(!) Warning: Cannot modify header information - headers already sent by (output started at /home/candela/btbits/x64_btbits/html/wifire_cb.php:11) in /home/candela/btbits/x64_btbits/html/javascript_hdr.php on line 245

Call Stack

#	Time	Memory	Function	Location
1	0.0000	357840	{main}( )	.../wifire_cb.php:0
2	0.0006	573440	require_once('/home/candela/btbits/x64_btbits/html/javascript_hdr.php')	.../wifire_cb.php:13
3	0.0006	573568	setcookie($name = 'devmode', $value = 1, $expires_or_options = 1713304314)	.../javascript_hdr.php:245

Candela TECHNOLOGIES Network Testing and Emulation Solutions

sales@candelatech.com
support@candelatech.com
+1 (360) 380-1618 [PST, GMT -8]

LANforge WiFIRE Cookbook

The LANforge WiFIRE Cookbook provides a set of high-level examples of how to setup useful test scenarios in LANforge-FIRE for WiFi traffic generation. Each example intends to give the reader a brief introduction to the test scenario and a set of step-by-step instructions on how to use the LANforge-GUI to configure the test.

All of the following examples will work on Linux systems running the LANforge software with the LANforge kernel and a sufficient license. If you are running another Linux kernel, you will not be able to exactly duplicate some of the examples, but there are usually work-arounds available to assist you. Please contact us at support@candelatech.com if you have any questions.

If you are using the Windows version of LANforge, you will have to modify ports using the Windows utilities.

LANforge WiFIRE WiFi Traffic Generation

Before attempting the examples below, ensure that you have successfully followed these software installation guides:

- LANforge-GUI Installation
- LANforge Server Installation

It is also recommended that you back up your current running LANforge Server database so that you may safely return to your current operating state.

Quick Examples

- LANforge-FIRE Examples
- LANforge WiFIRE Wireless Testing Examples

LANforge WiFIRE Detailed Cookbook Examples

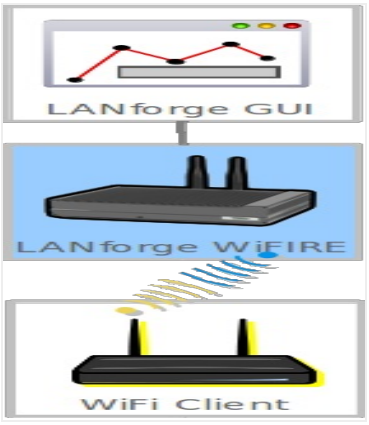
1. AP Testing: Basic AP Testing
2. AP Testing: Jamming a Channel With CT523
3. UE Testing: Creating a VAP in Bridge Mode
4. Audio Quality Testing: VoIP-Mobile Audio Quality Testing using POLQA/PESQ
5. Audio Quality Testing: Mobile-Mobile Audio Quality Testing using POLQA/PESQ
6. UE Testing: Creating a Routed VAP to Ethernet
7. AP Testing: Station Performance for 1, 50, and 128 Stations
8. AP Testing: Multicast-Download Test
9. UE Testing: Configuring a Virtual AP With Limited Stations
10. UE Testing: WiFi-Station-Association Code 17
11. Enterprise: HotSpot 2.0 With EAP-TTLS (Client Needs Only Username/Password)
12. Enterprise: HotSpot 2.0 With EAP-TLS and EAP-TTLS
13. Enterprise: HotSpot 2.0 With EAP-AKA
14. Enterprise: HotSpot 2.0 With EAP-SIM
15. Enterprise: Fast-Roaming Stations With 802.11r (FT-EAP)
16. Enterprise: Two-Access-Point Network With 802.11r (FT-EAP)
17. Enterprise: Four-Access-Point Network With 802.11r (FT-EAP)
18. Enterprise: Roaming With HotSpot 2.0
19. Enterprise: HotSpot-2.0 Release 2
20. AP Testing: Rate-vs.-Range Test
21. AP Testing: Attenuation-vs.-PDU-Size Test
22. AP Testing: WiFi-Capacity Test
23. AP Testing: Station-Reset Test
24. AP Testing: Emulating Degraded Stations
25. AP Testing: Scripted Attenuation of a WiFi Connection
26. AP Testing: External Traffic Generation
27. Captive Portal: Captive-Portal Testing
28. Captive Portal: Older Captive-Portal Login lf_ifup_post
29. UE Testing: Emulating Many VAPs With One Radio for Station-DUT Testing
30. UE Testing: Emulating a Degraded VAP: Station-DUT-Management-Frame-Retry Logic
31. AP Testing: Multiple Stations With a Hunt Script
32. UE Testing: Running UDP Traffic With Android
33. UE Testing: Verifying Android QoS
34. UE Testing: ScriptHunt With Test Groups
35. AP Testing: WPS Association
36. AP Testing: Station MU-MIMO Performance for 2x2 and 1x1 Stations
37. UE Testing: WiFi-Capacity Test With Layer 4-7
38. AP Testing: Sniffing WiFi Monitors With Wireshark
39. AP Testing: Scripting a Stand-Alone CT703 or CT704 Attenuator
40. AP Testing: DFS Testing With CT712 RF-Noise Generator and RADAR Simulator
41. Mesh Testing: Mesh-Testing Video Demonstration
42. Packet Capture: WiFi Packet Capture and Diagnostics
43. Packet Capture: Powersave Test Cases
44. Packet Capture: Scripting Multiple Simultaneous Packet Captures, Including OFDMA
45. Packet Capture: OFDMA vs. MU-MIMO Packet-Capture Analysis
46. Packet Capture: 6GHz Packet Capture
47. AP Testing: WiFi-Transportation-Testing Video Demonstration
48. UE Testing: WiFi-Gaming Demonstration Video
49. UE Testing: Setting Up a RADIUS server
50. UE Testing: Setting Up WPA3
51. UE Testing: Wifi Roaming With Opportunistic Key Caching (OKC)
52. UE Testing: Using iperf to Generate Traffic
53. UE Testing: Using flent to Generate Traffic
54. AP-Mode Corruptions: Overriding RSN Information Elements of Beacon Frames
55. AP-Mode Corruptions: Overriding RSN Information Elements of EAPOL-Key Message 3/4
56. AP-Mode Corruptions: Overriding RSNXE Information Elements of EAPOL-Key Message 3/4
57. AP-Mode Corruptions: Overriding SAE Commit Message
58. AP Testing: Inflight-Entertainment Video
59. AP Testing: Configuring EAP-PEAP and MSCHAPv2
60. Chamber View: Basic AP Testing
61. Chamber View: Setting up Attenuator Connections
62. Chamber View: Advanced VLAN AP Scenario

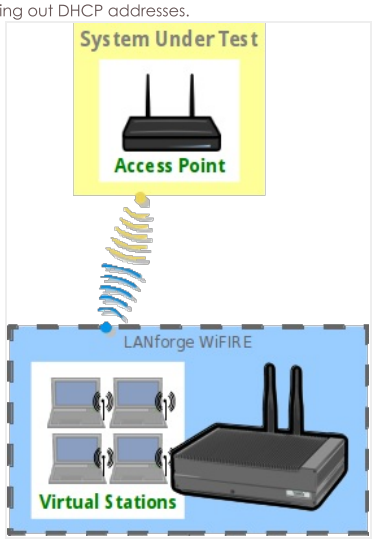
63. Chamber View: Receiver-Sensitivity Testing With LANforge
64. Chamber View: Dataplane Test: Throughput Over Different Packet Sizes
65. Chamber View: Dataplane Test: Throughput Over Different AP Orientations
66. Chamber View: Rate-vs.-Range Test: Throughput Over Different Signal Strengths
67. Chamber View: Hunt Test: Latency Over Different Packet Sizes
68. Chamber View: WiFi-Capacity Test: Throughput With Many Stations
69. Chamber View: WiFi-Station-Reset Test
70. Chamber View: Compare Multiple APs' Performance With Dataplane
71. Chamber View: Roam Test With 802.11k/v/r
72. Chamber View: Automating Tests With Script
73. AP-Auto Tests: Auto-AP Test: Setup
74. AP-Auto Tests: Auto-AP Test: Stability
75. AP-Auto Tests: Auto-AP Test: Capacity Test
76. AP-Auto Tests: Auto-AP Test: Throughput Test
77. AP-Auto Tests: Auto-AP Test: Band Steering Test
78. AP-Auto Tests: Auto-AP Test: Long-Term Test
79. TR-398 Tests: TR-398 Issue 1 Test: Receiver Sensitivity, Throughput, Capacity
80. TR-398 Tests: TR-398 Issue 2 & Mesh: Calibration, Max Throughput Test
81. TR-398 Tests: TR-398 Issue 2/3 & Mesh: Advanced Configuration
82. Chamber View: Automated Mesh-Topology Test
83. Chamber View: Creating Simple LANforge Access Points
84. Chamber View: Create 802.11k/v/r LANforge Access-Point Cluster
85. Chamber View: Automated CI/CD Regression Testing
86. Scripts: Python Script to Show Latency and Throughput at Different QoS
87. Scripts: Python Script to Create Many Connections and Capture Packets
88. AP Testing:Using the Port Monitor
89. Chamber View: CI/CD Lights-Out Chamber Setup
90. Scripts: Creating Test Scripts With the Realm Class
91. Scripts: Creating Layer-4 Test Scripts With Python
92. Scripts: Creating Generic Test Scripts With Python
93. Scripts: LAN-Theoretical-Throughput Calculator for 802.11a/b/g/n/ac Standards
94. Chamber View: Initial Setup to Run Scripts Test Suite for AP Testing
95. UE Testing: Mobile STA Mesh Test -TPUT/ROAM
96. Interop: Test Video Streaming with DASH Server and Interop APK
97. Interop: Setup LANforge InterOp for Android
98. Interop: Installing LANforge Server on Windows Machine
99. Interop: Installing LANforge Server on MacOS Machine
100. Interop: Installing LANforge Server on Linux and Ubuntu Machine
101. Interop: Android Real-Browser Test
102. Interop:Android Layer-3 Traffic Test
103. Interop:Video Streaming Test on Android
104. Interop: Setup LANforge InterOp for iOS

Generating Traffic for WLAN Testing

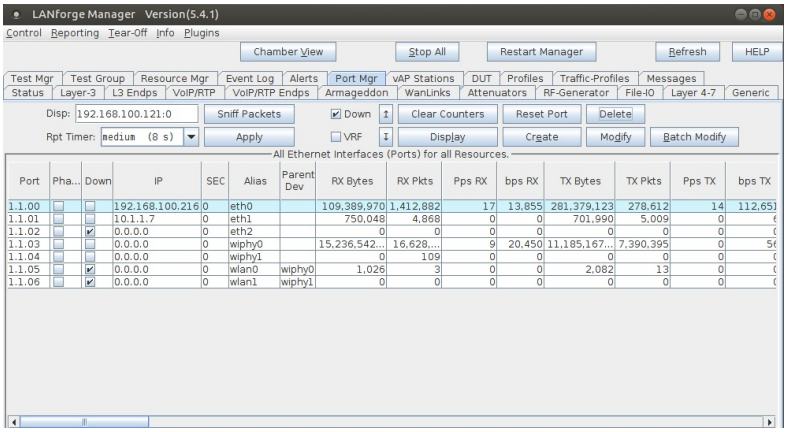
Goal: Setup and run Wireless LAN traffic using the LANforge CT523 or similar system.

In this test scenario, the LANforge CT523 is used to simulate 4 virtual wireless stations that associate with a third party access point. Three traffic tests will be configured and run to demonstrate possible wireless access point tests. **NOTE:** This cookbook assumes that you have already created a VAP, and have an interface that is handing out DHCP addresses.

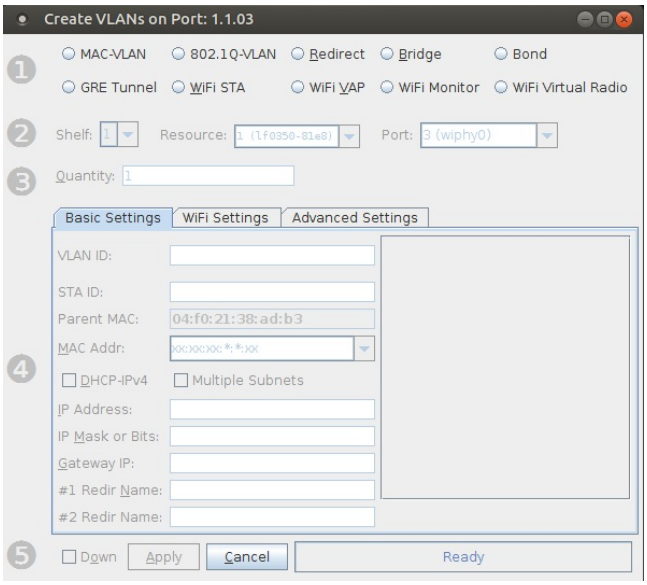




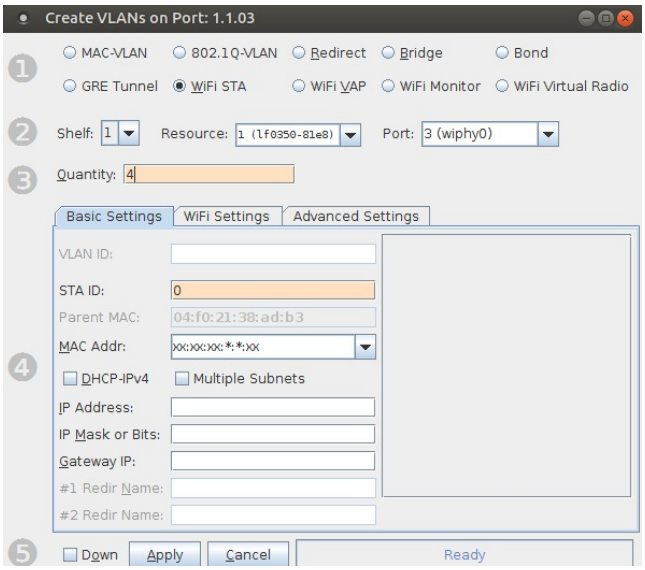
1. Create the virtual wireless stations.
   Note: All of its virtual stations will use the same wireless AP in this example, but each station may be configured for a different AP as long as all stations on the same radio use APs on the same channel.
   A. Go to the Port Manager



B. Select port wiphy0 and click Create



C. Select the **Wifi STA** button, then enter **MAC**, **Quantity**, **STA ID**, and **SSID**. Select the DHCP-IPv4 checkbox:



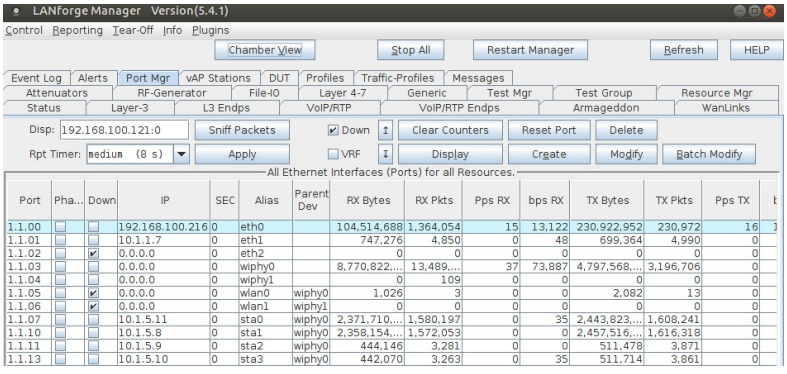
A. In this example, all 4 virtual stations will connect to the same access point

B. If your access point can serve DHCP, you can select the 'DHCP-IPv4' checkbox here to enable each virtual station as a DHCP client

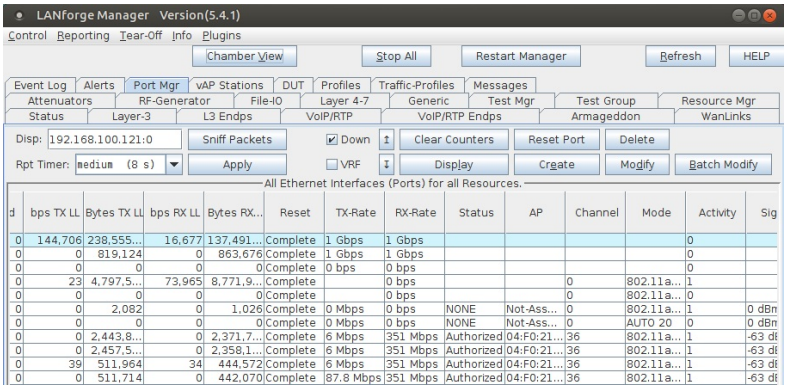
C. If you choose to enter IP addresses manually, the create function will increment the last octet of the IP address for each virtual station created

D. Click **Apply** when finished

D. Verify that the virtual wireless stations are created



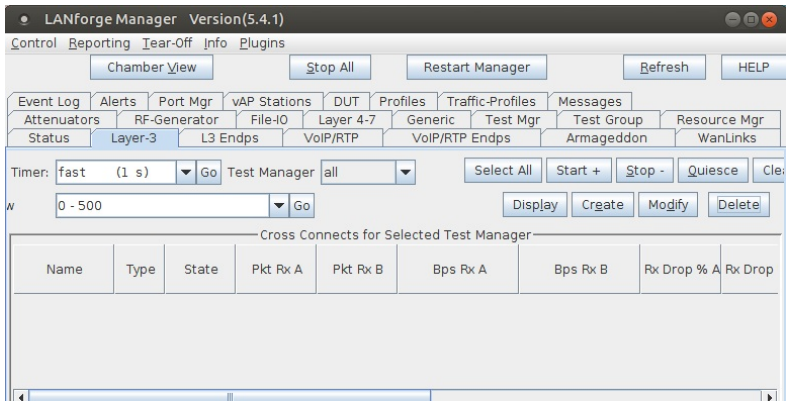
E. Scroll to the right to view each station's link quality and other interface details



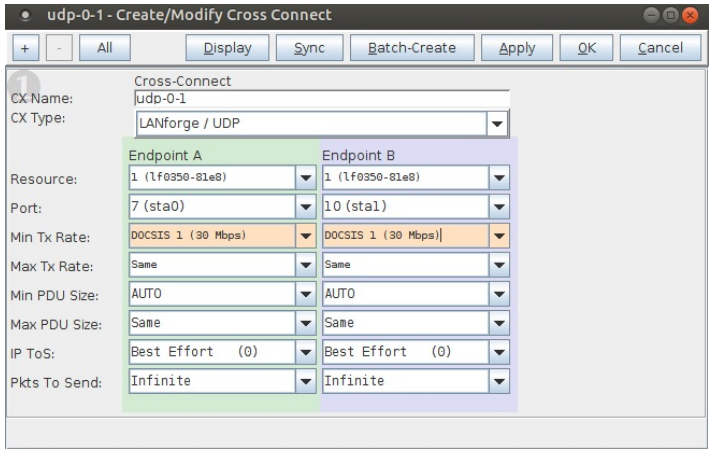
d	bps TX LL	Bytes TX LL	bps RX LL	Bytes RX...	Reset	TX-Rate	RX-Rate	Status	AP	Channel	Mode	Activity	Sig
0	144,706	238,555...	16,677	137,491...	Complete	1 Gbps	1 Gbps					0	
0	0	819,124	0	863,676	Complete	1 Gbps	1 Gbps					0	
0	0	0	0	0	Complete	0 bps	0 bps					0	
0	23	4,797,5...	73,965	8,771,9...	Complete		0 bps			0	802.11a...	1	
0	0	0	0	0	Complete		0 bps			0	802.11a...	0	
0	0	2,082	0	1,026	Complete	0 Mbps	0 bps	NONE	Not-Ass...	0	802.11a...	1	0 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Ass...	0	AUTO 20	0	0 dBm
0	0	2,443,8...	0	2,371,7...	Complete	6 Mbps	351 Mbps	Authorized	04:F0:21:...	36	802.11a...	1	-63 dB
0	0	2,457,5...	0	2,358,1...	Complete	6 Mbps	351 Mbps	Authorized	04:F0:21:...	36	802.11a...	1	-63 dB
0	39	511,964	34	444,572	Complete	6 Mbps	351 Mbps	Authorized	04:F0:21:...	36	802.11a...	1	-63 dB
0	0	511,714	0	442,070	Complete	87.8 Mbps	351 Mbps	Authorized	04:F0:21:...	36	802.11a...	1	-63 dB

For more information see LANforge User's Guide: Ports (Interfaces)

2. Create Layer-3 connections between the station interfaces.
    A. Go to the **Layer-3** tab and click **Create**

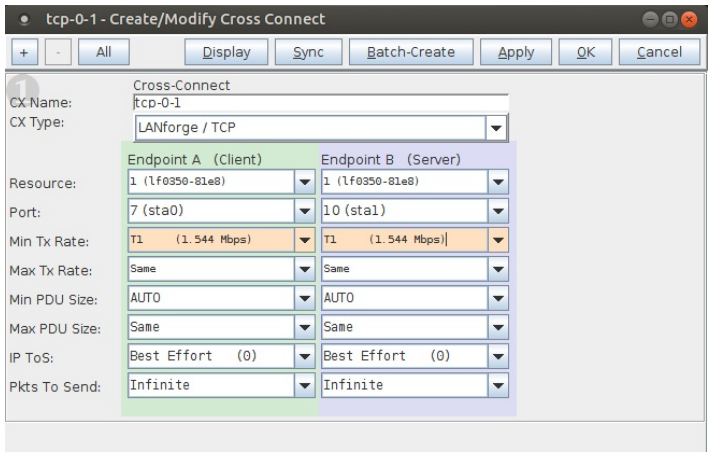


    B. Create a station-to-station UDP speed test:



    A. **Test 1**: sta0-sta1, UDP, 30Mbps

    C. Create a station-to-station TCP speed test:

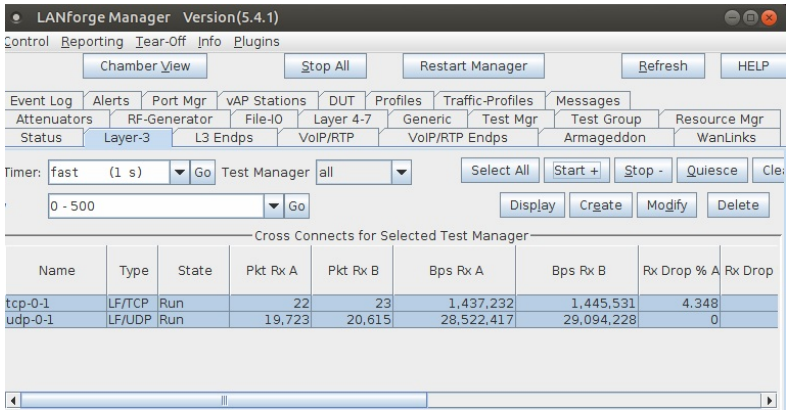


    A. **Test 1**: sta0-sta1, TCP, 1.54Mbps bi-directional traffic

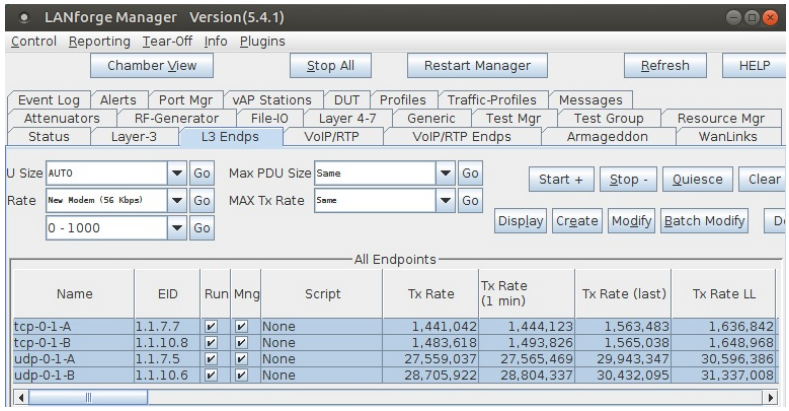
    For more information see LANforge User's Guide: Layer-3 Cross-Connects (FIRE)

3. Run traffic tests concurrently, and view results.

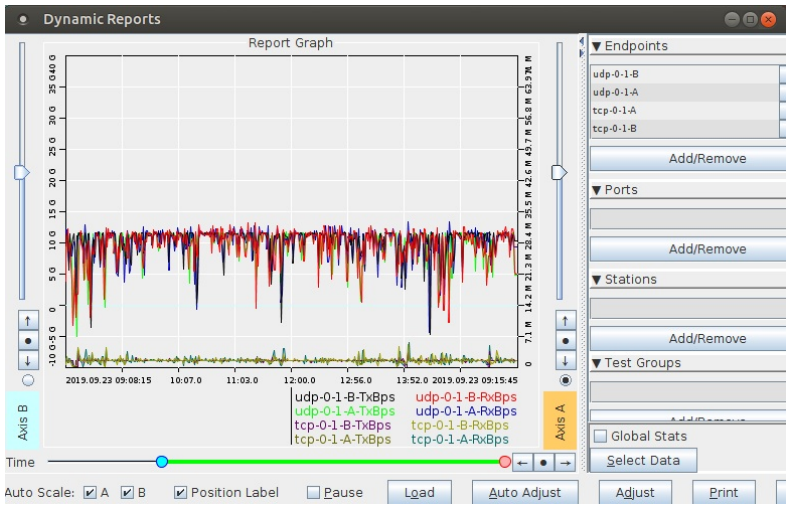
A. This example shows little packet loss, however being that this is traffic sent wirelessly via stations, their may be interruptions due to busy channel frequencies, if needed, adjust your Tx rate accordingly.



B. The **Layer-3 Endpoints** tab has more detail.



C. Select the cross-connects or endpoints and Right-Click → Dynamic Report on the **L3 Endp** or **Layer-3** table to view a live report of the connections.



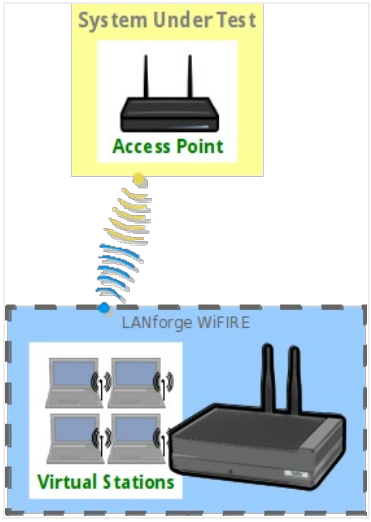
For more information see LANforge User's Guide: Layer-3 Endpoints (FIRE)

For more information see LANforge User's Guide: Reporting

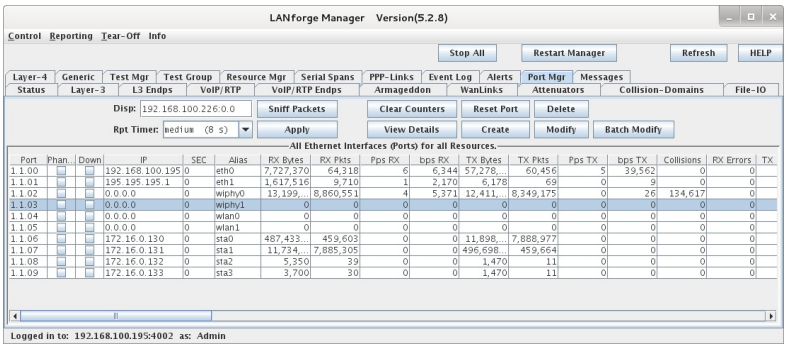
Generating Traffic to saturate a particular WiFi Channel

Goal: Setup and run Wireless LAN traffic using the LANforge CT523 in order to fully saturate a WiFi channel.

In this test scenario, the LANforge CT523 is used to simulate 4 virtual wireless stations on one radio that associate with an access point on the other radio. This allows a single LANforge machine to be a complete WiFi network. Please see the Wifi Testing cookbook example first. These steps assume the configuration from that cookbook is already in place.

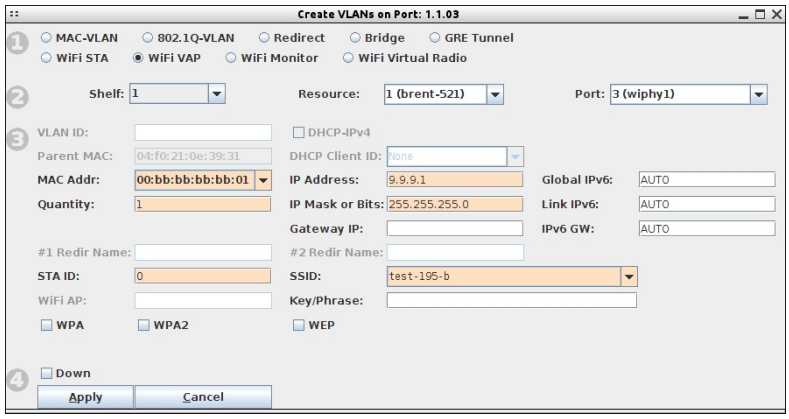


1. Create a virtual AP on wiphy1.
    A. Go to the Port Manager

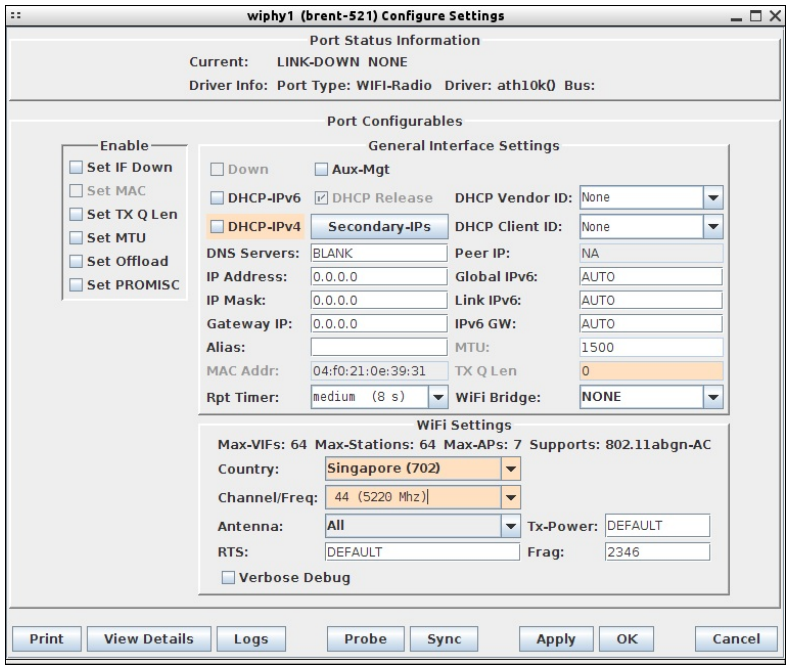


    B. Select port wiphy1 and click Create

C. Select the **Wifi VAP** button, then enter **MAC**, **Quantity**, **STA ID**, and **SSID**. Enter IPv4 address info:



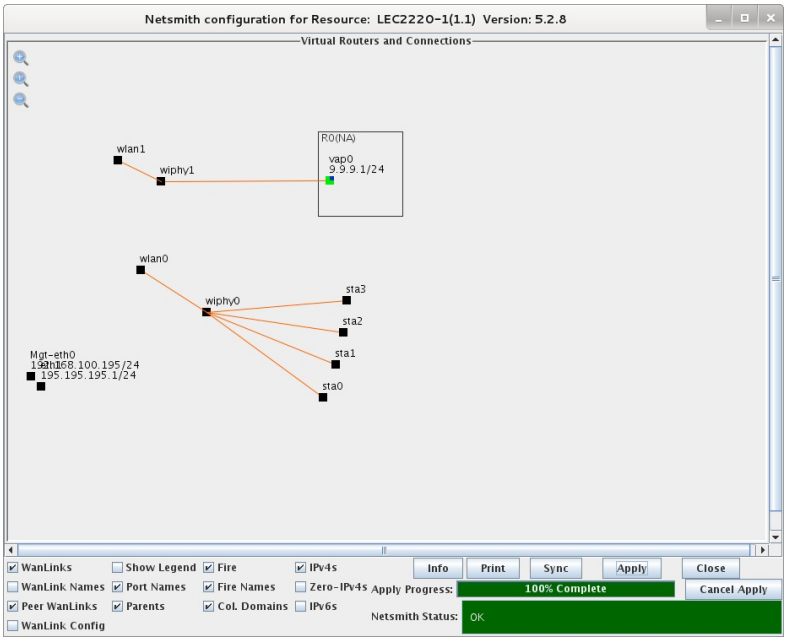
D. Configure the radio's channel (which will apply to the VAP that was just created). Select the wiphy1 interface in the Port-Mgr tab and click **Modify**. Select the channel, and optionally the country-code, and then press **Ok**. If you elect to change the country code, you must do so on all radios in your system in order for proper function.



For more information see LANforge User's Guide: Ports (Interfaces)

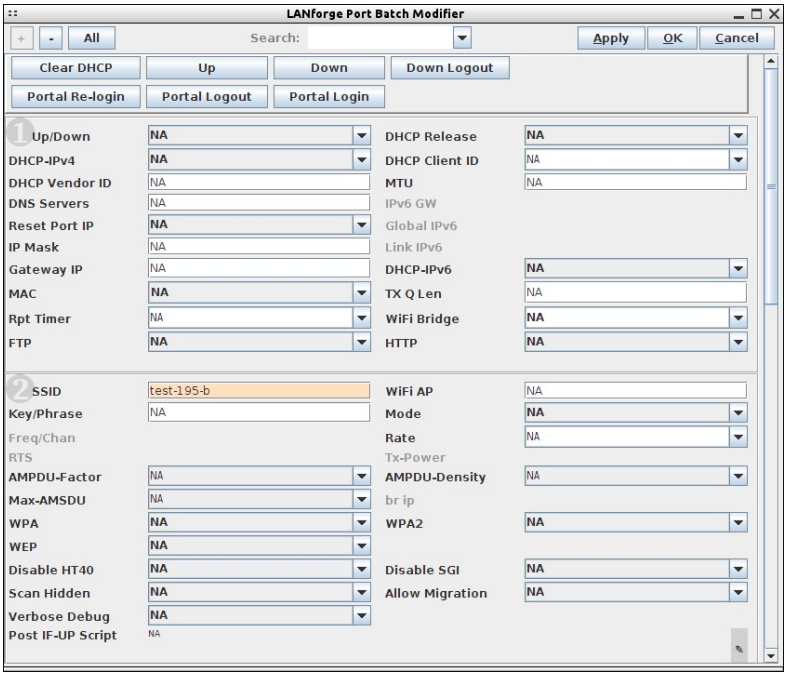
2. Use the Netsmith feature to create a virtual router and configure DHCP for the AP.
   A. Go to the **Status** tab, and click the Netsmith button for Resource 1.
   B. Drag the existing interfaces into a more pleasing layout.
   C. Right-Click in empty space and create a new Virtual Router. Just click OK, using the default configuration.
   D. Drag the VAP interface into the virtual router and click **Apply** at the bottom-right of the Netsmith window..
   E. Double-lick the vap0 icon (which should now be green) to bring up the **Create/Modify Connection** window.
   F. Select the DHCP option, and then fill in the **Range Min** and **Range Max** values appropriately.

G. Apply Netsmith again and the configuration should be done.

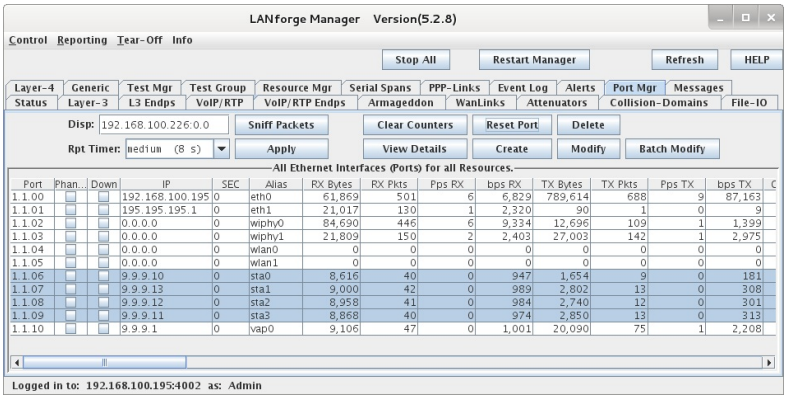


3. Use Batch-Modify to configure all of the existing WiFi stations to talk to the new AP.

A. Go to the **Port-Mgr** tab, select sta0, sta1, sta2, and sta3, and then click the **Batch Modify** button. Configure the SSID to match the AP, and then press OK

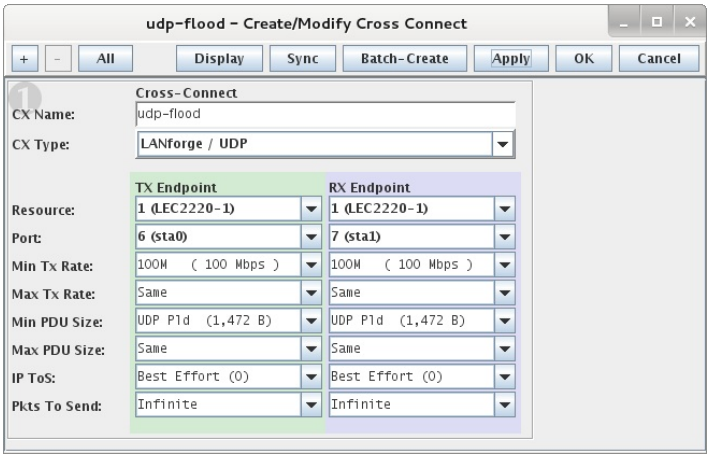


B. The station interfaces should now get DHCP addresses matching the new AP

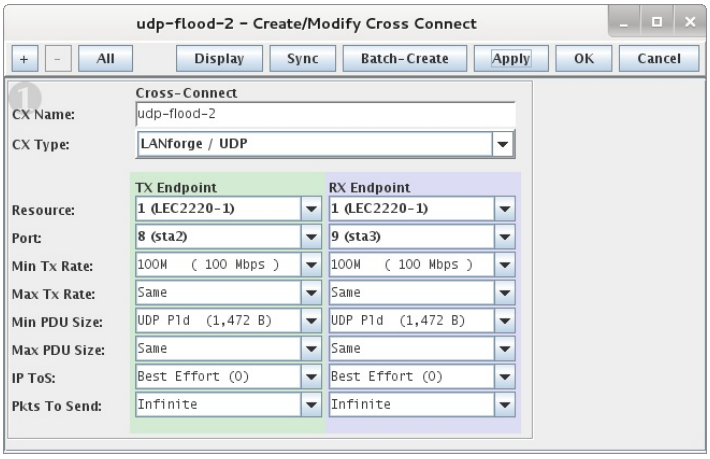


4. Create Layer-3 connections between the station interfaces.

A. Go to the **Layer-3** tab. If there are existing connections, stop and/or delete them, and then click Create. Give the test a name, select sta0 and sta1 for ports, set rate to 100M, and set payload size. When complete, press Apply to create the new CX.



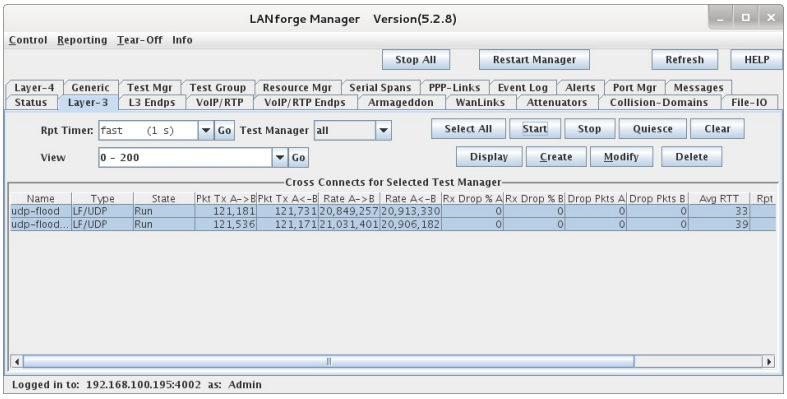
B. Create a second UDP connection on sta2 and sta3 interfaces. Change name, change ports, and press OK.



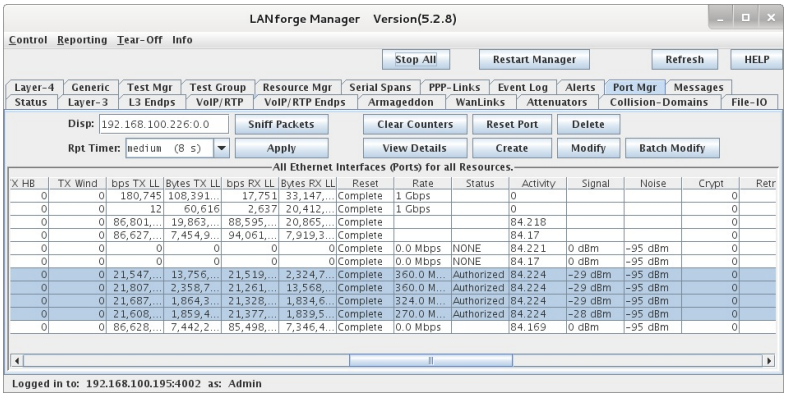
For more information see LANforge User's Guide: Layer-3 Cross-Connects (FIRE)

5. Run traffic tests concurrently, and view results.

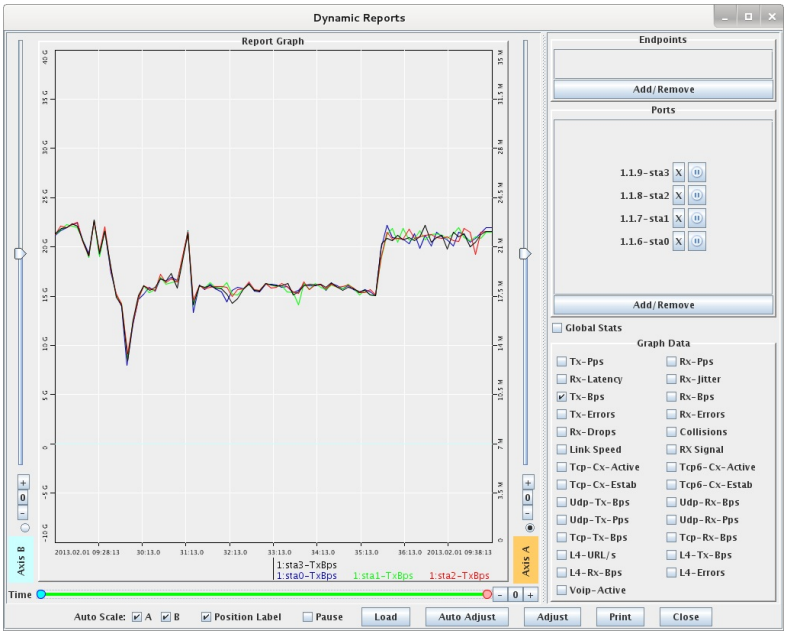
A. Select both CXs in the **Layer-3** tab, and click the **Start** button.



B. Go to the **Port-Mgr** tab, scroll to the right, and confirm that the Activity for this channel reports a fairly high percentage.



C. Select the 4 station ports and Right-Click → Dynamic Report on the **Port-Mgr** table to view a live report of the ports.

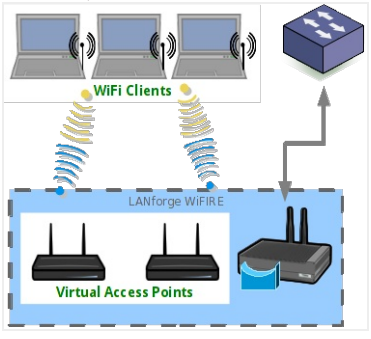


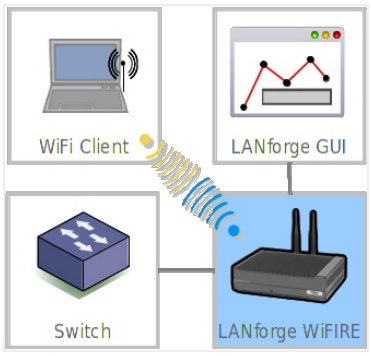
For more information see LANforge User's Guide: Reporting

Create Virtual AP in Bridge Mode

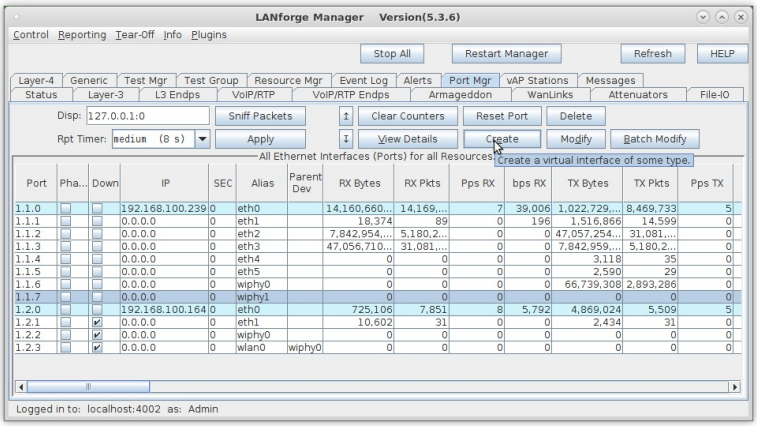
Goal: Create a Virtual AP and set it up to bridge with the wired Ethernet port.

Create a Virtual AP and set it up to bridge with the wired Ethernet port. This example uses the LANforge CT523 but the procedure should work on all CT521, CT522, CT523, CT525 and similar systems.

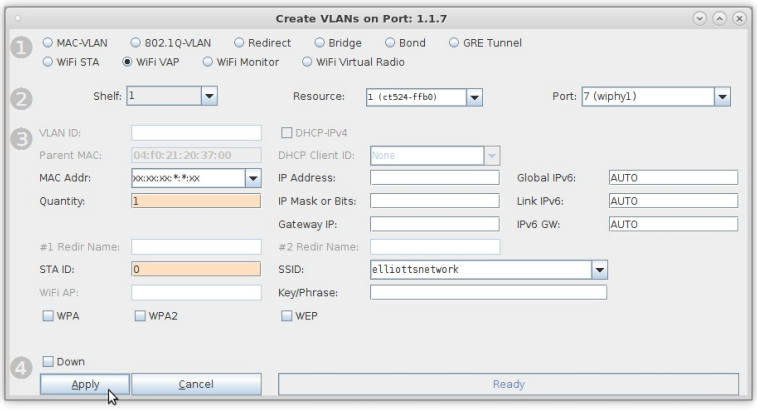




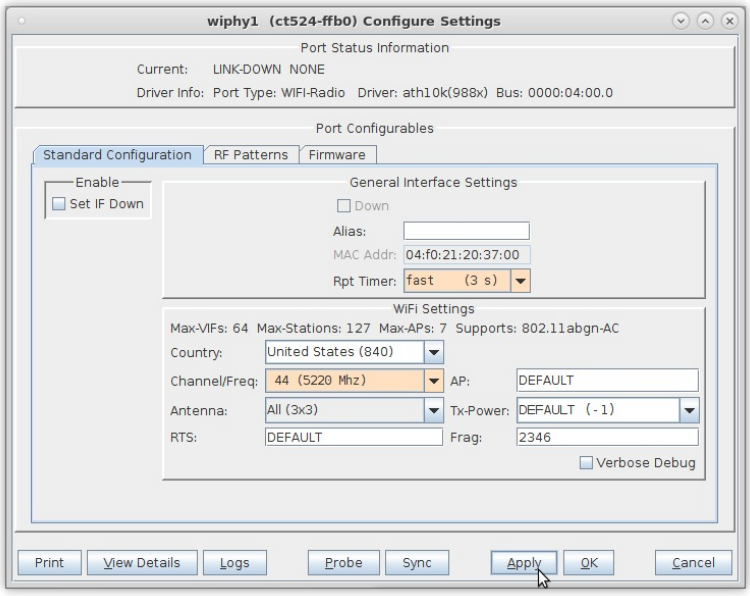
1. Create a virtual AP on wiphy1.
   A. Go to the Port Manager



   B. Select port `wiphy1` and click **Create.**
   C. Select the **Wifi VAP** button, then enter **MAC, Quantity, STA ID,** and **SSID:**



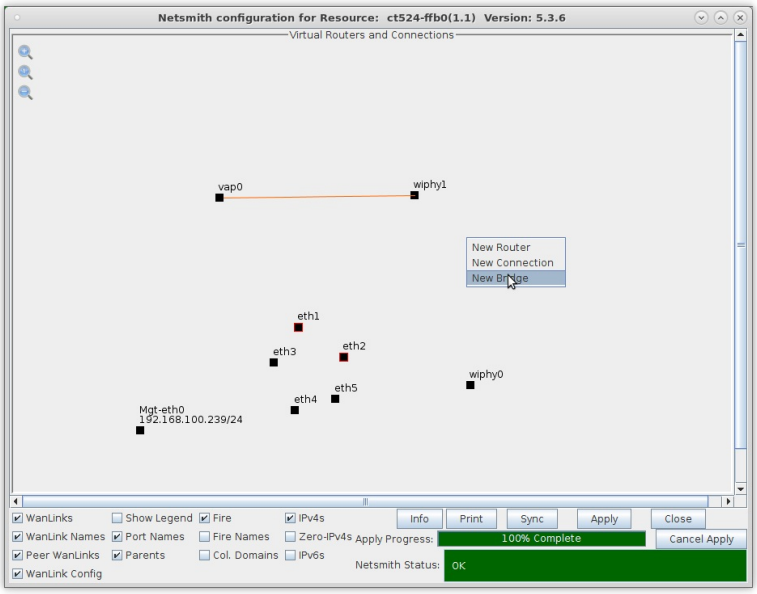
   D. Configure the radio's channel (which will apply to the VAP that was just created). Select the `wiphy1` interface in the Port-Mgr tab and click **Modify**. Select the channel, and optionally the country-code, and then press **OK**.



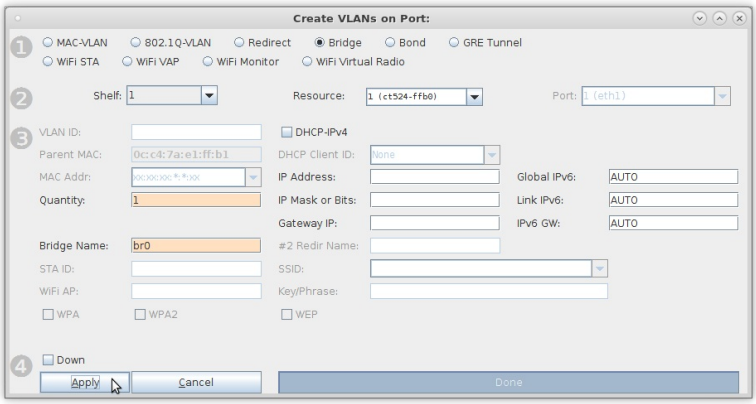
For more information see LANforge User's Guide: Ports (Interfaces)

2. Use the Netsmith feature to create a bridge device to hold the VAP and the Ethernet Port.
   A. Go to the **Status** tab, and click the **Netsmith** button for Resource 1.
   B. Drag the existing interfaces into a more pleasing layout and click **Apply**

C. Right-click in empty space and chose the **New Bridge** option.

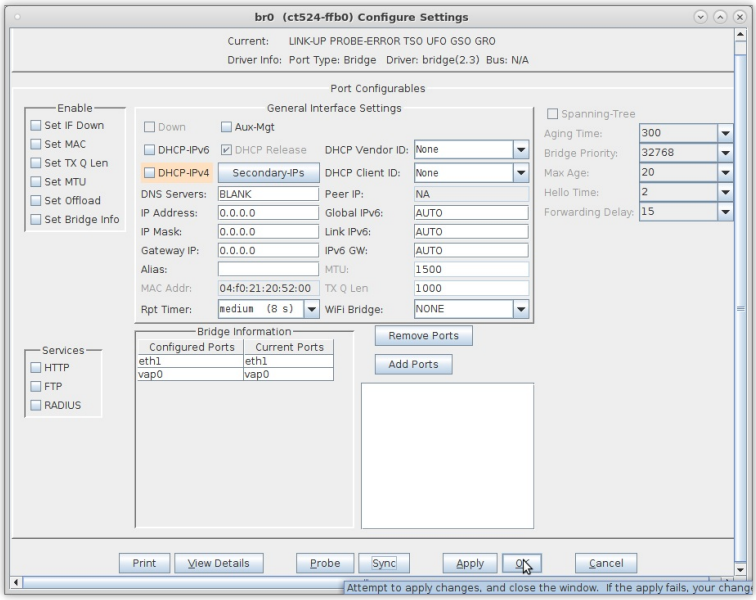


D. Enter Quantity `1`, and give it a name such as `br0`, and then click **Apply**



E. Click **SYNC** in the Netsmith window and the `br0` interface should appear.

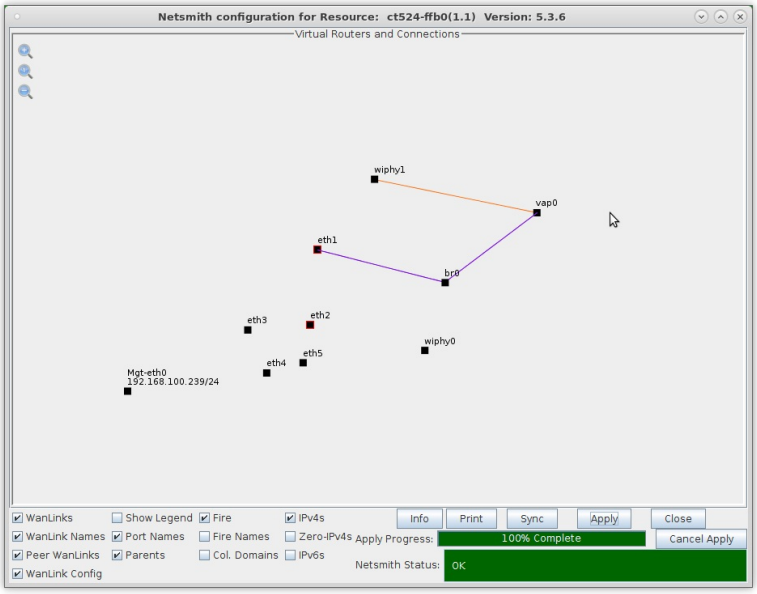
F. Right-click on the `br0` interface and select **Modify Port**



A. Add `vap0` and `eth1` to the bridge.

B. Click **Apply**, and then **Sync**.

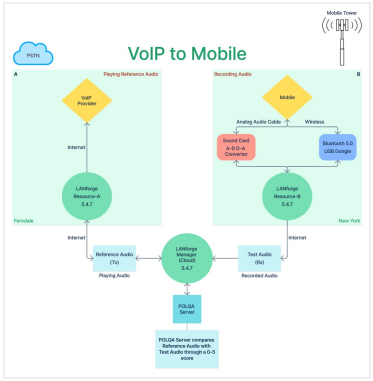
G. Go back to the **Netsmith** window, and you should see the `br0` device connected to `vap0` and `eth1` with purple lines. If you do not, try clicking once on the `br0` icon to force a re-draw. You now have a VAP in bridge mode. Connect the upstream network to `eth1`, and stations associated with `vap0` will be able to communicate with that upstream network.



VoIP-Mobile Audio Quality Testing using POLQA/PESQ

Goal: Evaluate the voice call quality made between VoIP-SIP and Mobile call through POLQA scoring server. PESQ can also be used in place of POLQA here.

Consider an example:
LANforge-A (resource) makes a VoIP-SIP phone call towards a Mobile device which is connected to LANforge-B (resource) via LANforge manager (cloud). LANforge-A plays a reference audio file over the VoIP-SIP call. The call is being recorded by LANforge-B. After the call completes, both the reference audio file and recorded audio file are evaluated by LANforge manager using the POLQA server. The POLQA server scores the recording based on audio quality loss during the call.



1. Requirements:
   A. LANforge systems (version 5.4.7). One manager and atleast one resource.
   B. LANforge licenses
   C. VoIP service provider
   D. POLQA license and server
   E. POLQA standard reference audio files
   F. Mobile Phone having Bluetooth connection and active SIM card.
   G. Bluetooth USB dongle
   H. Analog audio cable
   I. USB sound card
   J. Internet access
   K. Mobile network
2. Configuration:

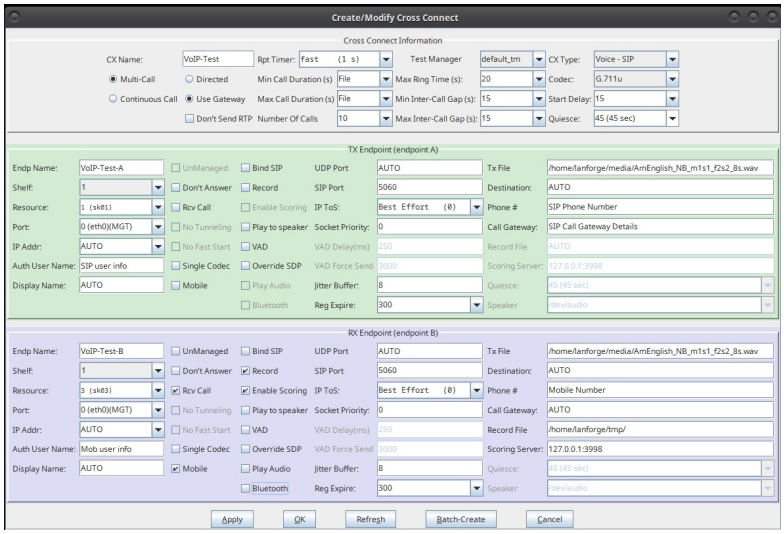
A. Connection between one or more LANforge resource systems towards one LANforge manager (Cloud) should be done till here.

B. LANforge and POLQA licenses are installed.

C. Installation steps: Follow /home/lanforge/audio-bluetooth/README.txt on all LANforge resources.

D. After installation, please `reboot` all the LANforge systems.

E. On the LANforge manager (cloud), open the `GUI`.
In the VoIP/RTP tab, select **Create**.



A. Cross Connect details to be filled are:

I. **TX Endpoint A:** The VoIP-SIP endpoint performs an outbound call over the Internet towards the Mobile phone mentioned in RX Endpoint B. During this call, SIP plays an audio Tx File over the call.

i. `Phone:` SIP Phone info

ii. `Call Gateway:` SIP Call Gateway info

iii. `Auth User Name:` SIP User info

iv. `Resource:` LANforge resource-A (hostname sk01 from Ferndale location in this example)

v. `Port:` Management Port with Internet access (eth0 in this example)

vi. `Display Name:` SIP Name or mac_address

vii. `Tx file:` reference audio file to be played on call

II. **RX Endpoint B:** The Mobile connected to LANforge resource-B receives an inbound call from VoIP-SIP over the Internet. LANforge resource-B starts recording the active incoming audio call via Bluetooth channel or analog audio cable option at the specified Record File location. The recorded file is then evaluated by POLQA server against the original Tx File.

i. `Phone:` Mob phone number

ii. `Call Gateway:` AUTO

iii. `Auth User Name:` AUTO

iv. `Resource:` LANforge resource-B (hostname sk03 from New York location in this example)

v. `Port:` Management Port with Internet access (eth0 in this example)

vi. `Display Name:` Mobile mac_address

vii. `Tx file:` same audio file as TX Endpoint-A Tx File

viii. Checkboxes:

i. `Mobile:` **True**

ii. `Rcv Call:` **True** (become a receiver)

iii. `Bluetooth:` **True** (Record call through Bluetooth channel optional)
`Bluetooth:` **False** (Record call through analog audio cable)

iv. `Record:` **True**

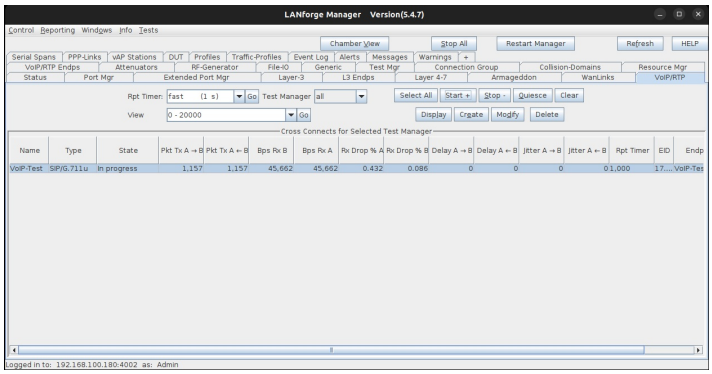
v. `Enable Scoring:` **True** (Enable POLQA scoring)

ix. `Record File:` Recording folder path

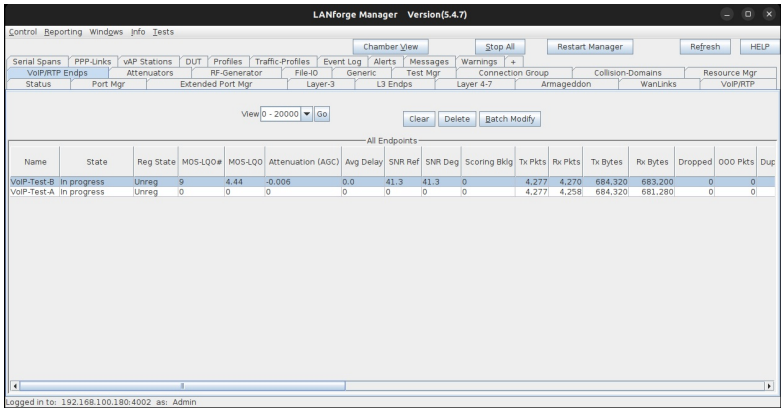
x. `Scoring Server:` POLQA Server Address

B. Select `Apply`, `OK`, and **START** the test.

C.



F. Go To `VoIP/RTP Endps` tab to get the **POLQA Scores.**
(MOS-LQO, Attenuation AGC, Avg. Delay, SNR Reference, SNR Degraded).

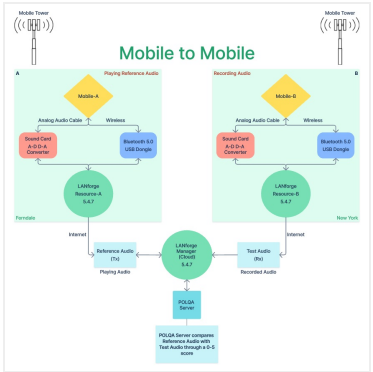


3. Sample POLQA Score Report from POLQA server.
4. If you need assistance, you can contact us at support@candelatech.com

Mobile-Mobile Audio Quality Testing using POLQA/PESQ

Goal: Evaluate the voice call quality made between two Mobile phones through POLQA scoring server. PESQ can also be used in place of POLQA here.

Consider an example:
LANforge-A (resource) makes a Mobile phone call towards another Mobile device connected to LANforge-B (resource). Both the LANforge resources are connected together via LANforge manager (cloud). LANforge resource-A plays a reference audio file over the call through analog audio cable or bluetooth. The incoming call is being recorded by LANforge resource-B through analog audio cable or bluetooth. After the call completes, both the reference audio file and recorded audio file are evaluated by LANforge manager (cloud) through the POLQA server. The POLQA server scores the recording, based on audio quality loss during the call.



1. Requirements:
   A. LANforge systems (version 5.4.7). One manager and atleast one resource.
   B. LANforge licenses
   C. POLQA license and server
   D. POLQA standard reference audio files
   E. Mobile Phones having Bluetooth connection and active SIM cards.
   F. Bluetooth USB dongles
   G. Analog audio cables

H. USB sound cards

I. Internet access

J. Mobile network

2. Configuration:

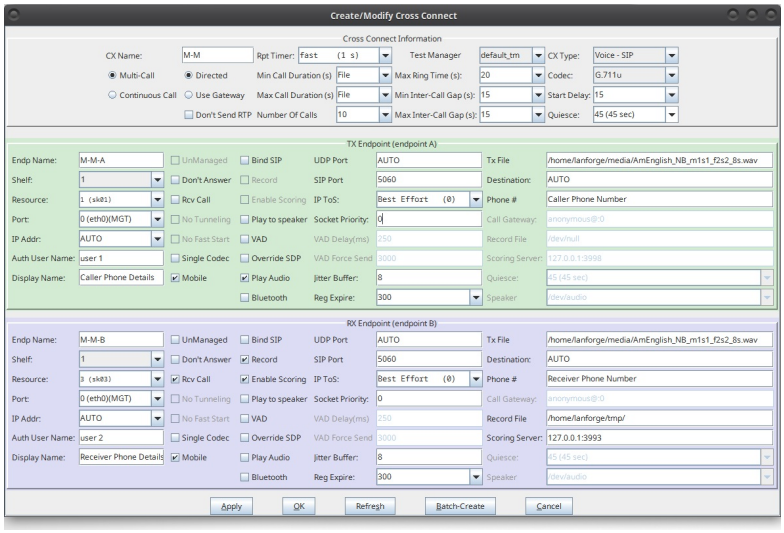
A. Connection between one or more LANforge resource systems towards one LANforge manager (Cloud) should be done till here.

B. LANforge licenses and POLQA licenses are installed.

C. Installation steps: Follow /home/lanforge/audio-bluetooth/README.txt on all LANforge resources.

D. After installation, please `reboot` all the LANforge systems.

E. On the LANforge manager (cloud), open the `GUI`.
In the VoIP/RTP tab, select **Create**.



A. Cross Connect details to be filled are:

I. **TX Endpoint A:** Mobile-A performs an outbound call towards Mobile-B. During this call, Mobile-A plays an audio Tx File over the call which is triggered from LANforge resource-A via analog cable or bluetooth connection.

i. `Phone:` Phone Number A

ii. `Auth User Name:` Mob User A

iii. `Resource:` LANforge resource-A (hostname sk01 from Ferndale location in this example)

iv. `Port:` Management Port with Internet access (eth0 in this example)

v. `Display Name:` Mobile A mac_address

vi. `Tx file:` reference audio file to be played on call

vii. Checkboxes:

i. `Mobile:` **True**

ii. `Rcv Call:` **False** (become a caller)

iii. `Bluetooth:` **True** (Play audio through Bluetooth channel optional)
`Bluetooth:` **False** (Play audio through analog audio cable)

iv. `Record:` **False**

v. `Enable Scoring:` **False**

vi. `Play Audio` **True**

II. **RX Endpoint B:** Mobile-B connected to LANforge resource-B receives an inbound call from Mobile-A. LANforge resource-B starts recording the active incoming audio call from Mobile-B via Bluetooth channel or analog audio cable option at the specified Record File location. The recorded file is then evaluated by POLQA server against the original Tx File.

i. `Phone:` Phone Number B

ii. `Auth User Name:` Mob User B

iii. `Resource:` LANforge resource-B (hostname sk03 from New York location in this example)

iv. `Port:` Management Port with Internet access (eth0 in this example)

v. `Display Name:` Mobile B mac_address

vi. `Tx file:` same audio file as TX Endpoint-A Tx File

vii. Checkboxes:

i. `Mobile:` **True**

ii. `Rcv Call:` **True** (become a receiver)

iii. `Bluetooth:` **True** (Record call through Bluetooth channel optional)
`Bluetooth:` **False** (Record call through analog audio cable)

iv. `Record:` **True**

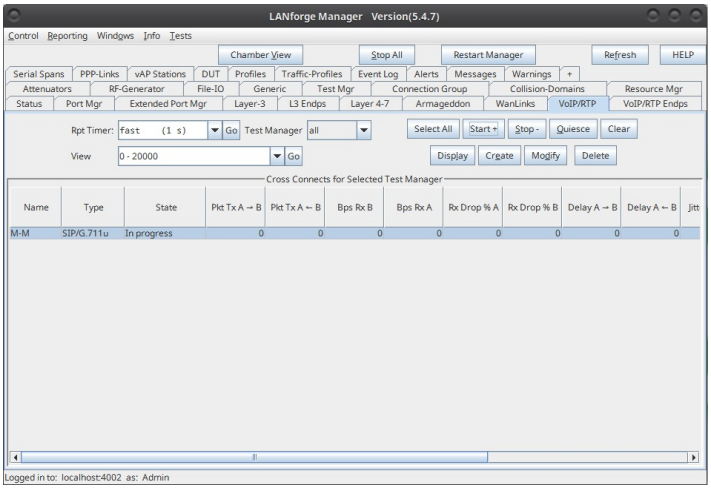
v. `Enable Scoring:` **True** (Enable POLQA scoring)

viii. `Record File:` Recording folder path

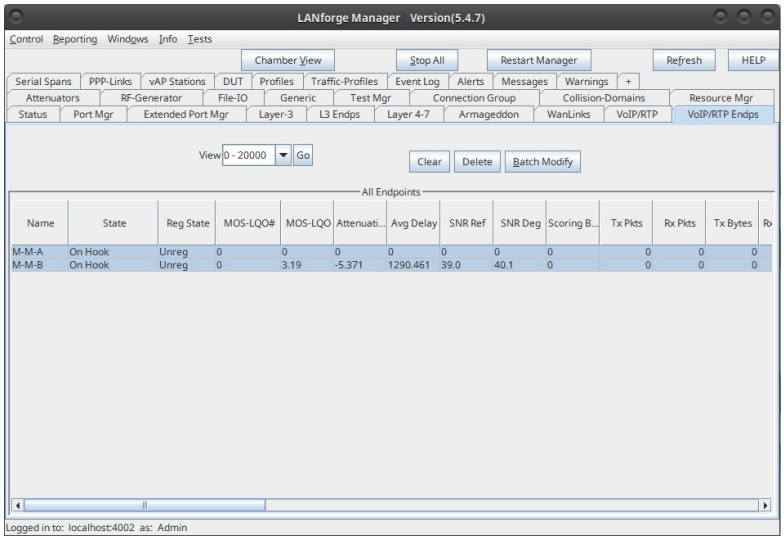
ix. `Scoring Server:` POLQA Server Address

B. Select `Apply`, `OK`, and **START** the test.

C.



F. Go To `VoIP/RTP Endps` tab to get the **POLQA Scores.**
(MOS-LQO, Attenuation AGC, Avg. Delay, SNR Reference, SNR Degraded).

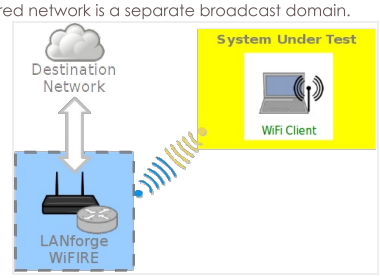


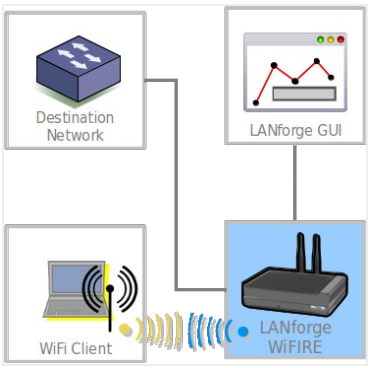
3. Sample POLQA Score Report from POLQA server.
4. If you need assistance, you can contact us at support@candelatech.com

Wifi: Routed Virtual AP with Ethernet Gateway

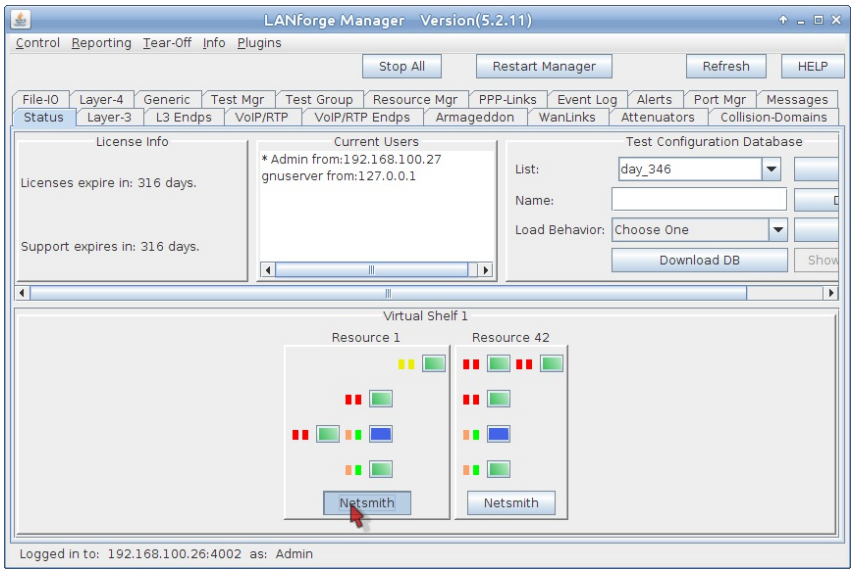
Goal: Configure LANforge CT523 or similar with as a virtual access point with access to an ethernet network.

LANforge WiFi systems can be configured as wireless access points that are gateways to wired networks where the wired network is a separate broadcast domain.

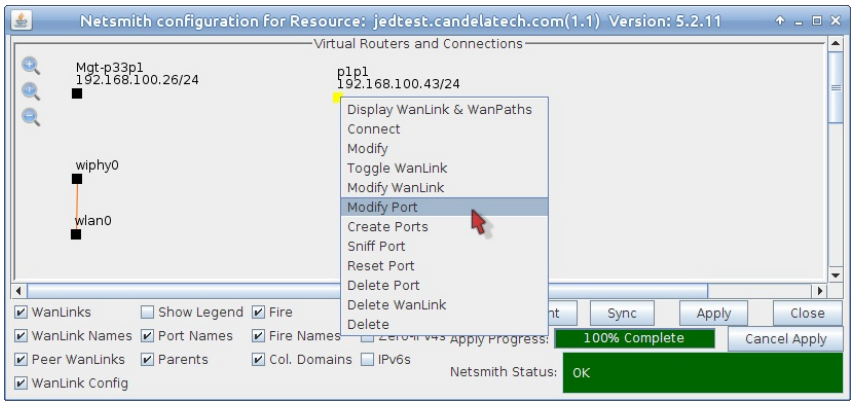




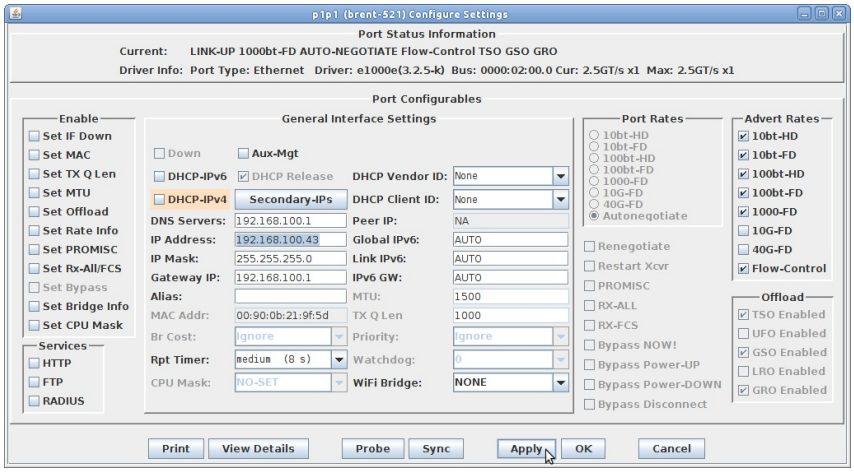
1. Configure the gateway port
2. From the **Status** tab, click **Netsmith**



3. In the **Netsmith** window, right click the ethernet port connected to the destination network and click **Modify Port**



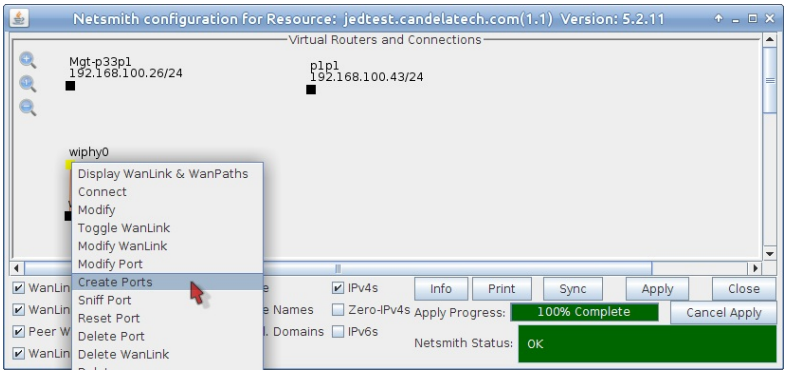
4. Configure the gateway port with these settings:



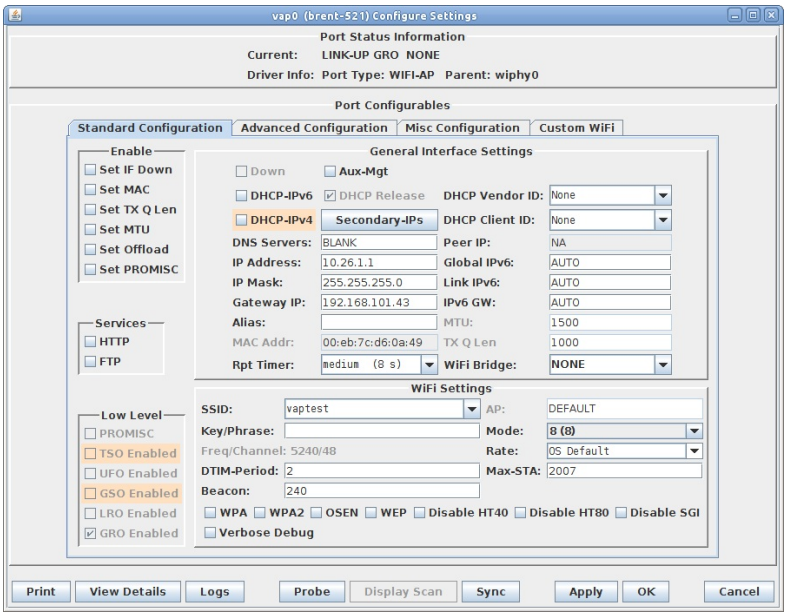
A. DNS Servers: `192.168.100.1`
B. IP Address: `192.168.100.43`
C. IP Mask: `255.255.255.0`
D. Gateway IP: `192.168.100.1`
E. Click **OK** to commit changes.

5. In the **Netsmith** window, click **Sync**
6. NOTE: It is also possible to assign NAT to the outbound gateway port.
   A. From the **Netsmith** screen, you would right click on `p1p1`
   B. Choose **Modify**
   C. Select **NAT**
   D. Click **OK**
   E. Click **Apply** in the **Netsmith** window.

