

# CISCO NETWORK SERVICES MANAGER 5.0.2

## TECHNICAL REFERENCE



January 8, 2013

### Contents

<b>Installation .....</b>	<b>4</b>
Overview .....	4
Prerequisites .....	5
Supported Equipment .....	7
Systems .....	7
Software .....	7
Preparation .....	7
Configure Management Access to Devices.....	8
Configure Data Path Connections Throughout the Stack.....	8
Installing the Components .....	9
Deploying and Configuring the Engine and Controller .....	9
Base Configuration.....	13
Preserving Static Routes on Devices .....	13
ASR 1004 .....	14
Nexus 7000 .....	14
DSN VSS.....	17
Nexus 5000 .....	19
UCS.....	20
Nexus 1000V .....	20
Topology and Business Model Configuration .....	21
Visualizing the Stack as a Topology Model .....	22
Topology Variations .....	24
Edit the Topology Model.....	26
Create the POD with this Request Body .....	26

Verify the Created POD.....	43
Verifying the Configuration.....	43
Check the Alerts View for the POD .....	43
Check the Controller Log.....	43
Advanced Installation Topics .....	44
Enabling the "shell" Password-Protected Access .....	44
Adding or Removing an L2 Aggregation Layer in the Stack .....	44
Substituting a Like Device for an Officially Supported Device .....	45
Device Command Modification for Specific Device Variants .....	46
Tenant Creation Dry Run.....	52
Setup .....	52
Tenant Create .....	52
Tenant Network Container (TNC) Creation .....	53
External Network Creation .....	55
Internet Edge Zone Creation.....	55
External Network Connection.....	56
Secured Internet Edge Zone .....	56
VLAN in Secured Internet Edge Zone.....	56
Verification.....	57
Troubleshooting.....	66
Syslog Aggregation.....	66
Controller Setup for Remote Syslog Logging .....	66
Engine Setup for Remote Syslog Logging.....	68
Controller Not Connecting to the Server .....	69
Gathering Logs and Technical Support information for Cisco TAC.....	69
Failure to Connect to the Devices (auth/SNMP etc.).....	74
Controller Fails to Identify a Device (SNMP READ Access Misconfiguration).....	75
Controller Fails to Log into a Device (CLI Credentials).....	77
Traffic Not Flowing Through Data Path .....	79
Management Network.....	79
Engine .....	80
Controller .....	81

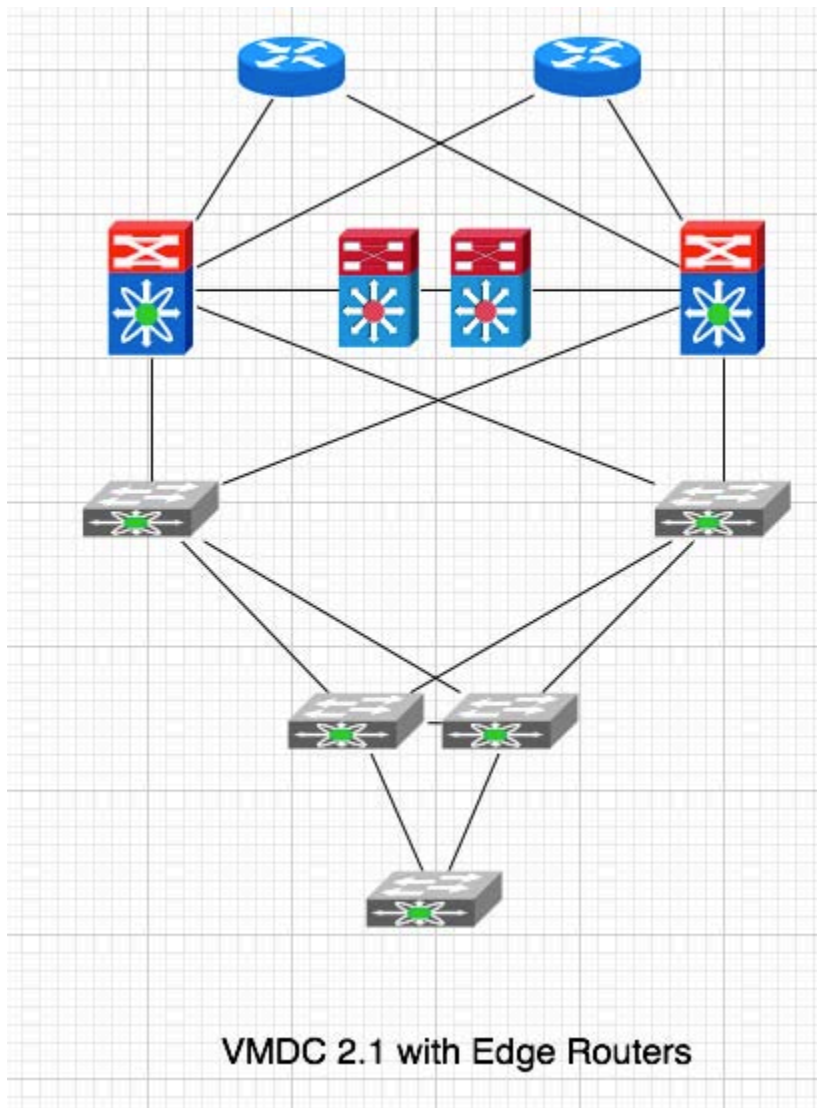
<b>API .....</b>	<b>81</b>
Task Information .....	81
<b>Release Notes/Known Issues.....</b>	<b>83</b>
IP Validation of Service Filters.....	83
Service Policies Between TNCs .....	83
Active Topology.....	83
POD Enablement.....	83
Backup/Restore.....	83
Redundancy of the Engine and Controller.....	84
<b>Obtaining Documentation, Obtaining Support, and Security Guidelines.....</b>	<b>85</b>

# Installation

After following the steps listed in this guide, the user will be able to explore the functionality that Cisco Network Services Manager (Network Services Manager) provides. The installation involves setting up the necessary hardware and software followed by configuring management access to every device in the lab.

## Overview

The following diagram depicts the network deployment architecture.



The first steps in deploying Network Services Manager are to wire and configure management access to all of the devices and then to wire and configure the data path (uplinks and downlinks).

The software is initialized by importing a model that has a standard metamodel which offers up the services detailed in the [Cisco Network Services Manager 5.0.2 NB API Specification and](#)

[Reference Guide](#). These include some predefined POD (or device stack) examples. These are the supported variations in physical architecture for Network Services Manager 5.0.2 but could be modified by Cisco Advanced Services to accommodate for other common variations. Any modifications to these default POD examples should be thoroughly tested in a sandbox environment before being deployed in a live system. See below for these variations (Topology Variations).

The Admin user will use a web client for common administrative functions. This web client can also be used to instantiate metamodels.

## Prerequisites

To deploy a Cisco Network Services Manager environment you need to have a system administrator level knowledge of: networking, VMware, server, and PC technologies.

A basic deployment consists of two Linux servers, which can each be deployed as an OVA.

These servers are usually deployed on a separate vCenter cluster (or single ESXi server) reserved for management of the equipment. For standard deployments where a single stack of equipment is expected, the guest VM requirements are:

- Engine 2 GB of RAM, 40 GB HDD and dual CPU. The base requirements for the Engine may change when managing multiple sites or multiple stacks of equipment.
- Controller 1 GB of RAM, 40 GB HDD and a dual CPU. These requirements will not change in the foreseeable future.

The devices are those required for a Cisco Virtualized Multi-Services Data Center (VMDC) 2.1 Compact POD, with a Cisco ASR 1000 (ASR 1000) pair above the Cisco Nexus 7000 (Nexus 7000) devices for connectivity to the Internet or a corporate network.

Management interfaces and data path configuration are the most common sources of delays in a deployment. The management interfaces and associated VLAN must be configured and tested for device access before starting the deployment. The data path for tenant data needs to be deployed and verified as part of the static configuration of the equipment stack (for example, VPC, trunks, port channels, etc.).

The deployment requires CLI and SNMP credentials for all of the equipment that will be managed. If the deployment matches one of the physical topology sample models, you can enter the credentials and interconnect information either via the WebUI or by preparing a REST request body for the POD as the physical topology is described and using that to create a new POD in Network Services Manager.

**Table 1. Networking Hardware for Tenant Data Path (live lab only)**

**Note** You can use the following tables to capture relevant information for your site.

Device	SNMP Version	SNMP V3 Username	SNMP Password or V2 Community	CLI Username	CLI Password
Edge Router 1 (ASR1000-1)					
Edge Router 2 (ASR1000-2)					
L3 Aggregation Switch 1 (N7000-1)					
L3 Aggregation Switch 2 (N7000-2)					
L3 Services Chassis (DSN-VSS)					
L2 Access Switch 1 (N5000-1)					
L2 Access Switch 2 (N5000-2)					
UCSM (UCS VIP)					
Virtual Access Switch 1 (N1000-1)					
Virtual Access Switch 2 (N1000-2)					
Virtual Access Switch 3 (N1000-3)					

**Table 2. Interconnects**

Interconnects	Upstream Port	Downstream Port	Interconnect Type
Edge Router 1 to L3 Aggregation Switch 1			
Edge Router 1 to L3 Aggregation Switch 2			
Edge Router 2 to L3 Aggregation Switch 1			
Edge Router 2 to L3 Aggregation Switch 2			
L3 Aggregation Switch 1 to L3 Services Chassis			
L3 Aggregation Switch 2 to L3 Services Chassis			
L3 Aggregation Switch 1 to L2 Access Switch 1			
L3 Aggregation Switch 1 to L2 Access Switch 2			
L3 Aggregation Switch 2 to L2 Aggregation Switch 1			
L3 Aggregation Switch 2 to L2 Aggregation Switch 2			

Interconnects	Upstream Port	Downstream Port	Interconnect Type
L2 Aggregation Switch 1 to UCSM			
L2 Aggregation Switch 2 to UCSM			
USCM to Virtual Access Switch 1			
USCM to Virtual Access Switch 2			
USCM to Virtual Access Switch 3			

## Supported Equipment

The devices that have been verified with Network Services Manager are available on [cisco.com](http://www.cisco.com/en/US/products/ps11636/products_device_support_tables_list.html) at [http://www.cisco.com/en/US/products/ps11636/products\\_device\\_support\\_tables\\_list.html](http://www.cisco.com/en/US/products/ps11636/products_device_support_tables_list.html).

In addition to this list, a device might be supported in a particular role if the CLI commands have parity with one of the supported devices listed below. For example, configuring MPLS, VRF, OSPF and sub-interfaces uses the same CLI directives on a Cisco 6500 device that it does on an ASR 1000 device. A simple field configuration change will allow device commands to be substituted from a supported device library in such a case. This is only recommended after thorough testing to verify tenant automation is not compromised by such a substitution.

## Systems

**Note** The OVAs will prompt for setup information upon first boot (except the WebUI default username and password).

The WebUI default username is *admin* and password is also set to *admin*.

Use the following table to record your passwords.

**Table 3. System Passwords**

System	Username	Default Password
Engine OVA console	admin	(none)
Controller OVA console	admin	(none)
WebUI	admin	admin
Northbound system	apiclient	overdrive

## Software

- Engine
- Controller

## Preparation

- Set aside equipment for the management hosts (ESXi).
- Set aside equipment for the device stack.

## Configure Management Access to Devices

Before Network Services Manager can start managing a device, the following basic configuration must be in place on the device:

- Admin user and password
- SSH or Telnet server enabled
- SNMP with read/write enabled—Only v2 and v3 are supported
- Management IP and default route configured

## Configure Data Path Connections Throughout the Stack

Data path connections (or interconnects) must be preconfigured on the switches as well. Network Services Manager 5.0.2 supports two types of interconnects:

- L3Routed—Used between routers and/or L3 capable switches.
- L2Trunk—Used between L2 switches capable of VLAN trunking.

Identify the management IP addresses for the various components listed in the following table:

**Table 4. Management IP Addresses**

Device	IP Address	Fully Qualified Domain Name
Engine		
Controller		
<b>Device Stack</b>		
Edge Router 1		
Edge Router 2		
L3 Aggregation Switch 1		
L3 Aggregation Switch 2		
L3 Services Chassis		
L2 Access Switch 1		
L2 Access Switch 2		
UCS 6xxx VIP		
Virtual Access Switch 1		
Virtual Access Switch 2		
Virtual Access Switch 3		



## Installing the Components

### Deploying and Configuring the Engine and Controller

Two methods can be used for deploying Network Services Manager components:

- Deploy OVF template
- Create a custom VM and mount the ISO file on the VM CD

- Note**
- Having both the controller and engine synchronized with a time server will greatly improve ease of troubleshooting when log files need to be examined. This is particularly true during initial configuration.
  - We recommend that you write down the administrative password. This password cannot be retrieved or reset without help from Cisco TAC.

### Before You Begin

Before you install the components, collect the information described in [Table 5](#) so that you can provide the information when prompted.

**Table 5. Parameter Prompts**

Prompt	Example Entry	Comment
Enter hostname:	eset-ncssam-1	Enter the hostname that will be registered to the DNS.
Enter IP address:	10.1.2.10	Enter the IP address for the engine or controller virtual appliance.
Enter IP netmask:	255.255.255.0	Enter the IP subnet mask.
Enter default gateway:	10.1.2.1	Enter the default gateway for the subnet.
Enter DNS Domain name:	cisco.com	Enter the domain name.
Enter primary nameserver:	10.1.2.2	Enter the main/primary name server (DNS).
Add/Edit another nameserver?	y	If possible, we recommend adding a secondary name server.
Enter secondary nameserver:	10.1.2.3	Enter the secondary name server (DNS).
Add/Edit another nameserver?	n	Enter <b>n</b> unless you need to add a third name server.
Enter primary NTP server:	10.1.2.5	Enter the main/primary NTP server IP address.
Add/Edit secondary NTP server?	y	If possible, we recommend adding a secondary NTP server.
Enter secondary NTP server:	10.1.2.6	Enter the secondary NTP server IP address.
Enter system timezone [UTC]:	PST8PDT	Enter the server timezone.
Enter password for admin user:	*****	Enter a strong 5-character (or more) password for admin-related tasks.
Confirm password:	*****	Reenter the admin user password.

## Creating a Controller Instance Using the configure Script

**Prerequisite** Enable the "shell" password-protected access by using the **shell\_enable** command as follows. This command can be set only via the console.

```
od-11-dsc/admin# shell_enable
Enter a password for shell access :
Confirm the password again :
Info: Shell access password was set successfully. Use shell command to access
shell
od-11-dsc/admin#
```

After the controller OVA is deployed, a controller instance must be created using the **configure** script. Note the controller name and controller password that you supplied when running the "configure" script. These items must match the information that is supplied when creating the POD on the engine.

**Table 6. Engine POD Parameters and Controller configure Script Parameters**

Engine POD Parameter	Controller configure Script Parameter
pod.controller.name	"Controller name"
pod.controller.password	"Controller password"

An example using the **configure** script follows:

```
[root@od-11-dsc ~]# /usr/local/overdrive/controller/bin/configure -f
Network Services Manager Controller configure script
Option '-f' specified, proceeding with force of new controller instance
configuration...
Controller name? [vmdc-controller] testone
Controller password? [password]
Re-enter controller password: [password]
Engine hostname or IP address ? 1.1.1.1
Syslog host? 192.168.66.13

-----
You entered:
-----
Controller name:      testone
Controller password: password
Engine hostname:     1.1.1.1
Syslog host:         192.168.66.13

(Configuration instance name: controller1)
(Configuration instance directory: /etc/overdrive/controller1)

Press Enter to continue, or Ctrl-C to quit
Creating controller config directory: /etc/overdrive/controller1
Creating controller custom device directory: /etc/overdrive/controller1
mkdir: created directory `/etc/overdrive/controller1/custom'
mkdir: created directory `/etc/overdrive/controller1/custom/cisco'
```

```

Creating controller persistence directory:
/usr/local/overdrive/controller/data/controller1
Remember to upgrade contents of demo cert /etc/overdrive/certs.p12 prior to
production use.

Created:
/etc/overdrive/controller1:
total 36
-rw-rw-rw- 1 root root 4295 Aug  8 13:09 agent.properties
-rw-r----- 1 root root 6360 Aug  8 13:09 boilerplates.xml
drwxrwxrwx 3 root root 4096 Aug  8 13:09 custom
-rw-rw-rw- 1 root root 2412 Aug  8 13:09 log4j.properties
-rw-rw-rw- 1 root root 1562 Aug  8 13:09 Overdrive.properties
-rw-r----- 1 root root 1648 Aug  8 13:09 ssl.properties
-rw-r----- 1 root root  363 Aug  8 13:09 staticroutes.router

/etc/overdrive/controller1/custom:
total 4
drwxrwxrwx 2 root root 4096 Aug  8 13:09 cisco

/etc/overdrive/controller1/custom/cisco:
total 0

/usr/local/overdrive/controller/data/controller1:
total 4
-rw-r----- 1 root root 3214 Aug  8 13:09 services.xml
[root@od-l1-dsc ~]#

```

## Updating Network Services Manager Certificates

**Prerequisite** Enable the "shell" password-protected access by using the **shell\_enable** command as follows. This command can be set only via the console.

```

od-l1-dsc/admin# shell_enable
Enter a password for shell access :
Confirm the password again :
Info: Shell access password was set successfully. Use shell command to access
shell
od-l1-dsc/admin#

```

Network Services Manager digital certificates are used for the web page SSL encryption and for encrypting the JMS messages between controllers and the server. The digital certificates also provide an authenticating mechanism for controllers.

The certificates are deployed as a p12 file (or pfx if you are using a Windows-based CA) and need to be signed by a trusted CA. The demo certificates that are provided as part of the Network Services Manager deployment are issued by a demo CA in Cisco's labs.

The Network Services Manager engine and controller are deployed with certificates; Network Services Manager 5.0.2 does not support using other certificates.

The first step is to prepare a p12 with the following:

- For the engine, a "server" certificate issued to the FQDN of the Network Services Manager engine. This is to avoid certificate errors when visiting the Network Services Manager WebUI via HTTPS. This certificate, along with its private key, should be packaged with the public key of the issuing CA (install all certificates in the certification path) into a password protected p12.
- For each controller, a client certificate issued to the hostname of the controller (the certificate name is not as important since there is no HTTPS page on the controller).

When you create the certificates, you need to provide a friendly name for the certificate when prompted to do so by your CA.

