UCS RHEL 6 KVM and Palo Integration

TABLE OF CONTENTS

1 Introduction

The purpose of this document is to provide a step by step sample configuration on integrating UCS Dynamic VNICs with RHEL KVM hypervisor.

Similar to UCS integration with ESX it is now possible to present Dynamic VNICs to a Virtual Machine running on a KVM hypervisor. You get all the same benefits with KVM as you do with ESX. Improved network performance, network visibility, and enhanced network options can all be applied to a VM.

2 Test Requirements

- Software
  - UCS 2.x image
- Hardware Requirements
  - UCS Test system
    - At least one B series blade with a Cisco Palo card
  - RHEL 6 or 6.1 with KVM installed and enabled

3 Test Cases

3.1 Prerequisites and Caveats

Integration with KVM has only been tested with RHEL 6 and 6.1. It is highly recommended to use 6.1. Other linux distributions with KVM have not been tested.

UCS requires the 2.x code be loaded. All components in the UCS system should be upgraded the 2.x code. This includes the BIOS and Palo Adapters.

The UCS Service Profile needs to have Dynamic VNICs assigned. The number of Dynamic VNICs that can be assigned to a Service Profile is still tied to the number of Virtual Interfaces (VIFs) that can be supported per blade slot. The number of VIFs is still based off the number of IOM to Fabric Interconnect Uplinks. That equation is as follows

(# of Uplinks * 15) – 2

This means the number of VIFs you can support with 1 uplink is 13, 2 uplinks is 28, and 4 uplinks is 58.

The running OS still needs to see at least one real VNIC for network connectivity though it is highly recommended to have two. One VNIC on fabric A and one VNIC on fabric B.

If you assume a minimum of 2 VNICs this means you can support a maximum number of Dynamic VNICs of 11 with 1 uplink, 26 with 2 uplinks, and 56 with 4 uplinks.

If you create virtual Fibre Channel interfaces that also requires a VIF and should be subtracted from the total number of available VIFs to create a dynamic VNIC from. The full equation is as follows

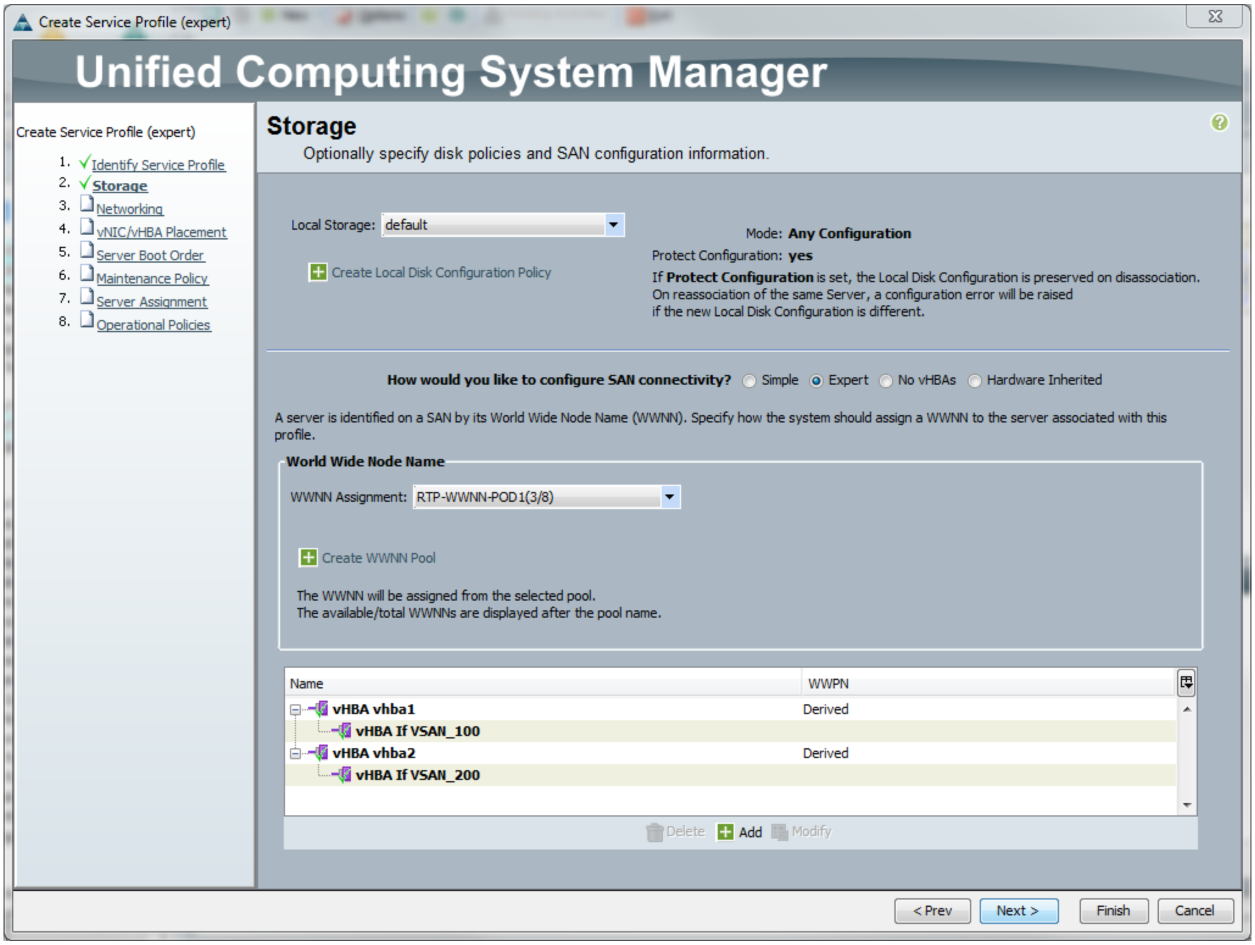
((# of Uplinks * 15) – 2) - #VNICs - #VFCs = # of dynamic VNICs that can be created

3.2 Creation of UCS Service Profile

A Service Profile for the RHEL host should be created. It should contain at least 2 VNICs (one on each fabric) and if desired 2 vHBAs (Also one on each Fabric).

To start we will define only 10 VNICs but this number can be increased later depending on the number of IOM uplinks that are active.

Here we create the vHBAs. Note we create two vHBAs using WWN pool. One vHBA on each fabric.



Create Service Profile (expert)

Unified Computing System Manager

Create Service Profile (expert)

1. ✓ Identify Service Profile
2. ✓ **Storage**
3. Networking
4. vNIC/vHBA Placement
5. Server Boot Order
6. Maintenance Policy
7. Server Assignment
8. Operational Policies

Storage

Optionally specify disk policies and SAN configuration information.

Local Storage: default

Create Local Disk Configuration Policy

Mode: **Any Configuration**

Protect Configuration: **yes**

If **Protect Configuration** is set, the Local Disk Configuration is preserved on disassociation.
On reassociation of the same Server, a configuration error will be raised
if the new Local Disk Configuration is different.

How would you like to configure SAN connectivity? ○ Simple ● Expert ○ No vHBAs ○ Hardware Inherited

A server is identified on a SAN by its World Wide Node Name (WWNN). Specify how the system should assign a WWNN to the server associated with this profile.