7. Create the virtual access point (AP)
   A. Right click the radio `wiphy0` and select **Create Port**:



B. Configure the AP with these settings:



A. Select **WiFi VAP**

B. Select a MAC Address: `00:eb:7c:d6:0a:49`

C. IP Address: `10.26.1.1`

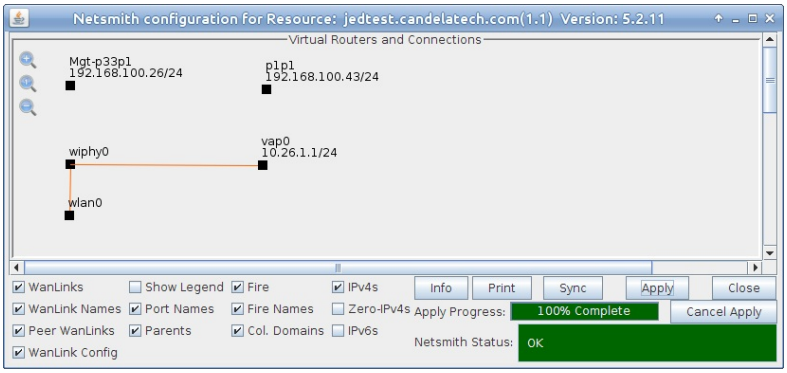
D. IP Mask: `255.255.255.0`

E. Gateway IP: `192.168.100.43`

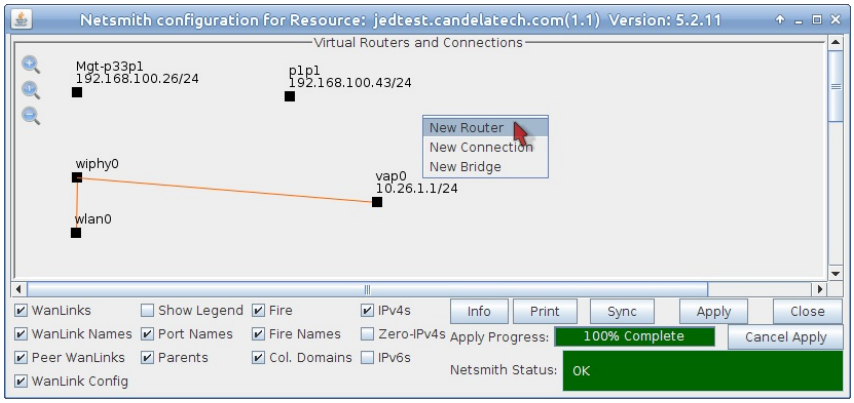
F. STA ID: `0`

G. SSID: `vaptest`

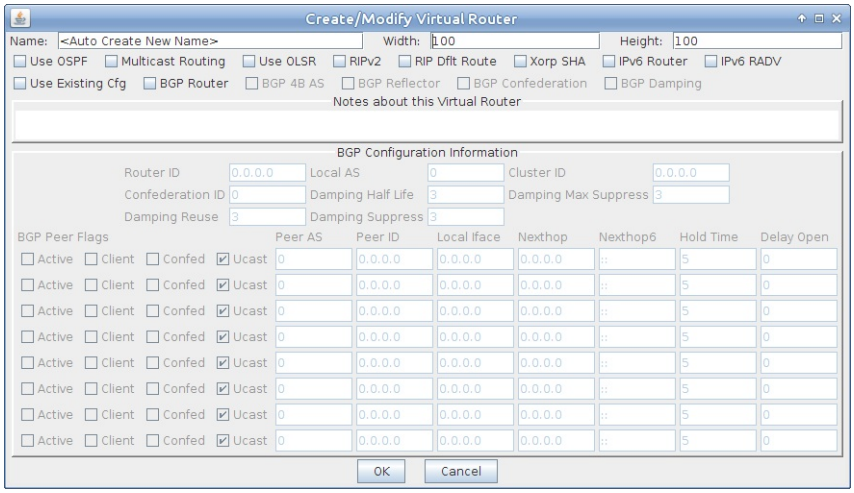
C. In the **Netsmith** window, click **Sync**



8. In the **Netsmith** window, right click and select **New Router**

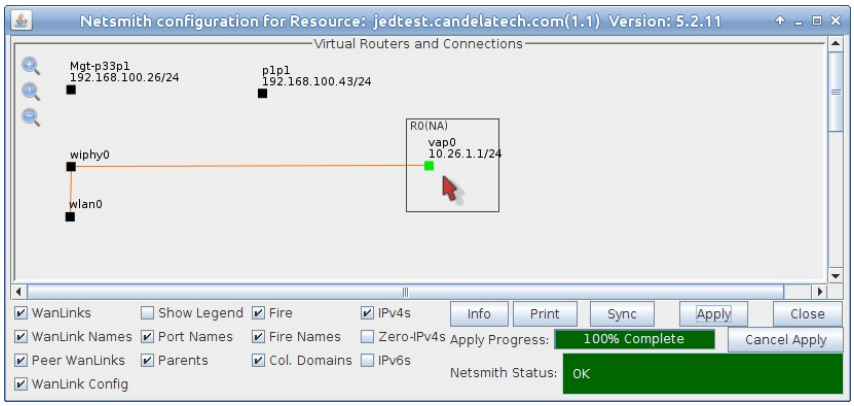


9. Create a router using default settings: click **OK**.



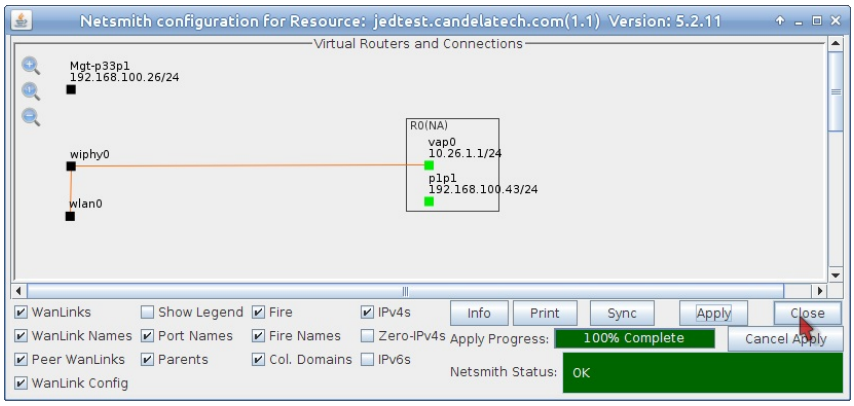
A. In the **Netsmith** window, click **Apply**

10. Drag `vap0` into the virtual router



A. In the **Netsmith** window, click **Apply**

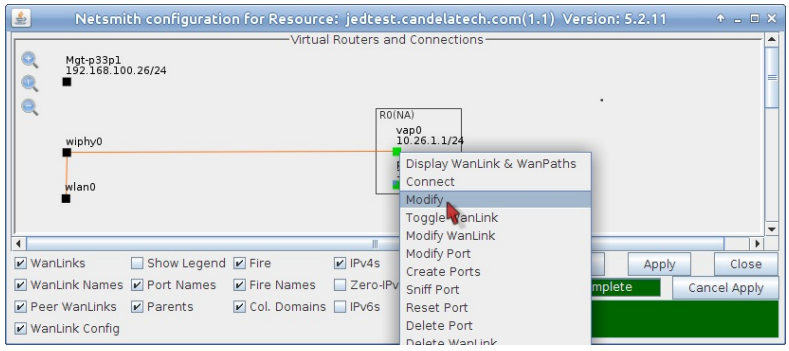
11. Drag `p1p1` into the virtual router



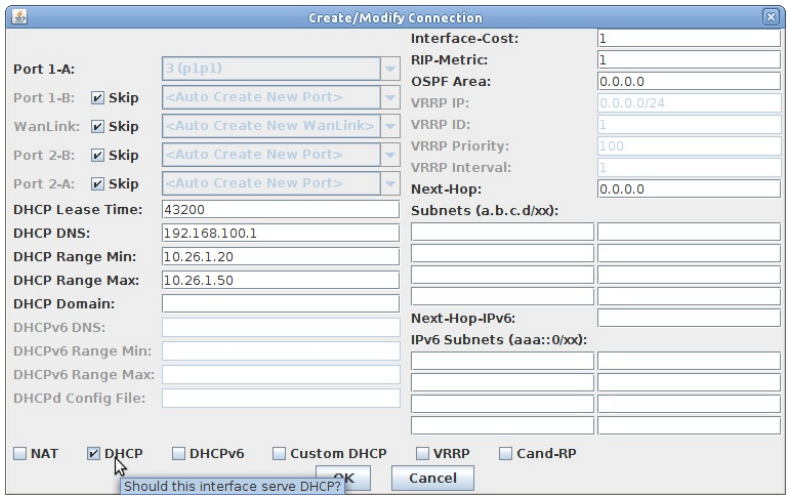
A. In the **Netsmith** window, click **Apply**

12. Assign DHCP to the `vap0` port

A. Right click on `vap0` and select **Modify**



B. Configure DHCP with these settings:



A. DHCP DNS: `192.168.100.1`

B. DHCP Range Min: `10.26.1.20`

C. DHCP Range Max: `10.26.1.50`

D. Select **DHCP**

E. Click **OK** to commit the settings.

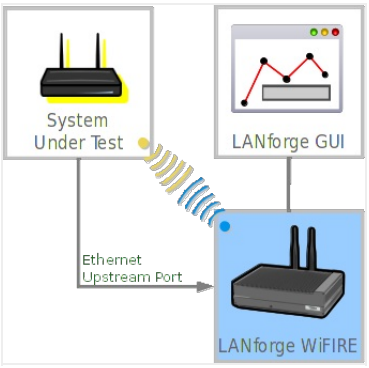
C. In the **Netsmith** window, click **Apply**

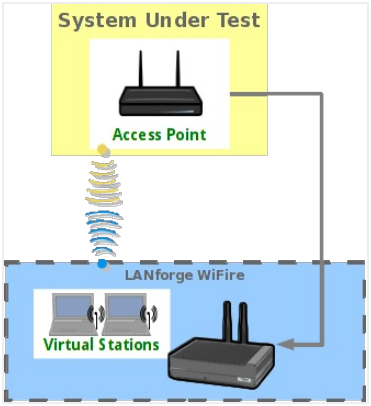
13. NOTE: To test this setup, you could use a laptop with WiFi or a desktop system with a USB WiFi dongle to associate with the Virtual AP. If you have a second radio in your LANforge CT523 you could create a virtual station on one of those extra radios, as well.

Test WiFi station upload throughput.

Goal: Test WiFi station upload throughput with various numbers of stations.

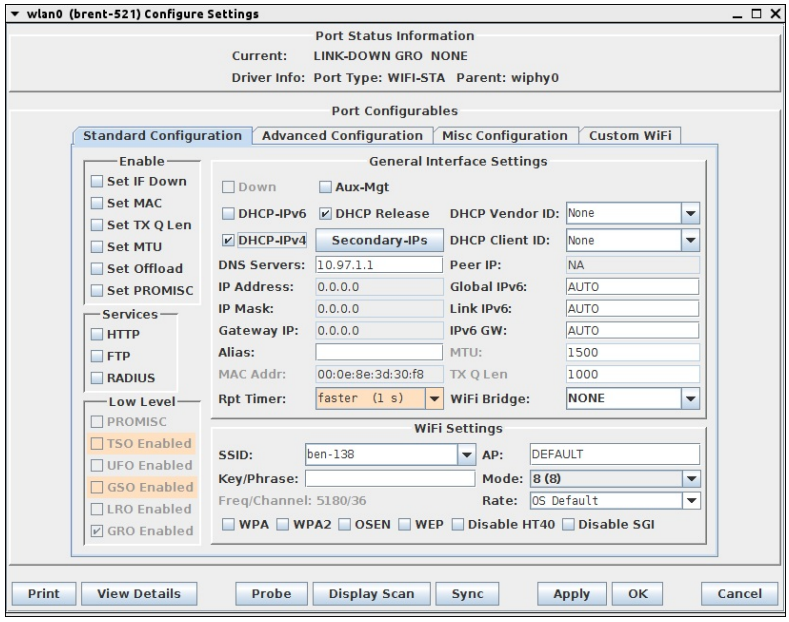
Test WiFi station upload throughput with various numbers of stations. This example uses a pair of LANforge CT520 systems, but the procedure should work on all CT521, CT522, CT523, CT525 and similar systems. Encrypted throughput (WPA2) will perform better on the CT523 and higher systems because their CPUs are more powerful. This test assumes you have already created a virtual AP on system 1 and configured it to act as a router and give out DHCP.



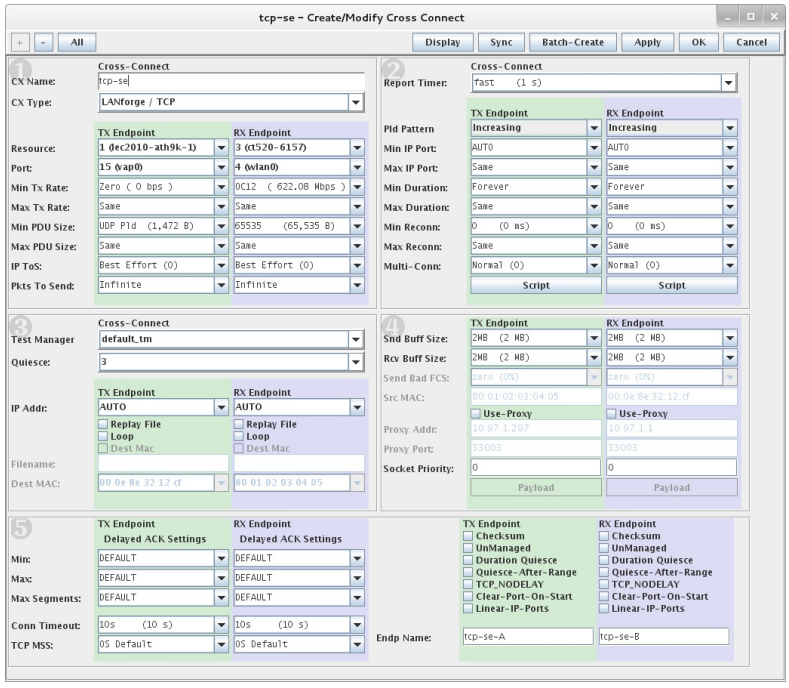


1. Test TCP upload throughput with one station (Open)

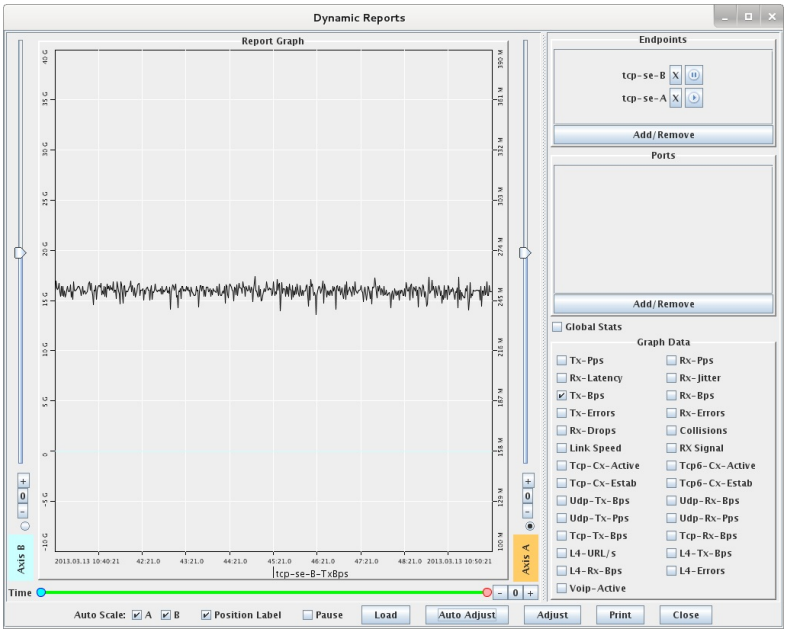
A. Go to the Port Manager, select the **wlan0** interface on the second system, and click **Modify**. Set the SSID to match the VAP on system 1.



B. Go to Layer-3 tab and click **Create** to build a TCP connection. Select the Protocol, ports, rates, PDU sizes, Send & Receive buffer sizes.

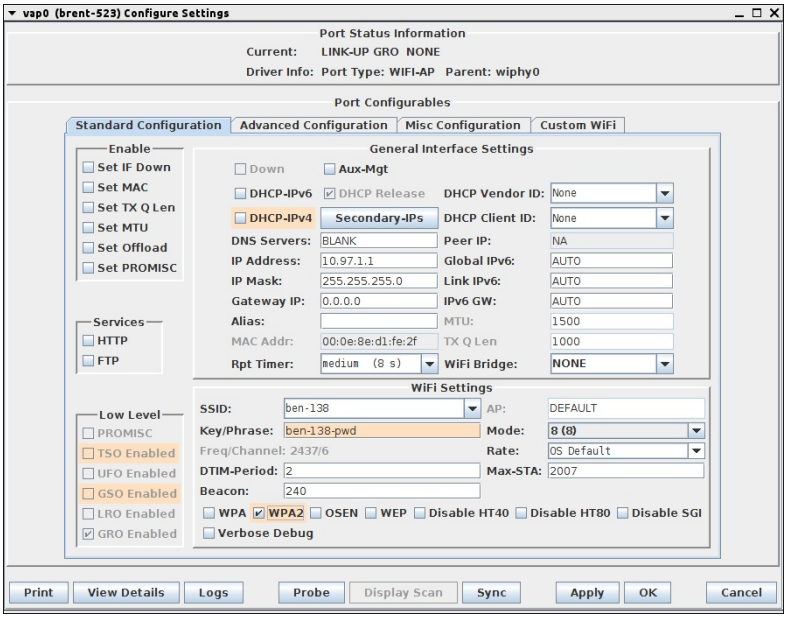


C. Start the test by selecting the **tcp-se** row and click **Start**. Right-click and select **Dynamic Report** to get a real-time graph of the throughput. We see about 250Mbps of TCP throughput.



2. Test TCP upload throughput with one station (WPA2)

A. Modify the VAP to use WPA2 instead of being open. Go to Port-Mgr tab, select vap0, and click **Modify**



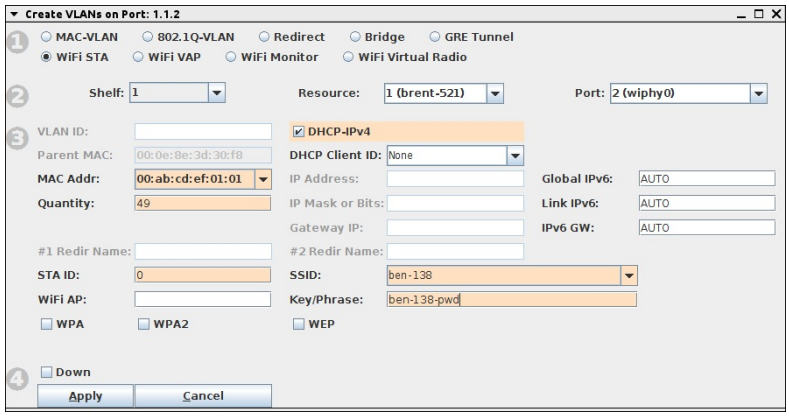
B. Configure wlan0 to use the same settings so it can connect to the VAP using WPA2 authentication.

C. Go to the Layer-3 tab, start the **tcp-se** connection, right-click and select **Dynamic Report** to get a real-time graph of the throughput. We see about 43Mbps of TCP throughput. LANforge currently does the WPA encryption in software on the CPU, so it is much slower than un-encrypted traffic.

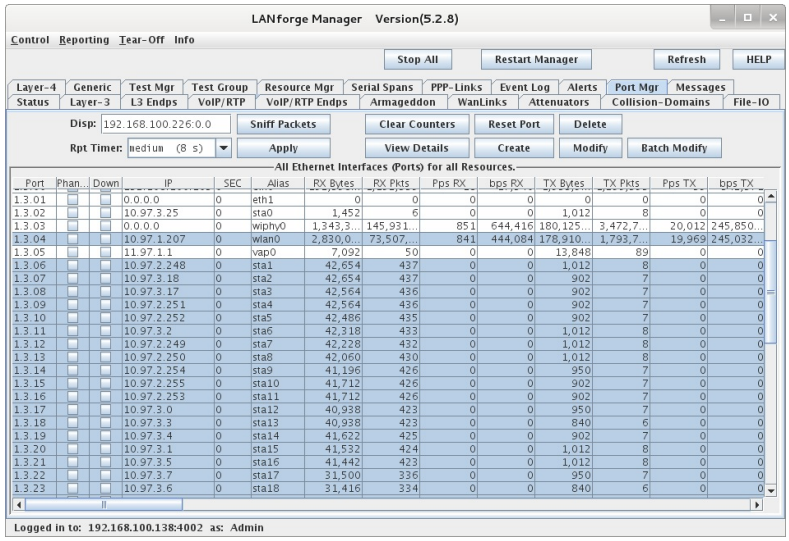
3. Test TCP throughput with 50 stations (Open)

A. Go to the **Port-Mgr** tab, select the existing wlan0 and vap0 interfaces, click **Modify**, un-select WPA2 on each of them, and click Apply. This changes wlan0 and vap0 back to Open mode.

B. Go to the **Port-Mgr** tab, select the wiphy0 interface on the second system, and click **Create**. Select **Wifi STA**, enter starting MAC, quantity of 49, select DHCP-IPv4, STA ID of 0, SSID, and Key/Phrase. When properly configured, click **Apply**

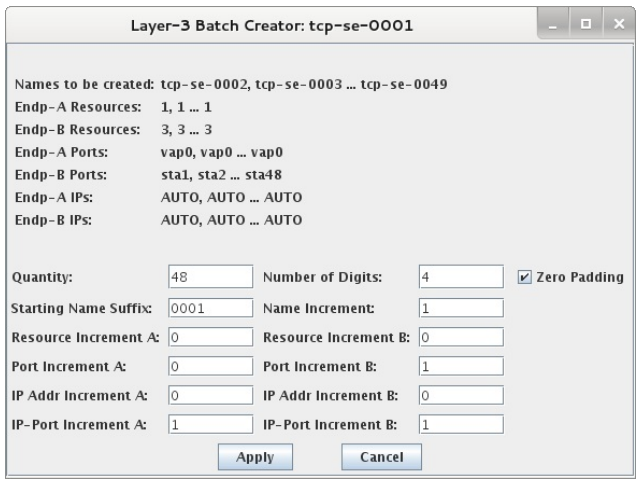


C. You should now see all 50 stations associated with IP addresses.

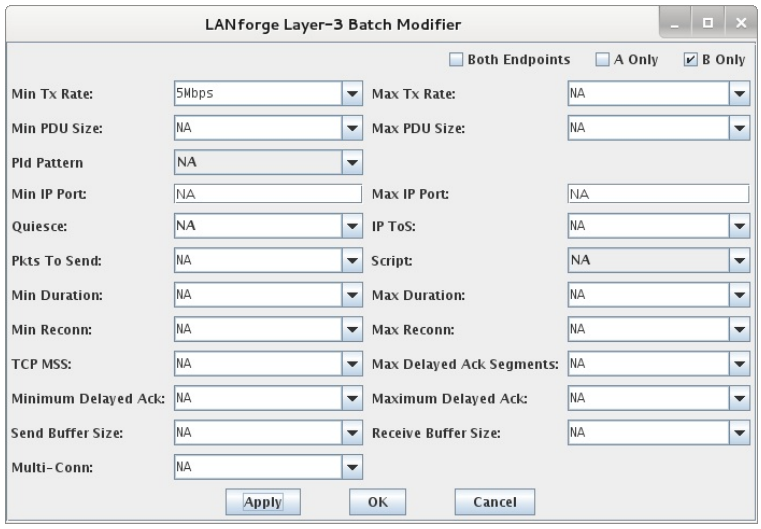


D. Go to the **Layer-3** tab, modify the 'tcp-se' connection. Change the name to tcp-se-0001, and change the RX Endpoint side's port from wlan0 to sta0. Click **OK** to create the new connection.

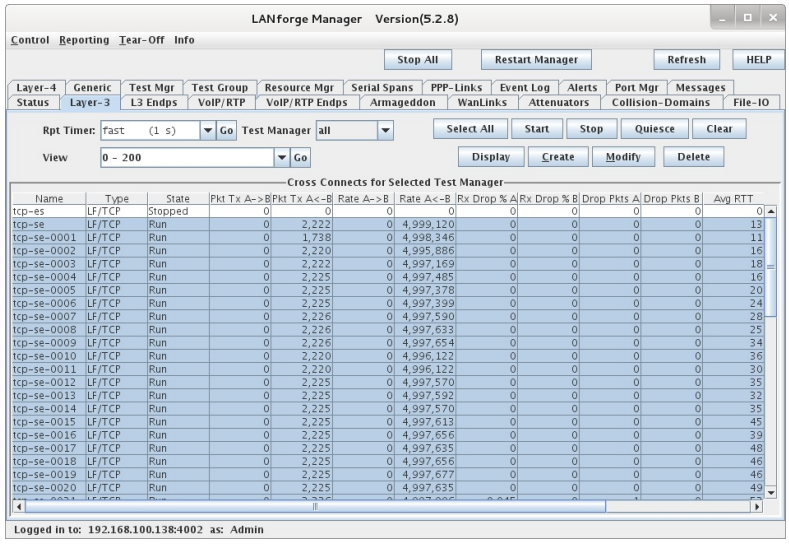
E. Now, create 48 more copies. Modify the tcp-se-0001 connection and click **Batch-Create**. Set quantity to 48, and Port Increment A to 0. Click **Apply** to create the 48 TCP connections, one on each of the 48 stations.



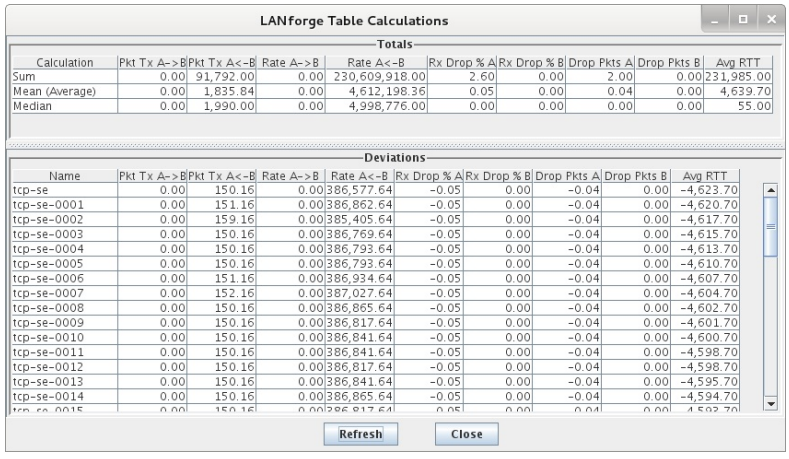
F. Select all of the tcp-se connections and click **Start**. Each of the connections are configured for higher rates than the network can actually handle. This is OK, but it will give un-even throughput results on different stations. So, you may wish to change the rate to something lower. If so, go to the **L3 Endps** tab, make sure all of the tcp-se endpoints are selected, and click **Batch Modify**. Select **B Only** since we want to just modify one side of the connection. Set Min Tx Rate to 5Mbps, and click **Apply**



G. Go back to the **Layer-3** tab and view the running connections.

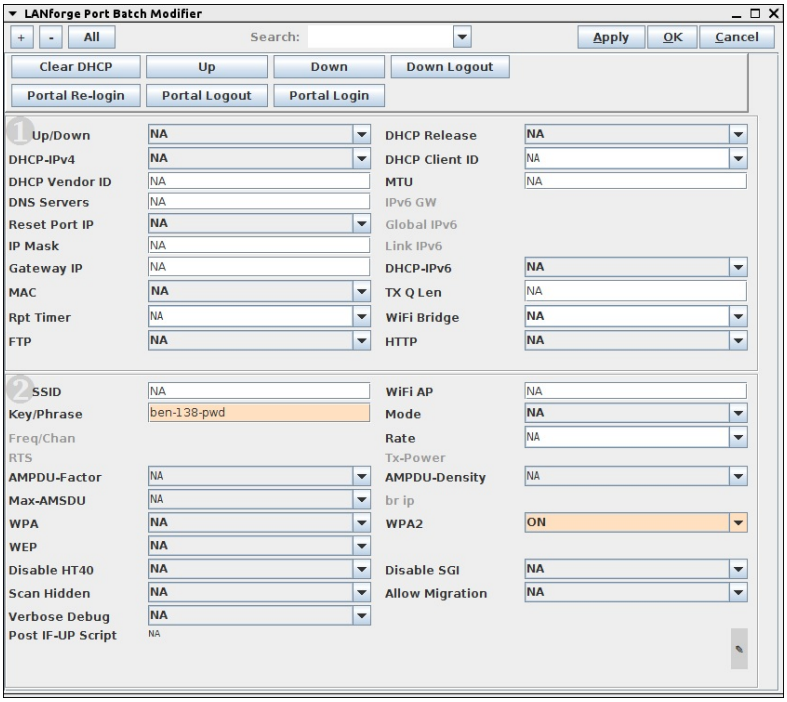


H. Determine total throughput: Select all of the running connections and then Right-Click and select **Calculations**. In this scenario, we see about 230Mbps of total TCP throughput.



4. Test TCP upload throughput with 50 stations (WPA2)

A. Modify the VAP and stations to use WPA2 instead of being open. Go to Port-Mgr tab, select vap0, and all of the stations and click **Batch Modify**. Change WPA2 to be 'ON' and enter password in the Key/Phrase box. Leave all othe values set to 'NA' so they are not changed.



B. Go back to the Layer-3 tab, select all of the connections, and click **Clear** to clear the counters. Right-Click and select Calculations to view the totals. In this scenario, we see about 50Mbps of TCP throughput.

5. Test TCP upload throughput with 128 stations

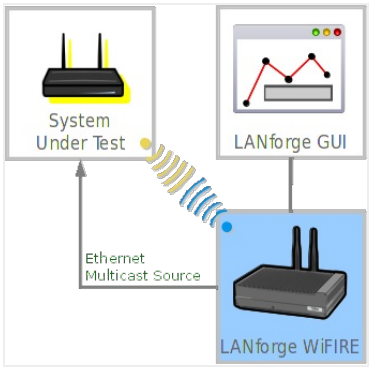
A. Use similar steps to those described above to create 78 more stations. In the Create VLANs window, use STA ID of 49 and Quantity of 78 this time (since stations 0-48 have already been created). When creating more TCP connections, you can click **Modify** on the tcp-se-0049 connection and click **Batch Create**.

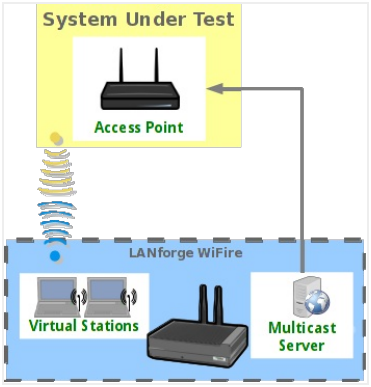
B. In this scenario, we see about 36Mbps throughput with TCP in WPA2 mode, and 28Mbps to 40Mbps in Open mode.

Test WiFi Multicast Download.

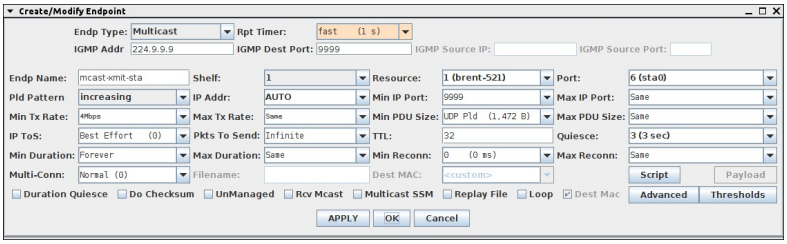
Goal: Test WiFi Multicast Download.

Test WiFi Multicast Download with a small number of stations. This example uses a pair of LANforge CT520 systems, but the procedure should work on all CT521, CT522, CT523 and CT525 systems. Multicast is a bit special in Wireless networks. Packets are normally transmitted at the lowest rate, so even a small bit of multicast traffic will slow down the entire network. In this example, we will transmit multicast frames from the AP to the stations. Many users will instead use the wired Ethernet port as the transmitter, but aside from the different port, the procedure should be the same. This example assumes you have already created and configured the desired amount of WiFi station interfaces.

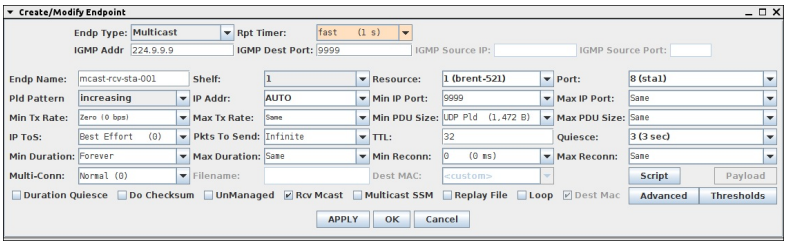




1. Create Multicast transmitter on AP interface
    A. Go to the **L3 Endps** tab, and click **Create**. Set the IGMP Addr, IGMP Dest Port, name, PDU Size, Port, etc and click OK when done.

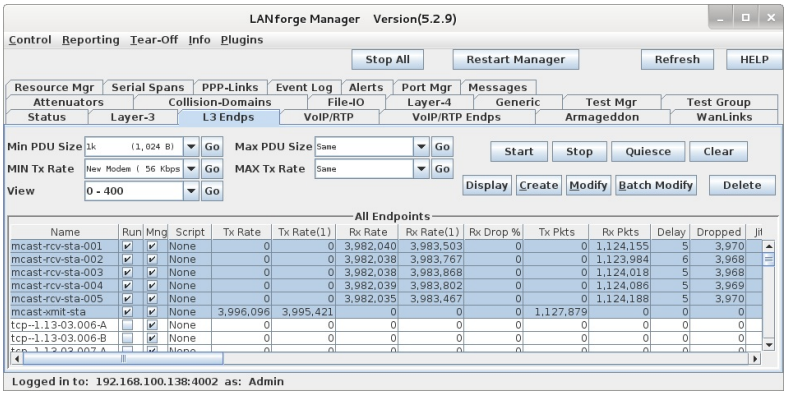


2. Create Multicast receivers on Station interfaces
    A. Go to the **L3 Endps** tab, and click **Create**. Set the IGMP Addr, Min IP Port, name, Port. The IGMP Addr and IP Port should match the transmitter. Make sure you also select the 'Rcv Mcast' checkbox. Click **Apply** when done. You can then change the name, change the Port, and click **Apply** again to create a duplicate endpoint on another station interface.



3. Test throughput

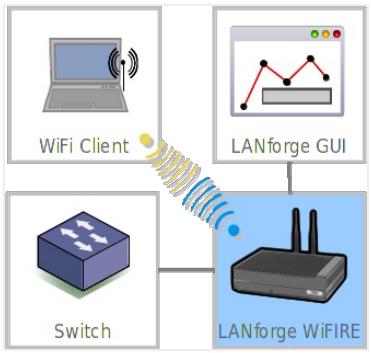
A. Select the Multicast transmitter and receiver endpoints on the **L3 Endps** tab and click **Start**. Observe transmit and receive rates, packet-loss, and other statistics to very performance is at expected value.

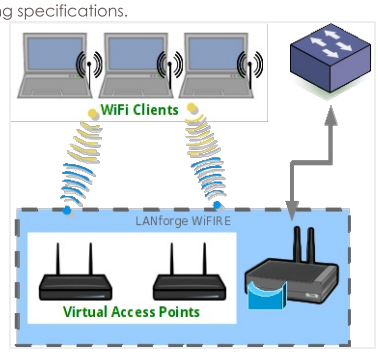


LANforge WiFi Dual Virtual AP Setup with Limited Stations

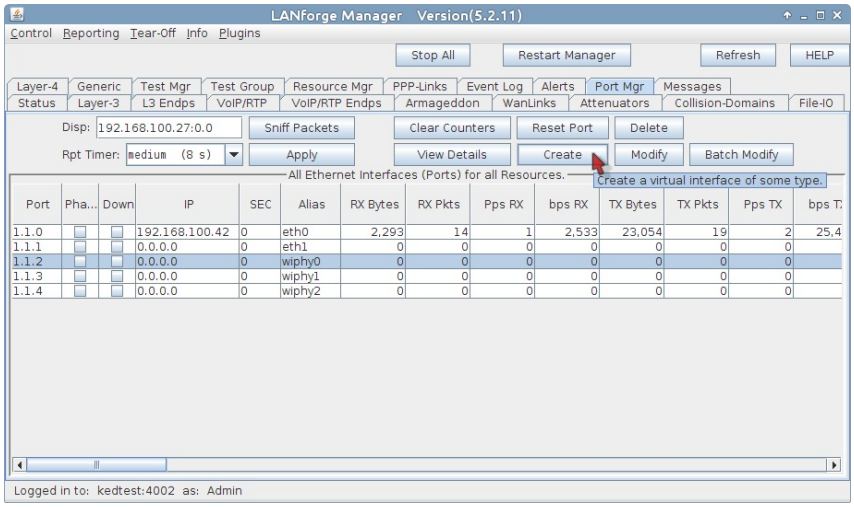
Goal: Configure Two virtual APs with a maximum of ten stations each.

Requires at least a two-radio WiFIRE system such as a CT522, CT523 or CT525. This cookbook provides a basic setup to put a Virtual AP on each of two radios. Both APs are operating on separate channels but share the same SSID. We layout a basic constellation of APs in WPA2 mode that quickly fill up and force a station to search multiple channels for a free AP. This is the condition for issuing a `Code 17` association refusal. This is not a roaming setup (Wireless Roaming/HS2.0 or 802.11r) because it is not configuring enterprise authentication or roaming specifications.

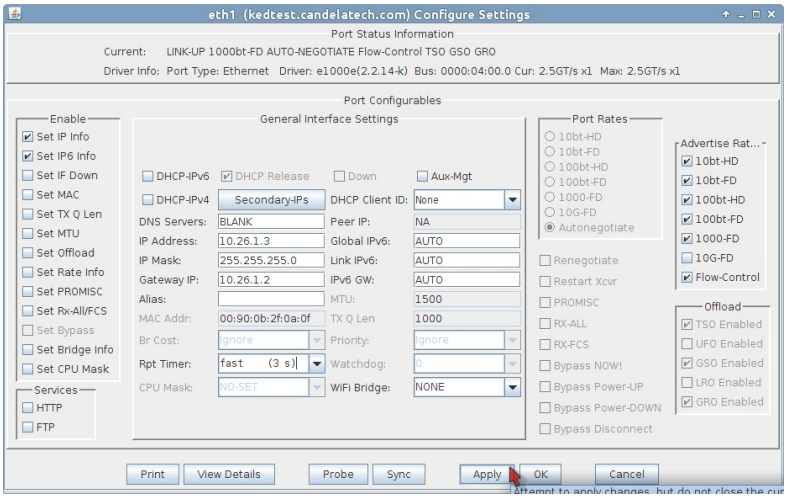




1. Configure two radios to serve the role of Virtual APs.

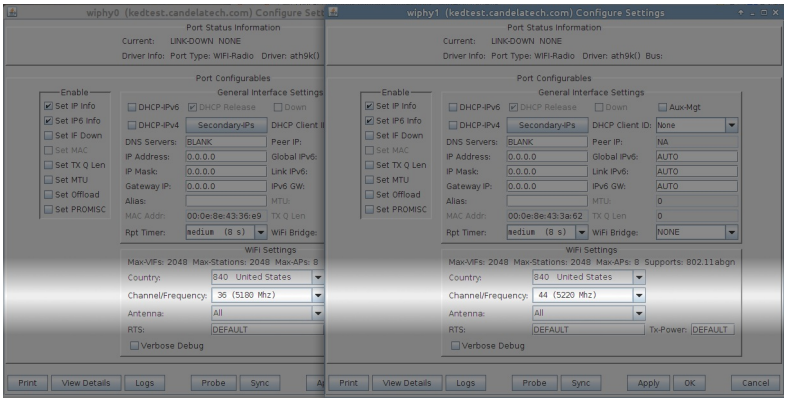


A. Start at the **Ports** tab. We will configure our second ethernet port as the upstream port for the virtual APs.



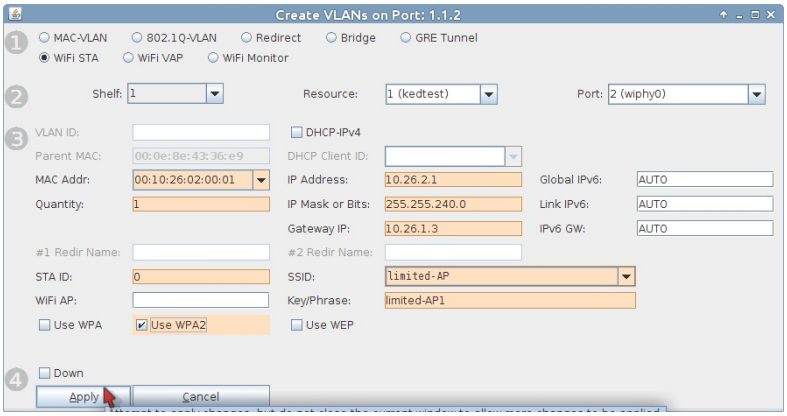
A. Highlight port `eth1` and click **Modify**
B. Set the IP address to `10.26.1.3`
C. and the netmask to `255.255.255.0`
D. with a gateway address of the `10.26.1.2`. (This gateway address matches the eth1 port of a separate system that hosts stations.
E. Click **OK**

B. In the **Ports** tab, select radios `wiphy0, wiphy1` and click **Modify**. You will get two modify windows that you will want to place side by side.



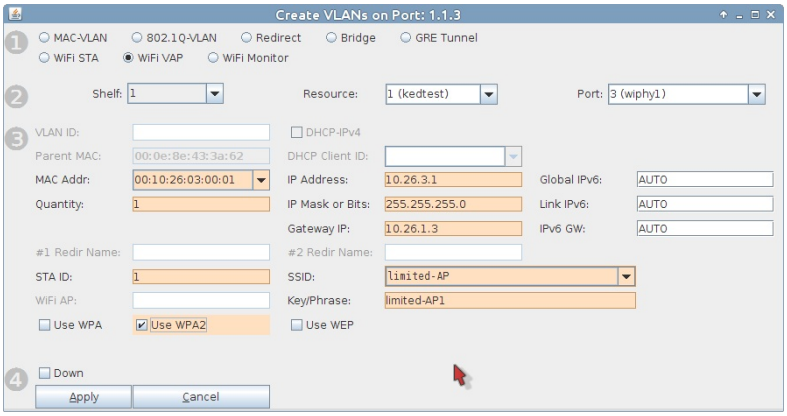
A. Set the channel of wiphy0 to `36`
B. and the channel of wiphy1 to `44`
C. Click **OK** in both windows.

C. Time to create the first virtual AP. In the **Ports** tab, select `wiphy0` and click **Create**.



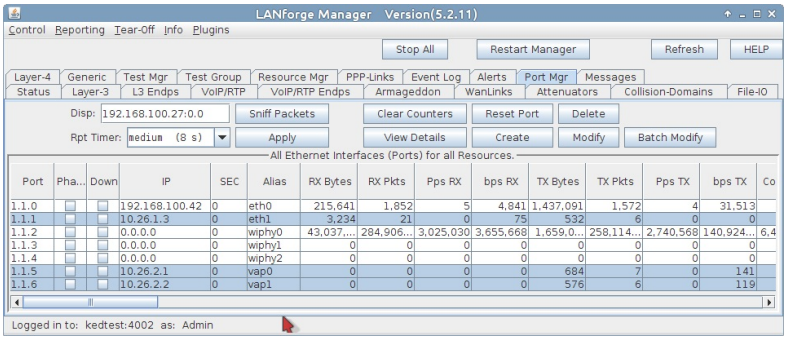
A. Select `WiFi VAP`
B. Enter a **MAC address**: `00:10:26:02:00:01`
C. **Quantity** is `1`
D. **IP** of `10.26.2.1`
E. Let's use a typical **subnet mask**: `255.255.255.0`
F. **Gateway IP** will be from eth1: `10.26.1.3`
G. We will setup **WPA2** with the **SSID** `limited-AP` and **Key/Phrase** `limited-AP1`
H. Click **Apply** to commit.
I. Click **Cancel** to close the window.

D. Now craft a second virtual AP. In the **Ports** tab, select `wiphy1` and click **Create**.

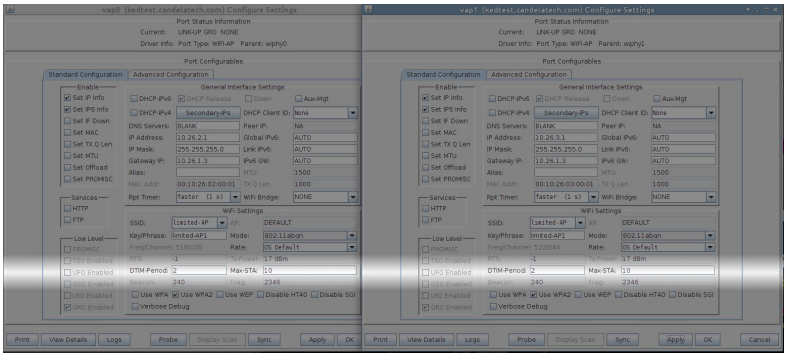


A. Select `WiFi VAP`
B. Enter a **MAC address**: `00:10:26:03:00:01`
C. **Quantity** is `1`
D. **IP** of `10.26.3.1`
E. And a typical **subnet mask** of `255.255.255.0`
F. **Gateway IP** will be from eth1: `10.26.1.3`
G. We will setup **WPA2** with the **SSID** `limited-AP` and **Key/Phrase** `limited-AP1`
H. Click **Apply** to commit.
I. Click **Cancel** to close the window.

E. We now have two virtual access points on two radios on separate channels. Each is on their own subnet. Now we limit the number of stations that can connect to them.



F. In the **Ports** tab, hightlight endpoints `vap0` and `vap1` and click **Modify**

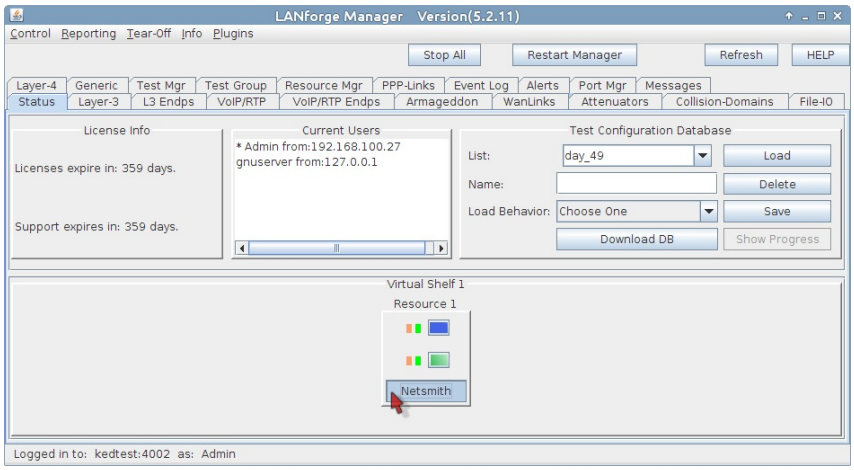


A. At the bottom of the **Port Configure Settings** window for `vap0`, change **Max-STA** to `10`

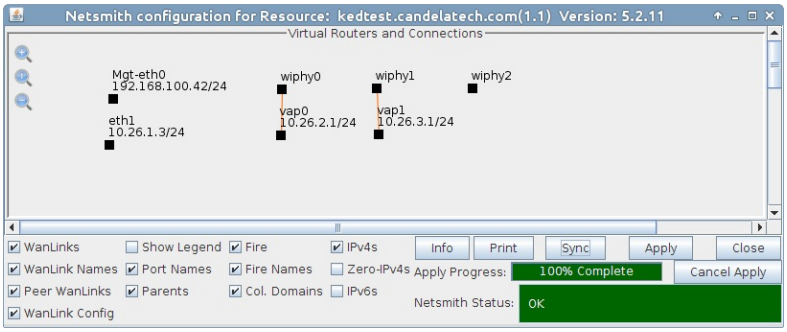
B. Click **OK**

C. Repeat this for `vap1`

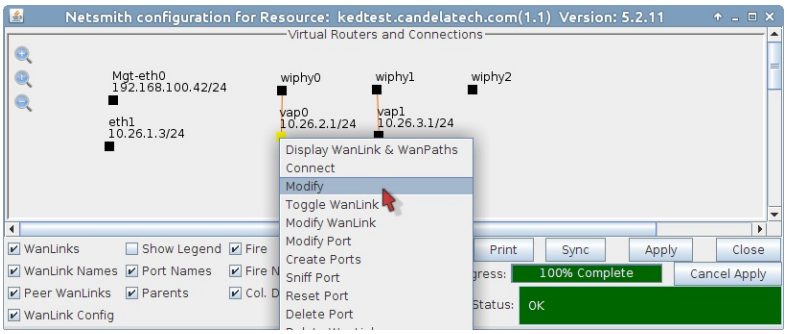
2. We now have two virtual access points that can both accept a small number of stations. We put them on different subnets because configuring DHCP on each VAP is simpler than adding a bridge device..
3. Let's proceed to configuring **DHCP** for the 10.26.2.0/24 and 10.26.3.0/24 networks.
4. In the **Status** tab, click on **Netsmith**



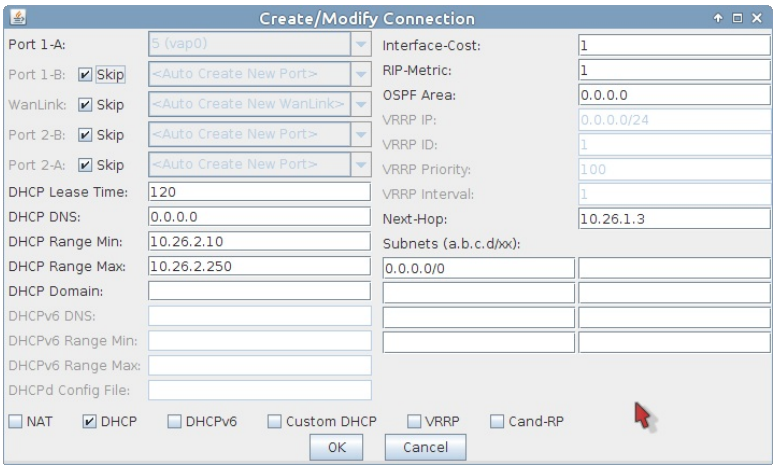
A. Arrange your ports in the **Netsmith** windows and click **Apply**



B. Select **vap0** and right click, choose **Modify**



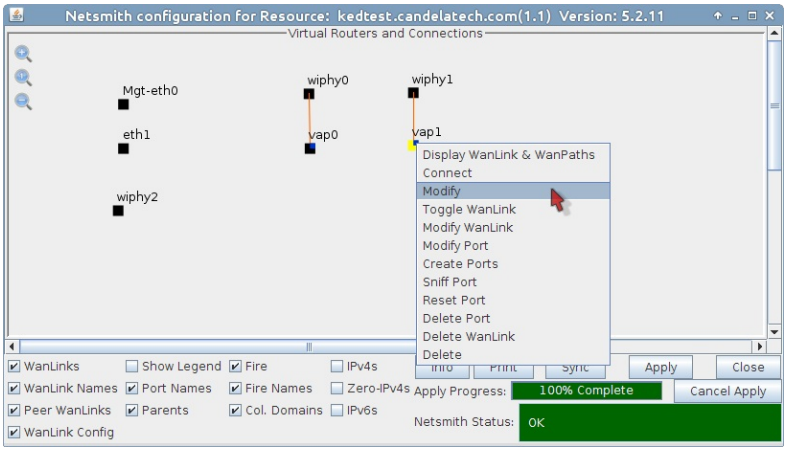
C. Enable the DHCP pool and the Next-hop for the port



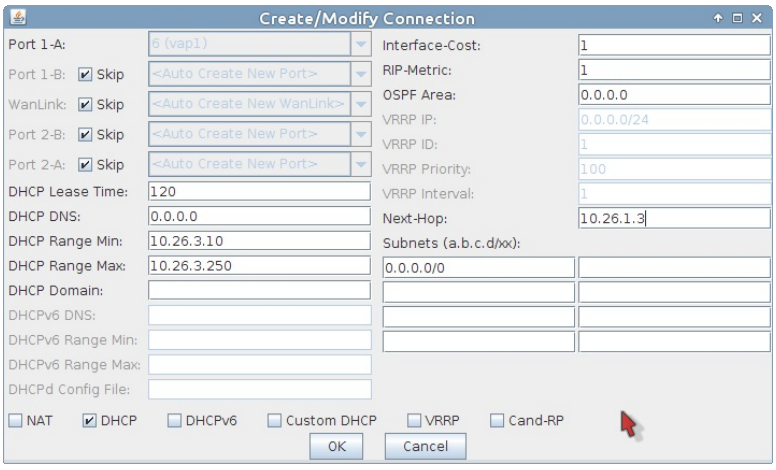
A. Select **DHCP**
B. Set **DHCP Lease Time** to `120`
C. Set **DHCP Range Min** to `10.26.2.10`
D. Set **DHCP Range Max** to `10.26.2.250`
E. Set **Next-Hop** to `10.26.1.3`
F. Add **Subnet** `0.0.0.0/0`
G. Click **OK**

D. Click **Apply** in the Netsmith window to commit the changes to LANforge. If you do not, these changes will disappear.

E. In the **Netsmith** window, select **vap1** and right click, choose **Modify**



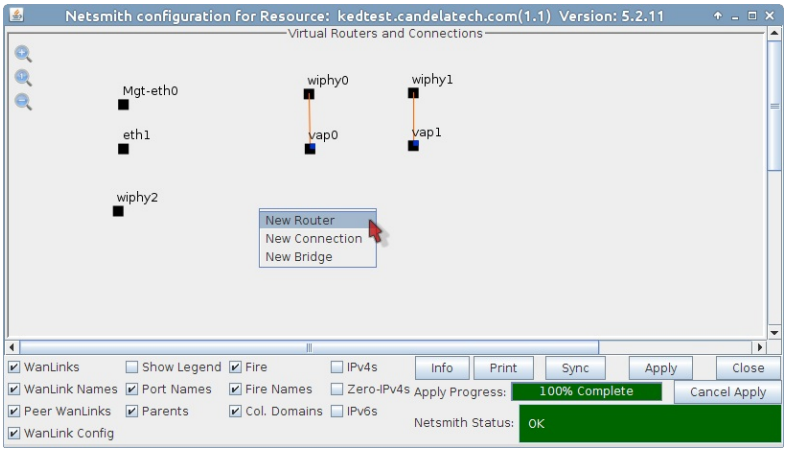
F. Configure the DHCP pool and the Next-hop for **vap1**



A. Select **DHCP**
B. Set **DHCP Lease Time** to `120`
C. Set **DHCP Range Min** to `10.26.3.10`
D. Set **DHCP Range Max** to `10.26.3.250`
E. Set **Next-Hop** to `10.26.1.3`
F. Add **Subnet** `0.0.0.0/0`
G. Click **OK**

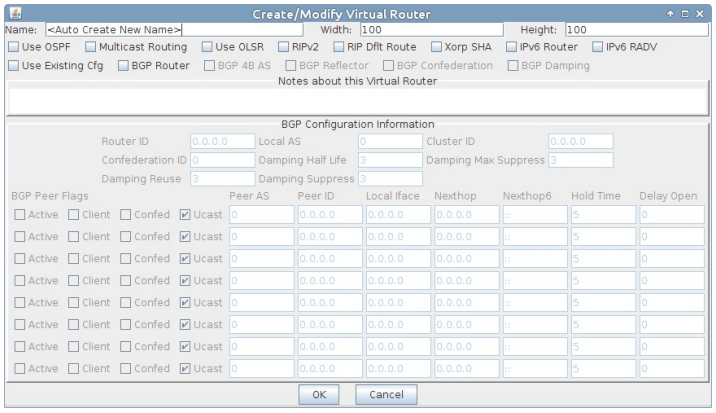
G. Click **Apply** in the Netsmith window to commit the changes to LANforge.

H. We will create a **virtual router** to allow upstream traffic to reach the virtual access points:

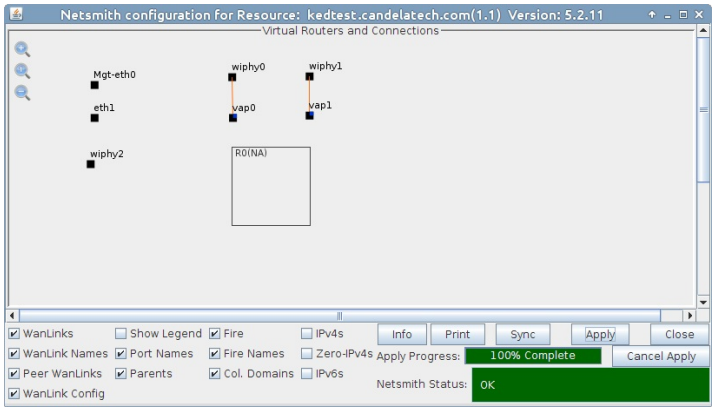


A. Right click in the **Netsmith** window and select **New Router**

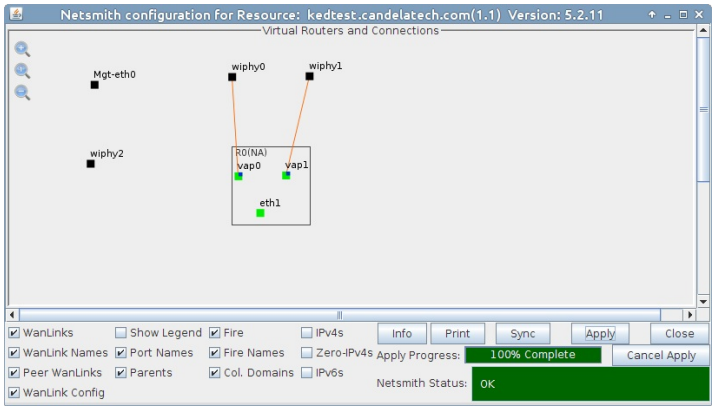
B. You will see the **Create/Modify Virtual Router** window. We can use a default configuration.



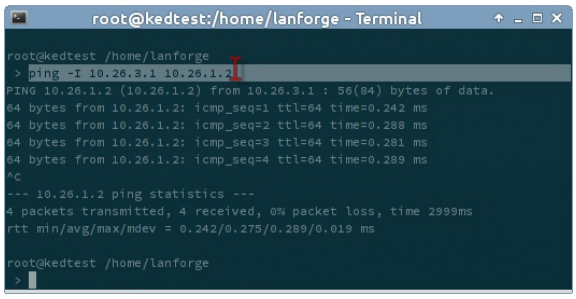
C. Click **OK** to save a default router



D. Drag the endpoints `vap0, vap1` and `eth1` into the virtual router, then click **Apply** in the Netsmith window.



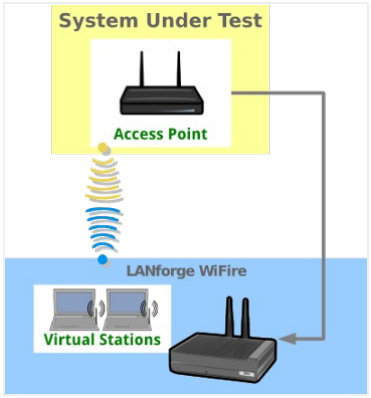
E. We can verify outbound traffic using the following **ping** command. Open a terminal on the LANforge system and run `ping -I 10.26.3.1 10.26.1.2`

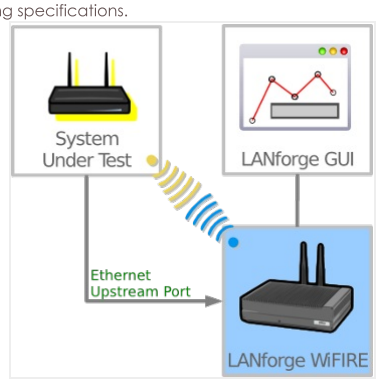


LANforge WiFi Station Association Code 17

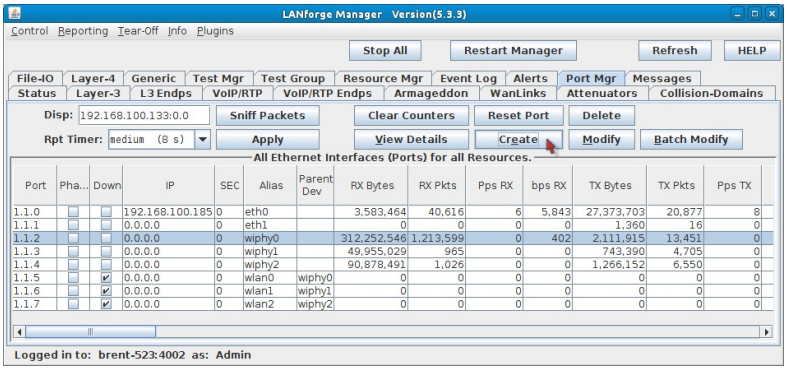
Goal: Configure Two virtual APs with a maximum of ten stations each.

This cookbook provides a basic station setup to emulate over-populating an AP with stations. Requires a WiFIRE system such as a CT520 or better than can provide more stations than your AP(s) under test will accept. This is not a roaming setup (Wireless Roaming/HS2.0 or 802.11r) because it is not exercising enterprise authentication or roaming specifications.

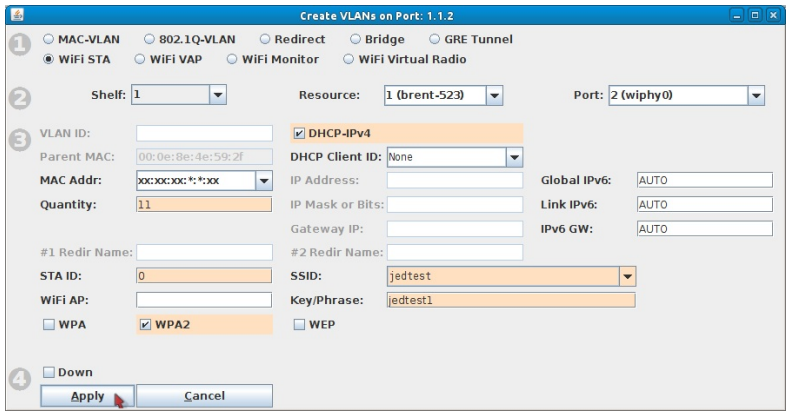




1. Set up one AP with a maximum of 10 stations permitted and an SSID of `jedtest`. (Not shown.)
2. Create 11 stations for SSID `jedtest`
   A. In the Port Mgr tab, select wiphy0 and click the **Create** button.



B. In the Create VLANs window, create 21 stations:



A. Select **WiFi STA**,

B. Select **DHCP-IPv4**

C. and choose pattern `xx:xx:xx:*:*:xx` for MAC Addr.

D. Then enter **11** for Quantity

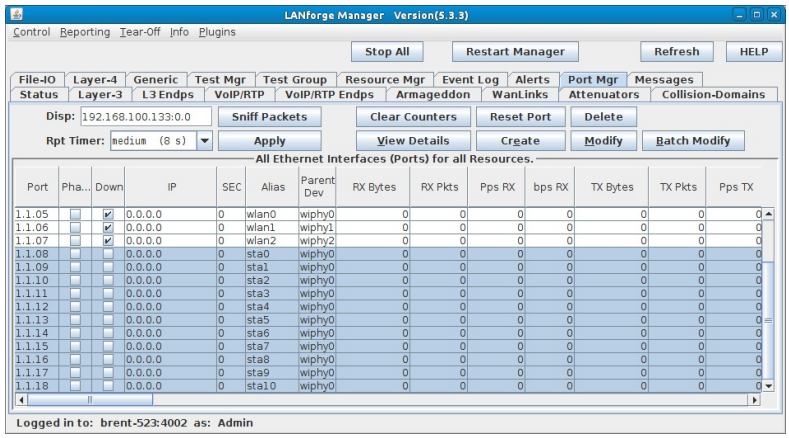
E. STA ID should be **0**

F. and SSID will be `jedtest` for this example.

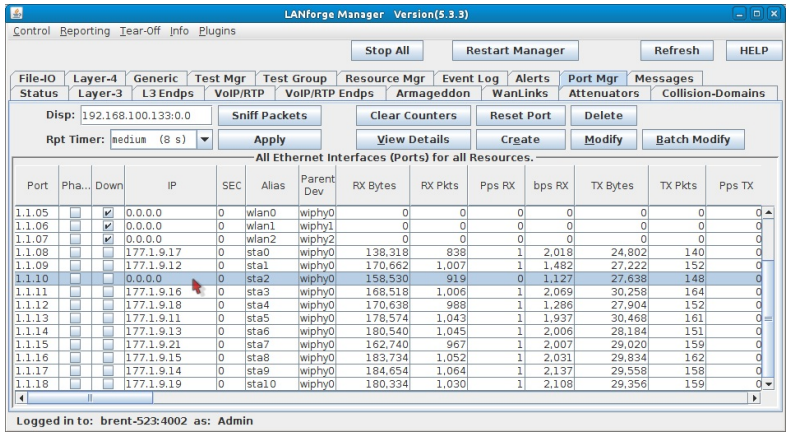
G. We also will select WPA2

H. ...and click the **Apply** button.

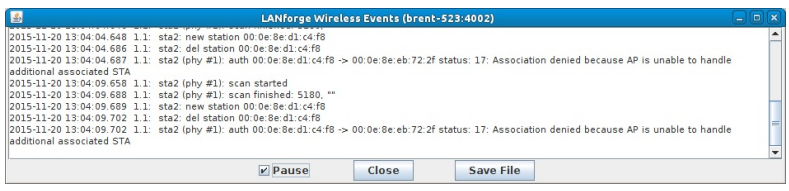
C. We will see 11 stations in the Port Mgr tab:



D. The stations should start and ten will associate.



E. In the LANforge Wireless Events window, you will see a warning for **Association denied**.



LANforge WiFi testing with HotSpot 2.0

Goal: Authenticate using HotSpot 2.0, 802.11u, and 802.1x EAP-TTLS.

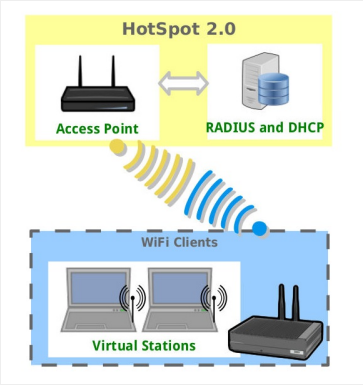
Simplest HotSpot 2.0 example using EAP-TTLS where the client only needs a username and password to authenticate.

There are four main components to setting up this test:

1. One or more Access Points
2. A RADIUS Server
3. A DHCP Server
4. One or more Clients

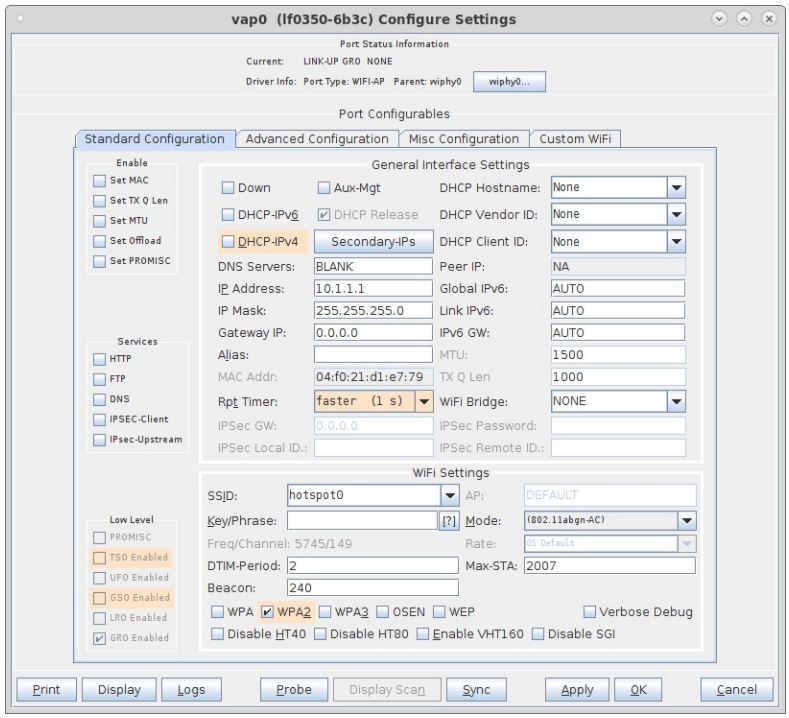
In this example, one LANforge system will fulfill the roles of APs, RADIUS and DHCP servers while another LANforge system will fulfill the role of Client devices.

Any one of the four main components can be replaced by 3rd party devices as required.

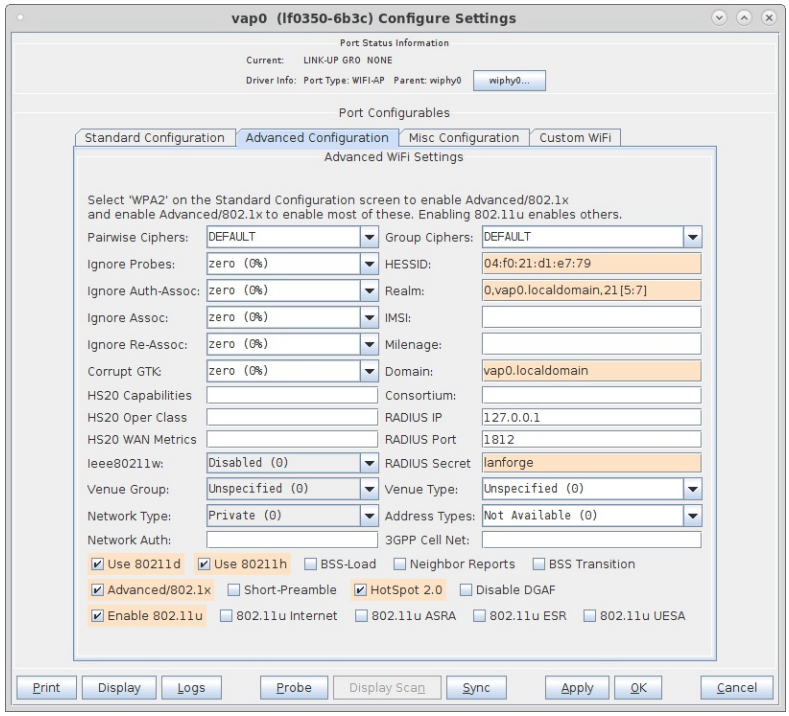


---

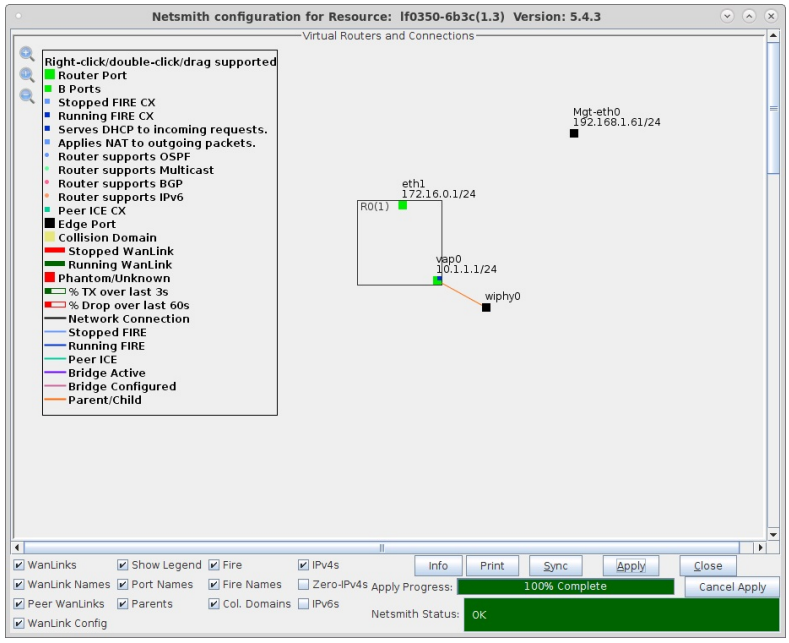
1. Configure Access Point 1
   A. Standard Configuration tab:



B. Advanced Configuration tab:



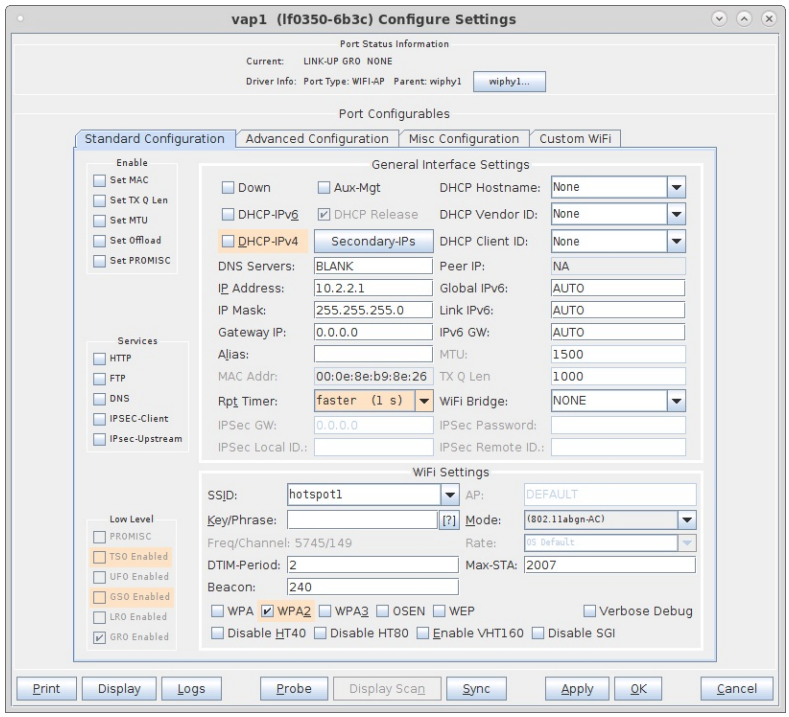
C. Create a Virtual-Router for DHCP Service:



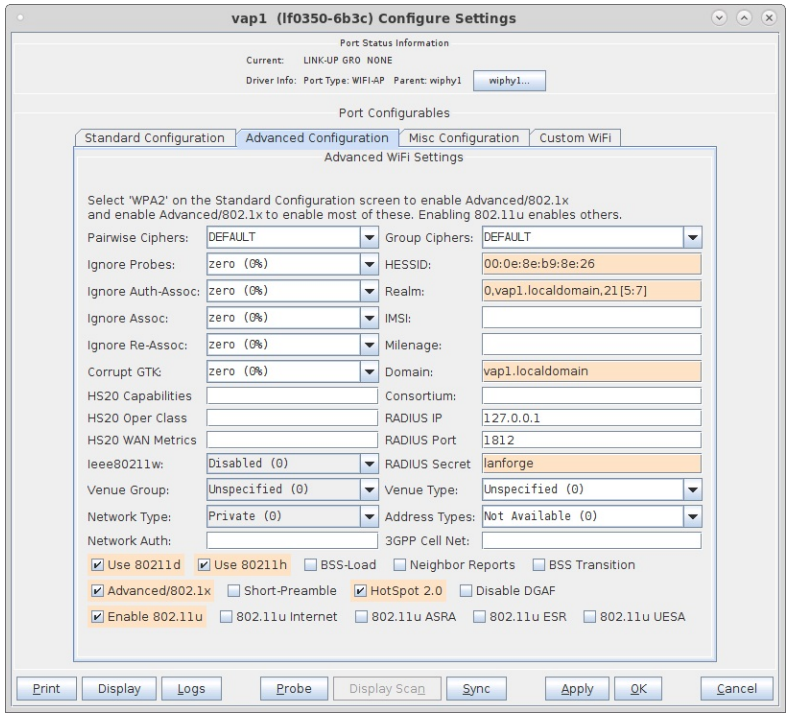
For more information see
LANforge User's Guide: Ports (Interfaces) ,
VAP Bridge Mode Cookbook ,
Virtual Router with DHCP Cookbook

2. Configure Access Point 2

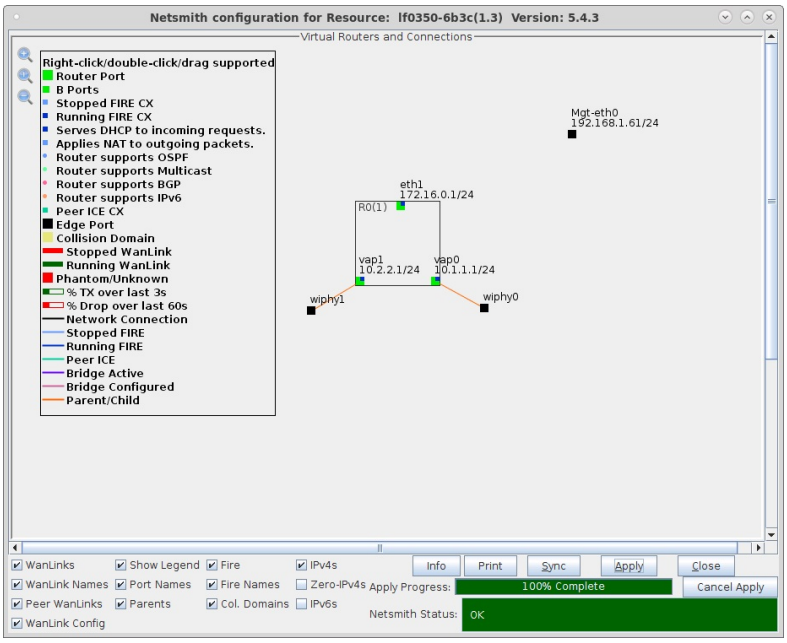
A. Standard Configuration tab:



B. Advanced Configuration tab:

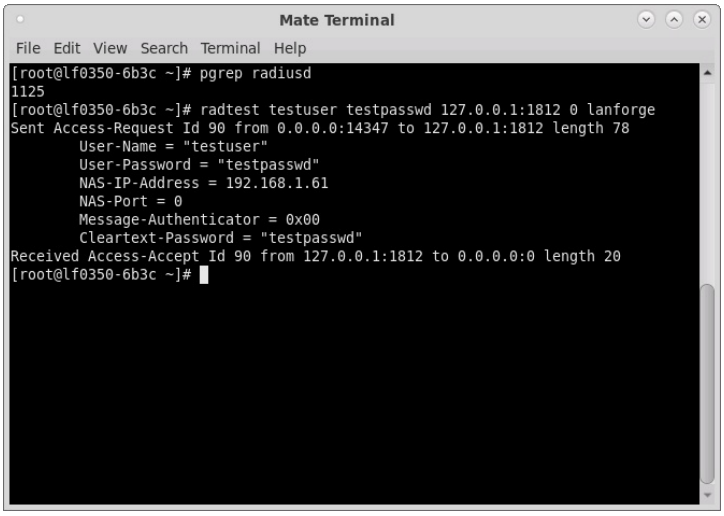


C. Add DHCP Service for the second AP:

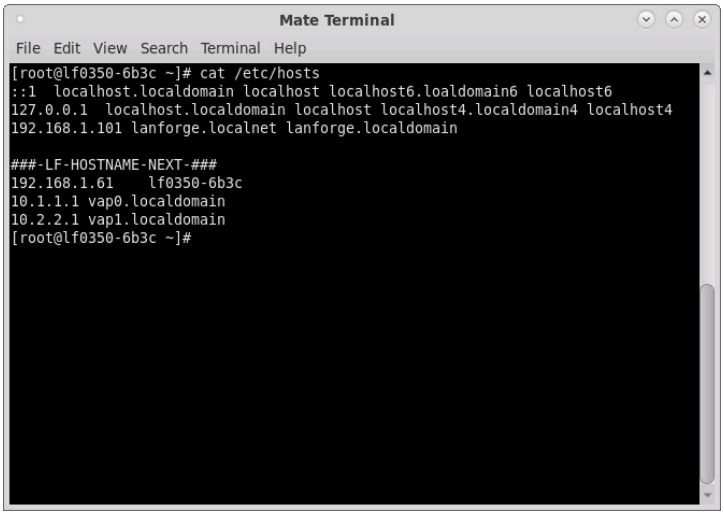


3. Verify Freeradius is running and setup /etc/hosts file.

A. Freeradius is installed on the LANforge system by default with the TTLS user/pass credentials **testuser** / **testpasswd**.

