Avaya Solution & Interoperability Test Lab

Juniper Networks EX-Series Virtual Chassis with Avaya Aura® Telephony Infrastructure – Issue 1.0

Abstract

These Application Notes describe the steps for configuring Juniper Networks EX 4300 and EX 9200 Ethernet Switches as a Virtual Chassis. The configuration includes 802.1x Authentication, Link Layer Discovery Protocol – Media Endpoint Discovery (LLDP-MED), Quality of Service (QoS) and Power over Ethernet (PoE) implemented an Avaya Aura® Telephony Infrastructure.

Information in these Application Notes has been obtained through interoperability compliance testing and additional technical discussions. Testing was conducted via the DevConnect Program at Avaya Solution and Interoperability Test Lab.

1. Introduction

These Application Notes describe a compliance-tested solution using Juniper Networks EX 4300 and EX 9200 Ethernet switches in a Virtual Chassis environment. Integration of an Avaya Aura® Telephony Infrastructure including Avaya Aura® System Manager, Avaya Aura® Session Manager, Avaya Aura® Communication Manger, Avaya Aura® Communication Manager Messaging, Avaya G450 Media Gateway and various Avaya endpoints were used to validate the solution.

Avaya Aura® Infrastructure components were connected to the EX9200 virtual chassis and the endpoints to the EX4300 virtual chassis. Quality of Service (QoS), Link Layer Discovery Protocol–Media Endpoint Discovery (LLDP-MED), 802.1x Authentication and Power over Ethernet (PoE) were implemented in the network and validated for interoperability.

FreeRADIUS was used to provide 802.1X RADIUS authentication for Avaya IP Telephones and the PCs that are connected to the Virtual Chassis switch. The Avaya IP Telephones and PCs are individually authenticated via communication between the Virtual Chassis Switch and the RADIUS Server using port level multiple supplicant support on the Virtual Chassis.

2. General Test Approach and Test Results

The general test approach was to verify interoperability between Juniper Networks Virtual Chassis Ethernet Switches with Avaya endpoints functioning in an Avaya Aura® Telephony Infrastructure. All test cases were executed manually.

DevConnect Compliance Testing is conducted jointly by Avaya and DevConnect members. The jointly-defined test plan focuses on exercising APIs and/or standards-based interfaces pertinent to the interoperability of the tested products and their functionalities. DevConnect Compliance Testing is not intended to substitute full product performance or feature testing performed by DevConnect members, nor is it to be construed as an endorsement by Avaya of the suitability or completeness of a DevConnect member's solution.

2.1. Interoperability Compliance Testing

The interoperability compliance testing included the following:

All test cases were performed manually.

- LAN connectivity between Avaya and Juniper Networks products.
- Registration of Avaya H.323 endpoints with Communication Manager.
- Registration of Avaya SIP endpoints with Session Manager.
- VoIP calls, including, hold, transfer and conferencing.
- QoS for voice signaling and voice media received higher priority based on 802.1p and DSCP settings.
- Configuration and Auto discovery of QoS parameters using LLDP-MED
- Configuration and Auto discovery of Voice VLAN using LLDP-MED
- Avaya Aura® Communication Manager Messaging voicemail and MWI works properly.
- 802.1x Authentication of Avaya IP Telephones and Personal Computers running windows 7.
- Power over Ethernet (PoE) for Avaya IP Telephones

Compliance testing focused on QoS, VLAN, 802.1x, and PoE implementation in the Avaya/Juniper Networks configuration. Specifically, compliance testing verified that when the Juniper Networks switch interfaces were oversubscribed with low priority data traffic, the higher priority VoIP media and signaling traffic still got through and achieved good voice quality. Prioritization of voice traffic was achieved by implementing Layer 3 DiffServ-based QoS and Layer 2 priority (801.p). Voice and data traffic were segmented in the enterprise network using VLANs. Auto discovery of Voice VLAN, and QoS parameters, using LLDP-MED were also verified along with the ability to power Avaya IP telephones via PoE.

2.2. Test Results

The Juniper Networks EX 4300 and EX9200 Ethernet Switches successfully achieved the above objectives and passed compliance testing. Quality of Service for VoIP traffic was maintained throughout testing in the presence of competing simulated traffic. 802.1x authentication was successful for both the IP telephones and PC's. LLDP-MED functioned correctly, however the cli show commands displayed invalid data. Juniper Networks has reported this as a known issue to be fixed in a later release.

2.3. Support

For technical support on the Juniper Networks product, contact Juniper Networks at (888) 314-5822, or refer to http://www.juniper.net

3. Reference Configuration

Figure 1 illustrates the configuration used in these Application Notes. 802.1X RADIUS authentication is enabled on the Virtual Chassis. All IP addresses are obtained via Dynamic Host Configuration Protocol (DHCP) unless noted. The "Telephony Infrastructure" VLAN with IP network 10.64.50.0/24, "Voice" VLAN with IP network 10.64.52.0/24, and "Data" VLAN with IP network 10.64.53.0/24 are used in the sample network.