World Wide Node Name

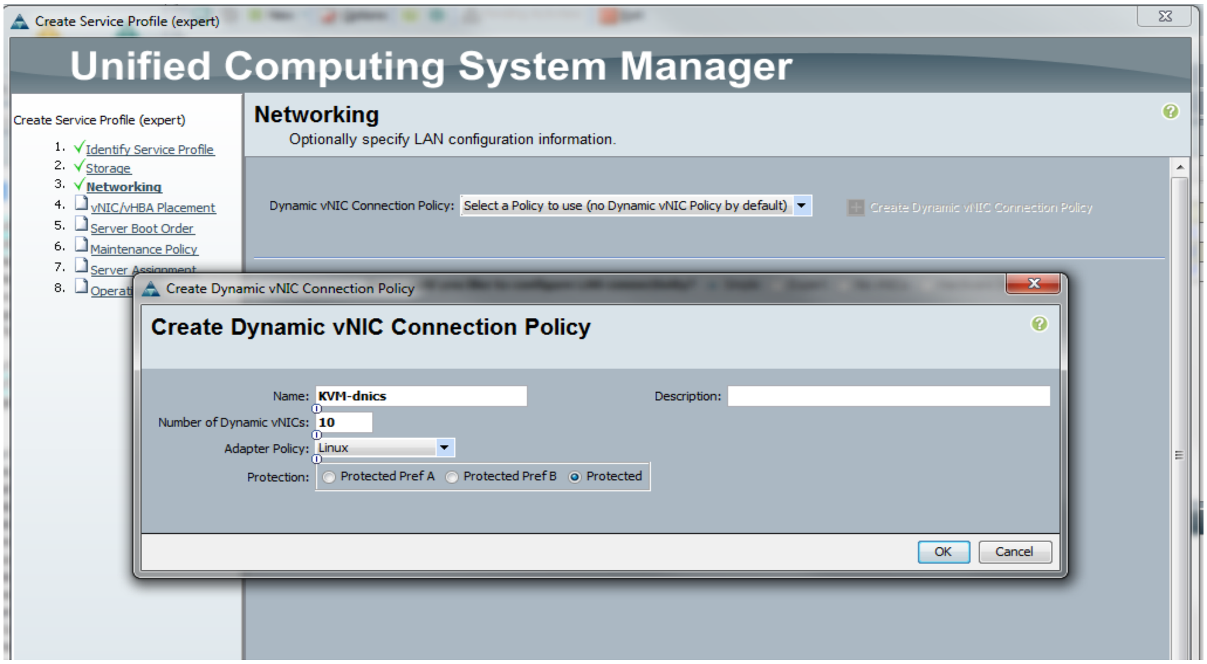
WWNN Assignment: RTP-WWNN-POD1(3/8)

Create WWNN Pool

The WWNN will be assigned from the selected pool.
The available/total WWNNs are displayed after the pool name.

Name	WWPN
vHBA vhba1	Derived
vHBA If VSAN_100	
vHBA vhba2	Derived
vHBA If VSAN_200	

Next we create the dynamic vNIC policy. Here we name the policy and create 10 dynamic vNICS.



Create Service Profile (expert)

Unified Computing System Manager

Create Service Profile (expert)

1. ✓ Identify Service Profile
2. ✓ Storage
3. ✓ **Networking**
4. vNIC/vHBA Placement
5. Server Boot Order
6. Maintenance Policy
7. Server Assignment
8. Operat

Networking

Optionally specify LAN configuration information.

Dynamic vNIC Connection Policy: Select a Policy to use (no Dynamic vNIC Policy by default)

Create Dynamic vNIC Connection Policy

Create Dynamic vNIC Connection Policy

Create Dynamic vNIC Connection Policy

Name: **KVM-dnics**

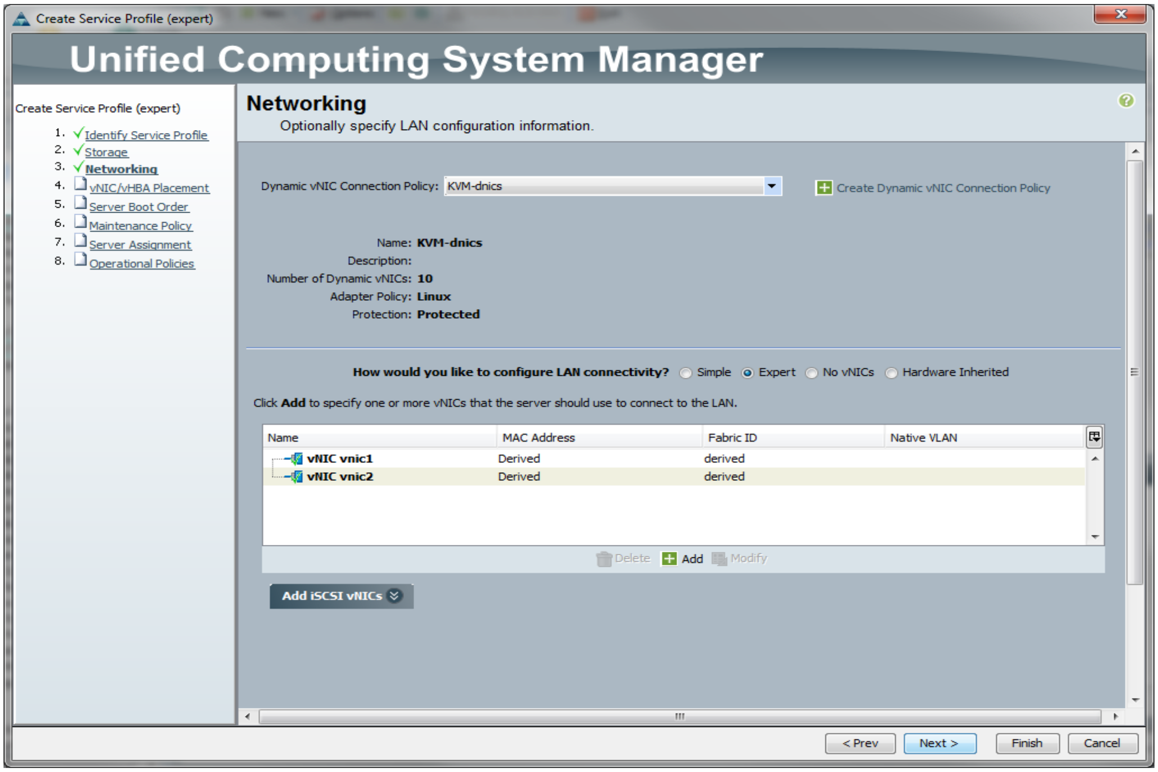
Description:

Number of Dynamic vNICs: **10**

Adapter Policy: Linux

Protection: ○ Protected Pref A ○ Protected Pref B ● Protected

Next create the two VNICs. One on fabric A and the other on Fabric B. Do NOT select Fabric Failover. Make sure to add the appropriate VLANs to each VNIC.


Create Service Profile (expert)
Unified Computing System Manager
Create Service Profile (expert)
1. ✓ Identify Service Profile
2. ✓ Storage
3. ✓ Networking
4. vNIC/vHBA Placement
5. Server Boot Order
6. Maintenance Policy
7. Server Assignment
8. Operational Policies

Networking

Optionally specify LAN configuration information.