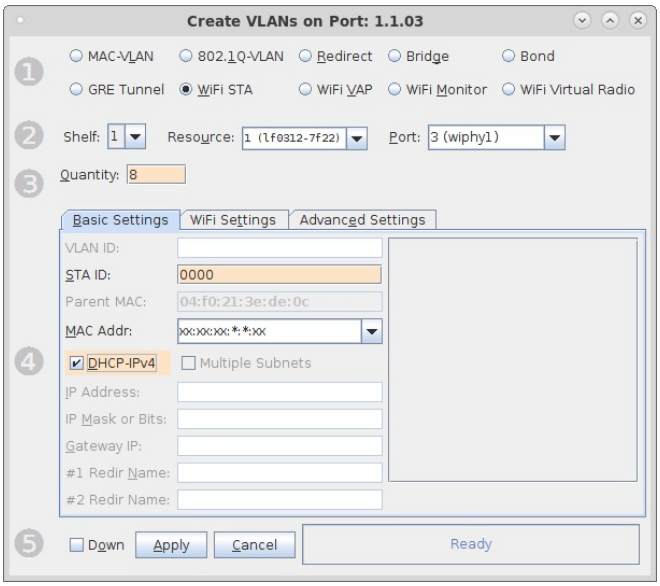


B. Edit /etc/hosts to add local domain names for each AP.

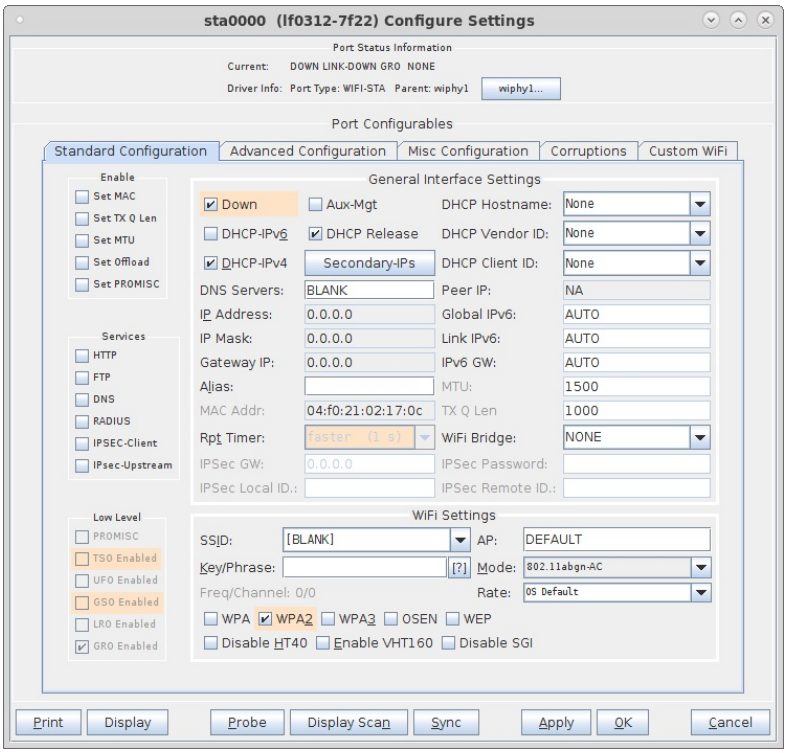


4. Create EAP-TTLS (802.1X username + password authentication) for one or more clients.

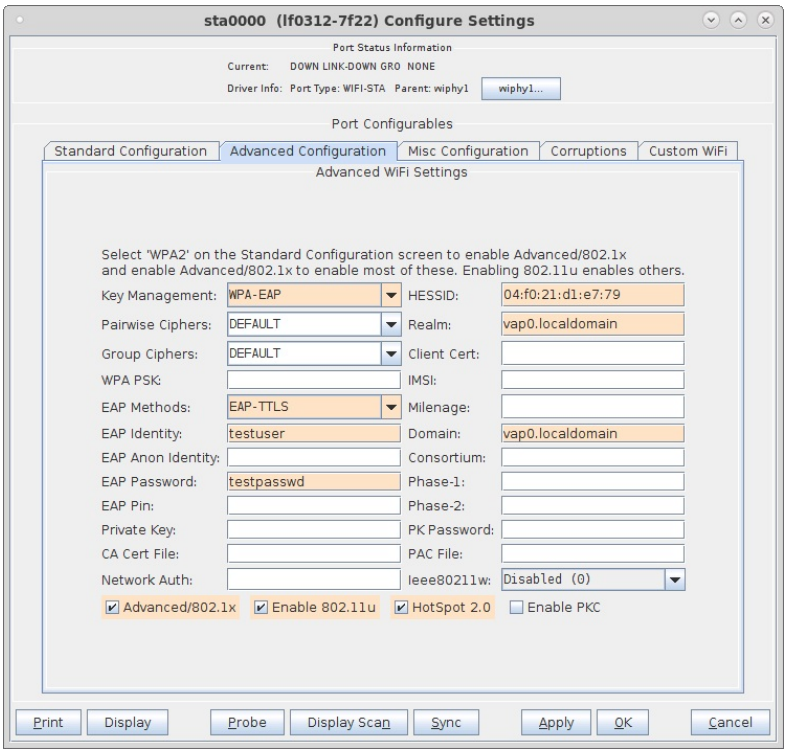
A. Create stations.



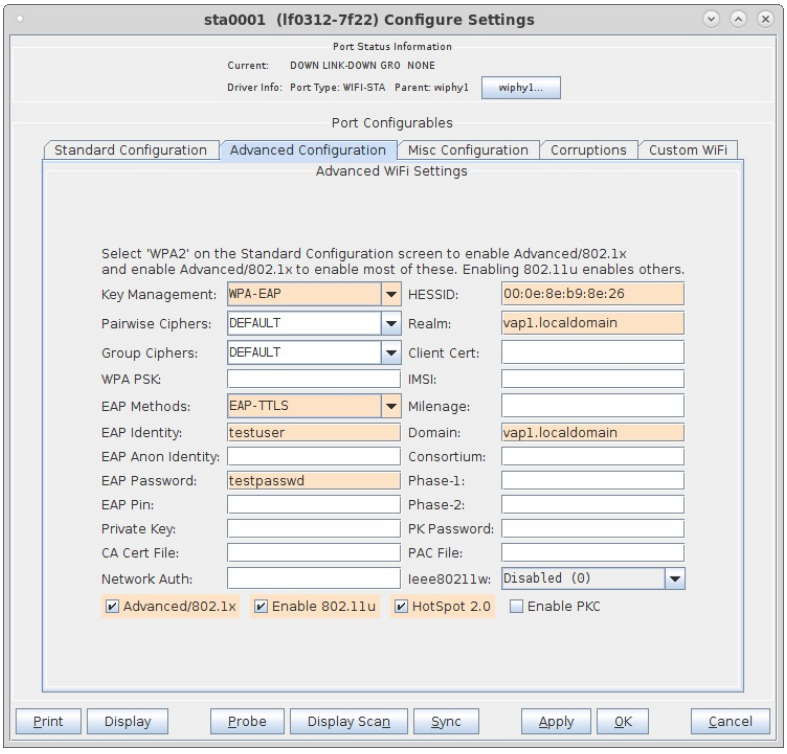
B. The SSID and Key/Password do not need to be configured when using HotSpot 2.0:



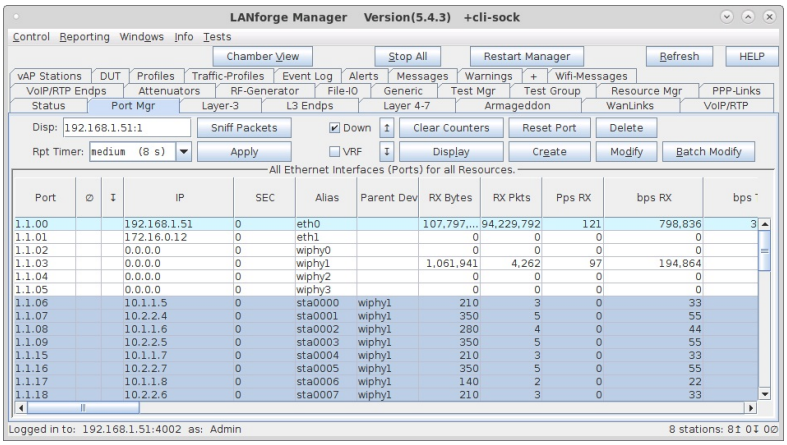
C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the 802.1x, 802.11u, HotSpot 2.0 and other information. The **EAP Identity** and **EAP Password** must match the configuration on your RADIUS server.



D. The next station is setup with the HESSID and Realm/Domain of the second access point.



E. Verify stations connect and obtain DHCP IP address configuration.

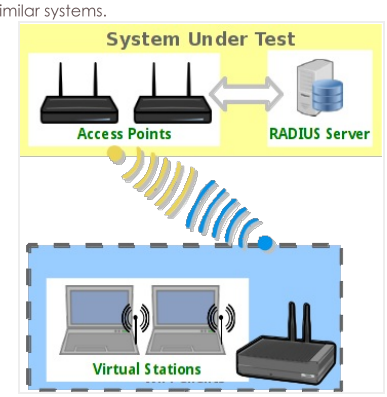


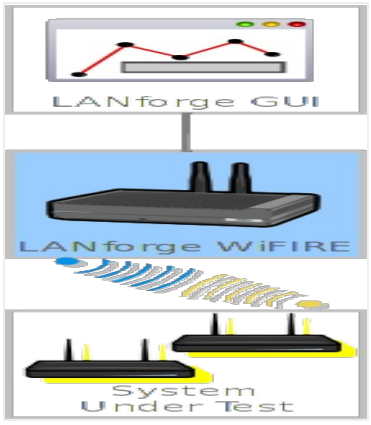
For more information see WiFi Station Cookbook

LANforge WiFi testing with HotSpot 2.0

Goal: Authenticate using HotSpot 2.0, 802.11u, and 802.1x EAP-TTLS and EAP-TLS.

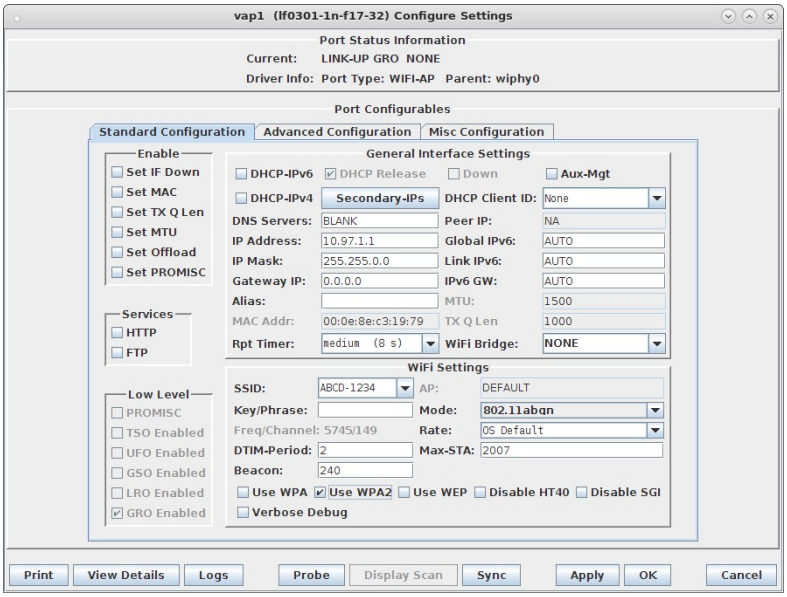
Requires LANforge 5.2.10 or later. Create a Virtual AP configured for HotSpot 2.0 and RADIUS (802.1x) authentication. Create two Station interfaces, one connecting with EAP-TLS and one with EAP-TTLS. This example uses two LANforge CT520 systems but the procedure should work on all CT521, CT522, CT523, CT525 and similar systems.



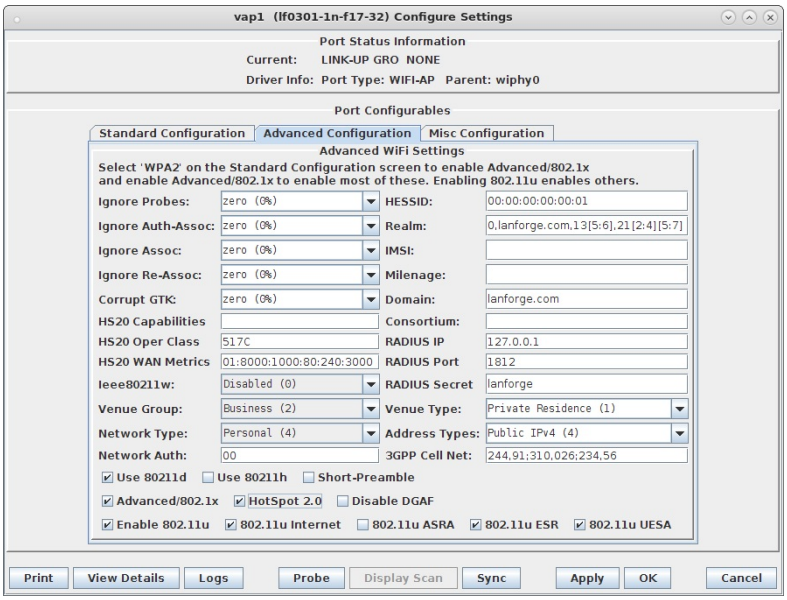


1. Create a virtual AP on wiphy0 of Resource 1. (Skip this if you are using your own AP)
   A. Go to the Port Manager tab, select wiphy0 on proper resource, click Create, fill out appropriate information and create basic Virtual AP interface.

B. The new VAP should appear in the Port-Mgr table. Double-click to modify. Configure IP Address information, SSID and select WPA2:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the 802.1x, 802.11u, HotSpot 2.0, RADIUS and other information. The RADIUS server can be the LANforge machine when freeradius is installed via lf_kinstall.pl --do_radius ... then just copy the client.p12 and ca.pem from the /home/lanforge/ directory on the RADIUS LANforge machine to the Station machine(s) and configure the stations to use those key files. Or, use your own RADIUS server and copy key files as appropriate:

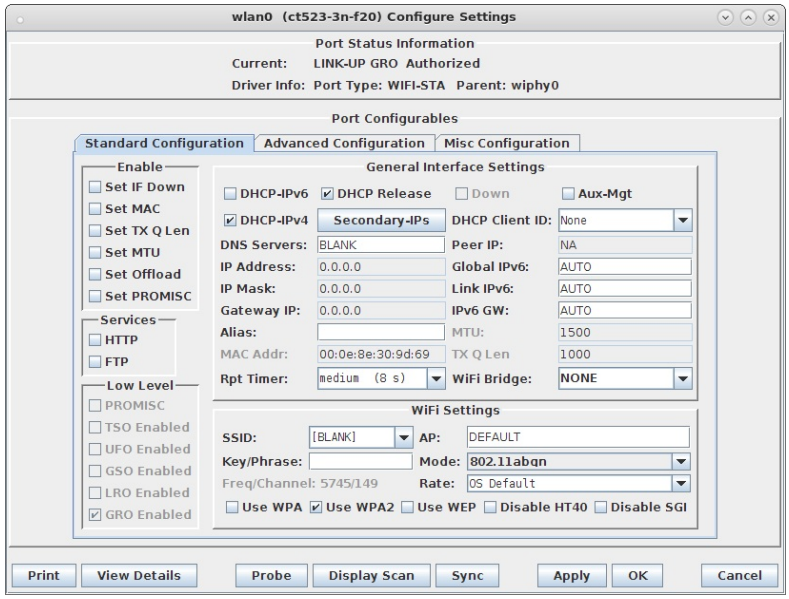


D. Use Netsmith to create Virtual-Router. Add the vapX interface to the Virtual router, configure the Virtual Router port object to serve DHCP. Optionally, add external Ethernet interface to virtual router so that it can route to upstream networks. You could also set up the VAP in bridge mode and use external DHCP server if preferred.

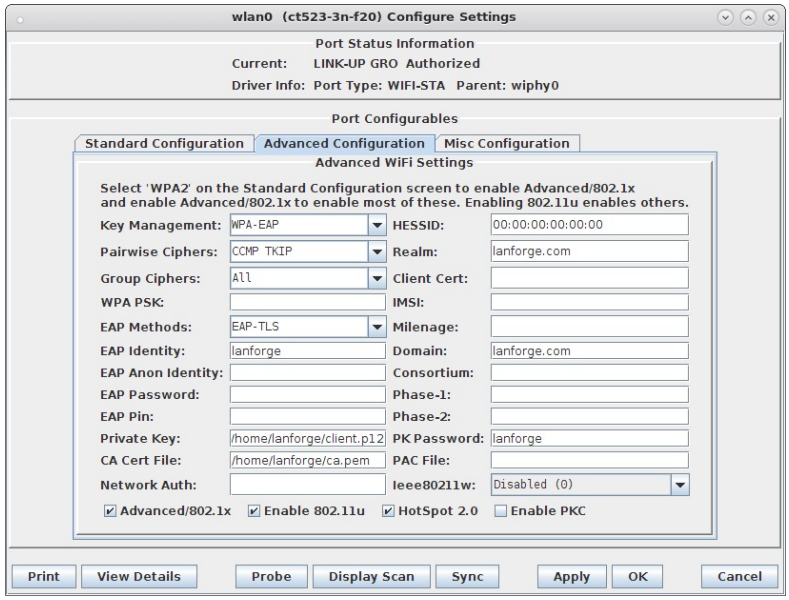
For more information see LANforge User's Guide: Ports (Interfaces) , VAP Bridge Mode Cookbook , Virtual Router with DHCP Cookbook (Skip the WanLink portion)

2. Create EAP-TLS (key certificate authentication) Station on wiphy0 of the second LANforge Resource.
   A. Go to the Port Manager tab, select wiphy0 on proper resource, click Create, fill out appropriate information and create a basic Virtual Station interface.

B. The new Station should appear in the Port-Mgr table. Double-click to modify. Select WPA2. The SSID and Key/Password do not need to be configured when using HotSpot 2.0:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the 802.1x, 802.11u, HotSpot 2.0 and other information. The **EAP Identity** is required, but it may not matter what value you enter unless your RADIUS server has specific requirements. The LANforge RADIUS server does not care. The **Private Key** and **CA Cert File** should come from the /home/lanforge directory on the RADIUS machine if using LANforge for the RADIUS server, or from your own RADIUS server's machine if using your own RADIUS.



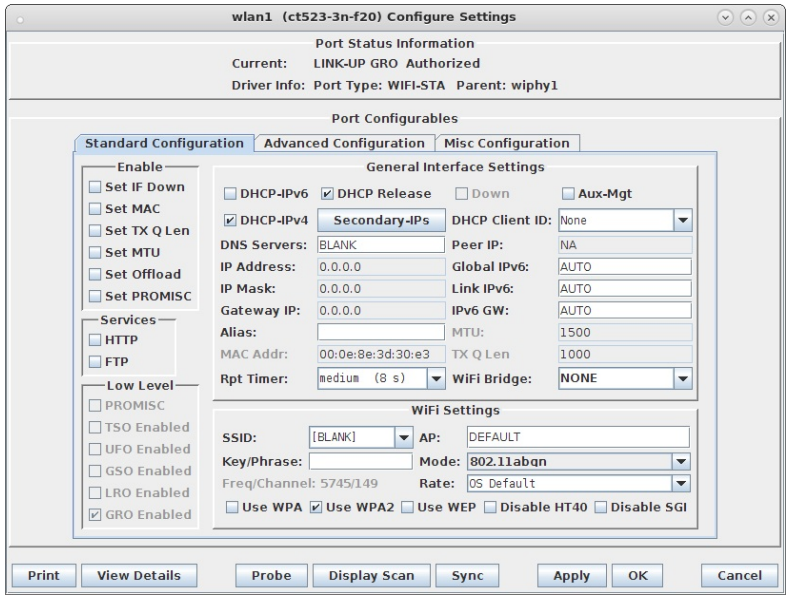
D. Verify Station connects and obtains DHCP IP Address configuration.

For more information see WiFi Station Cookbook

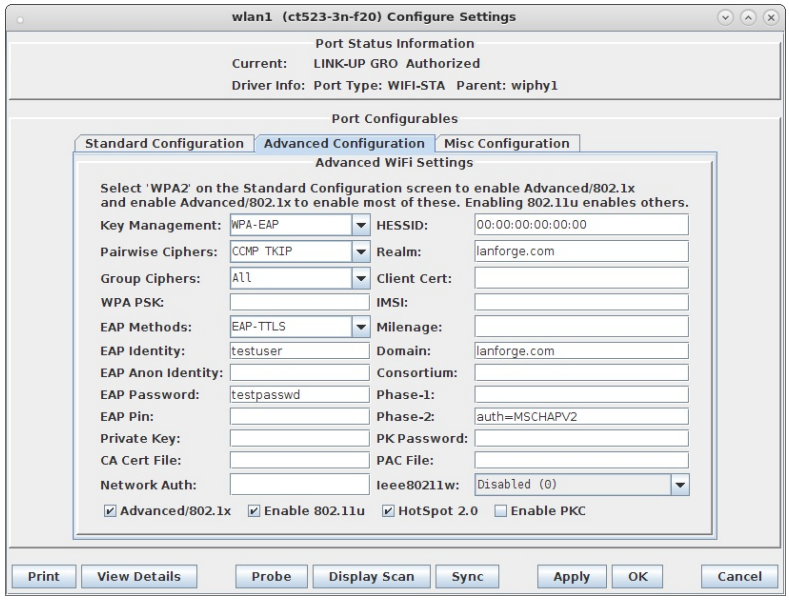
3. Create EAP-TTLS (802.1x username + password authentication) Station on wiphy0 of the second LANforge Resource.

A. Go to the Port Manager tab, select wiphy0 on proper resource, click Create, fill out appropriate information and create another basic Virtual Station interface.

B. The new Station should appear in the Port-Mgr table. Double-click to modify. Select WPA2. The SSID and Key/Password do not need to be configured when using HotSpot 2.0:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the 802.1x, 802.11u, HotSpot 2.0 and other information. The **EAP Identity** and **EAP Password** must match the configuration on your RADIUS server. LANforge RADIUS defaults to: testuser, testpasswd when freeradius is installed via lf_kinstall.pl --do_radius. Phase-2 must be configured as shown for EAP-TTLS with MSCHAPV2.

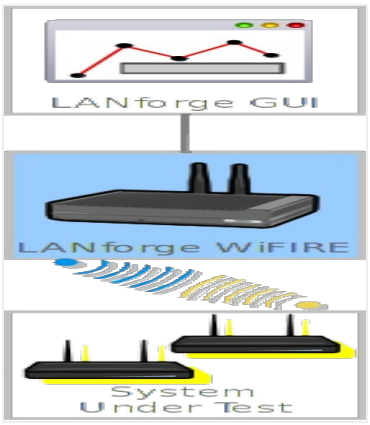


D. Verify Station connects and obtains DHCP IP Address configuration.

For more information see WiFi Station Cookbook

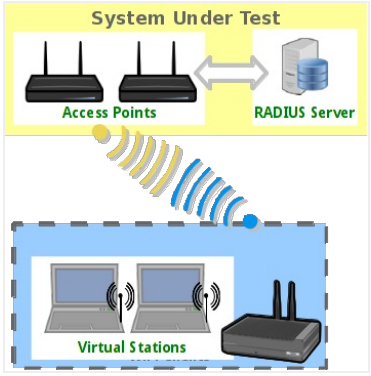
LANforge WiFi AP and Stations with HS20 and EAP-AKA

Goal: Use LANforge to create AP, RADIUS server, and Station that supports HotSpot 2.0 (HS20) and EAP-AKA authentication.

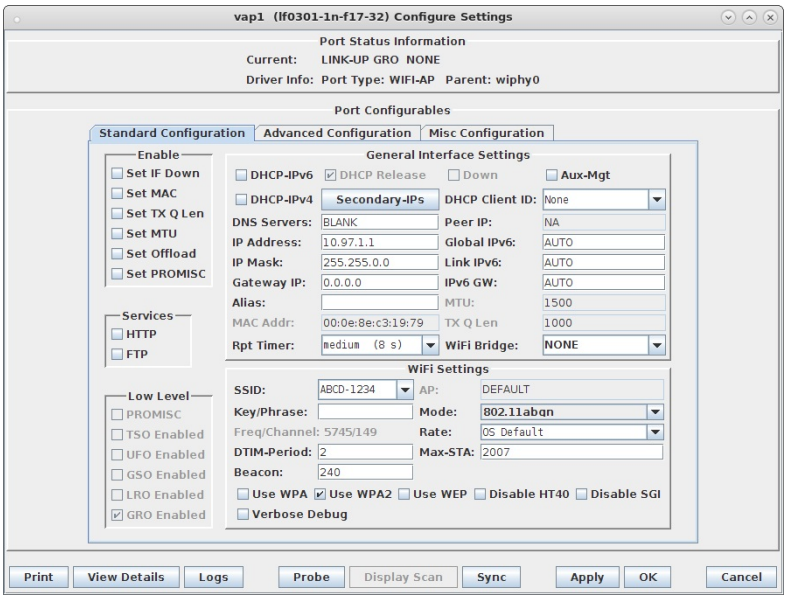


Requires LANforge 5.2.11 or later. Create a Virtual AP configured for HotSpot 2.0 and RADIUS (802.1x) authentication. Create second dummy AP to act as RADIUS server using hostapd. Configure back-end tools authenticate EAP-AKA. Create and configure LANforge WiFi station to test authentication. This example uses two LANforge CT520 systems but the procedure should work on all CT521, CT522, CT523 and CT525 systems. Information here should be useful for non-LANforge users creating their own AP using the hostapd program.

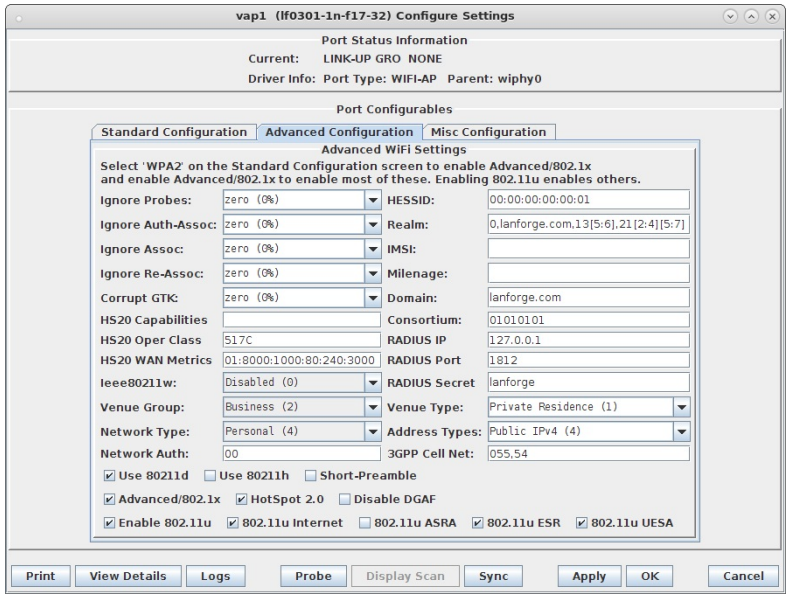
This example uses LANforge for all components, so it is both the test gear and the system under test. This cookbook is primarily intended to record information on how to set up various components of an HS20 EAP-AKA network for demo purposes. Users may choose to implement sub-sections of this cookbook and replace others with third-party APs, RADIUS servers, etc.



1. Create a virtual AP on wiphy0 of Resource 1.
   A. Go to the Port Manager tab, select wiphy0 on proper resource, click Create, fill out appropriate information and create basic Virtual AP interface.
   B. The new VAP should appear in the Port-Mgr table. Double-click to modify. Configure IP Address information, SSID and select WPA2:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the 802.1x, 802.11u, HotSpot 2.0, RADIUS and other information. Note that the 3GPP Cell Net entry must correspond to the IMSI we enter as the station's identity and the IMSI information in the hlr_auc_gw config file:



D. Use Netsmith to create Virtual-Router. Add the vapX interface to the Virtual router, configure the Virtual Router port object to serve DHCP. Optionally, add external Ethernet interface to virtual router so that it can route to upstream networks. You could also set up the VAP in bridge mode and use external DHCP server if preferred.

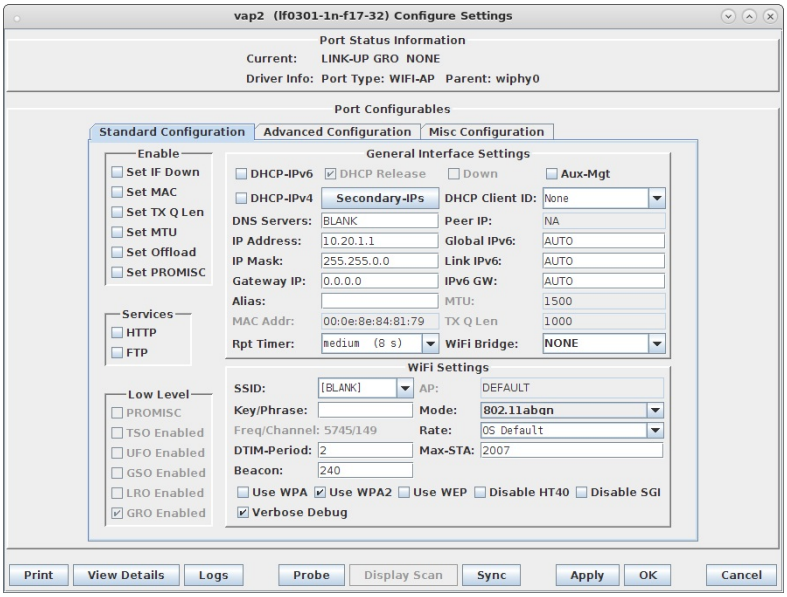
E. For those doing this manually, the hostapd.conf file looks like this:

```
interface=vap0
driver=nl80211
logger_syslog=-1
logger_syslog_level=2
logger_stdout=-1
logger_stdout_level=2
dump_file=/home/lanforge/wifi/hostapd_vap0.dump
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0
ssid=ben-138
bssid=80:01:02:03:04:05
country_code=US
ieee80211d=1
hw_mode=a
ieee80211n=1
beacon_int=240
dtim_period=2
max_num_sta=2007
rts_threshold=2347
fragm_threshold=2346
preamble=0
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
# Enable HT modes if you want 300Mbps+ throughput.
#ht_capab=[HT20][HT40-][HT40+][GF][SHORT-GI-20][SHORT-GI-40]
#      [TX-STBC][RX-STBC123][MAX-AMSDU-7935][DSSS_CCK-40][PSMP][LSIG-TXOP-PROT]
ht_capab=[HT20][HT40+][SHORT-GI-40][SHORT-GI-20]
#vht_capab=[HT20][HT80+][HT80-][SHORT-GI-80]
wmm_enabled=1
wmm_ac_bk_cwmin=4
wmm_ac_bk_cwmax=10
wmm_ac_bk_aifs=7
wmm_ac_bk_txop_limit=0
wmm_ac_bk_acm=0
wmm_ac_be_aifs=3
wmm_ac_be_cwmin=4
wmm_ac_be_cwmax=10
wmm_ac_be_txop_limit=0
wmm_ac_be_acm=0
wmm_ac_vi_aifs=2
wmm_ac_vi_cwmin=3
wmm_ac_vi_cwmax=4
wmm_ac_vi_txop_limit=94
wmm_ac_vi_acm=0
wmm_ac_vo_aifs=2
wmm_ac_vo_cwmin=2
wmm_ac_vo_cwmax=3
wmm_ac_vo_txop_limit=47
wmm_ac_vo_acm=0
channel=36
ieee8021x=1
own_ip_addr=127.0.0.1
auth_server_addr=127.0.0.1
auth_server_port=1812
auth_server_shared_secret=lanforge
wpa=2
wpa_pairwise=CCMP
wpa_key_mgmt=WPA-EAP
# 802.11u configuration
interworking=1
access_network_type=4
internet=1
asra=1
esr=1
uesa=1
venue_group=2
venue_type=1
hessid=00:00:00:00:00:01
roaming_consortium=01010101
venue_name=eng:LANforge Test Venue
network_auth_type=00
ipaddr_type_availability=04
domain_name=lanforge.org
anqp_3gpp_cell_net=055,54
nai_realm=0,lanforge.org,13[5:6],21[2:4][5:7]
# HotSpot 2.0 configuration
hs20=1
hs20_oper_friendly_name=eng:LANforge HotSpot 2.0
hs20_wan_metrics=01:8000:1000:80:240:3000
hs20_operating_class=517C
```

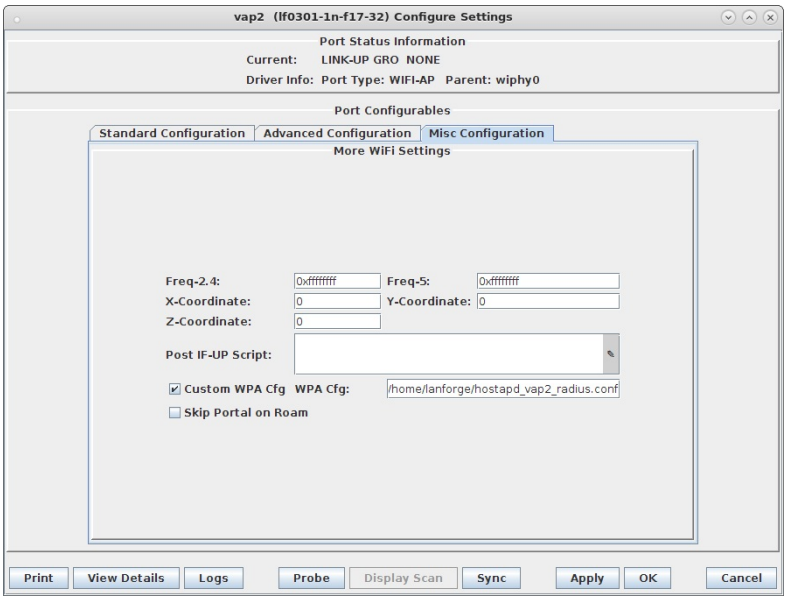
For more information see LANforge User's Guide: Ports (Interfaces) , VAP Bridge Mode Cookbook , Virtual Router with DHCP Cookbook (Skip the WanLink portion)

2. Create a dummy virtual AP on wiphy0 of Resource 1 to act as RADIUS server.
   A. Go to the Port Manager tab, select wiphy0 on proper resource, click Create, fill out appropriate information and create basic Virtual AP interface.

B. The new VAP should appear in the Port-Mgr table. Double-click to modify. Configure IP Address information, SSID and select WPA2:



C. Select the **Misc Configuration** tab in the Port-Modify window and select **Custom WPA Cfg** and specify the **WPA Cfg** file name:



D. We are just using LANforge to start/stop the hostapd process associated with the dummy AP. All interesting configuration is in the custom config file, which should appear similar to this:

```
interface=eth0
driver=wired
logger_syslog=-1
logger_syslog_level=2
logger_stdout=-1
logger_stdout_level=2
dump_file=/home/lanforge/wifi/hostapd_vap2.dump
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0
#ieee8021x=1
eapol_key_index_workaround=0
eap_server=1
eap_user_file=/etc/hostapd.eap_user
server_id=lf138.lanforge.com
eap_sim_db=unix:/tmp/hlr_auc_gw.sock
radius_server_auth_port=1812
radius_server_clients=/etc/hostapd.radius_clients
```

E. Create RADIUS client authentication file on the LANforge machine called **/etc/hostapd.radius_clients** with contents similar to:

```
192.168.100.0/24 lanforge
127.0.0.1/24 lanforge
```

F. Create the /etc/hostap.eap_user file, with contents similar to this:

```
"*@lanforge.com" TLS
"0"* AKA
"2"* AKA
"4"* AKA
```

3. Configure back-end authenticator for EAP-AKA.

A. On the LANforge machine, use your favorite editor to create the file **/etc/hlr_auc_gw.milenage_db** It should have contents similar to:

```
# Parameters for Milenage (Example algorithms for AKA).
# The example Ki, OPc, and AMF values here are from 3GPP TS 35.208 v6.0.0
# 4.3.20 Test Set 20. SQN is the last used SQN value.
# These values can be used for both UMTS (EAP-AKA) and GSM (EAP-SIM)
# authentication. In case of GSM/EAP-SIM, AMF and SQN values are not used, but
# dummy values will need to be included in this file.
# IMSI Ki OPc AMF SQN
232010000000000 90dca4eda45b53cf0f12d7c9c3bc6a89 cb9cccc4b9258e6dca4760379fb82581 61df 000000000000

# These values are from Test Set 19 which has the AMF separation bit set to 1
# and as such, is suitable for EAP-AKA' test.
555444333222111 5122250214c33e723a5dd523fc145fc0 981d464c7c52eb6e5036234984ad0bcf c3ab 16f3b3f70fc1
```

B. As root user, start the hlr_auc_gw tool:

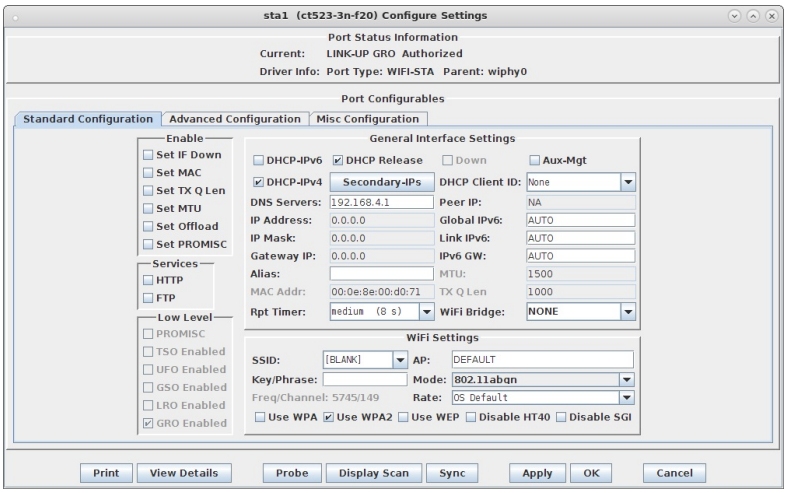
```
cd /home/lanforge
. lanforge.profile
hlr_auc_gw -m /etc/hlr_auc_gw.milenage_db > /tmp/hlr_auc_gw.log&
```

C. In the LANforge-GUI, select the dummy vap (vap2 in our example) and click **Reset** to restart the RADIUS process now that the hlr_auc_gw program is running.

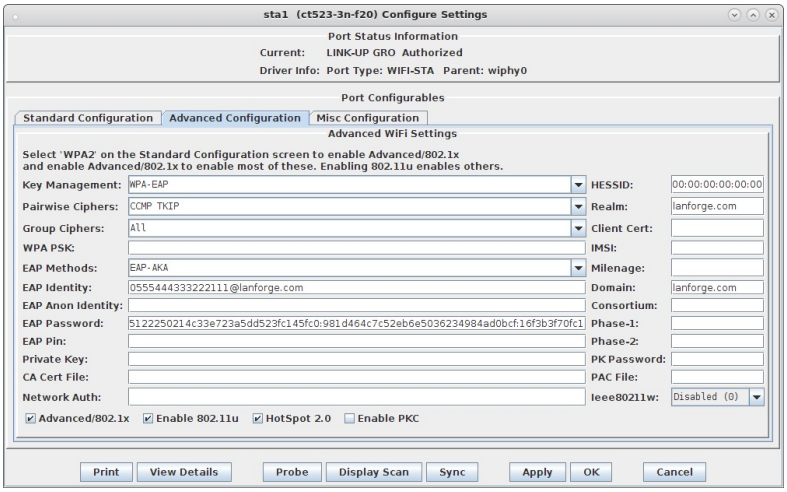
4. Create WiFi Station on second wiphy (and/or second LANforge) to test connectivity

A. Go to the Port Manager tab, select wiphyX on proper resource, click Create, fill out appropriate information and create a basic Virtual Station interface.

B. The new Station should appear in the Port-Mgr table. Double-click to modify. Set the SSID to [BLANK], and Select WPA2. The SSID and Key/Password do not need to be configured when using HotSpot 2.0:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the 802.1x, 802.11u, HotSpot 2.0 and other information. The **EAP Identity** and **EAP Password** must match the configuration on your RADIUS server, and in this case, that means it must match the hlr_auc_gw configuration we entered earlier. The HS20 Realm and Domain should be configured to match the HS20 AP.



D. Verify Station connects to the AP and obtains DHCP IP Address configuration. If it does not work, look at the Station's supplicant logs, the AP logs, the RADIUS server logs, and the hlr_auc_gw logs.

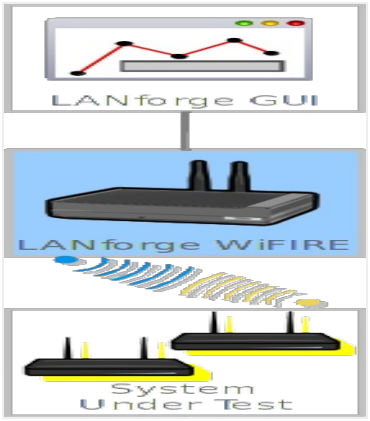
E. For those doing this manually, the wpa_supplicant.conf file looks like this:

```
ctrl_interface=/var/run/wpa_supplicant
fast_reauth=1
can_scan_one=1
# 802.11u / Interworking configuration.
interworking=1
auto_interworking=1
access_network_type=0
# HotSpot 2.0 configuration
hs20=1
bss_max_count=2000
network={
    interworking_defaults=1
    disable_ht=0
    disable_vht=1
    disable_ht40=0
    disable_sgi=0
    ht_mcs=""
    disable_max_amsdu=-1
    ampdu_factor=-1
    ampdu_density=-1
}
cred={
    username="0555444333222111@wlan.org"
    password="5122250214c33e723a5dd523fc145fc0:981d464c7c52eb6e5036234984ad0bcf:16f3b3f70fc1"
    realm="lanforge.org"
    domain="lanforge.org"
    eap=AKA
}
```

For more information see WiFi Station Cookbook

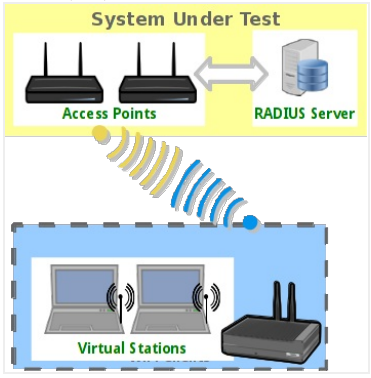
LANforge WiFi AP and Stations with HS20 and EAP-SIM

Goal: Use LANforge to create AP, RADIUS server, and Station that supports HotSpot 2.0 (HS20) and EAP-SIM authentication.



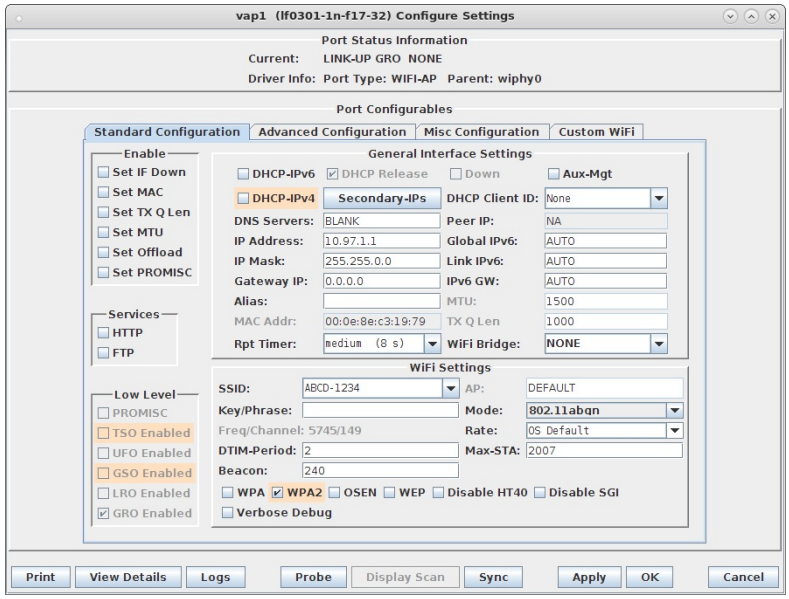
Requires LANforge 5.2.11 or later. Create a Virtual AP configured for HotSpot 2.0 and RADIUS (802.1x) authentication. Create a MAC-VLAN interface to act as RADIUS server using hostapd. Configure back-end tools authenticate EAP-SIM. Create and configure LANforge WiFi station to test authentication. This example uses two LANforge CT520 systems but the procedure should work on all CT521, CT522, CT523 and CT525 systems. Information here should be useful for non-LANforge users creating their own AP using the hostapd program.

This example uses LANforge for all components, so it is both the test gear and the system under test. This cookbook is primarily intended to record information on how to set up various components of an HS20 EAP-SIM network for demo purposes. Users may choose to implement sub-sections of this cookbook and replace others with third-party APs, RADIUS servers, etc.

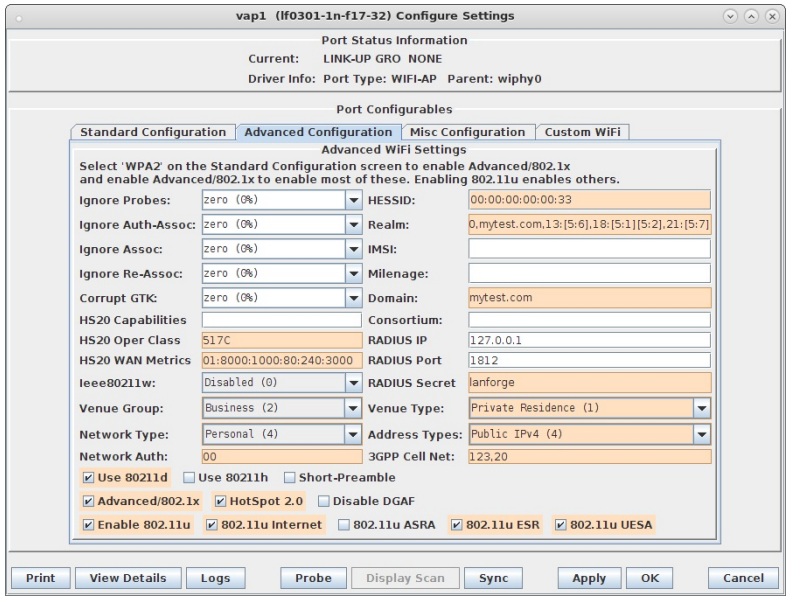


1. Create a virtual AP on wiphy0 of Resource 1.
    A. Go to the Port Manager tab, select wiphy0 on proper resource, click Create, fill out appropriate information and create basic Virtual AP interface.

B. The new VAP should appear in the Port-Mgr table. Double-click to modify. Configure IP Address information, SSID and select WPA2:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the 802.1x, 802.11u, HotSpot 2.0, RADIUS and other information. Note that the 3GPP Cell Net entry must correspond to the IMSI we enter as the station's identity and the IMSI information in the hlr_auc_gw config file. Also, note that the Realm must contain the EAP Method Type 18 (EAP-SIM) as described in http://www.iana.org/assignments/eap-numbers/eap-numbers.xhtml#eap-numbers-4:



D. Use Netsmith to create Virtual-Router. Add the vapX interface to the Virtual router, configure the Virtual Router port object to serve DHCP. Optionally, add external Ethernet interface to virtual router so that it can route to upstream networks. You could also set up the VAP in bridge mode and use external DHCP server if preferred.

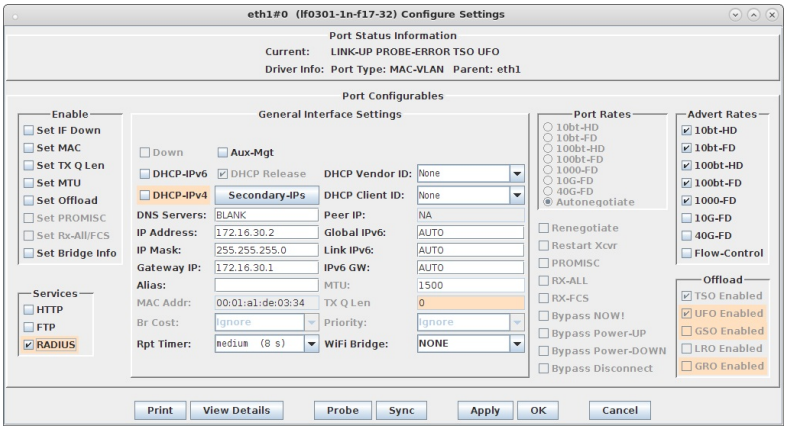
E. For those doing this manually, the hostapd.conf file looks like this:

```
interface=vap1
driver=nl80211
logger_syslog=-1
logger_syslog_level=2
logger_stdout=-1
logger_stdout_level=2
dump_file=/home/lanforge/wifi/hostapd_vap0.dump
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0
ssid=ABCD-1234
bssid=00:0e:8e:c3:19:79
country_code=US
ieee80211d=1
ieee80211h=0
ieee80211w=0
hw_mode=a
ieee80211n=1
beacon_int=240
dtim_period=2
max_num_sta=2007
rts_threshold=2347
fragm_threshold=2346
preamble=0
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
# Enable HT modes if you want 300Mbps+ throughput.
#ht_capab=[HT20][HT40-][HT40+][GF][SHORT-GI-20][SHORT-GI-40]
#      [TX-STBC][RX-STBC123][MAX-AMSDU-7935][DSSS_CCK-40][PSMP][LSIG-TXOP-PROT]
ht_capab=[HT20][HT40+][SHORT-GI-40][SHORT-GI-20]
#vht_capab=[HT20][HT80+][HT80-][SHORT-GI-80]
wmm_enabled=1
wmm_ac_bk_cwmin=4
wmm_ac_bk_cwmax=10
wmm_ac_bk_aifs=7
wmm_ac_bk_txop_limit=0
wmm_ac_bk_acm=0
wmm_ac_be_aifs=3
wmm_ac_be_cwmin=4
wmm_ac_be_cwmax=10
wmm_ac_be_txop_limit=0
wmm_ac_be_acm=0
wmm_ac_vi_aifs=2
wmm_ac_vi_cwmin=3
wmm_ac_vi_cwmax=4
wmm_ac_vi_txop_limit=94
wmm_ac_vi_acm=0
wmm_ac_vo_aifs=2
wmm_ac_vo_cwmin=2
wmm_ac_vo_cwmax=3
wmm_ac_vo_txop_limit=47
wmm_ac_vo_acm=0
channel=149
ieee8021x=1
own_ip_addr=127.0.0.1
auth_server_addr=127.0.0.1
auth_server_port=1812
auth_server_shared_secret=lanforge
wpa=2
wpa_pairwise=CCMP
wpa_key_mgmt=WPA-EAP WPA-EAP-SHA256
# 802.11u configuration
interworking=1
access_network_type=4
internet=1
asra=1
esr=1
uesa=1
venue_group=2
venue_type=1
hessid=00:00:00:00:00:33
venue_name=eng:LANforge Test Venue
network_auth_type=00
ipaddr_type_availability=04
domain_name=mytest.com
anqp_3gpp_cell_net=123,20
nai_realm=0,mytest.com,13:[5:6],18:[5:1][5:2],21:[5:7]
# HotSpot 2.0 configuration
hs20=1
hs20_oper_friendly_name=eng:LANforge HotSpot 2.0
hs20_wan_metrics=01:8000:1000:80:240:3000
hs20_operating_class=517C
```

For more information see LANforge User's Guide: Ports (Interfaces) , VAP Bridge Mode Cookbook , Virtual Router with DHCP Cookbook (Skip the WanLink portion)

2. Create a MAC-VLAN interface on eth1 of Resource 1 to act as RADIUS server.
   A. Go to the Port Manager tab, select eth1 on the proper resource, click Create, fill out appropriate information and create a basic MAC-VLAN interface.

B. The new interface should appear in the Port-Mgr table. Double-click to modify. Configure IP Address information and select the RADIUS checkbox which will allow a hostapd based RADIUS server on the interface using the config file /home/lanforge/wifi/hostapd_eth1#0.conf :



C. We are just using LANforge to start/stop the hostapd process associated with the MAC-VLAN interface. All interesting configuration is in the custom config file, which should appear similar to this:

```
interface=eth1#0
driver=wired
logger_syslog=-1
logger_syslog_level=2
logger_stdout=-1
logger_stdout_level=2
#dump_file=/home/lanforge/wifi/hostapd_eth1#0.dump
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0
ieee8021x=1
eapol_key_index_workaround=0
eap_server=1
eap_user_file=/etc/hostapd.eap_user
server_id=lf0301.mytest.com
eap_sim_db=unix:/tmp/hlr_auc_gw.sock
radius_server_auth_port=1812
radius_server_clients=/etc/hostapd.radius_clients

ca_cert=/etc/raddb/certs/ca.pem
server_cert=/etc/raddb/certs/server.pem
private_key=/etc/raddb/certs/server.key
private_key_passwd=lanforge
```

D. Create RADIUS client authentication file on the LANforge machine called **/etc/hostapd.radius_clients** with contents similar to:

```
192.168.100.0/24 lanforge
127.0.0.1/24 lanforge
```

E. Create the /etc/hostap.eap_user file, with contents similar to this:

```
"*@mytest.com" TLS
"0"* SIM,TTLS,TLS,PEAP,AKA
"1"* SIM,TTLS,TLS,PEAP,AKA
```

3. Configure back-end authenticator for EAP-SIM.

A. On the LANforge machine, use your favorite editor to create the file **/etc/hlr_auc_gw.milenage_db** It should have contents similar to:

```
# Parameters for Milenage (Example algorithms for AKA).
# The example Ki, OPc, and AMF values here are from 3GPP TS 35.208 v6.0.0
# 4.3.20 Test Set 20. SQN is the last used SQN value.
# These values can be used for both UMTS (EAP-AKA) and GSM (EAP-SIM)
# authentication. In case of GSM/EAP-SIM, AMF and SQN values are not used, but
# dummy values will need to be included in this file.
# IMSI Ki OPc AMF SQN
232010000000000 90dca4eda45b53cf0f12d7c9c3bc6a89 cb9cccc4b9258e6dca4760379fb82581 61df 000000000000

# These values are from Test Set 19 which has the AMF separation bit set to 1
# and as such, is suitable for EAP-AKA' test.
555444333222111 5122250214c33e723a5dd523fc145fc0 981d464c7c52eb6e5036234984ad0bcf c3ab 16f3b3f70fc1
```

B. As root user, start the hlr_auc_gw tool:

```
cd /home/lanforge
. lanforge.profile
hlr_auc_gw -m /etc/hlr_auc_gw.milenage_db > /tmp/hlr_auc_gw.log &
```

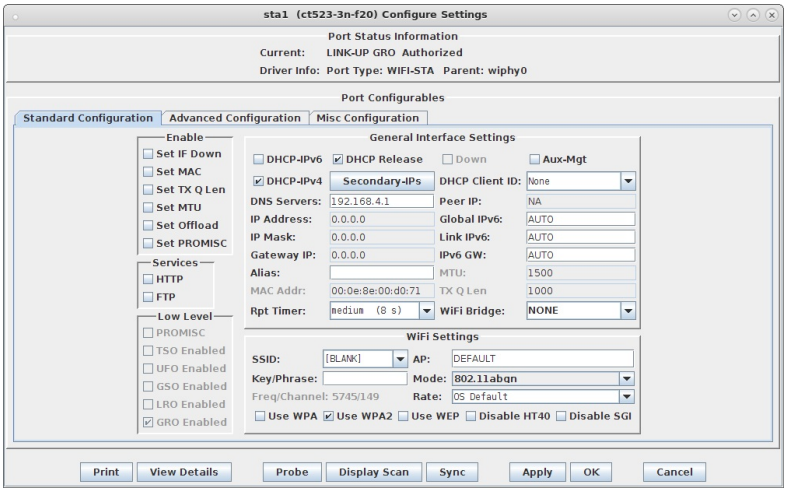
NOTE: If the hlr_auc_gw does not start, you may have to remove the file /tmp/hlr_auc_gw.sock first.

C. In the LANforge-GUI, select the MAC-VLAN interface (eth1#0 in our example) and click **Reset** to restart the hostapd RADIUS process now that the hlr_auc_gw program is running.

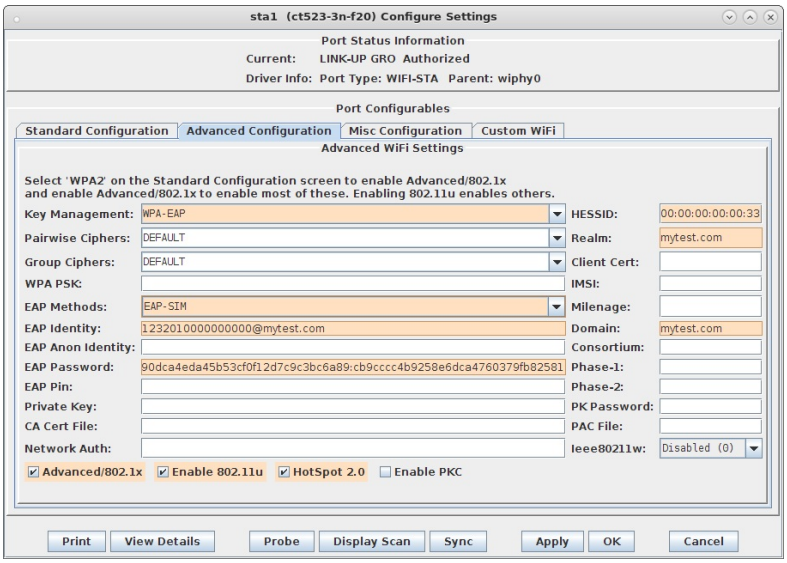
4. Create WiFi Station on second wiphy (and/or second LANforge) to test connectivity

A. Go to the Port Manager tab, select wiphyX on proper resource, click Create, fill out appropriate information and create a basic Virtual Station interface.

B. The new Station should appear in the Port-Mgr table. Double-click to modify. Set the SSID to [BLANK], and Select WPA2. The SSID and Key/Password do not need to be configured when using HotSpot 2.0:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the 802.1x, 802.11u, HotSpot 2.0 and other information. The **EAP Identity** and **EAP Password** must match the configuration on your RADIUS server, and in this case, that means it must match the hlr_auc_gw configuration we entered earlier. The HS20 Realm and Domain should be configured to match the HS20 AP.



D. Verify Station connects to the AP and obtains DHCP IP Address configuration. If it does not work, look at the Station's supplicant logs, the AP logs, the RADIUS server logs, and the hlr_auc_gw logs.

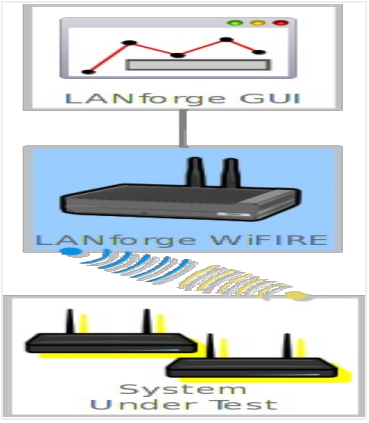
E. For those doing this manually, the wpa_supplicant.conf file looks like this:

```
ctrl_interface=/var/run/wpa_supplicant
fast_reauth=1
concurrent_assoc_ok=1
scan_cur_freq=1
min_scan_gap=5
p2p_disabled=1

# 802.11u / Interworking configuration.
interworking=1
hessid=00:00:00:00:00:33
auto_interworking=1
access_network_type=0
# HotSpot 2.0 configuration
hs20=1
bss_max_count=2000
network={
    interworking_defaults=1
    disable_ht=0
    disable_vht=1
    disable_ht40=0
    disable_sgi=0
    ht_mcs=""
    disable_max_amsdu=-1
    ampdu_factor=-1
    ampdu_density=-1
}
cred={
    username="1232010000000000@mytest.com"
    password="90dca4eda45b53cf0f12d7c9c3bc6a89:cb9cccc4b9258e6dca4760379fb82581"
    realm="mytest.com"
    domain="mytest.com"
    eap=SIM
}
```

For more information see WiFi Station Cookbook

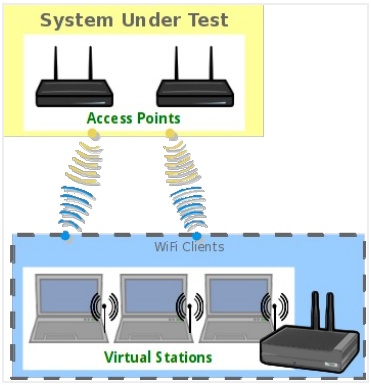
LANforge WiFi Testing Fast Roaming Stations with 802.11r

Goal: Use automated script to migrate stations between APs and report results.

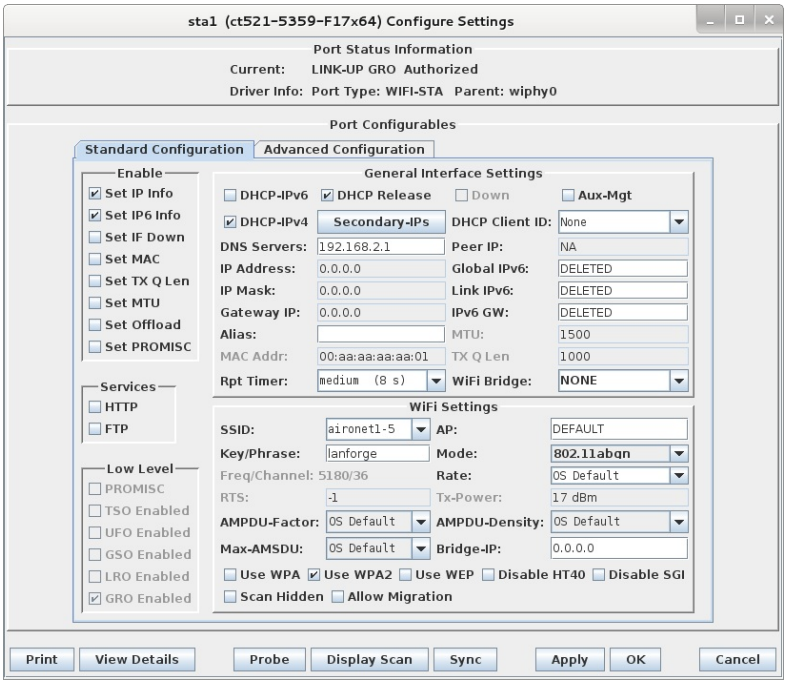


Requires LANforge 5.2.11 or later. Configure Stations to use FT-EAP (802.11r) and associate them with an 802.11r AP network. Use the 'WiFi Mobility' LANforge-GUI Plugin to automate roaming the stations between the APs. The plugin will create graphs and other reports that can be saved to HTML. This example uses a LANforge CT523 system but the procedure should work on all CT520, CT523 and similar systems.

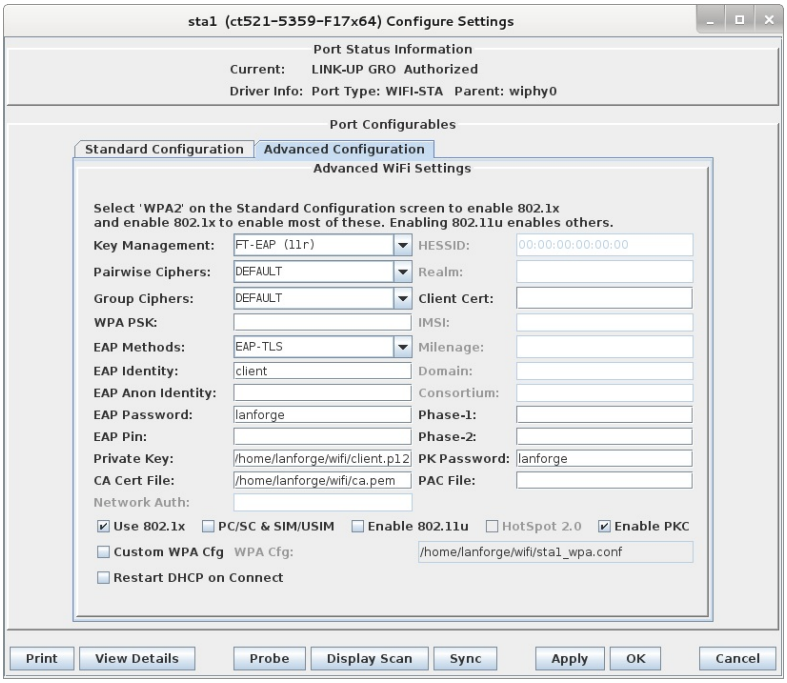
The two APs under test are on the same channel, so a single radio/NIC on LANforge can roam virtual stations between the APs. But, if the APs were on different channels, only a single station per radio would be supported. Multiple CT523 or other high-density systems can be used to migrate stations between APs on different channels.



1. Configure stations to connect to APs configured for 802.11r. This requires special AP software support and usually an AP Controller (APC).
   A. Go to the Port Manager tab, select wiphy0 on proper resource, click Create, fill out appropriate information and create desired number of Station interfaces.
   B. The new stations should appear in the Port-Mgr table. Double-click to modify one of them. Configure IP Address information, SSID and select WPA2:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the Key Management, Private Key and other values needed to connect to the APs. Be sure to un-select the **Restart DHCP on Connect** checkbox so that DHCP is not refreshed each time a station roams:

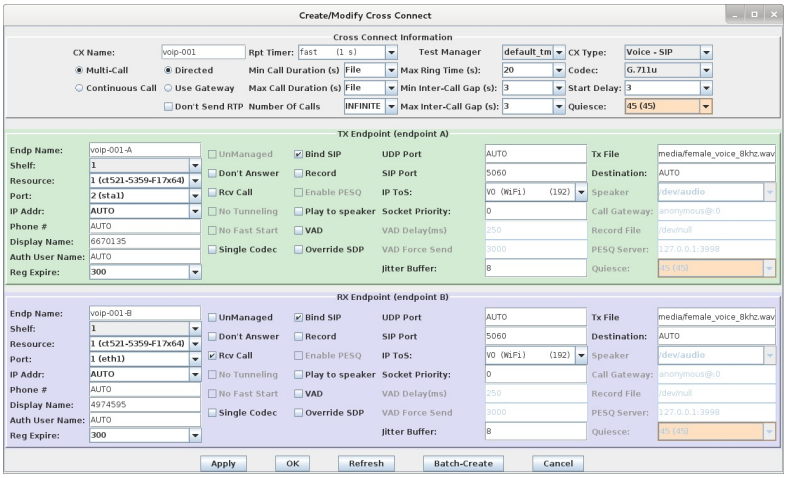


D. Once the single station is connecting properly, use Batch-Modify to configure the rest of the stations to match the first.

For more information see LANforge User's Guide: Ports (Interfaces) , WiFi Station Cookbook

2. Create VOIP connections between the wired Ethernet eth1 interface and the stations. This will add realistic traffic load to the network under test and allow LANforge to report packet-loss statistics during roaming. The VOIP feature costs extra, so you may wish to use a normal Layer-3 UDP connection which should also provide good reports and a realistic traffic load. The steps below are for VOIP, but Layer-3 would be very similar.

A. Go to the VOIP/RTP tab, click Create, and configure a VOIP connection on eth1 and the first station:

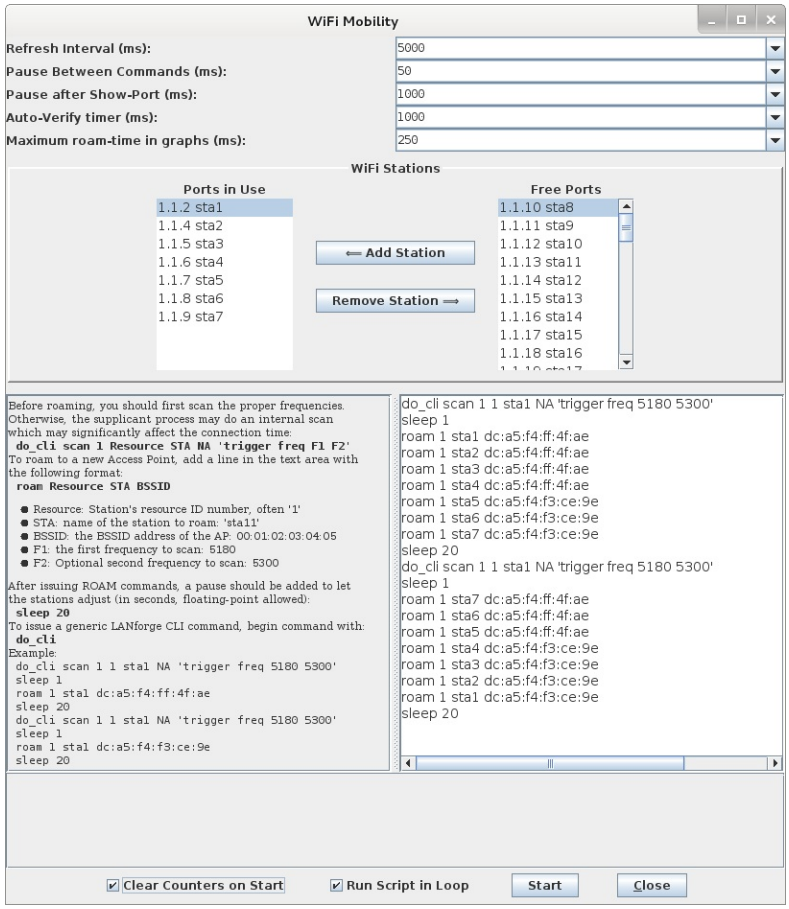


B. Apply the configuration and make sure the call can complete. Then click Modify on the VOIP connection and use Batch-Create to create one connection for each of the WiFi stations.

C. Select the VOIP and/or Layer-3 connections and start traffic flow. For this example, the connections should remain running while the roaming takes place. It would also be valid to do roaming without any traffic if that is the desired test case.

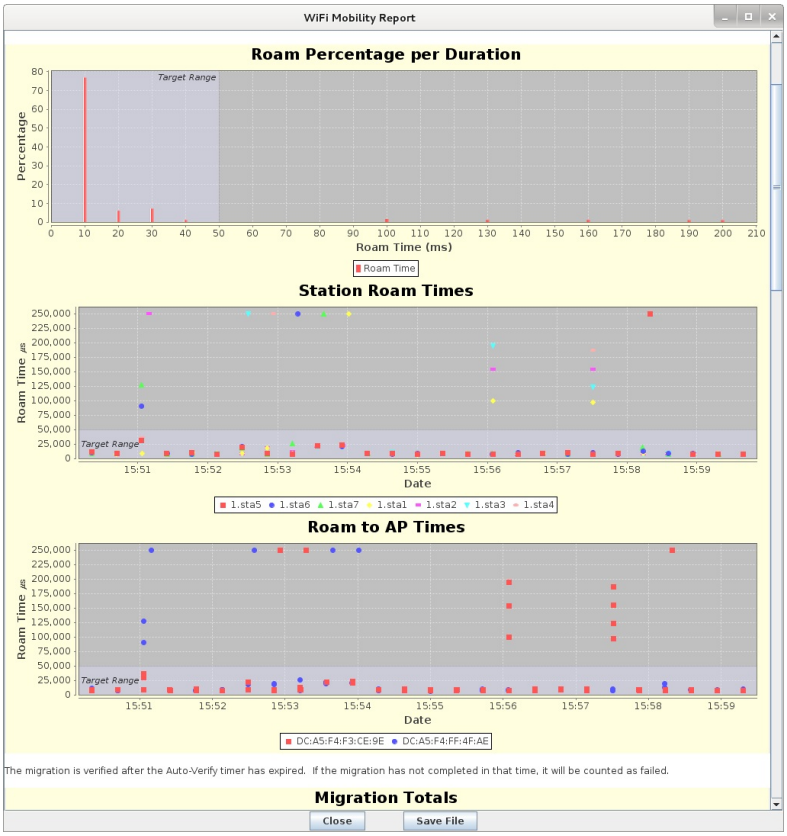
3. Start the WiFi Migration script.

A. Go to the Port Manager tab, select the stations you wish to roam, right-click and choose the **WiFi Mobility** menu option.

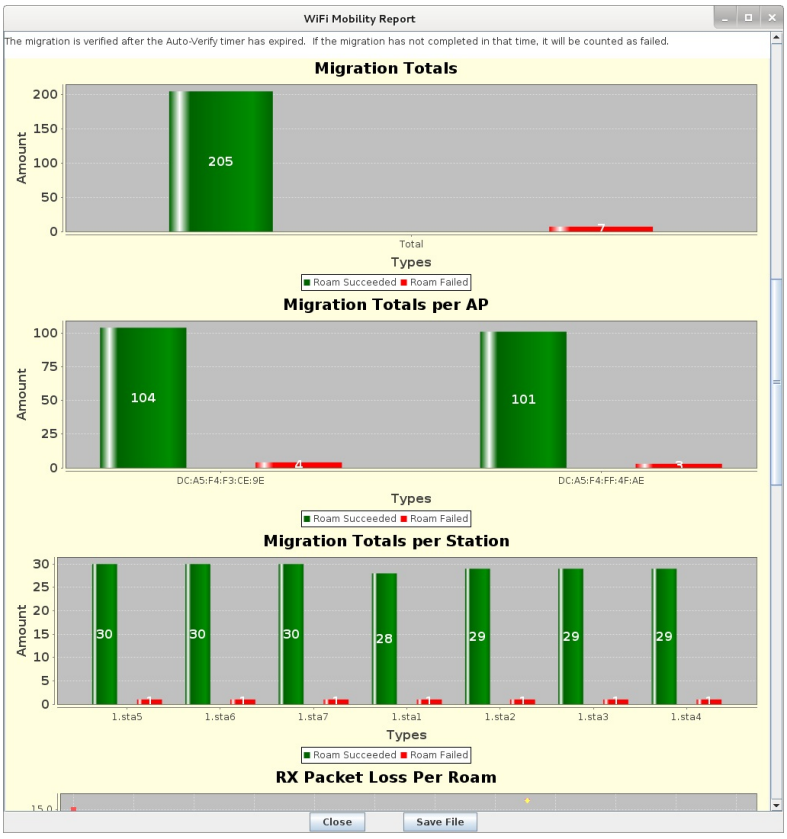


B. The options at the top default to common values and may not need to be changed. The ports will be automatically configured based on the selection on the Port Manager tab, and can be adjusted before starting the script. The Ports in Use should normally include all stations used in the script. The configuration requiring the most work from the user is the roaming script itself. There is a help section on the left, and a script-entry field on the right. Once the script is written, it should be saved in a text file on the user's PC so that it can easily be pasted into future WiFi Mobility scripts. Some key points are that you must scan about 1 second before roaming or the roam logic in the supplicant process will either fail or do it's own roaming. Either way, the results may be worse than if you do the roam properly in the script. It can take a bit of time for LANforge to get all of the data it needs to report on the roam attempt, so it is suggested that stations not roam more often that about once every 10-20 seconds. If reporting is less important, then the stations can roam more often.

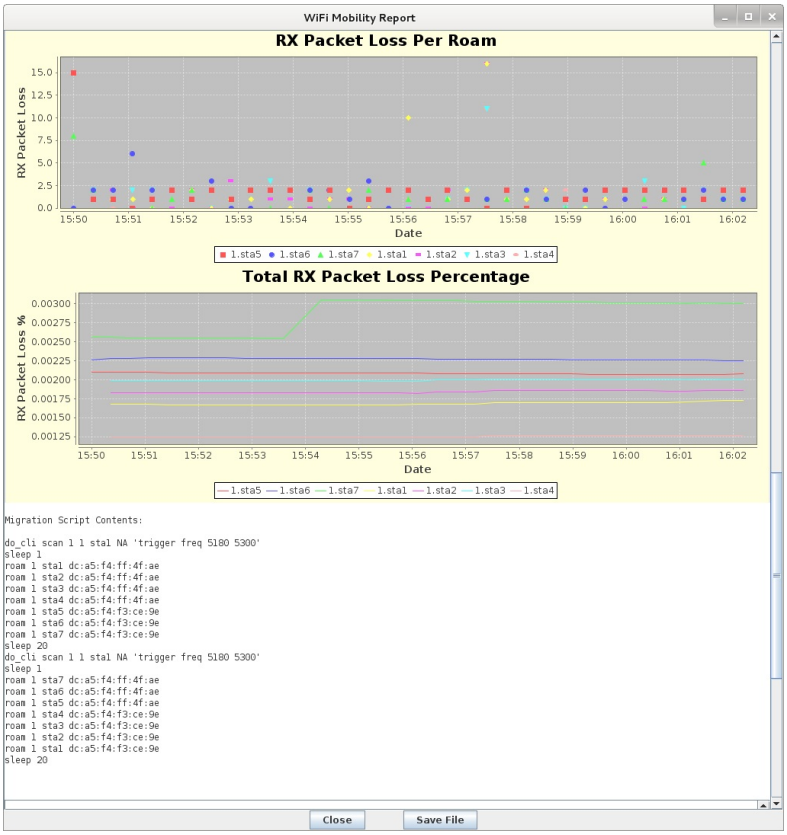
C. Once the script is properly configured, click Start to start the roaming. A window will pop up that has live-updating graphs of various reports. A text log is at the bottom for more detailed analysis, and the whole thing can be saved as HTML. The graphs can be scaled and configured through right-click menus if desired. It will take 1-2 complete roam attempts before the graphs are able to show any useful information.



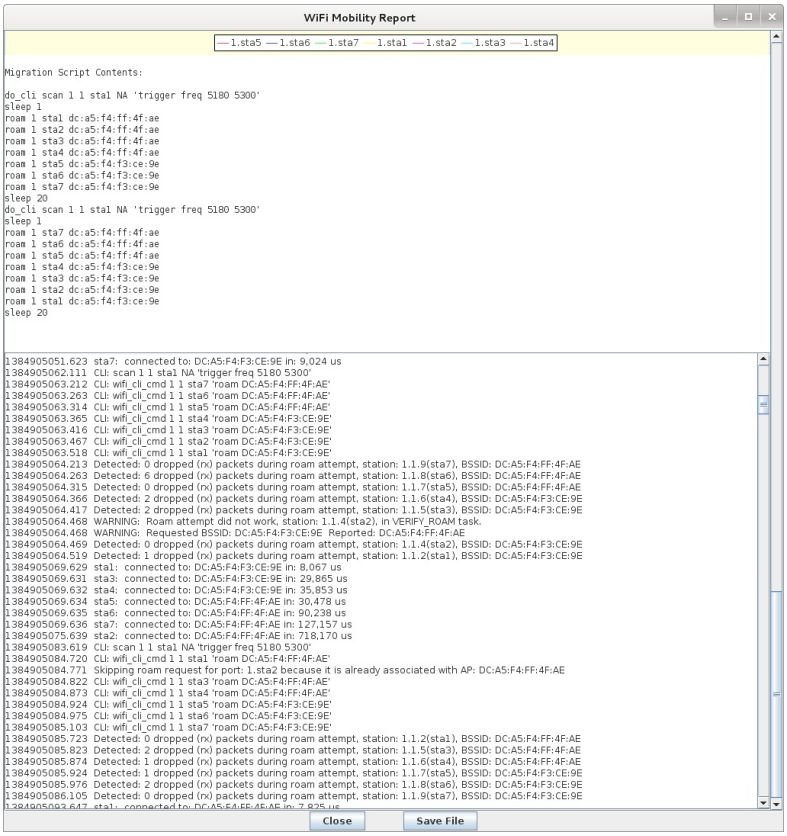
D. Migration totals graphs.