### To update a certificate:

1. Upload a new cert to the engine.
  - a. SSH onto the engine and enter **shell** at the command prompt to enter shell mode.
  - b. Use SCP or wget to download the certificate file from an accessible host.
  - c. Copy the new cert to file: `' /usr/local/jboss/server/wpserver/conf/certs.p12'`
  - d. Update password values for the new certificate in the following files. On initial installation, they are all set to "democertpassword":  
`'/usr/local/jboss/server/wpserver/conf/ssl.properties'`
  - e. Update the values for properties "keystore.password" and "truststore.password"  
`'/usr/local/jboss/server/wpserver/conf/jboss-service.xml'`
  - f. Update value for property "KeyStorePass"  
`'/usr/local/jboss/server/wpserver/deploy/jbossweb-tomcat55.sar/server.xml'`
2. Restart the Network Services Manager engine service to exit the shell and return to the admin# prompt:

```
service nsm-engine restart
```

### To upload a new certificate to the controller:

1. SSH onto the controller and enter **shell** at the prompt to get into the command line.
2. Use SCP or wget to download the certificate file from an accessible host.
3. Copy the new cert to the file `'/etc/overdrive/certs.p12'`.
4. Update the password value for the new certificate in the following file by replacing the value for property "keystore.password".  
For example, change "democertpassword" to the new password value, (such as `keystore.password=MY_NEW_CERT_PASSWORD`): `'/etc/overdrive/vmdc-controller/ssl.properties'`
5. Reboot the controller with the **reload** command or issue the application command to start the controller, as follows:

```
application start nsm-controller
```

## Base Configuration

The devices that are managed by Network Services Manager need to have been configured in the device stack and have certain standard base configurations applied to them. Device configurations should be saved before Network Services Manager begins to automate tenant activity configurations to provide a clean, unconfigured start state on the device.

**Note** Network Services Manager does not save the configurations but instead keeps them in running configuration. A good practice is to save the running configuration before Network Services Manager begins to manage the device to avoid having tenant artifacts saved on the devices.

Network Services Manager automates deploying tenant logical containers on a preconfigured stack of equipment. It does not do bare-metal configuration.

The following sections capture the base configuration prerequisites that apply to each device.

### Preserving Static Routes on Devices

**Prerequisite** Enable the "shell" password-protected access by using the **shell\_enable** command as follows:

```
od-11-dsc/admin# shell_enable
Enter a password for shell access :
Confirm the password again :
Info: Shell access password was set successfully. Use shell command to access
shell
od-11-dsc/admin#
```

**Known Issue** The controller, upon connecting to devices in the Router or Distribution switch role, deletes static routes present in the devices' default VRF. If the device routing tables are populated with routes that must remain untouched by the controller, create a boilerplate for each device to preserve the desired routes. When static routes are added manually to the routers, you must also configure new boilerplates on the controller to ensure that it does not remove them.

The boilerplate configuration is stored on the controller at:

```
/etc/overdrive/controller/boilerplates.xml
```

A sample boilerplate that will preserve six different static routes on a router with IP address 10.17.217.98 would look like this:

```
<boilerplate description="Add static routes for N7K-2">
  <filter>
    <action>write-routes</action>
    <property key="remoteLogin.targetAddress"
match="exactly">10.17.217.98</property>
  </filter>
  <body type="com.cisco.overdrive.routing.service.RouteXml">
    <staticRoutes>
```

```

        <staticRoute destAddr="10.3.0.0" destMask="255.255.0.0"
nextHop="145.34.56.2" />
        <staticRoute destAddr="10.3.1.0" destMask="255.255.255.0"
nextHop="145.34.56.2" />
        <staticRoute destAddr="10.4.0.0" destMask="255.255.0.0"
nextHop="145.34.56.2" />
        <staticRoute destAddr="10.4.1.0" destMask="255.255.255.0"
nextHop="145.34.56.2" />
        <staticRoute destAddr="10.17.217.0" destMask="255.255.255.0"
nextHop="145.34.56.2" />
        <staticRoute destAddr="192.0.2.1" destMask="255.255.255.255"
nextHop="145.34.56.2" />
        <staticRoute destAddr="10.1.1.0" destMask="255.255.255.0"
nextHop="145.34.56.2" />
    </staticRoutes>
</body>
</boilerplate>

```

## ASR 1004

Ensure that the ports supplied when configuring the L3Routed type interconnects for the device are created on the device and that their status is operational. For example, if interface "port-channel 10" was specified as the ASR downlink to the 7000 device, it should look similar to the following:

```

od-l1-asr1k-a# show running-config interface port-channel 10
Building configuration...

Current configuration : 75 bytes
!
interface Port-channel10
  description to od-l1-n7k-c
  no ip address
end

od-l1-asr1k-a# show running-config interface TenGigabitEthernet0/0/0
Building configuration...

Current configuration : 98 bytes
!
interface TenGigabitEthernet0/0/0
  no ip address
  cdp enable
  channel-group 10 mode active
end

od-l1-asr1k-a#

```

## Nexus 7000

**Tips** (see attached sample base config)

1. Bring up the management interface.
2. Create default route within management VRF.

3. Enter following commands:
  - a. **feature interface-vlan**
  - b. **feature ssh** OR **feature telnet**
  - c. **feature ospf**
  - d. **feature bfd**
  - e. **no password strength-check**
  - f. **username admin password cisco123 role vdc-admin**
  - g. **snmp-server user admin auth md5 cisco123 priv cisco123**

Ensure that uplinks (L3Routed interconnects) to the ASR devices are configured correctly and are operational, as shown in the following example:

```
od-11-n7k-c# show running-config int port-channel 10

!Command: show running-config interface port-channel10
!Time: Thu Oct 20 19:58:48 2011

version 5.2(1)

interface port-channel10
  description od-11-asr1k-a

od-11-n7k-c# show running-config interface Eth9/3

!Command: show running-config interface Ethernet9/3
!Time: Wed Nov 23 07:34:57 2011

version 5.2(1)

interface Ethernet9/3
  channel-group 10 mode passive
  no shutdown

od-11-n7k-c#
```

Confirm that the trunks to the L2 aggregation switches and the DSN-VSS links (L2Trunk interconnects) are configured in switchport mode and set to trunk:

<b>Note</b>	If you want VLANs to be explicitly allowed on L2 Trunks, those trunks must already have an allowed statement (including none for newly created trunks). L2 links between redundant L2 devices must implicitly allow all VLANs.
-------------	--

```
od-11-n7k-c# show running-config interface port-channel 5

!Command: show running-config interface port-channel5
!Time: Thu Oct 20 20:00:35 2011

version 5.2(1)

interface port-channel5
  description vPC to od-11-n5k-a+b
```

```
switchport
switchport mode trunk
switchport trunk allowed vlan 1
vpc 5
```

```
od-11-n7k-c# show running-config interface Eth9/5
```

```
!Command: show running-config interface Ethernet9/5
!Time: Wed Nov 23 07:33:45 2011
```

```
version 5.2(1)
```

```
interface Ethernet9/5
switchport
switchport mode trunk
switchport trunk allowed vlan 1
channel-group 5 mode active
no shutdown
```

```
od-11-n7k-c# show running-config interface Eth9/6
```

```
!Command: show running-config interface Ethernet9/6
!Time: Wed Nov 23 07:34:07 2011
```

```
version 5.2(1)
```

```
interface Ethernet9/6
switchport
switchport mode trunk
switchport trunk allowed vlan 1
channel-group 5 mode active
no shutdown
```

```
od-11-n7k-c#
```

Ensure the port-channels are up and operational:

```
od-11-n7k-c# show port-channel summary
```

```
Flags: D - Down      P - Up in port-channel (members)
       I - Individual H - Hot-standby (LACP only)
       s - Suspended r - Module-removed
       S - Switched  R - Routed
       U - Up (port-channel)
       M - Not in use. Min-links not met
```

```
-----
Group Port-      Type   Protocol Member Ports
Channel
-----
1   Po1(SU)   Eth    LACP    Eth9/1(P) Eth9/2(P)
5   Po5(SU)   Eth    LACP    Eth9/5(P) Eth9/6(P)
7   Po7(SU)   Eth    LACP    Eth9/7(P) Eth9/8(P)
10  Po10(RU)  Eth    LACP    Eth9/3(P)
11  Po11(RU)  Eth    LACP    Eth9/4(P)
od-11-n7k-c#
```



If using vPCs, make sure the vPC is up and permits VLANs, as in the following example:

```
od-11-n7k-c# show vpc
Legend:
    (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id          : 3
Peer status             : peer adjacency formed ok
vPC keep-alive status  : peer is alive
Configuration consistency status : success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role                : primary
Number of vPCs configured : 2
Peer Gateway            : Enabled
Peer gateway excluded VLANs : -
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled
Auto-recovery status    : Enabled (timeout = 240 seconds)

vPC Peer-link status
-----
id  Port  Status Active vlans
--  ---  -
1  Po1  up    1

vPC status
-----
id  Port  Status Consistency Reason          Active vlans
--  ---  -
5  Po5  up    success  success          1
7  Po7  up    success  success          1

od-11-n7k-c#
```

## DSN VSS

Ensure that the VSS is formed correctly:

```
od-c3-vss#show switch virtual
Switch mode          : Virtual Switch
Virtual switch domain number : 149
Local switch number  : 1
Local switch operational role: Virtual Switch Active
Peer switch number   : 2
Peer switch operational role : Virtual Switch Standby
od-c3-vss#
```

Ensure that the VSL link is up:

```
od-c3-vss#show switch virtual link
VSL Status : UP
VSL Uptime : 5 weeks, 4 days, 2 hours, 47 minutes
```

```
VSL SCP Ping : Pass
VSL ICC Ping : Pass
VSL Control Link : Te1/5/4
od-c3-vss#
```

## Service Modules

Enable multiple context mode:

```
od-c2-fwsm-a/3/act(config)# mode multiple
Security context mode: multiple
The flash mode has not been modified.
The requested mode is the SAME as the flash mode.
od-c2-fwsm-a/3/act(config)#
```

Configure the firewall service modules (FWSMs) into a failover pair, using an Active/Active failover setup. Ensure that the failover groups are operational on both modules.

```
od-c2-fwsm-a/3/act# show running-config failover
failover
failover lan unit primary
failover lan interface fa-lan Vlan50
failover link fa-state Vlan51
failover interface ip fa-lan 8.10.8.1 255.255.255.0 standby 8.10.8.2
failover interface ip fa-state 8.10.9.1 255.255.255.0 standby 8.10.9.2
failover group 1
  preempt
failover group 2
  secondary
  preempt
od-c2-fwsm-a/3/act#
```

Network Services Manager expects the same user credentials for the service modules as for the VSS chassis that houses the modules.

Network Services Manager also supports Adaptive Security Appliance (ASA) and Application Control Engine (ACE) service modules. ASA service modules are the default in a POD, but you can alter the default to be FWSM by updating an entry in `/usr/local/overdrive/controller/devices/cisco/config/CiscoDevice.properties`.

In `CiscoDevice.properties`, change the value for `service.module.locator` from `ASASM` to `FWSM`:

```
# Service Module Locator on Cat 6k can be ASASM or FWSM, default is ASASM
service.module.locator=ASASM
```

You need to change this entry only if the POD has both ASA and FW service modules and you want to use the FW service module.

## Nexus 5000

Verify that uplinks and downlink trunks are configured in switchport mode and set to trunk:

```
od-l1-n5k-b# show running-config int port-channel 5

!Command: show running-config interface port-channel5
!Time: Wed Oct 26 18:28:12 2011

version 5.0(3)N2(2)

interface port-channel5
 description UCS fabric interconnect A
 switchport mode trunk
 vpc 5
 switchport trunk allowed vlan 229
 spanning-tree port type edge trunk

od-l1-n5k-b# show running-config interface ethernet 1/5

!Command: show running-config interface Ethernet1/5
!Time: Tue Dec 13 18:10:25 2011

version 5.0(3)N2(2)

interface Ethernet1/5
 switchport mode trunk
 switchport trunk allowed vlan 229
 channel-group 5 mode active

od-l1-n5k-b#
```

If using vPCs, make sure the vPC is up and permits VLANs, as in the following example:

```
od-l1-n5k-b# show vpc
Legend:
    (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id          : 7
Peer status            : peer adjacency formed ok
vPC keep-alive status  : peer is alive
Configuration consistency status: success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role               : primary
Number of vPCs configured : 3
Peer Gateway          : Enabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled

vPC Peer-link status
-----
id  Port  Status Active vlans
--  ---  -----
1   Po1   up      1,229
```

```
vPC status
```

```
-----  
id   Port      Status Consistency Reason      Active vlans  
-----  
3    Po3        up    success  success      -  
5    Po5        up    success  success      229  
6    Po6        up    success  success      229  
-----
```

```
od-11-n5k-b#
```

## UCS

Network Services Manager 5.0.2 supports the UCS 61xxXP and 62xxUP with UCSM 1.4x in end-host mode, where the UCS does not let the administrator control its uplinks and, by default, permits any created VLANs on any interface configured as "Ethernet Uplink".

Because of this implementation design, in UCSM 1.4x, the Management VLANs for the ESXi blades connected to the UCS chassis will traverse the same physical links as the tenant data path. This limitation has been removed in UCSM 2.0x and, as such, does not limit the configuration.

Network Services Manager 5.0.2 will always create its VLAN in a Dual FC membership; that is, both Fabric Interconnects will have the VLAN available to a vNIC template assignment. For Network Services Manager 5.0.2, there is no option to choose to which Fabric Interconnect the VLAN will be created.

Preconfiguring the UCS 6120 for deployment involves creating up to two vNIC templates; these templates should be in updating mode and assigned to separate Fabric Interconnects, as in the following example:

```
od-11-ucs-A /org # show vnic-templ  
  
vNIC Template:  
Name              Type              Fabric ID  
-----  
od-11/od-11_vnic0 Updating Template A  
od-11/od-11_vnic1 Updating Template B  
od-11-ucs-A /org #
```

## Nexus 1000V

Verify the uplink port-profile is created and configured in switchport mode trunk, as shown in the following example:

```
od-11-vsm# show running-config port-profile n1kv-uplink0  
  
!Command: show running-config port-profile n1kv-uplink0  
!Time: Wed Oct 26 18:04:46 2011  
  
version 4.2(1)SV1(4a)  
port-profile type ethernet n1kv-uplink0  
  vmware port-group  
  switchport mode trunk
```

```
switchport trunk allowed vlan 229
channel-group auto mode on mac-pinning
no shutdown
system vlan 229
state enabled

od-l1-vsm#
```

## Topology and Business Model Configuration

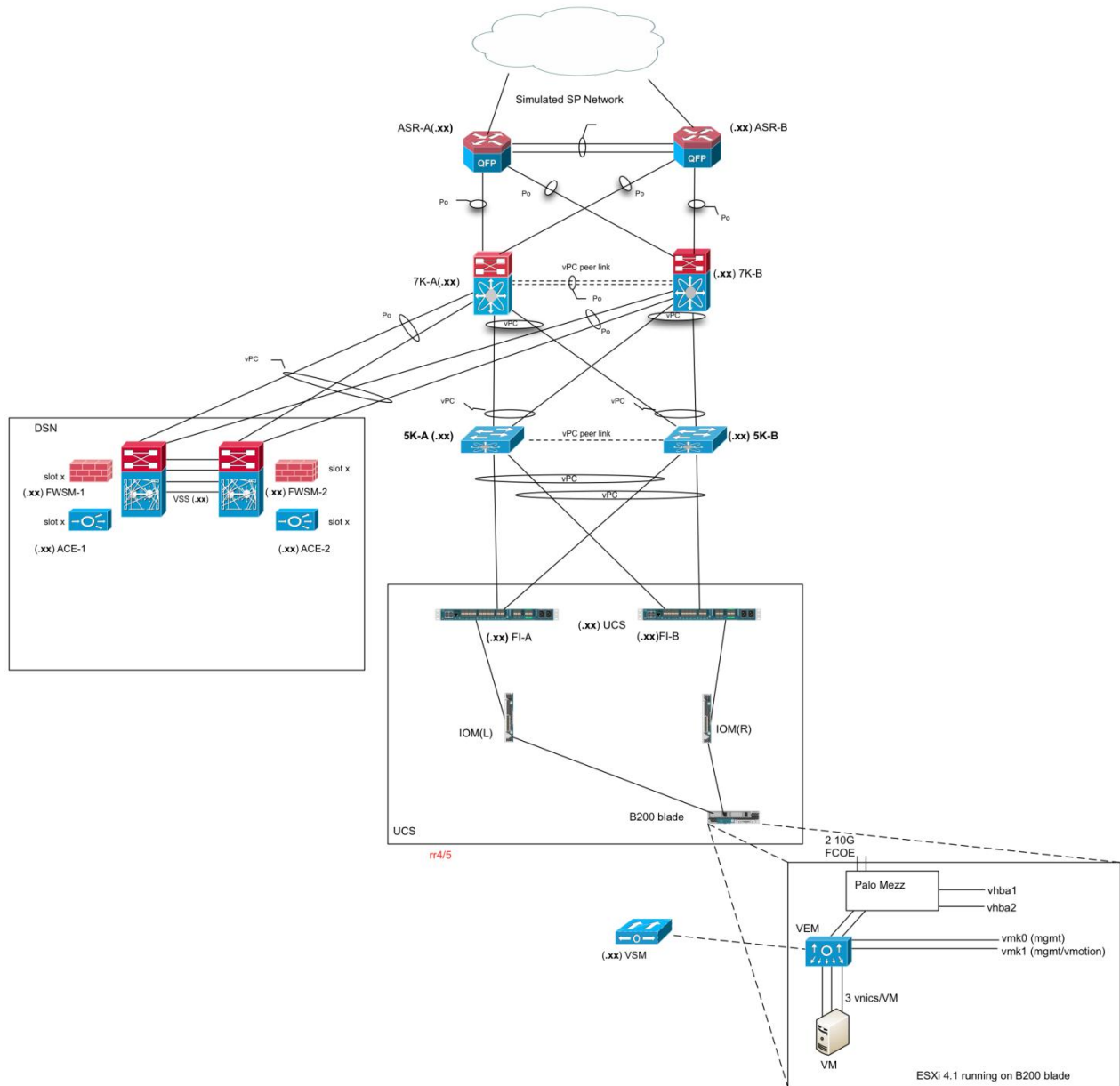
The Network Services Manager engine ships with a preconfigured business model that contains a metamodel for building a Provider and Tenants. Refer to the [Network Services Manager API documentation](#) for more information on the current API model.