Dynamic vNIC Connection Policy: KVM-dnics

Create Dynamic vNIC Connection Policy

Name: **KVM-dnics**
Description:
Number of Dynamic vNICs: **10**
Adapter Policy: **Linux**
Protection: **Protected**

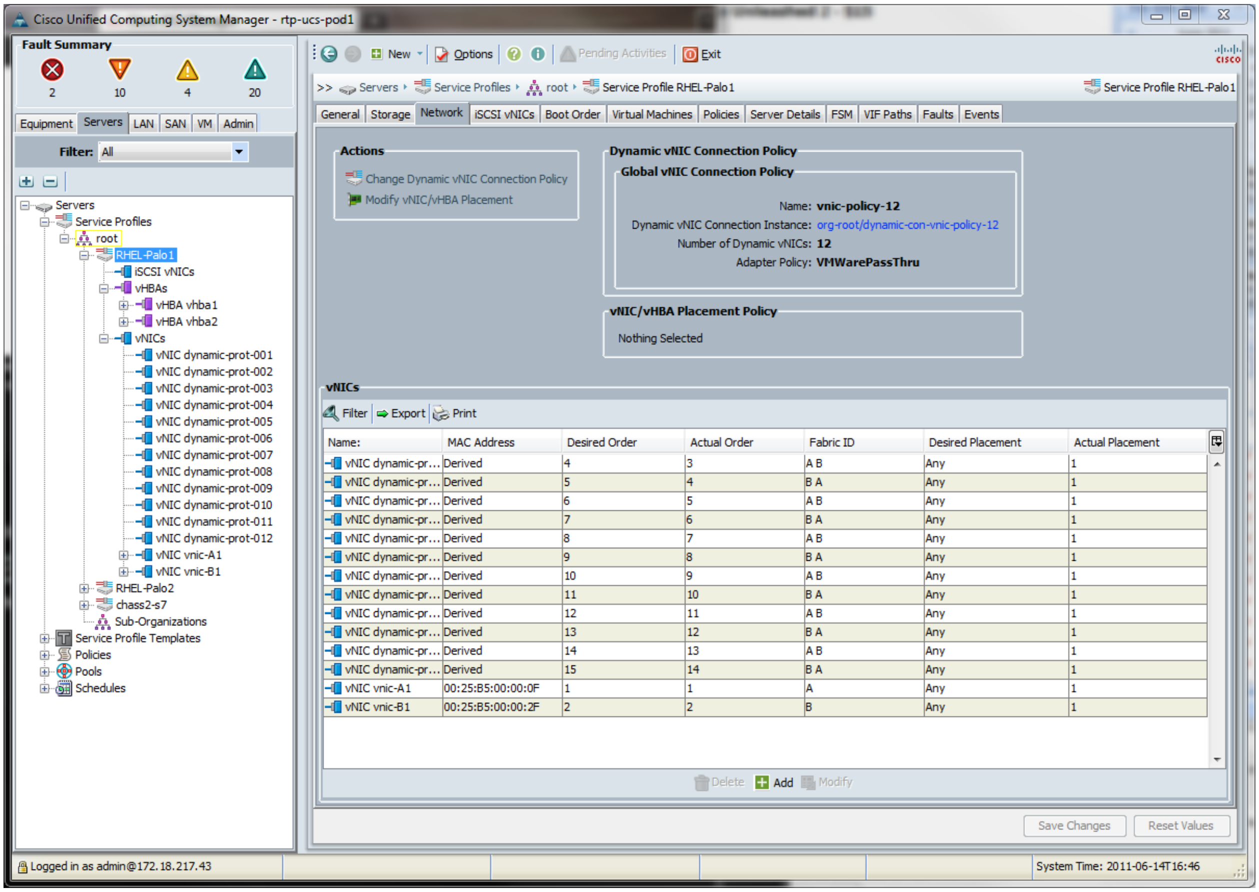
How would you like to configure LAN connectivity? ○ Simple ● Expert ○ No vNICs ○ Hardware Inherited

Click **Add** to specify one or more vNICs that the server should use to connect to the LAN.

Name	MAC Address	Fabric ID	Native VLAN
vNIC vnic1	Derived	derived	
vNIC vnic2	Derived	derived	

Finish the Service Profile creation and apply it to a blade that has a Cisco VIC/Palo card installed. Because Dynamic VNICs are defined the association will fail if you attempt to assign the Service Profile to a blade without a VIC/Palo card.

Here you can see the service profile assigned to a blade. Note how the Dynamic VNICs are load balanced across the A and B fabrics.



Dynamic vNIC Connection Policy

Global vNIC Connection Policy

Name: **vnic-policy-12**
Dynamic vNIC Connection Instance: org-root/dynamic-con-vnic-policy-12
Number of Dynamic vNICs: **12**
Adapter Policy: **VMWarePassThru**

vNIC/vHBA Placement Policy

Nothing Selected

vNICs

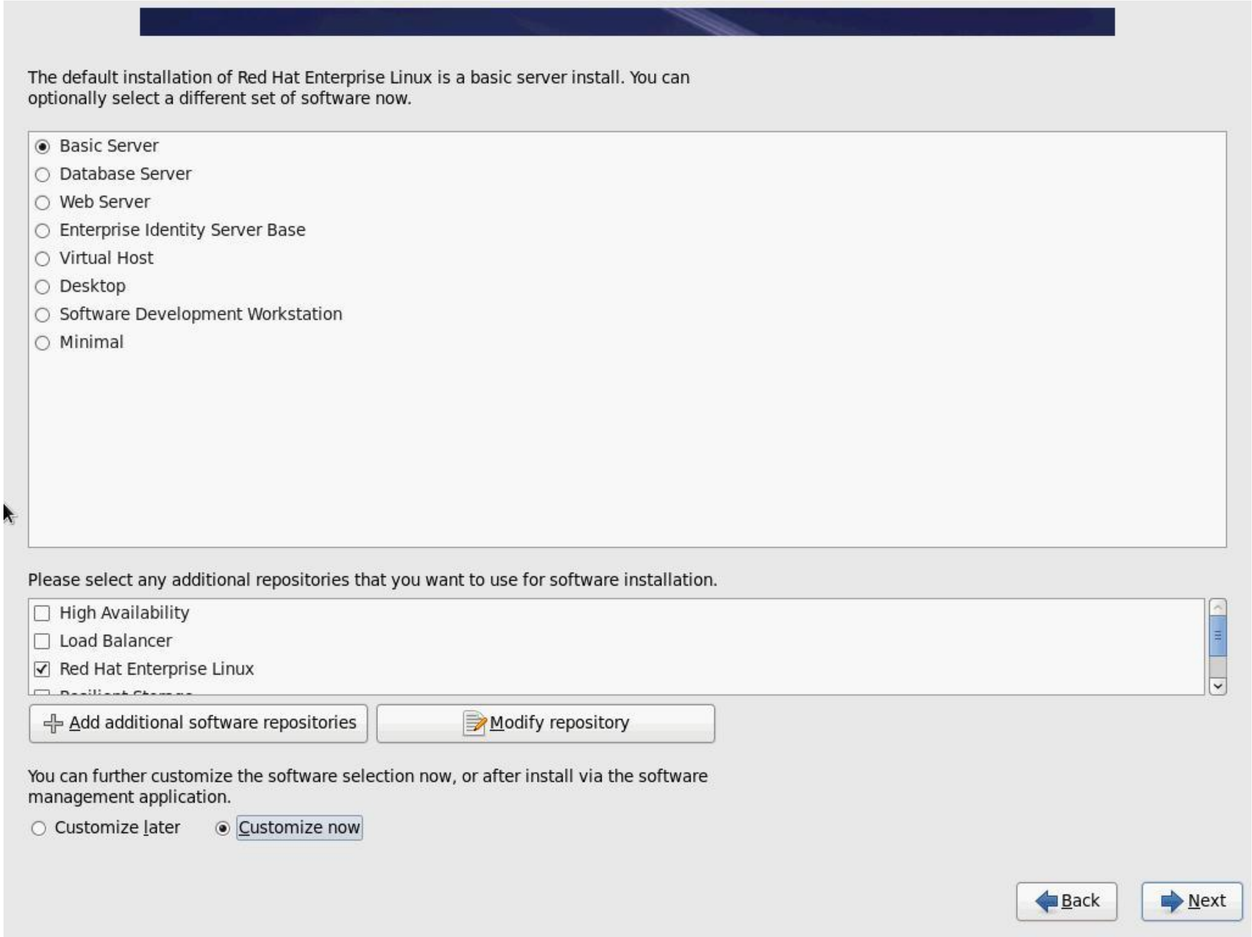
Name:	MAC Address	Desired Order	Actual Order	Fabric ID	Desired Placement	Actual Placement
vNIC dynamic-pr…	Derived	4	3	A B	Any	1
vNIC dynamic-pr…	Derived	5	4	B A	Any	1
vNIC dynamic-pr…	Derived	6	5	A B	Any	1
vNIC dynamic-pr…	Derived	7	6	B A	Any	1
vNIC dynamic-pr…	Derived	8	7	A B	Any	1
vNIC dynamic-pr…	Derived	9	8	B A	Any	1
vNIC dynamic-pr…	Derived	10	9	A B	Any	1
vNIC dynamic-pr…	Derived	11	10	B A	Any	1
vNIC dynamic-pr…	Derived	12	11	A B	Any	1
vNIC dynamic-pr…	Derived	13	12	B A	Any	1
vNIC dynamic-pr…	Derived	14	13	A B	Any	1
vNIC dynamic-pr…	Derived	15	14	B A	Any	1
vNIC vnic-A1	00:25:B5:00:00:0F	1	1	A	Any	1
vNIC vnic-B1	00:25:B5:00:00:2F	2	2	B	Any	1

3.3 Install RHEL 6.1

Now that the Service Profile is associated, load RHEL 6.1 on the blade. Make sure to install the virtualization components when you install RHEL.