E. Packet Loss graphs.



F. Text log with timestamps. Can be coorelated with wpa_supplicant logs and other log files to debug specific roam attempts.



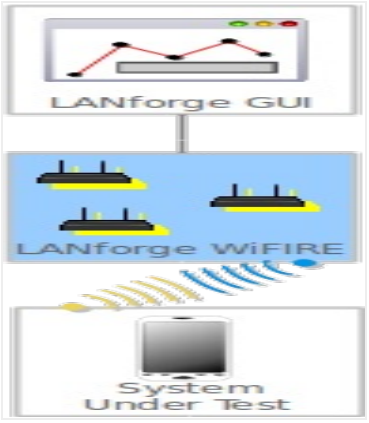
LANforge WiFi Access Point Network with 802.11r

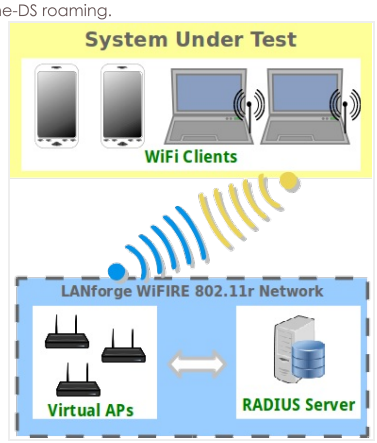
Goal: Configure a virtual AP network with 802.11r to allow testing fast transition (FT) clients.

Configure virtual Access Points to use 802.11r with FT-EAP. This example uses a LANforge CT523 system but the procedure will work on all CT522, CT523 and CT525 multi-radio systems.

The wifi clients under test are also 802.11r enabled so that they can initiate FT Requests and roam. Here we are using another LANforge WiFire as the system under test to emulate 802.11r stations and force them to roam.

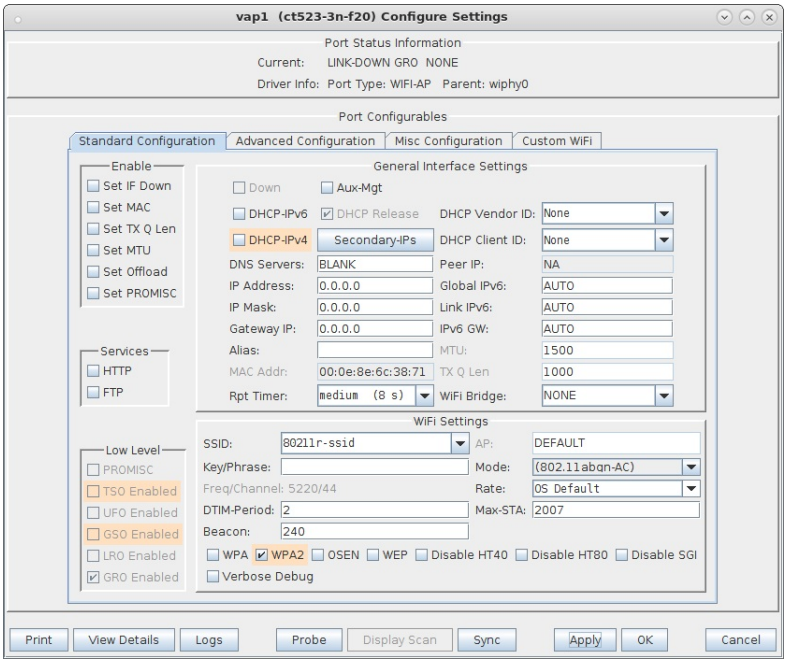
In LANforge, each virtual access point will be running its own hostapd process configured to enable 802.11r and bridged to other virtual access points. The bridged VAP network will emulate the Distributed System (DS) for FT over-the-DS roaming.



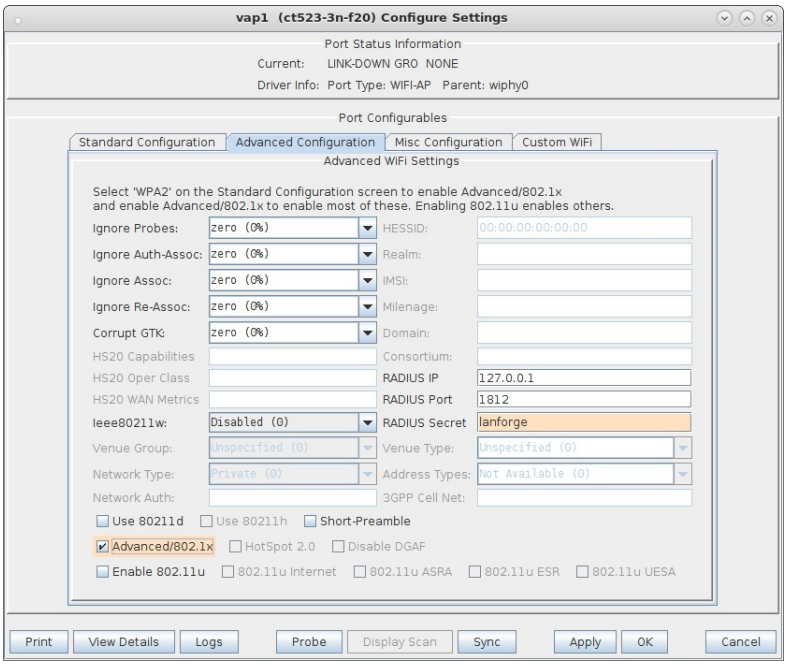


1. Setup a single virtual access point on each wifi NIC for at least two NICs and configure them for the same channel and SSID.
    A. Go to the Port Manager tab, select the parent device such as wiphy0, click Modify, set a specific channel/frequency. Repeat for wiphy1.
    B. Select wiphy0, click Create, fill out appropriate information and create a virtual access point. Repeat for wiphy1.

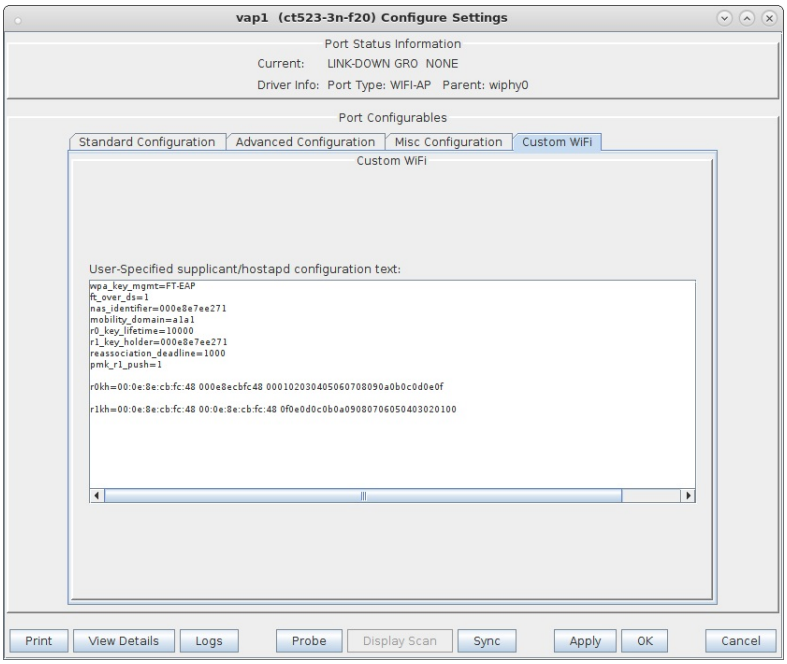
C. The new vap should appear in the Port-Mgr table. Double-click to modify. Configure SSID and select WPA2 but do not fill in the Key/Phrase:



D. Select the **Advanced Configuration** tab in the Port-Modify window and check the box Advanced/802.1x and fill in the RADIUS IP/Port/Secret. Here the RADIUS server will be another instance of hostapd configured on a bridge interface and accessible via localhost.



E. Select the **Custom WiFi** tab in the Port-Modify window to fill in the additional hostapd options to enable and configure 802.11r. These lines will be appended to the end of the LANforge generated hostapd configuration file located in /home/lanforge/wifi of the resource in use.



vap1 00:0e:8e:7e:e2:71 - Your MAC will be different.

```
wpa_key_mgmt=FT-EAP
ft_over_ds=1
nas_identifier=000e8e7ee271   #vap1 MAC without colon delimiters, yours will differ.
mobility_domain=a1a1
r0_key_lifetime=10000
r1_key_holder=000e8e7ee271   #vap1 MAC without colon delimiters, yours will differ.
reassociation_deadline=1000
pmk_r1_push=1

#r0kh is vap2 MAC address, vap2 nas identifier, AES key
r0kh=00:0e:8e:cb:fc:48 000e8ecbfc48 000102030405060708090a0b0c0d0e0f

#r1kh is vap2 MAC address, vap2 r1 key holder MAC, AES key
r1kh=00:0e:8e:cb:fc:48 00:0e:8e:cb:fc:48 0f0e0d0c0b0a09080706050403020100
```

full configuration file: hostapd_vap1.conf

vap2 00:0e:8e:cb:fc:48 - Your MAC will be different.

```
wpa_key_mgmt=FT-EAP
ft_over_ds=1
nas_identifier=000e8ecbfc48   #vap2 MAC without colon delimeters, yours will differ.
mobility_domain=a1a1
r0_key_lifetime=10000
r1_key_holder=000e8ecbfc48   #vap2 MAC without colon delimeters, yours will differ.
reassociation_deadline=1000
pmk_r1_push=1

#r0kh is vap1 MAC address, vap1 nas identifier, AES key
r0kh=00:0e:8e:7e:e2:71 000e8e7ee271 0f0e0d0c0b0a09080706050403020100

#r1kh is vap1 MAC address, vap1 r1 key holder MAC, AES key
r1kh=00:0e:8e:7e:e2:71 00:0e:8e:7e:e2:71 000102030405060708090a0b0c0d0e0f
```

full configuration file: hostapd_vap2.conf

If you wanted to add FT-PSK capability, add the following to the hostapd configuration file:

```
wpa_key_mgt=FT-EAP FT-PSK
wpa_passphrase=1234567890
```

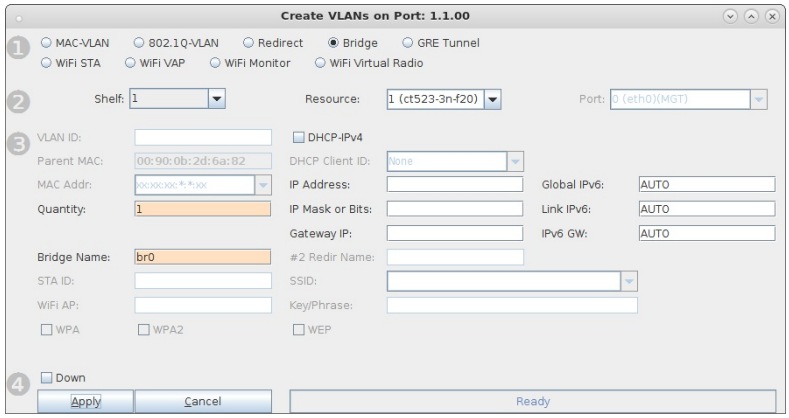
In this example, we are configuring push mode key distribution where the master key holder, R0KH, derives the R1 key for all secondary key holders, R1KH, listed in the configuration file and sends it to them over the DS via bridge interfaces. The R0KH and R1KH entries must be configured for all virtual access points in the 802.11r network.

For more information on hostapd 802.11r configuration, see:
general hostapd configuration
https://www.w1.fi/cgit/hostap/plain/hostapd/hostapd.conf
how to enable wifi roaming
https://blog.fem.tu-ilmenau.de/archives/1002-HowTo-enable-WiFi-roaming-with-hostapd-and-VLANs.html
802.11r hostapd example
ftp://ftp.raspberry-pi-geek.com/pub/listings/rasp-pi-geek.com/04/AccessPoint/Listing04.txt

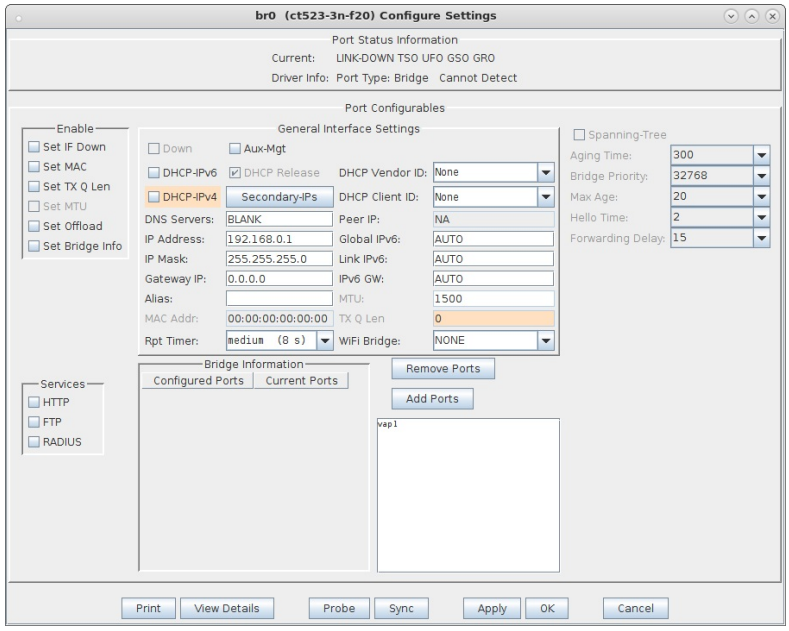
F. Repeat above steps A-E for vap2 on wiphy1.

2. Create a bridge device for the first virtual access point, vap1. This bridge will be placed inside a virtual router so that it can serve DHCP requests and act as a RADIUS authentication server.

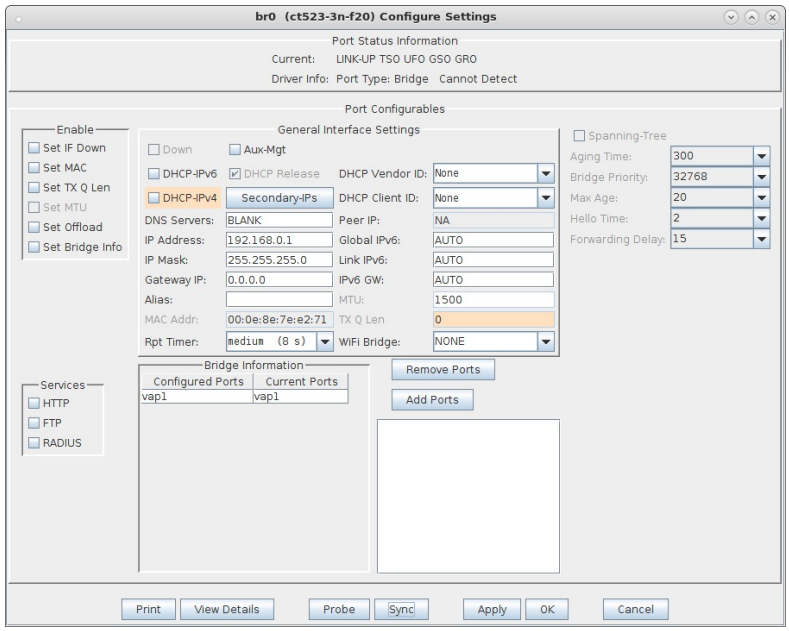
A. Go to the port manager tab, select Create, then select Bridge and enter Quantity 1 and a Bridge Name, then Apply to create the bridge.



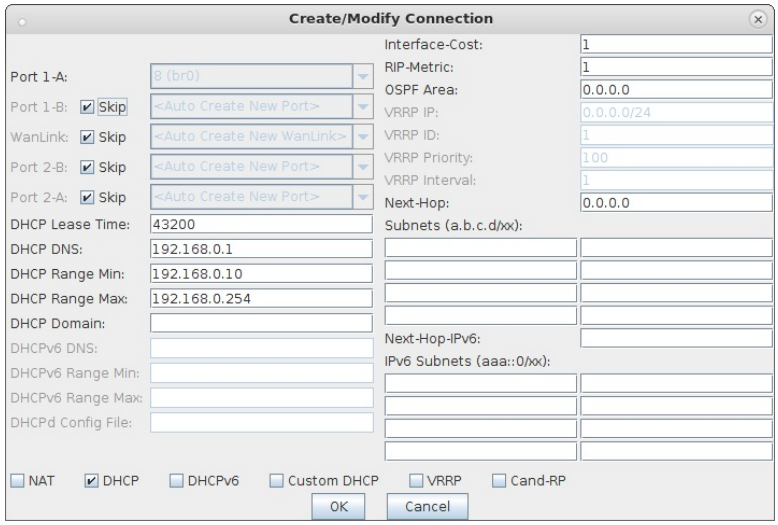
B. Modify the new bridge device to add vap1. Type vap1 in the text entry box, then select Add Ports, then select Apply.



C. Select Sync to verify vap1 is a configured and current bridge member.



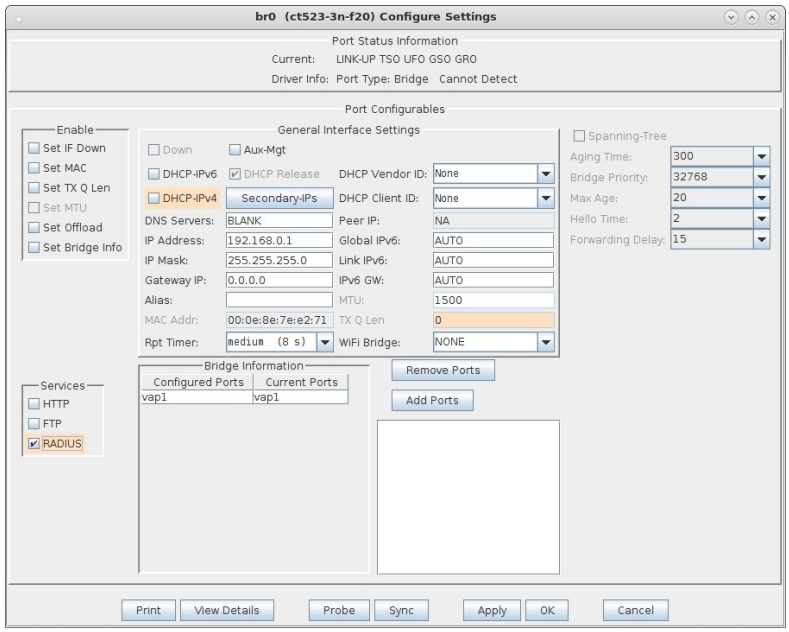
D. Go to Netsmith, right-click the bridge and select Modify to add DHCP service. Select the DHCP checkbox at the bottom, then fill in the DHCP Lease Time, DHCP DNS, DHCP Range Min, DHCP Range Max and DHCP Domain if needed, then select OK.



E. Go to Netsmith, right-click in a free area and select New Router and select OK. Then drag the bridge br0 into the virtual router and select Netsmith Apply.

For more information see Virtual Router with DHCP Cookbook (skip the wanlink portion)

3. Add a RADIUS server to the bridge device.
   A. Go to Netsmith, right-click the bridge and select Modify Port to add RADIUS service.
   B. Select the RADIUS checkbox, then select OK.



C. Setup the following configuration files to start the RADIUS service. You will need to create these files, but the certificate files can be created by running the lf_kinstall script with the --do_radius option.

/etc/hostapd.radius_clients

```
0.0.0.0/0 lanforge
```

/etc/hostapd.eap_user

```
"dot11r.user" PEAP
"dot11r.user" MSCHAPV2 "!!dot11r123" [2]
```

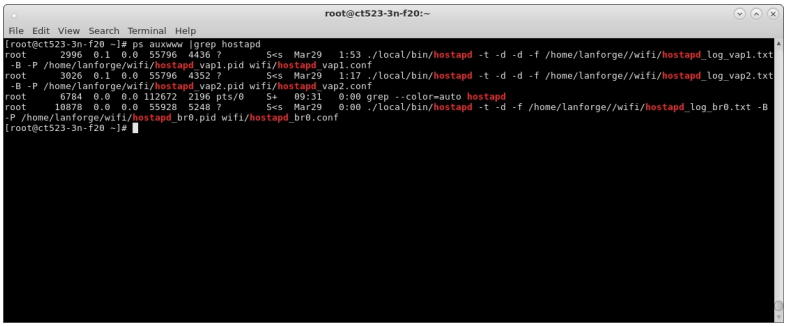
/home/lanforge/wifi/hostapd_br0.conf

```
interface=br0
driver=wired
logger_syslog=-1
logger_syslog_level=2
logger_stdout=-1
logger_stdout_level=2
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0
eapol_key_index_workaround=0
eap_server=1
eap_user_file=/etc/hostapd.eap_user
server_id=ct523-3n-f20   #Your server_id will be different.
radius_server_auth_port=1812
radius_server_clients=/etc/hostapd.radius_clients

ca_cert=/etc/raddb/certs/ca.pem
server_cert=/etc/raddb/certs/server.pem
private_key=/etc/raddb/certs/server.key
private_key_passwd=lanforge
```

D. Verify that there are three hostapd processes running with the command:

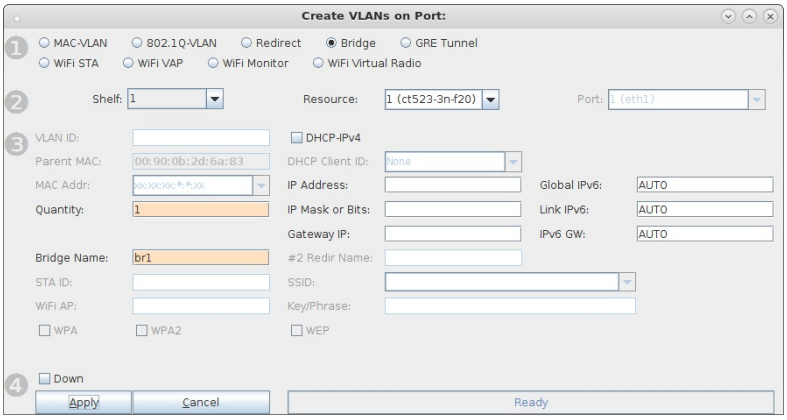
```
ps auxwww |grep hostapd
```

which should show something similar to the following:

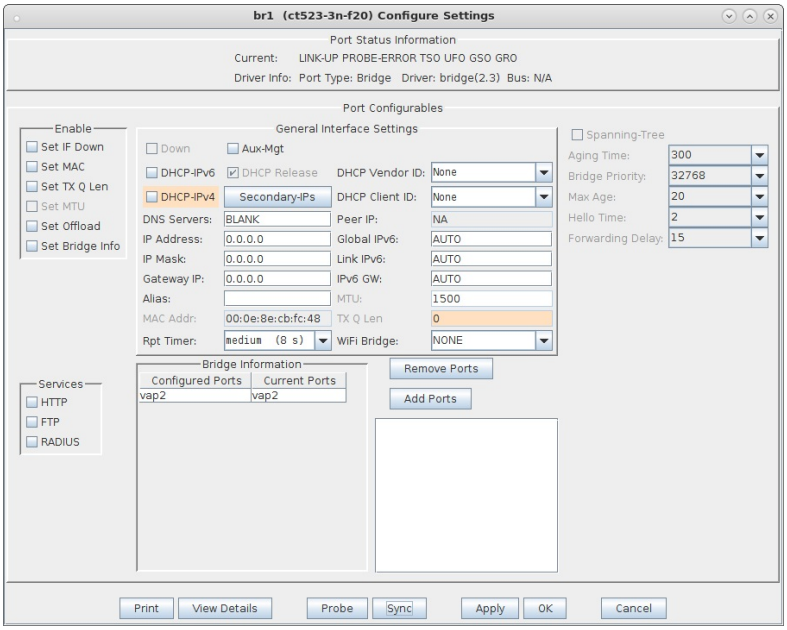


4. Create a second bridge device for the second virtual access point, vap2.
Each vap in the 802.11r network requires its own bridge so that the bridge device receive logic can correctly process packets from each vap during fast-transition client roaming.

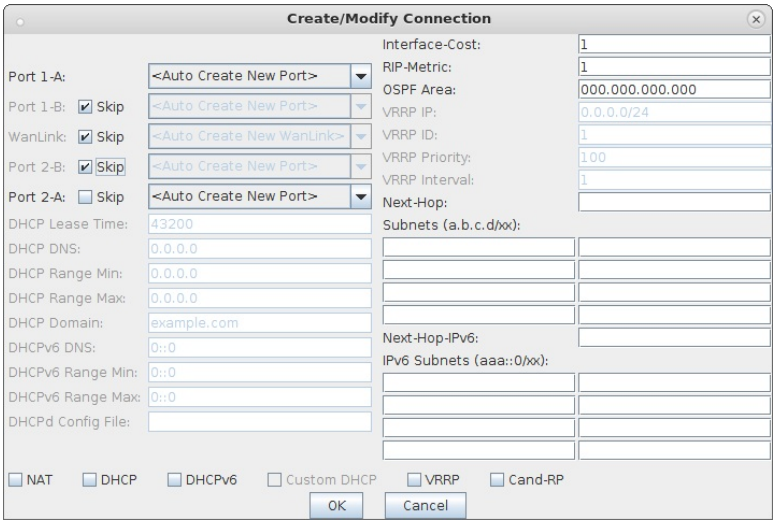
   A. Go to Netsmith, right-click in a free area, select New Bridge, enter Quantity 1 and a Bridge Name, then select Apply. Sync Netsmith to view the new bridge.



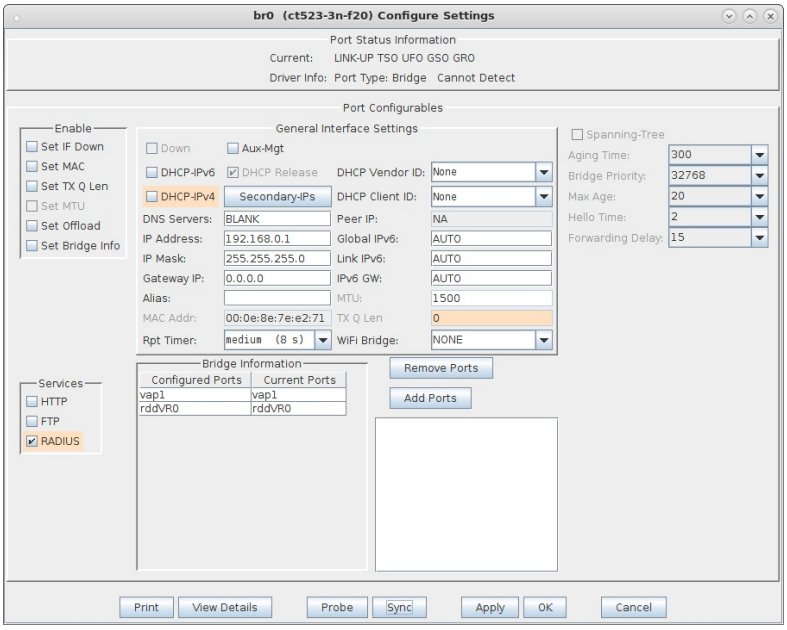
B. Right-click the new bridge and select Modify Port to add vap2 as a bridge member.



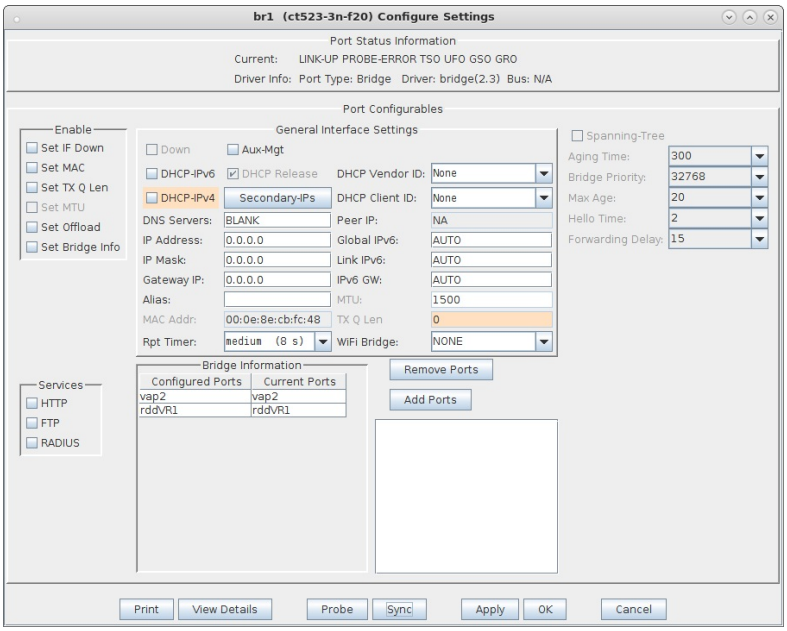
5. Each bridge will share a connection to a redirect device (rdd) pair so that FT messages can be sent and received.
   A. In Netsmith, right-click in a free area and select New Connection to create an rdd pair. Select **Skip** for Port 1-B, WanLink and Port 2-B then select OK. Select Netsmith Apply after creating the new connection.



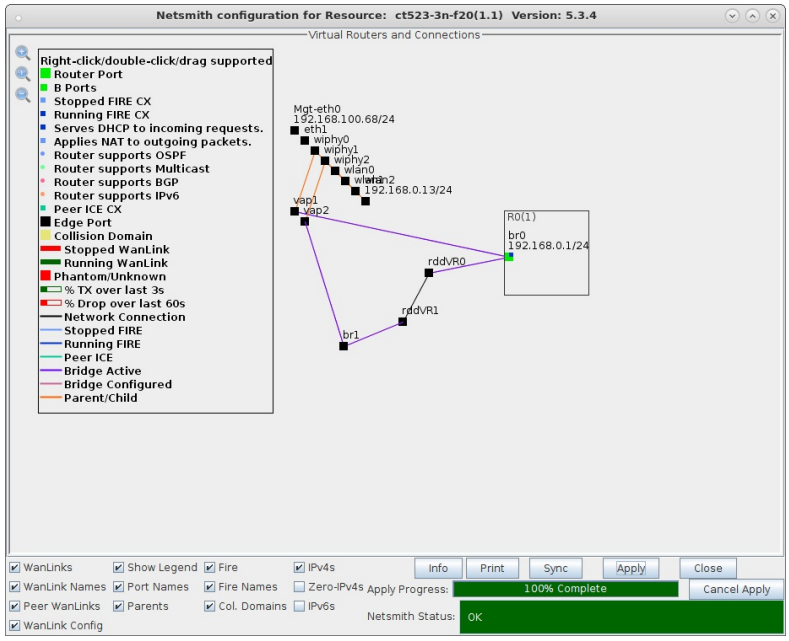
B. Right-click and select Modify Port br0, then add rddVR0 to br0, select Add Ports then select Apply. Your rddVRX numbering may differ depending on what other Netsmith objects are created.



C. Right-click and select Modify Port br1, then add rddVR1 to br1, select Add Ports then select Apply. Your rddVRX numbering may differ depending on what other Netsmith objects are created.

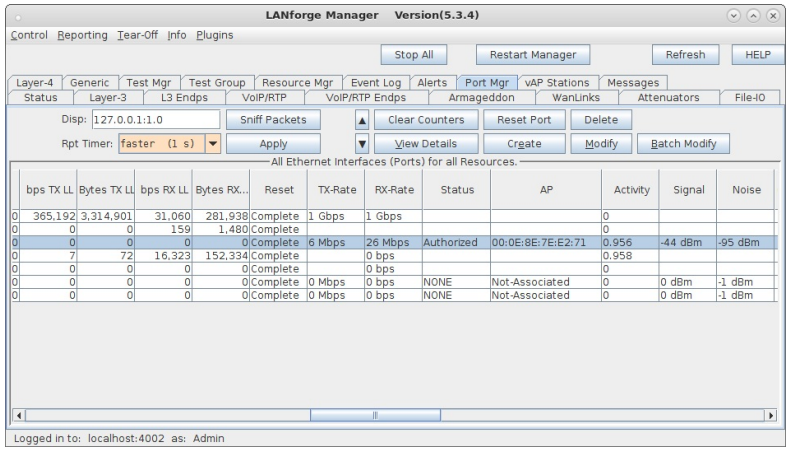


D. The final Netsmith display should show the two bridged virtual access points connected by a rdd pair

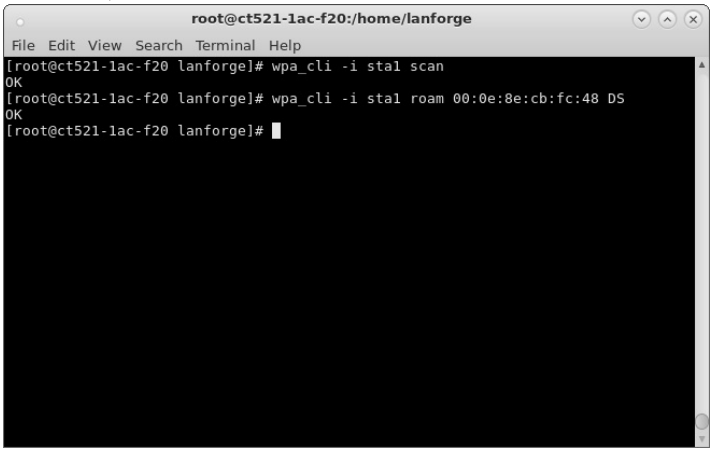


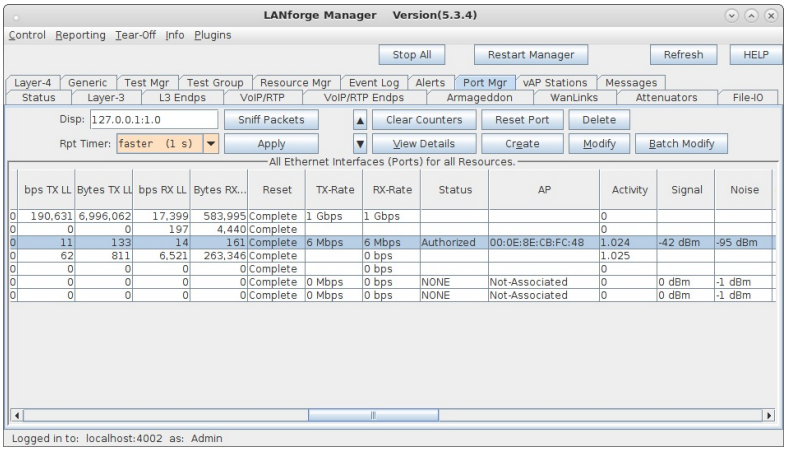
6. Connect clients and force them to roam from vap to vap. This can be accomplished with a wpa_cli command for one or two clients or the Mobility Plugin Script for many clients. If the system under test is not able to force a roam, a variable attenuator on each vap radio may help induce a client to roam as the signal strength from vap to vap is varied.

   A. Client connected to vap1.

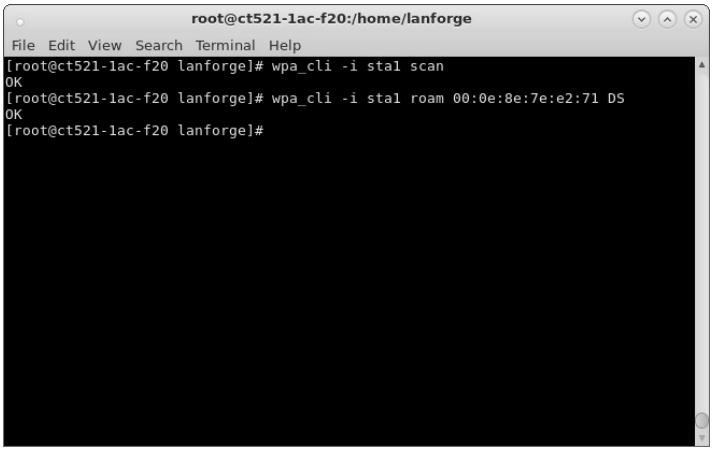


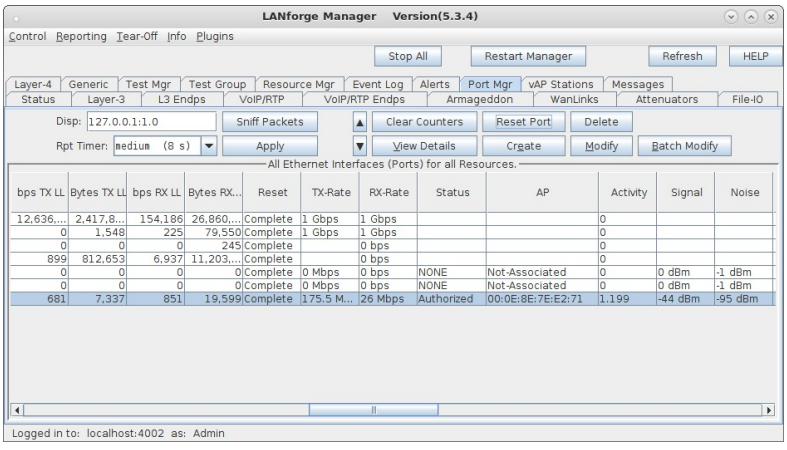
B. Client roams to vap2.



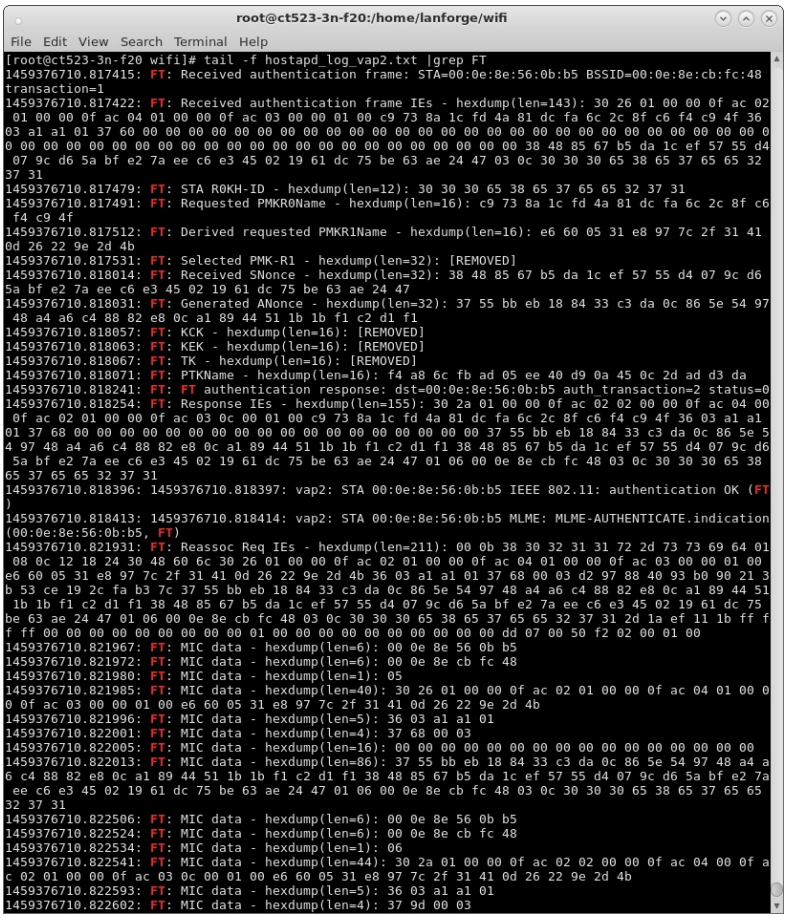


C. Client roams back to vap1.

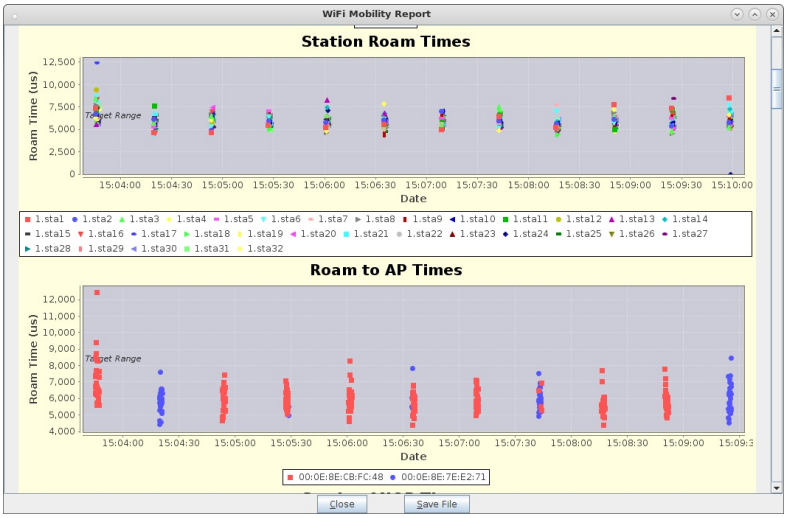




D. FT messaging in hostapd logs.


root@ct523-3n-f20:/home/lanforge/wifi
File Edit View Search Terminal Help
[root@ct523-3n-f20 wifi]# tail -f hostapd_log_vap2.txt |grep FT
1459376710.817415: FT: Received authentication frame: STA=00:0e:8e:56:0b:b5 BSSID=00:0e:8e:cb:fc:48 transaction=1
1459376710.817422: FT: Received authentication frame IEs - hexdump(len=143): 30 26 01 00 00 0f ac 02 01 00 00 0f ac 04 01 00 00 0f ac 03 00 00 01 00 c9 73 8a 1c fd 4a 81 dc fa 6c 2c 8f c6 f4 c9 4f 36 03 a1 a1 01 37 60 00 00 00

F. Output graph of the Mobility Plugin script of several roaming stations.



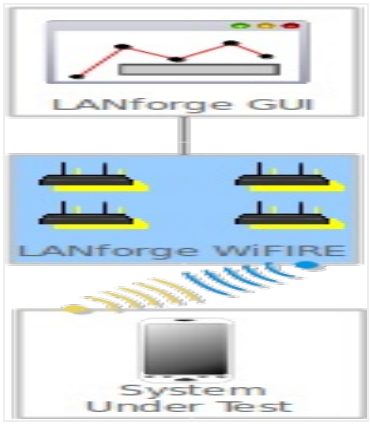
LANforge WiFi Access Point Network with 802.11r

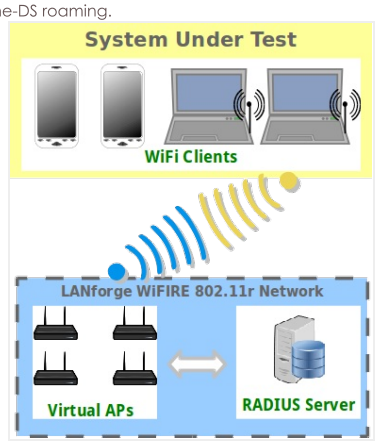
Goal: Configure four virtual APs in an 802.11r network to test fast transition (FT) clients.

Configure four virtual Access Points to use 802.11r with FT-EAP. This example uses a LANforge CT525 system but a similar procedure will work on all multi-radio systems.

The wifi clients under test are also 802.11r enabled so that they can initiate FT Requests and roam. Here we are using another LANforge WiFire as the system under test to emulate 802.11r stations and force them to roam.

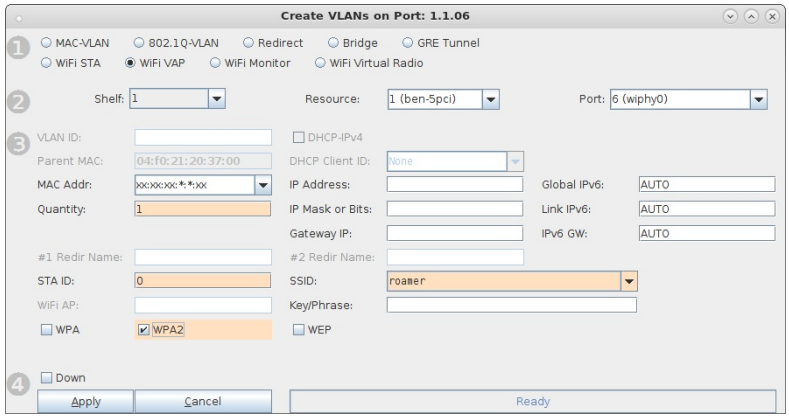
In LANforge, each virtual access point will be running its own hostapd process configured to enable 802.11r and bridged to other virtual access points. The bridged VAP network will emulate the Distributed System (DS) for FT over-the-DS roaming.



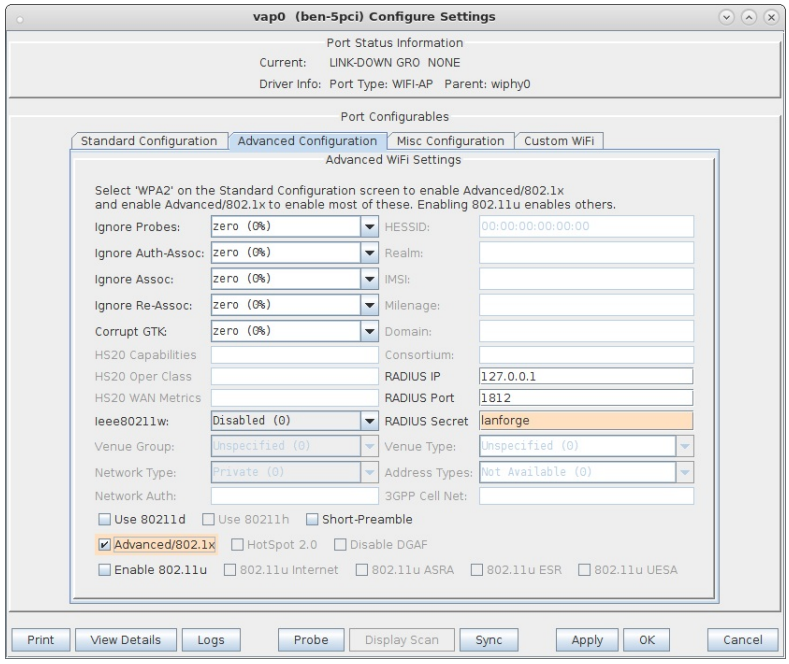


1. Setup a virtual access points on a wifi NIC and configure it for a channel and SSID.

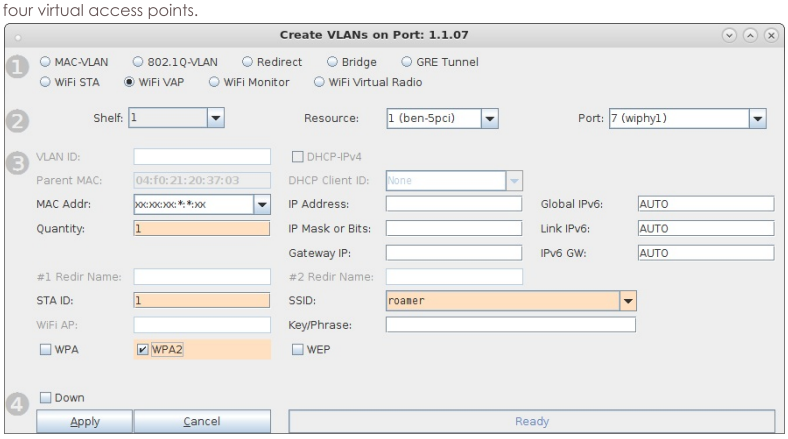
A. Go to the Port Mgr tab, select the parent device such as wiphy0, select Create, then fill out appropriate information and create a virtual access point.

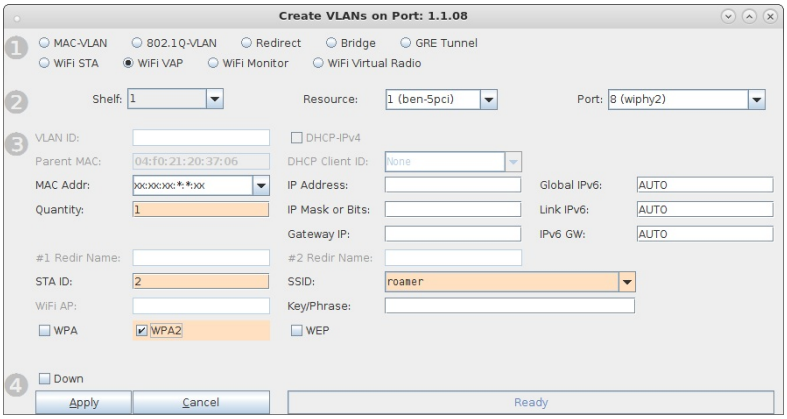


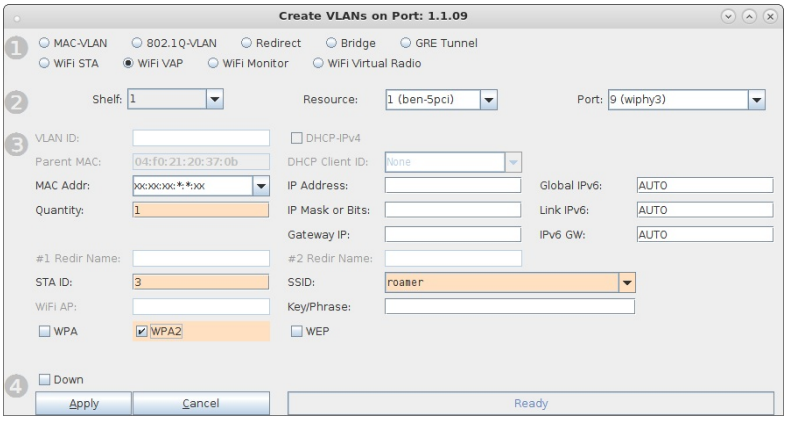
B. Modify the new vap, select the **Advanced Configuration** tab in the Port-Modify window and check the box Advanced/802.1x and fill in the RADIUS IP/Port/Secret. Here the RADIUS server will be a freeradius process configured on localhost. The default RADIUS secret is **lanforge** and the default login is **testuser/testpasswd** when freeradius is installed via lf_kinstall.pl --do_radius.



C. Repeat above steps A and B to create a single vap on wiphy1, wiphy2 and wiphy3 for a total of four virtual access points.





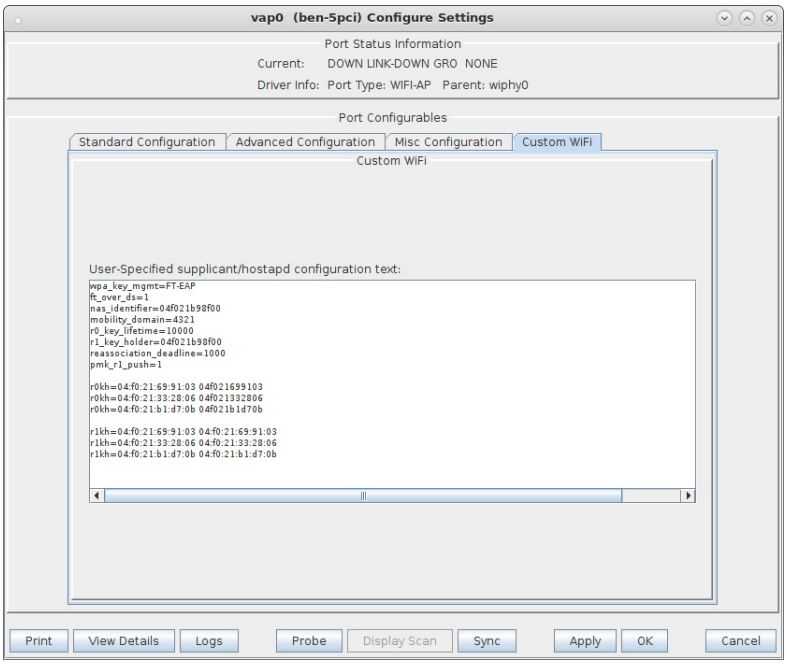


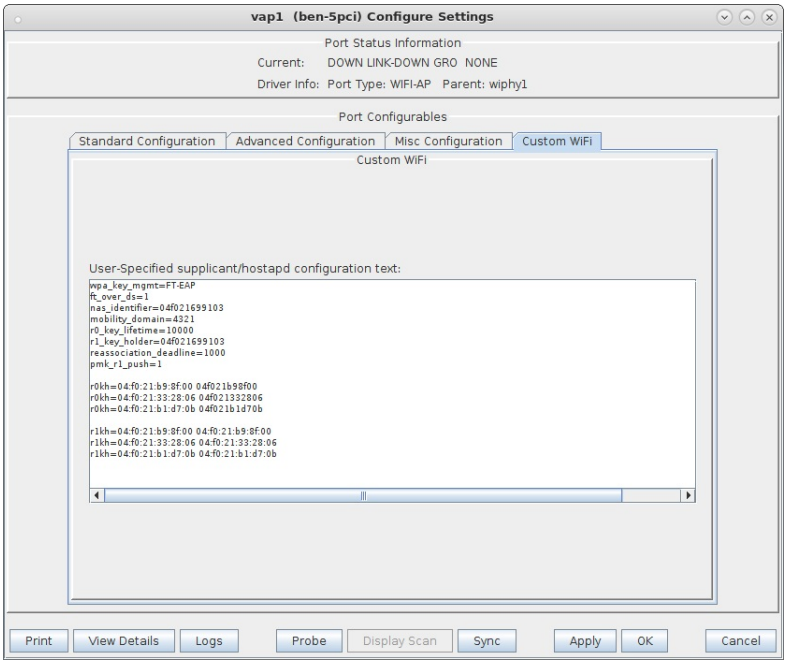
2. Modify each vap, select the **Custom Wifi** tab and enter the additional lines that will be appended to each vap hostapd configuration file.

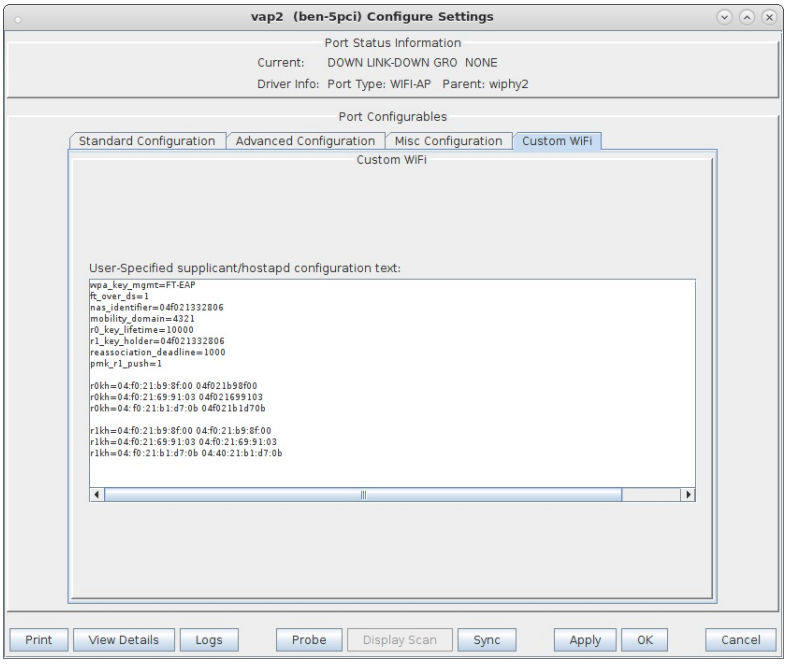
   A. For each vap, fill in the first 8 lines to enable 802.11r as well as the **first** part of the 3 r0kh entries and 3 r1kh entries for neighboring vap's.

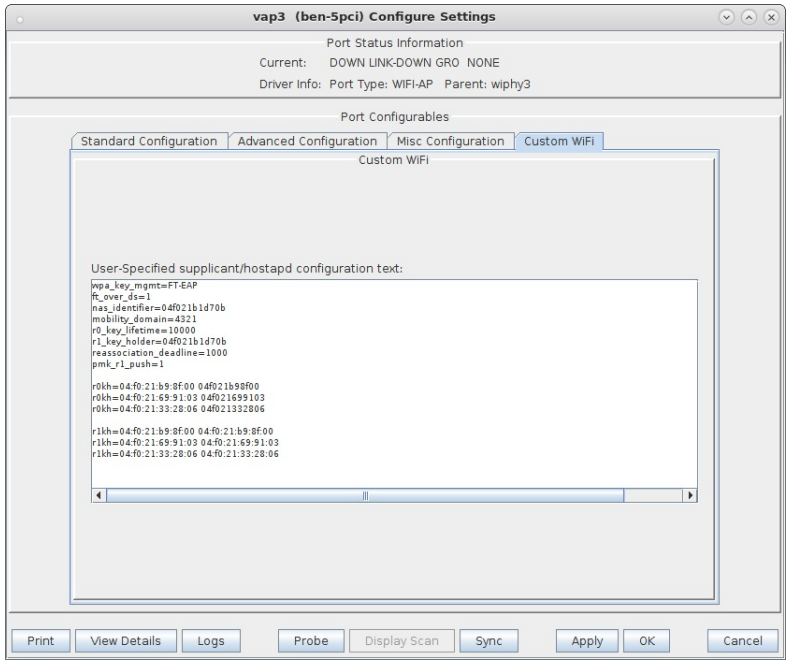
   The first part of the **r0kh** is the neighboring vap MAC address and NAS identifier which in this example is the MAC without colon delimeters.

   The first part of the **r1kh** is the neighboring vap MAC address and r1kh-id which in this example are the same.

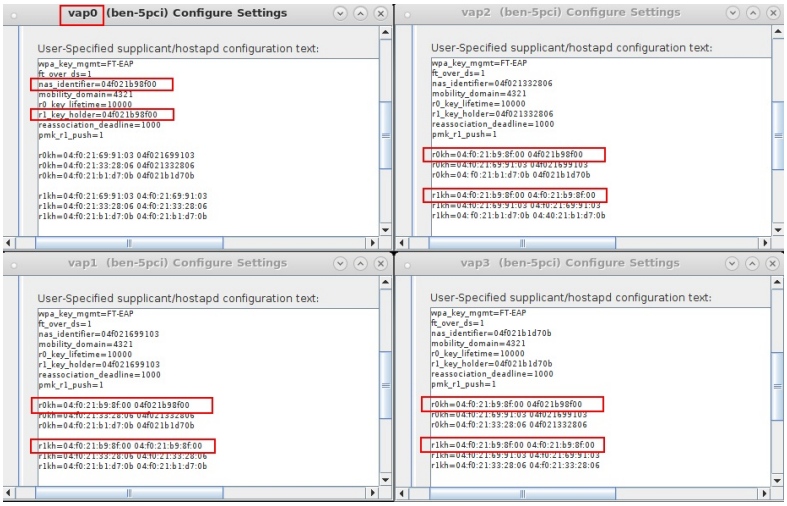

vap0 (ben-5pci) Configure Settings
Port Status Information
Current: DOWN LINK-DOWN GRO NONE
Driver Info: Port Type: WIFI-AP Parent: wiphy0
Port Configurables
Standard Configuration
Advanced Configuration
Misc Configuration
Custom WiFi
Custom WiFi
User-Specified supplicant/hostapd configuration text:
wpa_key_mgmt=FT-EAP
ft_over_ds=1
nas_identifier=04f021b98f00
mobility_domain=4321
r0_key_lifetime=10000
r1_key_holder=04f021b98f00
reassociation_deadline=1000
pmk_r1_push=1
r0kh=04:f0:21:69:91:03 04f021699103
r0kh=04:f0:21:33:28:06 04f021332806
r0kh=04:f0:21:b1:d7:0b 04f021b1d70b
r1kh=04:f0:21:69:91:03 04:f0:21:69:91:03
r1kh=04:f0:21:33:28:06 04:f0:21:33:28:06
r1kh=04:f0:21:b1:d7:0b 04:f0:21:b1:d7:0b
Print
View Details
Logs
Probe
Display Scan
Sync
Apply
OK
Cancel


vap1 (ben-5pci) Configure Settings
Port Status Information
Current: DOWN LINK-DOWN GRO NONE
Driver Info: Port Type: WIFI-AP Parent: wiphy1
Port Configurables
Standard Configuration
Advanced Configuration
Misc Configuration
Custom WiFi
Custom WiFi
User-Specified supplicant/hostapd configuration text:
wpa_key_mgmt=FT-EAP
ft_over_ds=1
nas_identifier=04f021699103
mobility_domain=4321
r0_key_lifetime=10000
r1_key_holder=04f021699103
reassociation_deadline=1000
pmk_r1_push=1
r0kh=04:f0:21:b9:8f:00 04f021b98f00
r0kh=04:f0:21:33:28:06 04f021332806
r0kh=04:f0:21:b1:d7:0b 04f021b1d70b
r1kh=04:f0:21:b9:8f:00 04:f0:21:b9:8f:00
r1kh=04:f0:21:33:28:06 04:f0:21:33:28:06
r1kh=04:f0:21:b1:d7:0b 04:f0:21:b1:d7:0b
Print
View Details
Logs
Probe
Display Scan
Sync
Apply
OK
Cancel

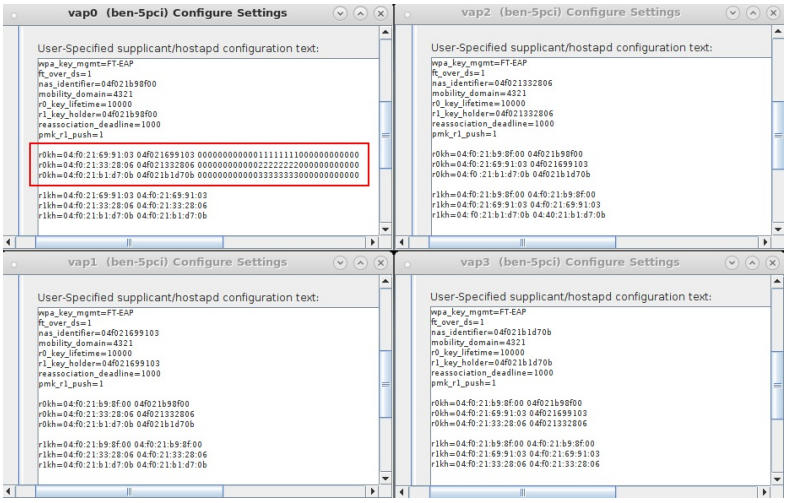

vap2 (ben-5pci) Configure Settings
Port Status Information
Current: DOWN LINK-DOWN GRO NONE
Driver Info: Port Type: WIFI-AP Parent: wiphy2
Port Configurables
Standard Configuration
Advanced Configuration
Misc Configuration
Custom WiFi
Custom WiFi
User-Specified supplicant/hostapd configuration text:
wpa_key_mgmt=FT-EAP
ft_over_ds=1
nas_identifier=04f021332806
mobility_domain=4321
r0_key_lifetime=10000
r1_key_holder=04f021332806
reassociation_deadline=1000
pmk_r1_push=1
r0kh=04:f0:21:b9:8f:00 04f021b98f00
r0kh=04:f0:21:69:91:03 04f021699103
r0kh=04:f0:21:b1:d7:0b 04f021b1d70b
r1kh=04:f0:21:b9:8f:00 04:f0:21:b9:8f:00
r1kh=04:f0:21:69:91:03 04:f0:21:69:91:03
r1kh=04:f0:21:b1:d7:0b 04:40:21:b1:d7:0b
Print
View Details
Logs
Probe
Display Scan
Sync
Apply
OK
Cancel



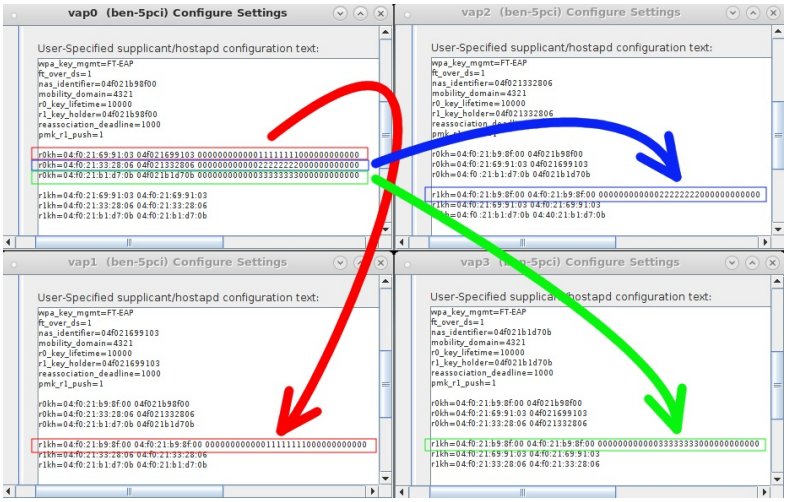
B. Each vap will be represented by a corresponding r0kh and r1kh entry in all of the neighboring vap's custom configuration sections. Here vap0 entries are highlighted.



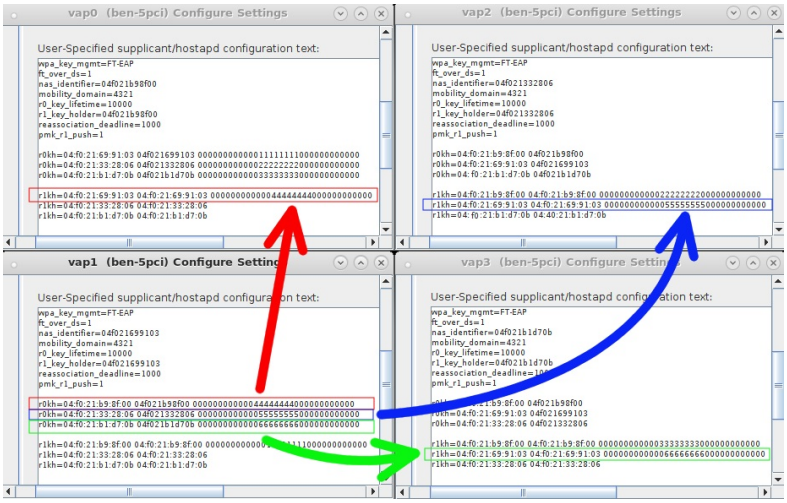
C. Generate a unique 128-bit AES hex key for each r0kh entry in the vap.

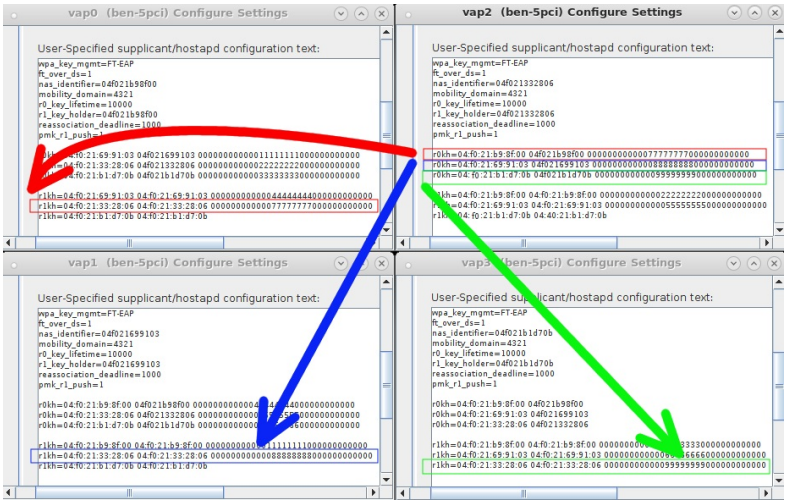


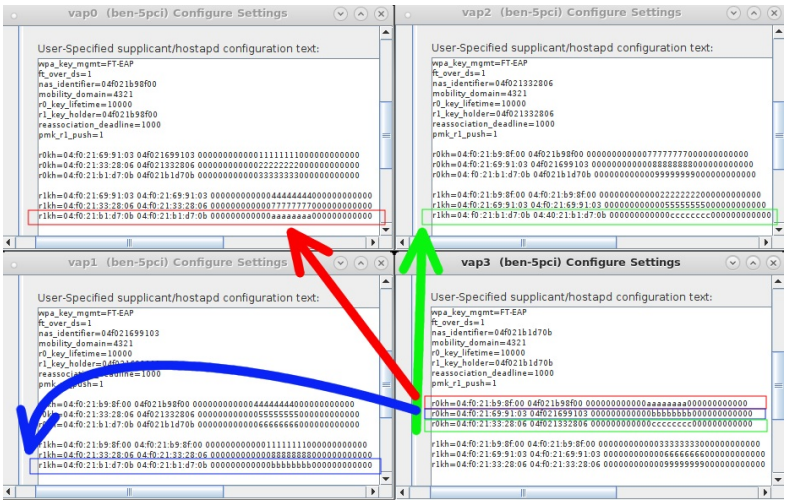
D. Copy the r0kh keys to the r1kh lines that correspond to the vap.



E. Repeat steps C and D for the other three vap's.



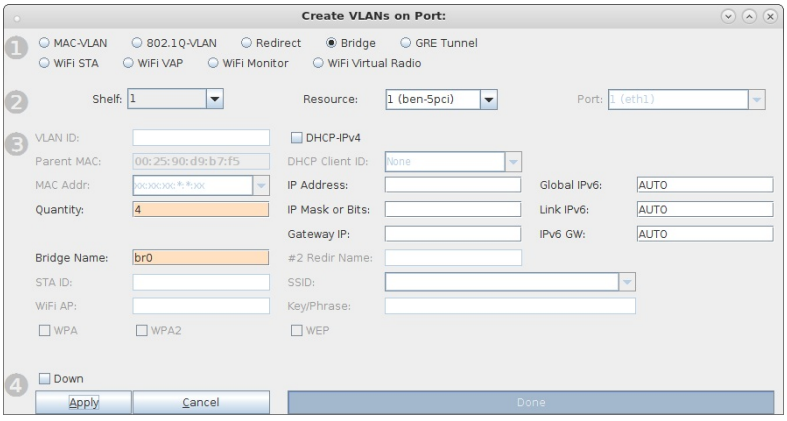




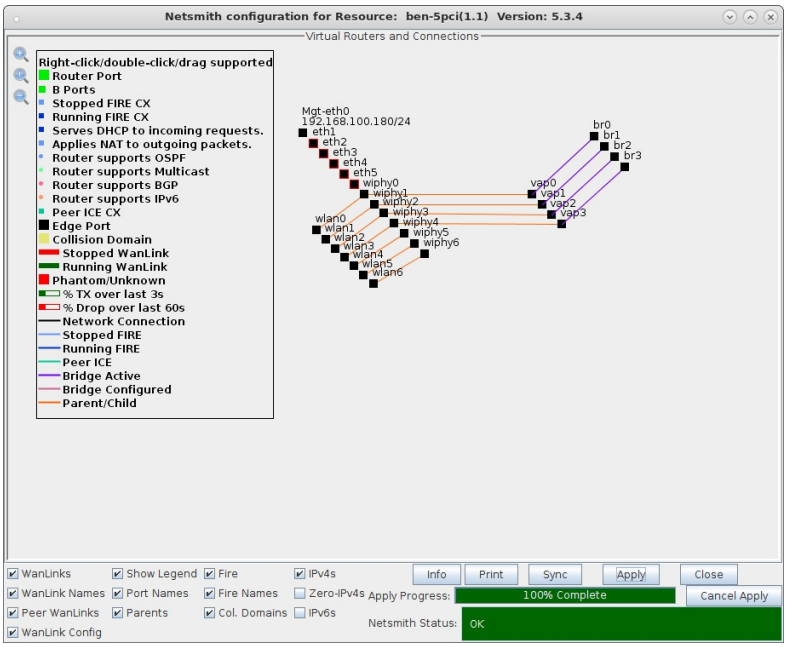
vap0: full hostapd configuration file
vap1: full hostapd configuration file
vap2: full hostapd configuration file
vap3: full hostapd configuration file

For more information see Two WiFi Access Point Network with 802.11r

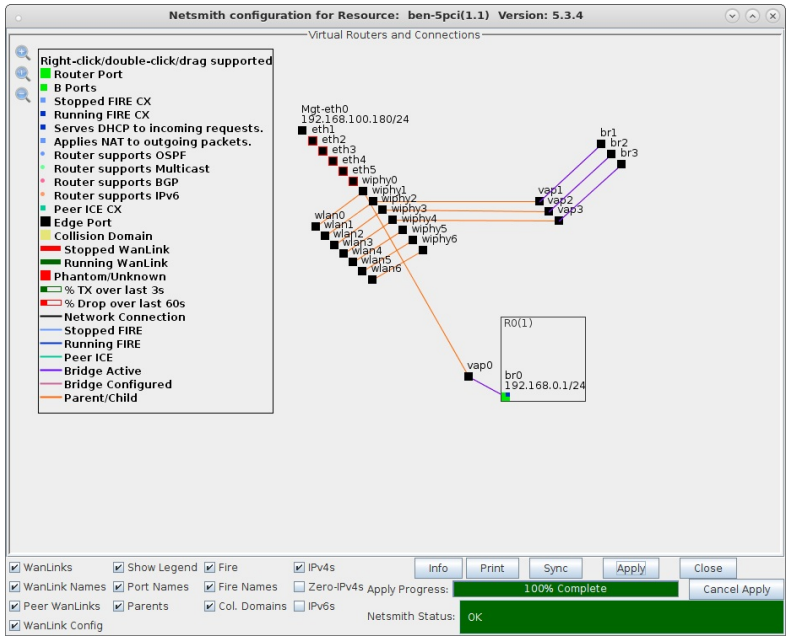
3. Create four bridge devices, one for each virtual access point.
   A. Go to the port manager tab, select Create, then select Bridge and enter Quantity 4 and a Bridge Name, then Apply to create the bridges.



   B. Modify each bridge device to add a vap. Netsmith will show a purple line when each vap has been added as bridge member.



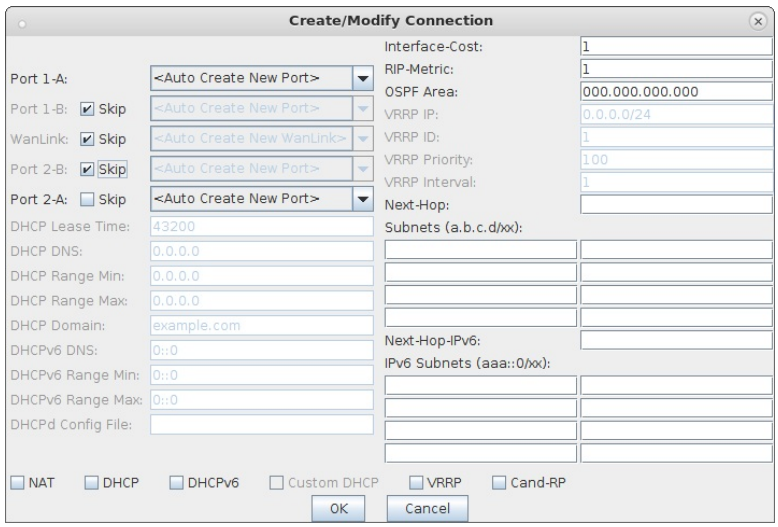
C. Add a single virtual router, drag br0 into the router, then give it an IP address and make it a DHCP server.



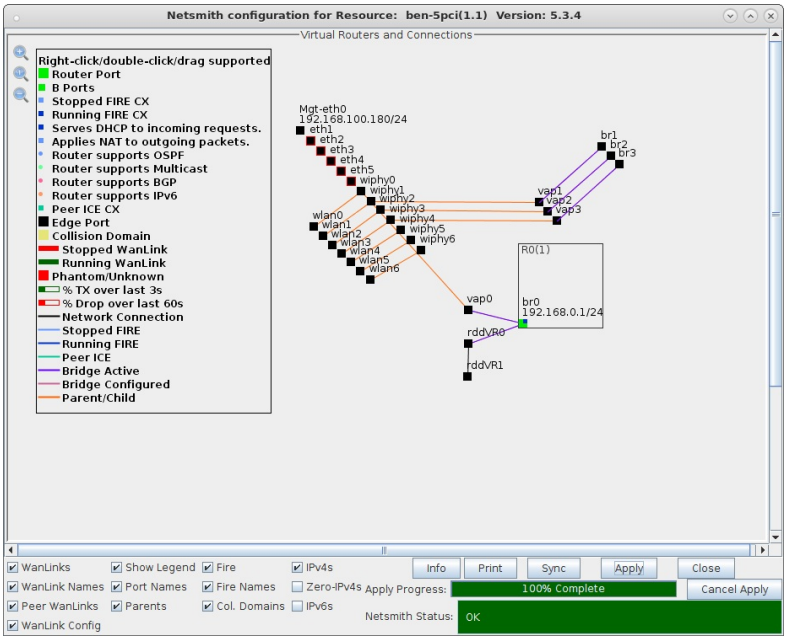
For more information see Virtual Router with DHCP Cookbook (skip the wanlink portion)

4. Each bridge will share a connection to a redirect device (rdd) pair so that FT messages can be sent and received.

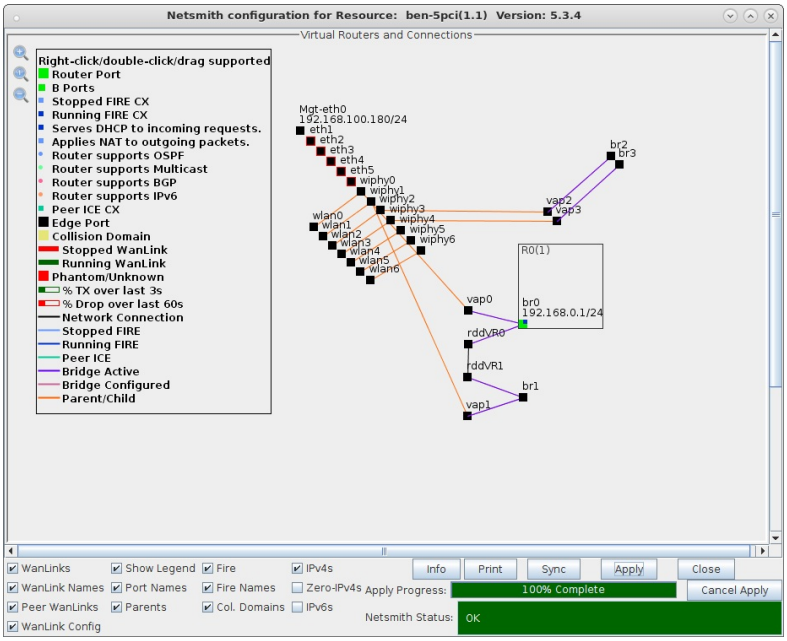
   A. In Netsmith, right-click in a free area and select New Connection to create an rdd pair. Select **Skip** for Port 1-B, WanLink and Port 2-B then select OK. Select Netsmith Apply after creating the new connection.



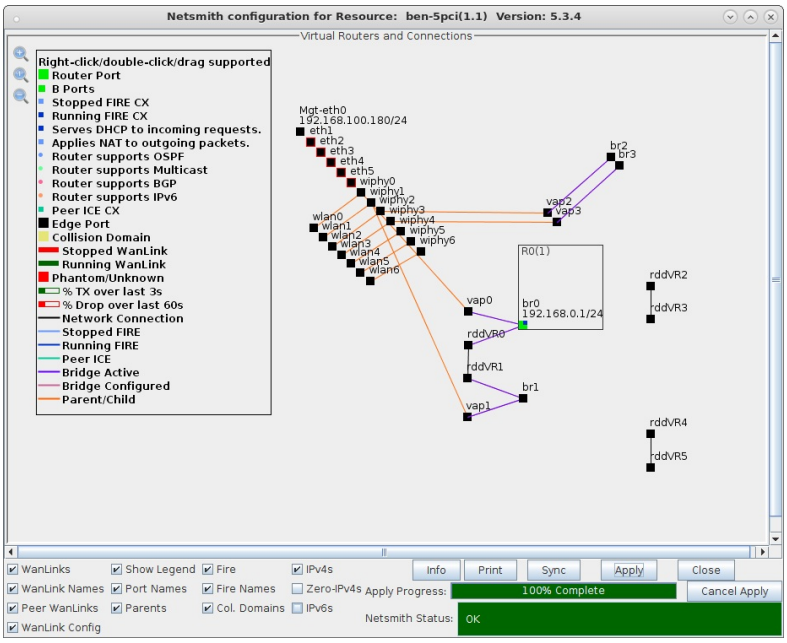
B. Right-click and select Modify Port br0, then add rddVR0 to br0. Your rddVRX numbering may differ depending on what other Netsmith objects are created.