Whereas the WebUI enables you to make minor edits of the device stack (such as updating credentials, or modifying port assignments), defining the entire stack is better achieved with a REST API call using a pre-populated REST request body. For more information on how to use the XML-based REST API, see the API document and the following training materials available from Applied Concepts, Inc.:

- [System Installation and Administration Training](#)
- [API Training](#)

The topology model duplicates the network diagram of one of the labs that were used to validate the device configurations and is the best fit for a Network Services Manager deployment. The following diagram shows how the network is configured.

**Figure 1 Simulated Topology Model**



## Visualizing the Stack as a Topology Model

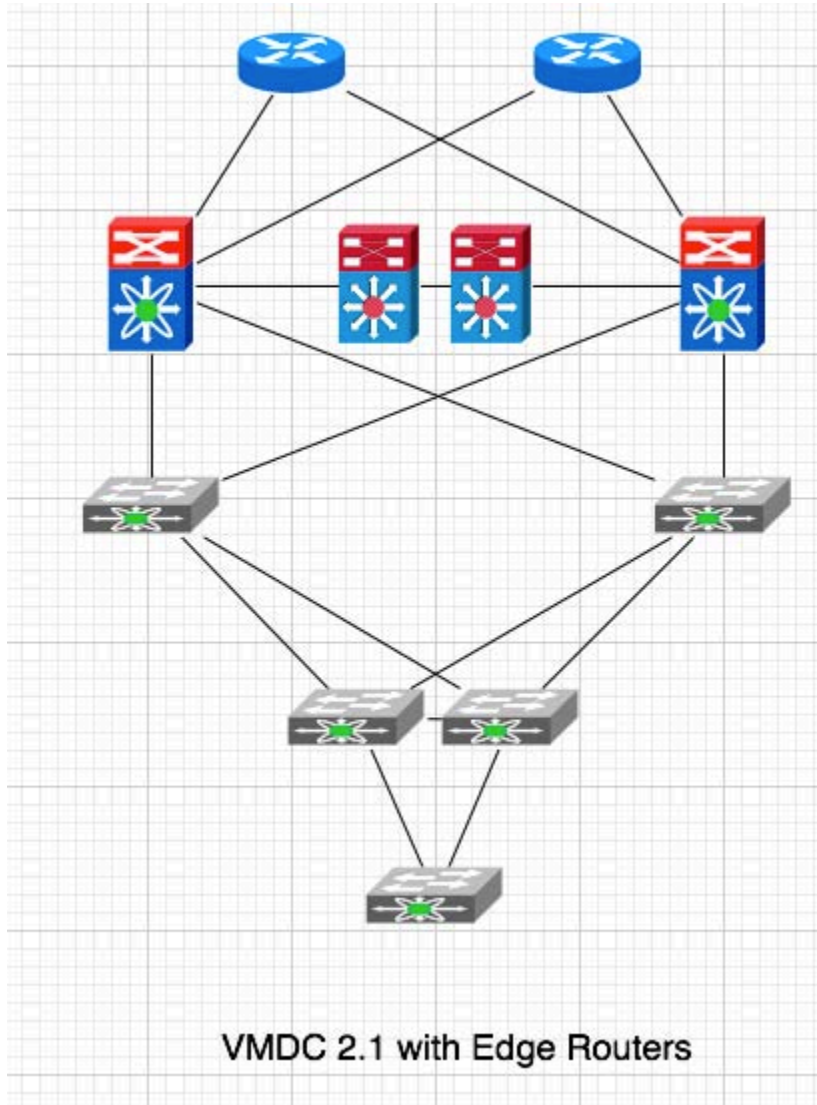
This task is complex and requires careful attention to detail. Network Services Manager will accept the definition of the interface names and numbers and create configurations based on the supplied information, so it is important to pay close attention to the device base configurations and credentials when you edit this file.

The first step is to build a graphical depiction of the stack. You might use the diagram above and fill in all of the following details:

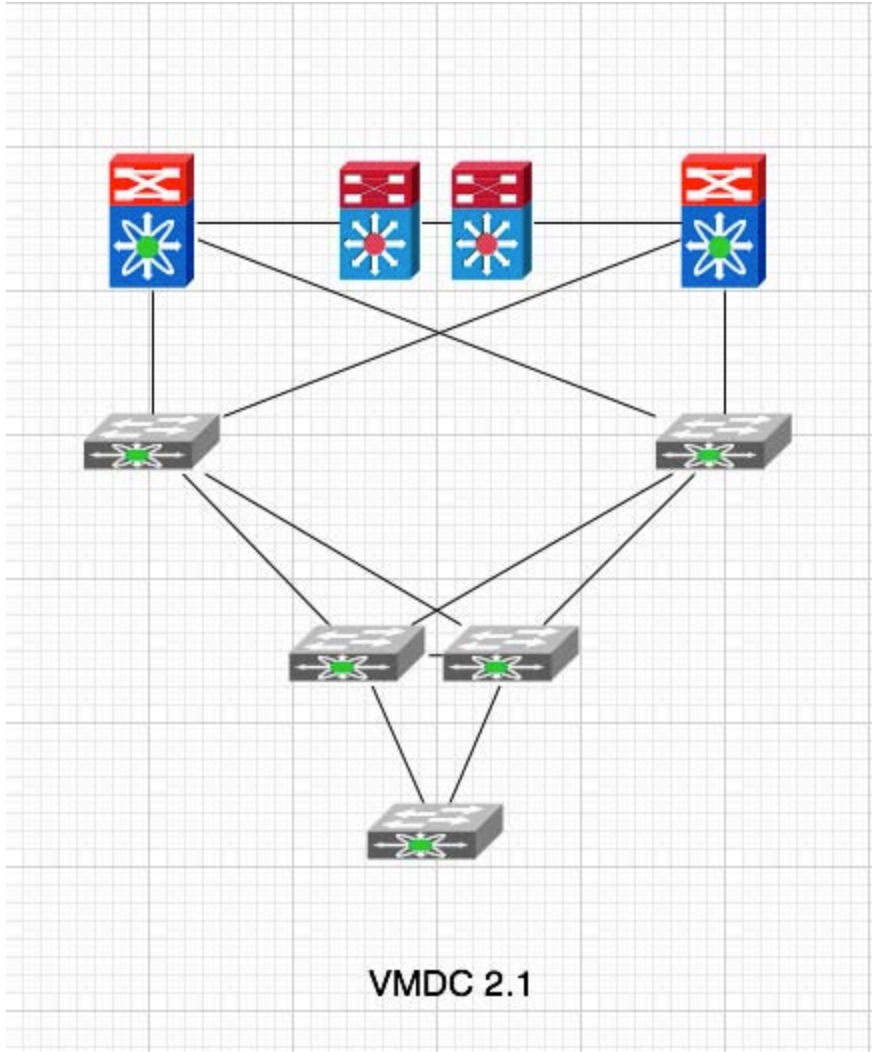
- Each device A and B with:
  - Login credentials
  - SNMP v2c RW communities or SNMP V3 auth credentials
  - Downlink/uplink port assignments (port channels or physical links)
- For each interconnecting cable, keep in mind the interconnect object in the model has an uplink that is attached to a downlink port on the device above it and a downlink that is attached to the uplink port of the device below it. So, for each interconnect, match the port assignment for each device it connects to.
- Make a note of the VLAN pool that will be used for tenant networks. This pool is described for each device. You should match the pools on the devices to permit VLANs to exist across the stack where needed.
- Decide on a service provider POD infrastructure pool. This pool needs to be an address pool that does not conflict with tenant-provided RFC1918 addresses or with publicly routed addresses. This should not be changed after tenants have been deployed.

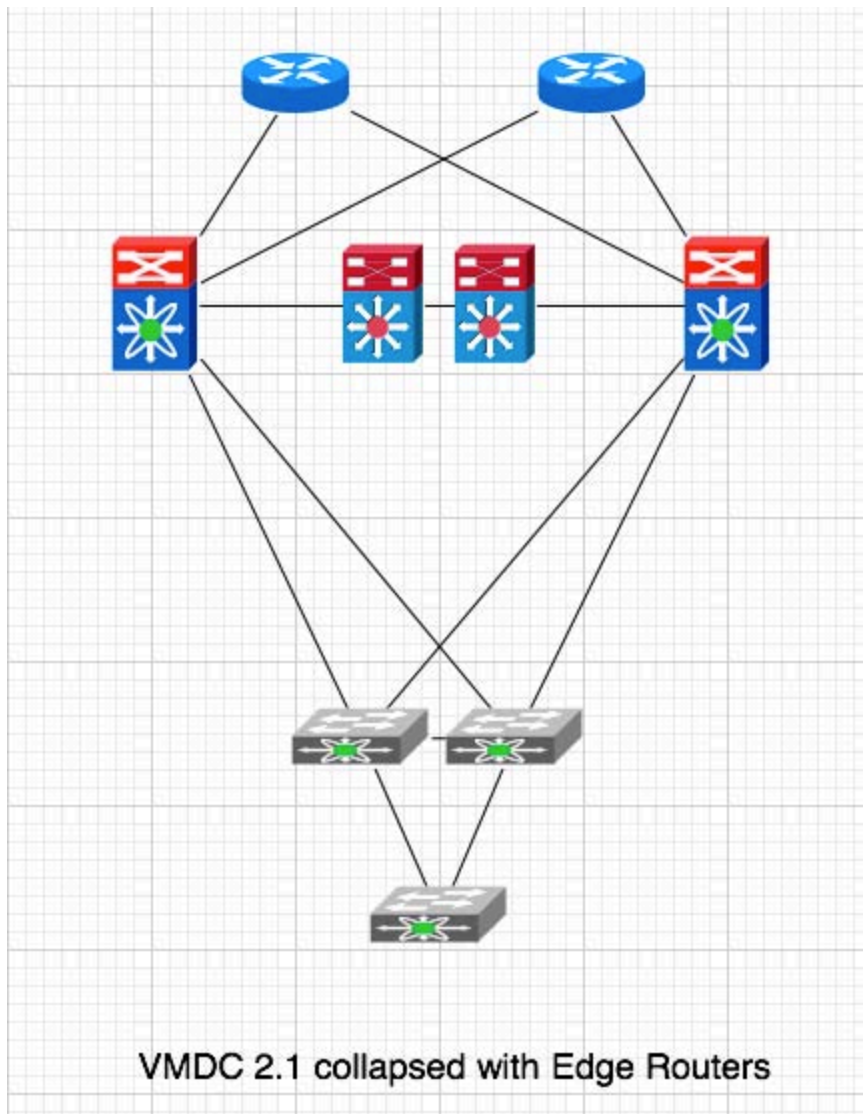
## Topology Variations

The following are topology variations that are supported in the metamodel shipped with Network Services Manager 5.0.2.









## Edit the Topology Model

Carefully edit the API request body with this information. Do not provide conflicting information for SNMP (that is, do not supply SNMP v3 relevant settings if v2c is used and vice versa). A detailed example follows.

## Create the POD with this Request Body

The REST request body for a simple "collapsed" stack is shown below as an example of the information that is requested for a POD create. Note that some parameters are set to empty value. For example, SNMP v3 information is not required if v2c is configured. Similarly, if you do not configure the management address of a device, it will be described in Network Services Manager but excluded from the topology model when the provisioning engine calculates the data path.

```
<?xml version="1.0" encoding="UTF-8"?>
<serviceOffering xmlns="http://www.cisco.com/NetworkServicesManager/1.1"
  xmlns:ns2="http://www.w3.org/2005/Atom" >
```

```

<uid>${INSERT SERVICE OFFERING UID HERE}</uid>
<name>${INSERT NAME HERE}</name>
<description />
<parameters>
  <parameter>
    <name>pod.infrastructure.pool</name>
    <type>subnets</type>
    <subnets>
      <subnet>
        <ipv4>129.0.0.0</ipv4>
        <mask>20</mask>
      </subnet>
    </subnets>
  </parameter>
  <parameter>
    <name>edge.router.2.rlogin.enable.pass</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>l3.aggregation.switch.2.rlogin.pass</name>
    <type>string</type>
    <string>cisco123</string>
  </parameter>
  <parameter>
    <name>l3.aggregation.switch.1.snmp.v3.priv.pass</name>
    <type>string</type>
    <string>cisco123</string>
  </parameter>
  <parameter>
<name>virtual.access.switch.1.snmp.v2c.comm.write</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>l3.aggregation.switch.1.snmp.v2c.comm.read</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>l2.access.switch.1.rlogin.type</name>
    <type>string</type>
    <string>ssh</string>
  </parameter>
  <parameter>
    <name>l2.access.switch.2.snmp.v3.auth.proto</name>
    <type>string</type>
    <string>MD5</string>
  </parameter>
  <parameter>
    <name>l3.aggregation.switch.1.snmp.ver</name>
    <type>string</type>
    <string>3</string>
  </parameter>
  <parameter>
    <name>virtual.access.switch.2.rlogin.enable.pass</name>

```

```

        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.1.snmp.v3.sec.name</name>
        <type>string</type>
        <string>admin</string>
    </parameter>
    <parameter>
        <name>l3.services.rlogin.user</name>
        <type>string</type>
        <string>admin</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.3.snmp.v3.priv.proto</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.2.rlogin.user</name>
        <type>string</type>
        <string>admin</string>
    </parameter>
    <parameter>
        <name>l2.access.switch.2.snmp.v3.priv.pass</name>
        <type>string</type>
        <string>cisc0123</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.1.snmp.v3.auth.proto</name>
        <type>string</type>
        <string>MD5</string>
    </parameter>
    <parameter>
        <name>l3.services.snmp.v3.auth.pass</name>
        <type>string</type>
        <string>cisc0123</string>
    </parameter>
    <parameter>
        <name>edge.router.2.to.l3.aggregation.switch.1</name>
        <type>string</type>
        <string>Po11</string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.1.snmp.v3.auth.proto</name>
        <type>string</type>
        <string>MD5</string>
    </parameter>
    <parameter>
        <name>ucsm.mgmt.addr</name>
        <type>string</type>
        <string>240.1.1.5</string>
    </parameter>
    <parameter>
        <name>edge.router.1.to.l3.aggregation.switch.2</name>
        <type>string</type>
        <string>Po11</string>

```

```

</parameter>
<parameter>
  <name>l3.services.snmp.v3.priv.proto</name>
  <type>string</type>
  <string>DES</string>
</parameter>
<parameter>
  <name>virtual.access.switch.1.snmp.v3.priv.proto</name>
  <type>string</type>
  <string>DES</string>
</parameter>
<parameter>
  <name>edge.router.1.snmp.v3.priv.pass</name>
  <type>string</type>
  <string>cisco123</string>
</parameter>
<parameter>
  <name>l3.services.to.l3.aggregation.switch.2</name>
  <type>string</type>
  <string>Po7</string>
</parameter>
<parameter>
  <name>virtual.access.switch.2.rlogin.type</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>virtual.access.switch.3.rlogin.enable.pass</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>l3.aggregation.switch.1.rlogin.pass</name>
  <type>string</type>
  <string>cisco123</string>
</parameter>
<parameter>
  <name>edge.router.1.snmp.v3.sec.name</name>
  <type>string</type>
  <string>admin</string>
</parameter>
<parameter>
  <name>l3.services.snmp.v3.auth.proto</name>
  <type>string</type>
  <string>MD5</string>
</parameter>
<parameter>
  <name>l3.aggregation.switch.1.to.l2.access.switch.1</name>
  <type>string</type>
  <string>Po5</string>
</parameter>
<parameter>
  <name>ucsm.snmp.v3.sec.name</name>
  <type>string</type>
  <string>snmp_admin</string>
</parameter>

```

```

    <parameter>
      <name>l3.services.snmp.ver</name>
      <type>string</type>
      <string>3</string>
    </parameter>
  <parameter>
    <name>l3.aggregation.switch.2.to.l2.access.switch.1</name>
    <type>string</type>
    <string>Po5</string>
  </parameter>
  <parameter>
    <name>l3.services.to.l3.aggregation.switch.1</name>
    <type>string</type>
    <string>Po7</string>
  </parameter>
  <parameter>
    <name>edge.router.2.snmp.v3.sec.name</name>
    <type>string</type>
    <string>admin</string>
  </parameter>
  <parameter>
    <name>virtual.access.switch.1.snmp.v3.priv.pass</name>
    <type>string</type>
    <string>cisco123</string>
  </parameter>
  <parameter>
    <name>edge.router.2.snmp.v3.auth.proto</name>
    <type>string</type>
    <string>MD5</string>
  </parameter>
  <parameter>
    <name>l2.access.switch.2.to.l3.aggregation.switch.1</name>
    <type>string</type>
    <string>Po3</string>
  </parameter>
  <parameter>
    <name>l2.access.switch.2.to.l3.aggregation.switch.2</name>
    <type>string</type>
    <string>Po3</string>
  </parameter>
  <parameter>
    <name>virtual.access.switch.2.mgmt.addr</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>virtual.access.switch.2.snmp.ver</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>l2.access.switch.2.to.ucsm</name>
    <type>string</type>
    <string>Po5,Po6</string>
  </parameter>