This testplan will not detail the full RHEL installation but will show where to select KVM installation.

When you get to the RHEL server installation type, make sure to select "Customize now" radio button and select "Next"



The default installation of Red Hat Enterprise Linux is a basic server install. You can optionally select a different set of software now.

- ● Basic Server
- ○ Database Server
- ○ Web Server
- ○ Enterprise Identity Server Base
- ○ Virtual Host
- ○ Desktop
- ○ Software Development Workstation
- ○ Minimal

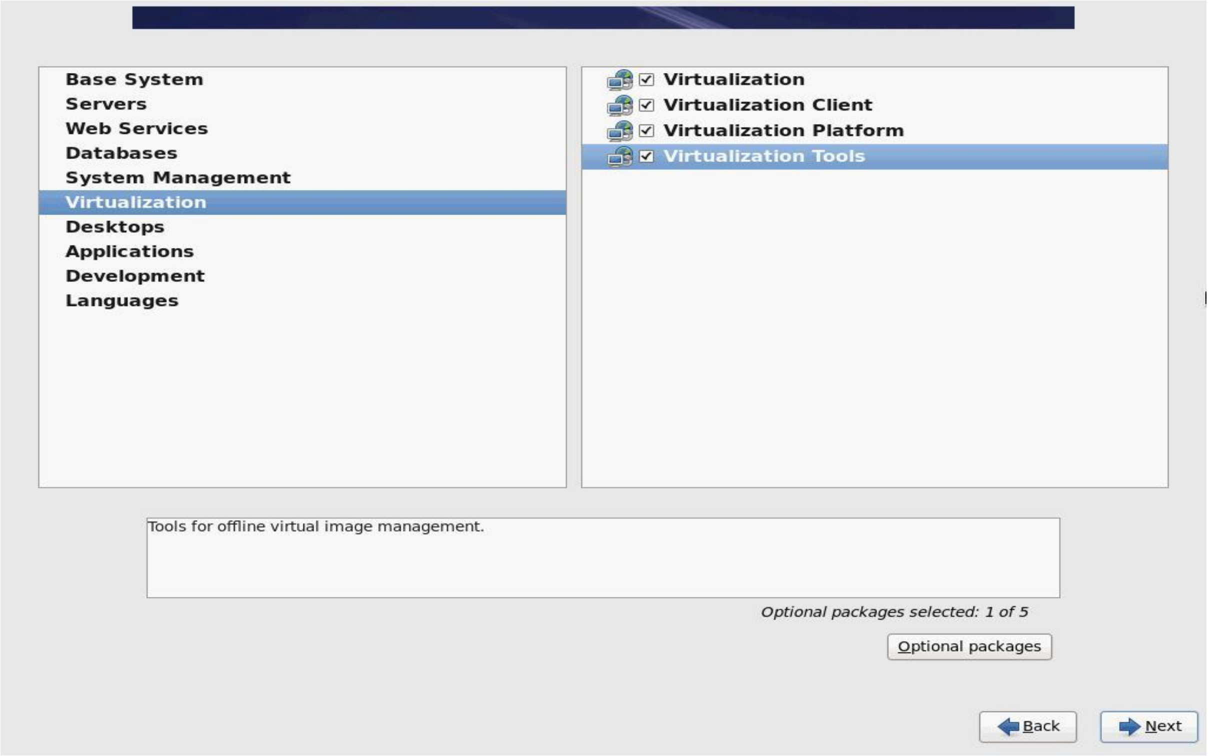
Please select any additional repositories that you want to use for software installation.

- □ High Availability
- □ Load Balancer
- ☑ Red Hat Enterprise Linux

You can further customize the software selection now, or after install via the software management application.

○ Customize later ● Customize now

Also make sure to select Virtualization and select all the virtualization services.



Tools for offline virtual image management.

Optional packages selected: 1 of 5

RHEL will start installing after you select next.

3.4 Configure KVM

Depending on your KVM configuration you may or may not need to modify qemu.conf file. In this testplan everything is done by root so we need to make a few changes to allow KVM to work with NFS shared storage.

Edit qemu.conf

```
[root@rhel-palo2 ~]# cd /etc/libvirt/
[root@rhel-palo2 libvirt]# vi qemu.conf
```

Change the following lines

```
# The default security driver is SELinux. If SELinux is disabled
# on the host, then the security driver will automatically disable
# itself. If you wish to disable QEMU SELinux security driver while
# leaving SELinux enabled for the host in general, then set this
# to 'none' instead
#
```

```
security_driver = "none"


# The user ID for QEMU processes run by the system instance
user = "root"

# The group ID for QEMU processes run by the system instance
group = "root"

# Whether libvirt should dynamically change file ownership
# to match the configured user/group above. Defaults to 1.
# Set to 0 to disable file ownership changes.
dynamic_ownership = 0
```

Now restart libvirtd with the following command

```
[root@rhel-palo2 libvirt]# service libvirtd restart
Stopping libvirtd daemon:                                  [  OK  ]
Starting libvirtd daemon:                                  [  OK  ]
```

3.5 Setup UCSM to RHEL Integration

In this step we will setup the integration between UCSM and the RHEL host. The setup between UCSM and RHEL is a little different than UCSM and VMware vCenter. Since RHEL does not have a vCenter equivalent we only need to create a port-profile and port-profile client. There is nothing else to create.