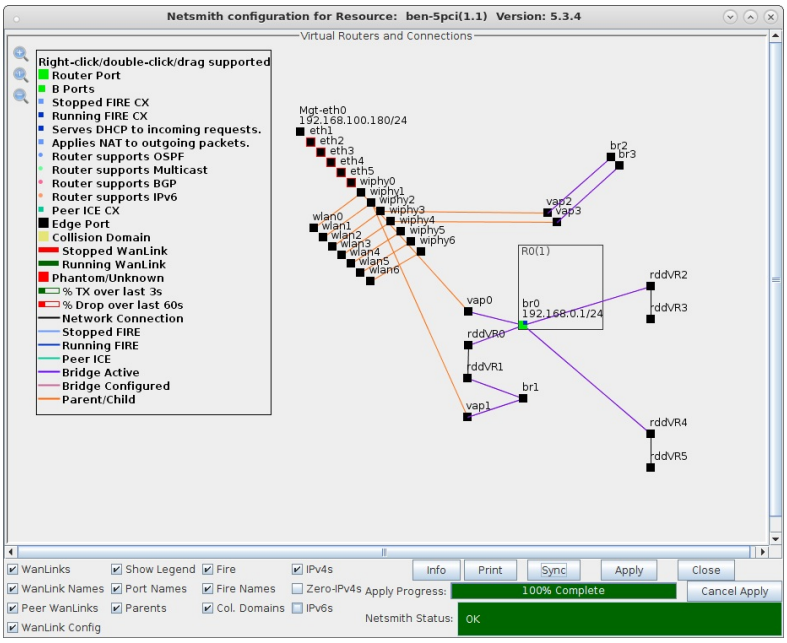
C. Right-click and select Modify Port br1, then add rddVR1 to br1. Your rddVRX numbering may differ depending on what other Netsmith objects are created.



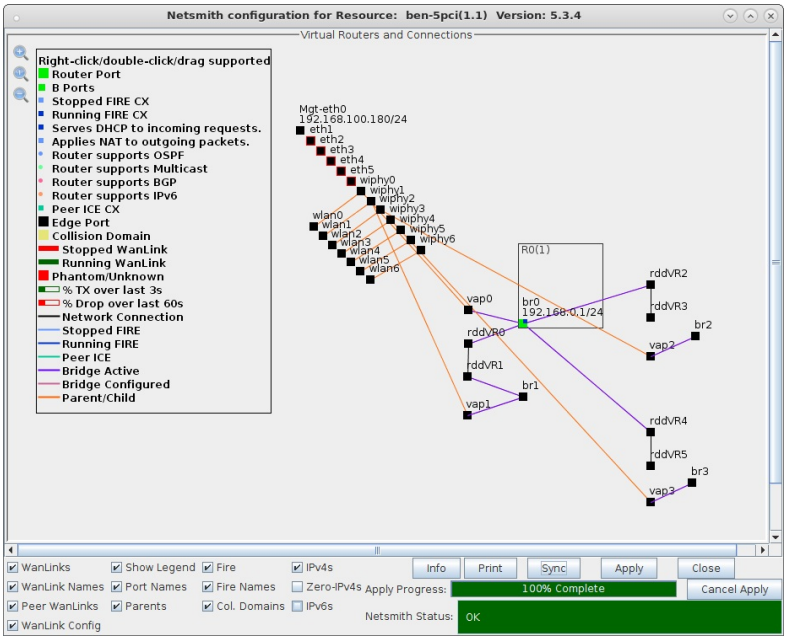
D. Create two more rdd pairs for bridges br2 and br3.



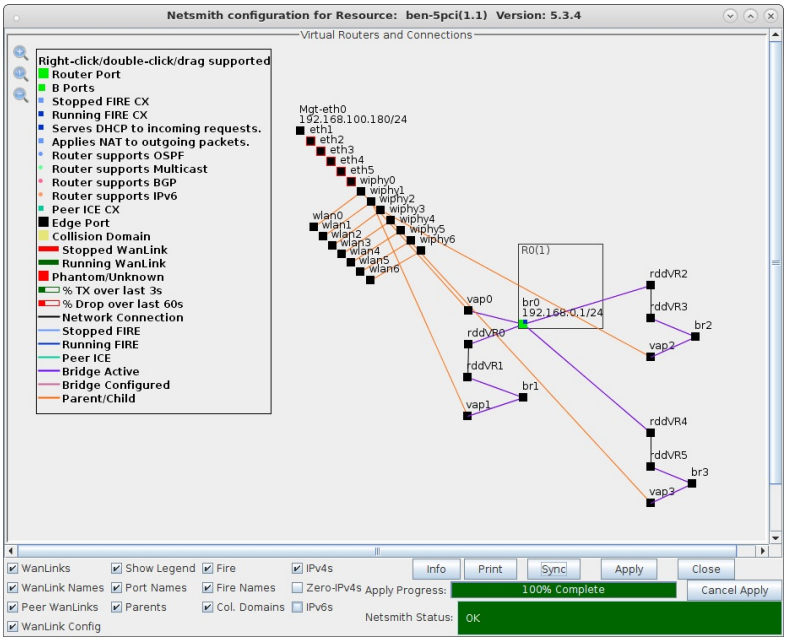
E. Add rddVR2 and rddVR3 to bridge br0.



F. The objects vap2/br2 and vap3/br3 can be moved so that their logical relationships can be visualized. Add rddVR3 to br2, then add rddVR5 to br3.

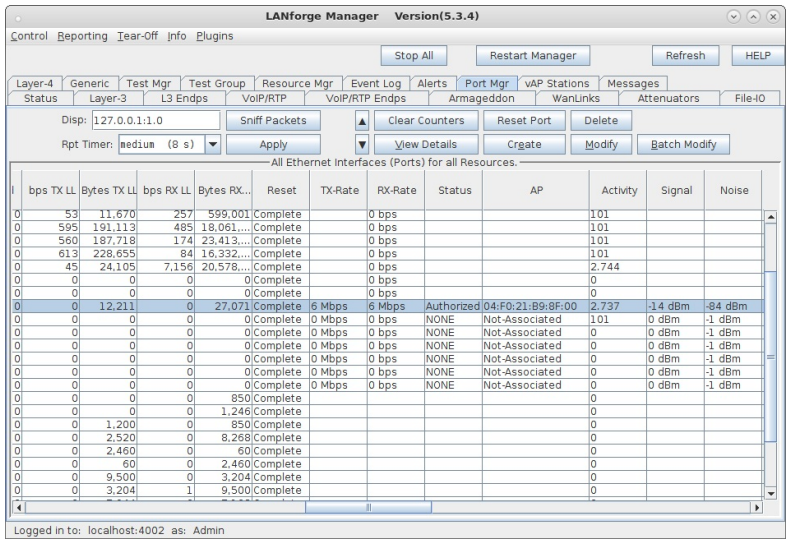


G. The final Netsmith display should show all four of the bridged virtual access points connected by a rdd pair, with br0 as the central bridge.



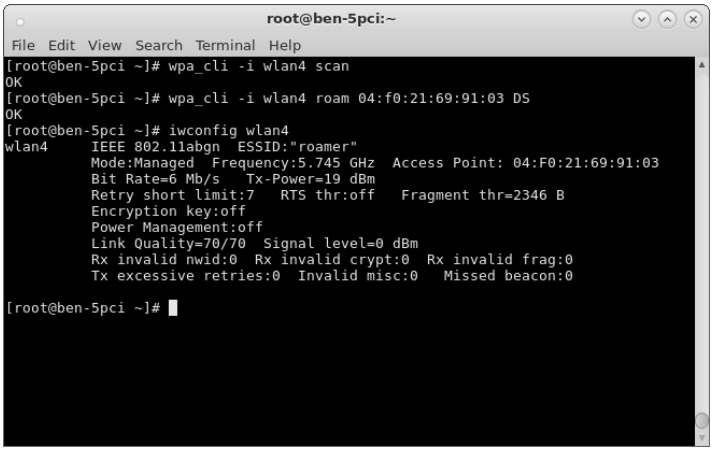
5. Connect LANforge clients and force them to roam from vap to vap. This can be accomplished with a wpa_cli command for one or two clients or the Mobility Plugin Script for many clients.

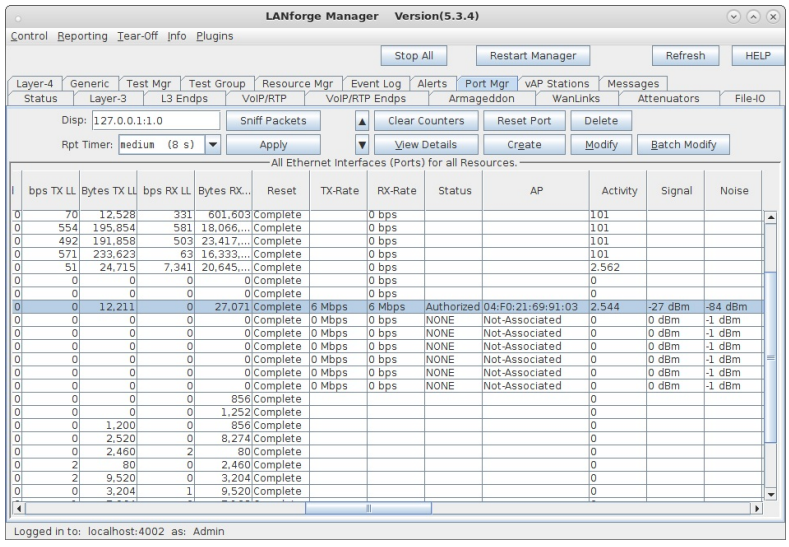
A. Client connected to vap0 **04:f0:21:b9:8f:00**.



l	bps TX LL	Bytes TX LL	bps RX LL	Bytes RX...	Reset	TX-Rate	RX-Rate	Status	AP	Activity	Signal	Noise
0	53	11,670	257	599,001	Complete		0 bps			101		
0	595	191,113	485	18,061,...	Complete		0 bps			101		
0	560	187,718	174	23,413,...	Complete		0 bps			101		
0	613	228,655	84	16,332,...	Complete		0 bps			101		
0	45	24,105	7,156	20,578,...	Complete		0 bps			2,744		
0	0	0	0	0	Complete		0 bps			0		
0	0	0	0	0	Complete		0 bps			0		
0	0	12,211	0	27,071	Complete	6 Mbps	6 Mbps	Authorized	04:F0:21:B9:8F:00	2,737	-14 dBm	-84 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	101	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	850	Complete					0		
0	0	0	0	1,246	Complete					0		
0	0	1,200	0	850	Complete					0		
0	0	2,520	0	8,268	Complete					0		
0	0	2,460	0	60	Complete					0		
0	0	60	0	2,460	Complete					0		
0	0	9,500	0	3,204	Complete					0		
0	0	3,204	1	9,500	Complete					0		

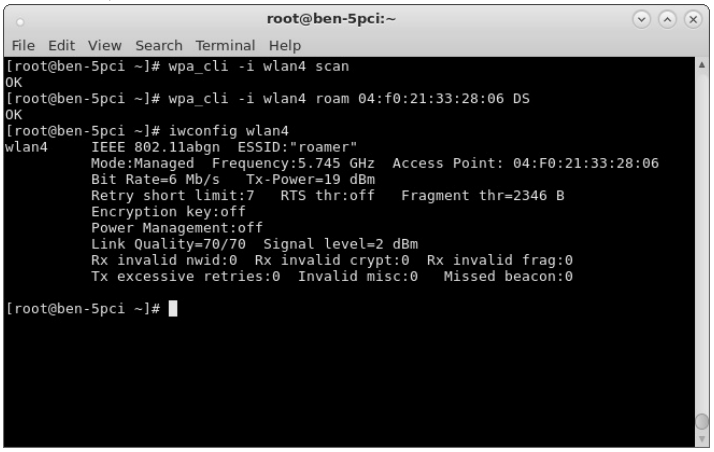
B. Client roams to vap1 **04:f0:21:69:91:03**.

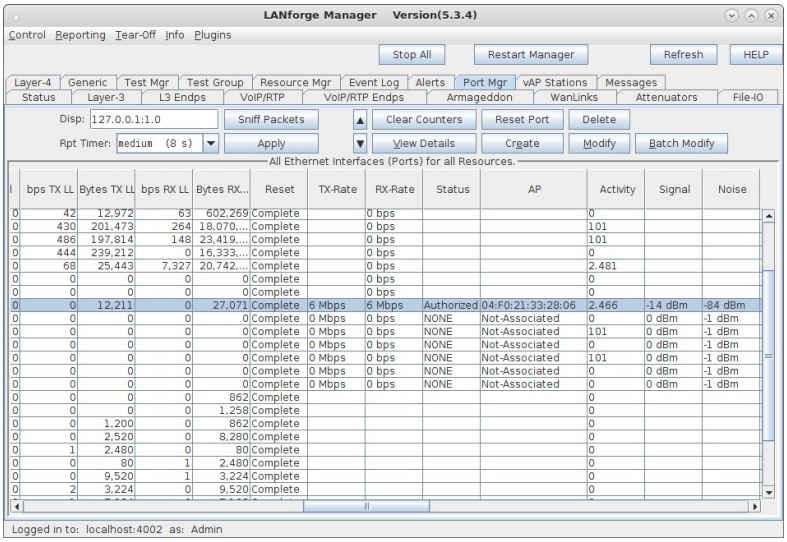




l	bps TX LL	Bytes TX LL	bps RX LL	Bytes RX...	Reset	TX-Rate	RX-Rate	Status	AP	Activity	Signal	Noise
0	70	12,528	331	601,603	Complete		0 bps			101		
0	554	195,854	581	18,066,...	Complete		0 bps			101		
0	492	191,858	503	23,417,...	Complete		0 bps			101		
0	571	233,623	63	16,333,...	Complete		0 bps			101		
0	51	24,715	7,341	20,645,...	Complete		0 bps			2,562		
0	0	0	0	0	Complete		0 bps			0		
0	0	0	0	0	Complete		0 bps			0		
0	0	12,211	0	27,071	Complete	6 Mbps	6 Mbps	Authorized	04:F0:21:69:91:03	2,544	-27 dBm	-84 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	856	Complete					0		
0	0	0	0	1,252	Complete					0		
0	0	1,200	0	856	Complete					0		
0	0	2,520	0	8,274	Complete					0		
0	0	2,460	2	80	Complete					0		
0	2	80	0	2,460	Complete					0		
0	2	9,520	0	3,204	Complete					0		
0	0	3,204	1	9,520	Complete					0		

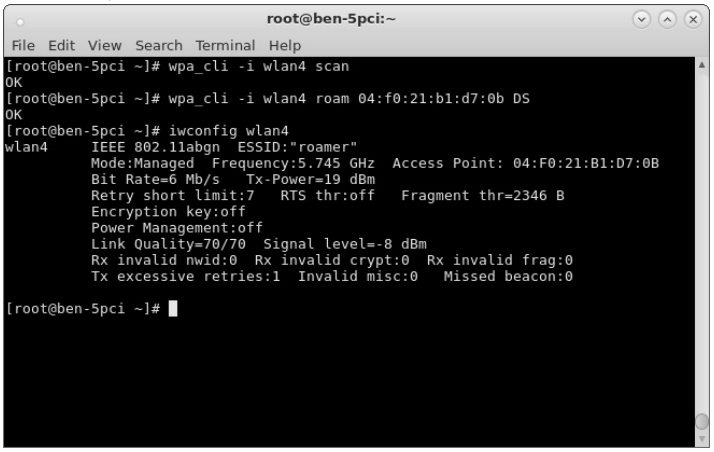
C. Client roams to vap2 **04:f0:21:33:28:06**.

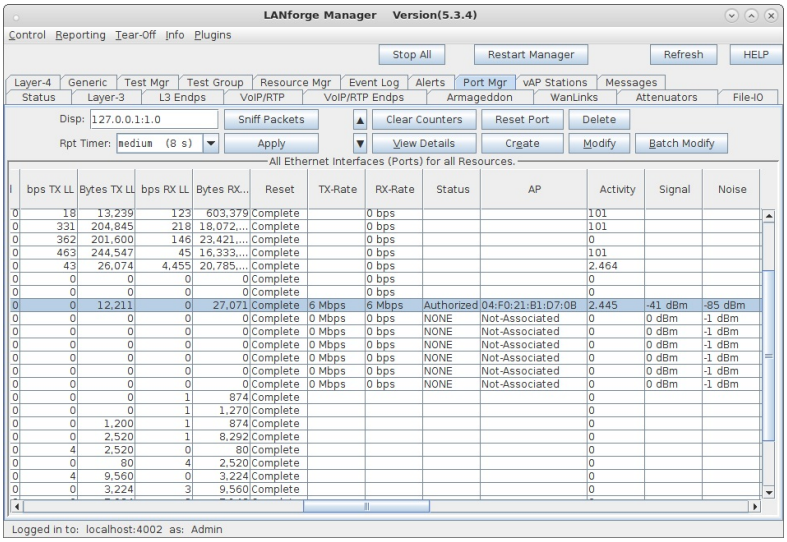




I	bps TX LL	Bytes TX LL	bps RX LL	Bytes RX...	Reset	TX-Rate	RX-Rate	Status	AP	Activity	Signal	Noise
0	42	12,972	63	602,269	Complete		0 bps			0		
0	430	201,473	264	18,070,...	Complete		0 bps			101		
0	486	197,814	148	23,419,...	Complete		0 bps			101		
0	444	239,212	0	16,333,...	Complete		0 bps			0		
0	68	25,443	7,327	20,742,...	Complete		0 bps			2.481		
0	0	0	0	0	Complete		0 bps			0		
0	0	0	0	0	Complete		0 bps			0		
0	0	12,211	0	27,071	Complete	6 Mbps	6 Mbps	Authorized	04:F0:21:33:28:06	2.466	-14 dBm	-84 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	101	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	101	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	862	Complete					0		
0	0	0	0	1,258	Complete					0		
0	0	1,200	0	862	Complete					0		
0	0	2,520	0	8,280	Complete					0		
0	1	2,480	0	80	Complete					0		
0	0	80	1	2,480	Complete					0		
0	0	9,520	1	3,224	Complete					0		
0	2	3,224	0	9,520	Complete					0		

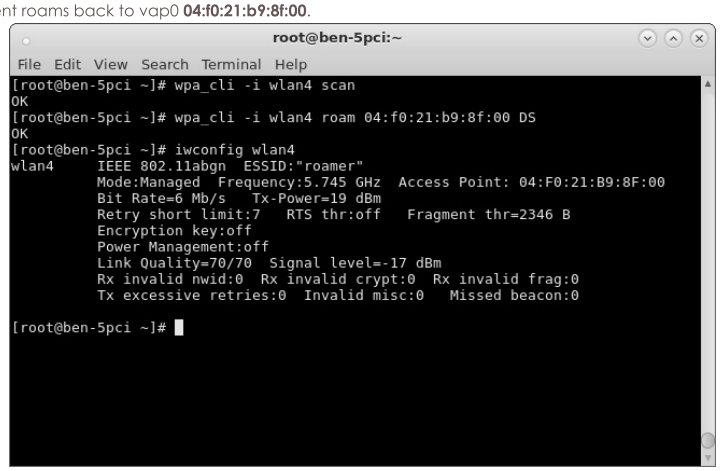
D. Client roams to vap3 **04:f0:21:b1:d7:0b**.

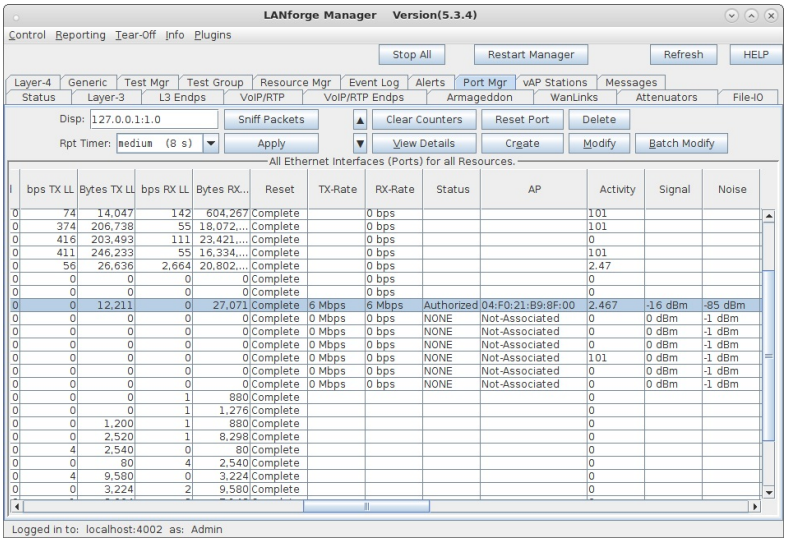




I	bps TX LL	Bytes TX LL	bps RX LL	Bytes RX...	Reset	TX-Rate	RX-Rate	Status	AP	Activity	Signal	Noise
0	18	13,239	123	603,379	Complete		0 bps			101		
0	331	204,845	218	18,072,...	Complete		0 bps			101		
0	362	201,600	146	23,421,...	Complete		0 bps			0		
0	463	244,547	45	16,333,...	Complete		0 bps			101		
0	43	26,074	4,455	20,785,...	Complete		0 bps			2.464		
0	0	0	0	0	Complete		0 bps			0		
0	0	0	0	0	Complete		0 bps			0		
0	0	12,211	0	27,071	Complete	6 Mbps	6 Mbps	Authorized	04:F0:21:B1:D7:0B	2.445	-41 dBm	-85 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	1	874	Complete					0		
0	0	0	1	1,270	Complete					0		
0	0	1,200	1	874	Complete					0		
0	0	2,520	1	8,292	Complete					0		
0	4	2,520	0	80	Complete					0		
0	0	80	4	2,520	Complete					0		
0	4	9,560	0	3,224	Complete					0		
0	0	3,224	3	9,560	Complete					0		

E. Client roams back to vap0 **04:f0:21:b9:8f:00**.





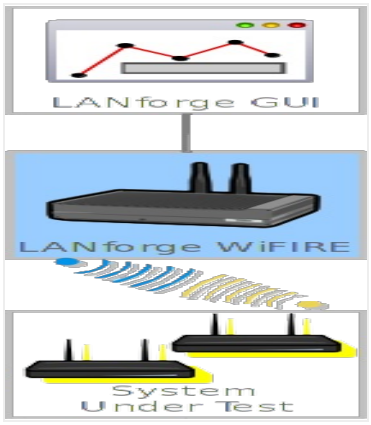
I	bps TX LL	Bytes TX LL	bps RX LL	Bytes RX...	Reset	TX-Rate	RX-Rate	Status	AP	Activity	Signal	Noise
0	74	14,047	142	604,267	Complete		0 bps			101		
0	374	206,738	55	18,072,...	Complete		0 bps			101		
0	416	203,493	111	23,421,...	Complete		0 bps			0		
0	411	246,233	55	16,334,...	Complete		0 bps			101		
0	56	26,636	2,664	20,802,...	Complete		0 bps			2.47		
0	0	0	0	0	Complete		0 bps			0		
0	0	0	0	0	Complete		0 bps			0		
0	0	12,211	0	27,071	Complete	6 Mbps	6 Mbps	Authorized	04:F0:21:B9:8F:00	2.467	-16 dBm	-85 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	101	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	0	0	Complete	0 Mbps	0 bps	NONE	Not-Associated	0	0 dBm	-1 dBm
0	0	0	1	880	Complete					0		
0	0	0	1	1,276	Complete					0		
0	0	1,200	1	880	Complete					0		
0	0	2,520	1	8,298	Complete					0		
0	4	2,540	0	80	Complete					0		
0	0	80	4	2,540	Complete					0		
0	4	9,580	0	3,224	Complete					0		
0	0	3,224	2	9,580	Complete					0		

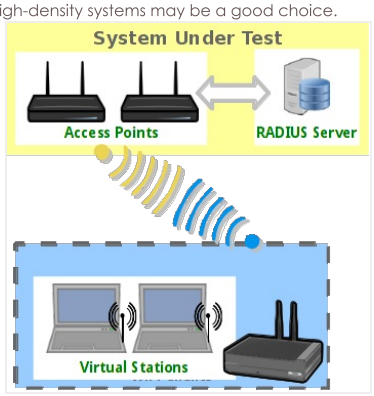
For more information see Two WiFi Access Point Network with 802.11r

LANforge WiFi testing Roaming and HotSpot 2.0

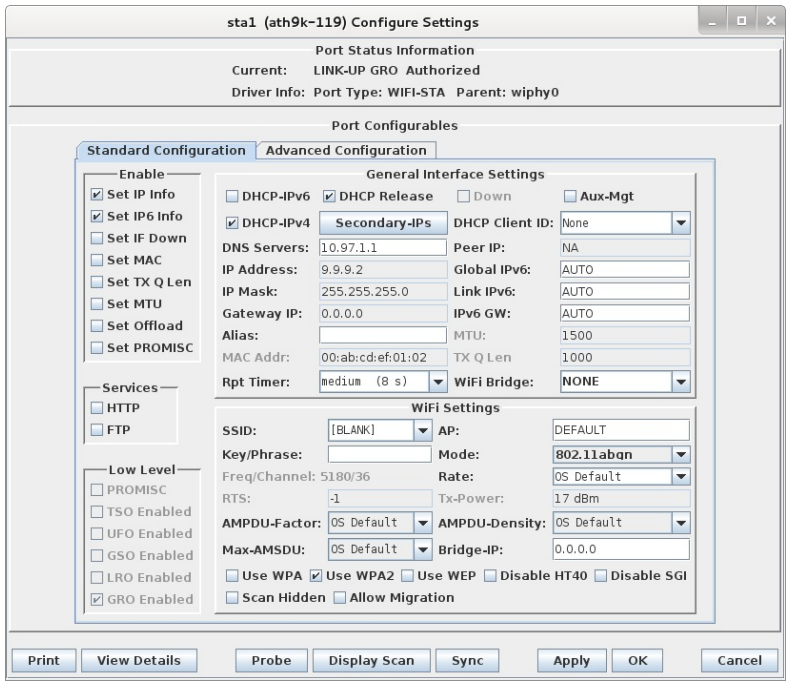
Goal: Use automated script to reconnect stations to an AP and report results.

Requires LANforge 5.2.11 or later. Configure Stations to use HotSpot 2.0 (802.1x, 802.1u, etc) and associate them with a HotSpot 2.0 AP. Use the 'WiFi Mobility' LANforge-GUI Plugin to automate re-connecting to the AP and querying ANQP. The plugin will create graphs and other reports that can be saved to HTML. This example uses a LANforge CT520 system but the procedure should work on all CT521, CT522, CT523, CT525 and similar systems. The AP in this test is another LANforge machine, but it could be any AP that supports HotSpot 2.0. A similar test could roam between multiple APs. If the APs are all on the same channel there are no restrictions, but if the APs are on different channels, then only a single station can be configured per LANforge radio. In that case, multiple 3-radio CT523 or other high-density systems may be a good choice.

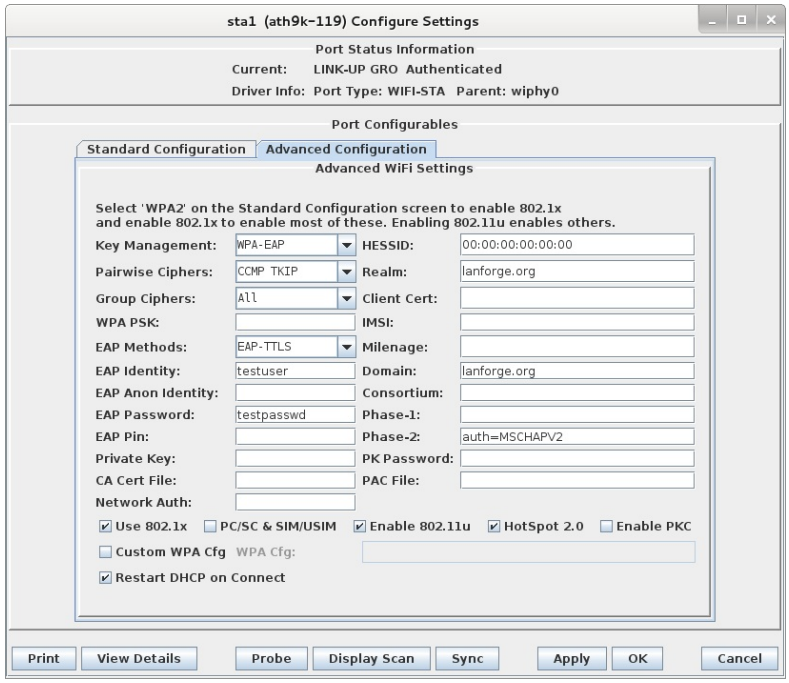




1. Configure stations to connect to an AP configured for HotSpot 2.0.
   A. Go to the Port Manager tab, select wiphy0 on proper resource, click Create, fill out appropriate information and create desired number of Station interfaces.
   B. The new stations should appear in the Port-Mgr table. Double-click to modify one of them. Configure IP Address information, set SSID to [BLANK] and select WPA2:



C. Select the **Advanced Configuration** tab in the Port-Modify window and configure the Key Management, EAP Method, passwords, select **Use 802.1x**, **Enable 802.11u** and **HotSpot 2.0**. If you want to report on DHCP negotiation times, be sure to select the **Restart DHCP on Connect** checkbox. If you want to get packet-drop statistics during roam, Un-select **Restart DHCP on Connect**:

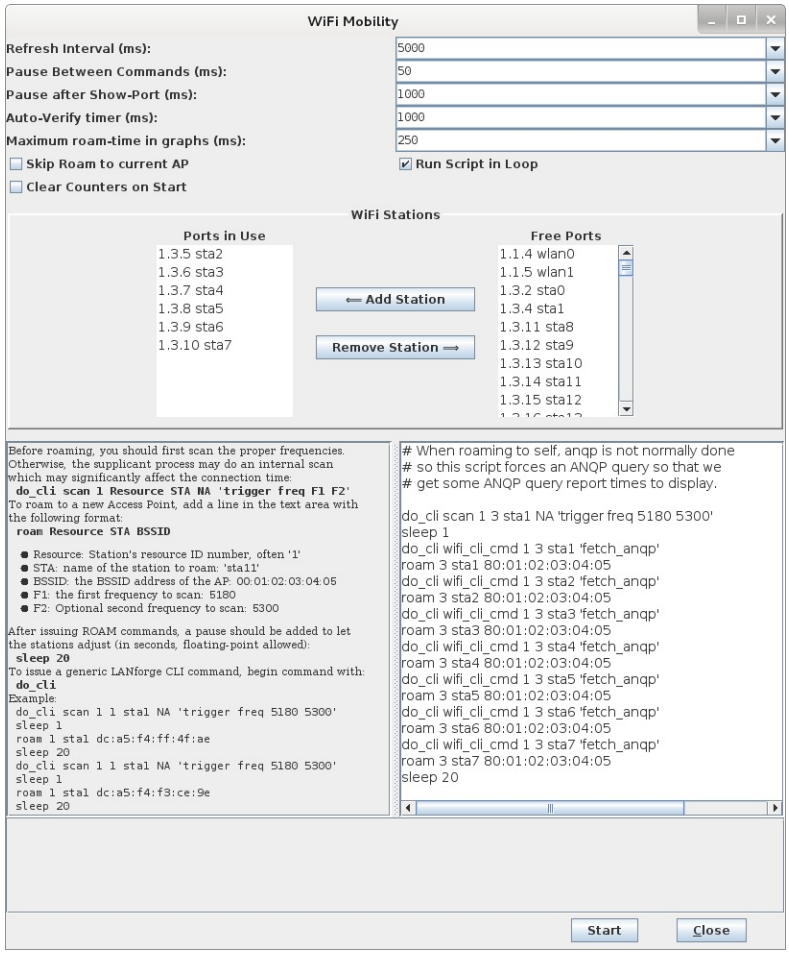


D. Once the single station is connecting properly, use Batch-Modify to configure the rest of the stations to match the first.

For more information see LANforge User's Guide: Ports (Interfaces) , WiFi Station Cookbook , WiFi HotSPot 2.0 Cookbook

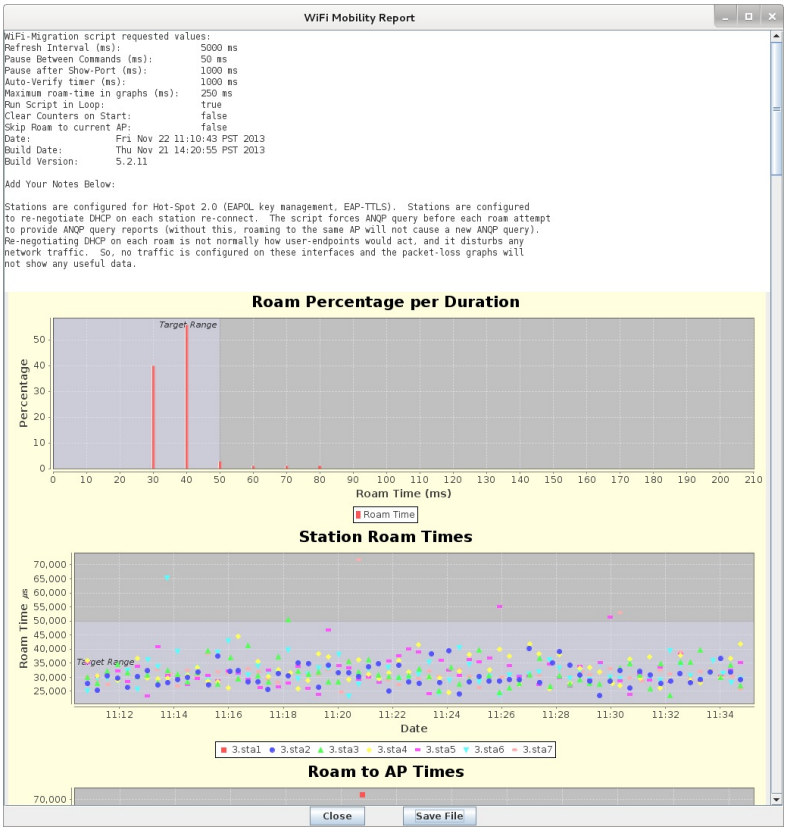
2. Start the WiFi Migration script.

A. Go to the Port Manager tab, select the stations you wish to roam, right-click and choose the **WiFi Mobility** menu option.

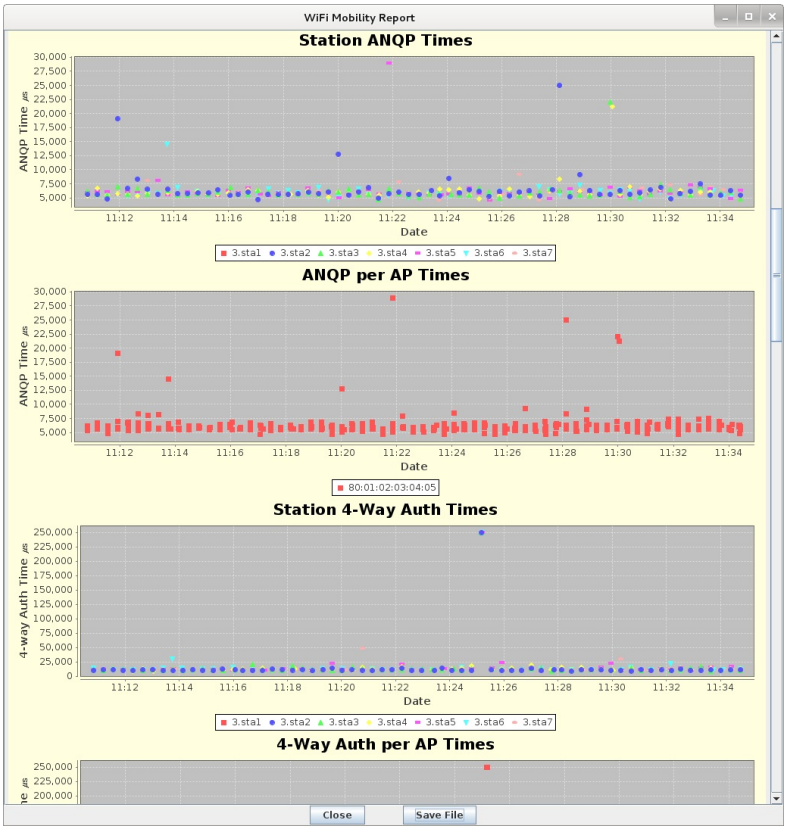


B. The options at the top default to common values and most do not need to be changed. For this example, you must unselect **Skip Roam to current AP** because the script is requesting exactly that. The ports will be automatically configured based on the selection on the Port Manager tab, and may be adjusted before starting the script. The **Ports in Use** should normally include all stations used in the script. The configuration requiring the most work from the user is the roaming script itself. There is a help section on the left, and a script-entry field on the right. Once the script is written, it should be saved in a text file on the user's PC so that it can easily be pasted into future WiFi Mobility scripts. Some key points are that you must scan about 1 second before roaming or the roam logic in the supplicant process will either fail or do its own roaming. Either way, the results may be worse than if you do the roam properly in the script. It can take a bit of time for LANforge to get all of the data it needs to report on the roam attempt, so it is suggested that stations not roam more often that about once every 10-20 seconds. If reporting is less important, then the stations can roam more often.

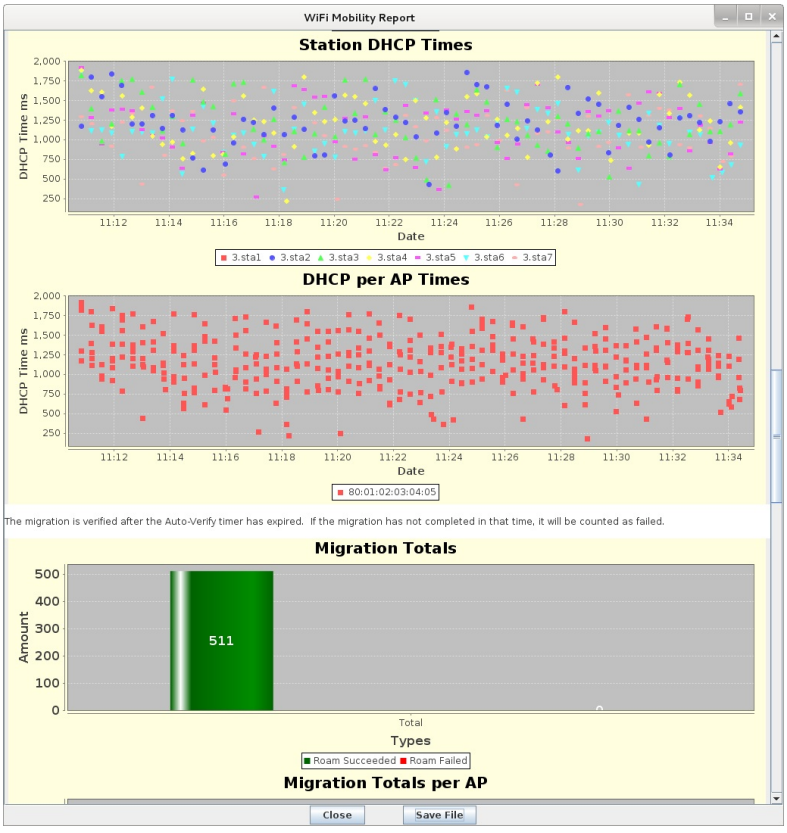
C. Once the script is properly configured, click Start to start the roaming. A window will pop up that has live-updating graphs of various reports. A text log is at the bottom for more detailed analysis, and the whole thing can be saved as HTML. The graphs can be scaled and configured through right-click menus if desired. It will take 1-2 complete roam attempts before the graphs are able to show any useful information.



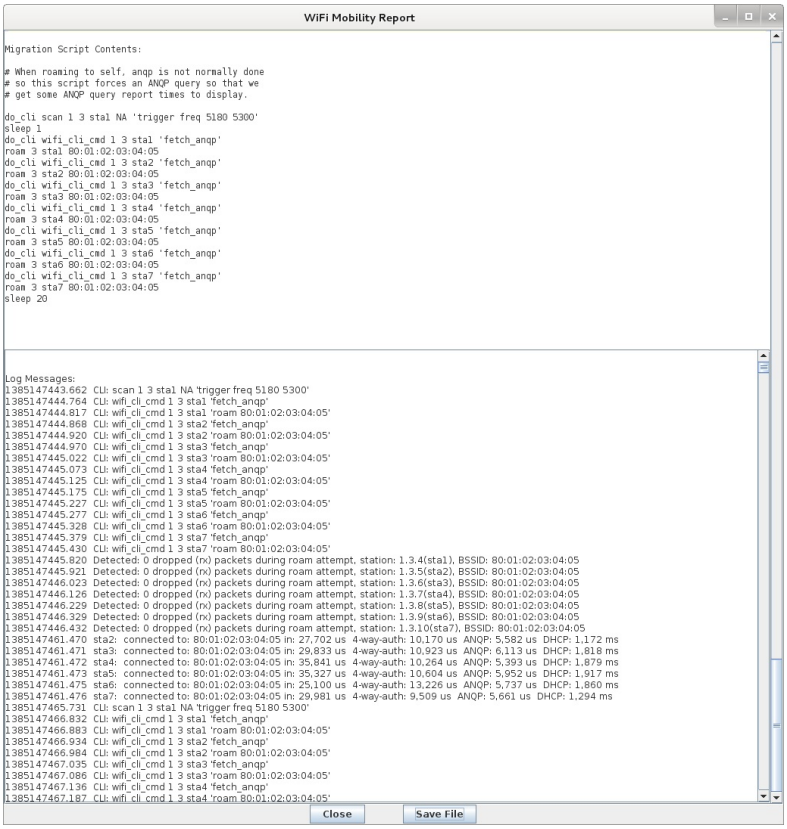
D. ANQP and 4-Way Authentication graphs.



E. DHCP Negotiation and Migration Totals graphs.



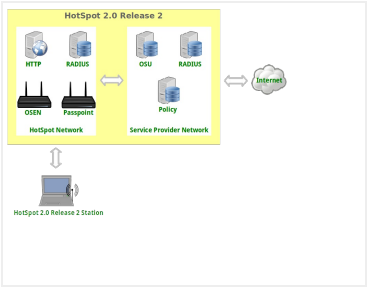
F. Text log with timestamps. Can be coorelated with wpa_supplicant logs and other log files to debug specific roam attempts.



For more information see Complete report for this test case

LANforge WiFi testing HotSpot 2.0 Release 2

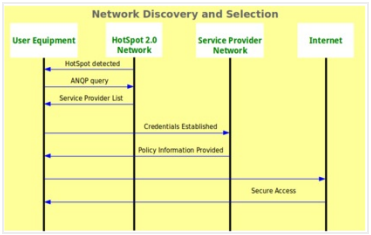
Goal: Setup HotSpot 2.0 Release 2 Example



Requires LANforge 5.4.2 or later on Fedora 20 or later (this example is using Fedora 27). VRF must be enabled (it is enabled by default). One LANforge system will be used as the AP side, and a second LANforge machine will be the WiFi station.

- Run LANforge install script to begin setup of HotSpot 2.0 R2 related servers and certificates.
- Configure the **O**SU **S**erver-only authenticated layer-2 **E**ncryption **N**etwork (OSEN) AP and Passpoint AP.
- Initiate **O**nline **S**ign-**U**p (OSU) procedure, select a provider and obtain an IP address from the Passpoint AP.
- Send traffic through the Service Provider Network.

For

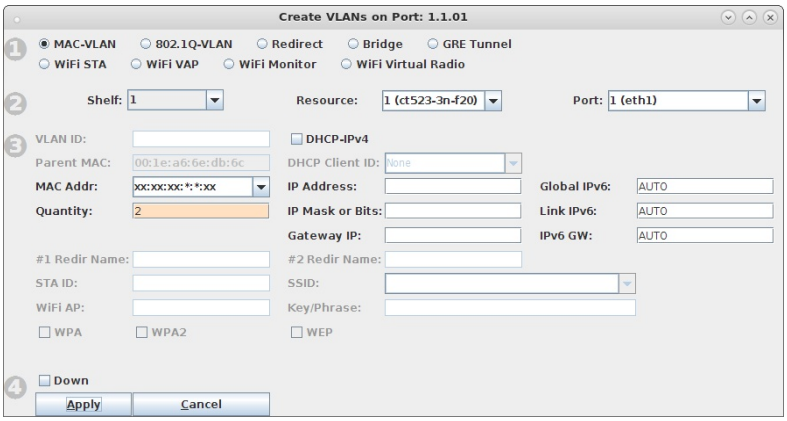


more information see:
WiFi Alliance Passpoint Release 2 Deployment Guidelines
https://www.wi-fi.org/file/passpoint-release-2-deployment-guidelines

---

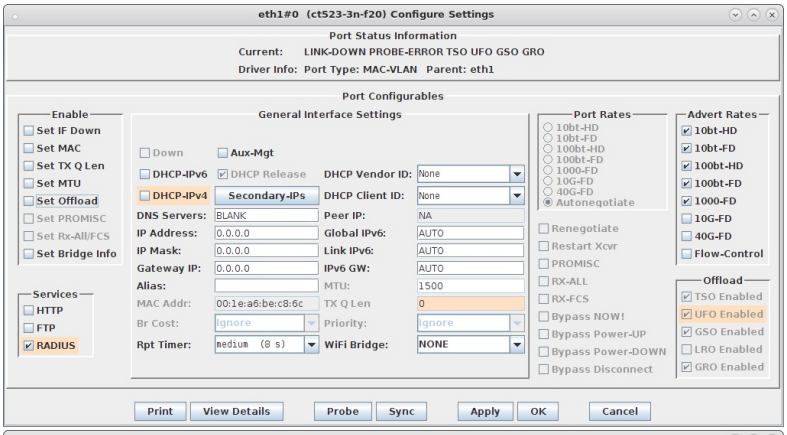
1. Run LANforge installation script to begin hostapd RADIUS, certificates and HotSpot 2.0 setup, as root user:
   ```
   cd /home/lanforge
   ./lf_kinstall.pl --do_radius --do_hs20 --force_new_certs
   ```
2. Make two copies of the `ca.pem` certificate to different directories:
   ```
   cp /home/lanforge/hs20/ca/ca.pem /home/lanforge/osu-ca.pem
   cp /home/lanforge/hs20/ca/ca.pem /home/lanforge/ota-ca.pem
   #On station machine, if different from AP machine
   cp /home/lanforge/hs20/ca/ca.pem /home/lanforge/wifi/osu_wlan2/osu-ca.pem
   ```
   ota-ca.pem is used by the client for Over-The-Air authentication to the OSEN AP
   osu-ca.pem is used by the client for the Online-Sign-Up server authentication before connecting to the Passpoint AP

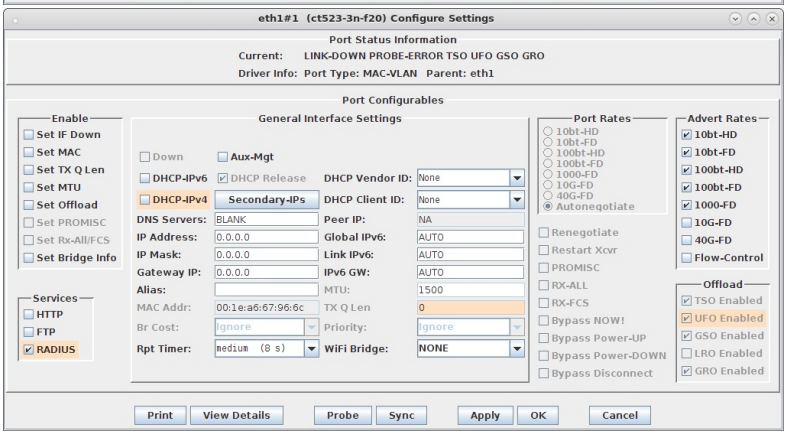
   Copy the ca.pem from the LANforge AP system to the LANforge Station system. And, if you are using a third-party client, then you will need to somehow install the ca.pem on it.
3. Create two MAC-VLANs for two hostapd RADIUS server instances.
   A. Go to the Port Manager tab, select eth1, select Create, select MAC-VLAN, quantity 2 then Apply.



B. Double-click each new MAC-VLAN interface in the Port-Mgr tab to modify. Select the RADIUS checkbox which will allow a hostapd based RADIUS server on the interfaces using the config files: `/home/lanforge/wifi/hostapd_eth1#0.conf` and `/home/lanforge/wifi/hostapd_eth1#1.conf`

In an all-in-one example, the hostapd RADIUS servers will be referenced by localhost and each MAC-VLAN interface will not need an IP address assigned. If the hostapd RADIUS servers were on different systems or networks, or need to be accessed from outside the LANforge system, the appropriate IP address would be assigned here.





C. Create config file. You will need to change the server_id to match your hostname. `/home/lanforge/wifi/hostapd_eth1#0.conf` for the hostapd RADIUS server on eth1#0.
NOTE: The eap_user_file, eap_sim_db and radius_server_auth_port are unique for each RADIUS server.

```
interface=eth1#0
driver=wired
logger_syslog=-1
logger_syslog_level=2
logger_stdout=-1
logger_stdout_level=2
dump_file=/home/lanforge/wifi/hostapd_eth1#0.dump
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0
ieee8021x=1
eapol_key_index_workaround=0
eap_server=1
eap_user_file=/home/lanforge/hs20/AS/hostapd-osen.eap_user
server_id=ct523-3n-f20
eap_sim_db=unix:/tmp/hlr_auc_gw.sock
radius_server_auth_port=1820
radius_server_clients=/home/lanforge/hs20/AS/hostap.radius_clients

ca_cert=/home/lanforge/hs20/ca/ca.pem
server_cert=/home/lanforge/hs20/ca/server.pem
private_key=/home/lanforge/hs20/ca/server.key
private_key_passwd=lanforge

ocsp_stapling_response=/home/lanforge/hs20/ca/ocsp-server-cache.der
```

D. Create config file. You will need to change the server_id to match your hostname.
`/home/lanforge/wifi/hostapd_eth1#1.conf` for the hostapd RADIUS server on eth1#1.
NOTE: The eap_user_file, eap_sim_db and radius_server_auth_port are unique for each RADIUS server.

```
interface=eth1#1
driver=wired
logger_syslog=-1
logger_syslog_level=2
logger_stdout=-1
logger_stdout_level=2
dump_file=/home/lanforge/wifi/hostapd_eth1#1.dump
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0
ieee8021x=1
eapol_key_index_workaround=0
eap_server=1
eap_user_file=sqlite:/home/lanforge/hs20/AS/DB/eap_user.db
server_id=ct523-3n-f20
eap_sim_db=unix:/tmp/hlr_auc_gw.sock db=/home/lanforge/hs20/AS/DB/eap_sim.db
radius_server_auth_port=1821
radius_server_clients=/home/lanforge/hs20/AS/hostap.radius_clients

ca_cert=/home/lanforge/hs20/ca/ca.pem
server_cert=/home/lanforge/hs20/ca/server.pem
private_key=/home/lanforge/hs20/ca/server.key
private_key_passwd=lanforge

ocsp_stapling_response=/home/lanforge/hs20/ca/ocsp-server-cache.der
```

E. Start the hlr_auc_gw tool:

```
cd /home/lanforge
. lanforge.profile
hlr_auc_gw -m /etc/hlr_auc_gw.milenage_db > /tmp/hlr_auc_gw.log &
```

NOTE: If the hlr_auc_gw does not start, you may have to remove the file /tmp/hlr_auc_gw.sock first.

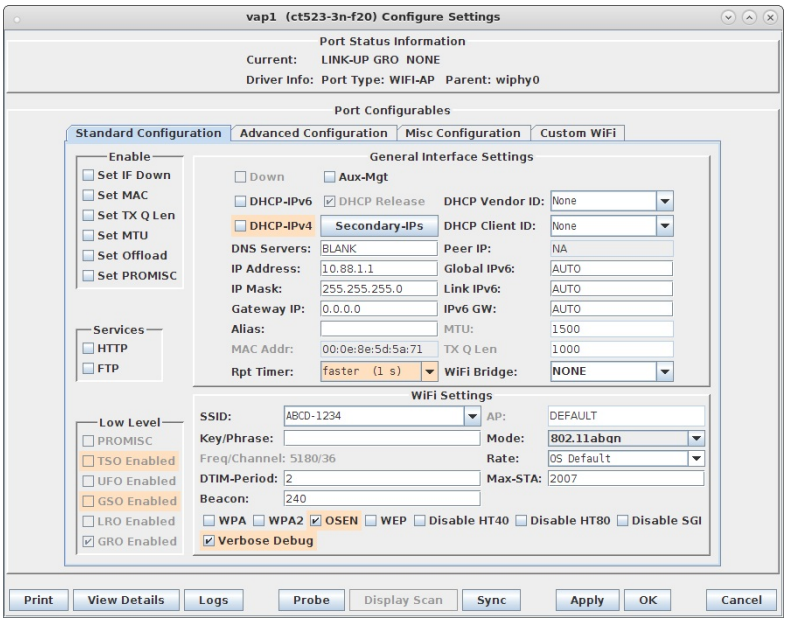
F. Reset the MAC-VLAN interfaces on the Port Mgr tab so that the new hostapd RADIUS servers are started. Check that they are running with the command:

```
ps auxwww |grep hostapd_eth
```

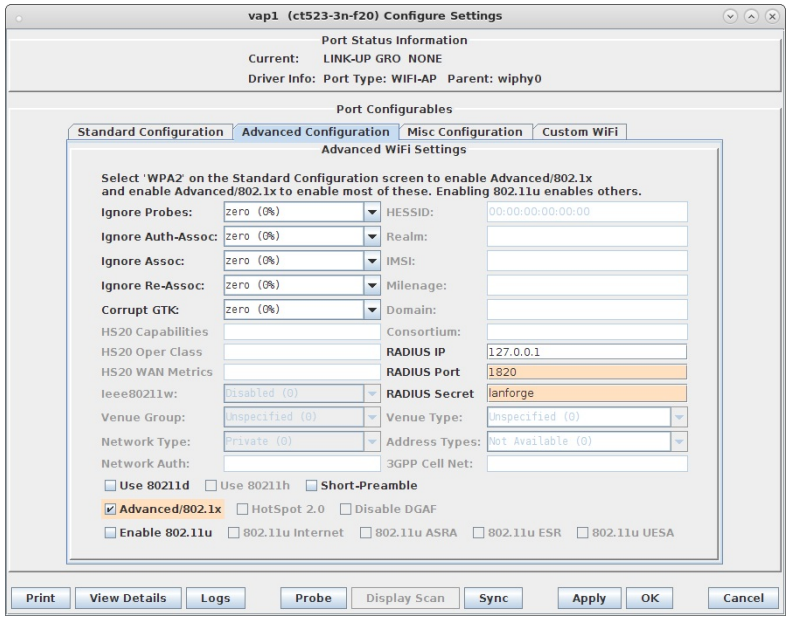
If they are not running, check the log files for problems:

```
cat /home/lanforge/wifi/hostapd_log_eth1#0.txt
cat /home/lanforge/wifi/hostapd_log_eth1#1.txt
```

4. Create two VAPs for the HotSpot 2.0 Release 2 Network. Skip this section if you are using third-party APs in this test.
   A. Go to the Port Mgr tab and create one VAP on wiphy0 and one VAP on wiphy1.
   B. Modify the first VAP on wiphy0 to be the **OSEN** AP. Configure IP Address and SSID.

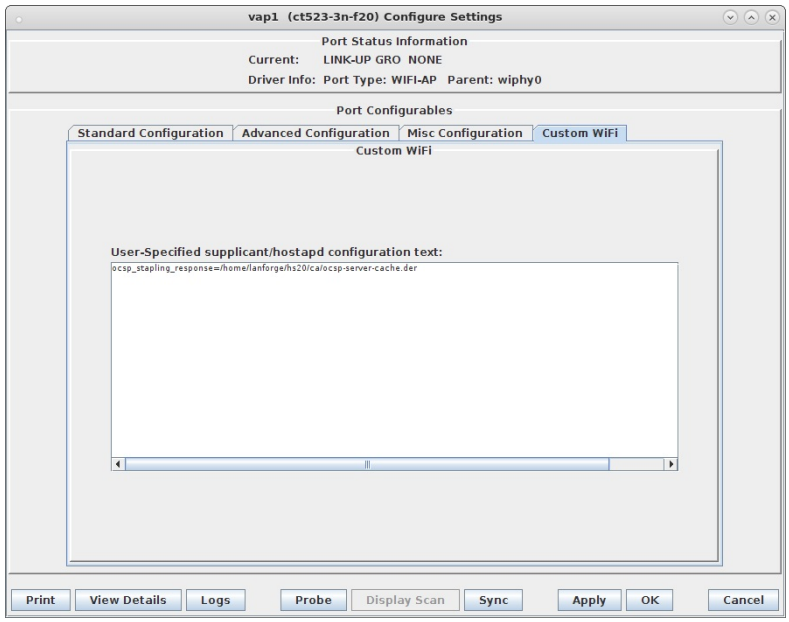


C. Select the **Advanced Configuration** tab in the Port-Modify window to configure 802.1x and RADIUS server information.

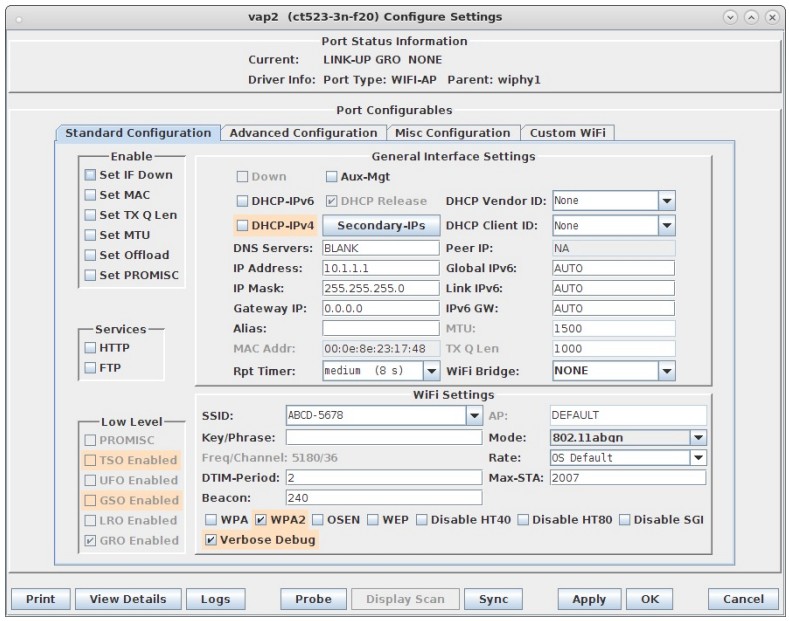


D. Select the **Custom WiFi** tab to add the following lines for HotSpot 2.0 Release 2.

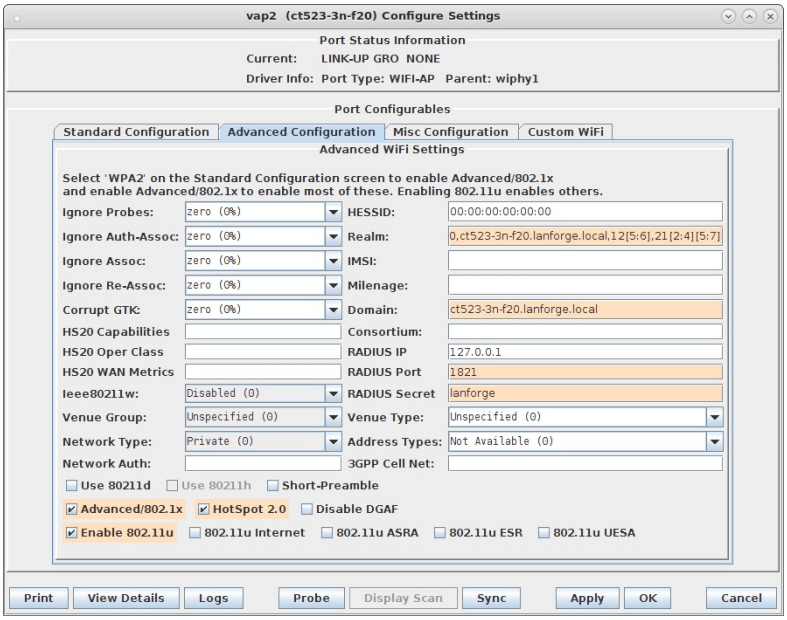
```
ocsp_stapling_response=/home/lanforge/hs20/ca/ocsp-server-cache.der
```



E. Modify the second VAP on wiphy1 to be the **Passpoint** AP. Configure IP Address and SSID.

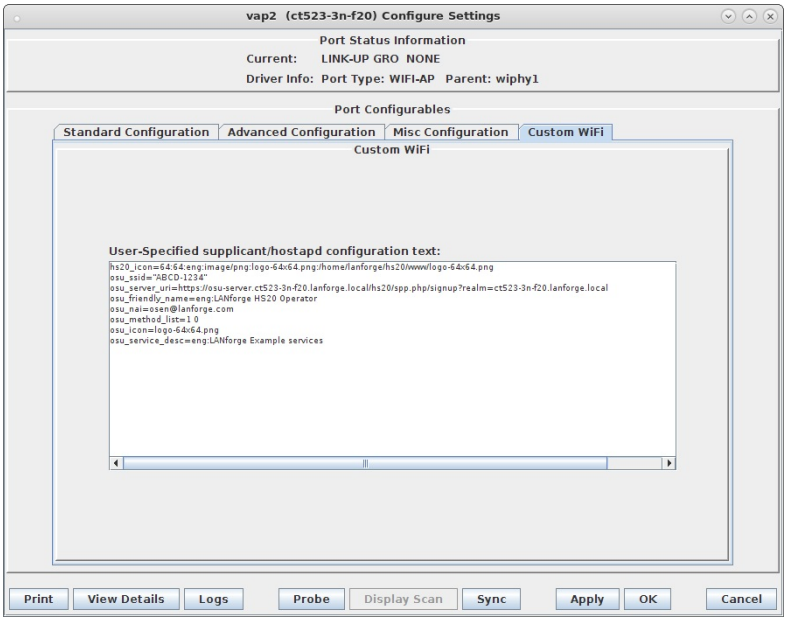


F. Select the **Advanced Configuration** tab in the Port-Modify window to configure 802.1x, 802.1u, HotSpot 2.0, RADIUS server and other information.



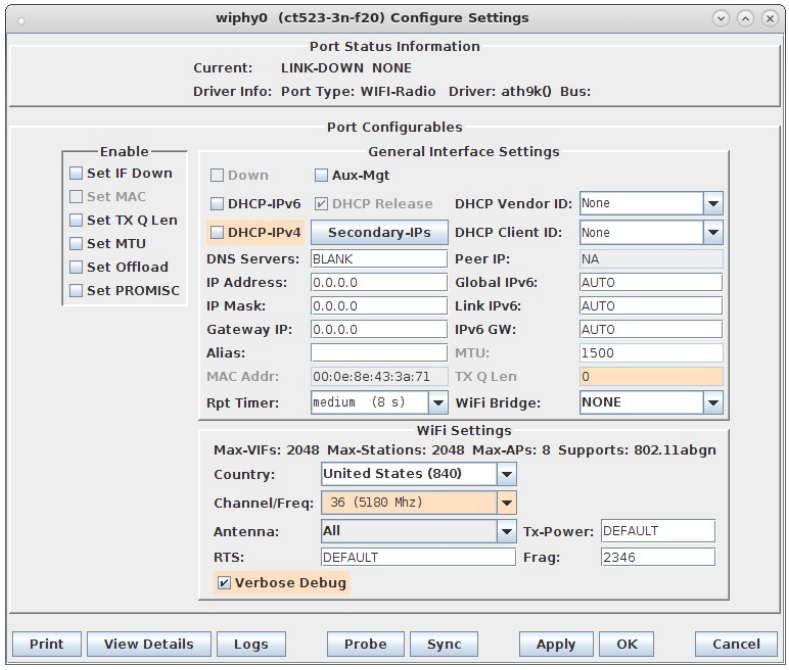
G. Select the **Custom WiFi** tab to add the following lines for HotSpot 2.0 Release 2. The hostname in the URL will need to match your actual hostname.

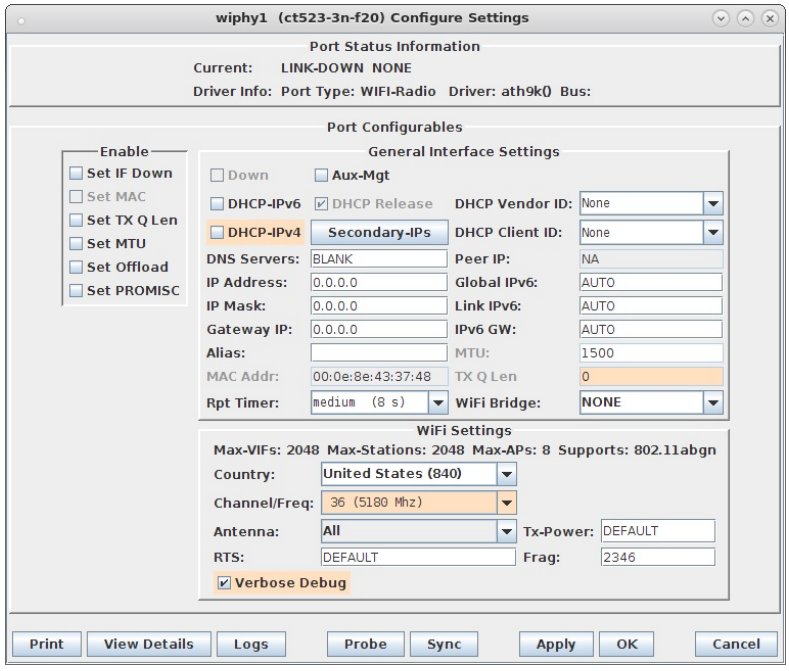
```
hs20_icon=64:64:eng:image/png:logo-64x64.png:/home/lanforge/hs20/www/logo-64x64.png
osu_ssid="ABCD-1234"
osu_server_uri=https://osu-server.ct523-3n-f20.lanforge.local/hs20/spp.php/signup?realm=ct523-3n-f20.lanforge.local
osu_friendly_name=eng:LANforge HS20 Operator
osu_nai=osen@lanforge.com
osu_method_list=1 0
osu_icon=logo-64x64.png
osu_service_desc=eng:LANforge Example services
```



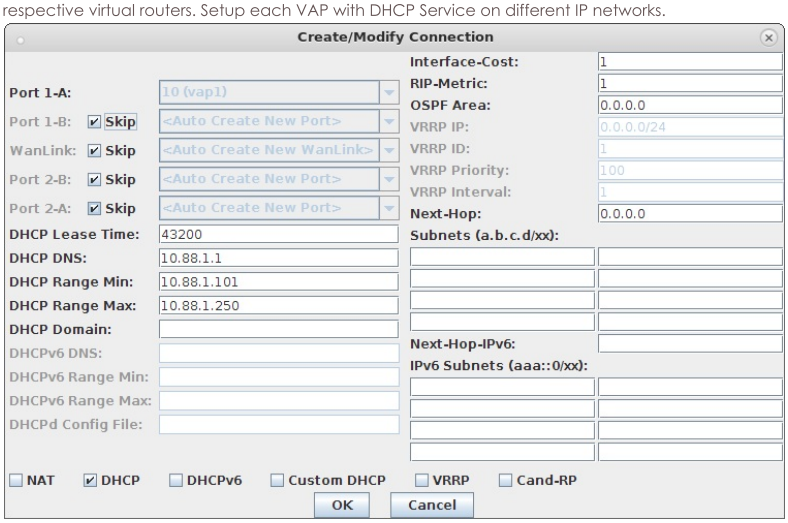
H. Example hostapd config file for OSEN AP
Example hostapd config file for WPA2 + ANQP AP. These were re-created after the fact, its possible that there is a typo, and the hostname differs from the instructions in this example.

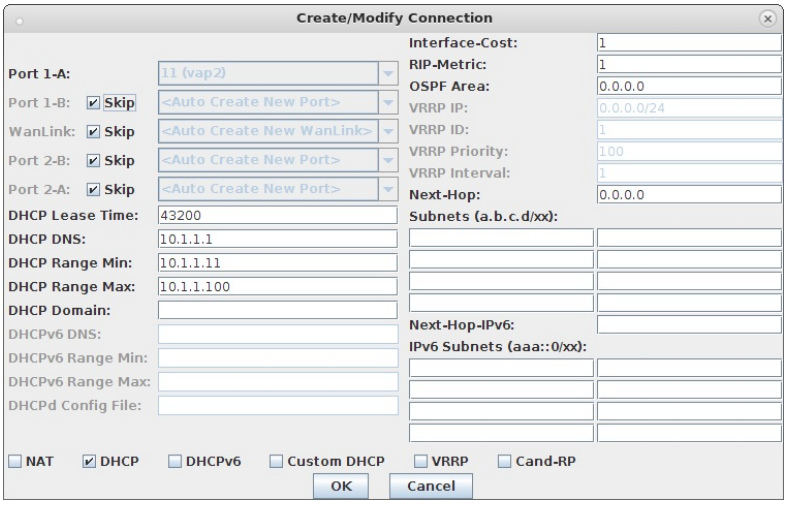
l. Modify wiphy0 and wiphy1 to be on the same channel and select OK. This should only be needed if you are trying to connect multiple LANforge virtual stations to the APs. If using a single station or a third-party device, the channels should not matter.





J. In Netsmith, create a Virtual router called OSEN for vap1 and PASSPOINT for vap2, place vap in their respective virtual routers. Setup each VAP with DHCP Service on different IP networks.





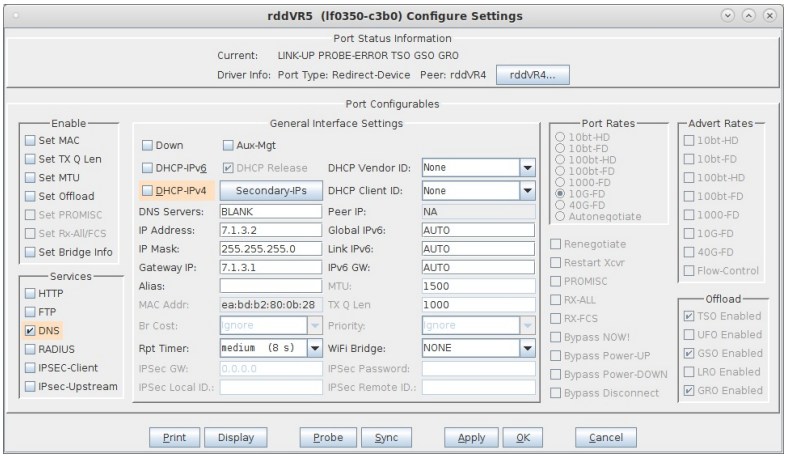
K. Check that the VAP hostapd processes are running with the command:

```
ps auxwww |grep hostapd_vap
```

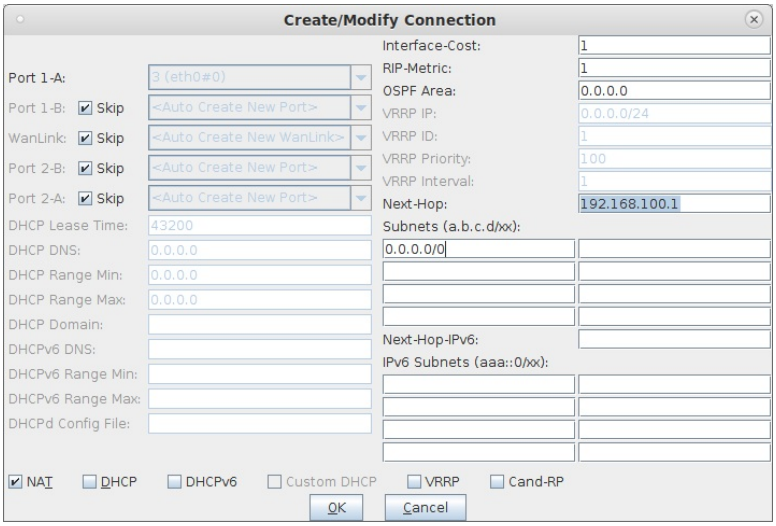
If they are not running, check the log files for problems:

```
tail -f /home/lanforge/wifi/hostapd_log_vap1.txt
tail -f /home/lanforge/wifi/hostapd_log_vap2.txt
```

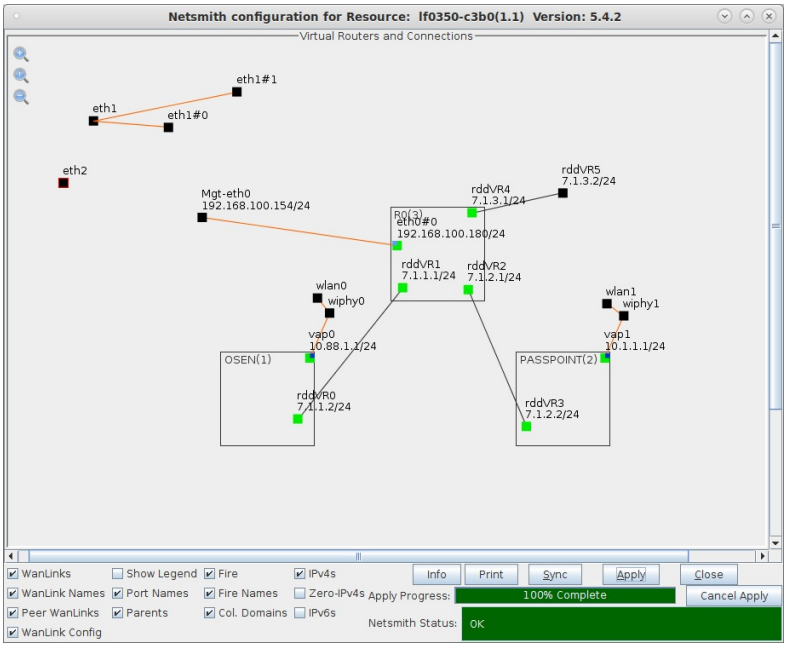
L. Create another virtual router to tie the OSEN and AP virtual routers together. Connect them with Netsmith connections (skipping WanLinks unless you are sure you want them.) Create another virtual port connection to run the OCSP responder and the local DNS server. Select the 'DNS' service. Make sure the default gateway points back towards rddVR4 in the virtual router. In this example, the OCSP responder interface is called rddVR5



M. Create a MAC-VLAN on the management port (or use some other port that can reach the internet). Drag this into the top virtual router, and configure Netsmith to use it for the default gateway by setting up the Next Hop and the 0.0.0.0/0 destination. Select NAT as well.