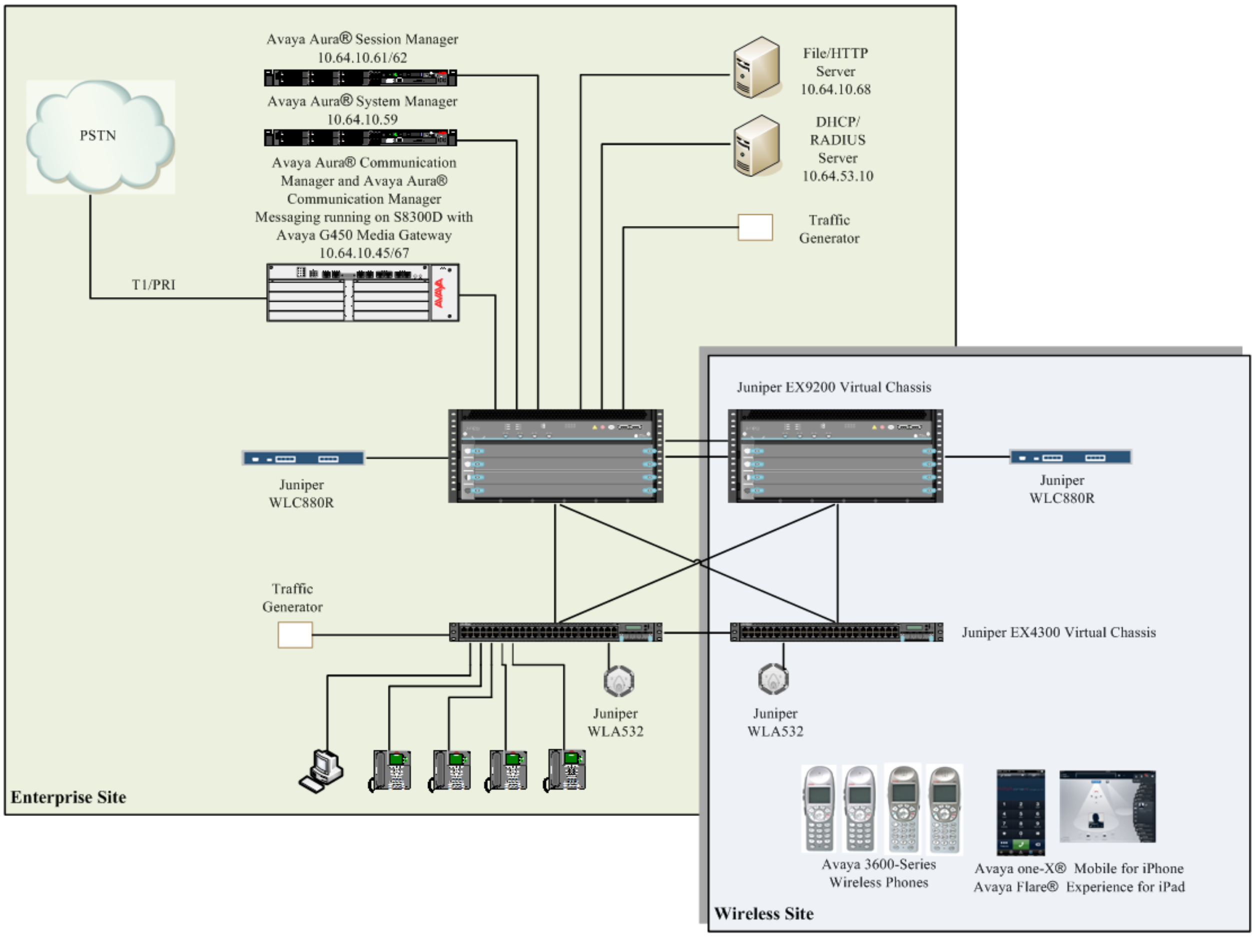

Avaya Aura® Session Manager
10.64.10.61/62
Avaya Aura® System Manager
10.64.10.59
Avaya Aura® Communication Manager and Avaya Aura® Communication Manager Messaging running on S8300D with Avaya G450 Media Gateway
10.64.10.45/67
PSTN
T1/PRI
File/HTTP Server
10.64.10.68
DHCP/ RADIUS Server
10.64.53.10
Traffic Generator
Juniper EX9200 Virtual Chassis
Juniper WLC880R
Juniper WLC880R
Traffic Generator
Juniper EX4300 Virtual Chassis
Juniper WLA532
Juniper WLA532
Enterprise Site
Avaya 3600-Series Wireless Phones
Avaya one-X® Mobile for iPhone
Avaya Flare® Experience for iPad
Wireless Site

Figure 1: Avaya Telephony Infrastructure with Juniper EX Virtual Chassis

4. Equipment and Software Validated

The following equipment and software were used for the sample configuration provided:

Equipment	Software/Firmware
Avaya PBX Products	
Avaya S8300D Server running Avaya Aura® Communication Manager	R016x.03.0.124.0
Avaya G450 Media Gateway MGP	HW 2 FW 31.20.0
Avaya Aura® Session Manager	
Avaya Aura® Session Manager HP Proliant DL360 G7	6.3.5.0
Avaya Aura® System Manager HP Proliant DL360 G7	6.3.5
Avaya Messaging (Voice Mail) Products	
Avaya Aura® Communication Manager Messaging (CMM)	6.3
Avaya Endpoints	
Avaya 96x0 Series IP Telephones	(H.323 3.2.1), (SIP 2.6.10)
Avaya 96x1 Series IP Telephones	(H.323 6.3.1), (SIP 6.3.1)
Avaya one-X® Communicator	6.2.0.024
Avaya 3600 Series	(SIP 1.1.1), (H.323 117.058)
Avaya one-X® mobile for iOS	6.2
Avaya Flare® Experience for iPad	1.2
Juniper Products	
EX4300 24-port 10/100/1000BASE-T Ethernet Switch with Power over Ethernet (PoE)	13.2X50-D17
EX9204: 4-slot modular chassis	13.2R2.4
WLC880 Wireless LAN Controller	9.0.2.5.0
WLA532 Access Points	9.0.2.5.0

5. Configure Juniper Networks Switches

This section describes the configuration for Juniper Networks EX4300 and EX9200 virtual chassis, as shown in **Figure 1** using the Command Line Interface (CLI).

Though not explicitly mentiononed, it is assumed that user saves the changes after configuration changes are made in each section, using **commit** command.

5.1. Configure EX4300 Virtual Chassis

1. Make a list of the serial numbers of all the switches to be connected in the Virtual Chassis.

2. Note the intended role (routing-engine or line-card) of each switch. If the member is configured with a routing-engine role, it is eligible to function in the master or backup role. If the member is configured with a line-card role, it is not eligible to function in the master or backup role.

3. Power on only the switch the master switch Once powered on, log in using appropriate credentials via SSH client.

4. Specify the pre-provisioned configuration mode:

```
[edit virtual-chassis]
regress@EX-EX4300-VC # set preprovisioned
```

5. Specify all the members that are included in the Virtual Chassis, listing each switch’s serial number with the desired member ID and role:

```
edit virtual-chassis]
regress@EX-EX4300-VC # set member 0 serial-number abc123 role routing-engine
regress@EX-EX4300-VC # set member 1 serial-number def456 role routing-engine
```

6. Optional. Recommended for a two-member Virtual Chassis. Disable the split and merge feature:

```
edit virtual chassis]
regress@EX-EX4300-VC # set no split detection
```

7. Power on the other member switches. The member IDs and roles have been determined by the configuration, so power on the member switches in any order.

8. Interconnect the member switches by using either the Quad (4-channel) Small Form-factor Pluggable **(**QSFP+) ports on the member switches or by connecting them through the uplink ports or enhanced small form-factor pluggable **(**SFP+) ports

9. On each individual member switch, configure the SFP+ optical ports that will be used to interconnect the EX4300 member switches into Virtual Chassis Ports (VCP). On each individual member switch , QSFP+ ports are configured into VCPs by default:

```
regress@EX-EX4300-VC >request virtual-chassis vc-port set pic slot 1 port 0
regress@EX-EX4300-VC >request virtual-chassis vc-port set pic slot 1 port 1
regress@EX-EX4300-VC >request virtual-chassis vc-port set pic slot 1 port 0
regress@EX-EX4300-VC >request virtual-chassis vc-port set pic slot 1 port 1
```

10. QSFP+ ports are configured into VCPs by default, they do not usually have to perform above step when they are using a QSFP+ port as a VCP.

5.2. Configure EX9200 Virtual Chassis

1. Ensure that both EX9200 member switches in the Virtual Chassis have dual Routing Engines installed.

2. Ensure all Routing Engines on both member switches are running the same version of Junos OS Release 13.2R2 or later.

3. Cable the Virtual Chassis member switches together. See Connecting a Fiber-Optic Cable to an EX Series Switch, Installing and Removing EX9204 Switch Hardware Components, Installing and Removing EX9208 Switch Hardware Components, or Installing and Removing EX9214 Switch Hardware Components.

4. Create and configure the configuration groups, as described in Creating Configuration Groups for an EX9200 Virtual Chassis.

5. Log onto the switch that needs to be assigned as member 0 in Virtual Chassis.

6. Specify the preprovisioned configuration mode:

```
edit virtual-chassis]
regress@EX-9200-VC # set preprovisioned
```

7. Please note that preprovisioned configuration mode must be used to configure an EX9200 Virtual Chassis.

8. Configure the Virtual Chassis by including both member switches in the Virtual Chassis configuration:

```
[edit virtual-chassis]
regress@EX-9200-VC # set member 0 serial-number serial-number role routing-engine
regress@EX-9200-VC # set member 1 serial-number serial-number role routing-engine
```

Serial-number is the chassis serial number of the member switch. Chassis serial number can be retrieved by using the **show chassis hardware** command output or by physically viewing the serial number label on the switch.