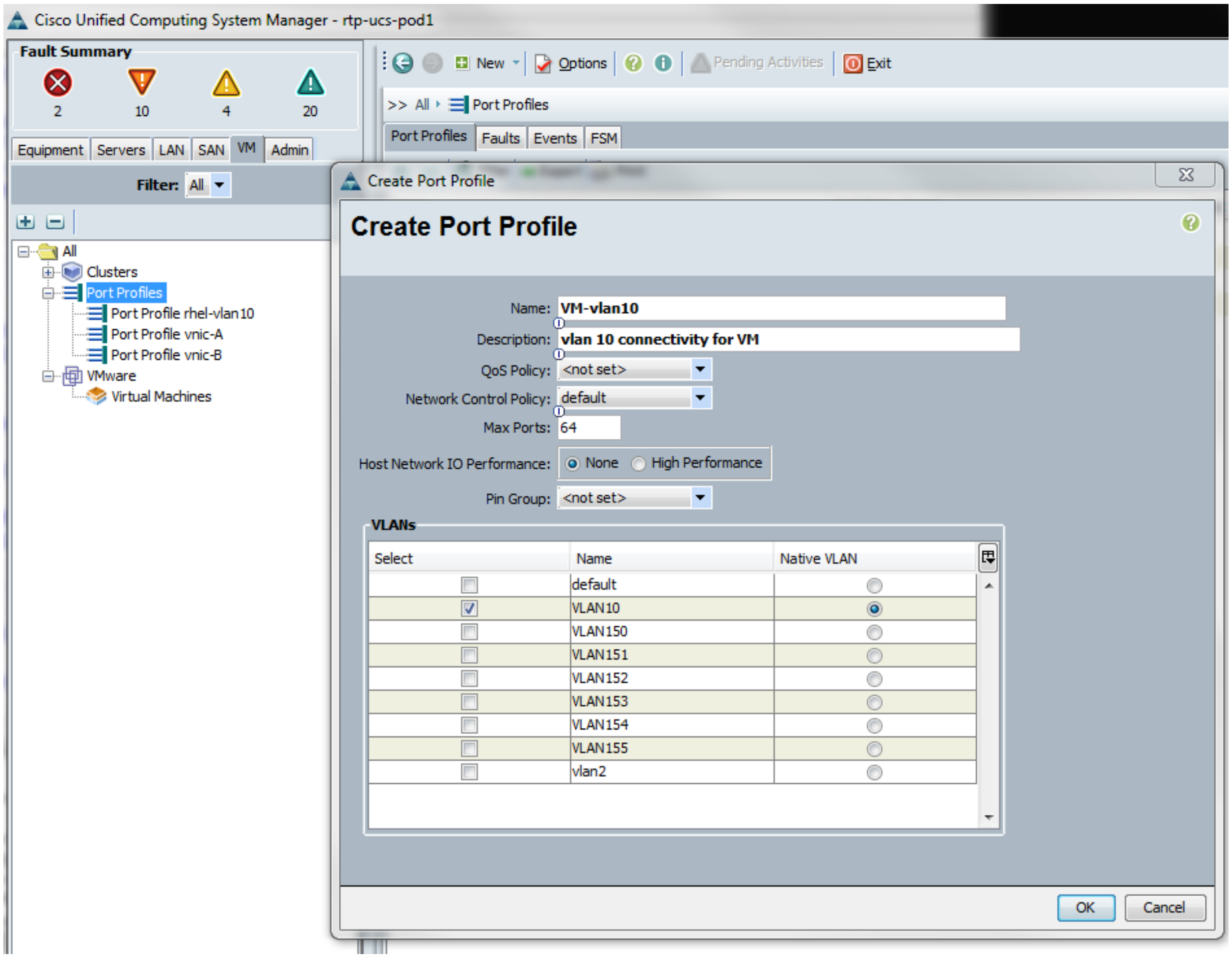
Create the port-profile in UCSM.

In UCSM browse to the VM tab, select “Port-profiles” from the tree and right click and create new port-profile.

Fill out the port-profile box to network settings you want a VM to have. In the example below we are creating an access port that will allow the VM to access vlan 10.

Note that most VMs expect a single VLAN to be presented. If you select multiple vlans you are essentially trunking to the VM.

Do not select “High Performance”. This option only works with VMware ESX 5.0



Create Port Profile

Name: VM-vlan10

Description: vlan 10 connectivity for VM

QoS Policy: <not set>

Network Control Policy: default

Max Ports: 64

Host Network IO Performance: ● None ○ High Performance

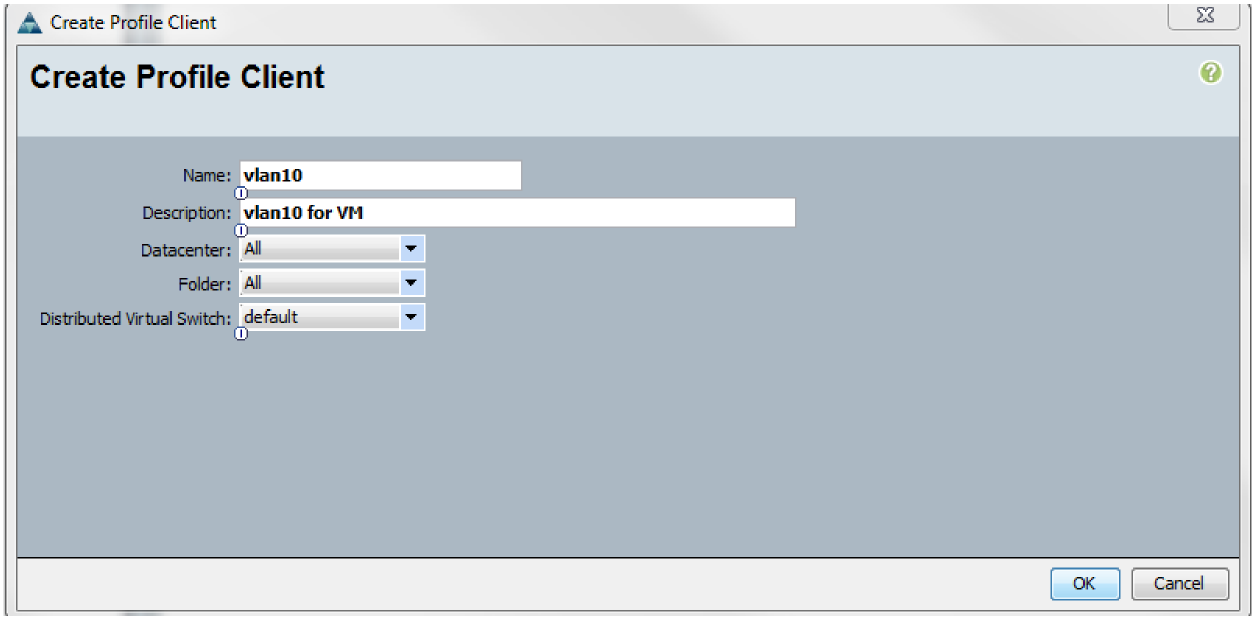
Pin Group: <not set>

VLANs

Select	Name	Native VLAN
□	default	○
☑	VLAN10	●
□	VLAN150	○
□	VLAN151	○
□	VLAN152	○
□	VLAN153	○
□	VLAN154	○
□	VLAN155	○
□	vlan2	○

Now that we have created the port-profile we need to create a port-profile client. The purpose of the client is to actually assign the port-profile to a host.

Select the port-profile, right click and select “create port-profile client”. Give the client a name and description and change the Distributed Virtual Switch to default. By setting it default you are telling UCSM that this is for KVM and not for VMware.



Create Profile Client

Name: vlan10

Description: vlan10 for VM

Datacenter: All

Folder: All

Distributed Virtual Switch: default

3.6 Verify Dynamic VNICs on Server

Linux sees dynamic VNICs just as it does regular VNICs. They show up as eth interfaces and can be viewable with “ifconfig” It can be difficult to tell which eth interfaces are dynamic and which are not. Generally Dynamic VNICs will not have MAC addresses assigned to them. So any eth interface without a MAC should be a Dynamic VNIC.

