qos-group 2
      class-map type queuing class-ip-multicast
        match qos-group 2
      class-map type network-qos class-fcoe
        match qos-group 1
      class-map type network-qos class-all-flood
        match qos-group 2
      class-map type network-qos class-ip-multicast
        match qos-group 2
      system qos
        service-policy type qos input fcoe-default-in-policy
        service-policy type network-qos fcoe-default-nq-policy
        service-policy type queuing input fcoe-default-in-policy
        service-policy type queuing output fcoe-default-out-policy
      class-map type control-plane match-any copp-system-class-arp
        match protocol arp
      class-map type control-plane match-any copp-system-class-bgp
        match protocol bgp
      class-map type control-plane match-any copp-system-class-bridging
        match protocol bridging

        police cir 1024 kbps bc 4800000 bytes
        class copp-system-class-icmp-echo
          police cir 640 kbps bc 3600000 bytes
        class copp-system-class-ospf
          police cir 1024 kbps bc 4800000 bytes
        class copp-system-class-reg
          police cir 1024 kbps bc 4800000 bytes
        class copp-system-class-rip
          police cir 1024 kbps bc 4800000 bytes
        class copp-system-class-unicast
          police cir 1024 kbps bc 3200000 bytes
        class copp-system-class-multicast
          police cir 1024 kbps bc 3200000 bytes
        class copp-system-class-ip-frag
```

```
        police cir 1024 kbps bc 3200000 bytes
      class copp-system-class-same-if
        police cir 1024 kbps bc 3200000 bytes
      class copp-system-class-ttl
        police cir 1024 kbps bc 3200000 bytes
      class copp-system-class-default
        police cir 2048 kbps bc 6400000 bytes
    policy-map type control-plane copp-system-policy-scaled-l3
      class copp-system-class-igmp
        police cir 1200 kbps bc 65535 bytes
      class copp-system-class-pim
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-bridging
        police cir 20000 kbps bc 4800000 bytes
      class copp-system-class-arp
        police cir 4000 kbps bc 3600000 bytes
      class copp-system-class-dhcp
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-mgmt
        police cir 12000 kbps bc 4800000 bytes
      class copp-system-class-lacp
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-lldp
        police cir 2048 kbps bc 4800000 bytes
      class copp-system-class-udld
        police cir 2048 kbps bc 4800000 bytes
      class copp-system-class-msdp
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-vic
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-cdp
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-bgp
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-eigrp
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-exception
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-glean
        police cir 4000 kbps bc 4800000 bytes
      class copp-system-class-hsrp-vrrp
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-icmp-echo
        police cir 4000 kbps bc 3600000 bytes
      class copp-system-class-ospf
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-reg
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-rip
        police cir 1024 kbps bc 4800000 bytes
      class copp-system-class-unicast
        police cir 1024 kbps bc 3200000 bytes
      class copp-system-class-multicast
        police cir 4000 kbps bc 3200000 bytes
      class copp-system-class-ip-frag
        police cir 1024 kbps bc 3200000 bytes
      class copp-system-class-same-if
        police cir 1024 kbps bc 3200000 bytes
      class copp-system-class-ttl
        police cir 1024 kbps bc 3200000 bytes
      class copp-system-class-default
        police cir 2048 kbps bc 6400000 bytes
    control-plane
      service-policy input copp-system-policy-default
```

```
slot 1
  port 39-48 type fc
snmp-server user admin network-admin auth md5 0x512632220ce767d7bf7aed2f7af4b8e9
 priv 0x512632220ce767d7bf7aed2f7af4b8e9 localizedkey
snmp-server enable traps entity fru
snmp-server community labpublic group network-operator

vrf context management
  ip route 0.0.0.0/0 10.29.184.1
vlan 1
vlan 200
  fcoe vsan 200
port-profile type ethernet Unused_Or_Quarantine_Uplink
  shutdown
  description Port-group created for Nexus1000V internal usage. Do not use.
  state enabled
port-profile type vethernet Unused_Or_Quarantine_Veth
  shutdown
  description Port-group created for Nexus1000V internal usage. Do not use.
  state enabled

vsan database
  vsan 200
fcdomain fcid database
  vsan 1 wwn 26:01:00:01:55:35:0c:d3 fcid 0x9a0000 dynamic
  vsan 1 wwn 21:00:00:c0:dd:14:57:4d fcid 0x9a0001 dynamic
  vsan 1 wwn 21:00:00:c0:dd:12:14:03 fcid 0x9a0002 dynamic
  vsan 1 wwn 21:00:00:c0:dd:15:21:7f fcid 0x9a0003 dynamic
  vsan 200 wwn 26:01:00:01:55:35:0c:d3 fcid 0xe50000 dynamic


interface vfc300
  bind interface Ethernet1/32
  switchport mode E
  switchport trunk allowed vsan 1
  switchport trunk allowed vsan add 200
  no shutdown
vsan database
  vsan 200 interface vfc300
  vsan 200 interface fc1/48


interface fc1/39

interface fc1/40
  no shutdown

interface fc1/41

interface fc1/42

interface fc1/43

interface fc1/44

interface fc1/45

interface fc1/46

interface fc1/47

interface fc1/48
  no shutdown
```

```
interface Ethernet1/1
  switchport mode trunk
  switchport trunk allowed vlan 1,1000
  spanning-tree port type edge trunk

interface Ethernet1/2

interface Ethernet1/3

interface Ethernet1/4

interface Ethernet1/5

interface Ethernet1/6

interface Ethernet1/7

interface Ethernet1/8

interface Ethernet1/9

interface Ethernet1/10

interface Ethernet1/11

interface Ethernet1/12

interface Ethernet1/13

interface Ethernet1/14

interface Ethernet1/15

interface Ethernet1/16

interface Ethernet1/17

interface Ethernet1/18

interface Ethernet1/19

interface Ethernet1/20

interface Ethernet1/21

interface Ethernet1/22

interface Ethernet1/23

interface Ethernet1/24

interface Ethernet1/25

interface Ethernet1/26

interface Ethernet1/27

interface Ethernet1/28

interface Ethernet1/29

interface Ethernet1/30

interface Ethernet1/31
```

```
interface Ethernet1/32
  switchport mode trunk

interface Ethernet1/33

interface Ethernet1/34

interface Ethernet1/35

interface Ethernet1/36

interface Ethernet1/37

interface Ethernet1/38

interface mgmt0
  ip address 10.29.186.194/22
line console
line vty
boot kickstart bootflash:/n5000-uk9-kickstart.5.1.3.N1.0.212.bin
boot system bootflash:/n5000-uk9.5.1.3.N1.0.212.bin
interface fc1/39
interface fc1/40
interface fc1/41
interface fc1/42
interface fc1/43
interface fc1/44
interface fc1/45
interface fc1/46
interface fc1/47
interface fc1/48
zone default-zone permit vsan 1
zone default-zone permit vsan 200
```