An EX9200 Virtual Chassis supports two member switches. Both switches should be assigned the routing-engine role.

For instance, if chassis with serial number JN1234567ABC needs to be configured as member 0 and the switch with chassis serial number JN9876543ZYX as member 1 in EX9200 Virtual Chassis:

```
[edit virtual-chassis]
regress@EX-9200-VC # set member 0 serial-number JN1234567ABC role routing-engine
regress@EX-9200-VC # set member 1 serial-number JN9876543ZYX role routing-engine
```

9. Disable the split and merge feature:

```
[edit virtual-chassis]
regress@EX-9200-VC # set no-split-detection
```

Disabling split and merge ensures that all interfaces on the member switch in the master Routing Engine role remain up if the member switch in the backup Routing Engine role fails.

Split and merge is enabled by default. If the member switch in the backup Routing Engine role fails when split and merge is enabled, all interfaces on all line cards that do not contain at least one VCP on the member switch in the master Routing Engine role also fail.

10. Commit the configuration:

```
[edit]
regress@EX-9200-VC # commit
```

11. Enable Virtual Chassis mode and set the member ID of the switch:

```
regress@EX-9200-VC >request virtual-chassis member-id set member 0
```

This command will enable virtual-chassis mode and reboot the system.
 Continue? [yes, no] (no) yes

Both Routing Engines must be rebooted on the switch to complete this step.

12. Log onto the switch that needs to be assigned as member 1 in Virtual Chassis.

13. Enable Virtual Chassis mode and set the member ID of the switch:

```
regress@EX-9200-VC >request virtual-chassis member-id set member 1
```

This command will enable virtual-chassis mode and reboot the system.
 Continue? [yes, no] (no) yes

Both Routing Engines must be rebooted on the switch to complete this step.

Log back onto member 0 after the reboot is complete. Configure the interfaces that need to be configured as VCPs:

```
regress@EX-9200-VC >request virtual-chassis vc-port set fpc-slot fpc-slot-number pic-slot
pic-slot-number port port-number
```

For instance, configure port 0 on PIC slot 1 in FPC slot 1 as a VCP using the following command:

```
regress@EX-9200-VC > request virtual-chassis vc-port set fpc-slot 1 pic-slot 1 port 0
```

14. Log back onto member 1 after the reboot is complete. Configure the interfaces that needs to be configured as VCPs:

```
{master:1}[edit]
regress@EX-9200-VC >request virtual-chassis vc-port set fpc-slot fpc-slot-number pic-slot
pic-slot-number port port-number
```

5.3. Configure LAG between EX4300VC and EX9200VC

1. login to EX4300 VC .Specify the aggregated Ethernet options and the interface ID of the uplinks to be included in LAG ae0.

```
{master:1}[edit]
user@EX-4300-VC # set interfaces ae0 aggregated-ether-options lacp active
{master:1}[edit]
user@EX-4300-VC # set interfaces ae0 aggregated-ether-options lacp periodic fast
{master:1}[edit]
user@EX-4300-VC # set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk
{master:1}[edit]
user@EX-4300-VC # set interfaces ae0 unit 0 family ethernet-switching vlan members all
{master:1}[edit]
user@EX-4300-VC # set interfaces xe-1/2/0 ether-options 802.3ad ae0
{master:1}[edit]
user@EX-4300-VC # set interfaces xe-0/2/0 ether-options 802.3ad ae0
```

2. Log in to EX9200 VC. Specify the aggregated Ethernet options and the interface ID of the uplinks to be included in LAG ae0.

```
{master:1}[edit]
```

```
regress@EX-9200-VC  # set interfaces ae0 aggregated-ether-options lacp active
{master:1}[edit]
regress@EX-9200-VC  # set interfaces ae0 aggregated-ether-options lacp periodic fast
{master:1}[edit]
regress@EX-9200-VC  # set interfaces ae0 unit 0 family ethernet-switching interface-mode
trunk
{master:1}[edit]
regress@EX-9200-VC  # set interfaces ae0 unit 0 family ethernet-switching vlan members all
{master:1}[edit]
regress@EX-9200-VC  # set interfaces xe-2/0/2 ether-options 802.3ad ae0
{master:1}[edit]
regress@EX-9200-VC  # set interfaces xe-14/0/3 ether-options 802.3ad ae0
```

5.4. Configure VLAN and port assignment

1. Log into the EX9200 Virtual Chassis switch using appropriate credential.

```
login: username
Password: ******
```

2. Enter configuration mode by typing configure at the prompt.

```
{master:1}
regress@EX-9200-VC> configure
Entering configuration mode
```

3. Create VLANs for telephony infrastructure, data and voice. The sample network uses VLAN tag **50** for telephony infrastructure, VLAN tag **52** for voice, VLAN tag **53** for data and VLAN tag **62** for wireless. The IP address of **10.64.53.64** will be used as the source IP address for sending RADIUS authentication to the FreeRADIUS server in **Section 8**.

```
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan50 description "Telephony Infrastructure"
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan50 vlan-id 50
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan52 description Voice
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan52 vlan-id 52
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan52 l3-interface irb.52
{master:1}[edit]
regress@EX-9200-VC# set interfaces vlan unit 1 family inet address 10.64.52.64/24
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan53 description Data
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan53 vlan-id 53
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan53 l3-interface irb.53
{master:1}[edit]
regress@EX-9200-VC# set interfaces vlan unit 0 family inet address 10.64.53.64/24
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan53 vlan-id 62
{master:1}[edit]
regress@EX-9200-VC# set vlans vlan53 l3-interface irb.62
{master:1}[edit]
regress@EX-9200-VC# set interfaces vlan unit 0 family inet address 10.64.62.64/24
```

4. Configure switch ports to support Telephony Infrastructure, DHCP, and RADIUS server connected to EX9200 Virtual chassis.

```
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/0 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/0 unit 0 family ethernet-switching vlan
members vlan50
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/1 unit 0 family ethernet-switching
interface-mode  access
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/1 unit 0 family ethernet-switching vlan
members vlan50
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/2 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/2 unit 0 family ethernet-switching vlan
members vlan50
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/3 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/3 unit 0 family ethernet-switching vlan
members vlan50
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/4 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/4 unit 0 family ethernet-switching vlan
members vlan50
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/5 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-9200-VC# set interfaces ge-1/0/5 unit 0 family ethernet-switching vlan
members vlan50
```