```

```

        </parameter>
        <parameter>
            <name>l3.aggregation.switch.2.snmp.v2c.comm.write</name>
            <type>string</type>
            <string></string>
        </parameter>
        <parameter>
            <name>l2.access.switch.2.snmp.v3.auth.pass</name>
            <type>string</type>
            <string>cisco123</string>
        </parameter>
        <parameter>
            <name>l3.aggregation.switch.2.snmp.v2c.comm.read</name>
            <type>string</type>
            <string></string>
        </parameter>
        <parameter>
            <name>virtual.access.switch.2.to.ucsm</name>
            <type>string</type>
            <string></string>
        </parameter>
        <parameter>
            <name>l2.access.switch.1.mgmt.addr</name>
            <type>string</type>
            <string>240.1.1.3</string>
        </parameter>
        <parameter>
            <name>virtual.access.switch.3.snmp.v3.auth.proto</name>
            <type>string</type>
            <string></string>
        </parameter>
        <parameter>
            <name>edge.router.1.snmp.v2c.comm.read</name>
            <type>string</type>
            <string></string>
        </parameter>
        <parameter>
            <name>l3.aggregation.switch.2.rlogin.enable.pass</name>
            <type>string</type>
            <string></string>
        </parameter>
        <parameter>
            <name>l3.aggregation.switch.1.snmp.v3.sec.name</name>
            <type>string</type>
            <string>admin</string>
        </parameter>
        <parameter>
            <name>edge.router.2.snmp.v2c.comm.read</name>
            <type>string</type>
            <string></string>
        </parameter>
        <parameter>
            <name>l2.access.switch.1.rlogin.user</name>
            <type>string</type>
            <string>admin</string>
        </parameter>
    
```

```

    <parameter>
      <name>l3.aggregation.switch.2.snmp.v3.priv.proto</name>
      <type>string</type>
      <string>DES</string>
    </parameter>
  <parameter>
    <name>ucsm.to.virtual.access.switch.1</name>
    <type>string</type>
    <string>od-l1/od-l1_vnic0,od-l1/od-l1_vnic1</string>
  </parameter>
<parameter>
  <name>virtual.access.switch.2.snmp.v2c.comm.write</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>edge.router.2.rlogin.user</name>
  <type>string</type>
  <string>admin</string>
</parameter>
<parameter>
  <name>ucsm.rlogin.user</name>
  <type>string</type>
  <string>admin</string>
</parameter>
<parameter>
  <name>virtual.access.switch.2.snmp.v3.auth.proto</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>ucsm.snmp.v3.priv.proto</name>
  <type>string</type>
  <string>DES</string>
</parameter>
<parameter>
  <name>l3.services.snmp.v3.sec.name</name>
  <type>string</type>
  <string>admin</string>
</parameter>
<parameter>
  <name>l2.access.switch.1.rlogin.pass</name>
  <type>string</type>
  <string>cisco123</string>
</parameter>
<parameter>
  <name>l3.aggregation.switch.2.to.edge.router.1</name>
  <type>string</type>
  <string>Poll</string>
</parameter>
<parameter>
  <name>virtual.access.switch.1.to.ucsm</name>
  <type>string</type>
  <string>n1kv-uplink0</string>
</parameter>
<parameter>

```



```

        <name>edge.router.2.rlogin.type</name>
        <type>string</type>
        <string>ssh</string>
    </parameter>
    <parameter>
        <name>l3.services.rlogin.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.1.snmp.ver</name>
        <type>string</type>
        <string>3</string>
    </parameter>
    <parameter>
        <name>ucsm.snmp.v2c.comm.write</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>l2.access.switch.1.snmp.v3.priv.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.2.snmp.v2c.comm.read</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>edge.router.1.snmp.v2c.comm.write</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.1.rlogin.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>pod.controller.pwd</name>
        <type>string</type>
        <string>password</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.1.rlogin.type</name>
        <type>string</type>
        <string>ssh</string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.2.rlogin.type</name>
        <type>string</type>
        <string>ssh</string>
    </parameter>
    <parameter>
        <name>l2.access.switch.1.snmp.v3.auth.pass</name>
        <type>string</type>

```

```

        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>ucsm.rlogin.type</name>
        <type>string</type>
        <string>ssh</string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.1.rlogin.user</name>
        <type>string</type>
        <string>admin</string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.2.to.edge.router.2</name>
        <type>string</type>
        <string>Po10</string>
    </parameter>
    <parameter>
        <name>ucsm.snmp.v3.auth.pass</name>
        <type>string</type>
        <string>ciscotest</string>
    </parameter>
    <parameter>
        <name>ucsm.rlogin.enable.pass</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>l2.access.switch.2.snmp.v2c.comm.read</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.3.snmp.v2c.comm.write</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>l2.access.switch.2.snmp.v3.sec.name</name>
        <type>string</type>
        <string>admin</string>
    </parameter>
    <parameter>
        <name>edge.router.1.rlogin.user</name>
        <type>string</type>
        <string>admin</string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.1.snmp.v3.priv.proto</name>
        <type>string</type>
        <string>DES</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.3.mgmt.addr</name>
        <type>string</type>
        <string></string>

```

```

</parameter>
<parameter>
  <name>ucsm.rlogin.pass</name>
  <type>string</type>
  <string>cisco123</string>
</parameter>
<parameter>
  <name>l3.services.snmp.v2c.comm.read</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>ucsm.to.virtual.access.switch.3</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>l2.access.switch.2.mgmt.addr</name>
  <type>string</type>
  <string>240.1.1.4</string>
</parameter>
<parameter>
  <name>pod.device.vlanpool</name>
  <type>range</type>
  <range>
    <start>400</start>
    <end>600</end>
  </range>
</parameter>
<parameter>
  <name>pod.controller.name</name>
  <type>string</type>
  <string>vmdc-controller</string>
</parameter>
<parameter>
  <name>l2.access.switch.1.snmp.v3.priv.proto</name>
  <type>string</type>
  <string>DES</string>
</parameter>
<parameter>
  <name>ucsm.snmp.v3.auth.proto</name>
  <type>string</type>
  <string>MD5</string>
</parameter>
<parameter>
  <name>l3.services.rlogin.type</name>
  <type>string</type>
  <string>ssh</string>
</parameter>
<parameter>
  <name>l3.aggregation.switch.2.snmp.v3.sec.name</name>
  <type>string</type>
  <string>admin</string>
</parameter>
<parameter>
  <name>l3.aggregation.switch.2.to.l2.access.switch.2</name>

```

```

        <type>string</type>
        <string>Po5</string>
    </parameter>
</parameter>

<name>12.access.switch.1.to.13.aggregation.switch.1</name>
    <type>string</type>
    <string>Po3</string>
</parameter>
</parameter>

<name>13.aggregation.switch.1.snmp.v2c.comm.write</name>
    <type>string</type>
    <string></string>
</parameter>
</parameter>
    <name>13.aggregation.switch.1.rlogin.type</name>
    <type>string</type>
    <string>ssh</string>
</parameter>
</parameter>
    <name>edge.router.2.mgmt.addr</name>
    <type>string</type>
    <string>240.2.1.2</string>
</parameter>
</parameter>
    <name>ucsm.snmp.v3.priv.pass</name>
    <type>string</type>
    <string>ciscotest</string>
</parameter>
</parameter>
    <name>13.services.snmp.v2c.comm.write</name>
    <type>string</type>
    <string></string>
</parameter>
</parameter>

<name>13.aggregation.switch.1.to.12.access.switch.2</name>
    <type>string</type>
    <string>Po5</string>
</parameter>
</parameter>
    <name>virtual.access.switch.1.mgmt.addr</name>
    <type>string</type>
    <string>240.1.1.7</string>
</parameter>
</parameter>
    <name>12.access.switch.1.rlogin.enable.pass</name>
    <type>string</type>
    <string></string>
</parameter>
</parameter>
    <name>virtual.access.switch.3.snmp.ver</name>
    <type>string</type>
    <string></string>
</parameter>
</parameter>

```

```

        <name>l3.services.mgmt.addr</name>
        <type>string</type>
        <string>240.4.0.15</string>
</parameter>
<parameter>
    <name>l2.access.switch.1.to.ucsm</name>
    <type>string</type>
    <string>Po5,Po6</string>
</parameter>
<parameter>
    <name>virtual.access.switch.2.snmp.v3.auth.pass</name>
    <type>string</type>
    <string></string>
</parameter>
<parameter>
    <name>l2.access.switch.2.rlogin.user</name>
    <type>string</type>
    <string>admin</string>
</parameter>
<parameter>
    <name>l2.access.switch.1.snmp.ver</name>
    <type>string</type>
    <string>3</string>
</parameter>
<parameter>
    <name>edge.router.1.to.l3.aggregation.switch.1</name>
    <type>string</type>
    <string>Po10</string>
</parameter>
<parameter>
    <name>virtual.access.switch.3.snmp.v3.priv.pass</name>
    <type>string</type>
    <string></string>
</parameter>
<parameter>
    <name>edge.router.1.snmp.v3.auth.pass</name>
    <type>string</type>
    <string>cisco123</string>
</parameter>
<parameter>
    <name>virtual.access.switch.3.snmp.v3.sec.name</name>
    <type>string</type>
    <string></string>
</parameter>
<parameter>
    <name>l3.aggregation.switch.2.snmp.v3.priv.pass</name>
    <type>string</type>
    <string>cisco123</string>
</parameter>
<parameter>
    <name>ucsm.snmp.ver</name>
    <type>string</type>
    <string>3</string>
</parameter>
<parameter>
    <name>l3.aggregation.switch.2.snmp.v3.auth.proto</name>
    <type>string</type>

```

```

        <string>MD5</string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.1.to.edge.router.1</name>
        <type>string</type>
        <string>Po10</string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.2.mgmt.addr</name>
        <type>string</type>
        <string>240.1.0.11</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.2.snmp.v3.priv.pass</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.1.to.l3.services</name>
        <type>string</type>
        <string>Po7</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.2.snmp.v3.priv.proto</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>edge.router.2.snmp.v3.priv.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>edge.router.1.rlogin.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>l2.access.switch.2.rlogin.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.3.snmp.v3.auth.pass</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>l2.access.switch.2.snmp.ver</name>
        <type>string</type>
        <string>3</string>
    </parameter>
    <parameter>
        <name>edge.router.1.snmp.ver</name>
        <type>string</type>
        <string>3</string>
    </parameter>

```

```

<parameter>
  <name>l3.aggregation.switch.1.to.edge.router.2</name>
  <type>string</type>
  <string>Poll</string>
</parameter>
<parameter>
  <name>edge.router.1.rlogin.enable.pass</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>l2.access.switch.2.snmp.v2c.comm.write</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>edge.router.2.snmp.ver</name>
  <type>string</type>
  <string>3</string>
</parameter>
<parameter>
  <name>edge.router.2.snmp.v3.priv.proto</name>
  <type>string</type>
  <string>DES</string>
</parameter>
<parameter>
  <name>ucsm.to.virtual.access.switch.2</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>l3.services.rlogin.enable.pass</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>virtual.access.switch.3.to.ucsm</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>l3.aggregation.switch.1.rlogin.enable.pass</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>edge.router.2.snmp.v2c.comm.write</name>
  <type>string</type>
  <string></string>
</parameter>
<parameter>
  <name>l2.access.switch.2.rlogin.type</name>
  <type>string</type>
  <string>ssh</string>
</parameter>
<parameter>
  <name>virtual.access.switch.3.snmp.v2c.comm.read</name>

```

```

        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.1.snmp.v3.auth.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>l2.access.switch.1.snmp.v3.sec.name</name>
        <type>string</type>
        <string>admin</string>
    </parameter>
    <parameter>
        <name>l2.access.switch.1.snmp.v2c.comm.read</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>edge.router.2.snmp.v3.auth.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.1.rlogin.enable.pass</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>edge.router.1.snmp.v3.priv.proto</name>
        <type>string</type>
        <string>DES</string>
    </parameter>
    <parameter>
        <name>l3.aggregation.switch.2.snmp.ver</name>
        <type>string</type>
        <string>3</string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.2.rlogin.pass</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>edge.router.2.rlogin.pass</name>
        <type>string</type>
        <string>cisco123</string>
    </parameter>
    <parameter>
        <name>l2.access.switch.1.snmp.v2c.comm.write</name>
        <type>string</type>
        <string></string>
    </parameter>
    <parameter>
        <name>virtual.access.switch.1.snmp.v3.auth.pass</name>
        <type>string</type>
        <string>cisco123</string>

```



```

    </parameter>
    <parameter>
      <name>l2.access.switch.1.to.l3.aggregation.switch.2</name>
      <type>string</type>
      <string>Po3</string>
    </parameter>
    <parameter>
      <name>l3.aggregation.switch.1.mgmt.addr</name>
      <type>string</type>
      <string>240.1.0.10</string>
    </parameter>
    <parameter>
      <name>l2.access.switch.1.snmp.v3.auth.proto</name>
      <type>string</type>
      <string>MD5</string>
    </parameter>
    <parameter>
      <name>virtual.access.switch.1.rlogin.user</name>
      <type>string</type>
      <string>admin</string>
    </parameter>
    <parameter>
      <name>l3.aggregation.switch.2.snmp.v3.auth.pass</name>
      <type>string</type>
      <string>cisco123</string>
    </parameter>
    <parameter>
      <name>l3.aggregation.switch.2.to.l3.services</name>
      <type>string</type>
      <string>Po7</string>
    </parameter>
    <parameter>
      <name>edge.router.2.to.l3.aggregation.switch.2</name>
      <type>string</type>
      <string>Po10</string>
    </parameter>
    <parameter>
      <name>edge.router.1.rlogin.type</name>
      <type>string</type>
      <string>ssh</string>
    </parameter>
    <parameter>
      <name>l3.services.snmp.v3.priv.pass</name>
      <type>string</type>
      <string>cisco123</string>
    </parameter>
    <parameter>
      <name>virtual.access.switch.2.snmp.v3.sec.name</name>
      <type>string</type>
      <string></string>
    </parameter>
    <parameter>
      <name>ucsm.snmp.v2c.comm.read</name>
      <type>string</type>
      <string></string>
    </parameter>
  </parameter>

```

```

    <parameter>
      <name>virtual.access.switch.3.rlogin.user</name>
      <type>string</type>
      <string></string>
    </parameter>
  </parameter>
  <parameter>
    <name>l2.access.switch.2.snmp.v3.priv.proto</name>
    <type>string</type>
    <string>DES</string>
  </parameter>
  <parameter>
    <name>virtual.access.switch.3.rlogin.pass</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>virtual.access.switch.2.rlogin.user</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>virtual.access.switch.3.rlogin.type</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>virtual.access.switch.1.snmp.v2c.comm.read</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>l2.access.switch.2.rlogin.enable.pass</name>
    <type>string</type>
    <string></string>
  </parameter>
  <parameter>
    <name>edge.router.1.mgmt.addr</name>
    <type>string</type>
    <string>240.2.1.1</string>
  </parameter>
  <parameter>
    <name>edge.router.1.snmp.v3.auth.proto</name>
    <type>string</type>
    <string>MD5</string>
  </parameter>
</parameters>
</serviceOffering>

```

<b>Note</b>	<ul style="list-style-type: none"> <li>• Login credentials must put the controller into priv 15 directly OR you need to use the same enable password as the login password.</li> <li>• AES is the only supported priv cypher for SNMP v3.</li> <li>• Do not copy and paste interconnects or network elements.</li> </ul>
-------------	--

## Verify the Created POD

After the POD is created, log into the WebUI and navigate to the POD's Metaproperties tab, and verify the settings against those you captured earlier.

In the WebUI, navigate to the Site (VMDC POD), click the **Alerts** tab, and examine any errors reported by the controller. These will include failures to log in and failures to successfully identify the device (due to incorrect SNMP or an unsupported device in the stack).

## Verifying the Configuration

Before the first tenant creation dry run, verify the base configurations and controller access to devices. You can also verify controller access by using the **Show** commands with the devices via the UI.

More detailed examples of common errors are described in [Troubleshooting](#).

## Check the Alerts View for the POD

In the WebUI, navigate to the Site (VMDC POD), click the **Alerts** tab, and examine any errors reported by the controller.

If there are any errors or alerts, these will need to be accommodated before proceeding.

If the WebUI alert panel shows no data, it means that no alarms have been triggered, and the system is running free of errors:



Reporting Object	Alert	Description	Severity	When
No data available				

<b>Known Issue</b>	Unmanaged VLANs discovered on devices (VLANs that are not in the list of VLANs managed by Network Services Manager) are shown as errors. These are reported but no action needs to be taken because Network Services Manager will ignore these VLANs.
--------------------	---

If the Alerts view says "No data available," it indicates that there are no errors.

Other than unmanaged VLANs, which can be ignored, other typical error conditions are described below.

## Check the Controller Log

If there are no errors in the WebUI, this step is probably not required. Systems Engineers at site installations will do this as an extra precaution to identify things error conditions that may not yet have reached the timeout threshold that would trigger an alert in the WebUI.

During initial configuration of the device stack, we recommend that you enable DEBUG mode in the controller logs. Make a note to turn DEBUG mode off before handing the environment over to the customer.

To enable DEBUG in the controller log, log into the CARs CLI and execute:

```
od-11-dsc/admin# conf t
Enter configuration commands, one per line. End with CNTL/Z.
od-11-dsc/admin(config)# logging loglevel 7
od-11-dsc/admin(config)#
```

A common technique to use when looking for controller issues is to use the following commands:

**Note** You must use the **shell** command described in [Enabling the "shell" Password-Protected Access](#) to enter these commands.

```
grep -i error /var/log/overdrive*.log
grep -i exception /var/log/overdrive*.log
```

## Advanced Installation Topics

### Enabling the "shell" Password-Protected Access

Use the following procedure to enable shell password-protected access:

```
od-11-dsc/admin# shell_enable
Enter a password for shell access :
Confirm the password again :
Info: Shell access password was set successfully. Use shell command to access
shell
od-11-dsc/admin#
```

### Adding or Removing an L2 Aggregation Layer in the Stack

For tenant data path creation, the L2 aggregation role configuration involves creating the VLAN and making sure that the VLAN is present on uplink and downlink ports.

The WebUI is best suited to this activity.

When adding L2 aggregation devices and interconnects, the settings are the same as described in [Installing the Components](#).

Identify the devices that you will remove. These devices will have downlinks and uplinks already associated with other devices in the stack.

When you delete devices, we recommend that you delete the device and all interconnects that represent uplinks from that device. The downlink interconnects will then be updated to show that they are connected to the new layer above them.

There is a known issue with creating an interconnect in the WebUI to the UCS. As a result, when adding or removing an L2 aggregation layer that connects to the UCS, **edit** the interconnect from the UCS by changing the uplink device and port.

## Substituting a Like Device for an Officially Supported Device

You can substitute a device in a system that has tenants deployed on it as well; because this does not change the data path, Network Services Manager will "true-up" the device to the configuration required to support the existing tenants when it first connects to the device.

1. On the controller, enter shell mode by entering **shell** at the command prompt and entering the password used to enable the shell after installation. For more information, see [Enabling the "shell" Password-Protected Access](#).
2. Open the file `/usr/local/overdrive/controller/devices/cisco/config/CiscoDeviceLocator.xml` for editing:

```
[root@od-c3-dsc bin]# vi
/usr/local/overdrive/controller/devices/cisco/config/CiscoDeviceLocator.xml
1
```

Each entry in this file maps a device recognized by a combination of its `sysObjectID` and `sysDesc` SNMP MIBs to a command set that the controller has stored for this device.