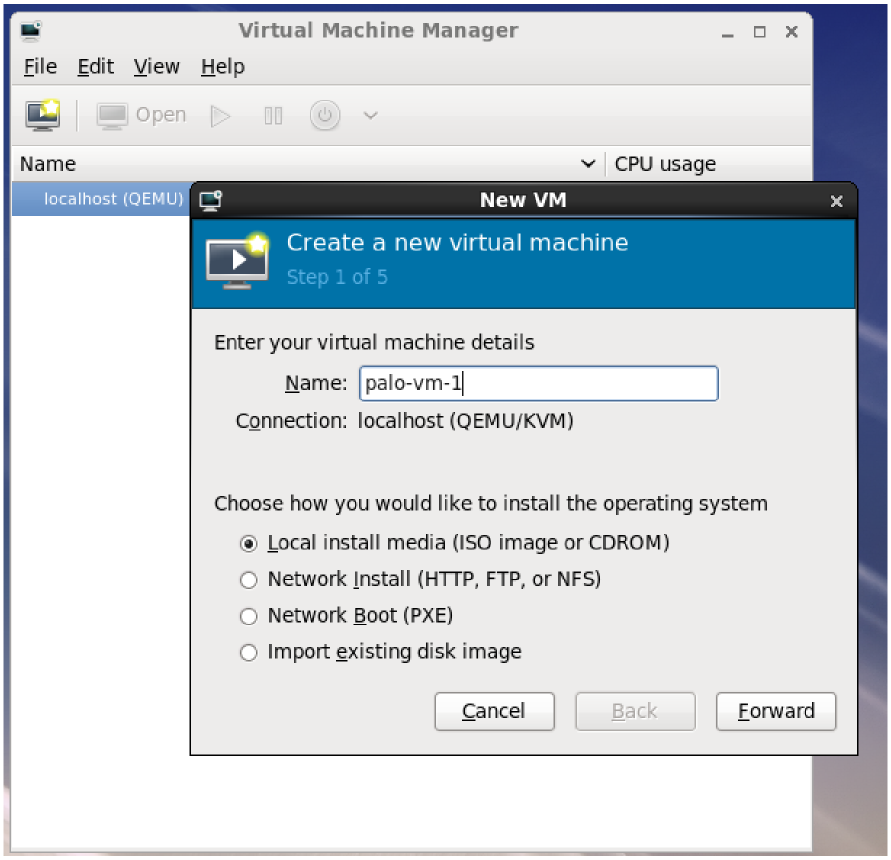
```
[root@rhel-palo2 log]# ifconfig -a | more
eth0      Link encap:Ethernet  HWaddr 00:25:B5:00:00:2E
          inet addr:172.18.217.195  Bcast:172.18.217.255
Mask:255.255.255.0
          inet6 addr: fe80::225:b5ff:fe00:2e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:7054 errors:0 dropped:0 overruns:0 frame:0
          TX packets:737 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:911661 (890.2 KiB)  TX bytes:240572 (234.9 KiB)

eth1      Link encap:Ethernet  HWaddr 00:25:B5:00:00:1F
          inet6 addr: fe80::225:b5ff:fe00:1f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:11976 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1619808 (1.5 MiB)  TX bytes:492 (492.0 b)

eth2      Link encap:Ethernet  HWaddr 00:00:00:00:00:00
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

Above you can see that eth0 and eth1 both have MAC addresses that match the MAC addresses assigned through the Service Profile. They are the real VNICs. Eth2 does not have a MAC address at all so it is a Dynamic VNIC.

3.7 Create KVM Virtual Machines

This example will use Virtual Machine Manager GUI but you can create the VM via the CLI.

Start the VMM and create a Virtual Machine.



Create a new virtual machine

Step 1 of 5

Enter your virtual machine details

Name: palo-vm-1

Connection: localhost (QEMU/KVM)

Choose how you would like to install the operating system

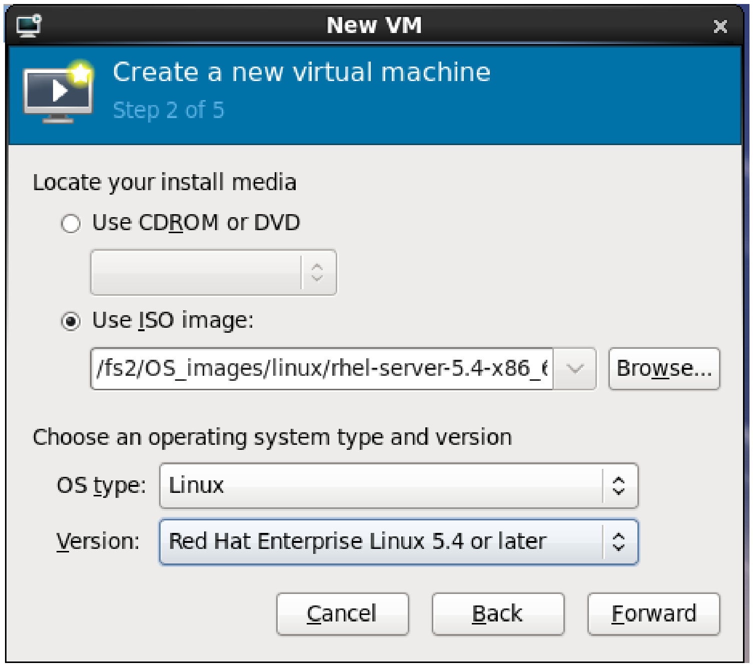
● Local install media (ISO image or CDROM)

○ Network Install (HTTP, FTP, or NFS)

○ Network Boot (PXE)

○ Import existing disk image

Choose your install media option.



Create a new virtual machine

Step 2 of 5

Locate your install media

○ Use CDROM or DVD

● Use ISO image:

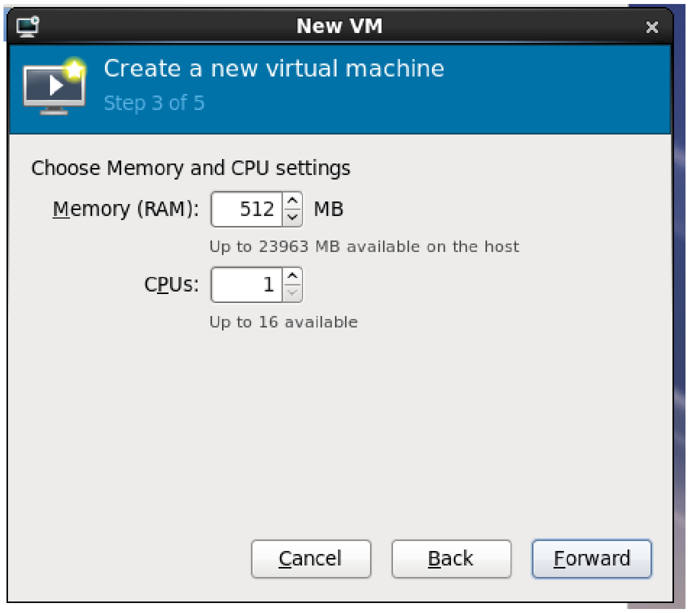
/fs2/OS_images/linux/rhel-server-5.4-x86_6

Choose an operating system type and version

OS type: Linux

Version: Red Hat Enterprise Linux 5.4 or later

Choose Memory and vCPU



Create a new virtual machine

Step 3 of 5

Choose Memory and CPU settings

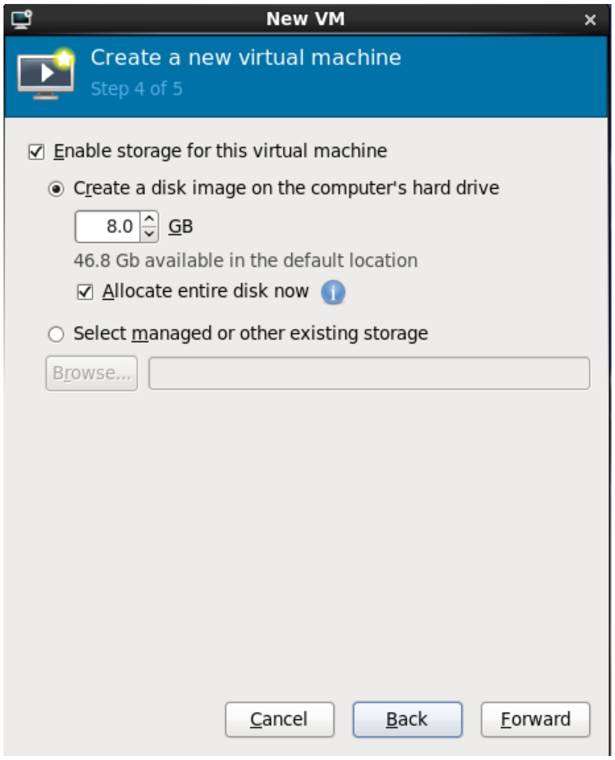
Memory (RAM): 512 MB

Up to 23963 MB available on the host

CPUs: 1

Up to 16 available

Allocate the amount of disk space to use



New VM

Create a new virtual machine

Step 4 of 5

☑ Enable storage for this virtual machine

● Create a disk image on the computer's hard drive

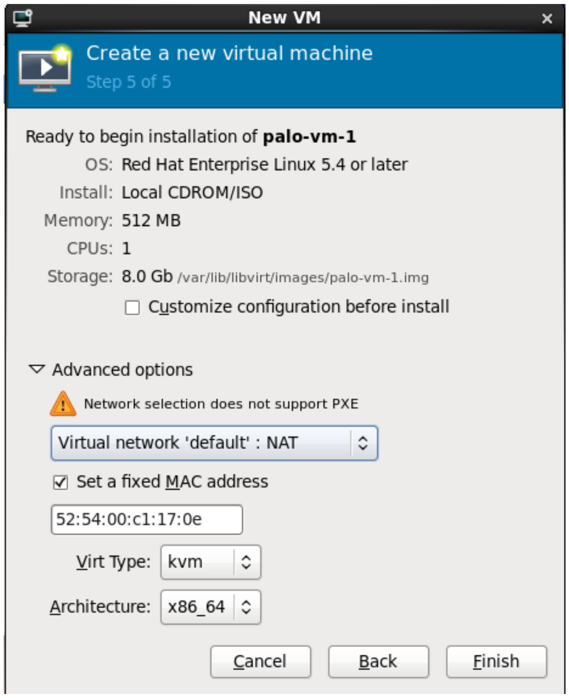
8.0 GB

46.8 Gb available in the default location

☑ Allocate entire disk now

○ Select managed or other existing storage

Verify the settings and create the Virtual Machine. Currently VMM does not have the ability to assign Dynamic VNICs. We need to first create the VM and then change the settings.



New VM

Create a new virtual machine

Step 5 of 5

Ready to begin installation of **palo-vm-1**

OS: Red Hat Enterprise Linux 5.4 or later

Install: Local CDROM/ISO

Memory: 512 MB

CPUs: 1

Storage: 8.0 Gb /var/lib/libvirt/images/palo-vm-1.img

□ Customize configuration before install

Advanced options

Network selection does not support PXE

Virtual network 'default' : NAT

☑ Set a fixed MAC address

52:54:00:c1:17:0e

Virt Type: kvm

Architecture: x86_64

3.8 Migrate the VM from NAT to Palo Dynamic VNIC

Once the OS is installed on the VM we migrate the VM from NAT based networking to an actual Dynamic VNIC hosted on the Palo card. Note that to do this we must modify the XML file that describes the VM we created. Currently this is the only way to move the VM to a Palo based Dynamic VNIC. You need to be comfortable with Linux CLI and editors.

Shutdown the VM.

Run the following command