Configure switch ports on EX4300 Virtual chassis to support Avaya IP Telephones, access points and Personal Computers.

```
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/0 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/0 unit 0 family ethernet-switching vlan
members vlan62
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/1 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan
members vlan62
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/2 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/2 unit 0 family ethernet-switching vlan
members vlan53
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/3 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
```

```
regress@EX-4300-VC# set interfaces ge-0/0/3 unit 0 family ethernet-switching vlan
members vlan53
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/4 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/4 unit 0 family ethernet-switching vlan
members vlan53
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/5 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/5 unit 0 family ethernet-switching vlan
members vlan53
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/6 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/6 unit 0 family ethernet-switching vlan
members vlan53
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/7 unit 0 family ethernet-switching
interface-mode access
{master:1}[edit]
regress@EX-4300-VC# set interfaces ge-0/0/7 unit 0 family ethernet-switching
vlamembers vlan53
{master:1}[edit]
regress@EX-4300-VC#set switch-options voip interface ge-0/0/2.0 vlan vlan52
{master:1}[edit]
regress@EX-4300-VC#set switch-options voip interface ge-0/0/3.0 vlan vlan52
{master:1}[edit]
regress@EX-4300-VC#set switch-options voip interface ge-0/0/4.0 vlan vlan52
{master:1}[edit]
regress@EX-4300-VC#set switch-options voip interface ge-0/0/5.0 vlan vlan52
{master:1}[edit]
regress@EX-4300-VC#set switch-options voip interface ge-1/0/6.0 vlan vlan52
master:1}[edit]
regress@EX-4300-VC#set switch-options voip interface ge-1/0/6.0 vlan vlan52
```

5.5. Configure LLDP-MED

1. Enable LLDP-MED on all interfaces.

```
{master:1}[edit]
regress@EX-4300-VC# set protocols lldp-med interface all
```

5.6. Configure PoE

1. Enable PoE on all interfaces.

```
{master:1}[edit]
regress@EX-4300-VC# set poe interface all
```

5.7. Configure DHCP relay

Configure DHCP relay on EX4300 Virtual chassis to forward all DHCP request to the DHCP server connected to EX9200 Virtual chassis.

```
master:1}[edit]
regress@EX-4300-VC #set forwarding-options dhcp-relay server-group srv1 10.64.10.16
master:1}[edit]
regress@EX-4300-VC #set forwarding-options dhcp-relay group relay2 active-server-group srv1
master:1}[edit]
```

```
regress@EX-4300-VC #set forwarding-options dhcp-relay group relay2 interface irb.53
master:1}[edit]
regress@EX-4300-VC #set forwarding-options dhcp-relay group relay52 active-server-group
srv1
master:1}[edit]
regress@EX-4300-VC #set forwarding-options dhcp-relay group relay52 interface irb.52
master:1}[edit]
regress@EX-4300-VC #set forwarding-options dhcp-relay group relay62 active-server-group
srv1
master:1}[edit]
regress@EX-4300-VC #set forwarding-options dhcp-relay group relay62 interface irb.62
```

5.8. Configure Quality of Service (QoS) for VoIP traffic

This section describes the step in configuring QoS for Avaya VoIP traffic on the EX 4300 Virtual-Chassis switch.

1. Define a new priority queue **6**. This priority queue will be used for VoIP traffic. By default all network control and best-effort traffic are assigned to priority queue 7 and 0 respectively.

```
{master:1}[edit]
regress@EX-EX4300-VC# set class-of-service forwarding-classes class voice queue-
         num 6
```

2. Create a new classifier profile. Import the default classifier to avoid defining all DiffServ Code Point (DSCP) values and reclassify DSCP value of the VoIP traffic that needs to be prioritized.

```
{master:1}[edit]
regress@EX-EX4300-VC# set class-of-service classifiers dscp avaya_dscp import
         default
{master:1}[edit]
regress@EX-EX4300-VC# set class-of-service classifiers dscp avaya_dscp forwarding-
         class voice loss-priority low code-points 101110
```

3. Configure the scheduler for the different traffic types.

```
{master:1}[edit]
regress@EX-EX4300-VC# set class-of-service schedulers network-control-scheduler
buffer-size percent 5
{master:1}[edit]
regress@EX-4300-VC# set class-of-service classifiers dscp avaya_dscp forwarding-
         class voice loss-priority low code-points 101110
{master:1}[edit]
regress@EX-4300-VC# set class-of-service schedulers network-control-scheduler
         priority strict-high
{master:1}[edit]
regress@EX-4300-VC# set class-of-service schedulers voice-scheduler priority
         strict-high
{master:1}[edit]
regress@EX-4300-VC# set class-of-service schedulers best-effort-scheduler transmit-
         rate percent 90
{master:1}[edit]
regress@EX-4300-VC# set class-of-service schedulers best-effort-scheduler buffer-
         size percent 90
```

4. Create a scheduler profile to map schedulers to a forwarding class. The sample network uses the profile **avaya_sch_prfl**.

```
{master:1}[edit]
regress@EX-4300-VC# set class-of-service scheduler-maps avaya_sch_prfl forwarding-
         class network-control scheduler network-control-scheduler
{master:1}[edit]
regress@EX-4300-VC# set class-of-service scheduler-maps avaya_sch_prfl forwarding-
         class voice scheduler voice-scheduler
{master:1}[edit]
regress@EX-4300-VC# set class-of-service scheduler-maps avaya_sch_prfl forwarding-
         class best-effort scheduler best-effort-scheduler
```

5. Apply the scheduler profile and classifiers to the access and uplink ports.

```
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/0 scheduler-map
         avaya_sch_prfl
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/0 unit 0 classifiers
         dscp avaya_dscp
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/1 scheduler-map
         avaya_sch_prfl
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/1 unit 0 classifiers
         dscp avaya_dscp
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/2 scheduler-map
         avaya_sch_prfl
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/2 unit 0 classifiers
         dscp avaya_dscp
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/3 scheduler-map
         avaya_sch_prfl
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/3 unit 0 classifiers
         dscp avaya_dscp
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/4 scheduler-map
         avaya_sch_prfl
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/4 unit 0 classifiers
         dscp avaya_dscp
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/5 scheduler-map
         avaya_sch_prfl
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/5 unit 0 classifiers
         dscp avaya_dscp
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/6 scheduler-map
         avaya_sch_prfl
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/6 unit 0 classifiers
         dscp avaya_dscp
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/7 scheduler-map
         avaya_sch_prfl
{master:1}[edit]
regress@EX-4300-VC# set class-of-service interfaces ge-0/0/7 unit 0 classifiers
         dscp avaya_dscp
```

6. Save the changes.

```
{master:1}[edit]
regress@EX-4300-VC# commit
```

5.9 Configure 802.1x RADIUS authentication

This section describe the step in configure 802.1x RADIUS multiple supplicant support for Avaya IP Telephone and PC. The multiple supplicant mode forces Avaya IP Telephone and the PC to each individually authenticate against the RADIUS server before access to the network is allowed.

1. Configure the RADIUS server information. The shared secret string must match what is configured on the FreeRADIUS server in **Section 8**. Please note that the IP addresses and secret displayed in the following screen capture is an example of the configuration used during compliance testing.

```
{master:1}[edit]
regress@EX-4300-VC# set access radius-server 10.64.59.246 secret 1234567890
{master:1}[edit]
regress@EX-4300-VC# set access radius-server 10.64.59.246 source-address
         10.64.53.64
{master:1}[edit]
regress@EX-4300-VC# set access profile extest authentication-order radius
{master:1}[edit]
regress@EX-4300-VC# set access profile extest radius authentication-server
         10.64.59.246
{master:1}[edit]
regress@EX-4300-VC# set protocols dot1x authenticator authentication-profile-name
         extest
```

2. Enable multiple supplicant support for the switch port.

```
{master:1}[edit]
regress@EX-4300-VC# set protocols dot1x authenticator interface ge-0/0/0.0
         supplicant multiple
```

6. Configure Wireless Controller

The following configuration steps can performed on the WLC after a basic device setup has been performed using the **quickstart** CLI command.