N. When the configuration is completed, Netsmith will look something like this. Apply it, wait a minute to let everything settle, and move to the next step



For more information see WiFi Testing: Configuring a Virtual AP with Limited Stations , Virtual Router with NAT Cookbook , Virtual Router with DHCP Cookbook

5. Start services on the AP system. You need to do this each time you re-apply Netsmith.

A. First, make sure that /etc/hosts on the AP system matches the IP addresses for the vap0, vap1, and the ocsp port (as shown in Netsmith). In this example, I added these rows to /etc/hosts:

```
###LANFORGE-HOSTS-START###
# This section may be over-written by lf_kinstall.pl
127.0.0.1 osu-client.lf0350-c3b0.lanforge.local
10.88.1.1 osu-server.lf0350-c3b0.lanforge.local
127.0.0.1 osu-revoked.lf0350-c3b0.lanforge.local
10.88.1.1 osu-signup.lf0350-c3b0.lanforge.local
7.1.3.2 ocsp.lf0350-c3b0.lanforge.local
###LANFORGE-HOSTS-STOP###

###-LF-HOSTAME-NEXT-###
192.168.100.154 lf0350-c3b0 lanforge-srv
```

B. Start the **O**nline **C**ertificate **S**tatus **P**rotocol (OCSP) script which will restart the OCSP Responder and update the cache once per minute. It is only required on the VAP or server side of a HotSpot 2.0 R2 network. Currently you will need to manually clean up old processes if you are re-doing this step.

```
su - root
cd /home/lanforge
. lanforge.profile
vrf_exec.bash rddVR5 ./ocsp.bash > /dev/null 2>&1 &
```

C. Start Apache httpd configured for HS20 on the Osen AP. You may need to manually stop old httpd processes.

```
vrf_exec.bash vap0 httpd -f /etc/httpd/conf/httpd-hs20.conf
```

6. This is the start of the Station side configuration. Do these actions on the Station LANforge system.

Create `devinfo.xml` and `devdetail.xml` files in /home/lanforge/wifi/osu_wlan2

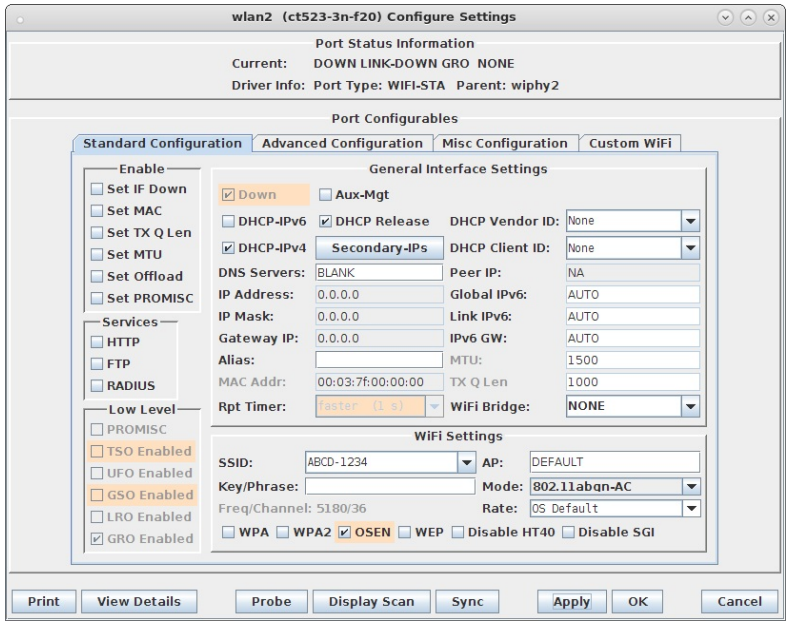
A. /home/lanforge/wifi/osu_wlan2/devinfo.xml

```
<DevInfo xmlns="urn:oma:mo:oma-dm-devinfo:1.0">
        <DevId>urn:Example:HS20-station:123456</DevId>
        <Man>Manufacturer</Man>
        <Mod>HS20-station</Mod>
        <DmV>1.2</DmV>
        <Lang>en</Lang>
</DevInfo>
```

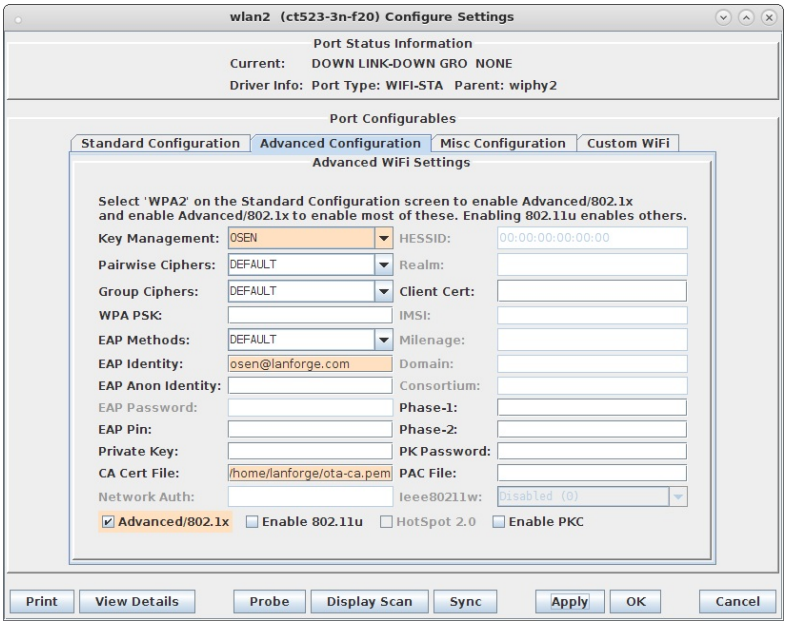
B. /home/lanforge/wifi/osu_wlan2/devdetail.xml

```
<DevDetail xmlns="urn:oma:mo:oma-dm-devdetail:1.0">
        <Ext>
                <org.wi-fi>
                        <Wi-Fi>
                                <EAPMethodList>
                                        <EAPMethod1>
                                                <EAPType>13</EAPType>
                                        </EAPMethod1>
                                        <EAPMethod2>
                                                <EAPType>21</EAPType>
                                                <InnerMethod>MS-CHAP-V2</InnerMethod>
                                        </EAPMethod2>
                                        <EAPMethod3>
                                                <EAPType>18</EAPType>
                                        </EAPMethod3>
                                        <EAPMethod4>
                                                <EAPType>23</EAPType>
                                        </EAPMethod4>
                                        <EAPMethod5>
                                                <EAPType>50</EAPType>
                                        </EAPMethod5>
                                </EAPMethodList>
                                <ManufacturingCertificate>false</ManufacturingCertificate>
                                <Wi-FiMACAddress>020102030405</Wi-FiMACAddress>
                                <IMSI>3100260000000000</IMSI>
                                <IMEI_MEID>imei:490123456789012</IMEI_MEID>
                                <ClientTriggerRedirectURI>http://localhost:12345/</ClientTriggerRedirectURI>
                                <Ops>
                                        <launchBrowserToURI></launchBrowserToURI>
                                        <negotiateClientCertTLS></negotiateClientCertTLS>
                                        <getCertificate></getCertificate>
                                </Ops>
                        </Wi-Fi>
                </org.wi-fi>
        </Ext>
        <URI>
                <MaxDepth>0</MaxDepth>
                <MaxTotLen>0</MaxTotLen>
                <MaxSegLen>0</MaxSegLen>
        </URI>
        <DevType>MobilePhone</DevType>
        <OEM>Manufacturer</OEM>
        <FwV>1.0</FwV>
        <SwV>1.0</SwV>
        <HwV>1.0</HwV>
        <LrgObj>false</LrgObj>
</DevDetail>
```

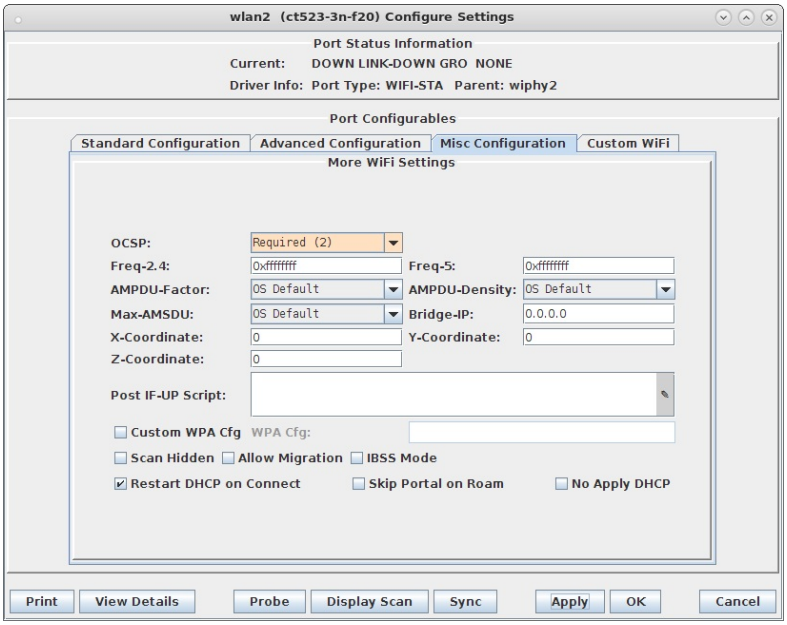
7. Setup wlan2 as the HotSpot 2.0 R2 client.
   A. Modify wlan2 on the Port Mgr tab and set the SSID to the OSEN AP's SSID 'ABCD-1234' in this example and set the authentication to **OSEN**.



B. In wlan2 Advanced WiFi Settings, select Advanced/802.1x, set Key Management, EAP Identity and CA Cert File.



C. In wlan2 Misc Configuration, set OCSP to Required.



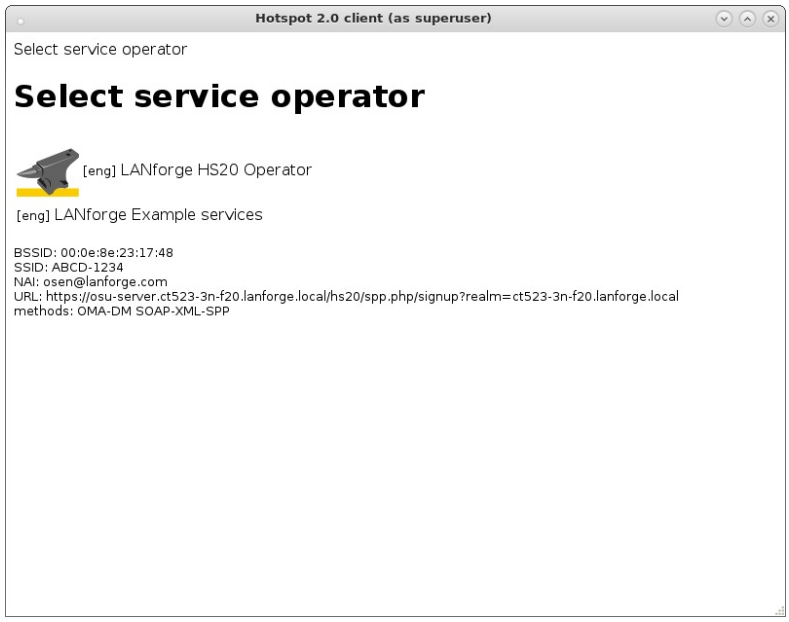
D. Admin up wlan2 and it will associate with the OSEN AP and obtain an IP address on the OSEN AP IP network.

8. Initiate Online Sign-Up

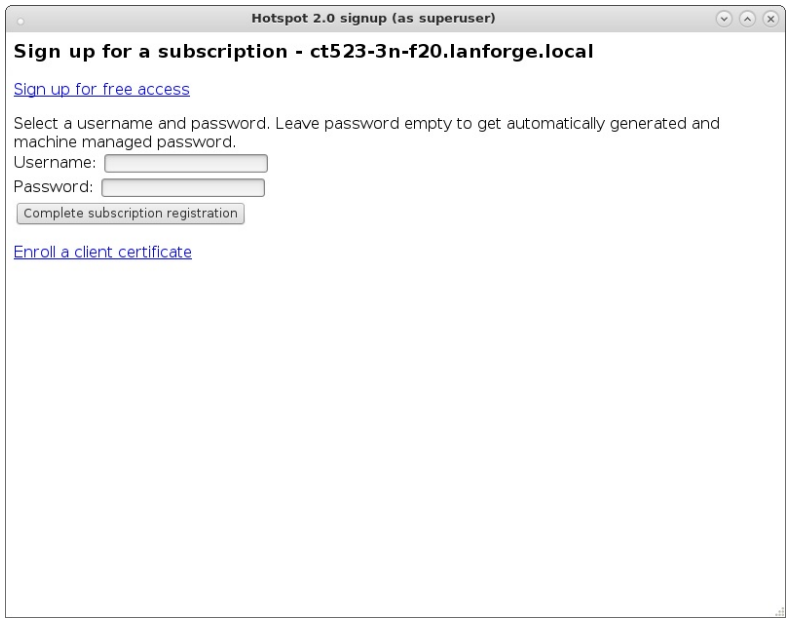
A. Some notes about the following command: We use the LD_PRELOAD trick to override the default DNS servers in the hs20-osu-client program. This means you need to know the DNS ahead of time, and set it with the NAMESERVER environment variable. A second DNS server could be set as NAMESERVER2. In a terminal window type the following:

```
cd /home/lanforge/wifi/osu_wlan2
LD_PRELOAD=/home/lanforge/local/lib/resolvconf-override.so NAMESERVER1=7.1.3.2 \
   vrf_exec wlan2 ~lanforge/local/hs20/client/hs20-osu-client -x /home/lanforge/local/hs20/spp/spp.xsd -dd -S wlan2 signup
```

B. Select 'LANforge HS20 Operator' from the Service Provider List.



C. Select 'Sign up for free access' from the Online Sign-Up page.



D. Select the Accept button to complete the Online Sign-Up.

9. The wlan2 station will obtain an IP address on the Passpoint AP IP network and should be able to access the internet
10. If wlan2 is reset or reassociates with the OSEN AP, you will have to remove the Service Provider (SP) directory before attempting the Online Sign-Up again.

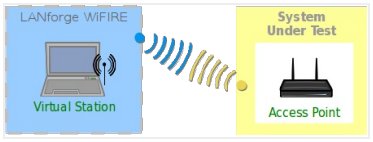
```
cd /home/lanforge/wifi/osu_wlan2
rm -rf SP
```

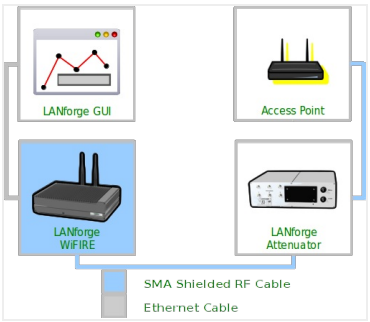
11. NOTES: We found it very difficult to get all of the details correct in this example. Here are some debug notes and links to certain files that may help others or ourselves debug this in the future.
    A. /etc/hosts on the AP system.
    B. /etc/hosts on the Station system.
    C. apache_hs20_config.tar.gz configuration files.
    D. To debug pem files: openssl x509 -in /home/lanforge/hs20/ca/signup-server.pem -text -noout

LANforge WiFi Rate v. Range Test

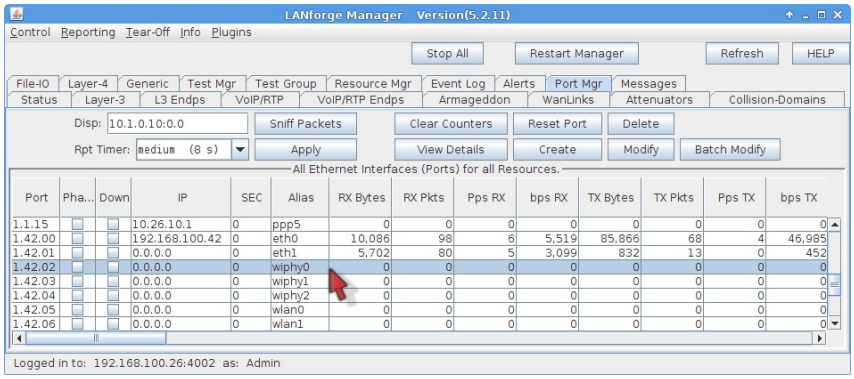
Goal: Compare WiFi performance for 'upload' traffic (client to AP) using a WiFi access point, a LANforge Attenuator and a LANforge Virtual Station. Traffic is generated by a RFC-2544 script on a Layer-3 UDP connection.

This demo consists of one WiFi access point and one CT523 LANforge WiFIRE machine connected to the LANforge Attenuator with coax SMA cables. (This is **not** over the air testing). This requires LANforge release 5.2.7 or higher.



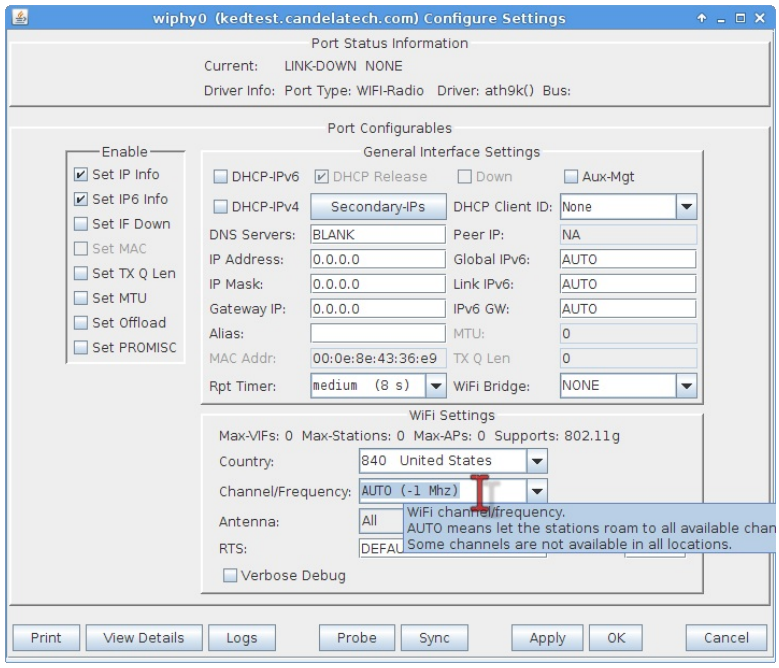


1. Create Virtual Station



   A. Select radio `wiphy1` and click **Modify**

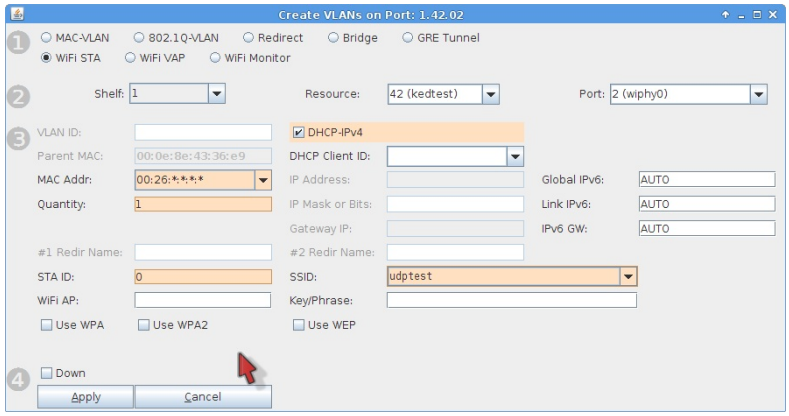
B. Set the frequency of the radio to **Auto**



A. Click **OK**

C. On the Port Modify tab, click **Create**

D. Create virtual station with these paramters:



A. Select **Wifi STA**

B. Choose **DHCP-IPv4**

C. Quantity **1**

D. Station ID `0`

E. SSID `'udptest'`

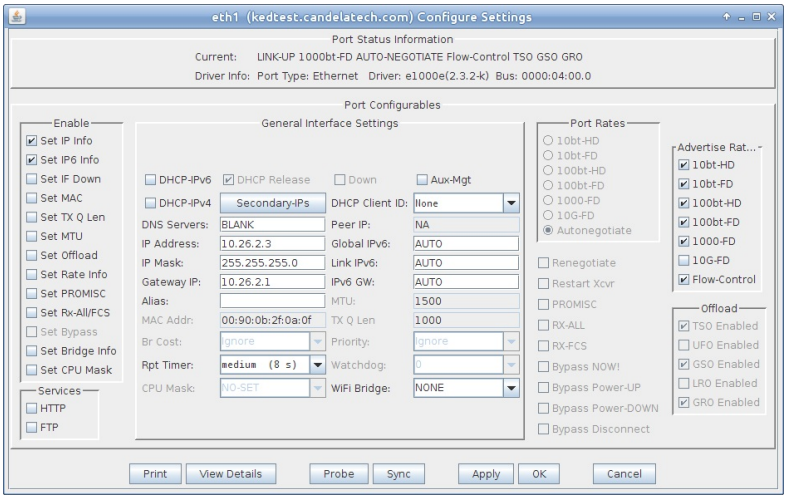
F. Click **Apply**

2. Create upstream port wired to AP

A. Wire the `eth1` port into the upstream connection of the AP under test.

B. On the **Ports** tab, click on the `eth1` port

C. Configure `eth1` port with an upstream IP, like 10.26.2.3

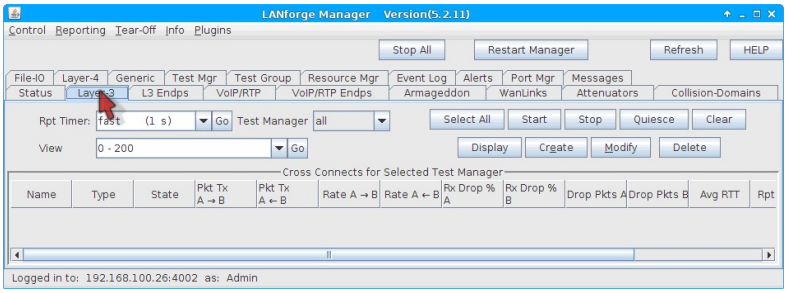


D. Set the **Gateway** to the AP wired interface IP.

E. Click **OK**

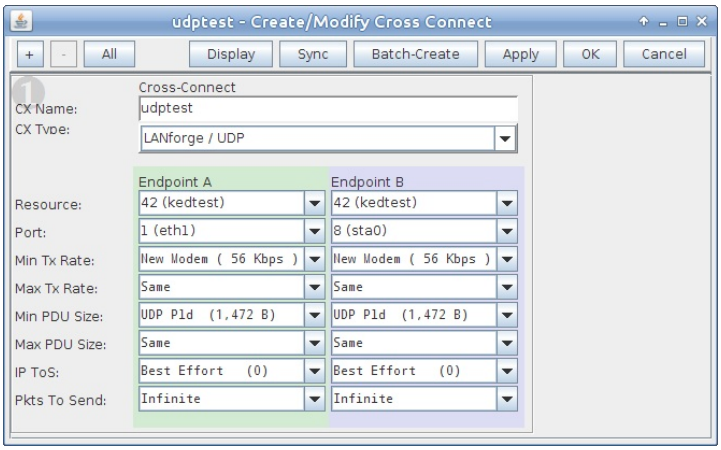
3. Create and Test Cross Connect

A. Go to the **Layer 3** tab



A. Click **Create**

B. Create a cross connect with these qualities:



A. Make sure Endpoint A is **eth1**

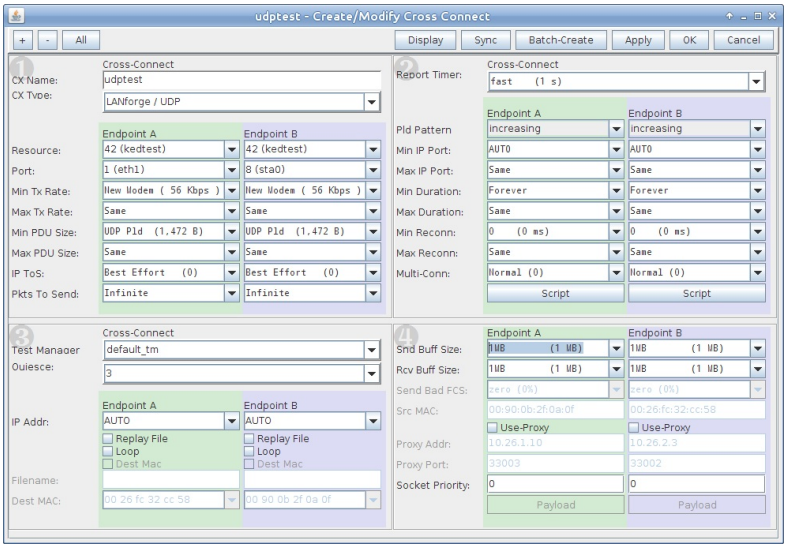
B. Make sure Endpoint B is **sta0**

C. Min PDU Size for both should be `1472`

D. **NOTE:** These rate and PDU size settings will be manipulated by the script we setup later.

C. Next, expand the screen to **Level 4** using the [+] button.

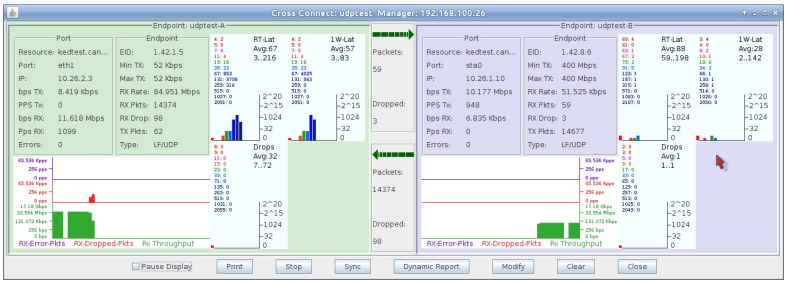
D. Configure the send buffer on **Endpoint B** to `1 MB`



A. Click **OK**

E. On the **Layer 3** tab, click **Start** to verify the AP and Station can connect

F. Click the **Display** button to monitor throughput

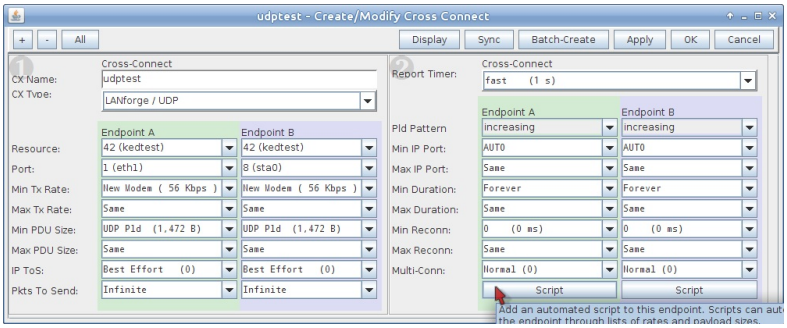


G. Only a short confirmation is necessary, click **Stop** on the Layer 3 tab

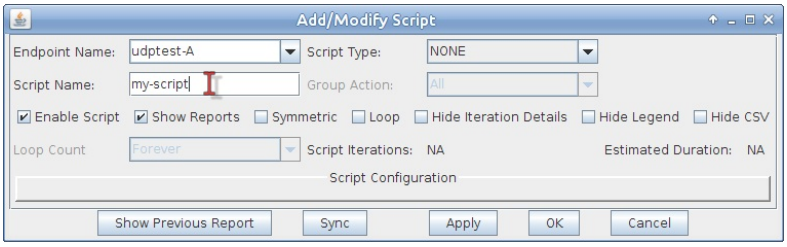
4. Configure Scripting for Cross Connect

A. On the **Layer-3** tab, click **Modify**

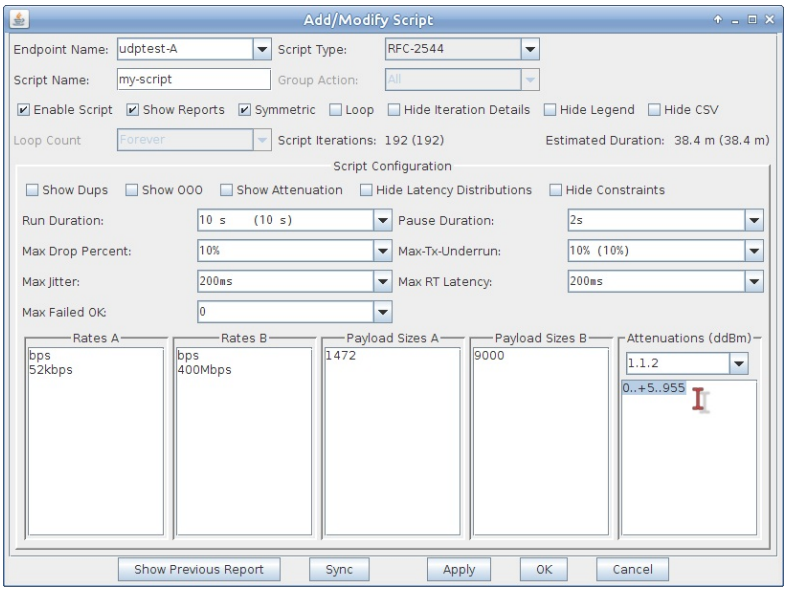
B. In the Level 2 box, click Endpoint A **Script** button



C. The Cross Connect Script window displays with no parameters



D. Select Script Type: **RFC-2544**. Set the following parameters:



A. Select **Symmetric**. This will increment both the A and B rates and payload sizes. (Instead of just side A).

B. Select **Show Attenuation**. This displays attenuation levels in the report.

C. Run Duration: `10 sec`. This is how long each rate setting will be held.

D. Pause Duration: `2 sec`. We give it some time to transition.

E. Max Drop Percent `10%`

F. Max Jitter `200ms`

G. Max RT Latency `200ms`

H. Rates A: `56kbps`. This sets the client upload target rate.

I. Rates B: `400Mbs`. This sets the client download target rate.

J. Pld A: `1472` This sets the client (sta0) MTU. This is a 1500 byte wire packet.

K. Pld B: `9000` This sets the upstream (eth1) MTU.

L. Attenuator Resource: **1.1.2**. You can find your attenuator resources in the **Attenuator** tab.

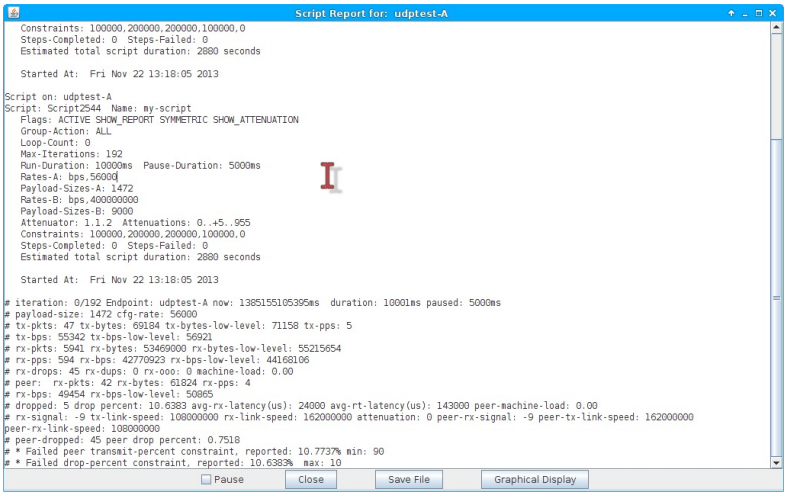
M. Attenuation: `0..+5..955`. This is shorthand for: Begin at zero dB attenuation, increase in 0.5dB steps, until 955 dB of attenuation. Individual dB steps could also be specified.

N. Click **OK**

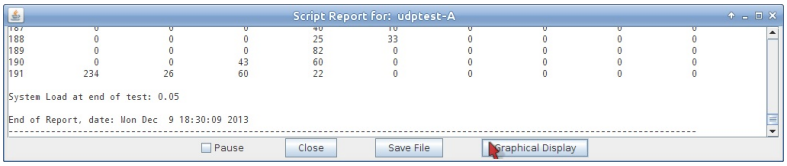
E. On the **Create/Modify Cross Connect** window, click **OK**

5. Run the Cross Connect and Generate a Report

A. On the **Layer-3** tab, click **Start**



B. The Scripting Report window will appear

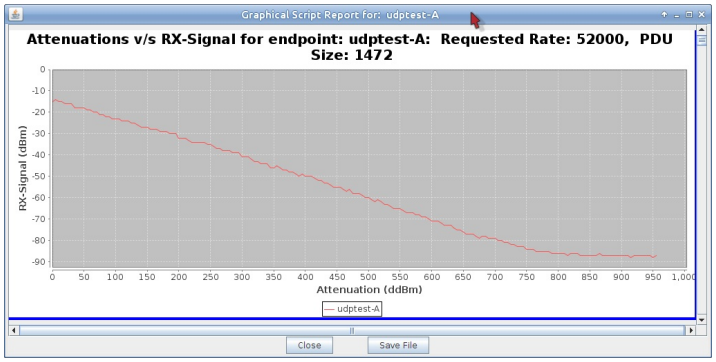


A. When the script completes, you can view the graphed results.

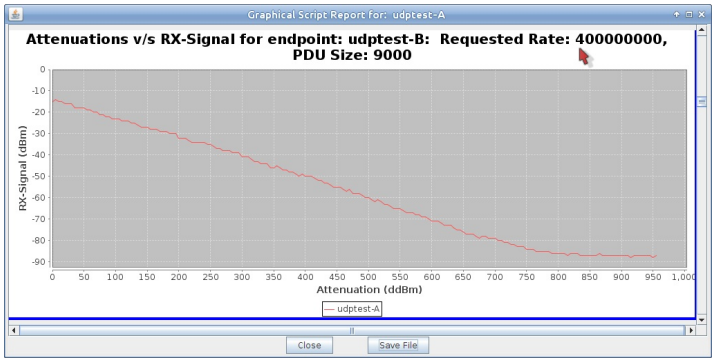
B. Click on **Graphical Display** and a window with the graphical report will display

C. Scroll to the top of the window to view the graphs. Highlights are shown below.

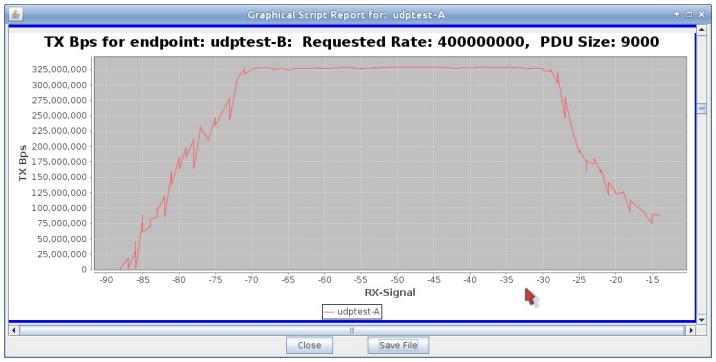
D. Attenuation v. RX signal, endpoint A



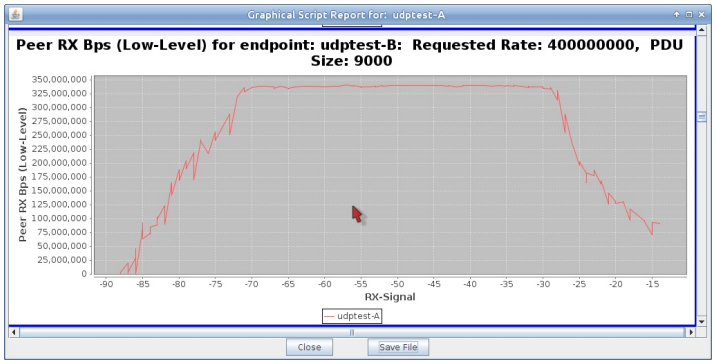
E. Attenuation v. RX signal, endpoint B



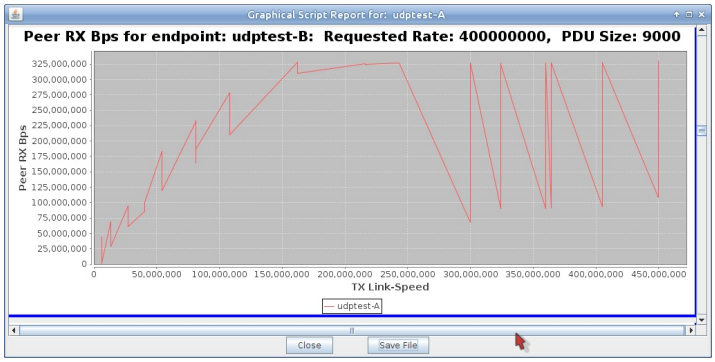
F. TX rate (UDP payload) v. RX signal, endpoint B



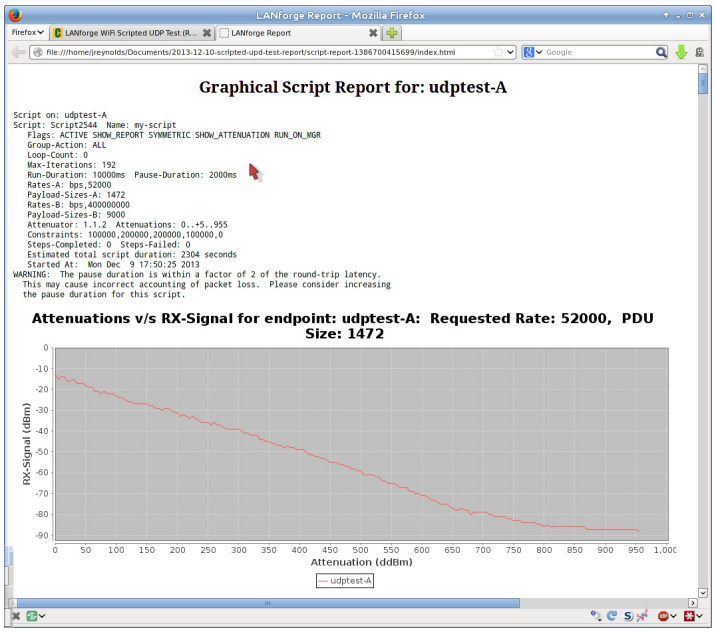
G. RX rate including frame headers (UDP payload with frame headers) v. RX signal, endpoint B



H. RX rate v. TX link speed. You see a sawtooth pattern becuase all attenuations are included.



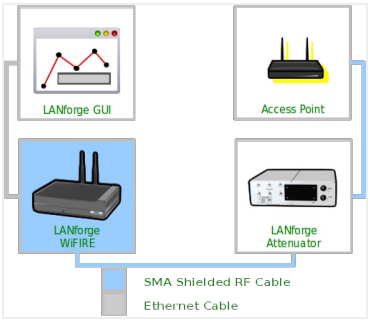
I. Click on **Save File** and your browser will appear the the HTML copy of the report.



LANforge WiFi Attenuation vs PDU Size Test

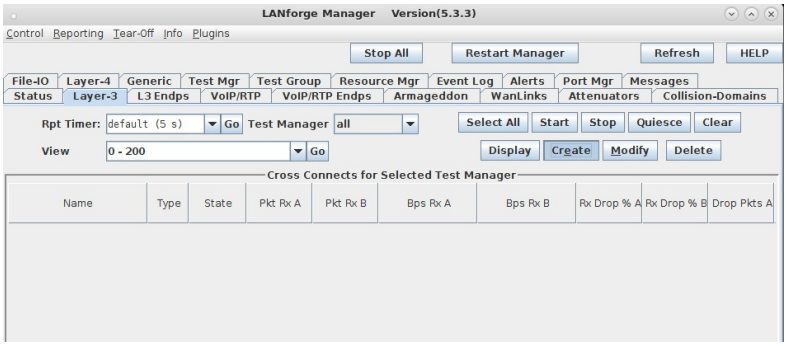
Goal: Compare WiFi performance for 'download' traffic (AP to Client) as Payload Size and RF Attenuation levels change using a WiFi access point, a LANforge Attenuator and a LANforge Virtual Station. Traffic is generated by a RFC-2544 script on a Layer-3 UDP connection.

This demo consists of one WiFi access point and one CT523 LANforge WiFIRE machine connected to the LANforge Attenuator with coax SMA cables. (This is **not** over the air testing).



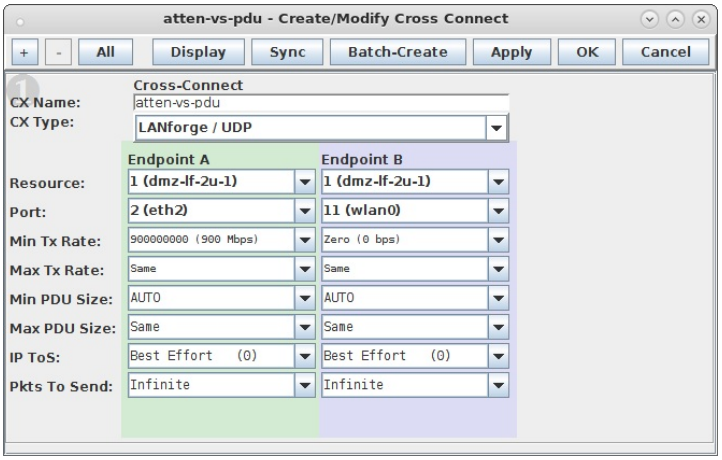
1. Create Layer-3 Cross Connect

A. Go to the **Layer 3** tab



A. Click **Create**

B. Create a cross connect with these settings:



A. Make sure Endpoint A is **eth2** or the wired port to the AP

B. Make sure Endpoint B is **wlan0** or the station associated with the AP

C. Min PDU Size for both should be `AUTO`

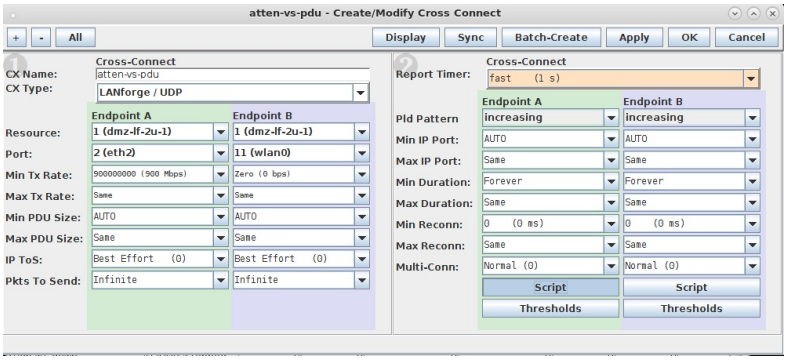
D. **NOTE:** These rate and PDU size settings will be manipulated by the script we setup later.

C. Verify that the connection is operational before adding a script.

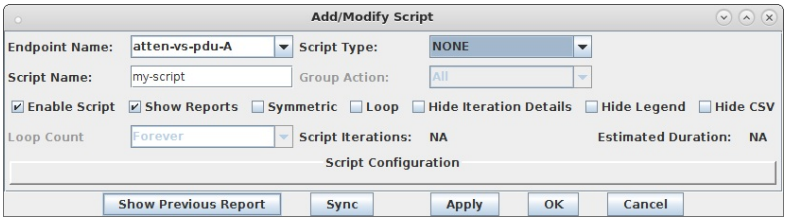
2. Configure Scripting for Cross Connect

A. On the **Layer-3** tab, click **Modify**

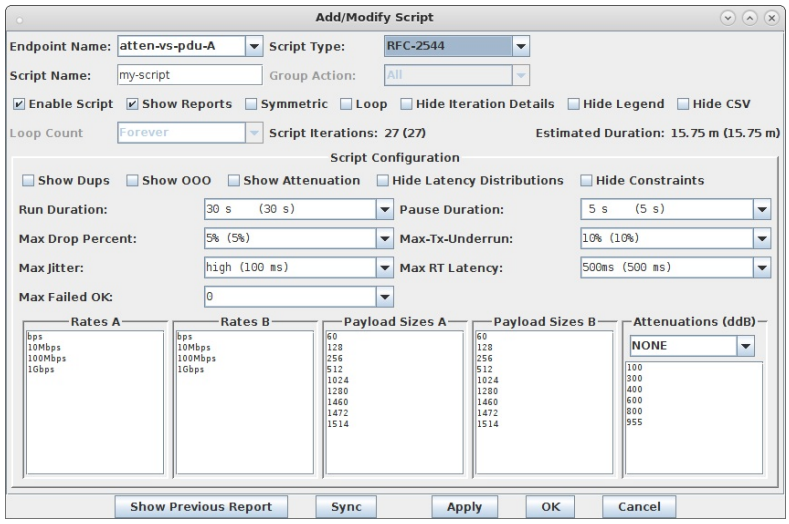
B. In the Level 2 box, click Endpoint A **Script** button



C. The Cross Connect Script window displays with Script Type: **NONE**



D. Select Script Type: **RFC-2544** and default values appear:

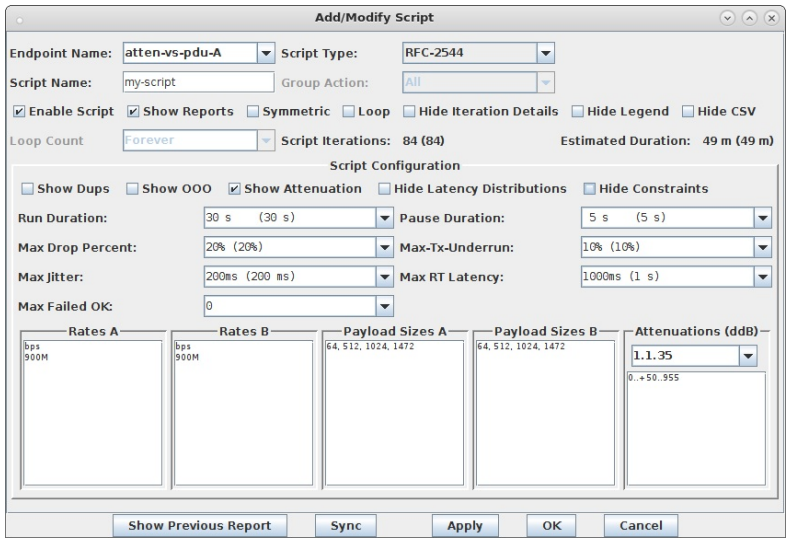


A. Select **Show Attenuation**. This displays attenuation levels in the report.

B. Run Duration: `30 sec`. This is how long each iteration will run.

C. Pause Duration: `5 sec`. We give it some time to transition.

E. Setup Pass/Fail Criteria and Iteration Steps for the script. For each Attenuation Level, the script will step through each PDU size at the desired rate. If there were multiple rates then for each Attenuation Level, the script would step through each PDU size for each rate.



A. Max Drop Percent `20%`

B. Max Jitter `200ms`

C. Max RT Latency `1000ms`

D. Rates A: `900Mbps`. This sets the client download target rate.

E. Pld A: `64,512,1024,1472`

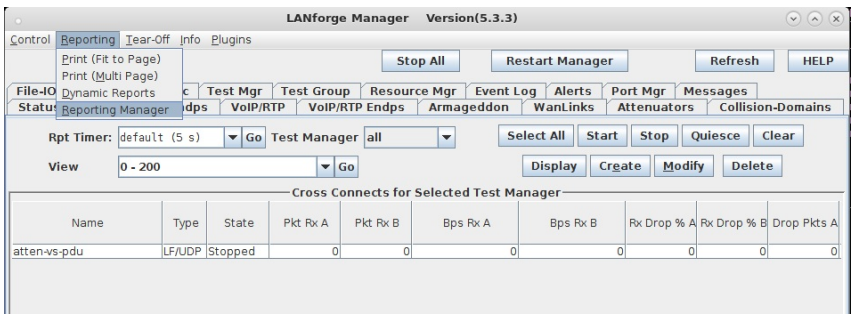
F. Attenuator Resource: **1.1.35**. You can find your attenuator resources in the **Attenuator** tab.

G. Attenuation: `0..+50..955`. This is shorthand for: Begin at zero dB attenuation, increase in 5.0dB steps, until 955 dB of attenuation. Individual dB steps could also be specified.

H. Click **OK**

F. On the **Create/Modify Cross Connect** window, click **OK**

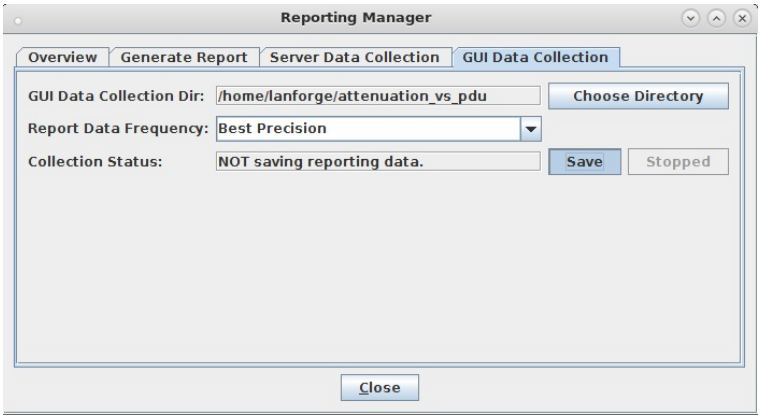
3. Save Data to be able to view past results.



A. Go to Reporting Menu and select Reporting Manager

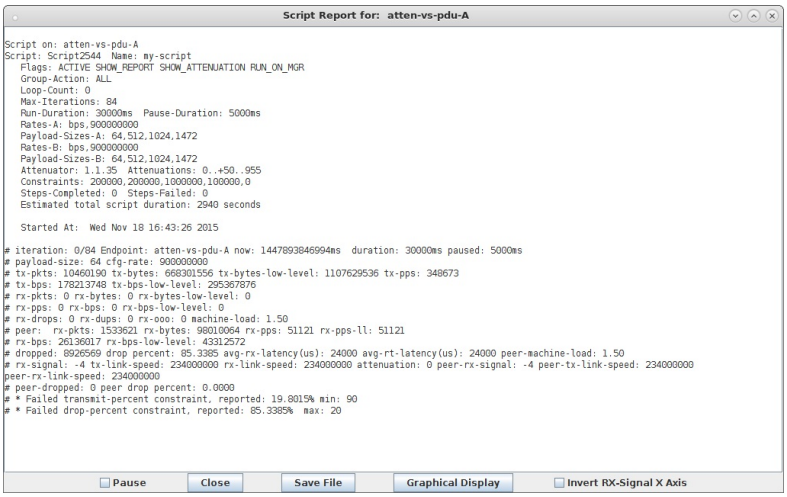
B. Select the GUI Data Collection tab

C. Choose a directory and select Save



4. Run the Cross Connect and Generate a Report

A. On the **Layer-3** tab, click **Start**, and the Script Report window will appear.



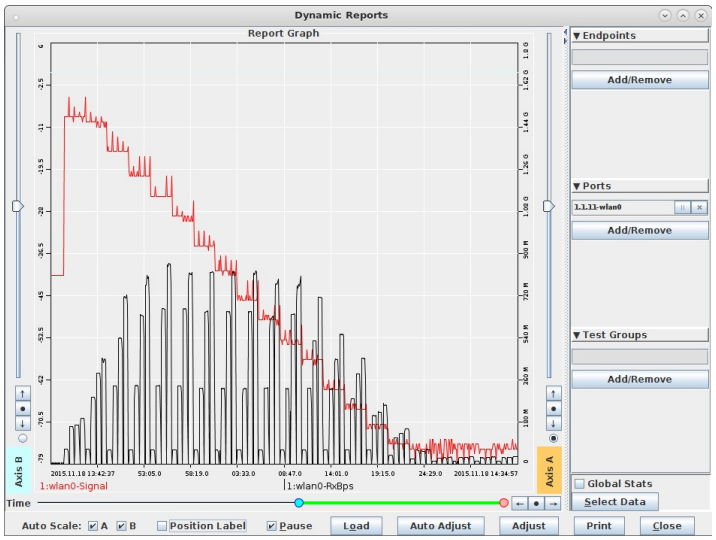
B. View the Dynamic Report

A. While the script is running, you can view the real-time results of the running script.

B. Right-Click on **wlan0** and select Dynamic Report

C. Setup the Dynamic Report to view the data you are interested in.

D. wlan0: RX Signal level shown on Axis-B and RX bps shown on Axis-A



E. To view Dynamic Report data after the test completes:

F. Set the Reporting Manager, Generate Report to the directory containing saved data.

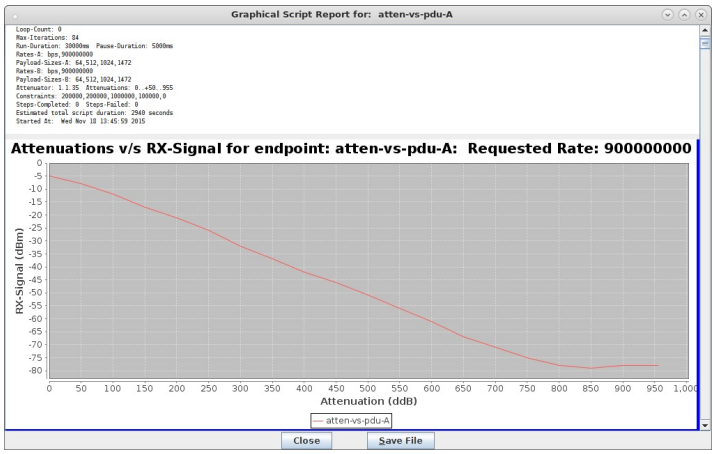
G. Adjust the time scale and Load the data in the Dynamic Report window.

C. View the graphical results of the script when it completes.

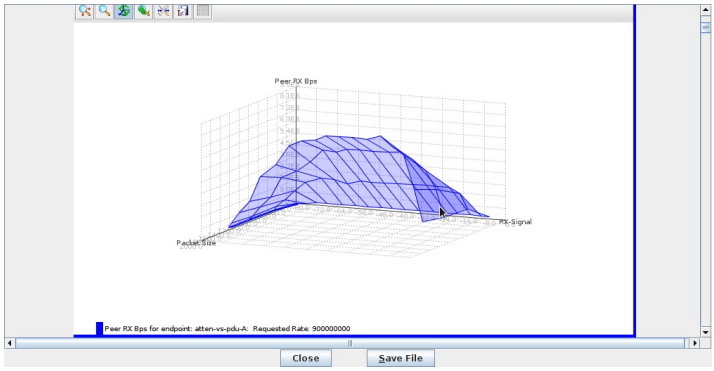
A. In the Script Report window, click on **Graphical Display** and a window with the graphical report will display

B. Scroll to the top of the window to view the graphs. Examples are shown below.

C. Attenuation v. RX signal, endpoint A



D. 3D graph of Peer RX Bps



E. Click on **Save File** and your browser will appear the the HTML copy of the report.
See also: Full Report Raw Report Text

LANforge WiFi Capacity Testing

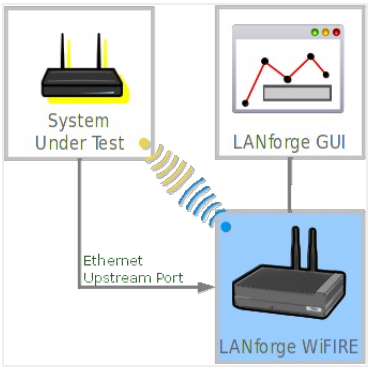
Goal: Use the WiFi Capacity plugin to emulate traffic from hundreds of virtual stations across an access point and report the results.

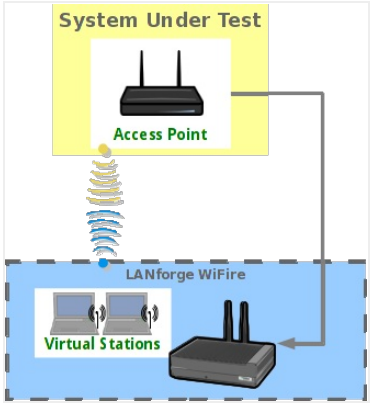
Requires LANforge 5.2.11 or later. Configure 128 stations and assign them an SSID. Use the 'WiFi Capacity' LANforge-GUI plugin to emulate:

- Station download traffic through the AP
- Station-to-station traffic

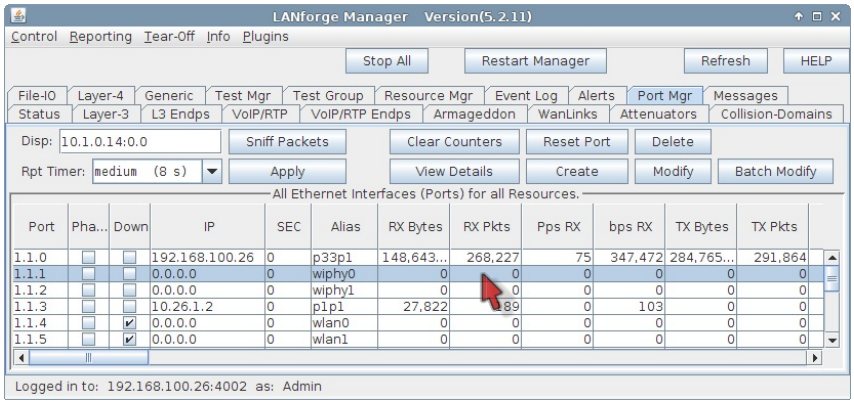
The `eth1` port of our LANforge in this example is connected to the upstream network of the AP.

This example uses a LANforge CT523 system but the procedure should work on all CT520, CT523, CT525 and similar systems.

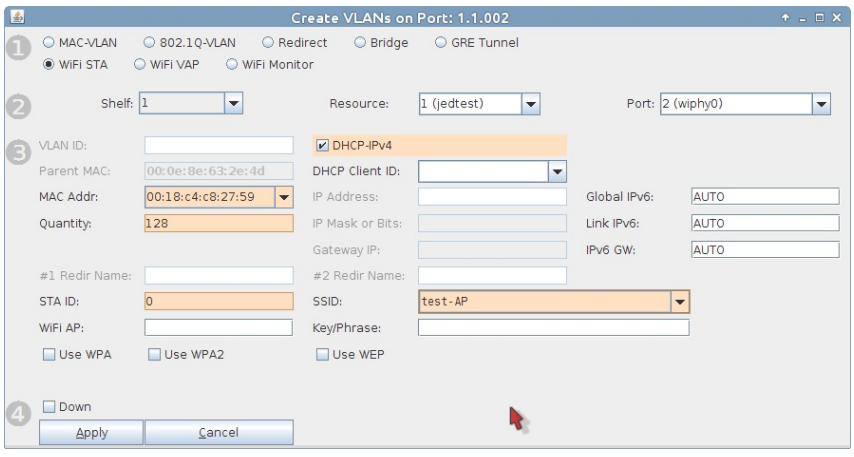




1. In the **Port Mgr** tab, select the radio `wiphy0` and click **Create**.



2. The **Create VLANs** window appears. Set the values below.



A. Check **WiFi STA**.

B. In the **MAC Addr** dropdown, choose `RANDOM`.

C. Select **DHCP-IPv4**.

D. Specify `128` for **Quantity**.

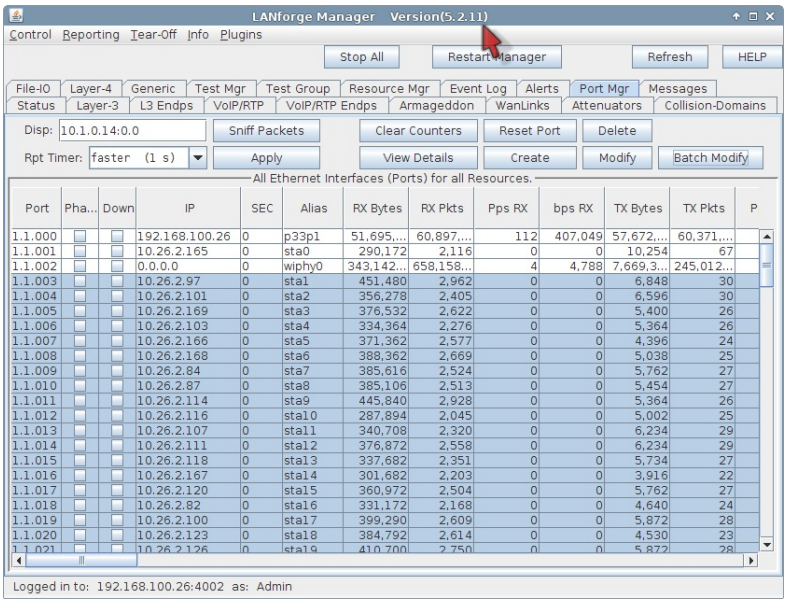
E. Set **STA ID** at `0`.

F. Our AP for this example is using **SSID** `testAP` with **Keyphrase** `test-AP1`. (no dashes allowed)

G. Click the **Apply** button.

H. And then close the window by pressing the **Cancel** button.

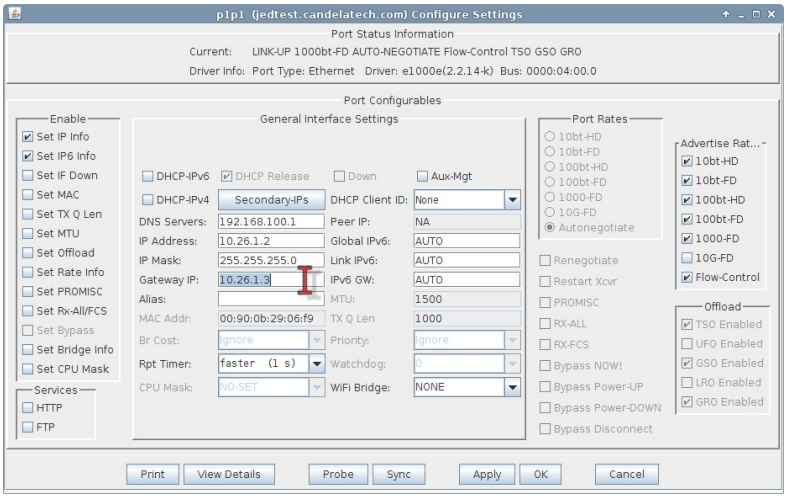
I. In the **Port Mgr** tab you will see the new WiFi stations:



3. Discussion of Capacity Test

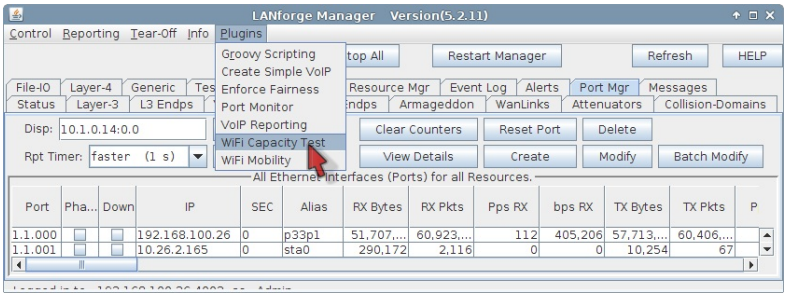
A. **Computing Intervals and Duration:** The WiFi Capacity Test is intended to run in multiple intervals. How many intervals is a product of the number stations in the test divided by the interval increment: `Intervals = (num stations / station increment).` The duration of the test is the product of the number of intervals times the sum of the interval duration plus setup timeout between intervals. Given the setup time between intervals is J seconds, a test of 128 stations with a station increment of 8 and an interval duration of 30 seconds, the entire test duration is: `(128 / 8) * (30 + J) = 480 + 16J seconds`. The actual setup time depends on two factors: a) If **Seek Lower Rates** is enabled, which will attempt to test each rate to try and even out connection rates, and b) DHCP performance. Adding piles of DHCP assignments for every increment can take several seconds, and actual performance depends on your test environment.

B. **Configuring Routes:** the upstream port of your LANforge system should be able to ping the virtual stations. You can use a command from the LANforge command prompt to test this out: `ping -I eth1 10.26.2.100` where eth1 is your upstream port on the same switch as the AP, and 10.26.2.100 is the IP of the virtual station. In the `eth1` **Port Properties** window, you will probably want to set the gateway address for the port to the IP of the AP. In this case it is `10.26.1.2`. Without this routing configured, the only test performed will be the station-to-station test.

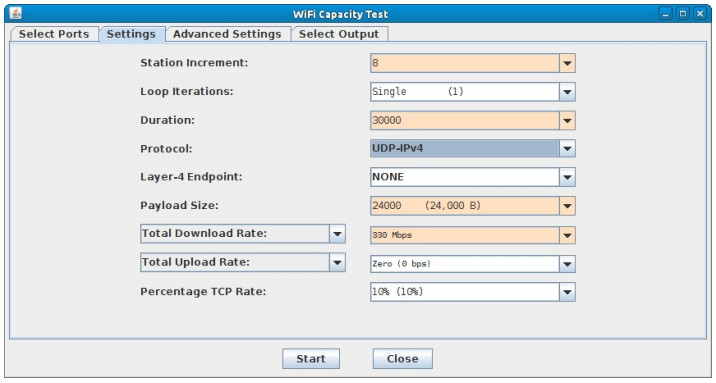


4. Downloading via 128 Virtual Stations

A. In the **Plugins** dropdown menu, select **WiFi Capacity Test**, then begin to configure ports for the test.



A. Highlight free ports that you want to add to the test, this includes the upstream port `eth1` and the virtual stations `sta0 - sta127`.



B. Set **Station Increment** to `8`. This will run 16 intervals with 8 stations added each time going up to 128 total.

C. Set the increment **Duration** to thirty seconds (in milliseconds) so `30000`.

D. Select `UDP-IPv4` for the **Protocol**.

E. For the **Payload** size select `24,000 B`.

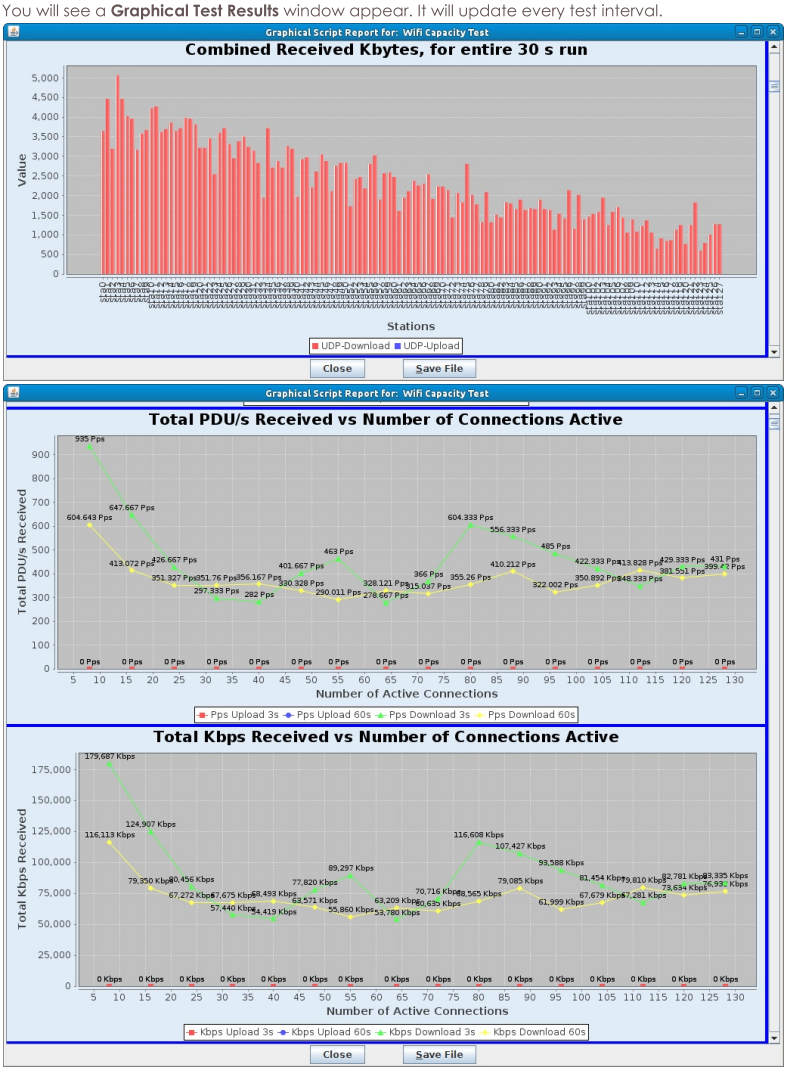
F. Configure the **Download Rate** at `330 Mbps`. The theoretical maximum for AP throughput on one radio is about 340 Mbps with perfect conditions. Even 330 Mbps might not be achieved. This download rate is the rate requested of the AP by all the virtual stations. With eight virtual stations, the target download rate per station is 41.25 Mbps.

G. In **Advanced Settings**, set a **Socket Buffer** of `1MB`.

H. Deselect **Try Lower Rates**. This will reduce interval setup time.

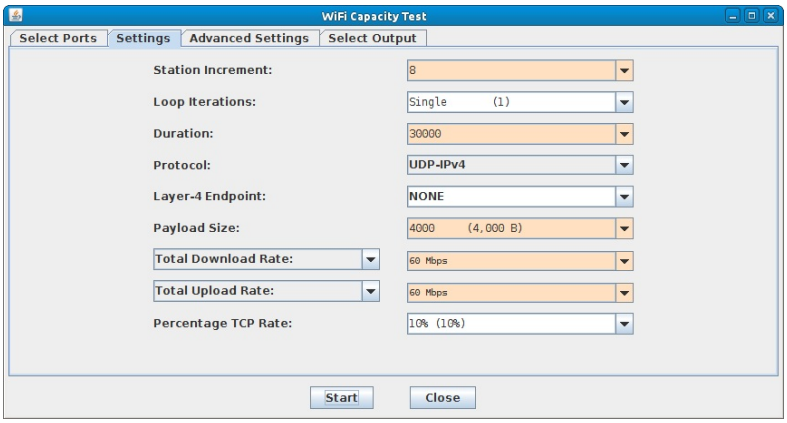
B. Click the **Start** button to begin the test.

C. You will see a **Graphical Test Results** window appear. It will update every test interval.



For more information see LANforge GUI User Guide: Plugins:WiFi Capacity Test.

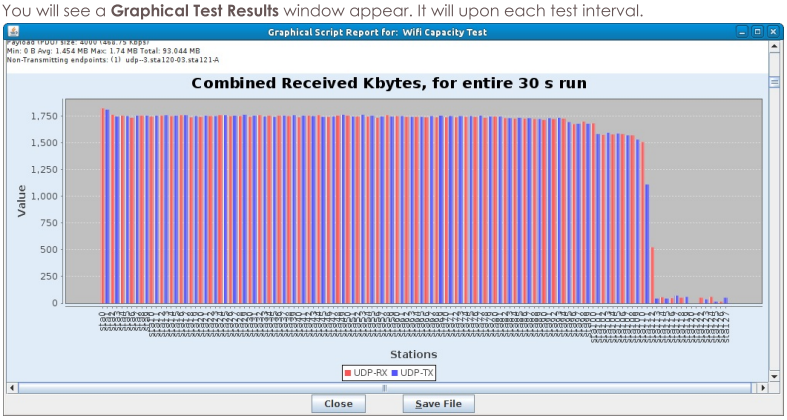
5. Emulating station-to-station traffic with 128 virtual stations:
   A. Go to the **WiFi Capacity Test** window.
   B. This test is very similar. We remove the upstream port `eth1` from the **Ports in Use** list (in the **Select Ports** tab). We then configure the traffic to use smaller TCP packets.

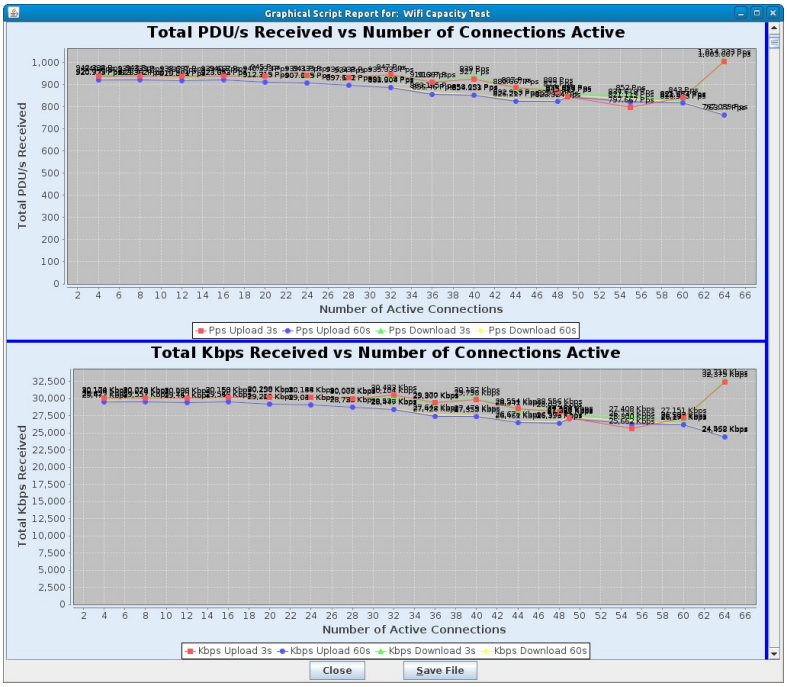


   A. In the settings tab, set **Protocol** and **Payload** to `TCP-IPv4` and `4000 B`.
   B. Change **Total Download** and **Total Upload** rate to `60 Mbps` and `60 Mbps`. These settings should allow up to 120 Mbps between two stations, which will clearly saturate an AP.

C. Click **Start** to begin the test.

D. You will see a **Graphical Test Results** window appear. It will upon each test interval.

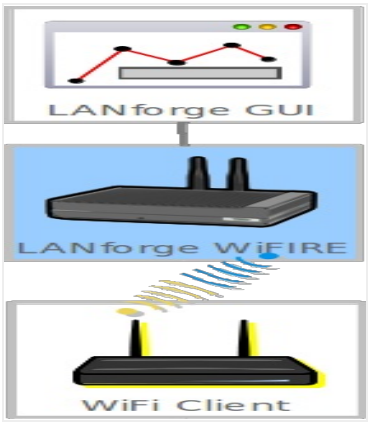


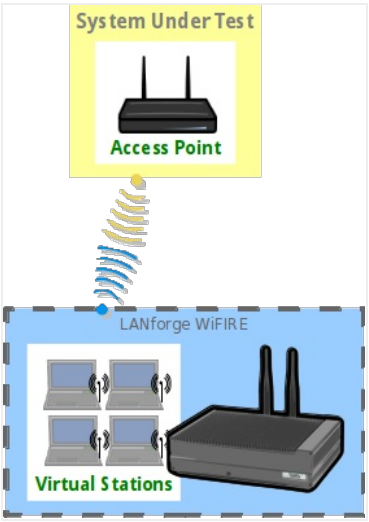


LANforge WiFi Station Reset Testing

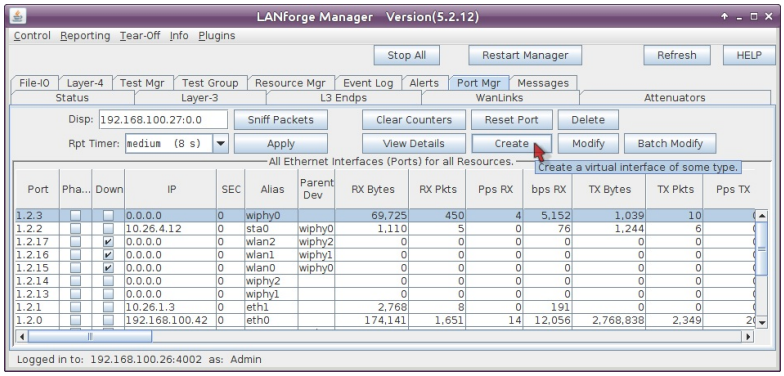
Goal: Use the WiFi Port Reset plugin to emulate restarting of stations associated to your WiFi network.

Requires LANforge 5.2.12 or later. Restarting a WiFi station exercises the whole network stack because it forces negotiation across the wired network: your AP, your DHCP server, your AP controller, and possibly your RADIUS server. Performing this test at an unusually high frequency is a challenging robustness test for your wireless topology and can expose possible race conditions when large numbers of stations reset simultaneously while passing traffic. This cookbook assumes the System Under Test is your AP and that a CT520 (or better) is emulating stations to be reset on the network.

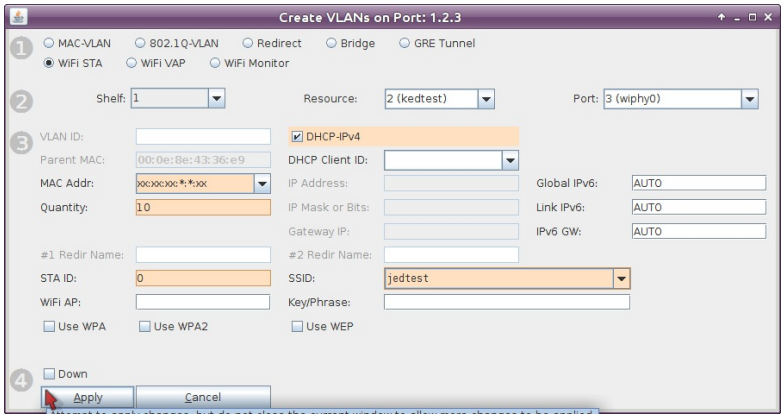




1. Create WiFi stations
   A. In the **Ports** tab, select wiphy0 and click **Create**



   B. In the **Create VLANS** window, craft ten wifi stations:



A. Select WiFi STA

B. For MAC address, choose `xx:xx:xx:*:*:xx`

C. Select DHCP-IPv4

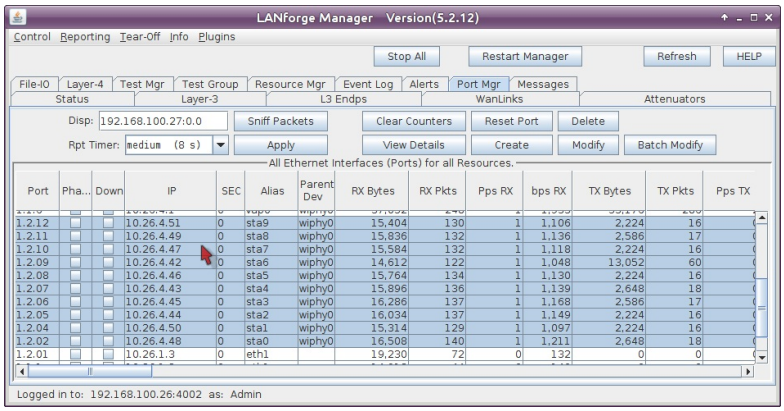
D. Enter Quantity **10**

E. Specify **0** for STA ID

F. The example SSID for this cookbook is `jedtest`

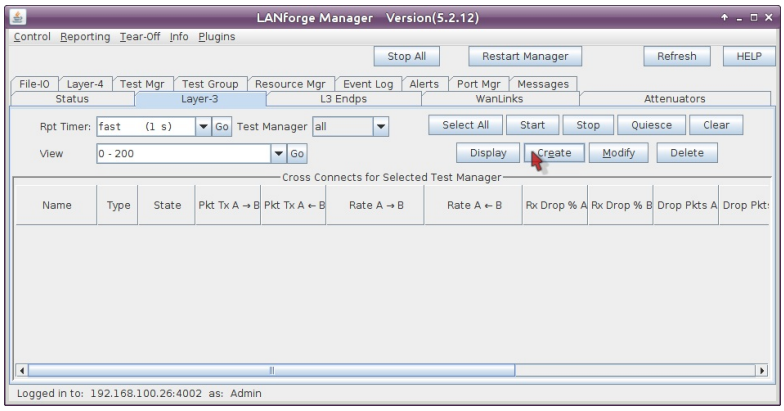
G. ...and then click **Apply**

C. You will see ten stations created:

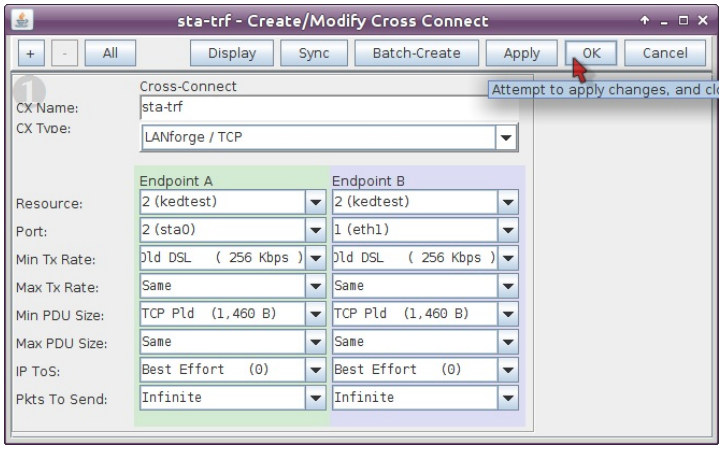


2. Create Connections to Stations

A. In the **Layer-3** tab, click **Create**



B. Create station download traffic



A. This example connection is named sta-trf

B. Connection Type is **LANforge / TCP**

C. This example resource is **kedtest**, where our stations live

D. The Endpoint A Port will be the station **sta0**,

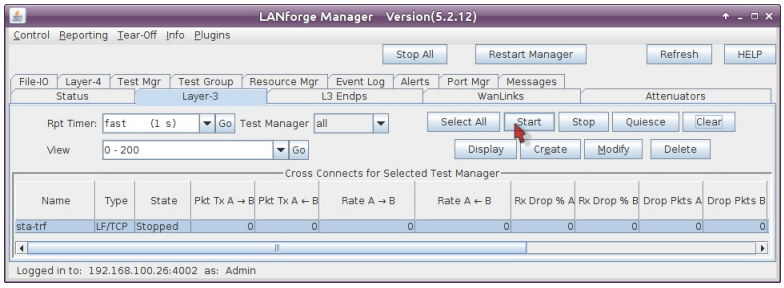
E. and the Endpoint B Port will be upstream of the ap, **eth1**.