3. Edit the file as needed to support the device you are adding.  
For example, to add support for a UCS device with a `sysObjectID` of "SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.9.12.3.1.3.849", you would add a line similar to the following to the file:

```
<device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.849" sysDescr="Cisco
NX-OS\(tm\) ucs"
  commandSet="CiscoUCS"
className="net.linesider.overdrive.cisco.device.CiscoUCS" />
```

4. After you edit the file, confirm that the resulting `CiscoDeviceLocator.xml` looks similar to the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<devices>
  <device sysObjectID="1.3.6.1.4.1.9.1.923" sysDescr="Cisco .*(IOS-XE)"
    commandSet="CiscoASR"
className="net.linesider.overdrive.cisco.device.CiscoASR" />
  <device sysObjectID="1.3.6.1.4.1.9.1.924" sysDescr="Cisco .*(IOS-XE)"
    commandSet="CiscoASR"
className="net.linesider.overdrive.cisco.device.CiscoASR" />
  <device sysObjectID="1.3.6.1.4.1.9.1.1165" sysDescr="Cisco .*(IOS-XE)"
    commandSet="CiscoASR"
className="net.linesider.overdrive.cisco.device.CiscoASR" />
  <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.612" sysDescr="Cisco NX-
OS\(tm\) (n5000|n7000|nexus)"
    commandSet="NXOSDevice"
className="net.linesider.overdrive.cisco.device.NXOSDevice" />
  <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.777" sysDescr="Cisco NX-
OS\(tm\) (n5000|n7000|nexus)"
    commandSet="NXOSDevice"
className="net.linesider.overdrive.cisco.device.NXOSDevice" />
```

```

    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.1" sysDescr="Cisco NX-
OS\ (tm\ ) titanium"
        commandSet="NXOSDevice"
className="net.linesider.overdrive.cisco.device.NXOSDevice" />
    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.840" sysDescr="Cisco NX-
OS\ (tm\ ) .*nexus-1000v"
        commandSet="NX1KSwitch"
className="net.linesider.overdrive.cisco.device.NX1KSwitch" />
    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.719" sysDescr="Cisco NX-
OS\ (tm\ ) (n5000|n7000|nexus)"
        commandSet="NX5KOSDevice"
className="net.linesider.overdrive.cisco.device.NXOSDevice" />
    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.1084" sysDescr="Cisco NX-
OS\ (tm\ ) (n5000|n7000|nexus)"
        commandSet="NX5KOSDevice"
className="net.linesider.overdrive.cisco.device.NXOSDevice" />
    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.798" sysDescr="Cisco NX-
OS\ (tm\ ) (n5000|n7000|nexus)"
        commandSet="NX5KOSDevice"
className="net.linesider.overdrive.cisco.device.NXOSDevice" />
    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.936" sysDescr="Cisco NX-
OS\ (tm\ ) (n5000|n7000|nexus)"
        commandSet="NX5KOSDevice"
className="net.linesider.overdrive.cisco.device.NXOSDevice" />
    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.847" sysDescr="Cisco NX-
OS\ (tm\ ) ucs"
        commandSet="CiscoUCS"
className="net.linesider.overdrive.cisco.device.CiscoUCS" />
    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.899" sysDescr="Cisco NX-
OS\ (tm\ ) ucs"
        commandSet="CiscoUCS"
className="net.linesider.overdrive.cisco.device.CiscoUCS" />
    <device sysObjectID="1.3.6.1.4.1.9.1.896" sysDescr="Cisco IOS"
        commandSet="CiscoVSS"
className="net.linesider.overdrive.cisco.device.CiscoVSS" />
    <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.849" sysDescr="Cisco NX-
OS\ (tm\ ) ucs"
        commandSet="CiscoUCS"
className="net.linesider.overdrive.cisco.device.CiscoUCS" />
</devices>

```

5. Save the file and restart the controller application:

```

[root@od-c3-dsc bin]# application stop nsm-controller
[root@od-c3-dsc bin]# application start nsm-controller

```

## Device Command Modification for Specific Device Variants

<b>Note</b>	The commands in this section require shell access. At the command prompt, enter shell mode by entering <b>shell</b> and the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

This information can be used to make small adjustments to device configurations in cases where, for example, a new device version is released and a specific command has a slightly different syntax or where a specific command can be enhanced.

Do NOT use this information to attempt to change the way in which a service is configured by Network Services Manager.

## File Details

The set of commands that are executed on devices are externalized on the controller in files located in /usr/local/overdrive/controller/devices/cisco/config.

The files located here contain device configuration commands, error patterns and responses.

## Device Locator XML

CiscoDeviceLocator.xml specifies mapping from device sys description and sys objectId to the corresponding command set along with corresponding java class implementation.

```
<devices>
  <device sysObjectID="1.3.6.1.4.1.9.1.923" sysDescr="Cisco .*(IOS-XE)"
    commandSet="CiscoASR"
    className="net.linesider.overdrive.cisco.device.CiscoASR" />
  <device sysObjectID="1.3.6.1.4.1.9.1.924" sysDescr="Cisco .*(IOS-XE)"
    commandSet="CiscoASR"
    className="net.linesider.overdrive.cisco.device.CiscoASR" />
  <device sysObjectID="1.3.6.1.4.1.9.12.3.1.3.840" sysDescr="Cisco NX-
OS(tm) .*nexus-1000v"
    commandSet="NX1KSwitch"
    className="net.linesider.overdrive.cisco.device.NX1KSwitch" />
</devices>
```

## Command Set XML

The commandSet specified in the above case, is defined as catalog name in the CiscoCommandSet.xml. The command set has a list of command entries that are referred in the implementing Java class for the device. Each entry can refer to one or more command names or a single regex pattern to parse the result. Using the same entry name across devices OS version ensures that new command sets can be created and loaded for support of new versions of devices. Special execution property prompt is specified as property for the entry name; for example, for CommandSet.xml.

```
<catalog name="CiscoASR" extends="SupportedCommands">
  <entry name="assignment" command="showVlans" />
  <entry name="showOspfArea" command="showOspfArea"
    pattern="showOspfAreaPtttn" />
  <entry name="configOspfArea"
    command="setOspfProcess,setOspfArea,setBfdAllinterfaces">
    <property name="prompt" value="router" />
  </entry>
</catalog>
<catalog name="NXOSDevice" extends="SupportedCommands">
  <entry name="setVlanNo" command="setVlanNo" />
  <entry name="setVrfName" command="setVrfName" />
  <entry name="createVrf" command="createVrfCmd">
    <property name="prompt" value="vrf" />
  </entry>
</catalog>
```

For **show** commands, the result of the command execution is returned from device. When multiple commands are to be executed but the result of one command has to be returned, the 'result=' keyword is specified for the command, as in the following example:

```
<catalog name="CiscoUCS">
  <entry name="showVlanInVNIC"
command="scopeOrg,enterVnicTemplate,result=showEthernetIf,commitBuffer,end"
pattern="vlanNameInNic"/>
</catalog>
```

### Command XML

The command names refer to the commands that are defined in CiscoCommand.xml. The command names are not catalog-specific. For commands that require parameter, the parameter names are specified with a param tag.

```
<device-commands>
  <command name="config" string="show run" />
  <command name="assignment" string="show vlan brief" />
  <command name="natRules" string="show run | include static" />
  <command name="ucsconfig" string="show conf | no-more" />
  <command name="showVrfSectionInOspfCmd"
    string="show run ospf | section ospf._processNo_ | section
vrf._vrfName_"
    <param name="_processNo_" />
    <param name="_vrfName_" />
  </command>
</device-commands>
```

### Result Pattern XML

The regex patterns to parse the result are specified in CiscoResultPattern.xml. The regex patterns are also not catalog-specific. The implementation class gets the parsed values by specifying the group name.

```
<result-patterns>
  <pattern name="vrfNameInOspfPtt" expression="\s+vrf (\S+)"
    flags="MULTILINE">
    <group name="vrfName" position="1" />
  </pattern>
  <pattern name="checkOspfExistPtt" expression="router ospf (\S+)"
    flags="MULTILINE">
    <group name="ospfProcessNo" position="1" />
  </pattern>
  <pattern name="checkInterfaceExistPtt" expression="(.) is up, line
protocol is up"
    flags="MULTILINE">
    <group name="subInterface" position="1" />
  </pattern>
</result-patterns>
```



## Sample Configuration Changes

<b>Note</b>	For Network Services Manager 5.0.2, there is no formal process for upgrading and preserving the command adjustments that are made in these files. Part of the process of making these changes would include backing them up and, after an upgrade, generating a diff on the command files to merge the changes back in.
<b>CAUTION</b>	Modifying these commands could impact what is built for the tenant and must be tested extensively to verify that routing and other services are not impacted by these changes.

Before making any of the following changes:

1. SSH into the controller.
2. At the command prompt, enter shell mode by entering **shell** and then the password used to enable the shell after installation. For more information, see [Enabling the "shell" Password-Protected Access](#).
3. Navigate to `/usr/local/overdrive/controller/devices`.
4. Back up the existing command directory with the **tar** command, as in the following example:

```
tar -cvzf cisco_device_01022011.tgz cisco
```

(A date is included as an example for the filename.)

5. Navigate to `/usr/local/overdrive/controller/devices/cisco/config/`.

### Disable HSRP Preemption from HSRP Groups on the Distribution Layer (VMDC Aggregation)

By default for all tenants, Network Services Manager configures preemption for all HSRP groups. If the service provider does not want to have preemption enabled for HSRP you can modify it as follows:

1. Open the file `CiscoCommandSet.xml` and find the catalog entry for Distribution role and the related commands for configuring and unconfiguring HSRP on the Nexus 7000 device.
2. Open the file `CiscoCommand.xml` and find the commands referenced for HSRP configuration. The command is adjusted as follows, before:

```
<command name="setHsrpPreemptCmd" string="preempt"/>
```

and after:

```
<command name="setHsrpPreemptCmd" string="no preempt"/>
```

When the SVI is deleted, the HSRP groups are cleaned up so the **undo** command is not required.

## Modify the OSPF on the ASR

For purposes of illustration, assume we have decided that for the ASR we want to change the OSPF as follows:

1. Instead of an *NSSA totally stubby* area, we want to use a plain *NSSA*.
2. Advertise a default route into the area using a type 7 LSA.

This will result in permitting link state advertisements of type 3 into the area and will advertise a default route as an LSA type.

1. Open the file `CiscoCommandSet.xml` and find the catalog entry for ASR and the related commands for configuring and unconfiguring ospf on the ASR. The lines with this information are:

```
<entry name="configOspfArea"  
command="setOspfProcess,setOspfAreaASR,setBfdAllinterfaces">  
<entry name="configNoOspfArea" command="setOspfProcess,setNoOspfArea">
```

If common commands are used on various devices, the commands are shared. If a specific command is needed for a device it will have a specific name (e.g. with ARS appended).

2. The commands to set and unset the ospf area can be found in `CiscoCommand.xml` and are adjusted as shown below, before:

```
<command name="setOspfAreaASR" string="area _areaNo_ nssa no-summary">  
  <param name="_areaNo_" />  
</command>  
<command name="setNoOspfArea" string="no area _areaNo_ nssa no-summary">  
  <param name="_areaNo_" />  
</command>
```

and after:

```
<command name="setOspfAreaASR" string="area _areaNo_ nssa default-  
information-originate">  
  <param name="_areaNo_" />  
</command>  
<command name="setNoOspfArea" string="no area _areaNo_ nssa default-  
information-originate">  
  <param name="_areaNo_" />  
</command>
```

## Device Substitution with Command Differences

This feature is supported in Network Services Manager 5.0.2 as part of a Cisco Advanced Services engagement. The `sysObjectID` and `sysDesc` combination map to a Java class that represents the device library. This library consults the commands in the externalized command set shown in the examples above.

A new device with command differences will need to be accommodated as shown above where the same class is used along with command aliases on the device itself.

## Boilerplates

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

Each boilerplate has a body representing configuration to apply to a device. The contents of this body will vary across different device types.

Each boilerplate contains a filter that identifies the conditions under which the configuration in the body is applicable.

Each boilerplate has an optional order number. When multiple boilerplates are applicable, the ordered boilerplates are applied first (in increasing numerical order), and then the unordered boilerplates are applied. Ordered boilerplates must have a unique number in this file.

Each boilerplate has an optional description.

The boilerplate configuration is stored on the controller at:

```
/etc/overdrive/controller/boilerplates.xml
```

### Enable ICMP Inspection Engine on All FWSM Contexts

<b>Known Issue</b>	Due to an outstanding bug, a boilerplate designated only for a context also needs a command to be executed on the system context (section <module>) and the VSS (section <vss>).
--------------------	--

Example:

```
<boilerplate description="Enable ICMP inspection Engine on Contexts">
  <filter>
    <device-type>CiscoVSS</device-type>
    <property
key="remoteLogin.targetAddress">192.168.66.129</property>
    <action>create-context</action>
    <property key="context.roles"
match="contains">firewall</property>
  </filter>
  <body type="com.cisco.overdrive.device.boilerplate.VSSBoilerplate">
    <vss>
      # show version
    </vss>
    <module>
      # show version
    </module>
    <context>
      # conf t
      (config)# fixup protocol icmp
      (config)# exit
    </context>
  </body>
</boilerplate>
```

## Add Static Routes to Particular Device Roles at a Site

Example:

```
<boilerplate description="Add static routes for N7K-2">
  <filter>
    <action>write-routes</action>
    <property key="remoteLogin.targetAddress"
match="exactly">10.17.217.98</property>
  </filter>
  <body type="com.cisco.overdrive.routing.service.RouteXml">
    <staticRoutes>
      <staticRoute destAddr="10.3.0.0" destMask="255.255.0.0"
nextHop="145.34.56.2" />
      <staticRoute destAddr="10.3.1.0" destMask="255.255.255.0"
nextHop="145.34.56.2" />
      <staticRoute destAddr="10.4.0.0" destMask="255.255.0.0"
nextHop="145.34.56.2" />
      <staticRoute destAddr="10.4.1.0" destMask="255.255.255.0"
nextHop="145.34.56.2" />
      <staticRoute destAddr="10.17.217.0" destMask="255.255.255.0"
nextHop="145.34.56.2" />
      <staticRoute destAddr="192.0.2.1" destMask="255.255.255.255"
nextHop="145.34.56.2" />
      <staticRoute destAddr="10.1.1.0" destMask="255.255.255.0"
nextHop="145.34.56.2" />
    </staticRoutes>
  </body>
</boilerplate>
```

## Tenant Creation Dry Run


### Setup

The final step to verify the configuration is to do a dry run of a tenant VLAN (networksegment) creation. We recommend that you create a tenant "Internet Edge" zone, a "Secured Internet Edge" zone, and a VLAN (networksegment) in each zone to verify the data path configuration and service configuration for all enrolled devices.

The following procedures assume that a Provider and POD have already been created in the environment.

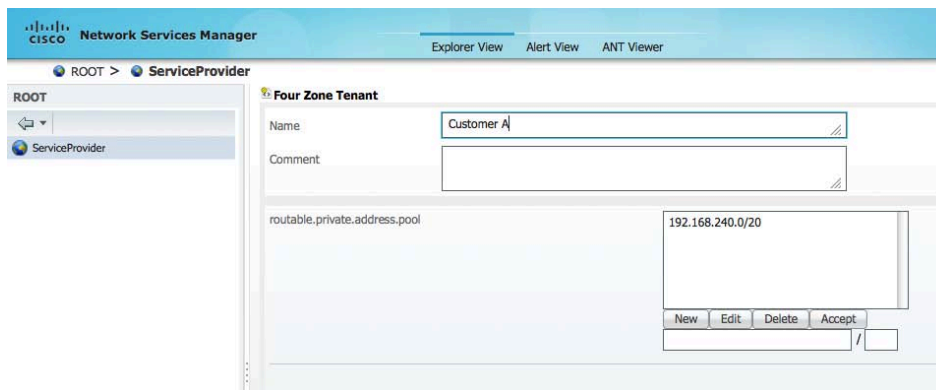
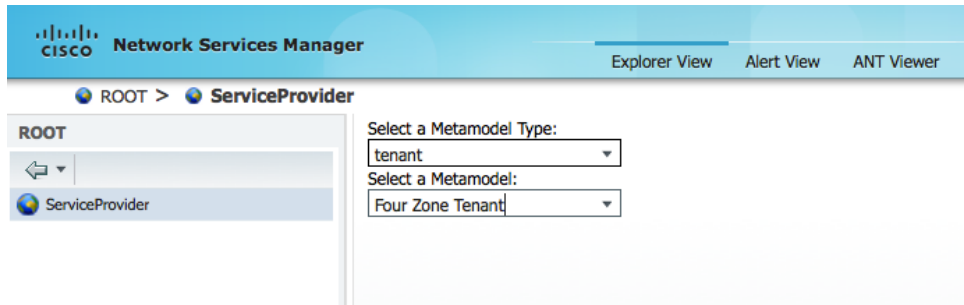
The WebUI is ideally suited for this.

### Tenant Create

1. Log into the WebUI.
2. In the navigation pane on the left, select the provider.
3. Click the Metamodel Wizard icon  .
4. In the drop-down lists, choose **tenant** and **Four Zone Tenant**, and then click **Next**.
5. Enter the information in the fields and click **Done**.

This procedure:

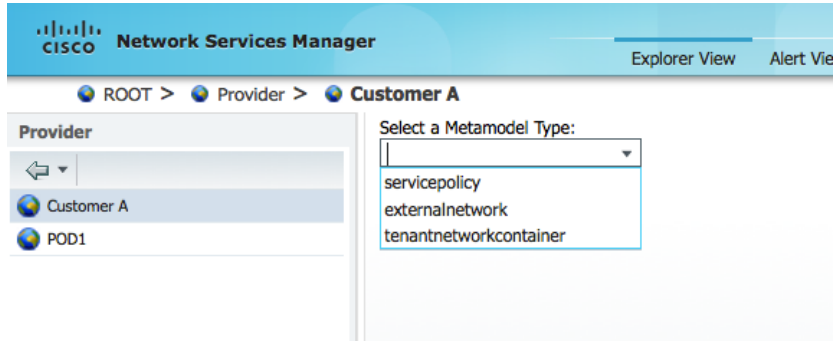
- Creates a tenant domain in the Network Services Manager engine.
- Does not configure anything on the device stack.



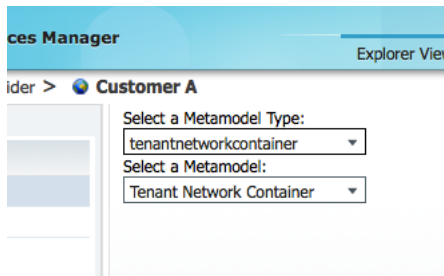
## Tenant Network Container (TNC) Creation

1. In the Provider navigation pane on the left, select the tenant, then click the **Metamodel Wizard** icon.

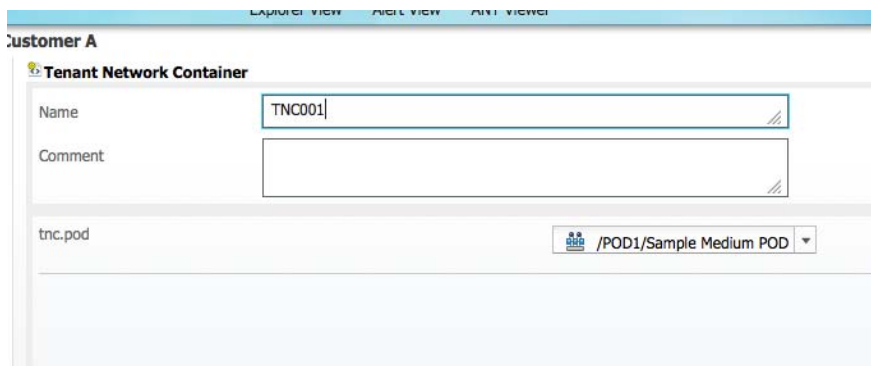
2. In the Metamodel Type drop-down list, select **tenantnetworkcontainer**.



3. In the Metamodel drop-down list, choose **Tenant Network Container**, and then click **Next**.



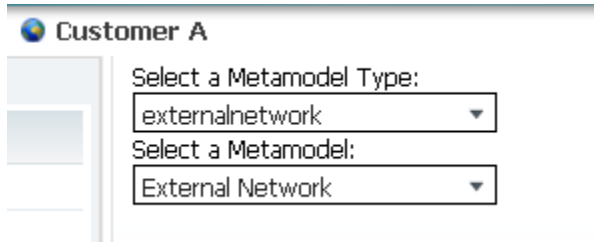
4. Enter the name of the tenant network container, choose the POD on which the tenant network container is created, and then click **Done**.



This procedure does not configure anything on the device stack.

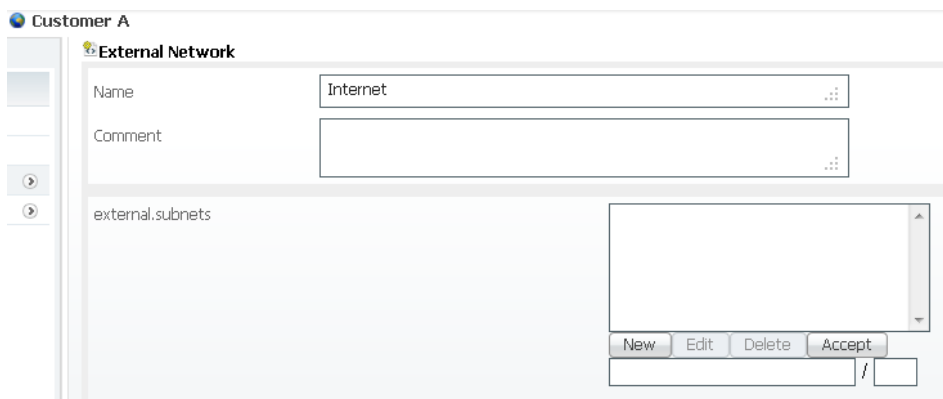
## External Network Creation

1. In the navigation pane, select the tenant, and then click the **Metamodel Wizard** icon.
2. In the drop-down lists, choose **externalnetwork** and **External Network**, and then click **Next**.



The screenshot shows the Metamodel Wizard interface for 'Customer A'. It features two dropdown menus. The first dropdown, labeled 'Select a Metamodel Type:', has 'externalnetwork' selected. The second dropdown, labeled 'Select a Metamodel:', has 'External Network' selected.

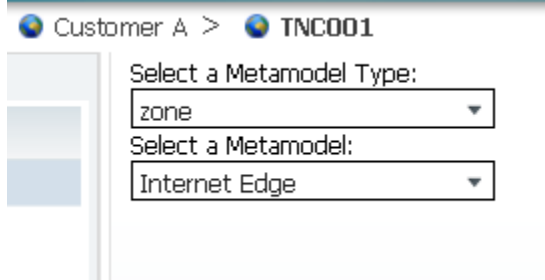
3. Enter the information for the external network, and then click **Done**.



The screenshot shows the configuration form for an 'External Network' under 'Customer A'. The 'Name' field contains 'Internet'. The 'Comment' field is empty. Below these fields is a section for 'external.subnets' with a large empty text area and buttons for 'New', 'Edit', 'Delete', and 'Accept'.

## Internet Edge Zone Creation

1. In the navigation pane, select the tenant network container that was created in [Tenant Network Container \(TNC\) Creation](#), and then click the **Metamodel Wizard** icon.
2. In the drop-down lists, choose **zone** and **Internet Edge**, and then click **Next**.

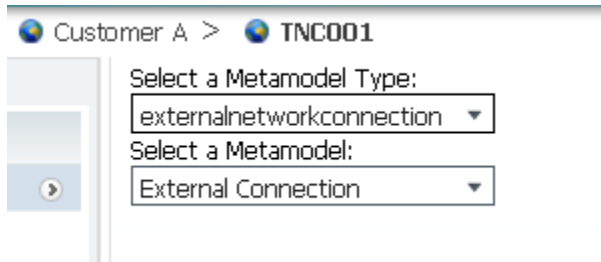


The screenshot shows the Metamodel Wizard interface for 'Customer A > TNC001'. It features two dropdown menus. The first dropdown, labeled 'Select a Metamodel Type:', has 'zone' selected. The second dropdown, labeled 'Select a Metamodel:', has 'Internet Edge' selected.

3. Enter the information for the internet edge zone, and click **Done**.

## External Network Connection

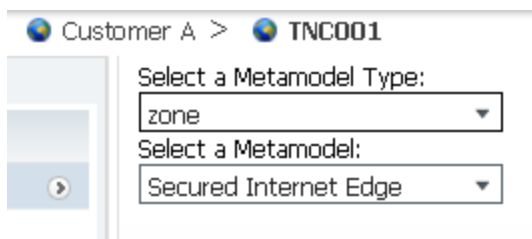
1. In the navigation pane, select the tenant network container, and then click the **Metamodel Wizard** icon.
2. In the drop-down lists, choose **externalnetworkconnection** and **External Connection**, and then click **Next**.



3. Enter the information for the external network connection, and then click **Done**.

## Secured Internet Edge Zone

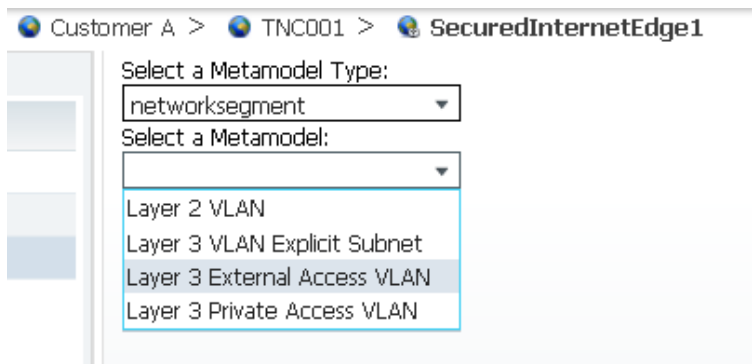
1. In the navigation pane, select the tenant network container, and then click the **Metamodel Wizard** icon.
2. In the drop-down lists, choose **zone** and **Secured Internet Edge**, and then click **Next**.



3. Enter the information for the secured internet edge zone, and click **Done**.

## VLAN in Secured Internet Edge Zone

1. In navigation pane, select the Secured Internet Edge Zone that you just created, and then click the **Metamodel Wizard** icon.
2. In the drop-down lists, choose **networksegment** and **Layer 3 External Access VLAN**, and click **Next**.



3. Enter the information for the Layer 3 external access VLAN, and click **Done**.



## Verification

1. Navigate to the VMDC POD site in the WebUI and click the **Alerts View** tab. Investigate any errors other than Unmanaged VLANs errors.
2. Review the device configurations to verify that the tenant has been provisioned through the device stack.

To do this, click on the Network Element summary view in the VMDC POD and select each network element in turn, and request a Run Command to show the configuration. Sample outputs of configurations from a created tenant are provided in the following subsections.

## ASR/Router

Network Services Manager should create subinterfaces configured with IP addresses and VRF memberships (optional), VRFs (optional), and OSPF routing processes with area and network definitions.

### Sample tenant config:

```
!  
interface Port-channel2.2002  
  encapsulation dot1Q 2002  
  ip address 172.15.0.9 255.255.255.252  
  no ip redirects  
  ip ospf priority 100  
  bfd interval 999 min_rx 999 multiplier 3  
!  
interface Port-channel3  
  description od-c2-n7k-b  
  no ip address  
  no negotiation auto  
!  
interface Port-channel3.2001  
  encapsulation dot1Q 2001  
  ip address 172.15.0.1 255.255.255.252  
  no ip redirects  
  ip ospf priority 100  
  bfd interval 999 min_rx 999 multiplier 3  
!  
router ospf 4  
  area 0.0.0.1 nssa no-summary  
  area 0.0.0.1 range 45.0.14.0 255.255.255.0  
  network 172.15.0.0 0.0.0.3 area 0.0.0.1  
  network 172.15.0.8 0.0.0.3 area 0.0.0.1  
  bfd all-interfaces  
!
```

## Nexus 7000/Distribution

Network Services Manager should create VLANs, control trunk ports to neighboring L2 switches, VLAN interfaces configured with IP addresses and HSRP groups and with virtual IP addresses, subinterfaces configured with IP addresses and VRF memberships, VRFs, OSPF routing processes with area and network definitions, static routes in created VRFs.

## Sample tenant config:

```
vrf context NH-8cef5222
 ip route 0.0.0.0/0 172.15.0.20
vrf context NH-f39971ad
 ip route 45.0.12.0/24 172.15.0.28
 ip route 45.0.15.253/32 172.15.0.28
 ip route 45.0.16.0/24 172.15.0.28
 ip route 45.0.17.0/24 172.15.0.28

vlan 2000
 name 45.0-L3PubRout-~2000
vlan 2004
 name 45.0-L3PubRout-~2004
vlan 2005
 name VLAN2005~2005
vlan 2006
 name VLAN2006~2006
vlan 2007
 name 45.0-L3PrivRout~2007
vlan 2008
 name 45.0-L3PrivRout~2008

interface Vlan2000
 no shutdown
 vrf member NH-f39971ad
 no ip redirects
 ip address 45.0.14.3/24
 ip ospf passive-interface
 hsrp version 2
 hsrp 2000
 preempt
 priority 80
 ip 45.0.14.2

interface Vlan2004
 no shutdown
 vrf member NH-8cef5222
 no ip redirects
 ip address 45.0.12.3/24
 ip ospf passive-interface
 hsrp version 2
 hsrp 2004
 preempt
 priority 120
 ip 45.0.12.2

interface Vlan2005
 no shutdown
 vrf member NH-8cef5222
 no ip redirects
 ip address 172.15.0.17/29
 ip ospf passive-interface
 hsrp version 2
 hsrp 2005
 preempt
```

```
    priority 80
    ip 172.15.0.18

interface Vlan2006
no shutdown
vrf member NH-f39971ad
no ip redirects
ip address 172.15.0.25/29
ip ospf passive-interface
hsrp version 2
hsrp 2006
    preempt
    priority 120
    ip 172.15.0.26

interface Vlan2007
no shutdown
vrf member NH-8cef5222
no ip redirects
ip address 45.0.16.3/24
ip ospf passive-interface
hsrp version 2
hsrp 2007
    preempt
    priority 80
    ip 45.0.16.2

interface Vlan2008
no shutdown
vrf member NH-8cef5222
no ip redirects
ip address 45.0.17.3/24
ip ospf passive-interface
hsrp version 2
hsrp 2008
    preempt
    priority 120
    ip 45.0.17.2

interface port-channel13
switchport
switchport mode trunk
switchport trunk allowed vlan 229,2000,2004,2007-2008
vpc 13

interface port-channel15.2002
encapsulation dot1q 2002
no shutdown
vrf member NH-f39971ad
no ip redirects
ip address 172.15.0.10/30

interface port-channel16.2003
encapsulation dot1q 2003
no shutdown
vrf member NH-f39971ad
no ip redirects
```

```

ip address 172.15.0.14/30

interface port-channel19
description od-c2-vss
switchport
switchport mode trunk
switchport trunk allowed vlan 1,2005-2006,2350
vpc 19

router ospf 4
vrf NH-f39971ad
bfd
network 45.0.14.0/24 area 0.0.0.1
network 172.15.0.8/30 area 0.0.0.1
network 172.15.0.12/30 area 0.0.0.1
area 0.0.0.1 nssa no-summary
redistribute direct route-map tenant-networks
redistribute static route-map tenant-networks

```

### VSS Pair Device Service Node

The service contexts are not accessible directly from the WebUI so you will need to log into the VSS and then session into the FWSM module to access the tenant context using the following commands.

<b>Note</b>	The FWSM reports problems when multiple users are actively configuring it. Before logging into the module, we recommend that you stop the controller.
-------------	---

```

od-c2-fwsm-a/3/act# show cont
Context Name   Class   Interfaces      Mode   URL
*admin         default Vlan229         Routed disk:/admin.cfg
49228deb719a4082b0bdd1344f8ae738 default Vlan2005,Vlan2006 Routed
disk:/49228deb719a4082b0bdd1344f8ae738.cfg

Total active Security Contexts: 2
od-c2-fwsm-a/3/act#
od-c2-fwsm-a/3/act# changeto context 49228deb719a4082b0bdd1344f8ae738
od-c2-fwsm-a/3/stby/49228deb719a4082b0bdd1344f8ae738# show run
: Saved
:
FWSM Version 4.1(7) <context>
!
hostname 49228deb719a4082b0bdd1344f8ae738
enable password 8Ry2YjIyt7RRXU24 encrypted
names
dns-guard
!
interface Vlan2006
 nameif f39971ad
 security-level 50
 ip address 172.15.0.28 255.255.255.248 standby 172.15.0.29
!
interface Vlan2005
 nameif 8cef5222

```

```

security-level 50
ip address 172.15.0.20 255.255.255.248 standby 172.15.0.21
!
passwd 2KFQnbNIdI.2KYOU encrypted
same-security-traffic permit inter-interface
access-list in_f39971ad extended permit icmp 45.0.14.0 255.255.255.0
45.0.17.0 255.255.255.0
access-list in_f39971ad extended permit tcp 45.0.14.0 255.255.255.0 45.0.17.0
255.255.255.0 eq ssh
access-list in_f39971ad extended permit icmp 45.0.14.0 255.255.255.0
45.0.12.0 255.255.255.0
access-list in_f39971ad extended permit tcp 45.0.14.0 255.255.255.0 45.0.12.0
255.255.255.0 eq ssh
access-list in_f39971ad extended permit icmp 45.0.14.0 255.255.255.0
45.0.16.0 255.255.255.0
access-list in_f39971ad extended permit tcp 45.0.14.0 255.255.255.0 45.0.16.0
255.255.255.0 eq ssh
access-list in_f39971ad extended permit tcp any 45.0.12.0 255.255.255.0 eq
www
access-list in_f39971ad extended permit tcp any 45.0.12.0 255.255.255.0 eq
https
access-list in_f39971ad extended permit icmp any 45.0.12.0 255.255.255.0
access-list in_f39971ad extended permit tcp any 45.0.12.0 255.255.255.0 eq
ssh
access-list in_8cef5222 extended permit ip 45.0.17.0 255.255.255.0 any
access-list in_8cef5222 extended permit icmp 45.0.17.0 255.255.255.0 any
access-list in_8cef5222 extended permit icmp 45.0.16.0 255.255.255.0
45.0.14.0 255.255.255.0
access-list in_8cef5222 extended permit tcp 45.0.16.0 255.255.255.0 45.0.14.0
255.255.255.0 eq ssh
access-list in_8cef5222 extended permit icmp 45.0.12.0 255.255.255.0
45.0.14.0 255.255.255.0
access-list in_8cef5222 extended permit tcp 45.0.12.0 255.255.255.0 45.0.14.0
255.255.255.0 eq ssh
access-list in_8cef5222 extended permit ip 45.0.12.0 255.255.255.0 any
access-list in_8cef5222 extended permit icmp 45.0.12.0 255.255.255.0 any
access-list in_8cef5222 extended permit icmp 45.0.17.0 255.255.255.0
45.0.14.0 255.255.255.0
access-list in_8cef5222 extended permit tcp 45.0.17.0 255.255.255.0 45.0.14.0
255.255.255.0 eq ssh
pager lines 24
mtu f39971ad 1500
mtu 8cef5222 1500
no asdm history enable
arp timeout 14400
global (f39971ad) 1 45.0.15.253 netmask 255.255.255.255
nat (8cef5222) 1 45.0.17.0 255.255.255.0
access-group in_f39971ad in interface f39971ad
access-group in_8cef5222 in interface 8cef5222
route f39971ad 0.0.0.0 0.0.0.0 172.15.0.26 1
route 8cef5222 45.0.17.0 255.255.255.0 172.15.0.18 1
route 8cef5222 45.0.16.0 255.255.255.0 172.15.0.18 1
route 8cef5222 45.0.12.0 255.255.255.0 172.15.0.18 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp 0:00:02
timeout sunrpc 0:10:00 h323 1:00:00 h225 1:00:00 mgcp 0:05:00
timeout mgcp-pat 0:05:00 sip 0:30:00 sip_media 0:02:00

```

```

timeout sip-invite 0:03:00 sip-disconnect 0:02:00
timeout pptp-gre 0:02:00
timeout uauth 0:05:00 absolute
no snmp-server location
no snmp-server contact
telnet timeout 5
ssh timeout 5
!
class-map inspection_default
  match default-inspection-traffic
!
!
policy-map global_policy
  class inspection_default
    inspect dns maximum-length 512
    inspect ftp
    inspect h323 h225
    inspect h323 ras
    inspect netbios
    inspect rsh
    inspect skinny
    inspect smtp
    inspect sqlnet
    inspect sunrpc
    inspect tftp
    inspect sip
    inspect xdmcp
    inspect icmp
!
service-policy global_policy global
Cryptochecksum:0000000000000000000000000000000000000000000000000000000000000000
: end
od-c2-fwsm-a/3/stby/49228deb719a4082b0bdd1344f8ae738#

```

## Nexus 5000

Network Services Manager should create VLANs, control trunk ports to neighboring L2 switches.

Sample tenant configuration:

```

vlan 2000
  name 45.0-L3PubRout-~2000
vlan 2004
  name 45.0-L3PubRout-~2004
vlan 2007
  name 45.0-L3PrivRout~2007
vlan 2008
  name 45.0-L3PrivRout~2008
vlan 2009
  name 45.0-L2Unrout-P~2009

interface port-channel3
  description to n7k-a+b
  switchport mode trunk

```

```

switchport trunk allowed vlan 229,2000,2004,2006-2008,2010,2014
vpc 3

interface port-channel5
description od-c2-ucs-A
switchport mode trunk
switchport trunk allowed vlan 229,2000,2003-2004,2006-2010,2014
spanning-tree port type edge trunk
vpc 5

interface port-channel6
description od-c2-ucs-B
switchport mode trunk
switchport trunk allowed vlan 229,2000,2003-2004,2006-2010,2014
spanning-tree port type edge trunk
vpc 6

```

## UCS

Network Services Manager should create VLANs, and create vNIC interfaces on specified vNIC templates.

Sample tenant config:

```

vnic-templ od-c2/vNIC_1
show eth-if|no-more
Name: 45.0-L2Unro-2009
Dynamic MAC Addr: Derived
Default Network: No

Name: 45.0-L3Priv-2007
Dynamic MAC Addr: Derived
Default Network: No

Name: 45.0-L3Priv-2008
Dynamic MAC Addr: Derived
Default Network: No

Name: 45.0-L3PubR-2000
Dynamic MAC Addr: Derived
Default Network: No

Name: 45.0-L3PubR-2004
Dynamic MAC Addr: Derived
Default Network: No

Name: VLAN229
Dynamic MAC Addr: Derived
Default Network: No

od-c2-ucs-A /org/vnic-templ #
vnic-templ od-c2/vNIC_2
show eth-if|no-more
Name: 45.0-L2Unro-2009
Dynamic MAC Addr: Derived
Default Network: No

```

```
Name: 45.0-L3Priv-2007
Dynamic MAC Addr: Derived
Default Network: No
```

```
Name: 45.0-L3Priv-2008
Dynamic MAC Addr: Derived
Default Network: No
```

```
Name: 45.0-L3PubR-2000
Dynamic MAC Addr: Derived
Default Network: No
```

```
Name: 45.0-L3PubR-2004
Dynamic MAC Addr: Derived
Default Network: No
```

```
Name: VLAN229
Dynamic MAC Addr: Derived
Default Network: No
```

```
od-c2-ucs-A /org/vnic-templ #
```

## Nexus 1000

Network Services Manager should do all of the following:

- Create VLANs
- Control the uplink port-profile
- For each VLAN, create a port-profile with the same name as the VLAN ID and designate the port-profile as a VMware port-group.

Sample configuration:

```
vlan 2000
  name 45.0-L3PubRout-~2000
vlan 2004
  name 45.0-L3PubRout-~2004
vlan 2007
  name 45.0-L3PrivRout~2007
vlan 2008
  name 45.0-L3PrivRout~2008
vlan 2009
  name 45.0-L2Unrout-P~2009

port-profile type ethernet n1kv-uplink0
  vmware port-group
  switchport mode trunk
  switchport trunk allowed vlan 229,2000,2004,2007-2009
  channel-group auto mode on mac-pinning
  no shutdown
  system vlan 229
  state enabled
port-profile type vethernet 2000
  vmware port-group
```



```
switchport mode access
switchport access vlan 2000
no shutdown
description Created by Overdrive
state enabled
port-profile type vethernet 2004
vmware port-group
switchport mode access
switchport access vlan 2004
no shutdown
description Created by Overdrive
state enabled
port-profile type vethernet 2007
vmware port-group
switchport mode access
switchport access vlan 2007
no shutdown
description Created by Overdrive
state enabled
port-profile type vethernet 2008
vmware port-group
switchport mode access
switchport access vlan 2008
no shutdown
description Created by Overdrive
state enabled
port-profile type vethernet 2009
vmware port-group
switchport mode access
switchport access vlan 2009
no shutdown
description Created by Overdrive
state enabled
```

## Troubleshooting

### Syslog Aggregation

The engine and controller both use syslog for logs so these can be aggregated on a remote server for service assurance. This will need to be developed along with search strings to help analyze the logs. This is not currently on the short term roadmap.

### Controller Setup for Remote Syslog Logging

#### Setting Up Syslog Host upon Initial Controller Configuration

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

A remote syslog host can be set up when the controller is initially set up by using the **configure** script. When configured for a remote syslog host, the controller will use the syslog facility "daemon" by default. Instructions on how to change this are included in the following section. For instructions on how to invoke the script, see the above section. A sample setup with remote syslog host set to 192.168.66.13 is captured here:

```
[root@od-11-dsc ~]# /usr/local/overdrive/controller/bin/configure -f
Network Services Manager Controller configure script
Option '-f' specified, proceeding with force of new controller instance
configuration...
Controller name? [vmdc-controller] testone
Controller password? [password]
Re-enter controller password: [password]
Engine hostname or IP address ? 1.1.1.1
Syslog host? 192.168.66.13

-----
You entered:
-----

Controller name:   testone
Controller password: password
Engine hostname:  1.1.1.1
Syslog host:      192.168.66.13

(Configuration instance name: controller1)
(Configuration instance directory: /etc/overdrive/controller1)

Press Enter to continue, or Ctrl-C to quit
Creating controller config directory: /etc/overdrive/controller1
Creating controller custom device directory: /etc/overdrive/controller1
mkdir: created directory `/etc/overdrive/controller1/custom'
mkdir: created directory `/etc/overdrive/controller1/custom/cisco'
Creating controller persistence directory:
/usr/local/overdrive/controller/data/controller1
Remember to upgrade contents of demo cert /etc/overdrive/certs.pl2 prior to
production use.
```

```

Created:
/etc/overdrive/controller1:
total 36
-rw-rw-rw- 1 root root 4295 Aug 8 13:09 agent.properties
-rw-r----- 1 root root 6360 Aug 8 13:09 boilerplates.xml
drwxrwxrwx 3 root root 4096 Aug 8 13:09 custom
-rw-rw-rw- 1 root root 2412 Aug 8 13:09 log4j.properties
-rw-rw-rw- 1 root root 1562 Aug 8 13:09 Overdrive.properties
-rw-r----- 1 root root 1648 Aug 8 13:09 ssl.properties
-rw-r----- 1 root root 363 Aug 8 13:09 staticroutes.router

/etc/overdrive/controller1/custom:
total 4
drwxrwxrwx 2 root root 4096 Aug 8 13:09 cisco

/etc/overdrive/controller1/custom/cisco:
total 0

/usr/local/overdrive/controller/data/controller1:
total 4
-rw-r----- 1 root root 3214 Aug 8 13:09 services.xml
[root@od-l1-dsc ~]#

```

## Setting Up a Syslog Host after Initial Controller Configuration

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

After the controller has been created, you can add the remote syslog server so that logs will be sent to it. The steps required to do this are:

1. Log into CARS Linux shell:

```

od-l1-dsc/admin# shell
Launching root shell...
[root@od-l1-dsc ~]#

```

2. Stop the controller process:

```

[root@od-l1-dsc ~]# /etc/init.d/nsm-controller stop
Stopping nsm-controller: controller named L1-controller [ OK ]
[root@od-l1-dsc ~]#

```

3. Use a text editor to edit the controller log configuration file:

```

[root@od-l1-dsc ~]# vi /etc/overdrive/controller/log4j.properties

```

4. Locate the following lines:

```

log4j.appender.S.syslogHost =
log4j.appender.S.facility = daemon

```

5. To set the remote syslog host to 192.168.66.13 and the logging facility to daemon (or any other facility desired), modify the file as following:

```
log4j.appender.S.syslogHost = 192.168.66.13
log4j.appender.S.facility = daemon
```

6. Save the file, and start the controller process:

```
[root@od-l1-dsc ~]# /etc/init.d/nsm-controller start
Starting nsm-controller: controller named L1-controller [ OK ]
[root@od-l1-dsc ~]#
```

## Engine Setup for Remote Syslog Logging

### Setting Up Syslog Host on a Running Engine

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

The engine does not have an option of setting up remote syslog host logging upon setup, thus you can do this only after the initial setup has been run.

1. Login to CARS Linux shell:

```
od-l1-nsve/admin# shell
Launching root shell...
[root@od-l1-nsve ~]#
```

2. Stop the engine process:

```
[root@od-l1-nsve ~]# /etc/init.d/nsm-engine stop
Waiting 30 seconds for application (29176) to stop..... stopped.
[root@od-l1-nsve ~]#
```

3. Use a text editor to edit the engine log configuration file:

```
[root@od-l1-nsve ~]# vi
/usr/local/jboss/server/wpserver/conf/wplog4j.properties
```

4. Add the proper "appender" S to the root category:

```
log4j.logger.com.pfn = INFO, R, S
```

5. Locate the following for lines:

```
log4j.appender.S.syslogHost = localhost
log4j.appender.S.facility = LOCAL6
```

- To set the remote syslog host to 192.168.66.13 and the logging facility to LOCAL6 (or any other facility desired), modify the file as follows:

```
log4j.appender.S.syslogHost = 192.168.66.13
log4j.appender.S.facility = LOCAL6
```

- Save the file, and start the engine process:

```
[root@od-11-nsve ~]# /etc/init.d/nsm-engine start
JBOSS_CMD_START = cd /usr/local/jboss/bin; /usr/local/jboss/bin/run.sh -c
wpserver -b 0.0.0.0
Application is running as PID 29418
[root@od-11-nsve ~]#
```

## Controller Not Connecting to the Server

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

It is important that you look for the top-most log error or exception to troubleshoot this issue. For example, a certificate installation problem will also cause a timeout exception connecting to the server so you might think it is a server reachability issue when it is a certificate issue. A good strategy is to start at the top of the log file and search forward for "exception" or "error". In vi, */s\exception* (case insensitive).

<b>Note</b>	With Network Services Manager 5.0.2, the controller setup default values specify default information for controller name and password based on the business model provided with Network Services Manager. If the default values were not chosen during installation, the best solution is to rerun the configure script on the controller.
-------------	--

- SSH onto the controller and log in as admin.
- Reset the controller to factory default.
- Enter **application nsm-controller reset-config**.
- Enter **shell** to get to a root shell.
- Enter **/usr/local/overdrive/controller/bin/configure**.

## Gathering Logs and Technical Support information for Cisco TAC

### Configuring the Disk Repository

Perform this one-time procedure to create a backup archive file for your engine or controller in preparation for making this information available to Cisco:

- Open a console window to the engine VM or, if SSH is enabled, SSH to the VM.
- Log in as user admin.

3. Enter **config** to enter configuration mode.
4. Enter **repository disk** to define a repository.
5. Enter **url disk:**.
6. Enter **end** to exit configuration mode.
7. Enter **write memory** to save these changes for subsequent backups.
8. If desired, test the repository's definition with the **show running-config** command; for example:

```
show running-config | include repository
```

If you see "repository disk" in the output, then the preparatory step has been successful.

An example of the above commands follows:

```
cnh-engine/admin# config
Enter configuration commands, one per line. End with CNTL/Z.
cnh-engine/admin(config)# repository disk
cnh-engine/admin(config-Repository)# url disk:
cnh-engine/admin(config-Repository)# end
cnh-engine/admin# write memory
Generating configuration...
cnh-engine/admin# show running-config | include repository
repository disk
cnh-engine/admin#
```

## Capturing Data

The following three shell commands are used on both the engine and controller to capture information for Cisco Technical Support:

- [tech-support](#)
- [backup-logs](#)
- [backup](#)

After you generate the output of these commands, you can contact Cisco and forward the files as required. For more information on sending these files to Cisco, see [Sending Data to Cisco](#).

### tech-support

The format of the **tech-support** command is:

```
show tech-support file filename
```

where *filename* is a unique filename.

We recommend that you include the hostname and the string "\_support-" and date in the filename. This recommendation applies only to this command. The other two commands include this information automatically.

For example:

```
show tech-support file myengine_support-20121212
```

This command creates output to the console similar to the following. The **dir** command at the end lists the contents of the local "disk" repository:

```
cnh-carscontroller/admin# show tech-support file cnh-carscontroller_support-20121212
nsm-controller tech-support output started at Thu Dec 13 00:30:48 UTC 2012
nsm-controller tech-support output finished at Thu Dec 13 00:31:01 UTC 2012
cnh-carscontroller/admin# dir

Directory of disk:/

      51393 Dec 13 2012 00:31:01  cnh-carscontroller_support-20121212.tar.gz
      2432 Dec 05 2012 19:11:02  log4j.properties
     16384 Dec 05 2012 18:55:32  lost+found/

      Usage for disk: filesystem
                146919424 bytes total used
                5617405952 bytes free
                6078058496 bytes available
cnh-carscontroller/admin#
```

## backup-logs

The format of the backup-logs command is:

```
backup-logs <backup-name> repository <repository-name> encryption-key plain
<plain-text-key>
```

We recommend that you include the hostname and the string "\_logs" in the backup-name; for example:

```
backup-logs cnh-carscontroller_logs repository disk encryption-key plain
NSMforTAC123
```

An example with console output follows, with the dir command output showing the created file:

```
cnh-carscontroller/admin# backup-logs cnh-carscontroller_logs repository disk
encryption-key plain NSMforTAC123
% Creating log backup with timestamped filename: cnh-carscontroller_logs-
121213-0033.tar.gpg
cnh-carscontroller/admin# dir

Directory of disk:/

     1485259 Dec 13 2012 00:33:29  cnh-carscontroller_logs-121213-0033.tar.gpg
      51393 Dec 13 2012 00:31:01  cnh-carscontroller_support-20121212.tar.gz
```

```
2432 Dec 05 2012 19:11:02 log4j.properties
16384 Dec 05 2012 18:55:32 lost+found/
```

```
Usage for disk: filesystem
148410368 bytes total used
5615915008 bytes free
6078058496 bytes available
```

```
cnh-carscontroller/admin#
```

## backup

The format of the **backup** command is:

```
backup <backup-name> repository <repository-name> encryption-key plain  
<plain-text-key>
```

We recommend that you include the hostname and the string "\_data" in the backup name; for example:

```
backup cnh-carscontroller_data repository disk encryption-key plain  
NSMforTAC123
```

An example with console output follows, with the **dir** command output showing the created file:

```
cnh-carscontroller/admin# backup cnh-carscontroller_data repository disk  
encryption-key plain NSMforTAC123  
% Creating backup with timestamped filename: cnh-carscontroller_data-121213-  
0035.tar.gpg  
cnh-carscontroller/admin# dir  
  
Directory of disk:/  
  
4685 Dec 13 2012 00:35:47 cnh-carscontroller_data-121213-0035.tar.gpg  
1485259 Dec 13 2012 00:33:29 cnh-carscontroller_logs-121213-0033.tar.gpg  
51393 Dec 13 2012 00:31:01 cnh-carscontroller_support-20121212.tar.gz  
2432 Dec 05 2012 19:11:02 log4j.properties  
16384 Dec 05 2012 18:55:32 lost+found/  
  
Usage for disk: filesystem  
148418560 bytes total used  
5615906816 bytes free  
6078058496 bytes available  
cnh-carscontroller/admin#
```

## Sending Data to Cisco

Use the **copy** command to copy the three files created in the [Capturing Data](#) section to an external host or a URL.

The **copy** command is run from a console session and uses the format:

```
copy <disk:/path/file> <targetURL>
```



In the following example:

- The **dir** command confirms the presence and names of the files to be copied from the VM.
- SFTP prompts for a username and password, which are entered interactively.

```
cnh-carscontroller/admin# dir
Directory of disk:/

   4685 Dec 13 2012 00:35:47  cnh-carscontroller_data-121213-0035.tar.gpg
 1485259 Dec 13 2012 00:33:29  cnh-carscontroller_logs-121213-0033.tar.gpg
   51393 Dec 13 2012 00:31:01  cnh-carscontroller_support-20121212.tar.gz
   2432 Dec 05 2012 19:11:02  log4j.properties
  16384 Dec 05 2012 18:55:32  lost+found/

Usage for disk: filesystem
      148418560 bytes total used
      5615906816 bytes free
      6078058496 bytes available
cnh-carscontroller/admin# copy disk:/cnh-carscontroller_data-121213-0035.tar.gpg sftp://od-build-d20/tmp
Username: odbuild
Password:
cnh-carscontroller/admin# copy disk:/cnh-carscontroller_logs-121213-0033.tar.gpg sftp://od-build-d20/tmp
Username: odbuild
Password:
cnh-carscontroller/admin# copy disk:/cnh-carscontroller_support-20121212.tar.gz sftp://od-build-d20/tmp
Username: odbuild
Password:
cnh-carscontroller/admin#
```

### Controller Name/Password Not Configured as Specified in the Server

The following log entry shows a controller that has the correct name, but the password is set incorrectly. Not authenticated signifies that it has found the message queue but is being refused connection as a result of authentication failure.