[perform steps 6.1 through 6.6 on the Primary WLC]

6.1 Configure VLAN and Port Assignments

Create the Campus Voice VLAN and map it to port one of the controller

```
> set vlan 62 name campus-voice
> set vlan 62 port 1
```

6.2 Assign IP address 10.64.62.5 to interface 62

```
> set interface 62 ip 10.64.62.5 255.255.255.0
```

6.3 Assign System IP Address and Country Code

```
> set system ip-address 10.64.62.5
> set system countrycode US
```

6.4 Create Virtual Controller Cluster

```
> set mobility-domain mode seed domain-name AvayaTest
> set mobility-domain member 10.64.62.6
> set cluster mode enable
```

6.5 Configure Access Points

Access points in the WL solution can be either statically configured or allowed to automatically join a WLC using a default profile. The example below configures the system for Auto AP.

```
> set ap auto mode enable
```

6.6 Configure WLAN services

Create the Service Profile (SSID) for the Voice network.

```
> set service-profile ca-voice ssid-name ca-voice
> set service-profile ca-voice auth-fallthru last-resort
> set service-profile ca-voice psk-encrypted 045f5a555b204e195141574714090f5d73782
571606d244b024454530e5a5e500d0c52484e0059070a040c51590951500c030217025008590456201948
0b495246
> set service-profile ca-voice rsn-ie cipher-ccmp enable
> set service-profile ca-voice rsn-ie auth-psk enable
> set service-profile ca-voice rsn-ie auth-dot1x disable
> set service-profile ca-voice rsn-ie enable
> set service-profile ca-voice attr vlan-name campus-voice
```

Add the Service Profile to the default Radio Profile and modify voice parameters of the Radio Profile.

```
> set radio-profile default dtim-interval 2
> set radio-profile default rf-scanning mode passive
> set radio-profile default rf-scanning channel-scope operating
> set radio-profile default wmm-powersave enable
> set radio-profile default power-policy cell-parity
> set radio-profile default power-policy cell-parity 11bg-power 12
> set radio-profile default power-policy cell-parity 11a-power 12
> set radio-profile default cac video mode enable
> set radio-profile default cac voice mode enable
> set radio-profile default service-profile ca-voice
```

[perform steps 6.7 through 6.10 on the Secondary WLC]

6.7 Configure VLAN and Port Assignments

Create the Campus Voice VLAN and map it to port one of the controller.

```
> set vlan 62 name campus-voice
> set vlan 62 port 1
```

6.8 Assign IP address 10.64.62.6 to interface 62

```
> set interface 62 ip 10.64.62.6 255.255.255.0
```

6.9 Assign System IP Address and Country Code

```
> set system ip-address 10.64.62.6
> set system countrycode US
```

6.10 Create Virtual Controller Cluster

```
> set mobility-domain mode secondary-seed domain-name AvayaTest seed-ip 10.64.62.5
> set mobility-domain member 10.64.62.5
> set cluster mode enable
```

7 Configure IP Telephone

The Juniper Networks switches support “Client-Based” authentication to ensure multiple clients sharing the same port are authenticated individually. This is a required to support secure authentication of both the Avaya IP Telephone with an attached PC to be independently authenticated and provisioned in different VLANs when connected to Juniper Networks switches.

Avaya IP Telephones support three 802.1X operational modes. The operational mode can be changed by pressing “mute80219#” (“mute8021x”) on the Avaya 46xx IP Telephones or by pressing the Craft Access Code (the default is “<mute>craft#” or “<mute>27283#”) on the Avaya 96xx IP Telephones.

- **Pass-thru Mode** – Unicast supplicant operation for the IP Telephone itself, with PAE multicast pass-through for the attached PC, but without proxy Logoff (default).

- **Pass-thru with logoff Mode (p –t w/Logoff)** – Unicast supplicant operation for the IP Telephone itself, with PAE multicast pass-through and proxy Logoff for the attached PC. When the attached PC is physically disconnected from the IP Telephone, the phone will send an EAPOL-Logoff for the attached PC (**recommended mode**).

- **Supplicant Mode** – Unicast or multicast supplicant operation for the IP Telephone itself, without PAE multicast pass-through or proxy Logoff for the attached PC.

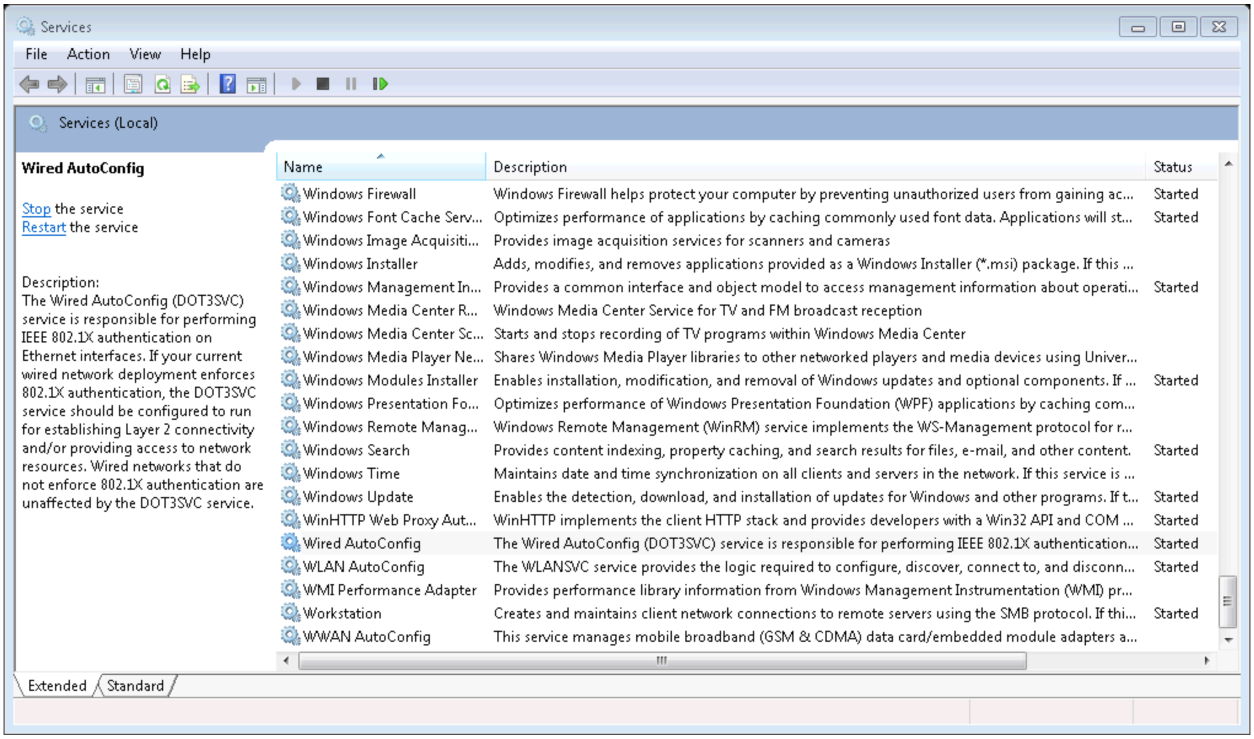
Since most 802.1X clients use the special PAE group multicast MAC address for the EAPOL messages, the IP Telephone must be configured to the **pass-thru** or **p-t w/Logoff** mode to pass-through these Multicast messages. It is recommended to use the **p-t w/Logoff** mode for improved security. This is because when the phone is in the **p-t w/Logoff** mode, the phone will do a proxy logoff on behalf of the attached PC when the PC is physically disconnected. When the Juniper Networks switches receive the EAPOL logoff message, it will immediately remove the PC from the authorized MAC list.