F. We'll set the Min Tx Rate for both sides to **265 Kbps**

G. and set the PDU Size to `TCP Pld (1,460 B)`.

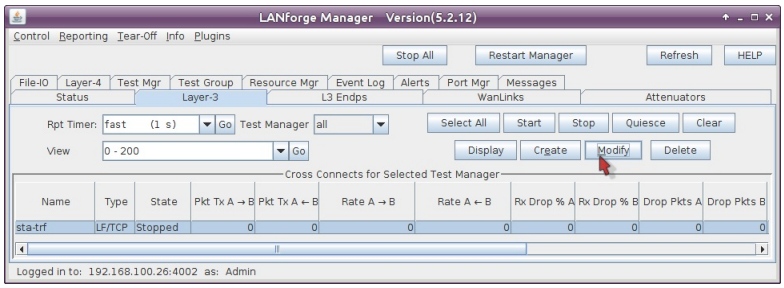
H. ...then click **OK**

C. Test this station by selecting it and clicking **Start**

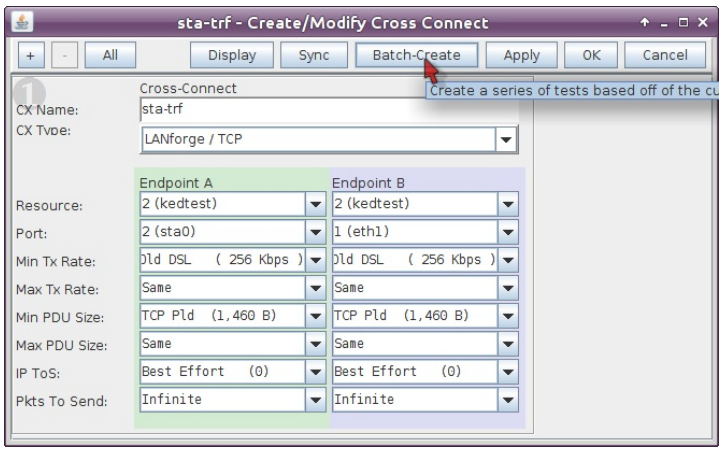


D. Click **Stop** when you are done testing the connection

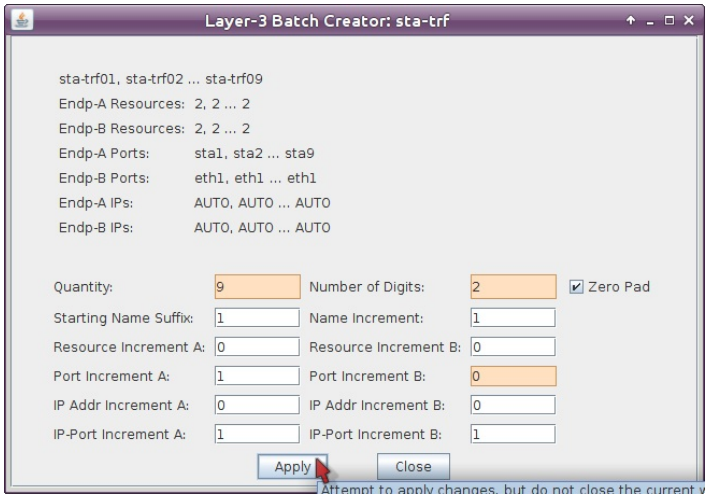
E. Click **Modify** for `sta-trf` and we will batch create nine more:



F. Click **Batch Create** in the Create-Modify Cross Connect window



G. Create nine more cross connects like this one:

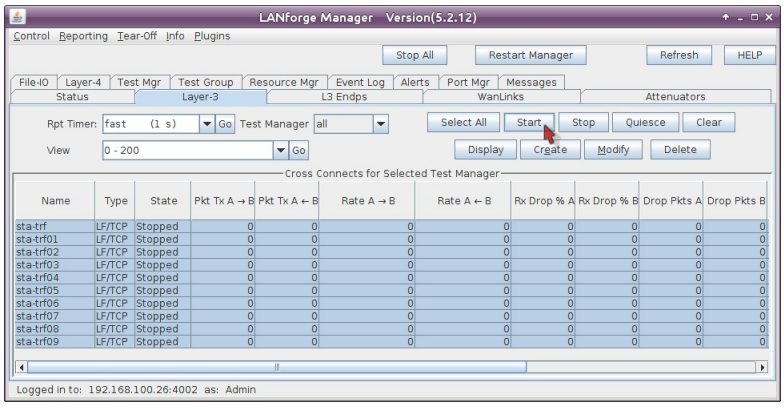


A. Set Quantity to **9**

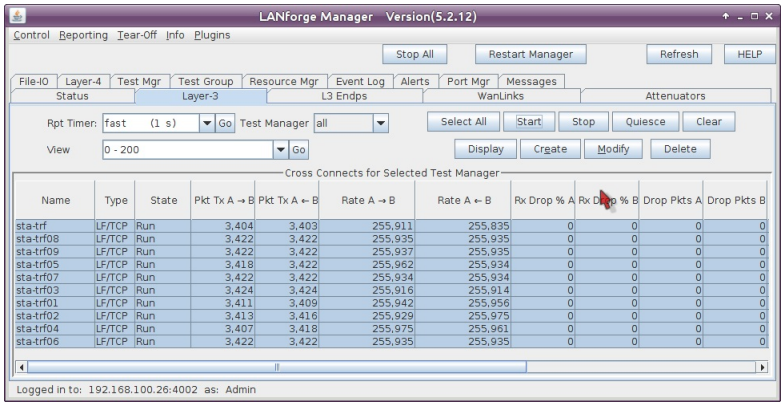
B. Set Number of Digits to **2**

C. We are not changing the B-side port, so we do not need to increment it. Set the Port Increment B to **0**

H. Select all connections and click **Start**

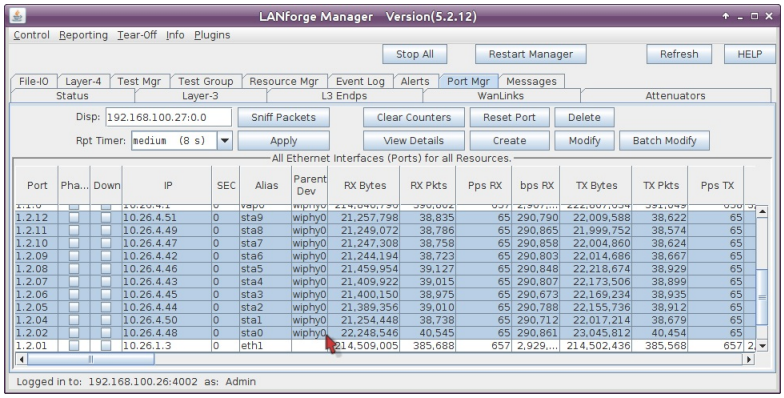


I. Connections should not show dropped packets in the __Rx Drop % A or Rx Drop % B__ columns

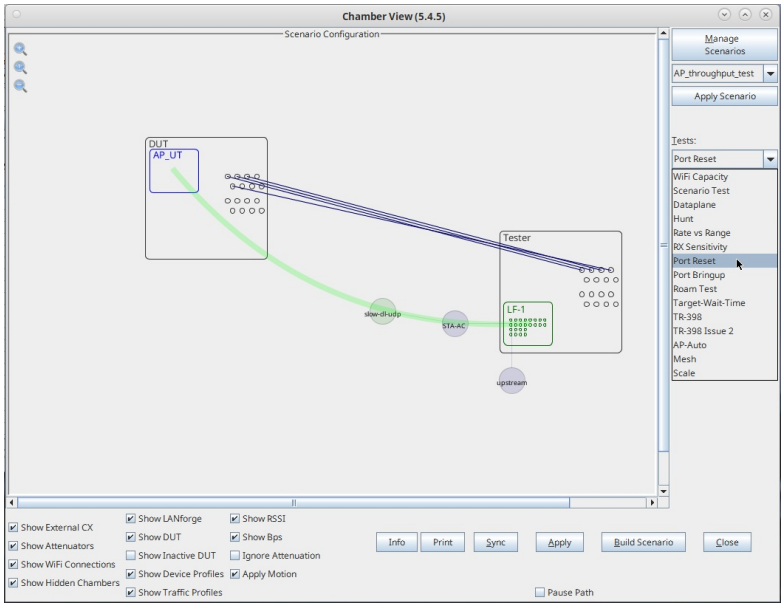


3. Configure Port Reset Script

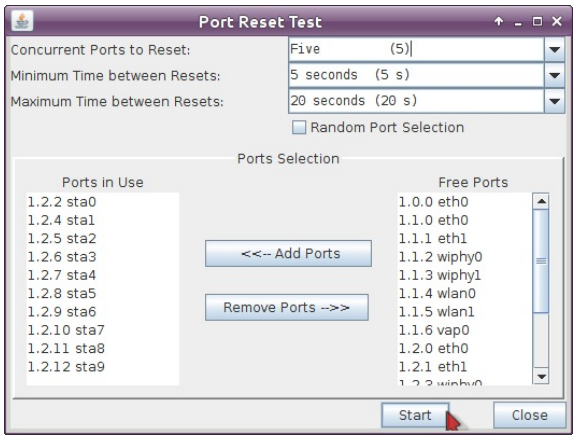
A. In the **Port Manager** tab, select stations **sta0 - sta9**



B. In the **LANforge Manager** windows, select the Chamber View→Tests→Port Reset option→Run Test



C. In the Port Reset Test window, you will see the ten ports already selected. We will configure quick resets for this test:



A. Set Concurrent Ports to Reset to **5**

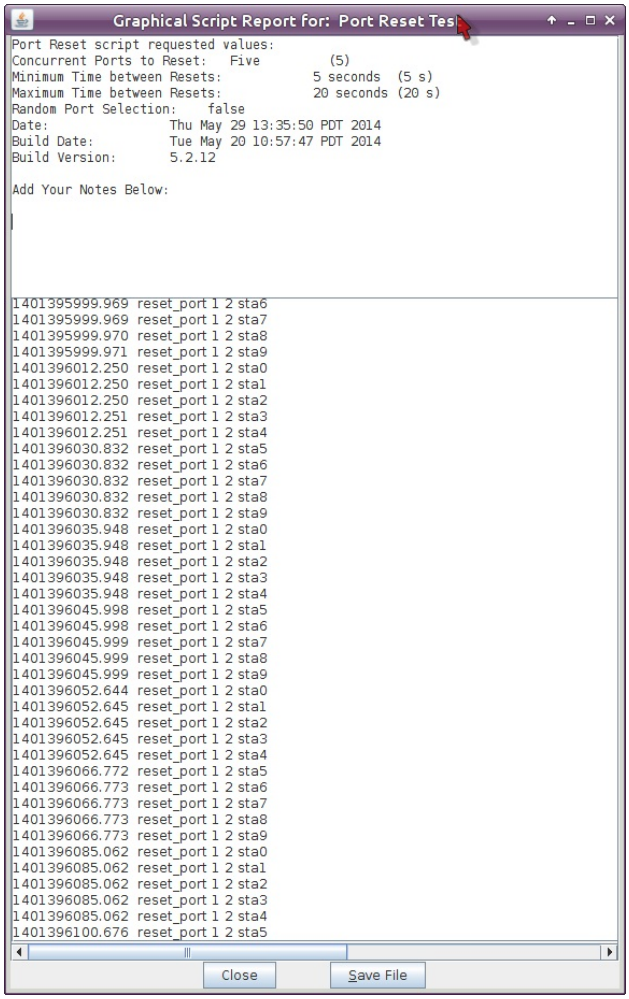
B. Set Minimum Time between Resets to **5 seconds**

C. and Maximum Time between Resets to **20 seconds**

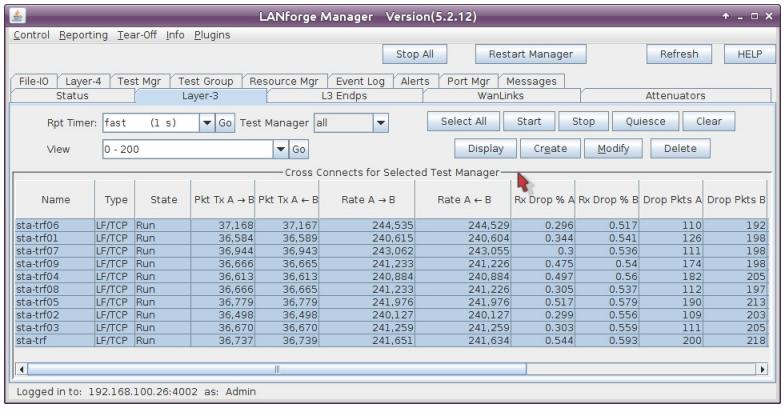
D. ...and click **Start**

4. Observe Results

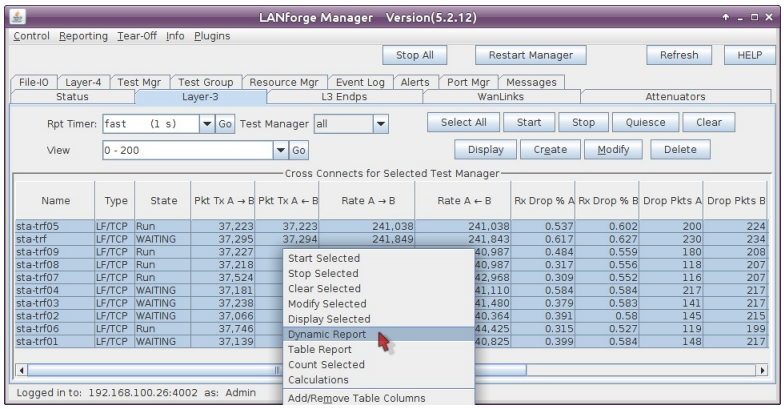
A. The Port Reset Test Results window will show the list of ports getting reset.



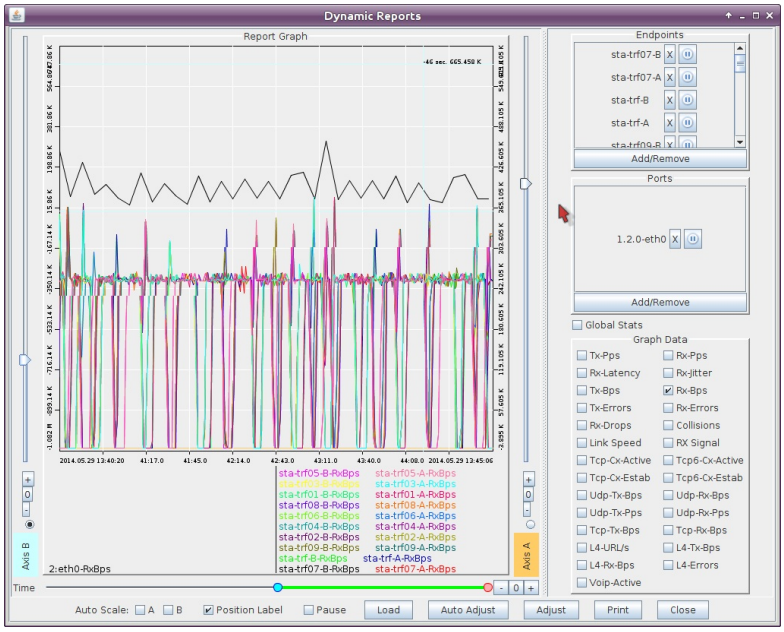
B. The Layer-3 tab will show the amount of lost and dropped packets.



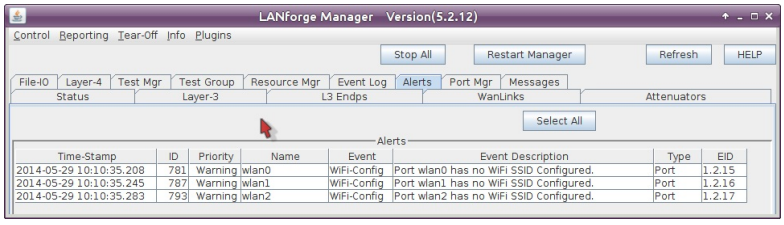
C. We can graph the throughput of the connections with the Dynamic Report menu option



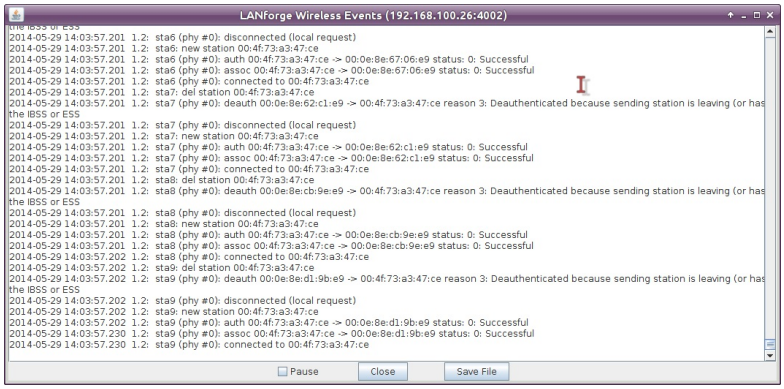
D. In the Dynamic Reports window, we are graphing the **Rx-Bps** for each connection in axis-A, and in axis-B we are graphing the Rx-Bps for the upstream port, eth1



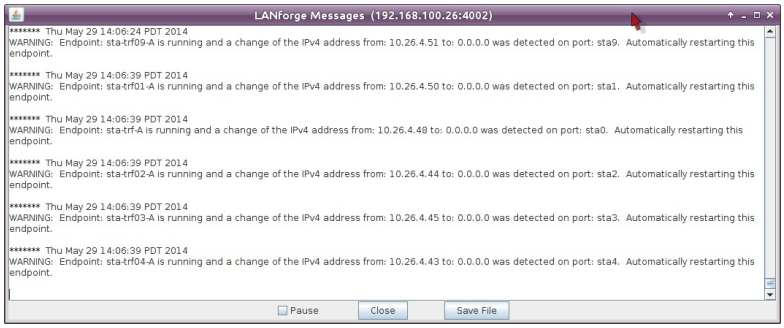
E. You will also want to watch for warnings and failures. In the Alerts tab, you will see persistent alerts. The alerts in this picture can be safely ignored:



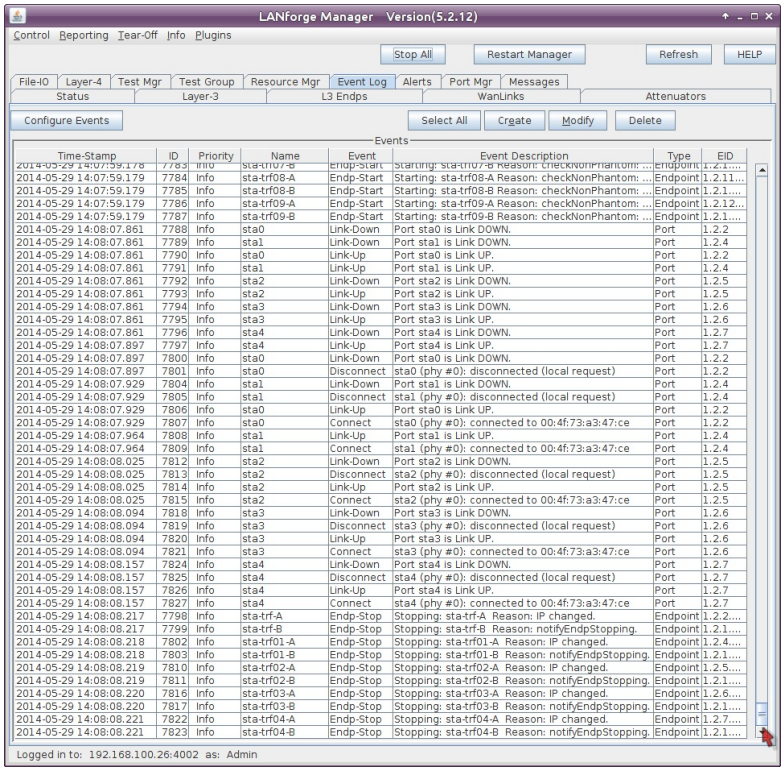
F. Check the LANforge Wireless Events window.You will see station connects and disconnects and failure notices. The messages shown in this picture are normal:



G. In the LANforge Messages window, you will see connection warnings. Connections will warn when their IP changes, and this is normal, as shown in this picture:



H. And in the Event Log tab, you can review more detailed link-up, link-down and connection events that occur with these station restarts:

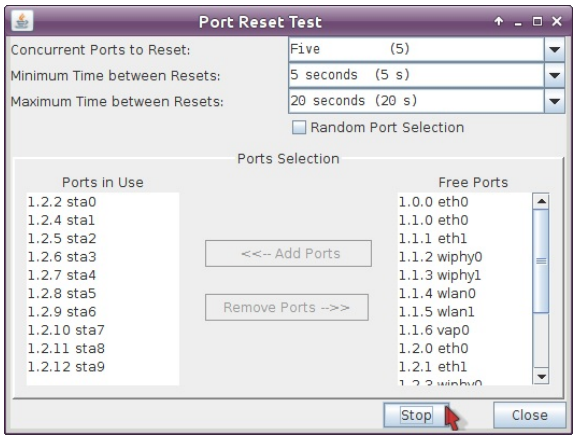


I. Use these monitoring sources to check for undesirable trends:

- Undesirable slowing trend in traffic
- stations that do no return
- DHCP failures or pool exhaustion
- or uncommonly long station re-association events

.

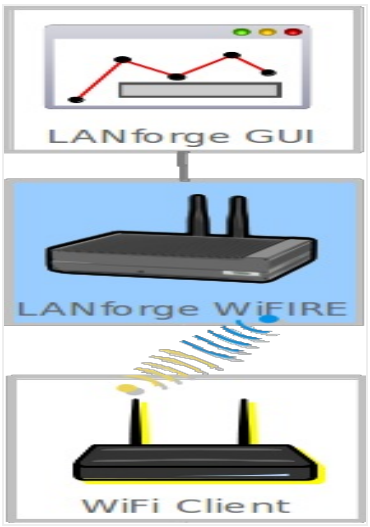
5. Halt the test when you are finished.



WiFi: Emulating Degraded Stations

Goal: Create a scenario where WiFi stations send out corrupted packets that can cause handshake and authentication failures. Learn techniques to capture and inspect packets to view corruption and scan log files to find indications of LANforge corruption injections.

We will learn to use the some WiFi packet corruption features to emulate malfunctioning station equipment. This consists of enabling the corruption features and looking for errors when stations attempt to associate. Part of this will include capturing packets and inspecting them. This scenario requires LANforge version 5.3.6, and a two-radio LANforge system with one radio set in monitor mode.

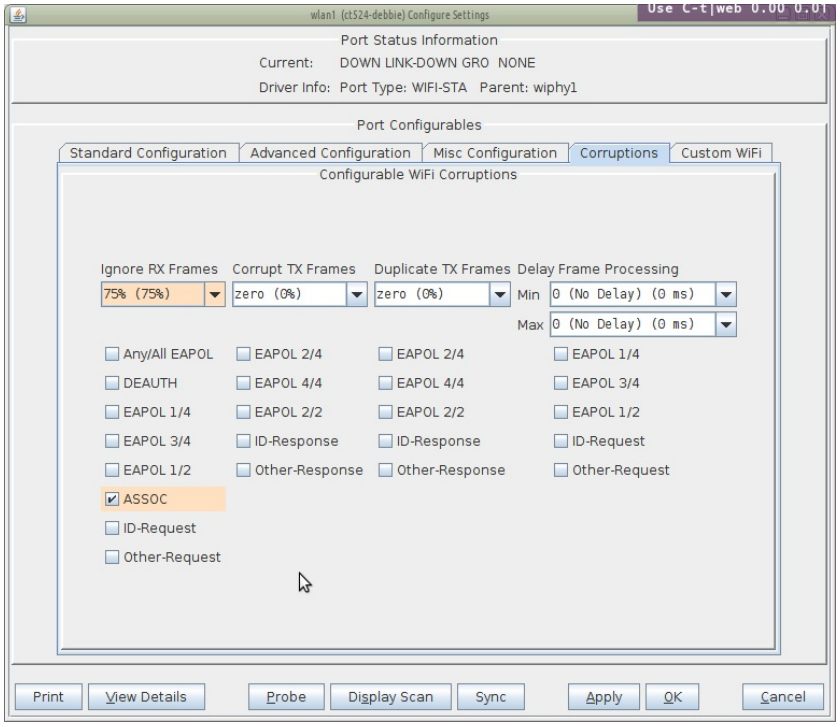


Open Authentication Test Scenario

We will begin with a basic open authentication test scenario with a single virtual station on a LANforge system connecting to an AP. (See Also: Generating Traffic for WLAN Testing) This test scenario also works for Captive Portal testing.

Create a Station with Corruptions

We will use the simplest form of corruption, ignored frames. Select the Port → Corruptions tab, and set Ignore RX Frames very high, like 75%. To limit this to association frames, select **ASSOC**.

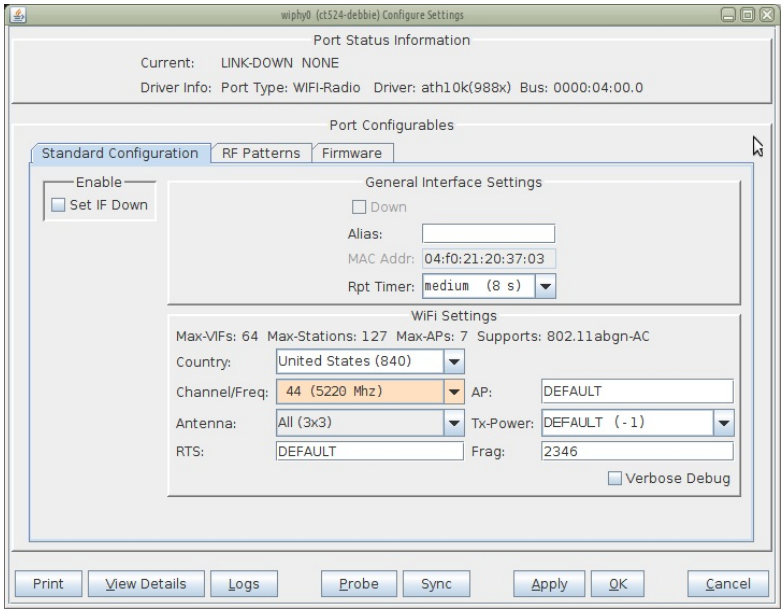


Associate a Station with Corruptions

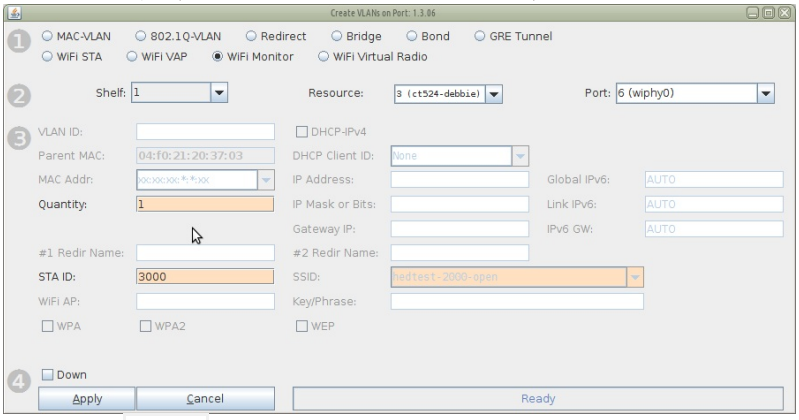
Introducing corruptions is simple. Watching the effects takes some effort. With aggressive association corruptions, the basic effect will appear as if your station takes an unreasonably long time to associate. Let's

set up Wireshark at different interfaces to understand better the kinds of traffic at play.

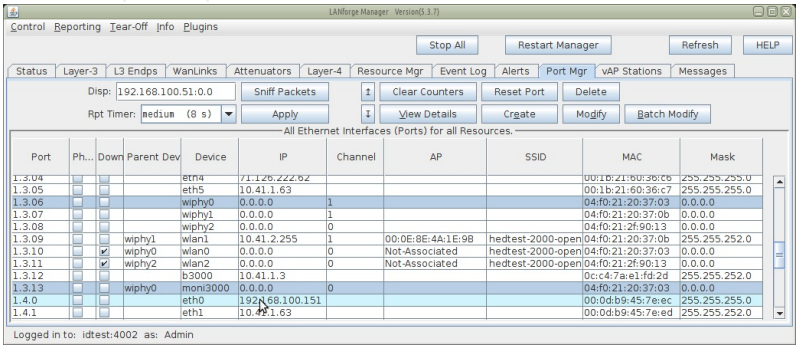
1. Select a radio for monitoring and set its channel to match the AP channel. If your AP is on channel 44, modify your LANforge wiphy monitor radio to also listen on channel 44. Our test AP is named hedtest-2000-open



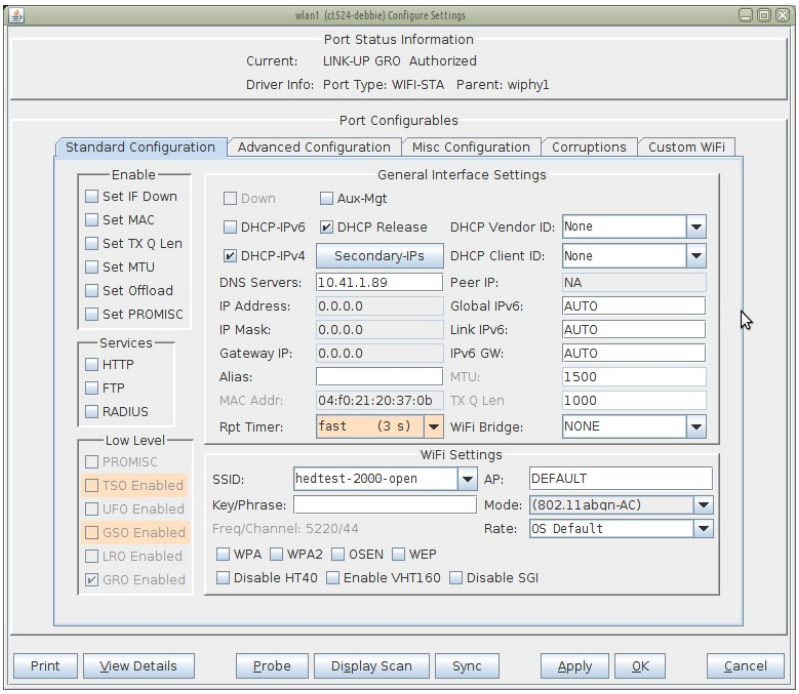
2. Create a **monitor** port (Port tab → Select Radio → Create → Monitor)



You will see the `moni3000` port below. The channel number will not display.

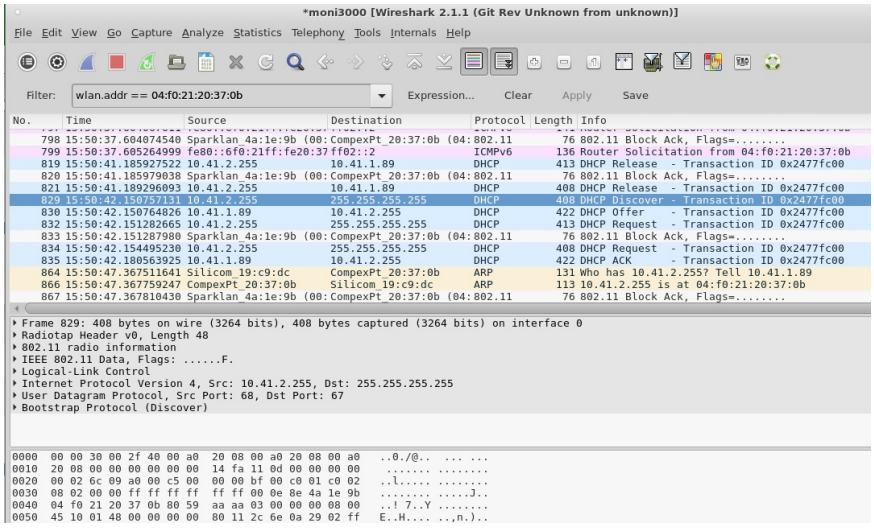


3. Create a station (wiphy1 → wlan1 will work) and associate it with hedtest-2000-open. Highlight the station in the Ports tab and bring it up.

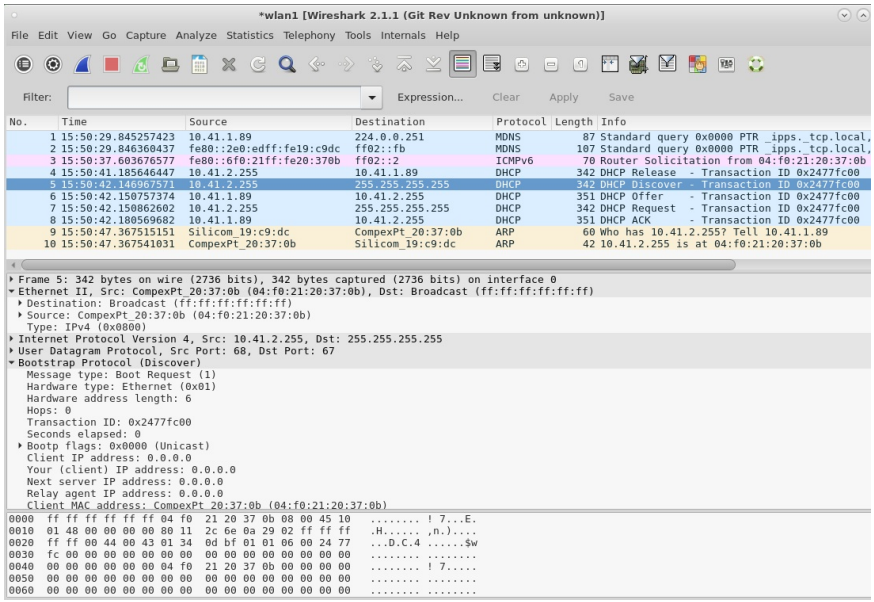


Watch Traffic

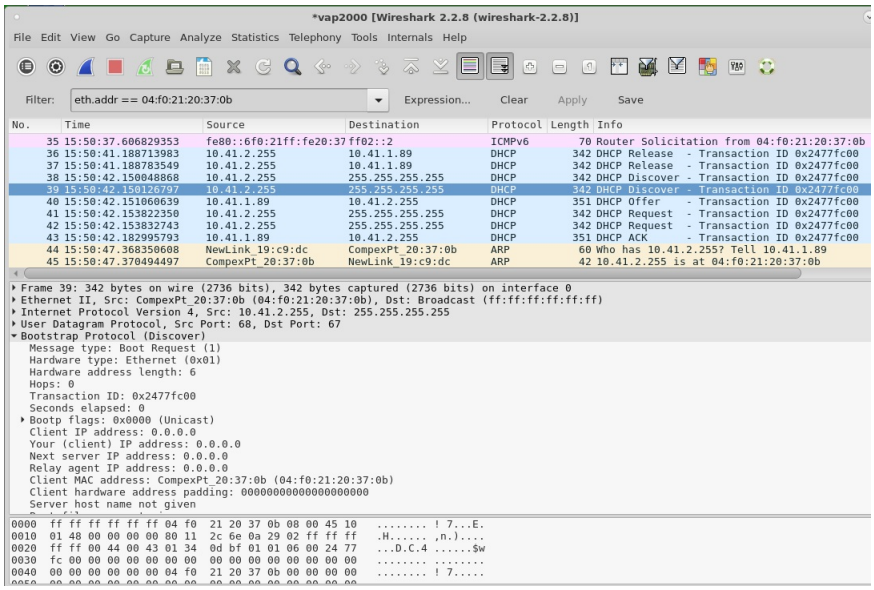
With 75% ignore the chances of the station actually associating are very port. On our monitor interface, we see that there are multiple discover and request packets.



On our station interface (wlan1), we see a different number of DHCP requests.



At our AP, we see repeated attempts.

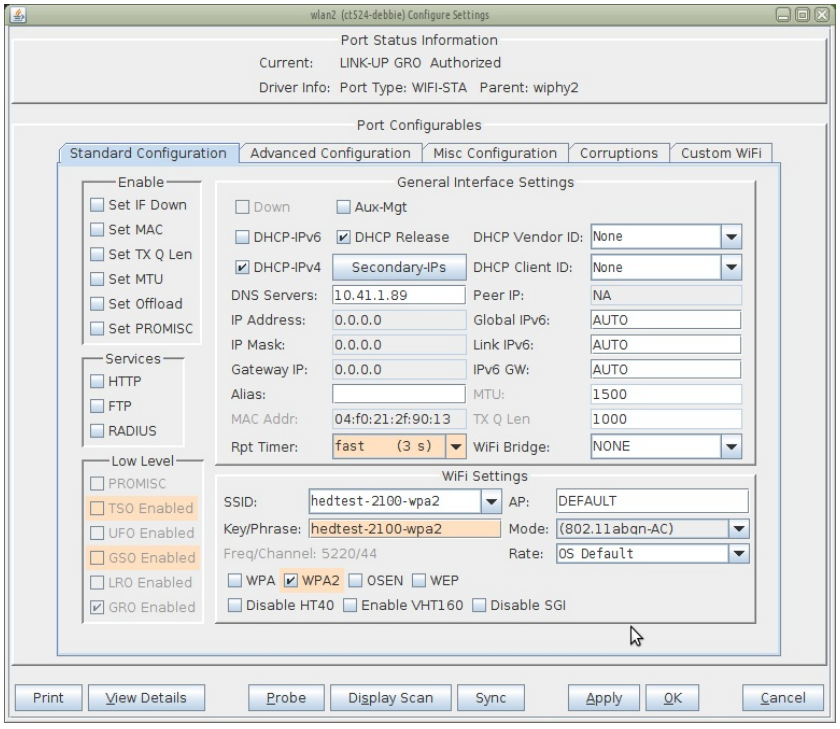


WPA2 Authentication Test Scenario

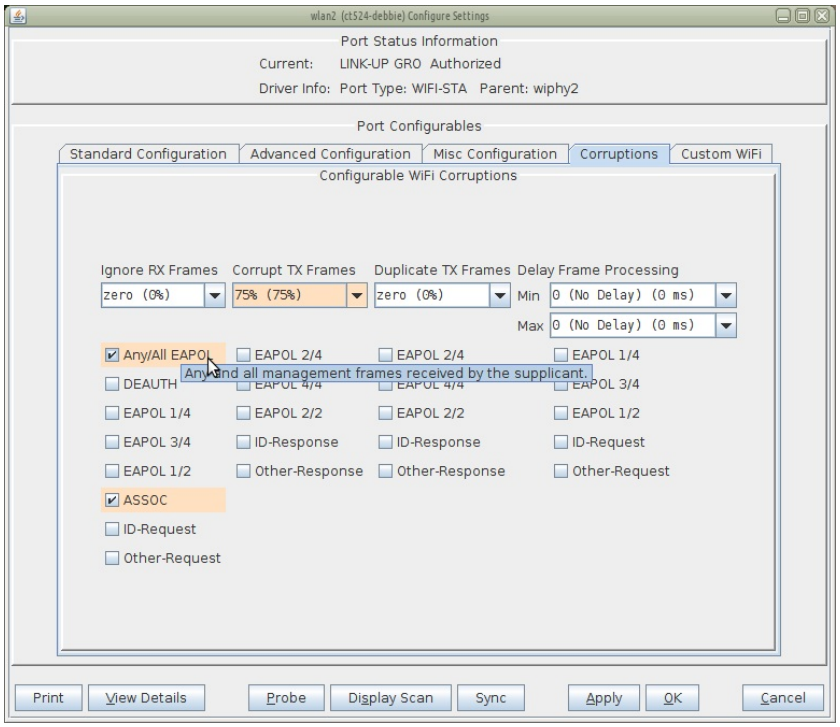
We will configure a station with WPA2 PSK encryption. We will also use Wireshark to decrypt the packets in order to see that they are corrupt.

Configure the Station

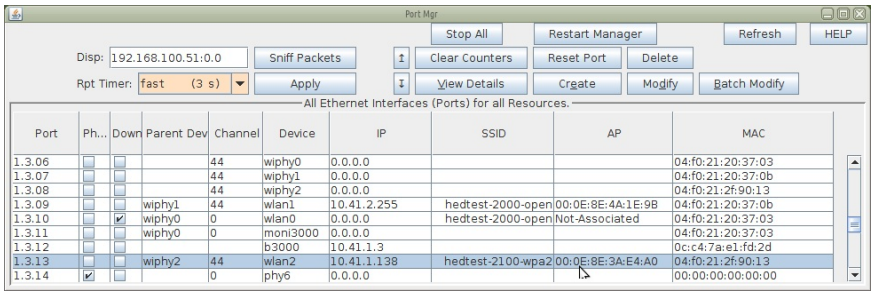
Configure station with WPA2 (See also: Test WiFi station upload throughput)



On the Corruptions tab, you can set stations:



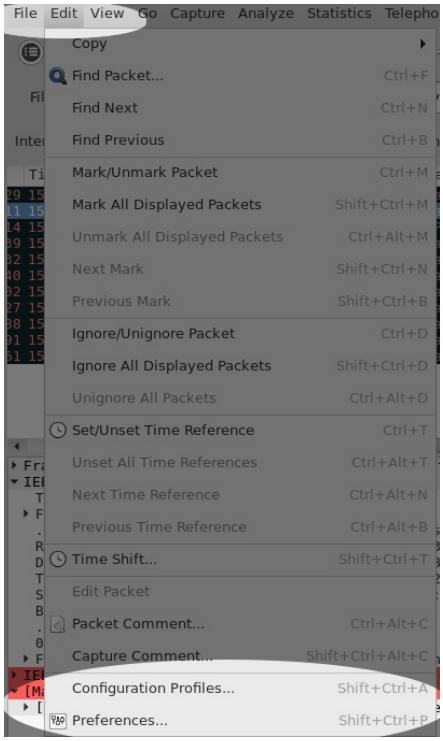
Highlight and activate the station:



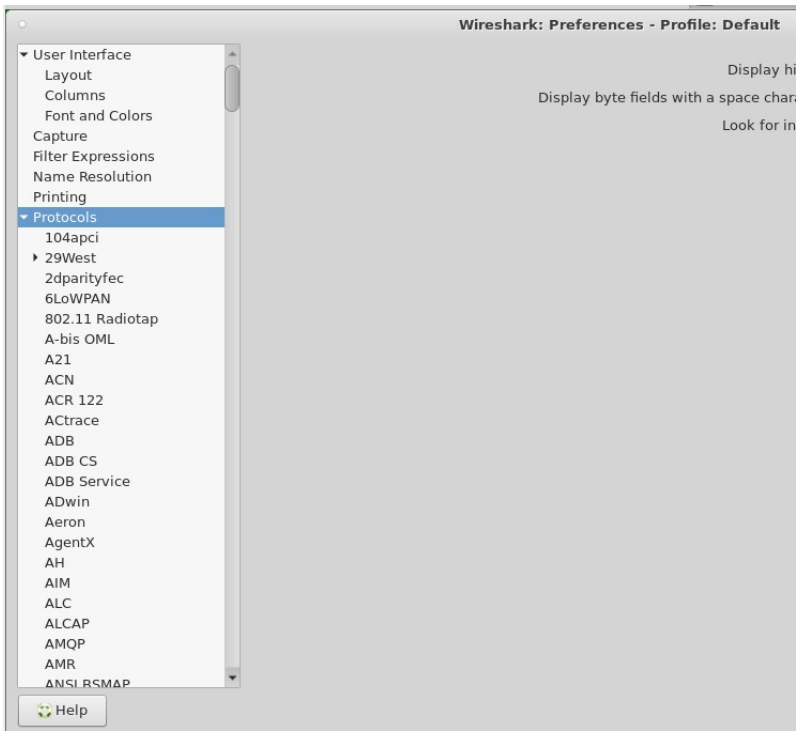
Setup Packet Decryption

We will not be able to inspect packets unless we configure Wireshark to decrypt the packets from this capture. Follow these steps:

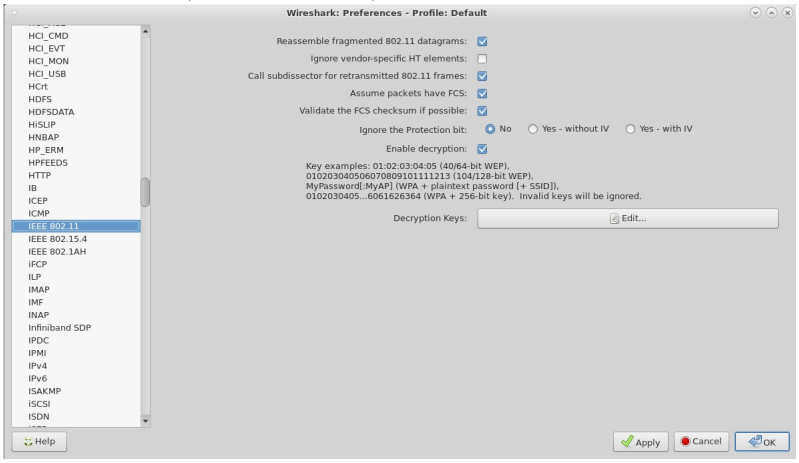
1. Wireshark Preferences (Edit → Preferences)



2. Protocol Preferences (Preferences → Protocols)



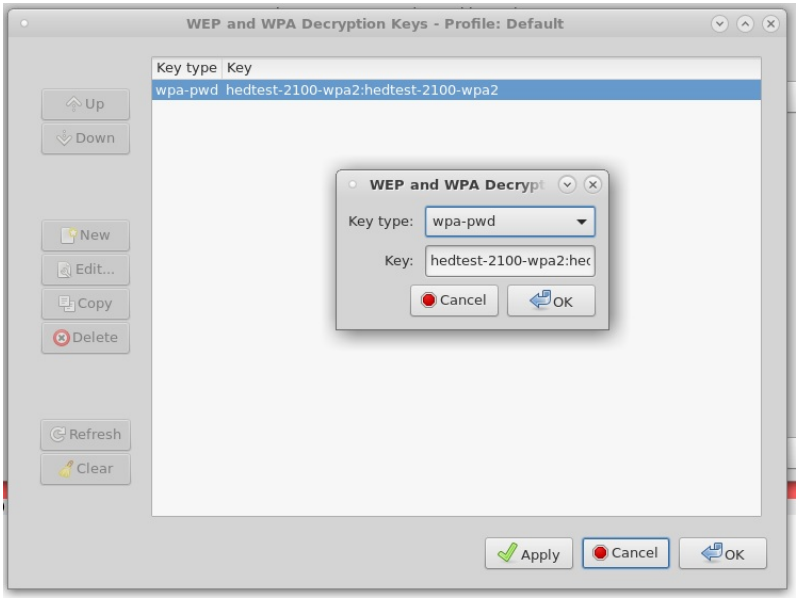
3. IEEE 802.11 Preferences (Protocols → IEEE 802.11)



   - Select **Reassemble Fragmented 802.11 datagrams**
   - Select **Call subdissector for retransmitted 802.11 frames**
   - Select **Assume Packets have FCS**
   - Select **Validate the FCS Checksum if possible**
   - Select **No** for Ignore the Protection bit
   - Select **Enable Decryption**

   Add the SSID and password to the list of Decryption Keys. Select **wpa-pwd**: and type in the string `$SSID:$PSK` to match your password from your AP:

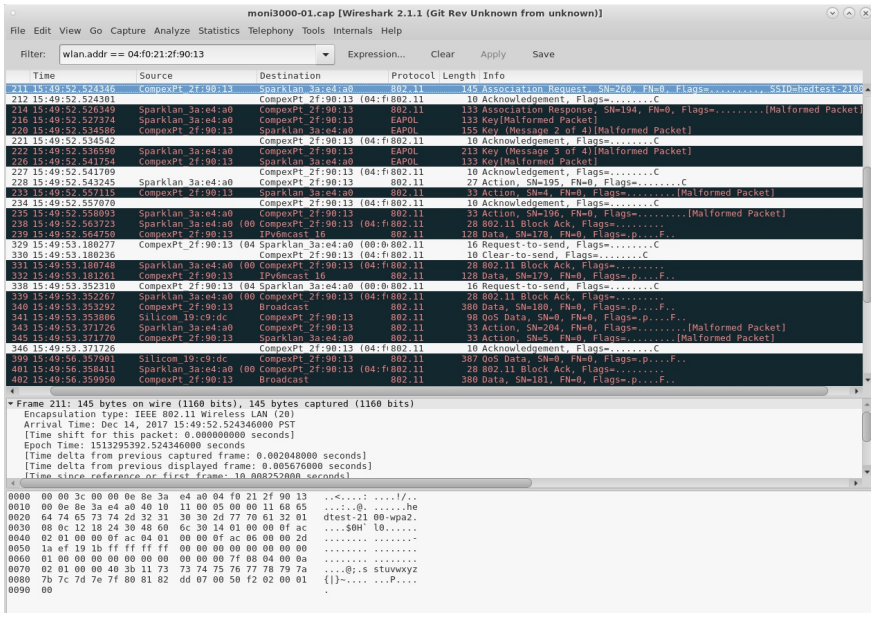
```
hedtest-2100-wpa2:hedtest-2100-wpa2
```



With our decryption enabled we can now inspect the captured packets:

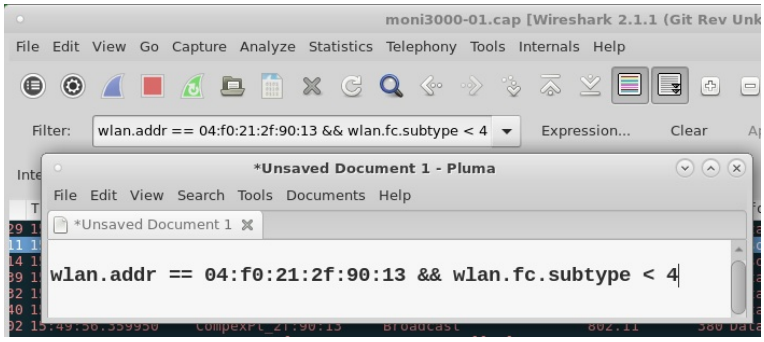
Packets from `moni3000` are filtered to show the station of interest using the display filter:

```
wlan.addr == 04:f0:21:2f:90:31
```

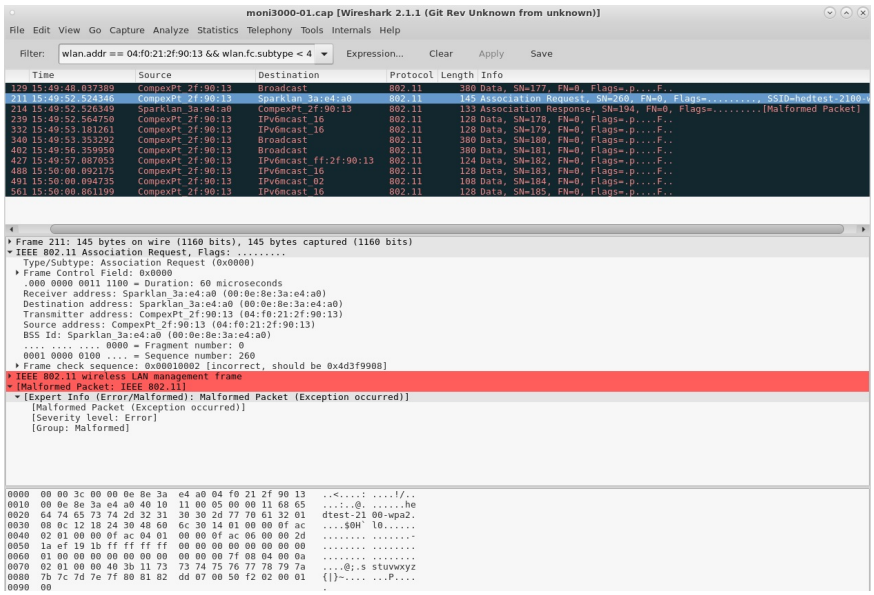


We can focus into the association by using a display filter:

```
wlan.addr == 04:f0:21:2f:90:31 && wlan.fc.subtype < 4
```

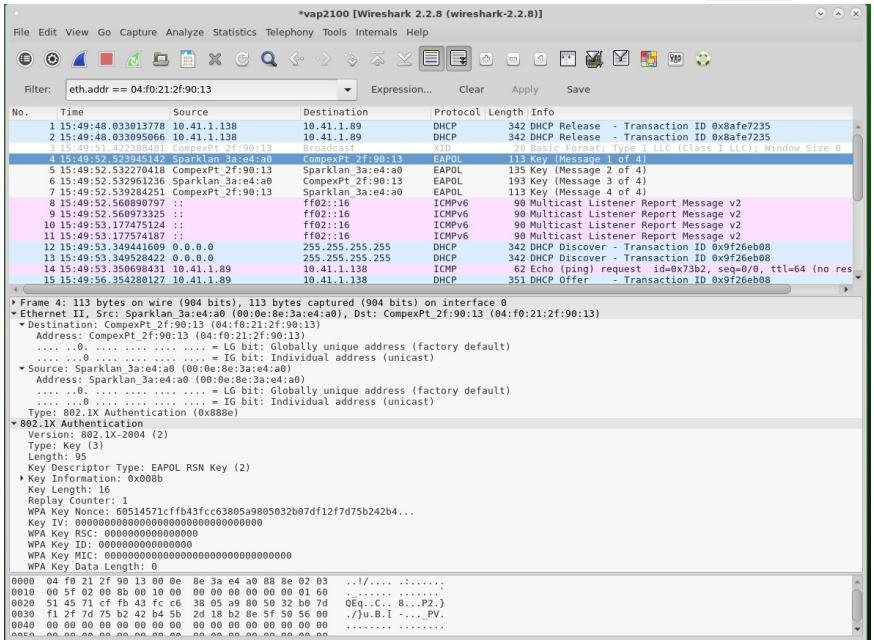


And the packets that form the association:



Notice how the AP only really gets the uncorrupted packets. What we hope will happen is that our WiFi drivers will discard corrupted packets before passing them up to userspace.

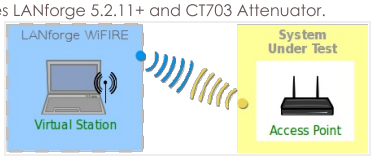
We did not need to decrypt packets from our AP (only possible if monitoring inside the AP) `vap2100:`

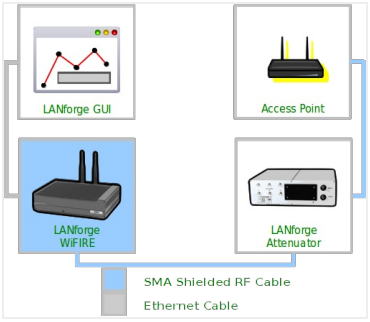


LANforge Scripted Attenuation of a WiFi Connection

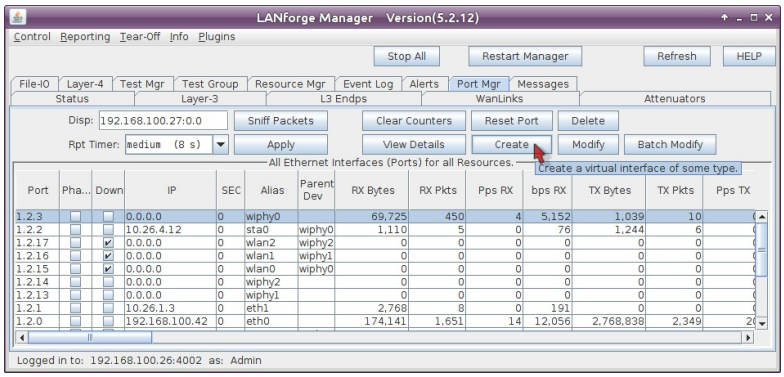
Goal: Operate the CT703 WiFi Attenuator with a script to see the effect on a Layer-3 connection.

The LANforge GUI provides scripting support for the CT703 Attenuator. This cookbook provides a basic example of how to graph connection throughput and signal strength with a single station connection. Use this example as a basis for building more complex attenuation scripts. Requires LANforge 5.2.11+ and CT703 Attenuator.

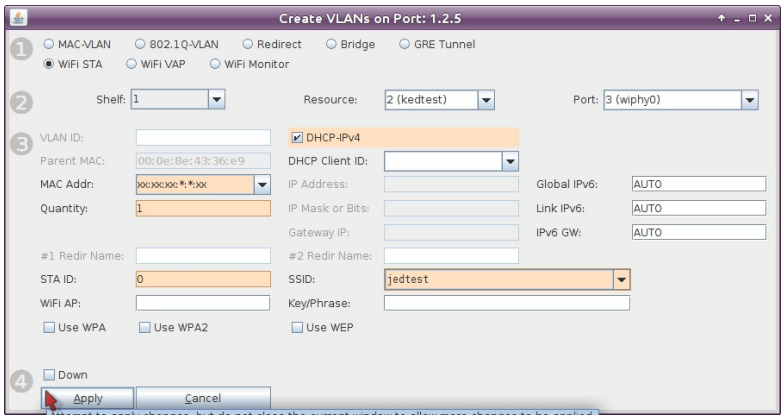




1. Create a WiFi station
   A. In the **Ports** tab, select wiphy0 and click **Create**

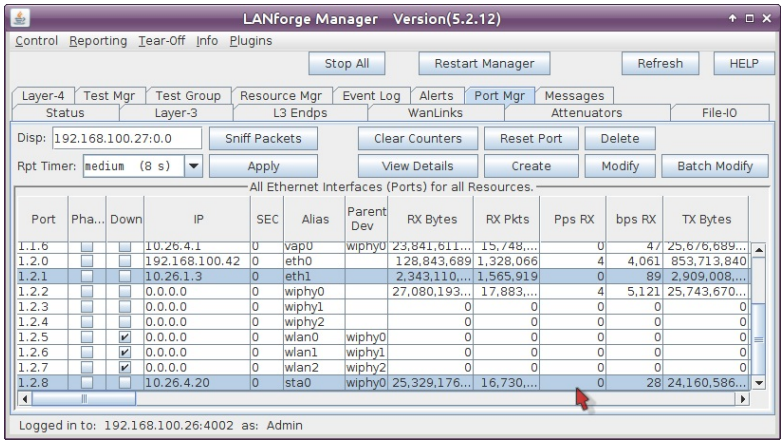


   B. In the **Create VLANS** window, craft one wifi station:



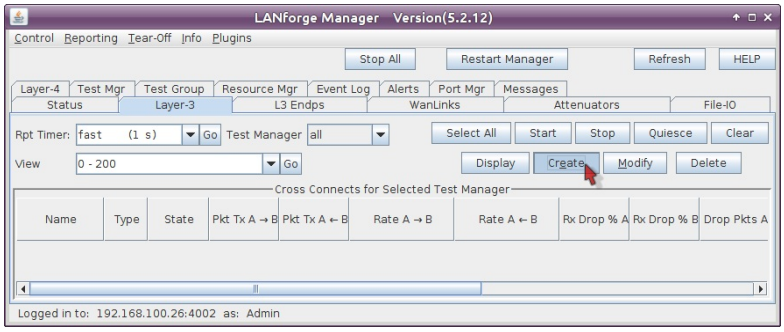
      A. Select WiFi STA
      B. For MAC address, choose `xx:xx:xx:*:*:xx`
      C. Select DHCP-IPv4
      D. Enter Quantity **1**
      E. Specify **0** for STA ID
      F. The example SSID for this cookbook is `jedtest`
      G. ...and then click **Apply**

   C. You will see a station created:

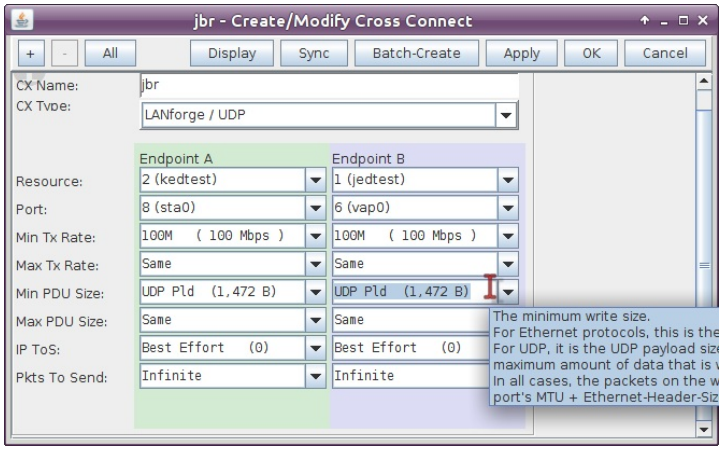


2. Create a connection to the Station

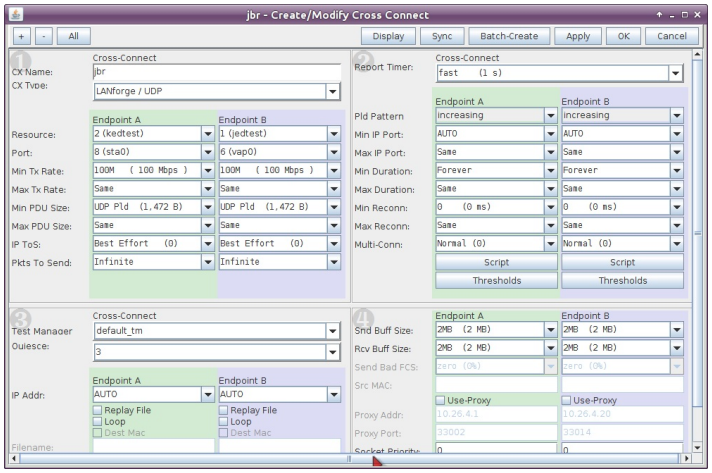
   A. In the **Layer-3** tab, click **Create**



   B. Create two-way station traffic

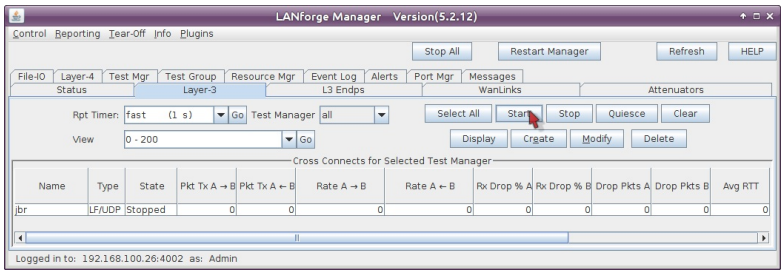


   A. This example connection is named jbr
   B. Connection Type is **LANforge / UDP**
   C. This example resource is **kedtest**, where our stations live
   D. The Endpoint A Port will be the station **sta0**,
   E. and the Endpoint B Port will be upstream of the ap, **eth1**.
   F. We'll set the Min Tx Rate for both sides to **100 Mbps**
   G. and set the PDU Size to `UDP Pld (1,472 B)`.
   H. Use the **+** button to expand the window to panel 4 and add **2MB** of buffers to the connection:

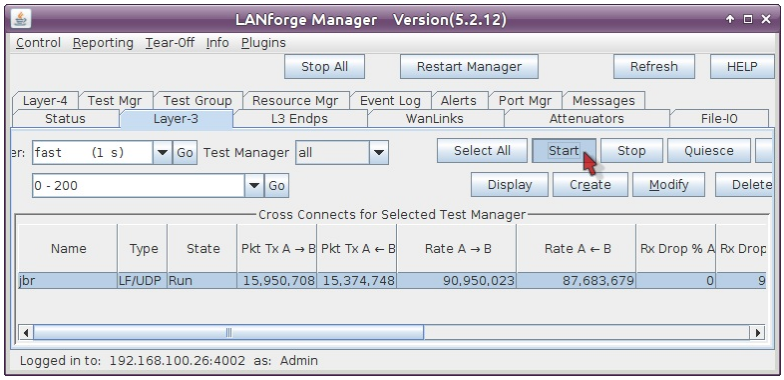


I. ...then click **OK**

C. You will see connection jbr in the Layer-3 tab now:

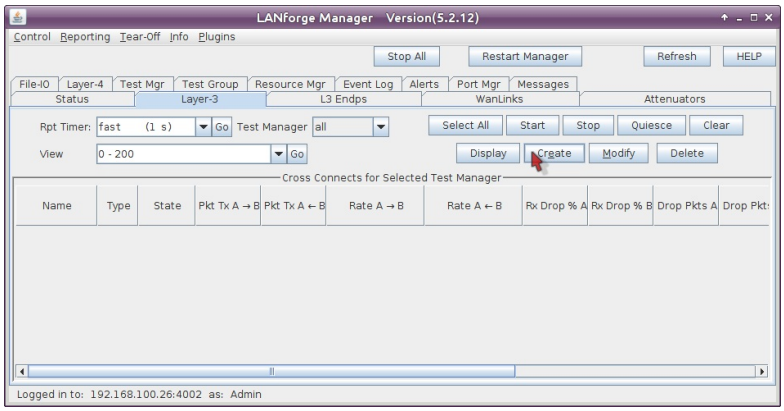


D. Begin traffic on this station by selecting it and clicking **Start**

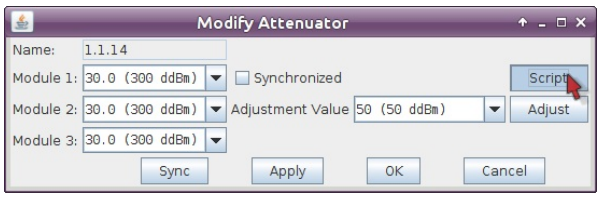


3. Create an Attenuator Script

A. In the **Attenuators** tab, select your attenuator and click **Modify**

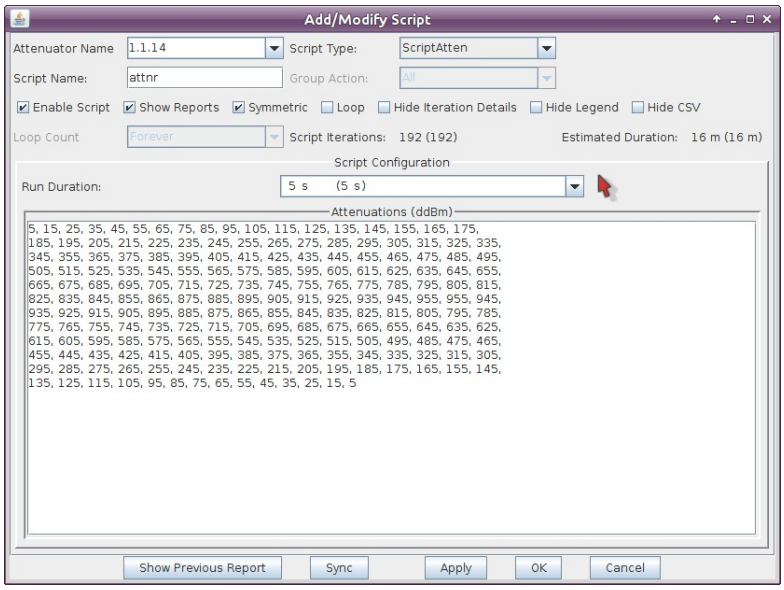


B. You will see the Modify Attenuator window. Click on the **Script** button.



C. The Add/Modify Script window opens. In the picture below you see a huge list of numbers (the attenuation levels). These numbers are not pre-populated. You can copy and paste them out of

this document or create a series with other commands listed below.



A. In the Script Type pulldown, select **ScriptAtten**

B. Type in the name for the script, this example is called `attnr`

C. Select Symmetric

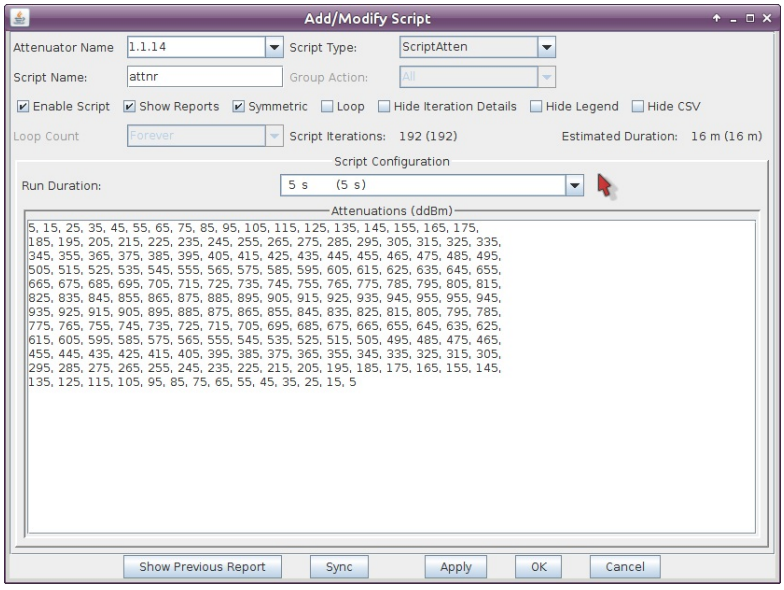
D. Choose **5 s** for Run Duration. This will run the Attenuator at each ddB value for this period of time.

E. Enter the following attenuation values for a 16 minute long test. This will send the attenuator from 5 to 955ddB and back:

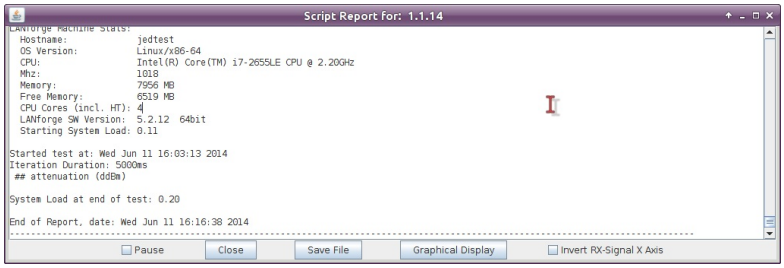
F.
```
5, 15, 25, 35, 45, 55, 65, 75, 85, 95, 105, 115, 125, 135, 145, 155, 165, 175,
185, 195, 205, 215, 225, 235, 245, 255, 265, 275, 285, 295, 305, 315, 325, 335,
345, 355, 365, 375, 385, 395, 405, 415, 425, 435, 445, 455, 465, 475, 485, 495,
505, 515, 525, 535, 545, 555, 565, 575, 585, 595, 605, 615, 625, 635, 645, 655,
665, 675, 685, 695, 705, 715, 725, 735, 745, 755, 765, 775, 785, 795, 805, 815,
825, 835, 845, 855, 865, 875, 885, 895, 905, 915, 925, 935, 945, 955, 955, 945,
935, 925, 915, 905, 895, 885, 875, 865, 855, 845, 835, 825, 815, 805, 795, 785,
775, 765, 755, 745, 735, 725, 715, 705, 695, 685, 675, 665, 655, 645, 635, 625,
615, 605, 595, 585, 575, 565, 555, 545, 535, 525, 515, 505, 495, 485, 475, 465,
455, 445, 435, 425, 415, 405, 395, 385, 375, 365, 355, 345, 335, 325, 315, 305,
295, 285, 275, 265, 255, 245, 235, 225, 215, 205, 195, 185, 175, 165, 155, 145,
135, 125, 115, 105, 95, 85, 75, 65, 55, 45, 35, 25, 15, 5
```

G. ...click **OK**

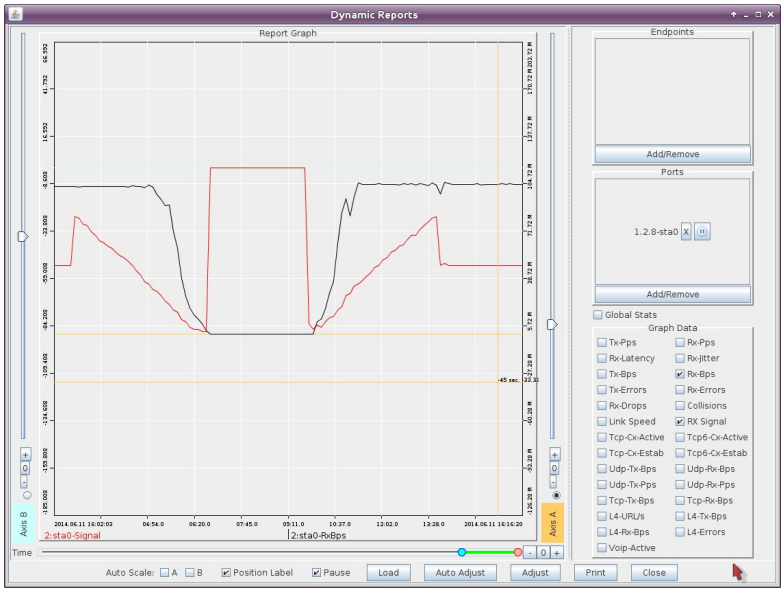
D. In the Attenuators tab, you will notice that your attenuator now reads **Enabled** in the Script column. Select the attenuator and click the **Start** button.



E. The window Script Report will appear. This monitors the script behavior of the attenuator script.



F. To monitor the progress of traffic and signal as it progresses, you will go to the Ports tab. Click on **sta0**, right click and select Dynamic Display. The Dynamic Reports window will open. Choose the following selections to refine the display:



A. Unselect Tx-Bps

B. Select Rx-Bps

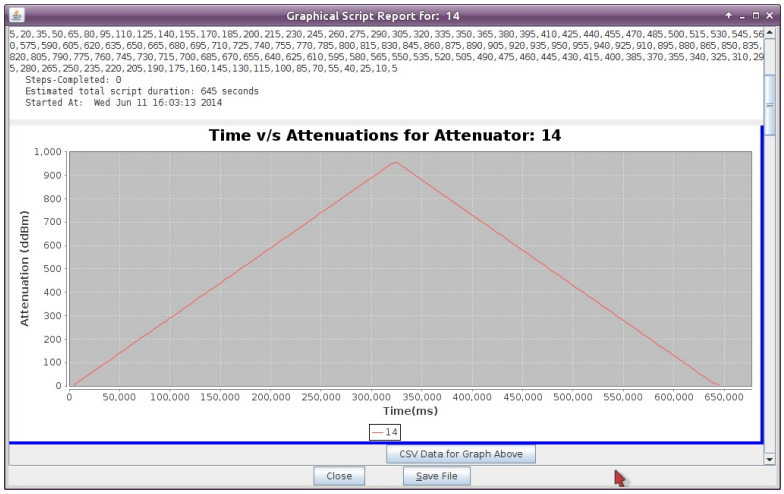
C. Select Rx-Signal

D. Drag the label **sta0-signal** to the left side of the X axis

E. Click **Adjust** and set Maximum Time to **30 min**, click **OK**

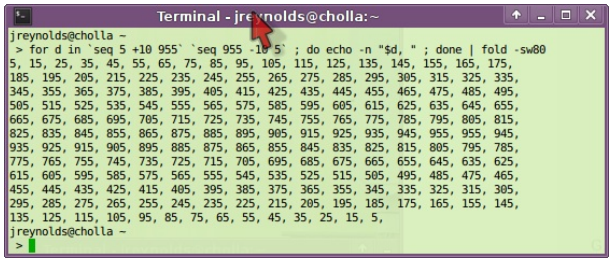
F. Click **Auto Adjust**

G. At the end of the attenuator script run, the Layer-3 connection will still be running, but the Script Report window will say `End of Report`. At this time, click on **Graphical Display**. You will see a graph of the attenuation level over time.



4. Creating attenuation sequences

5. It is relatively simple to generate a sequence of numbers right from a Terminal window (on Linux). Below are some examples:



A. The 10 ddB step series we used in this example:
`for d in `seq 5 +10 955` `seq 955 -10 5` ; do echo -n "$d, " ; done | fold -sw80`

B. A 5 ddB step series, as precise as the CT703 can run:
`for d in `seq 0 +5 955` `seq 955 -5 0` ; do echo -n "$d, " ; done | fold -sw80`

C. A 15 ddB step series from full attenuation to 25 ddB:
`for d in `seq 955 -15 25` ; do echo -n "$d, " ; done | fold -sw80`

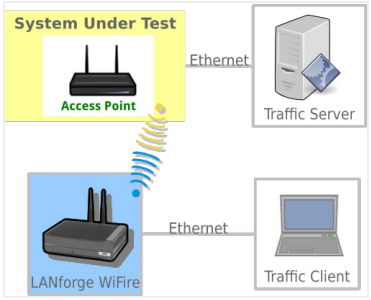
D. A sawtooth ranging from 950 to 0 ddB:
`i=0; while [ $i -lt 100 ]; do echo -n $(( $i%11 * 95))", "; i=$[ $i+1 ]; done | fold -sw80`

Generate WiFi Traffic from an External Connection

Goal: Create a bridged WiFi station that redirects traffic from an ethernet port.

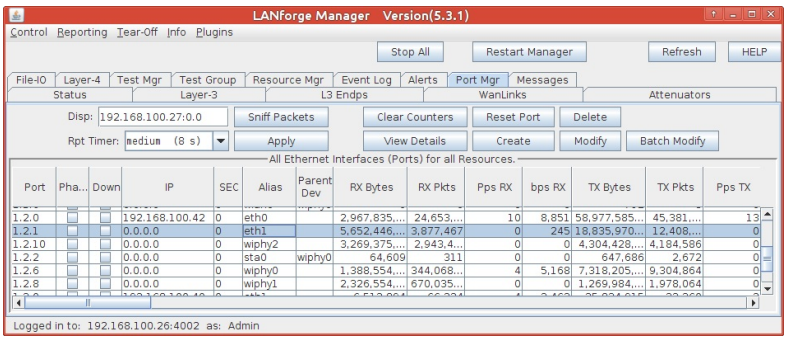
It is possible to use a system on a wired LAN to generate traffic through a LANforge virtual WiFi station. This example will show a Windows PC driving traffic across a LANforge CT520 STA to the associated AP. This is a useful scenario for driving custom traffic between traffic testing software endpoints distinct from the LANforge system. A virtualized Windows PC will work with this scenario.

The primary technique in this cookbook uses a technique where the MAC address of the WiFi STA is spoofing the address of the Windows PC. It is also possible to use this technique with IPv4 addresses and not MAC addresses. However, using MAC addresses will allow IPv6 traffic to work.

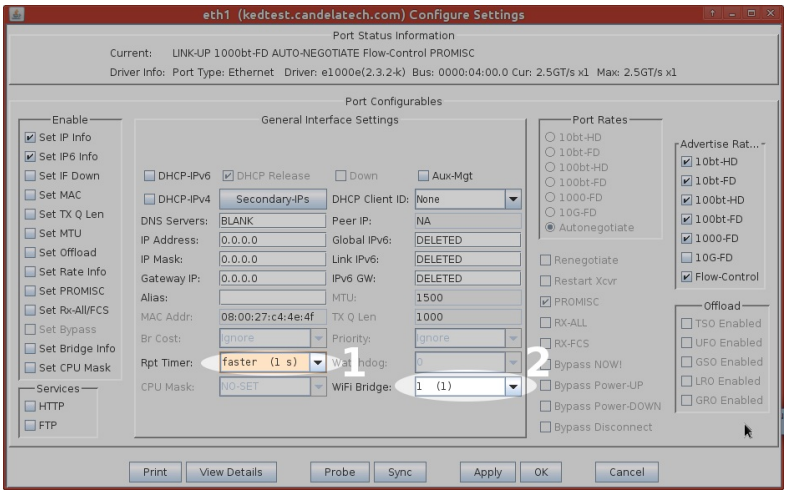


1. Find the MAC Address of the Windows PC.
   A. We will use MAC address `08:00:27:c4:4e:4f`. This will be used when you configure the WiFi STA on the LANforge machine.
   B. Please set the IP address of the interface if it is not yet set.

   For more information see Windows IP Addresses
2. Configure your LANforge Wireless Station. We will be configuring the MAC addresses of a station to be the MAC address of the Windows PC we just found.
   A. In the Ports tab, double-click on the ethernet port on the same switch as the Windows PC (eth1 in this example).