```
[root@rhel-palo2 ~]# virsh list --all
 Id Name                 State
----------------------------------
  - palo-vm-1            shut off
```

Now run virsh edit "VM name" of the VM you want to change

```
[root@rhel-palo2 ~]# virsh edit palo-vm-1
```

Move to the network section. You will have some lines that look like

```
    <interface type='network'>
      <mac address='52:54:00:c1:17:0e'/>
      <source network='default'/>
      <model type='virtio'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'/>
    </interface>
```

We will edit this section to add

1. A MAC address – Currently you create the MAC. Use a common MAC scheme.
2. Port-profile – This should match the port-profile you created in UCSM
3. One of the dynamic eth interfaces – use one that has not been assigned. We will use eth8

Change the networking section to look like the following

```
   <interface type='direct'>
     <mac address='00:16:36:84:14:ed'/>
     <source dev='eth8' mode='private'/>
     <virtualport type='802.1Qbh'>
       <parameters profileid='VM-vlan10'/>
     </virtualport>
     <model type='virtio'/>
     <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'/>
   </interface>
```

Save the file and exit. If you get an error on exit go back and reedit the file and make sure there are no mistakes

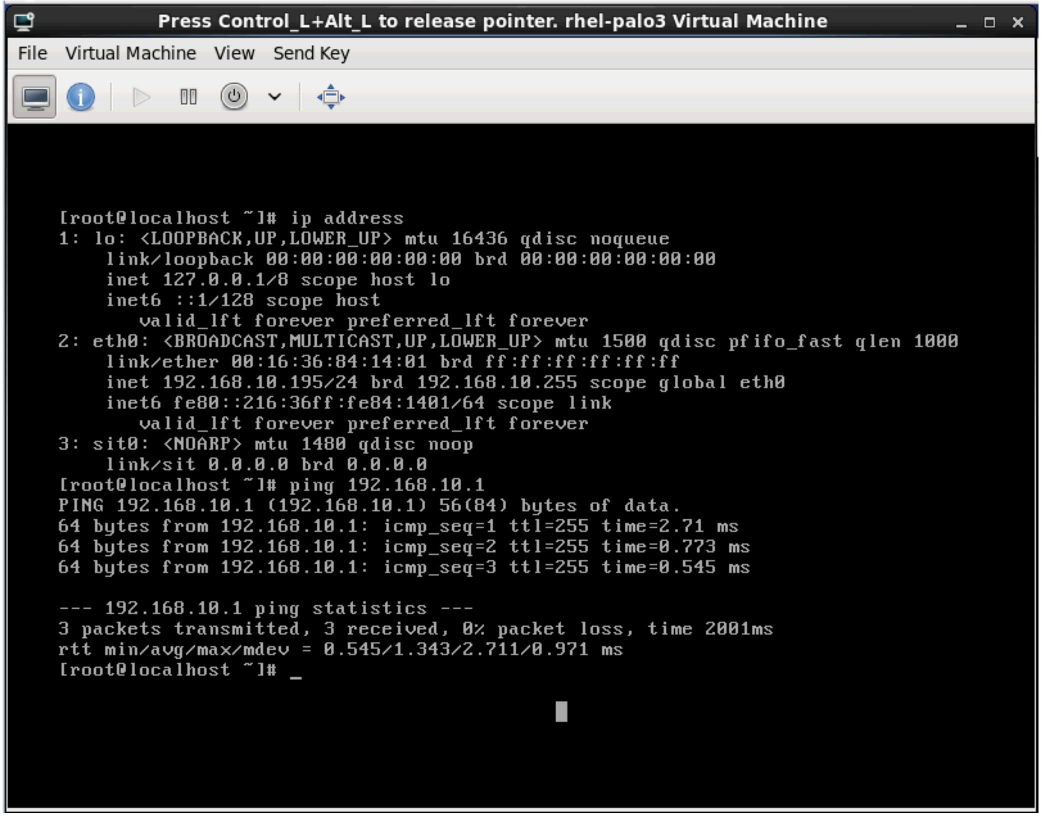
3.9 Start the VM and verify networking

Start the VM and verify that networking works.
You can start the VM with virsh

```
[root@rtp-ucs-pod1-196 ~]# virsh start palo-vm-1
Domain palo-1 started

[root@rtp-ucs-pod1-196 ~]# virsh list --all
 Id Name                      State
----------------------------------
  1 rhel-palo2                running
  2 rhel-palo3                running
  4 palo-vm-1                    running
```

You now need to get on the console of the VM using VMM or issuing the command virt-viewer palo-vm-1.

Once on the console login and verify that you have network connectivity.



Press Control_L+Alt_L to release pointer. rhel-palo3 Virtual Machine
File
Virtual Machine
View
Send Key

```
[root@localhost ~]# ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:16:36:84:14:01 brd ff:ff:ff:ff:ff:ff
    inet 192.168.10.195/24 brd 192.168.10.255 scope global eth0
    inet6 fe80::216:36ff:fe84:1401/64 scope link
       valid_lft forever preferred_lft forever
3: sit0: <NOARP> mtu 1480 qdisc noop
    link/sit 0.0.0.0 brd 0.0.0.0
[root@localhost ~]# ping 192.168.10.1
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.
64 bytes from 192.168.10.1: icmp_seq=1 ttl=255 time=2.71 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=255 time=0.773 ms
64 bytes from 192.168.10.1: icmp_seq=3 ttl=255 time=0.545 ms

--- 192.168.10.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.545/1.343/2.711/0.971 ms
[root@localhost ~]# _
```

Verify the eth interface you used for the VM now is up and has the MAC address you assigned.