When proxy logoff is not enabled, the Juniper Networks switch is unable to detect a link loss when the PC is disconnected from the phone and will defer cleanup of the authorized MAC list until no more packets with the PC MAC address have been seen for a duration specified by the ‘logoff-period’ (default timeout is 5 min).

NOTE: it is strongly recommended to not use “port-based” 802.1X authentication on ports connected to IP phones, since this mode only authenticates the first client device that connects. As long as the port has been opened by an authenticated device, the port will remain opened until that device disconnects or the authentication session expires. Thus, once an IP phone is authenticated, any device plugged into the back of the phone would have full access to the network without needing to authenticate and effectively bypassing Network Access Control.

8 Configuring 802.1X Windows 7 Client

Click the Windows Start button and then enter **services** in the search box (Not Shown) to open the Services window. Scroll to the bottom of the list and start the **Wired AutoConfig** service.



Wired AutoConfig

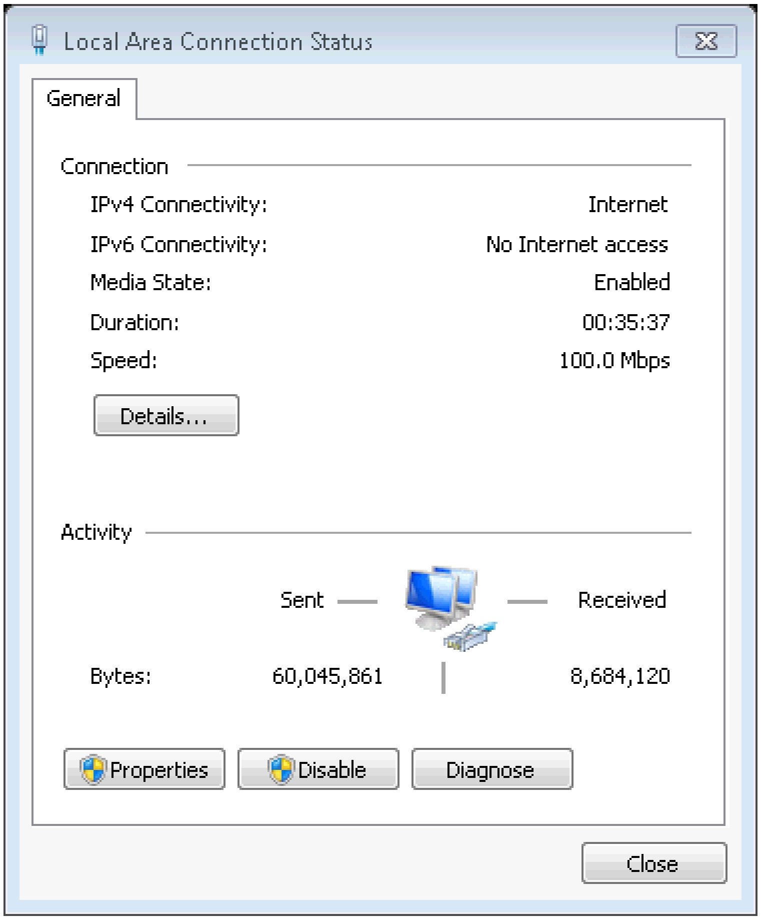
Stop the service
Restart the service

Description:
The Wired AutoConfig (DOT3SVC) service is responsible for performing IEEE 802.1X authentication on Ethernet interfaces. If your current wired network deployment enforces 802.1X authentication, the DOT3SVC service should be configured to run for establishing Layer 2 connectivity and/or providing access to network resources. Wired networks that do not enforce 802.1X authentication are unaffected by the DOT3SVC service.

Name	Description	Status
Windows Firewall	Windows Firewall helps protect your computer by preventing unauthorized users from gaining ac...	Started
Windows Font Cache Serv...	Optimizes performance of applications by caching commonly used font data. Applications will st...	Started
Windows Image Acquisiti...	Provides image acquisition services for scanners and cameras	
Windows Installer	Adds, modifies, and removes applications provided as a Windows Installer (*.msi) package. If this ...	
Windows Management In...	Provides a common interface and object model to access management information about operati...	Started
Windows Media Center R...	Windows Media Center Service for TV and FM broadcast reception	
Windows Media Center Sc...	Starts and stops recording of TV programs within Windows Media Center	
Windows Media Player Ne...	Shares Windows Media Player libraries to other networked players and media devices using Univer...	
Windows Modules Installer	Enables installation, modification, and removal of Windows updates and optional components. If ...	Started
Windows Presentation Fo...	Optimizes performance of Windows Presentation Foundation (WPF) applications by caching com...	
Windows Remote Manag...	Windows Remote Management (WinRM) service implements the WS-Management protocol for r...	
Windows Search	Provides content indexing, property caching, and search results for files, e-mail, and other content.	Started
Windows Time	Maintains date and time synchronization on all clients and servers in the network. If this service is ...	
Windows Update	Enables the detection, download, and installation of updates for Windows and other programs. If t...	Started
WinHTTP Web Proxy Aut...	WinHTTP implements the client HTTP stack and provides developers with a Win32 API and COM ...	Started
Wired AutoConfig	The Wired AutoConfig (DOT3SVC) service is responsible for performing IEEE 802.1X authentication...	Started
WLAN AutoConfig	The WLANSVC service provides the logic required to configure, discover, connect to, and disconn...	Started
WMI Performance Adapter	Provides performance library information from Windows Management Instrumentation (WMI) pr...	
Workstation	Creates and maintains client network connections to remote servers using the SMB protocol. If thi...	Started
WWAN AutoConfig	This service manages mobile broadband (GSM & CDMA) data card/embedded module adapters a...	

Click the Windows Start button and then enter **ncpa.cpl** in the search box (Not Shown) to open the Network Connections window.

Double click the LAN connection (Not Shown) and then click the **Properties** button.