```
Controller_160_54 []\-WARN agent.Controller - Failed to execute state `JMS
initialization'
: User: Controller_95_160_95_54 is NOT authenticated
```

### Engine Name/IP Address Incorrect on the Controller Configuration

The following log entry shows a controller with an incorrect engine name or IP address, or an engine that is not reachable due to a firewall:

```
Controller_160_54 []\-WARN agent.Controller - Failed to execute state `JMS
initialization'
: Could not obtain connection to any of these urls: 172.26.160.53:11099 and
discovery failed
with error: javax.naming.CommunicationException: Receive timed out
```

```
[Root exception is java.net.SocketTimeoutException: Receive timed out]
```

## Failure to Connect to the Devices (auth/SNMP etc.)

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

The first thing to establish is whether this is a credentials problem or an issue with the device not being included in the library of supported devices.

The devices in the stack are all supported for this deployment but the controller establishes a device identity by pulling the sysDescr and sysObjectID from SNMP.

Thus, if SNMP is not correctly set up it will fail to identify the device.

The following controller log entry will confirm this:

- Verify SNMP v3 access with the following command from a machine with snmpwalk:

```
snmpwalk -v 3 -a MD5 -A cisco123 -u admin -l authPriv -x DES -X cisco123  
<host-ip> sysDescr
```

where *cisco123* is the auth password and priv phrase.

- Verify SNMP v2c access with the following command:

```
snmpwalk -v 2c -c overdrive 172.23.39.34
```

where *overdrive* is the RW community.

Verify that you have correctly set the credentials by navigating to the device in the UI and inspecting the credentials there.

**Note** You cannot verify passwords in the UI. If needed, you can set new passwords in the WebUI.

## Failure to Connect to the Devices (sysDescr and/or sysObjectID)

The controller recognizes the devices by checking SNMP MIBs under sysObjectID and sysDesc. If the device sysObjectID and SysDesc do not match those of a device saved in the controller's supported list, the controller cannot recognize the device and refuses to connect to it. If you are confident that the given device uses the same CLI commands as the supported one, you can update the controller to support the device. Use the following steps to enable the user to edit the list of devices recognized by the controller:

If the device has CLI parity with the one of the supported devices in this role, you can follow the instructions in [Substituting a Like Device for an Officially Supported Device](#).

## Controller Fails to Identify a Device (SNMP READ Access Misconfiguration)

The example bellow shows a how to identify a misconfigured SNMP v3 username configured on an ASR 1004 router

### WebUI

Alert View				
Reporting Object	Alert	Description	Severity	When
/ROOT/VMDC_POD_C2/ASR-2	Alert	[DeviceProblem message='Error obtaining SNMP device info: Invalid SNMP authorization information for 192.168.66.121...	Error	1
				Wed Feb 01 2012 10:55:49 GMT-0500 (EST)

### Check the Engine Logs for Errors

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

Examine the logfile's contents around the time when the failure occurred, looking for messages received from the controller that report alarming about a failure:

```
2012-02-01 10:55:49,233 INFO [impl.PepEventListenerBean] Received event from C2-agent type=DSC Problem
2012-02-01 10:55:49,235 INFO [impl.PepEventListenerBean] C2-agent: problem event: Problem(raise: "[DeviceProblem message='Error obtaining SNMP device info: Invalid SNMP authorization information for 192.168.66.121/161', device=ASR-2 (d847407ba0c94b37b55b8ee7565dc8d1) ]")
```

### Check the Controller Logs for Errors

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

1. Enter shell mode and use a text editor to view the controller log file:

```
vi /var/log/overdrive*.log
```

2. Examine the logfiles contents around the time when the failure occurred, looking for a line similar to the following:

```
C2-agent 2012-02-01 10:55:49,225 [ool thread 45]-DEBUG DeviceDiscovery -
Device ASR-2 Exception message: Error obtaining SNMP device info: Invalid
SNMP authorization information for 192.168.66.121/161
C2-agent 2012-02-01 10:55:49,225 [ool thread 45]-DEBUG DeviceDiscovery -
Failed to connect to device ASR-2, exception details:
net.linesider.overdrive.device.locator.DeviceConnectException: Error
obtaining SNMP device info: Invalid SNMP authorization information for
192.168.66.121/161
    at
net.linesider.overdrive.agent.device.DeviceDiscovery.getDeviceInfo(DeviceD
iscovery.java:82)
    at
net.linesider.overdrive.agent.device.DeviceDiscovery.getDevice(DeviceDisco
very.java:107)
    at
com.pfn.wirepower.pep.device.DeviceConfigHandler.addDevice(DeviceConfigHan
dler.java:392)
    at
com.pfn.wirepower.pep.device.DeviceConfigHandler.access$400(DeviceConfigHa
ndler.java:53)
    at
com.pfn.wirepower.pep.device.DeviceConfigHandler$AddDevice.call(DeviceConf
igHandler.java:445)
    at
com.pfn.wirepower.pep.device.DeviceConfigHandler$AddDevice.call(DeviceConf
igHandler.java:438)
    at java.util.concurrent.FutureTask$Sync.innerRun(FutureTask.java:303)
    at java.util.concurrent.FutureTask.run(FutureTask.java:138)
    at
net.linesider.overdrive.agent.concurrent.SerialExecutionService$Work.run(S
erialExecutionService.java:273)
    at
net.linesider.overdrive.agent.concurrent.SerialExecutionService$WorkQueueR
unner.run(SerialExecutionService.java:350)
    at
net.linesider.overdrive.agent.concurrent.Recurrent$1.run(Recurrent.java:21
)
    at
java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:441)
    at java.util.concurrent.FutureTask$Sync.innerRun(FutureTask.java:303)
    at java.util.concurrent.FutureTask.run(FutureTask.java:138)
    at
java.util.concurrent.ThreadPoolExecutor$Worker.runTask(ThreadPoolExecutor.
java:886)
```

```
at
java.util.concurrent.ThreadPoolExecutor$Worker.run (ThreadPoolExecutor.java
:908)
at com.pfn.wirepower.pep.agent.Agent$2$1.run (Agent.java:611)
at java.lang.Thread.run (Thread.java:662)
```

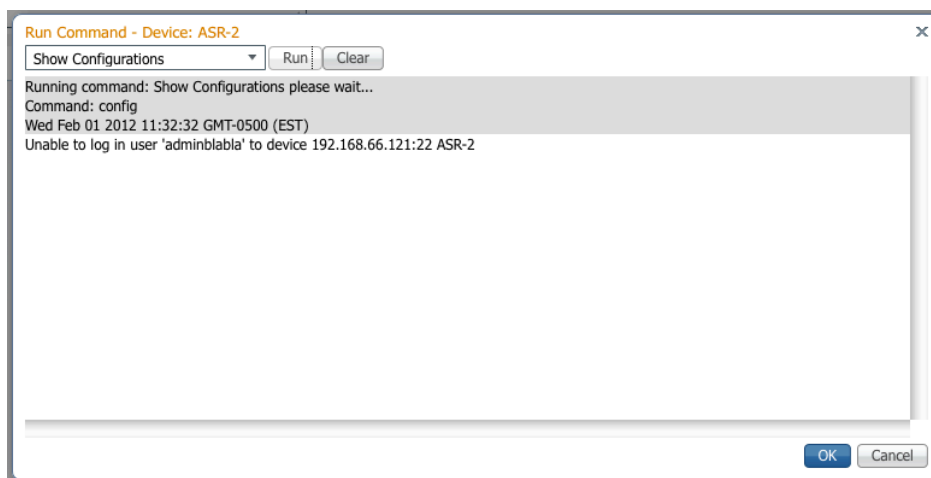
## Controller Fails to Log into a Device (CLI Credentials)

The example below shows a how to identify a misconfigured SSH username configured on an ASR 1004 router

### WebUI

**Known Issue** The controller does not report SSH login failures to devices until a VLAN is provisioned.

To work around the issue, use the WebUI to navigate to the VMDC POD, and use the **Run Commands** option to run a command on each device. Depending on the POD's settings, this command will trigger an SSH or Telnet login to the device, as shown in the following figure:



### Check the Engine Logs for Errors

The controller will not report CLI (SSH or Telnet) access failures prior to provisioning to the engine; thus, CLI login failures will not be reported in the logs.

## Check the Controller Logs for Errors

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

Examine the logfile contents for the time when the failure occurred, looking for a line similar to the following:

```
C2-agent 2012-02-01 11:32:00,157 [_45_agent-pep]-INFO MessageHandler -
Sending device command: show run
C2-agent 2012-02-01 11:32:00,158 [_45_agent-pep]-DEBUG CmdUtil      - send
command for config timeOut = 360
C2-agent 2012-02-01 11:32:00,159 [_45_agent-pep]-INFO AbstractCommandHandler
- Log in to device's ssh server: 192.168.66.121
C2-agent 2012-02-01 11:32:02,599 [_45_agent-pep]-DEBUG AbstractCommandHandler
- Failed to login to device: Unable to log in user 'adminblabla' to device
192.168.66.121:22
com.pfn.wirepower.pep.connector.CommandHandlerLoginException: Unable to log
in user 'adminblabla' to device 192.168.66.121:22
    at
com.pfn.wirepower.pep.connector.GanymedWrapper.open (GanymedWrapper.java:38)
    at
com.pfn.wirepower.pep.connector.AbstractCommandHandler.establishConnection (Ab
stractCommandHandler.java:153)
    at
net.linesider.overdrive.cisco.connector.CommandHandler.establishConnection (Co
mmandHandler.java:113)
    at
net.linesider.overdrive.cisco.connector.IOSCommandHandler.establishConnection
(IOSCommandHandler.java:27)
    at
com.pfn.wirepower.pep.connector.AbstractCommandHandler.getConnectionWrapper (A
bstractCommandHandler.java:119)
    at
com.pfn.wirepower.pep.connector.AbstractCommandHandler.send (AbstractCommandHa
ndler.java:223)
    at
com.pfn.wirepower.pep.connector.AbstractCommandHandler.getSysname (AbstractCom
mandHandler.java:528)
    at
net.linesider.overdrive.cisco.connector.CommandHandler.setPrompt (CommandHandl
er.java:241)
    at
com.pfn.wirepower.pep.connector.AbstractCommandHandler.sendUserCommand (Abstra
ctCommandHandler.java:441)
    at
com.pfn.wirepower.pep.device.MonitoredCommandHandler$9.call (MonitoredCommandH
andler.java:173)
    at
com.pfn.wirepower.pep.device.MonitoredCommandHandler$9.call (MonitoredCommandH
andler.java:171)
```

```

    at
com.pfn.wirepower.pep.device.MonitoredCommandHandler.issue (MonitoredCommandHa
ndler.java:36)
    at
com.pfn.wirepower.pep.device.MonitoredCommandHandler.sendUserCommand (Monitore
dCommandHandler.java:171)
    at
net.linesider.overdrive.cisco.cmd.CmdUtil.sendCommandWithTimeOut (CmdUtil.java
:414)
    at
net.linesider.overdrive.cisco.device.CiscoDevice.executeCommand (CiscoDevice.j
ava:608)
    at
com.pfn.wirepower.pep.agent.MessageHandler$GetSwitchStatus.doAction (MessageHa
ndler.java:1385)
    at
com.pfn.wirepower.pep.agent.MessageHandler$RequestAction.doAction (MessageHand
ler.java:561)
    at
com.pfn.wirepower.pep.agent.MessageHandler.onMessage (MessageHandler.java:294)
    at org.jboss.mq.SpyMessageConsumer.run (SpyMessageConsumer.java:696)
    at java.lang.Thread.run (Thread.java:662)

```

## Traffic Not Flowing Through Data Path

If there are no more errors in the controller logs or the Command Center, but the VMs still do not have network connectivity (that is, they cannot ping the default gateway), check the health status of the data path links (interconnects). Log into each network device that is deployed and refer to [Base Configuration](#) for more information about the base config for each device.

Also, make sure that the ports that are configured for data path connectivity are in STP Forwarding state. If one of the ports is in Blocking or stuck in Learning state, resolve the L2 loop condition and ensure that the port successfully transitions into Forwarding state.

## Management Network

If the network setup involves a firewall between the engine and controller appliances, ensure that the following ports are permitted through the firewall to enable successful communication between the engine and the controller:

Port	Protocol	Client-Engine	Controller-Engine	Service Description
8094	TCP		Y	org.jboss.mq.il.uil2.UILEngineILService
8095	TCP	Y		org.jboss.mq.il.uil2.UILEngineILService
8443	TCP	Y		org.jboss.web.WebService (SSL)
11098	TCP	Y	Y	org.jboss.naming.NamingService
11099	TCP	Y	Y	org.jboss.naming.NamingService

The connection to the engine is initiated by the controller, so the controller can be also placed behind a NAT device. Although the communication between the appliances is established over SSL, both appliances are capable of using a software firewall (iptables) to implement even tighter security measures.

The devices in the stack to be managed need to have management interfaces defined. All devices will need to have base configurations that give both CLI and SNMP (pref v3) access.

The following details are captured for additional settings in the base configurations.

## Engine

### Log Files

<b>Note</b>	The commands in this section require shell access. Enter shell mode by entering <b>shell</b> at the command prompt and entering the password used to enable the shell after installation. For more information, see <a href="#">Enabling the "shell" Password-Protected Access</a> .
-------------	--

```
/usr/local/jboss/server/wpserver/log/
```

To enable Debug mode logging, log into CARs CLI:

```
od-11-nsve/admin# conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
od-11-nsve/admin(config)# logging loglevel 7  
od-11-nsve/admin(config)#
```

### Services

```
/etc/init.d/nsm-engine (start\|stop\|restart\|status)
```

### Reset to Factory Default

**Note** This script removes all configuration and logs on the system and restores it to an initial state.

```
/usr/local/overdrive/engine/bin/reset-config.sh
```

When you reset the engine, you need to stop the controller until you have redefined the POD in the engine. Use the following command to stop and then start the controller:

```
/etc/init.d/nsm-controller [start|stop|status|restart]
```

**Note** If you reset the engine and controller in a configured environment, the existing tenant configurations on the devices will be left in place and orphaned. Manual cleanup of the devices is also required.



Complete the following tasks to ensure that all aspects of resetting the engine and controller to factory default values are addressed:

1. Stop and reset the engine by using the script `/usr/local/overdrive/engine/bin/reset-config.sh`.
2. Stop and reset the controller by using the script `/usr/local/overdrive/controller/bin/reset-config.sh`.
3. Manually reset or clean device configurations.
4. Configure the controller to connect to the engine by using the script `/usr/local/overdrive/controller/bin/configure`.

## Controller

### Log Files

log4j config file: `/etc/overdrive/<controller-name>/log4j.properties`

```
/var/log/overdrive-<controller-name>.log
```

To adjust debug level, log into CARS CLI:

```
od-11-dsc/admin# conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
od-11-dsc/admin(config)# logging loglevel 7  
od-11-dsc/admin(config)#
```

### Config Files

```
/etc/overdrive/controller/  
/usr/local/overdrive/controller/  
/usr/local/overdrive/controller/bin/  
/usr/local/overdrive/controller/config-templates  
/usr/local/overdrive/controller/data/controller
```

### Service

```
/etc/init.d/nsm-controller [start|stop|status|restart]
```

## API

The Network Services Manager 5.0.2 API is described in [Cisco Network Services Manager 5.0.2 NB API Specification and Reference Guide](#).

## Task Information

New in Network Services Manager 5.0.2 is an ability to determine when NBI requests have been operationally fulfilled by the completion of all required device configurations. Network Services Manager is "state-based" rather than "task-based", and has the following implications on how taskStatus is reported.

When an NBI request is successfully received by Network Services Manager, the business model is updated, and device configurations are subsequently attempted to be brought into conformance with this updated model. The Task Resource that is created for this NBI request holds both status and result information which is updated by Network Services Manager as progress is made in the configuring of devices. The taskStatus is a synthesis of the progress of the multiple devices operations that may be required to fulfill a single NBI request. The task resource may be polled via the NBI "get" operation obtain the most recent information about the original request's progress.

This "taskStatus" within the task resource will be one of the following values:

- Pending – the requested operation was successfully received and queued for processing by Network Services Manager, and not all device configuration updates have completed. The NBO should periodically poll the task resource for updated information.
- Success – the requested operation was successfully processed by Network Services Manager and all device configurations have been updated correctly.
- Failure – the requested operation was successfully processed by Network Services Manager, but one or more device configurations failed to update correctly. Any such failure is an exception to normal processing, and indicates a condition that should be investigated by a system administrator.

When a failure is reported, the original API request will continue to remain active. Network Services Manager retains knowledge of the requested provisioning operation and will continually retry the operation until successful device configuration has been achieved.

The intent of the system is to always bring device configurations into the state needed to fulfill the original NBI request, that is, to achieve "Success" status. Success should be automatically achieved when all underlying failure causes, "faults", are corrected or removed. E.g. if a device update is failing because a device is offline, bringing it online should correct that particular fault.

The "faults" section of the task resource in "Failure" status contains all the problem(s) that Network Services Manager is detecting, which have prevented the original NBI request from succeeding. While these may or may not require manual intervention on the part of the system administrator to correct, we recommend that the NBO should not process additional requests until the failure reasons have been assessed.

Thus, in a case where a device may be temporarily not responding, the reported problem may be able to fully correct itself without intervention. It remains the responsibility, however, of the NBO to handle and alert the system administrator to reported failures, in the event that manual intervention is required.

Failure status will be perpetually reported until the underlying causes system is corrected, OR, the NBO chooses to "roll back" the original API request, such as requesting Network Services Manager to "delete" the object of a failing "create" request.

If there is any doubt as to which problems are present within the system, the Alerts view in the WebUI can be reliably consulted to determine the presently reported problems.

# Release Notes/Known Issues

## IP Validation of Service Filters

When you specify a subnet or specific set of addresses, the Network Services Manager engine validates that the specified address is a subset of the addresses provided for the External Network. For example, if your external network is 10.0.0.0/16, you will not be permitted to use an address outside of that range for fine-grained rules.

This validation is not performed in a Zone; addresses not specifically assigned to a zone can be used in fine-grained traffic filters.

## Service Policies Between TNCs

Support for service policies that span TNCs is not available in Network Services Manager 5.0.2. To create connections between customer resources in different TNCs you must create two separate service policies (one in each TNC) and use a remote network for the other side of the policy. For example, if you have a resource in TNC A and TNC B that you want to allow to communicate, you would create a service policy in TNC A to allow the local resource and a remote network containing the subnet data for the resource in TNC B. In TNC, you would do the reverse.

## Active Topology

The active topology contains all of the information relative to the persisted object model and relationships. This information is available in JSON via the REST V2 API but not exposed via the NB API at this time.

A visual tool or summary tool for pulling out relevant information from the topology model is highly desirable.

## POD Enablement

Enabling a POD requires close attention to detail. There has been some work in the metamodel schema to make it possible to create a metamodel for setting up network elements and the topology model (permitting an NB API call for this) but this is not currently on the short-term roadmap.

A tool for generating the JSON import file from a visual depiction of the stack required. This is not currently on the short-term roadmap.

## Backup/Restore

Using VMware backup tools, such as Snapshots and VMware Data Recovery (VDR), the snapshot might be out of synch with the controller and devices, so what should our prescribed steps be?

The recommended actions for restore are:

1. Stop the controller.
2. Reload all of the devices.

If you saved the device configurations after Network Services Manager configured them, you will need to manually reset the devices to clean starting configurations.

3. Restore the engine.
4. Reset and reconfigure the controller.
5. Make sure all of the devices are online.
6. Start the controller.

## **Redundancy of the Engine and Controller**

Currently only VMware HA features will be used to support this.

# Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Cisco Network Services Manager 5.0.2 Technical Manual

© 2012, 2013 Cisco Systems, Inc. All rights reserved.