```
[root@rtp-ucs-pod1-196 ~]# ifconfig eth8
eth8      Link encap:Ethernet  HWaddr 00:16:36:84:14:ed
          inet6 addr: fe80::6cfe:32ff:fee6:d45e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1569 errors:0 dropped:0 overruns:0 frame:0
          TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:143255 (139.8 KiB)  TX bytes:1368 (1.3 KiB)
```

Use ip link to verify

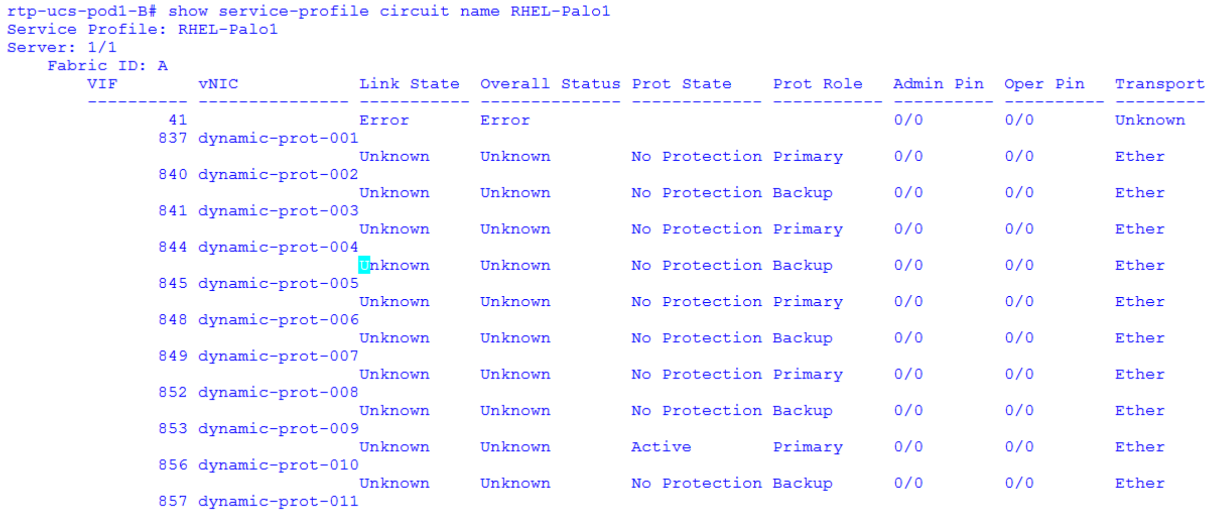
```
 [root@rtp-ucs-pod1-196 ~]# ip link show eth8
10: eth8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:16:36:84:14:ed brd ff:ff:ff:ff:ff:ffip link to verify as well
```

Note that the link state is up

3.10 Troubleshooting commands

From the CLI of UCSM

Use show service-profile circuit name to see which Dynamic VNICs are up and running from a UCSM standpoint. Look for VIFS that show as UP



```
rtp-ucs-pod1-B# show service-profile circuit name RHEL-Palo1
Service Profile: RHEL-Palo1
Server: 1/1
    Fabric ID: A
        VIF        vNIC            Link State  Overall Status Prot State    Prot Role   Admin Pin  Oper Pin   Transport
        ---------- --------------- ----------- -------------- ------------- ----------- ---------- ---------- ---------
                41                 Error       Error                                    0/0        0/0        Unknown
               837 dynamic-prot-001
                                   Unknown     Unknown        No Protection Primary     0/0        0/0        Ether
               840 dynamic-prot-002
                                   Unknown     Unknown        No Protection Backup      0/0        0/0        Ether
               841 dynamic-prot-003
                                   Unknown     Unknown        No Protection Primary     0/0        0/0        Ether
               844 dynamic-prot-004
                                   Unknown     Unknown        No Protection Backup      0/0        0/0        Ether
               845 dynamic-prot-005
                                   Unknown     Unknown        No Protection Primary     0/0        0/0        Ether
               848 dynamic-prot-006
                                   Unknown     Unknown        No Protection Backup      0/0        0/0        Ether
               849 dynamic-prot-007
                                   Unknown     Unknown        No Protection Primary     0/0        0/0        Ether
               852 dynamic-prot-008
                                   Unknown     Unknown        No Protection Backup      0/0        0/0        Ether
               853 dynamic-prot-009
                                   Unknown     Unknown        Active        Primary     0/0        0/0        Ether
               856 dynamic-prot-010
                                   Unknown     Unknown        No Protection Backup      0/0        0/0        Ether
               857 dynamic-prot-011
```

Verify the port-profile config from NXOS

```
rtp-ucs-pod1-B(nxos)# show running-config port-profile

!Command: show running-config port-profile
!Time: Tue Jun 21 19:11:29 2011

version 5.0(3)N2(2.1)
port-profile type vethernet VM-vlan10
  switchport trunk native vlan 10
  switchport trunk allowed vlan 10
  switchport mode trunk
  pinning server pinning-failure link-down
  vmware port-group
  max-ports 64
  description vlan 10 connectivity for VM
  state enabled
port-profile type vethernet rhel-vlan10
  switchport trunk native vlan 10
  switchport trunk allowed vlan 10
  switchport mode trunk
  pinning server pinning-failure link-down
  vmware port-group
  max-ports 64
  state enabled
port-profile type vethernet vnic-A
  switchport trunk native vlan 2
  switchport trunk allowed vlan 1-2, 10, 150-155
  switchport mode trunk
  pinning server pinning-failure link-down
  vmware port-group
  max-ports 64
  state enabled
port-profile type vethernet vnic-B
  switchport trunk native vlan 1
  switchport trunk allowed vlan 1-2, 10, 150-155
  switchport mode trunk
  pinning server pinning-failure link-down
  vmware port-group
  max-ports 64
  state enabled
```

Use show system on NXOS

```
rtp-ucs-pod1-B(nxos)# show system internal vim info int Veth32770
interface Veth32770
  iod: 53
  if_index: 1c080010
  fsm_state: VIM_VETH_FSM_ST_CREATED
  num_attaches: 1
  detach_reason: never detached
  vim_down_reason: initializing
  curr profile: rhel-vlan10
```

```
flags: veth=00000180, veth rt_info=0008
desc:
pinning info:
  ds_id:  (12)
  ds_port_uuid: 00:16:36:84:14:01 (3)
  vNIC sam fixed: FALSE
  vNIC ID: 0
  hpp: 0
  connectee_mac_addr: 00:16:36:84:14:01
  peer sync done : no
attach info:
  src: unknown
  status: active
  fsm_state: VIM_NIV_VIF_FSM_ST_UP_ACTIVE
  pp_alias: rhel-vlan10 (8)
  ds_id:  (12)
  ds_port_uuid: 00:16:36:84:14:01 (3)
  conn_name:
  conn_dev:
  conn_uuid: ffff880660452ae1 (11)
  host_uuid: ffff880660452af1 (12)
  conn_mac_addr: 00:16:36:84:14:01
```