IPv4 Connectivity: Internet

IPv6 Connectivity: No Internet access

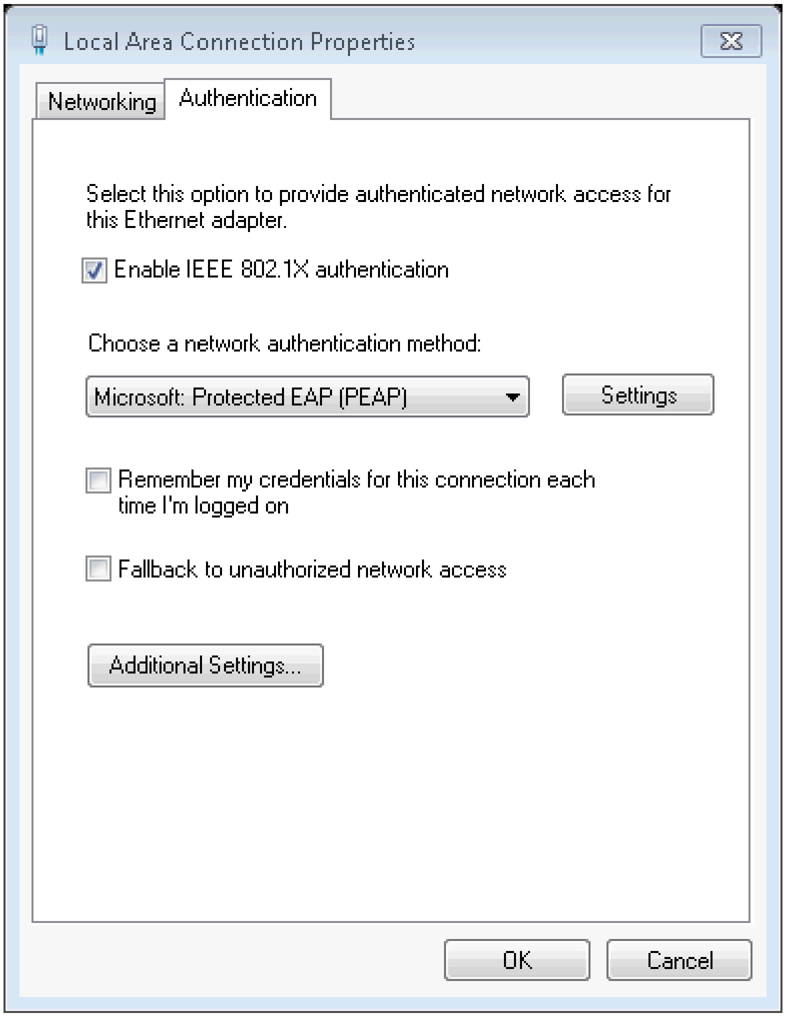
Media State: Enabled

Duration: 00:35:37

Speed: 100.0 Mbps

Bytes: 60,045,861 8,684,120

From the Local Area Connection Properties window click the **Authentication** Tab and check **Enable IEEE 802.1X authentication** box. Continue by clicking the **Additional Settings** button.



Local Area Connection Properties

Select this option to provide authenticated network access for this Ethernet adapter.

☑ Enable IEEE 802.1X authentication

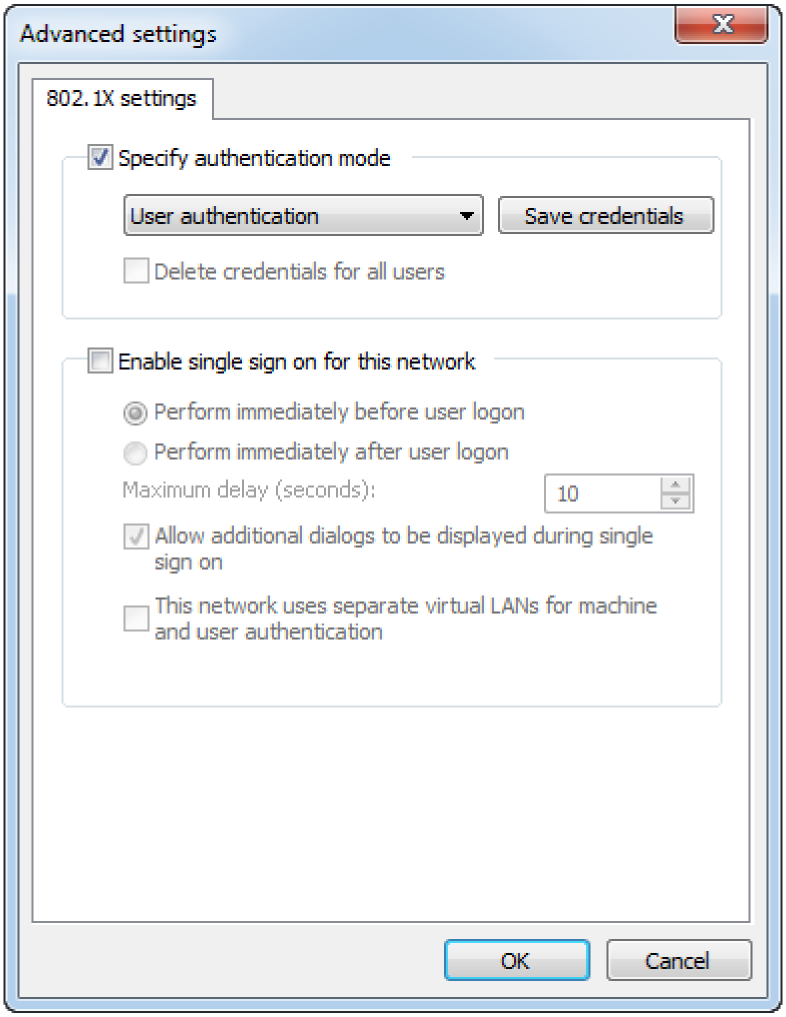
Choose a network authentication method:

Microsoft: Protected EAP (PEAP)

□ Remember my credentials for this connection each time I'm logged on

□ Fallback to unauthorized network access

From this window check the **Specify Authentication Mode** box and select the appropriate authentication mode from the pull-down box. The compliance test used **User Authentication** and required adding the user credentials by clicking the **Save Credentials** button and adding the credentials.



☑ Specify authentication mode

User authentication

□ Delete credentials for all users

□ Enable single sign on for this network

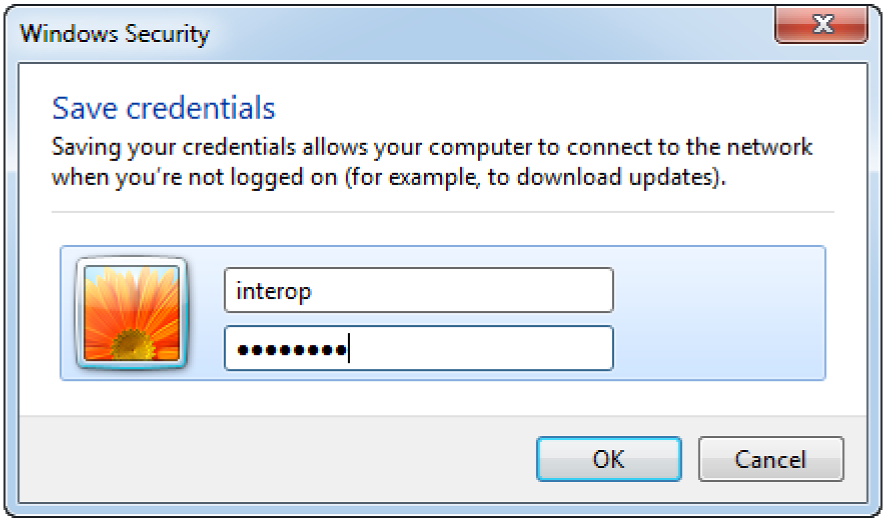
● Perform immediately before user logon

○ Perform immediately after user logon

Maximum delay (seconds): 10

☑ Allow additional dialogs to be displayed during single sign on

□ This network uses separate virtual LANs for machine and user authentication



Save credentials

Saving your credentials allows your computer to connect to the network when you're not logged on (for example, to download updates).

interop

•••••••••

Once completed click the **OK** buttons on each of the open windows and the LAN connection should be successfully authenticated and become active.

9 Configuring RADIUS Server

In the sample configuration, Linux FreeRADIUS was used as the authentication server. The intent of this section is to illustrate relevant aspects of the configuration used for the testing.

For the compliance test the following entry was added to the **/etc/freeradius/clients.conf** file to support the Juniper Networks Switches.

```
client 10.64.53.0/24 {
        # This is the shared secret between the Authenticator (the
        # access point) and the Authentication Server (RADIUS).
        secret          = 1234567890123
        shortname       = juniper
   }
```

The following entries were added to the **/etc/freeradius/users** file to support the Avaya Telephones.

```
#9641
b4b0178d3c24 Cleartext-Password := "123456"
#9670
001B4F43FE2F Cleartext-Password := "123456"
#9611
B4B0178996CE Cleartext-Password := "123456"
#PC User
interop Cleartext-Password := "123456"
```

10 Verification Steps

The following steps may be used to verify the configuration:

1. Use the **show virtual-chassis** command to verify virtual-chassis status is **Prsnt**.

```
{master:1}
regress@EX-4300-VC> show virtual-chassis

Preprovisioned Virtual Chassis

Virtual Chassis ID: 762c.7c4e.189e

                                                Mstr           Mixed Neighbor List
Member ID  Status   Serial No    Model          prio  Role      Mode ID  Interface
0 (FPC 0)  Prsnt    PG3113080001 ex4300-24t     129   Master*     NA  1  vcp-255/1/1

1 (FPC 1)  Prsnt    PD3113060042 ex4300-48p     129   Backup      NA  0  vcp-255/1/1
```

```
{master:member0-re0}
root@EX9200-A> show virtual-chassis vc-port
member0:

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

Interface       Type              Trunk  Status       Speed        Neighbor
or                                 ID                 (mbps)       ID  Interface
Slot/PIC/Port
2/0/0           Configured          5    Up           10000        1   vcp-2/0/0
2/0/1           Configured          5    Up           10000        1   vcp-2/0/1

member1:
---------------------------------------------------------------------------

Interface       Type              Trunk  Status       Speed        Neighbor
or                                 ID                 (mbps)       ID  Interface
Slot/PIC/Port
2/0/0           Configured          5    Up           10000        0   vcp-2/0/0
2/0/1           Configured          5    Up           10000        0   vcp-2/0/1
```

2. Use the **show dot1x interface** command to verify endpoint authentication.

```
{master:1}
regress@EX-4300-VC> show dot1x interface
802.1X Information:

Interface     Role           State            MAC address          User
ge-1/0/4.0    Authenticator  Authenticated    00:1B:4F:43:FE:2F    001B4F43FE2F
ge-1/0/5.0    Authenticator  Authenticated    B4:B0:17:8D:3C:24    b4b0178d3c24
```

3. Use the **show lldp neighbors** command to display LLDP neighbor information.

```
{master:1}
regress@EX-4300-VC> show lldp neighbors
Local Interface    Parent Interface    Chassis Id          Port info          System
Name
xe-0/2/2           ae0                 78:19:f7:70:96:52   port 1       AvayaTestWLC2
xe-1/2/0           ae0                 78:19:f7:72:c9:74   port 1       AvayaTestWLC1
ge-1/0/5           -                   10.64.52.100       b4:b0:17:8d:3c:24 AVX8D3C24
ge-1/0/4           -                   10.64.52.102       b4:b0:17:89:96:ce AVX8996CE
ge-1/0/3           -                   10.64.52.104       00:07:3b:e1:92:2c AVTE1922C
ge-1/0/0           -                   10.64.62.100       ac:4b:c8:37:ee:80
JB0212388400
ge-1/0/1           -                   10.64.62.101        ac:4b:c8:37:ef:00
JB0212388399
```

4. Use the **show lldp neighbor interface** command to verify detail LLDP information learned from a switch port.

   The following is an example of what is displayed for an Avaya 9608 IP Telephone.

```
{master:1}
root@EX4300-VC> show lldp neighbors interface ge-1/0/4
LLDP Neighbor Information:
```

```
Local Information:
Index: 10 Time to live: 120 Time mark: Thu Feb 13 06:50:44 2014 Age: 14 secs
Local Interface    : ge-1/0/4
Parent Interface   : -
Local Port ID      : 526
Ageout Count      : 0



Neighbour Information:
Chassis type       : Network address
Chassis ID         : 10.64.52.102
Port type          : Mac address
Port ID            : b4:b0:17:89:96:ce
System name        : AVX8996CE

System capabilities
      Supported: Bridge Telephone
      Enabled  : Bridge


Management address
        Address Type      : IPv4(1)
        Address           : 10.64.52.102
        Interface Number  : 1
        Interface Subtype : SysPortNum(3)
        OID               : 43.6.1.4.1.181.105.1.69.6.2.

Media endpoint class: Class III Device


MED Hardware revision : 9611
MED Firmware revision : S96x1_UKR_V12r2497_V12r2497.tar
MED Software revision : S96x1_SALBR6_2_2r17_V4r70.tar
MED Serial number     : 11WZ04556570
MED Manufacturer name : Avaya
MED Model name        : 9611


Organization Info
       OUI      : IEEE 802.3 Private (0x00120f)
       Subtype  : MAC/PHY Configuration/Status (1)
       Info     : Autonegotiation [supported, enabled (0x3)], PMD Autonegotiation
Capability (0x0), MAU Type (0x1e00)
       Index    : 1


Organization Info
       OUI      : 0.12.bb
       Subtype  : 1
       Info     : 002303
       Index    : 2


Organization Info
       OUI      : 0.12.bb
       Subtype  : 2
       Info     : 014069A8
       Index    : 3
```

5. Use the **show poe interface** command to verify PoE information.

```
{master:1}
regress@EX-4300-VC> show poe interface ge-0/0/0
PoE interface status:
PoE interface                :  ge-1/0/3
Administrative status        : Enabled
Operational status           :   ON
Power limit on the interface : 7.0W
Priority                     : Low
Power consumed               : 3.5W
Class of power device        :        2
PoE Mode                     :   802.3at
```

11 Conclusion

These Application Notes describe a compliance-tested configuration comprised of Avaya Aura® Telephony Infrastructure connected to Juniper Network Switches configured as a Virtual Chassis. The Juniper Networks Virtual Chassis Switch enforced auto discovery of Voice VLAN and L2/L3 QoS parameters using LLDP-MED. Additionally Juniper Networks Virtual Chassis Switch provided Power over Ethernet for the Avaya Telephones and performed 802.1x authentication using RADIUS. Prioritization of VoIP traffic and good voice quality was successfully achieved in the Avaya/Juniper Networks configuration described in Figure 1. Juniper Networks successfully passed the compliance test. Refer to **Section 2.2** for more details and listed observations.

12 Additional References

The documents referenced below were used for additional support and configuration information.

Product documentation for Avaya products may be found at http://support.avaya.com

[1] *Administering Avaya Aura® Communication Manager*, Release 6.3, Issue 3, October 2013
[2] *Administering Avaya Aura® Session Manager*, Release 6.3, Issue 3, October 2013

Product documentation for Juniper Networks products may be found at http://www.juniper.net

[3] *Complete Software Guide for JUNOS for EX-series Software*, Release 13.2, and Revision R4.

Please e-mail any questions or comments pertaining to these Application Notes along with the full title name and filename, located in the lower right corner, directly to the Avaya DevConnect Program at devconnect@avaya.com.