B. Configure the port MAC address and WiFi-Bridge settings to be:



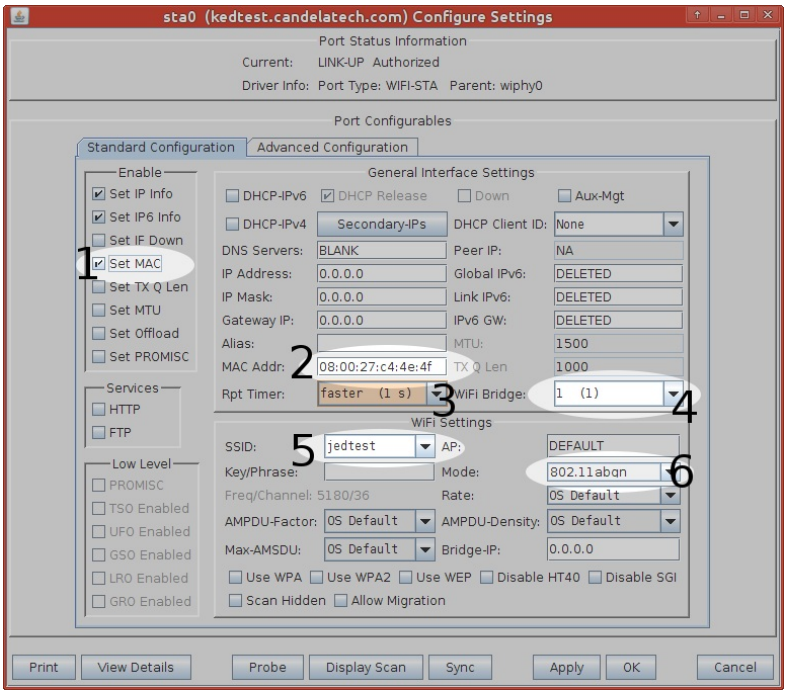
A. Do not set the mac address or the ip address of the port

B. Set Rpt Timer to `faster (1 s)`

C. Select `1` for WiFi Bridge

D. Click the **OK** button

C. Configure a WiFi station. This example will be connecting to a WiFi AP with the SSID jedtest.



A. Enable **Set MAC**

B. Use `08:00:27:c4:4e:4f` for the MAC Addr

C. Set Rpt Timer to `faster (1 s)`

D. Select `1` for WiFi Bridge

E. Enter `jedtest` for the SSID

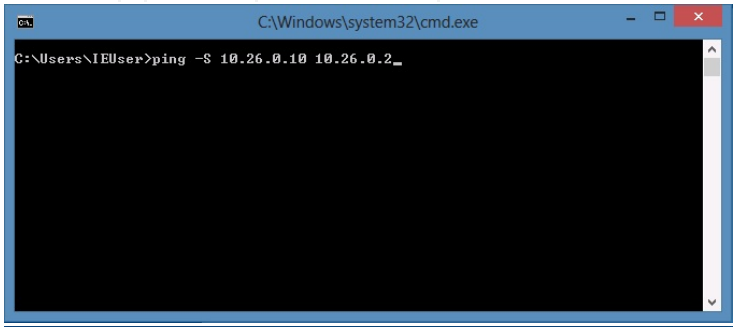
F. Select `802.11abgn` for the Mode

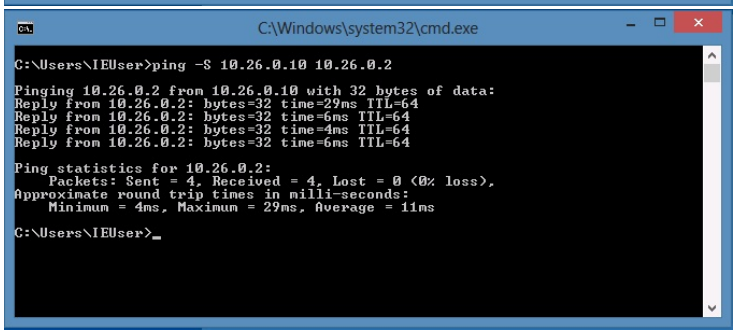
G. Click the **OK** button

D. (Note: these MAC addresses will remain persistent even through a LANforge Manager restart. To restore the physical mac addresses, you need to reboot the LANforge machine or use ethtool to find the physical hardware address.)

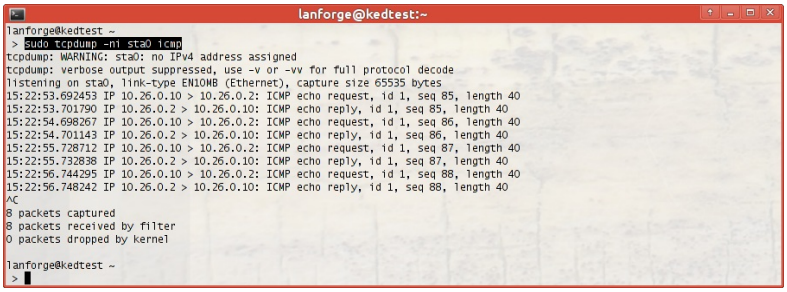
3. Sending and Validating Traffic

A. Use ping on the Windows machine to reach 10.26.0.2, the AP machine. To ping from a specific interface, use the `-S` switch like so: `ping -S 10.26.0.10`



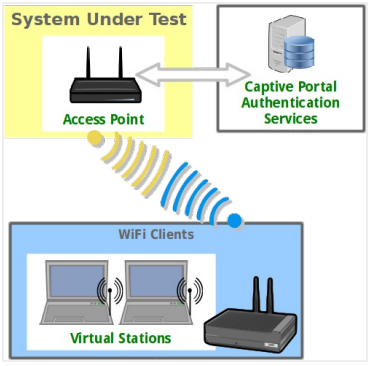


B. On LANforge, use tcpdump to view packets traversing the Station interface, `sudo tcpdump -ni sta0 icmp`



WiFi Captive Portal Bot (portal-bot)

Goal: Execute a battery of of captive portal logins from virtual wifi stations using the newer portal-bot.pl script.



Public access open WiFi service is often gated with a web sign-on form (a captive portal). LANforge virtual stations can emulate sign-in to the captive portal using the `portal-bot.pl` script. This script is by necessity incomplete because many captive portals have different behaviors and login form requirements. With this script, you provide a bot plugin that bridges the gap. This cookbook will coach you through a basic portal-bot integration and then you will create ten stations that authenticate through a captive WiFi portal.

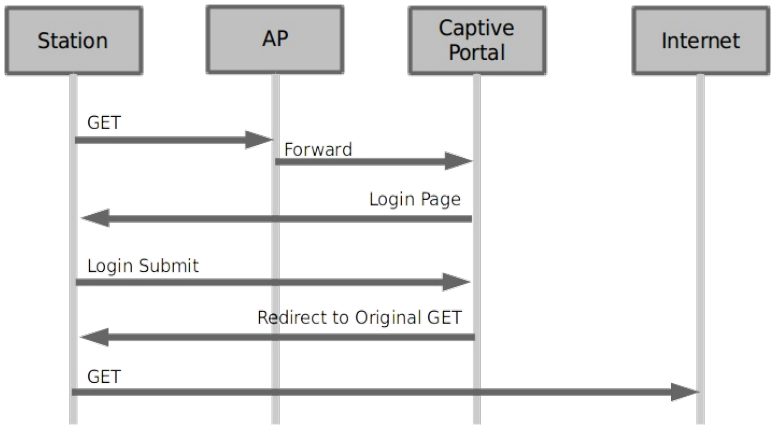
In this example, we will be testing agains a simple LAMP server on the upstream side of the AP. Do no use your LANforge server as the LAMP server because the routing will be difficult. In this chapter, a LAMP server is at 10.26.1.254, and there is an /etc/hosts entry for basic-portal to that address.

Basic Interactions of a Captive Portal

The basic order of operations of a captive portal are summarized in these steps:

1. A WiFi station accesses the LAN and is assigned a DHCP address.
2. The AP redirects any DNS and HTTP(s) request from the station. It returns either

    - a login page directly
    - a 301-Redirect to the login page
3. The station user submits this form. This form knows where to submit itself to, but it is possible that the form does not submit to the same address or service that it came from.
4. A successful authentication provides one of these responses:
    - The originally requested page, either as a 301-Redirect or as a proxied result.
    - A portal-div providing a logout or service menu and the original content inside.
    - A redirect page that uses javascript or meta-refresh mechanisms to tell the browser to reload the originally requested page.



Configuring a Demo Captive Portal

Provide Login/Logout pages

If you wish to set up a login and logout page on an Apache/PHP server to test with, you can copy the below files to the `/var/www/html` directory on the LAMP server.

login.php:

```
<!DOCTYPE html !>
<?php
$valid = true;
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
   /* custom error reporting, see get_explanation */
   if (!array_key_exists('username', $_POST)) {
      header("HTTP/1.1 400 Bad Request");
      header("X-err-no: 9400");
      header("X-err-msg: missing username");
      $valid = false;
   }
}
?>
<html>
<head>
   <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<?php if($valid) { ?>
   <title>Login</title>
<?php } else { ?>
   <title>Bad Request</title>
<?php } ?>
</head>
<body>
   <?php if ($_SERVER['REQUEST_METHOD'] == 'POST') { ?>
      <?php if(!$valid) { ?>
         <h1>Bad Request</h1>
      <?php return; } ?>

      <?= $_POST['username'] ?> access granted.
   <?php } else { ?>
      <form method="post" action="">
         Login:<input type="text" name="username" value="" /><br />
         <input type="submit" name="login" value="Login" />
      </form>
   <?php } ?>
</body>
</html>
```

Provide a Redirect in lieu of Portal Capture

Getting a redirect to the login page does not have to be very complex. The portal-bot script will first start off requesting whatever URL you wish, so request `http://basic-portal/start`. Here is an Apache configuration line to redirect that URI to `login.php`:

httpd.conf

```
<Location /start>
   Redirect /start /login.php
</Location>
```

After adding this redirect, restart your Apacher service using this command:

```
sudo apachectl configtest && sudo apachectl restart
```

Testing your redirect

You can use the command `curl -sqv http://basic-portal/start` to test out the redirect you just created.

```
 > curl -sqv http://basic-portal/start
* STATE: INIT => CONNECT handle 0x25bd9e8; line 1034 (connection #-5000)
* Added connection 0. The cache now contains 1 members
* STATE: CONNECT => WAITRESOLVE handle 0x25bd9e8; line 1071 (connection #0)
*   Trying 10.26.1.254...
* bind-local, addr: (nil)  dev: (nil)
* STATE: WAITRESOLVE => WAITCONNECT handle 0x25bd9e8; line 1151 (connection #0)
* Connected to basic-portal (10.26.1.254) port 80 (#0)
* Marked for [keep alive]: HTTP default
* STATE: WAITCONNECT => DO handle 0x25bd9e8; line 1229 (connection #0)
> GET /start HTTP/1.1
> User-Agent: curl/7.41.0-DEV
> Host: basic-portal
> Accept: */*
>
* STATE: DO => DO_DONE handle 0x25bd9e8; line 1314 (connection #0)
* STATE: DO_DONE => WAITPERFORM handle 0x25bd9e8; line 1441 (connection #0)
* STATE: WAITPERFORM => PERFORM handle 0x25bd9e8; line 1454 (connection #0)
* HTTP 1.1 or later with persistent connection, pipelining supported
< HTTP/1.1 302 Found
< Date: Fri, 04 Sep 2015 22:52:53 GMT
* Server Apache/2.4.7 (Ubuntu) is not blacklisted
< Server: Apache/2.4.7 (Ubuntu)
< Location: http://basic-portal/login.php
< Content-Length: 290
< Content-Type: text/html; charset=iso-8859-1
<
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>302 Found</title>
</head><body>
<h1>Found</h1>
<p>The document has moved <a href="http://basic-portal/login.php">here</a>.</p>
<hr>
<address>Apache/2.4.7 (Ubuntu) Server at basic-portal Port 80</address>
</body></html>
```

Using the Portal Bot bash script

Before we get straight to working with `portal-bot.pl`, let's see how it is used. Your LANforge installation has an example script called `portal-bot.bash-example` for you to copy and modify. This script is intended for you to login and logout separately. The LANforge manager will call `portal-bot.pl` differently when building up the station or tearing down the station, these actions are similar:

> **`./portal-bot.bash`** will log your station in

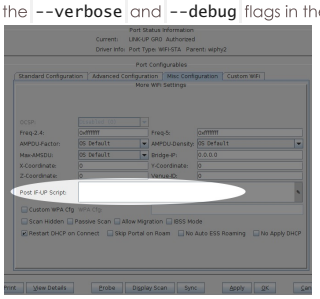
> **`./portal-bot.bash --logout`** will log your station out

Inside the bash script

The `portal-bot.bash` script is for exercising your `portal-bot.pl` script options from the command line while you develop with it. This is very close to the values you will place in the Ports→Misc/Post IF-UP field.

Switches you won't use in the GUI

You will never place the `PBOT_NOFORK` option in the Ports→Misc/Post IF-UP field because that will interrupt the processing of the LANforge Manager process. You will also never place `$*` in that field, either. You can place the `--verbose` and `--debug` flags in there, but it can fill your disk with log output more quickly.



Below is an example `portal-bot.bash` script with `\` line-continuation characters formatted for clarity:

```
PBOT_NOFORK=1 ./portal-bot.pl \
   --dev          sta100                    \
   --bot          bp.pm                     \
   --ip4          10.26.2.30                \
   --dns          192.168.100.1             \
   --mgt          /dev/null                 \
   --delays       0,1,3                     \
   --user         "bob"                     \
   --pass         "secret"                  \
   --ap_url       "http://basic-portal/"    \
   --start_url    "http://basic-portal/start"  \
   --login_form   "login.php"               \
   --login_action "login.php"               \
   --logout_url   "logout.php"              \
   --verbose --debug $*
```

Below is the same script using short switches:

```
PBOT_NOFORK=1 ./portal-bot.pl \
   -i    sta100                    \
   -b    bp.pm                     \
   --ip4 10.26.2.30                \
   --dns 192.168.100.1             \
   --mgt /dev/null                 \
   --delays 0,1,3                  \
   -u    "bob"                     \
   -p    "secret"                  \
   -a    "http://basic-portal/"    \
   -s    "http://basic-portal/start"  \
   -n    "login.php"               \
   -o    "login.php"               \
   -t    "logout.php"              \
```

```
  -v -d $*
```

Using the portal-bot.bash command on the command-line:

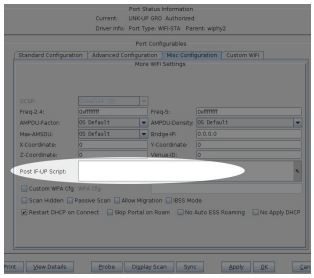
A common misconception is thinking that `$*` is a command-line argument. It is only used in bash scripts. Do not put `$*` on the command-line.

```
PBOT_NOFORK=1 ./portal-bot.pl -i sta100  -b bp.pm --ip4 10.26.2.30 \
        --dns 192.168.100.1 --mgt /dev/null -u "bob" -p "secret" \
        -a "http://basic-portal/"  -s "http://basic-portal/start"
        -n "login.php" -o "login.php" -t "logout.php" -v -d
```

Using the `portal-bot.pl` perl script

Tips:

- First thing to do: edit a copy of that script and adjust it for your station device and it's IP address.
- Add `-d` to add more debugging messages. That makes `dbg()` statements print.
- Add `--print` after you get the script to work. This will print out the format of the arguments useful for putting the statements into the GUI Ports→Misc/Post IF-UP field.



The first six arguments are provided by LANforge when you use `portal-bot.pl` with a station. You want to populate these in your bash script, but not in the Post IF_UP field.

PBOT_NOFORK
: This environment variable tells the `portal-bot.pl` script to not fork. **Use it only when developing.** Omitting this is normal and allows for multi-processing of web requests from LANforge.

-i
: station name

--bot
: The bot plugin you provide

--ip4
: The IP of the station. This script is useless if there has been no DHCP lease.

--ip6
: Use `''` for no IPv6 address.

--dns
: The DNS addresses provided from the DHCP lease

--mgt
: The FIFO that signals the LANforge server. You don't use it when testing.

The second set of arguments describe your own AP environment:

--user | -u
: portal user name

--pass | -p
: portal user password

--ap_url | -a
: A string to prepend to URLs when talking to the AP. Not necessary, but if you don't use it, you have to provide fully qualified URLs to --login_form, --login_action, and --logout_form.

--start_url | -s
: The first URL requested from the AP, this should provide either a login page or a redirect to a login page. If you get your destination page (like, if you request baidu.com and actually get it), your station has probably not been logged out from the captive portal.

--login_form | -n
: This is what you request to get a login form. Often it is returned in the redirect, but sometimes you cannot get a cookie assignment if you do not request it specifically.

--login_action | -o
: Submit your login credentials to this URL.

--delays
: Comma separated list of seconds to delay at certain points:

1. `$::delays[0]` Used to delay the very first 'start_url' GET request
2. `$::delays[1]` Used to delay the first POST request in 'submit_login'
3. `$::delays[2]` Used to delay the 'submit_logout' request.
4. `$::delays[3+]` Your bot can utilize further delays if you specify

You may specify skips by adding a zero: `--delays 1,0,2`

You may specify a random time by using 'random': `--delays 1,random,2`

You may specify just one time for all delays: `--delays 2`

You may specify a random range: `--delays 3-20,4-25`

--logout_form | -t
Submit to this URL to log out of the captive portal

-v -d
Verbose and debug output, respectively.

--print
Skips process and prints out formatted arguments.

$*
Expands to all remaining shell arguments

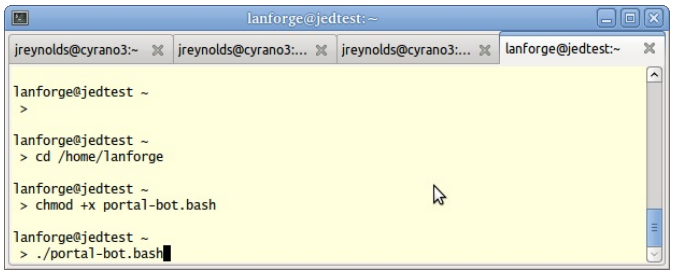
We will connect to our LANforge system*. You want to copy this file to your own `./portal-bot.bash` file, edit it and then make it executable.

ℹ * You can connect via VNC, PuTTY or other SSH client.

ℹ Use **`chmod +x portal-bot.bash`** to make your script executable.

Now let's see how to use this script with station **sta100**. Run the commands:

```
$ cd /home/lanforge
$ chmod +x portal-bot.bash
$ ./portal-bot.bash
```



You will see a lot of output, it will show the contents of the web pages it finds.

```
INTERPRET given HTTP/1.1 200 OK
Date: Fri, 04 Sep 2015 22:57:12 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.11
Vary: Accept-Encoding
Content-Length: 192
Content-Type: text/html

<!DOCTYPE html !>
<html>
<head>
   <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
   <title>Login</title>
</head>
<body>
        bob access granted.
   </body>
</html>found ref to data?         bob access granted.result for sta100: OKportal-bot result for sta100: OK(elapsed=0.0452320575714111 0.045232057
5714111)

lanforge@jedtest ~
>
```

Watching the Logs

Typically you won't need to look at this output in the terminal, and you will not add **-d -v** flags to your LANforge stations. You very likely will need to check the log output from these scripts in case you need to diagnose connection problems during your test. Each virtual station leaves a log in the `/home/lanforge/wifi` directory, like `wifi/portal-bot.sta100.log`

ℹ Watch logs using tail: **`tail -F wifi/portal-bot.sta100.log`**

Executing the LANforge `curl` commands yourself

To find the actual `curl` commands being executed, you want to `grep` the logs. Below is an example of grepping the logs and running the curl command.

```
$ cd /home/lanforge/wifi
$ grep Submitting portal-bot-sta100.log
Submitting: /home/lanforge/local/bin/curl -sLki -c /tmp/sta100_cookie.txt -b /tmp/sta100_cookie.txt -4 --interface sta10
Submitting: /home/lanforge/local/bin/curl -sLki -c /tmp/sta100_cookie.txt -b /tmp/sta100_cookie.txt -4 --interface sta10
```

You might noticed that some of the commands in the log might appear repeated, there are areas of redundant logging. There is a case where you can legitimately see repeated commands: when you have an Post IF_UP value configured for the port you are testing with. (Remember that the Post IF_UP field should be blank when developing the script.)

Remember, this `curl` command cannot be run without first doing a `source /home/lanforge/lanforge.profile` in your shell (our curl is a custom build). Here is an example. We take a command similar to the one above, add `-qv` and cancel it using Ctl-C :

```
$ cd /home/lanforge
 $ source lanforge.profile
 # add a -qv to see header details
 $ /home/lanforge/local/bin/curl -qv -sLki -c /tmp/sta100_cookie.txt -b /tmp/sta100_cookie.txt -4 --interface sta100 --
```

```
* STATE: INIT => CONNECT handle 0xa80158; line 1397 (connection #-5000)
* Added connection 0. The cache now contains 1 members
*   Trying 10.51.0.254...
* TCP_NODELAY set
* bind-local, addr: 10.41.4.223  dev: sta100
* SO_BINDTODEVICE sta100 failed with errno 1: Operation not permitted; will do regular bind
* Name 'sta100' family 2 resolved to '10.41.4.223' family 2
* Local port: 0
* STATE: CONNECT => WAITCONNECT handle 0xa80158; line 1450 (connection #0)
^C
```

Explaining the `curl` Command

There are many arguments to the curl command, but in general, you should be able to copy and paste the command into a terminal and it should work (see note about `lanforge.profile` above). Below is an example of a `curl` command, with `\` characters as line-continuation marks, formatted for clarity.

```
$ /home/lanforge/local/bin/curl -qv \
   -sLki                            \
   -c /tmp/sta100_cookie.txt        \
   -b /tmp/sta100_cookie.txt        \
   -4                               \
   --interface sta100               \
   --localaddr 10.41.4.223          \
   --dns-interface sta100           \
   --dns-ipv4-addr 10.41.4.223      \
   http://basic-portal/start
```

Switch	Example Value	Purpose
-q		Suppress page output
-v		Verbose, prints diagnostic steps
-s		Suppresses page output
-L		Follow redirects
-k		Suppress certificate validation errors
-i		Print HTTP headers
-c	sta100_cookie.txt	Send cookies from file
-b	sta100_cookie.txt	Save cookies to file
-4		Use IPv4
--interface	sta100	bind to this interface
--localaddr	10.41.4.223	bind to this address
--dns-interface	sta100	send DNS queries from this interface
--dns-ipv4-addr	10.41.4.223	bind to this address when sending DNS queries
--dns-interface	sta100	send DNS queries from this interface
-X	GET	Use HTTP GET method
	POST	Use HTTP POST method
-d	'username=bob'	URL encoded form parameters used during POST method

Your `portal-bot.bash` script is intended to be a way of focusing on the development of your bot plugin and not repetitively typing a long curl command.

Writing your Bot Plugin

Your bot plugin, the Perl module you will write for your captive portal, is central to the operation of the portal-bot.pl script. It is also important that you do not alter the `portal-bot.pl` script unless absolutely necessary, because your changes could be overwritten by upgrades. Any alteration to the time at which the `fork()` call is made in this script can make the LANforge server grind to a halt.

> ℹ Only edit your bot perl module, please.

The Bot Subroutines

The example bot, bp.pm, provided with LANforge defines four subroutines. In order:

find_redirect_url

This subroutine receives the response of the HTTP(S) GET of your --start_url parameter. Look through this to see if:

- you are already getting destination content--if so, you were not logged out,
- you get a login form directly and not a redirect,
- or you get a redirect to a login page (possibly on a separate port like :8080)

If you get a redirect to another port, compare the --login_url value to this. If it is different, consider updating your login_url parameter.

There might be many form parameters, like ones for a session id, a `PHP_SESSID`, a cookie, a base64 encoded string indicating your originally requested url (or just a plain URL-encoded url), and any possible co-branding parameters that might indicate any advertising campaigns associated with this captive portal. Missing some of these might make submitting the form give you an error. Store these values as necessary in your `bot::` namespace. You do not submit your login page in this method.

> ℹ Define a package scope variable using `our $thing;` after your package statement.

submit_login

Here is where you submit your login page forms. The `botlib::request()` function is provided to make GET and POST requets verbose logging and debugging. The page is returned as lines in the `@response` array.

```
my $post_data = "username=".uri_escape($user_name);
my @response  = ();
request({'curl_args'=> $::curl_args,
  'url'       => $post_url,
  'method'    => 'POST',
  'delay'     => '0,3',      # see --delays option
  'post_data' => $post_data,
  'print'     => 1},         # turns on debugging
  \@response);
```

The submit_login function uses the `$::delay[1]` parameter if `--delays` were set. See paragraph on randomDelay.

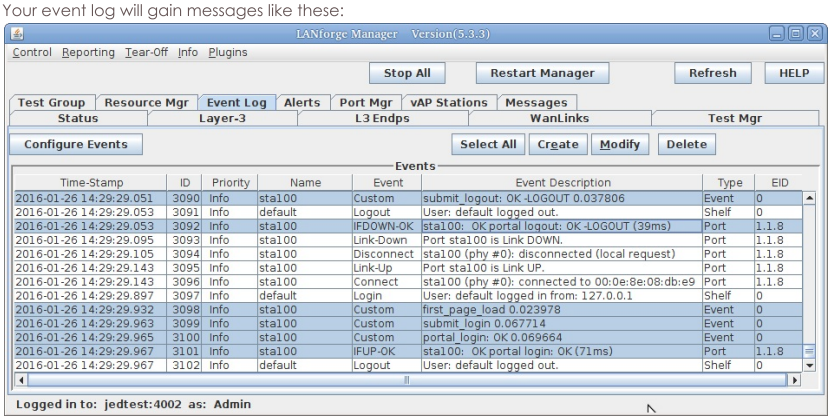
interpret_login_response

Here you determine if you are getting an access denied error or are being forwarded to your original start_url destination. Set your `$result` variable to **OK** or **FAIL**. Use the logg() method to add information for the wifi/portal-bot log.

In order to add events, such as page load time, you want to use the `botlib::newEvent()` function:

```
my $page_time = botlib::time_milli() - $::start_at;
newEvent("portal_login: $result", $page_time, $::dev);
```

Your event log will gain messages like these:



get_explanation

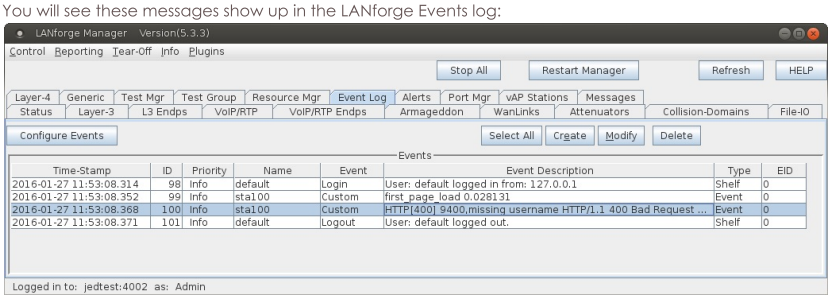
Some web applications can provide customized error messages in their response. You can add a `get_explanation()` function to your bot to collect this information. The `botlib::dbgdie()` method will take advantage of this method if available. Below is an excerpt from the method found in `bp.pm`:

```
sub get_explanation {
   for $line (@$ra_result) {
      ($err_code) = $line =~ /^X-err-no: (.*)$/
         if ($line =~ /^X-err-no: /);
      ($err_msg ) = $line =~ /^X-err-msg: (.*)$/
         if ($line =~ /^X-err-msg: /);
   }
   return "$err_code, $err_msg";
}
```

Notice how this parses out the HTTP headers found if the parameter `username` were missing when doing a POST to `basic-portal/login.php`:

```
header("X-err-no: 9400");
header("X-err-msg: missing username");
```

You will see these messages show up in the LANforge Events log:



submit_logout

Many captive portals do not publicise their logout URLs, so it might be available only on an admin page for the AP. You will know when the logout_url parameter works if you can do a logout with that station, and then successfully log back in using the same station and seeing the captive portal sign-in page again.

randomDelay

The delay parameter to `botlib::request()` has many overloads to the call:

- A simple number is a simple delay in seconds. No other units are used.
- If you specify 'random' in the delay parameter, the `botlib::randomDelay()` is called,

producing a range between [1 - 119] seconds.

- If you specify '3-16', `randomDelay(3, 16)` is called to produce a random range between [3 - 16] seconds.
- If you specify two numbers separated by a comma, it looks at your `@::delays` list, and picks the second argument if it can, the last item of `@::delays` if the list is too short, or the first argument if there are no items in the delay list.

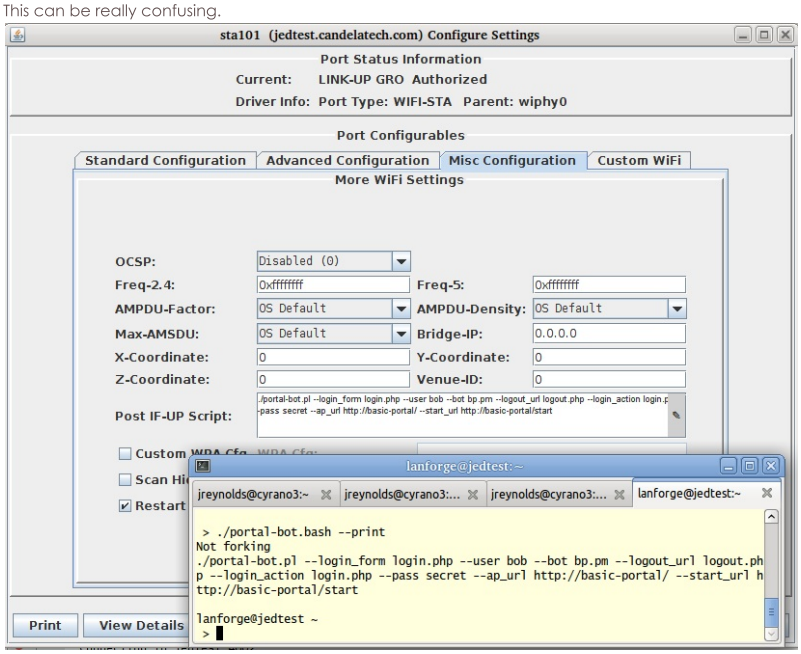
We have now covered all of the scripting development areas for the `portal-bot.pl` plugin you will write.

Configuring your Stations

A Single Station

We assume you have `portal-bot.bash` working at this point. This is how you can configure a single station:

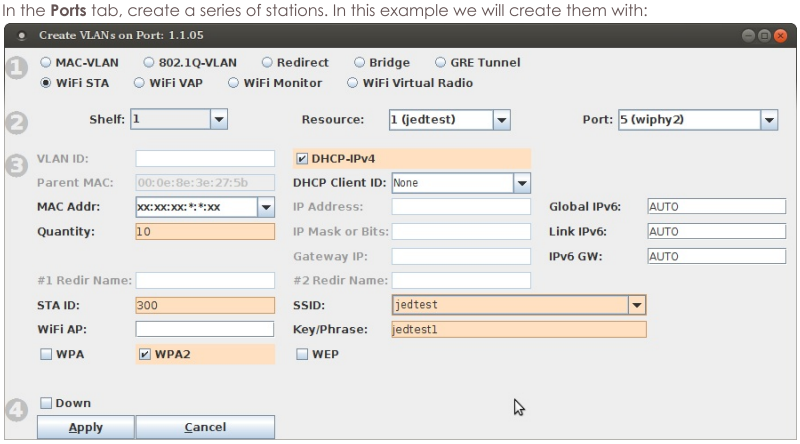
1. Use the `portal-bot.pl --print` command to print out the arguments.
2. Copy the result (starting with "portal-bot.pl") into the Port->Misc window. Avoid populating this field while you are developing the script! If you place a value into that field, your portal-bot script will not only execute, but the Manager process will also execute the script specified in the POST_IFUP field. This can be really confusing.



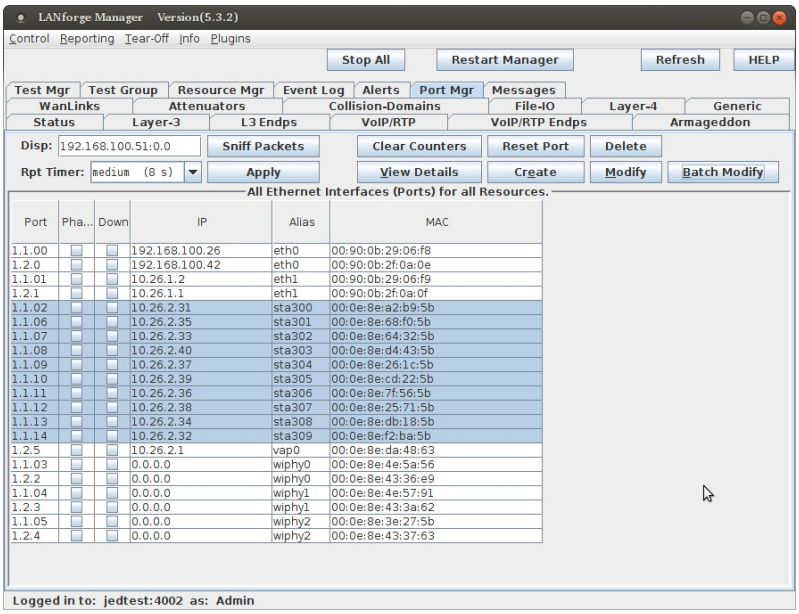
Multiple Stations

To get multiple virtual stations logging in an out using the GUI, we just need a few of those parameters for the station configuration. We will use the Batch Modify feature to alter a series of stations.

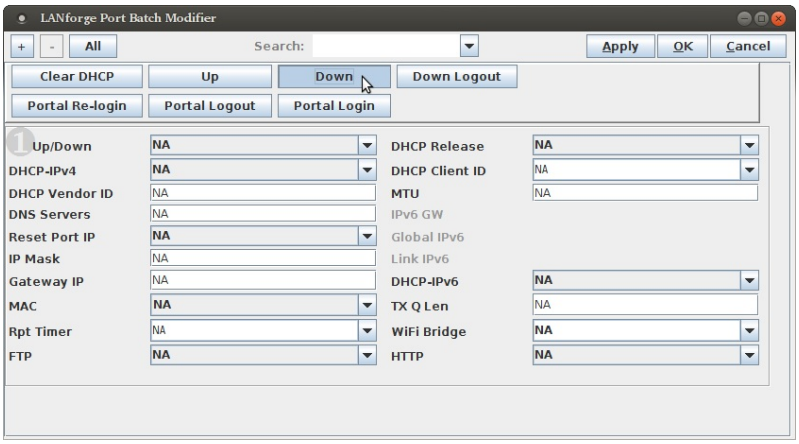
1. In the **Ports** tab, create a series of stations. In this example we will create them with:



   - Port: wiphy2
   - Select DHCP-IPv4
   - Quantity: 10
   - STA ID: 300
   - SSID: jedtest
   - Passphrase: jedtest1
   - Select WPA2
   - Select Down
2. Highlight them and click **Batch Modify**.



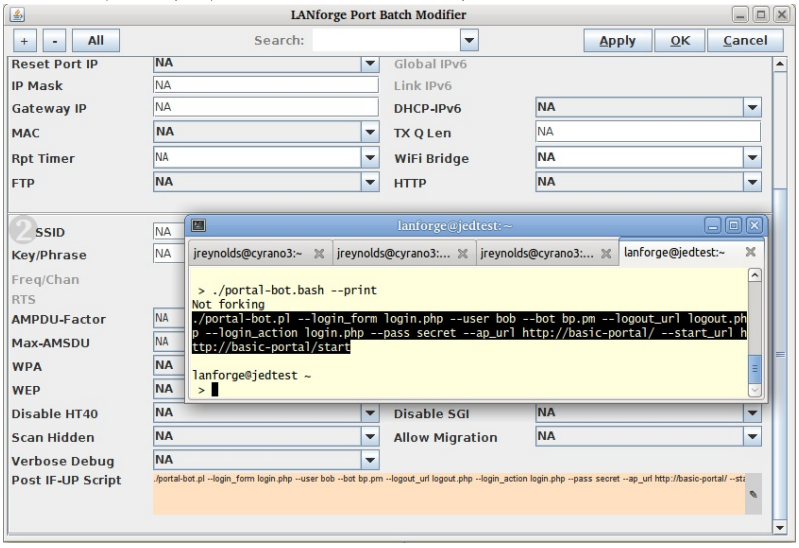
3. Click the **Down** button.



4. In your terminal, invoke the `portal-bot.bash` with the `--print` argument:

```
./portal-bot.bash --print
portal-bot.pl --bot bp.pm --user bob --pass bob1 --ap_url http://basic-portal/
 --start_url  http://basic-portal/start --login_form login.php --login_action login.php
 --logout_form logout.php
```

5. Use the **[+]** button to expand to Box 2. We will enter the following version of our command into the Post IF-UP Script area. (The picture shows the short switches.)



Click **OK**

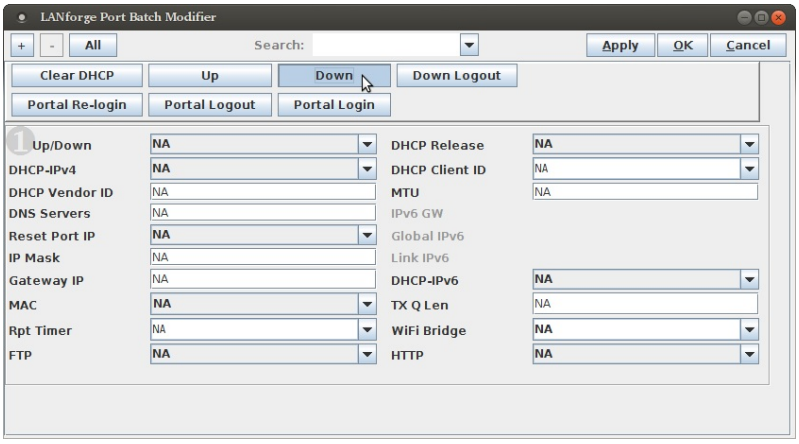
6. In the **Ports** tab, double click sta300 and in the Misc Configuration tab, you will see the Post IF-UP Script values.

Testing a Station

Exercising these stations starts with bringing them up and down using the Batch Modify tool.

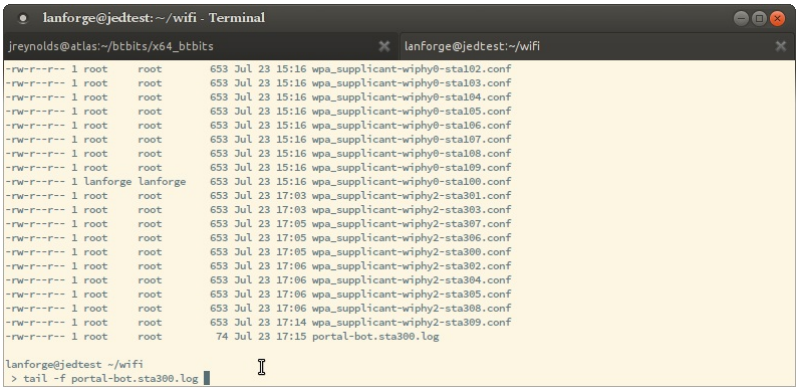
1. Highlight one station, `sta300`, and click **Batch Modify**.

2. Click the **Down** button to admin-down the station.



3. In a shell on the LANforge, got to `/home/lanforge/wifi` and tail the log for station 300:

```
tail -f portal-bot.sta300.log
```

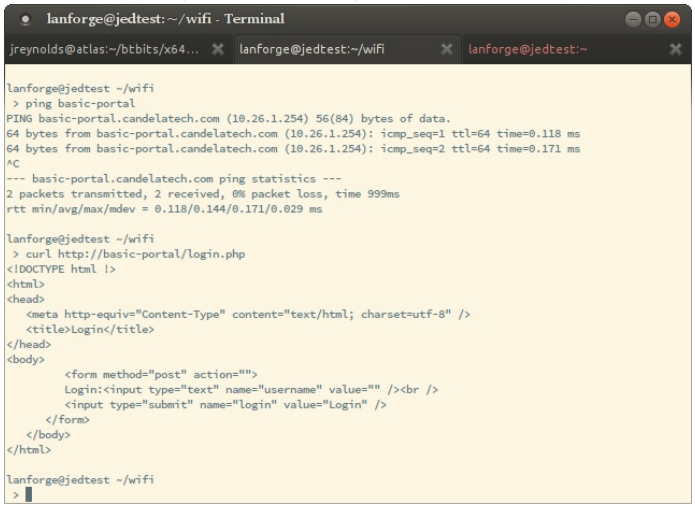


4. Click the **Up** button to admin-up the station.
5. Click the **Portal Login** button force the station to login if you do not see any messages in the log file you are tailing.

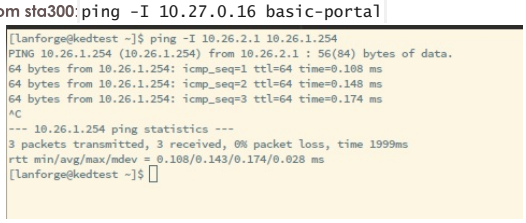
Troubleshooting Techniques

If your station cannot talk to the captive portal, like you have a time-out, these steps will help identify where there is a misconfiguration:

1. **Ping the portal from LANforge**: `ping basic-portal`



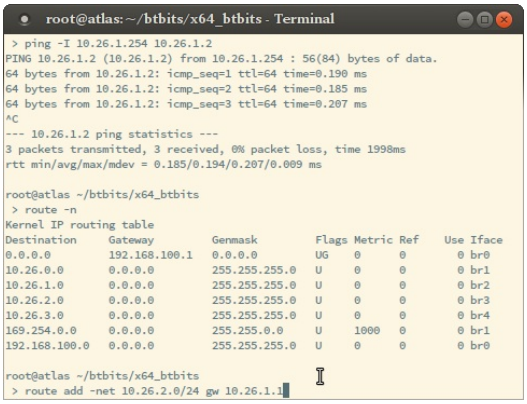
2. **Ping the portal from sta300**: `ping -I 10.27.0.16 basic-portal`



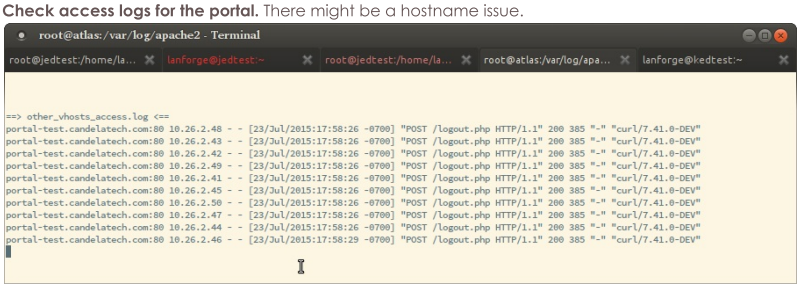
3. **Use curl** to download the portal page by hand: `curl -sqv http://basic-portal/login.php`
4. **Check the route** on the portal side if you are routing. Some examples:

```
route -n
```

```
route add -net 10.27.0.0/23 gw 10.26.1.1
```



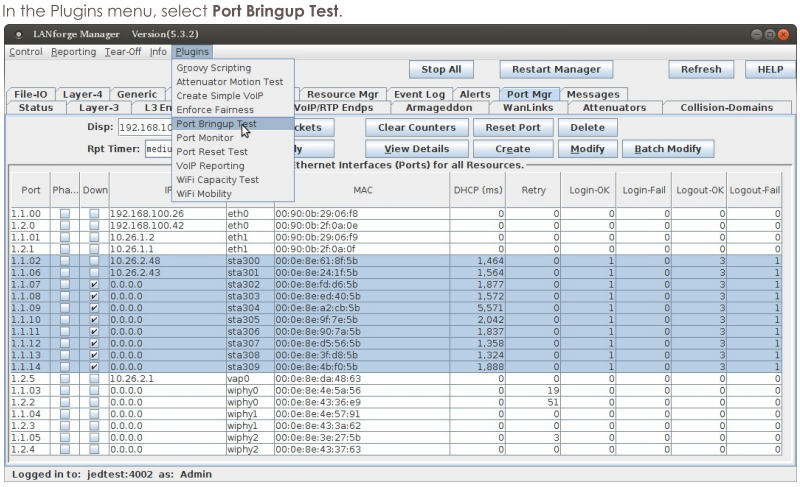
5. **Check access logs for the portal.** There might be a hostname issue.



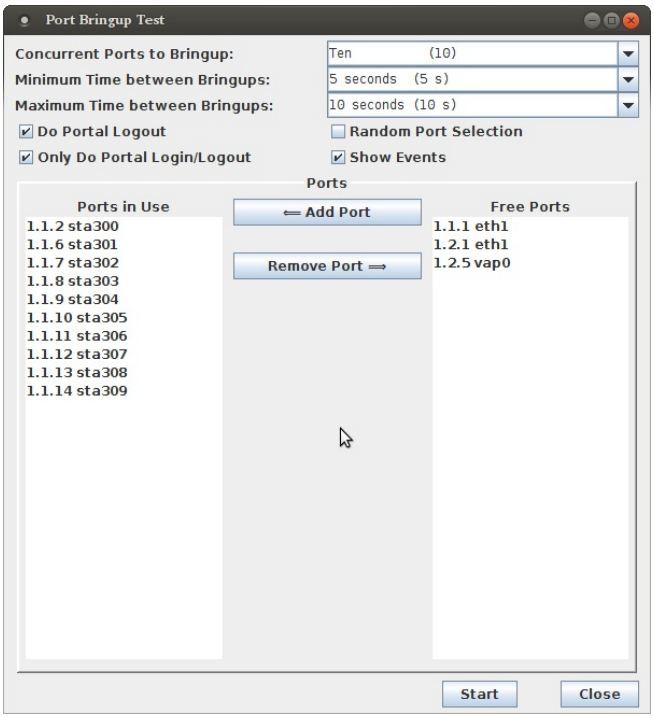
Using the Port Bringup Plugin

Using the Port Bringup Plugin is a much more fun way to get data than looking at log files.

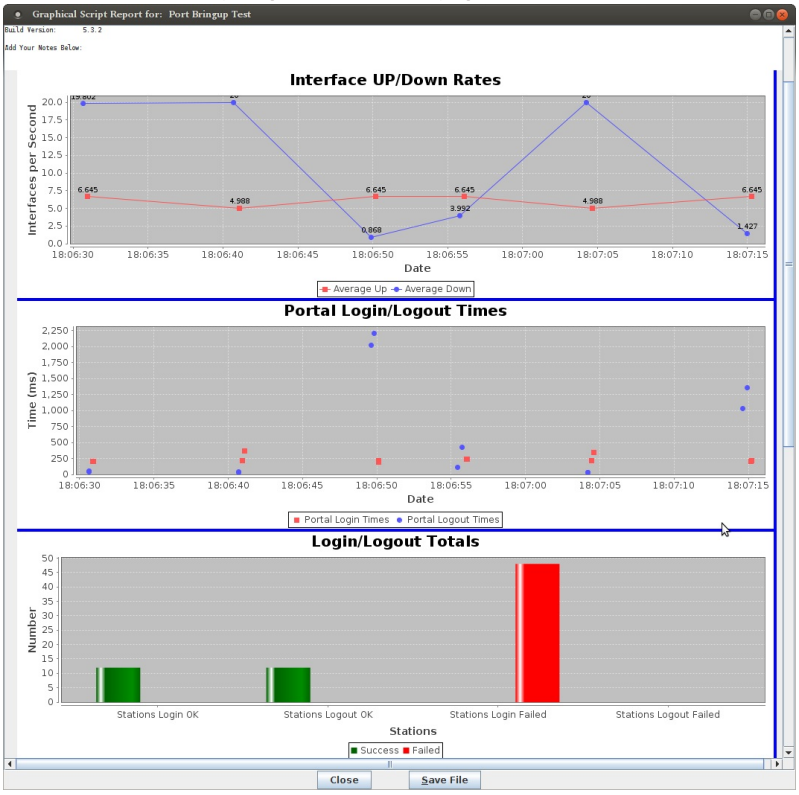
1. In the Plugins menu, select **Port Bringup Test**.



2. Highlight a series of stations and click **Add Port**:



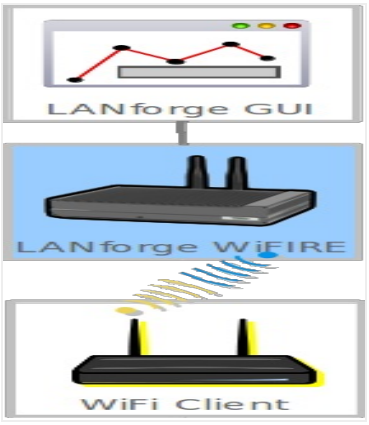
3. Click **Start**
4. You will see the reporting window. It often takes many seconds or a few minutes for stations to aquire DHCP addresses and start reporting information into the plugin.

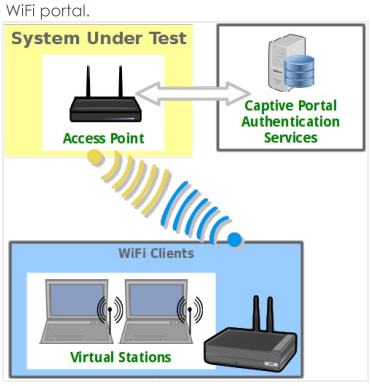


WiFi Captive Portal Login

Goal: Create many user sessions to a WiFi captive portal gateway.

Airports, arenas and coffee shops often offer open WiFi service that is gated with a web sign-on form. This is called a captive portal. LANforge can run a custom login script on a virtual station to emulate sign-in on the captive portal web page. The following example will create one hundred stations and have them authenticate through a captive WiFi portal.

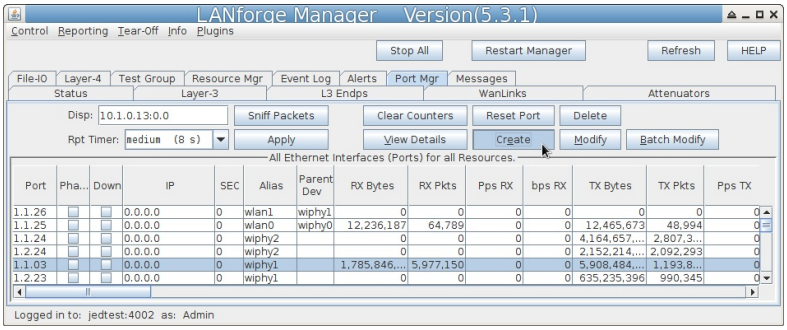




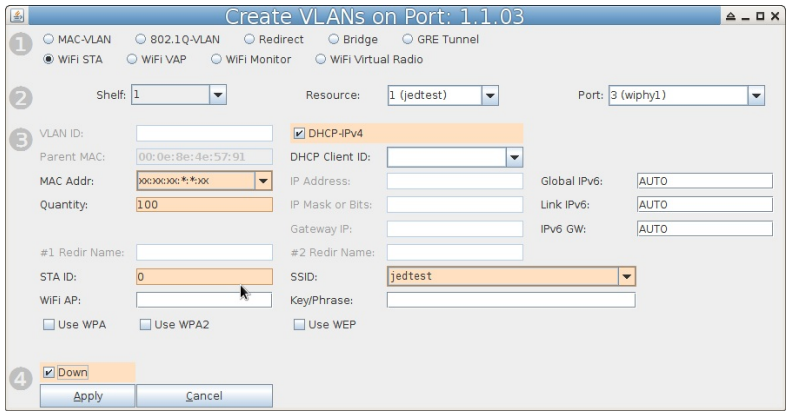
---

1. Prepare a portal login script (`lf_ifup_post`)
2. The `lf_ifup_post` script will be called after DHCP assignment for a station occurs. It can also be called before DHCP release. This script is called from the LANforge resource hosting the virtual station.
    A. The script should be in directory `/home/lanforge/`.
    B. LANforge will pass these arguments to the script
        A. `-i` - station device
        B. `--ip4` - station ip address
        C. `--ip6` - station ipv6 address
        D. `--dns` - station DNS address
        E. `--mgt` - pipe name for reporting results to LANforge
    C. Custom parameters to the script can be provided.
    D. The script can have another name.
    E. LANforge expects these return values:
        A. **OK**
        B. **FAIL**
        C. **FAIL:reason**, this provides feedback on failure occurrance.

3. Create WiFi stations
    A. In the **Ports** tab, select wiphy1 and click **Create**



B. In the **Create VLANS** window, craft ten wifi stations:



A. Select WiFi STA

B. For MAC address, choose `xx:xx:xx:*:*:xx`

C. Select DHCP-IPv4

D. Enter Quantity **100**

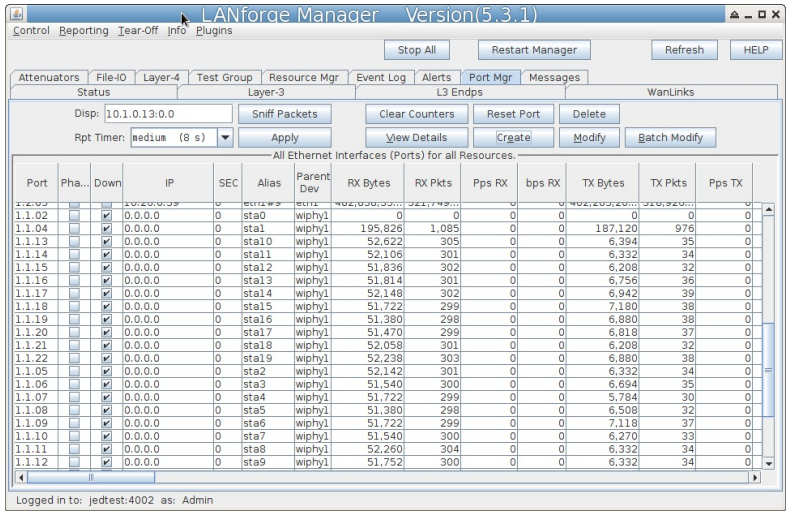
E. Specify **0** for STA ID

F. The example SSID for this cookbook is `jedtest`

G. Select the **Down** option. This postpones the stations making a DHCP request until they are explicity admin up.

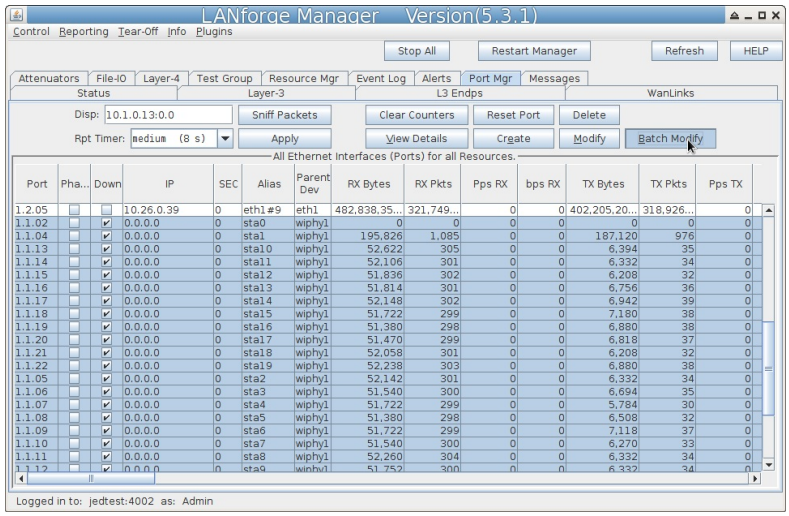
H. ...and then click **Apply**

C. You will see ten station created:

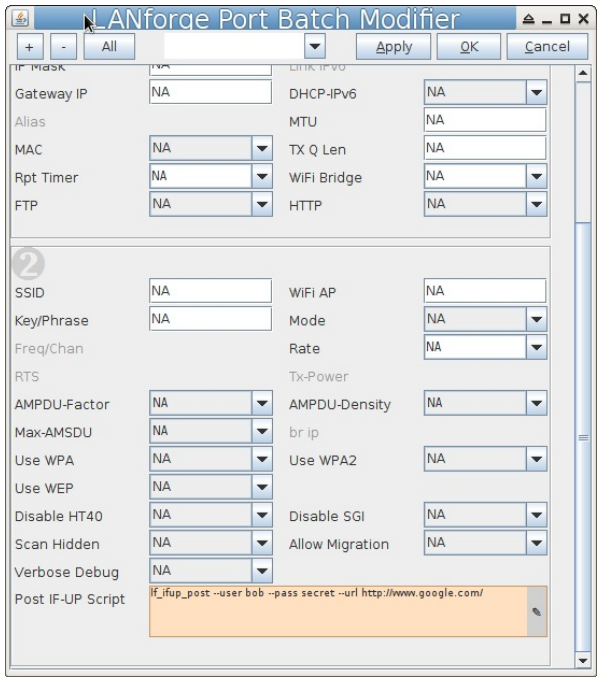


4. Batch Modify Stations in order to update `lf_ifup_post` parameters

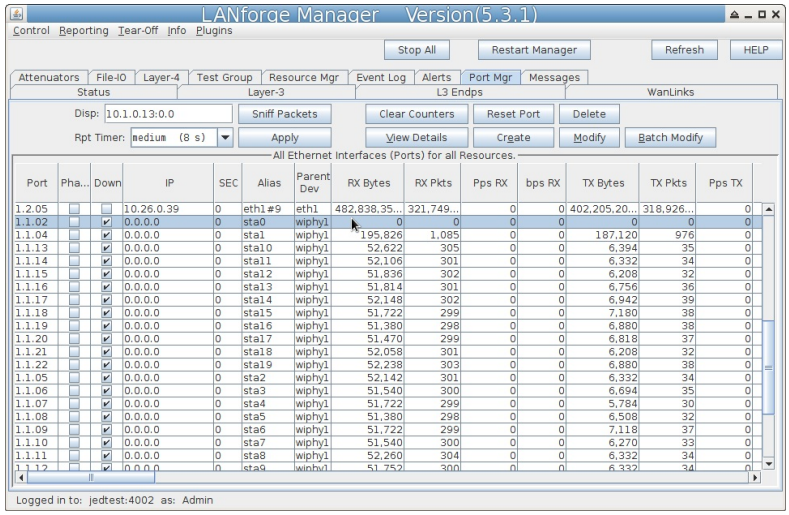
A. Highlight stations and click **Batch Modify**



B. In the Batch Modify screen, click the **+** button and expand to Group 2.

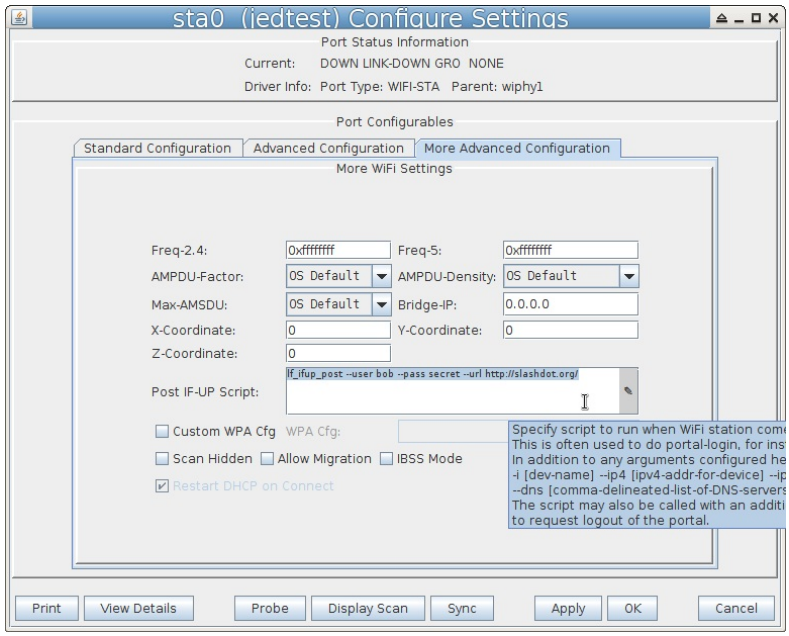


C. Edit the IF POST field. Enter the file name and any extra arguments that the script will want for this port. Example: `lf_ifup_post --user bob --pass secret --url http://slashdot.org/`



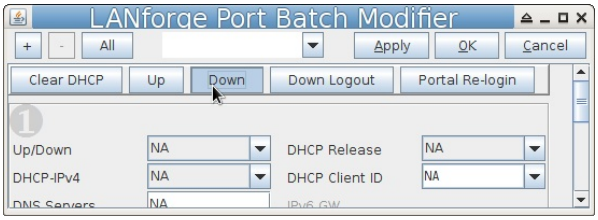
D. Click the **Apply** button to apply the changes. Do not close the window yet.

E. To check the value of each port's **IF/UP Post** you can use the Ports tab. In the port Configure Settings window, in the **More Advanced Tab** you can find the IF/UP script value.

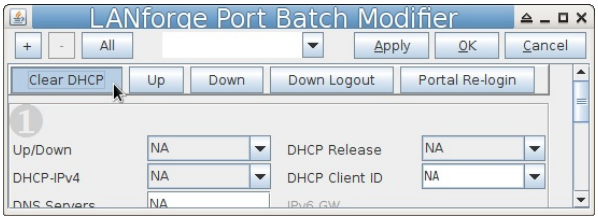


5. Testing a station. We will toggle it up and down and look at the logs to find problems.
    A. Highlight one of your stations in the Ports tab.
    B. Set the Report Time to `1s` and click **Apply**
    C. In the Batch Modify window, click the **Up** button. This is the same as settting **Up/Down** to **UP** and clicking **Apply**
    D. Watch the Wireless Messages and LANforge Messages windows for error messages. The Ports tab will update the station status.

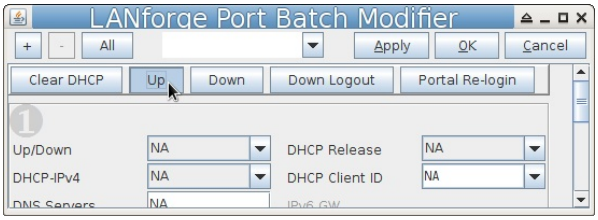
6. Use the Batch Modify window to **Force DHCP Renewal**
    A. Click **Down**



    B. click **Clear DHCP**

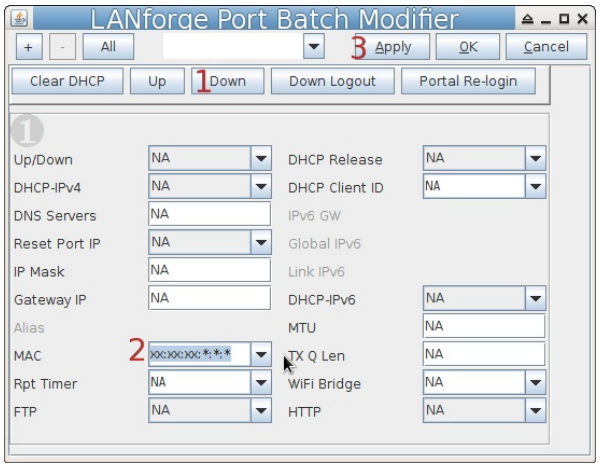


    C. click **Up**



7. Use the Batch Modify window to **Re-Login to Portal**
    A. Click **Portal Re-Login**
    B. If you wanted to change other parameters:
    C. Click **Down Logout**
    D. Set Up/Down to **Down**
    E. Change another station parameters and then click **Apply**
    F. Click **Up**

8. Use the Batch Modify window to **Change station MAC addresses**



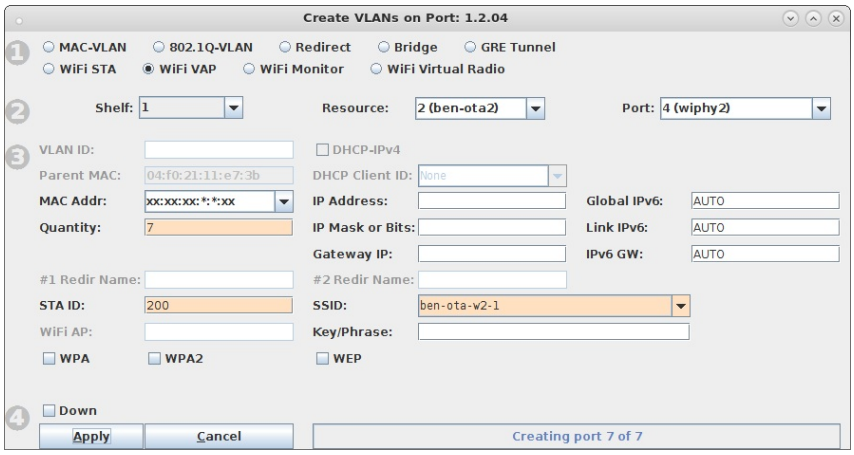
    A. Click **Down**
    B. Set MAC Addr to `xx:xx:xx:*:*:*` randomize the mac address
    C. Click **Apply**
    D. Click **Up**

LANforge WiFi Many vAP Testing

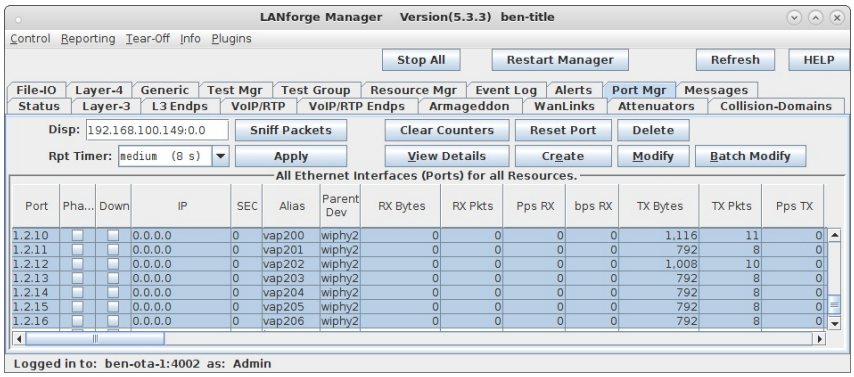
Goal: Create 7 vAP on a single a/b/g/n/AC radio to emulate a busy environment and test that station devices associate to the proper AP.

Requires LANforge 5.3.3 or later. Configure 7 vAP, add the vAP to a bridge and set up DHCP. The Device Under Test (DUT) in this case is a mobile handset or other wifi station device. Verify that station can handle many APs and select an appropriate AP from the available scan results. This example uses a LANforge CT523 system but the procedure should work on all CT521, CT522, CT523 and CT525 systems.

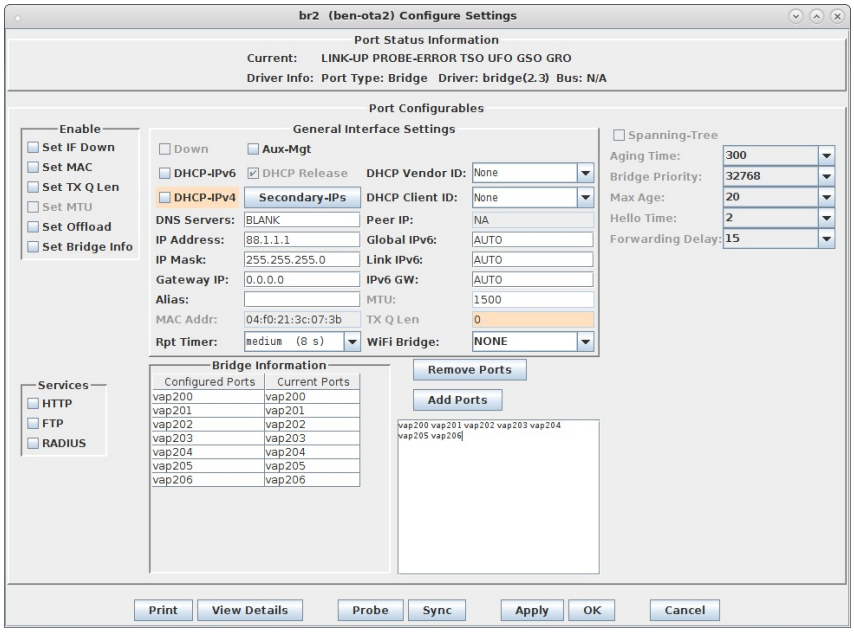
1. In the **Ports** tab, select the radio `wiphy2` and click **Create.** Configure the values appropriately and click create.



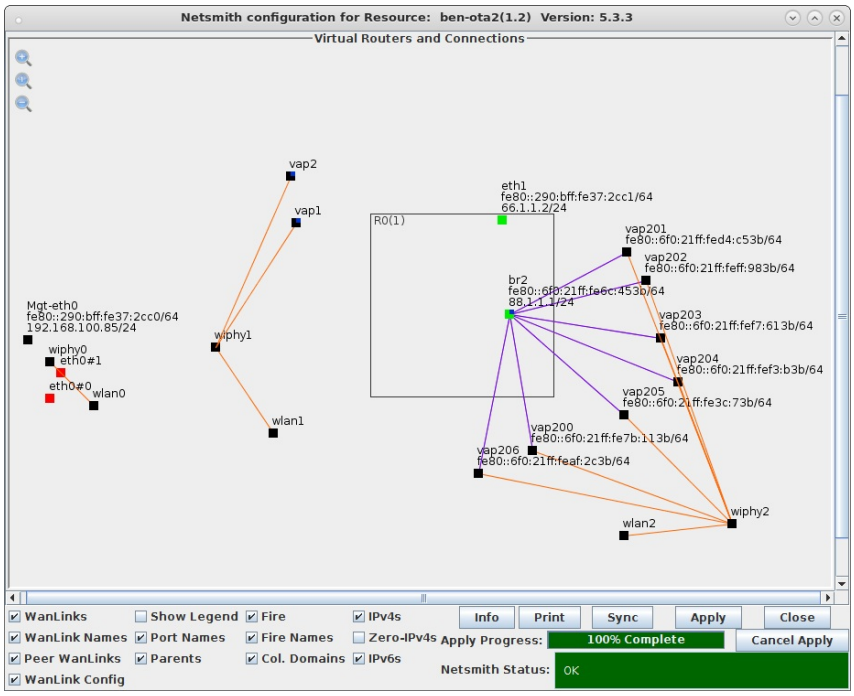
2. In the **Ports** tab you will see the new WiFi vAP:



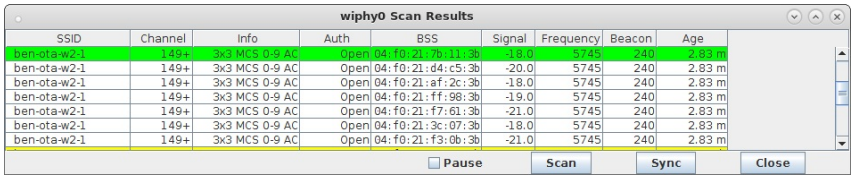
3. Select the **Status** panel in the LANforge GUI, and click the Netsmith button for the appropriate resource. Right-click and select the 'New Bridge' option. In this example, I selected 'br2' as the bridge name. After creating the bridge, click Sync to show the new bridge device. Right-click on it and select Modify Port. Add each of the vAP you just created to the bridge and then apply:



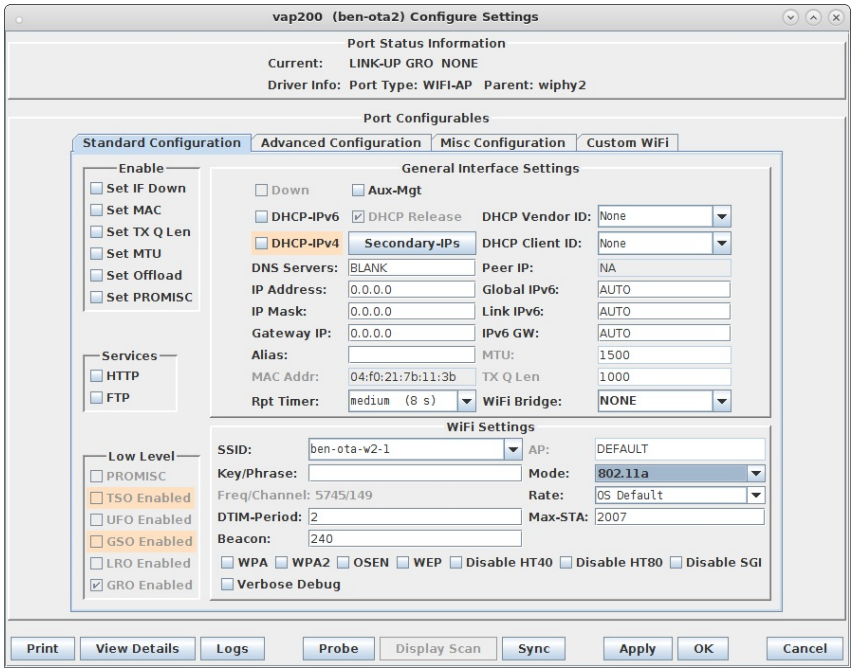
4. Create a virtual router in Netsmith and add br2, and optionally a wired port (eth1) to the router. Double-click the br2 port and configure DHCP to match its IP address. When complete, Netsmith should look something like this:



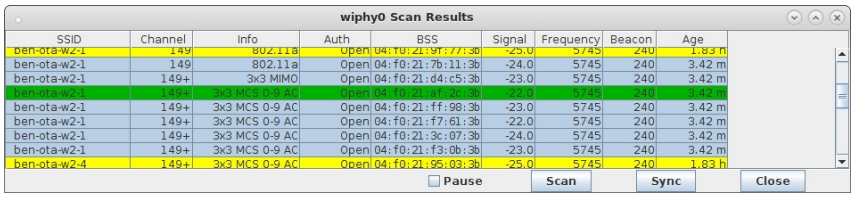
5. Now, we should have 7 vAP able to accept stations and give out DHCP addresses. Depending on the DUT, the user may wish to run iperf on LANforge, or on an upstream device connected to the LANforge eth1 port. For an initial test, make sure the DUT can connect to one of the vAP and get an IP address. The DUT should also see each of the vAP in its listing of available APs. This example will use LANforge WiFi Station on a different radio as the DUT. Here is a listing of the scan results:



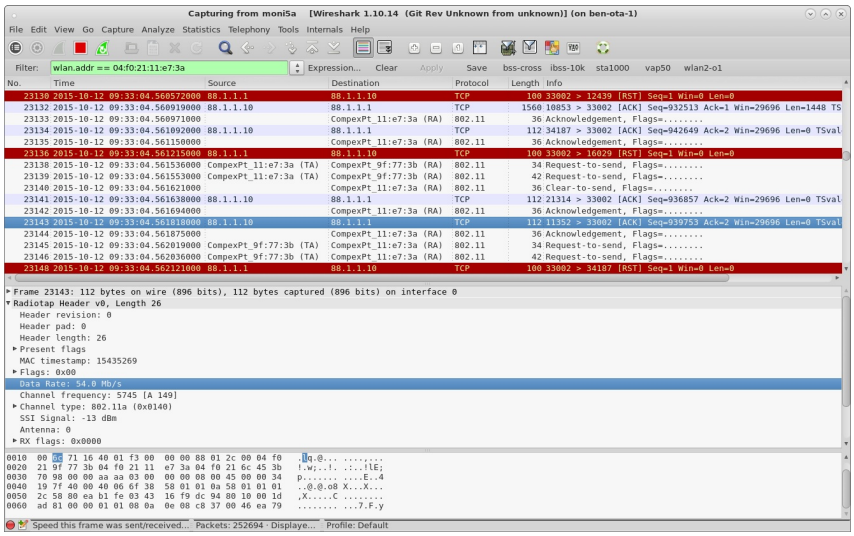
6. To make it a bit more interesting, we will now set the operating modes for one AP to be 802.11a, a second to be 802.11n, and the rest will remain 802.11AC. These APs are running on channel 149, so b and g mode are not available on this radio. To set the mode, double-click the vap200 row and set the Mode to be 802.11a and click OK to apply. Use similar procedure to set vap201's mode to 802.11an:



7. Now, request the DUT to re-scan and re-associate to the network. There should now be one 802.11a, one 802.11n, and 5 802.11AC vAP in the scan results. A well behaved DUT should attempt to connect to the AP with the higest rate that the DUT supports. In this case, the LANforge Station properly selected the 802.11AC vAP:



8. For additional testing, you may wish to use additional LANforge radios to create more vAP, change the SSID and configure DUT to connect to a particular SSID, admin down vAP to make sure DUT will properly connect to a new AP, and much more. You may also run traffic on the different APs to ensure that if a DUT connects to a 802.11a AP, then it does not try to send any 802.11n (HT) encoded traffic. A LANforge radio configured for monitor mode could verify this, as could third-party sniffers.:

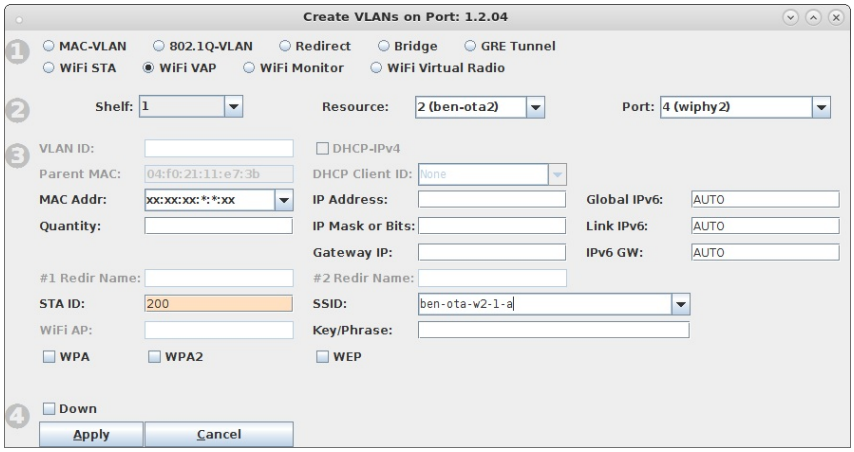


LANforge WiFi Degraded vAP Testing

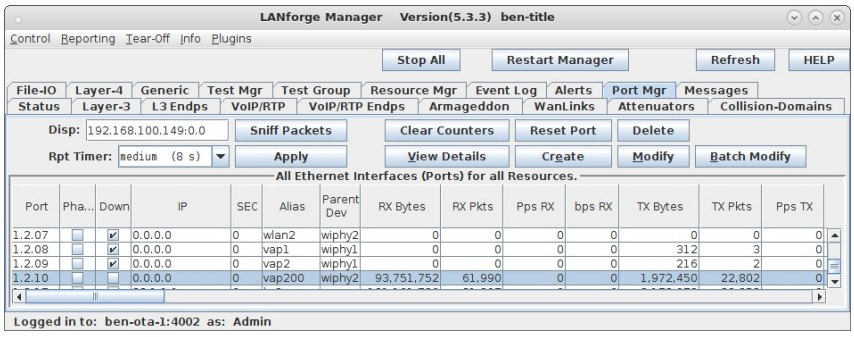
Goal: Create 1 vAP on a single a/b/g/n/AC radio and configure it to drop 50% of management frames to test that station devices can handle lost management frames properly.

Requires LANforge 5.3.2 or later. Configure 1 vAP, add the vAP to a bridge and set up DHCP. The Device Under Test (DUT) in this case is a mobile handset or other wifi station device. Verify that station can handle associating with an AP that drops many management frames. This example uses a LANforge CT523 system but the procedure should work on all CT520, CT521, CT522, CT523 and CT525 systems.

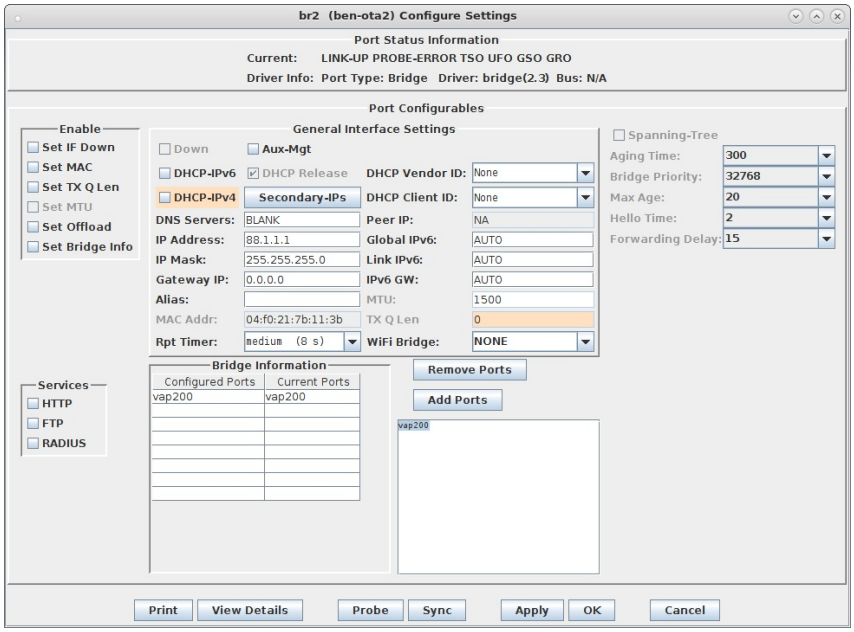
1. In the **Ports** tab, select the radio `wiphy2` and click **Create.** Configure the values appropriately and click create.



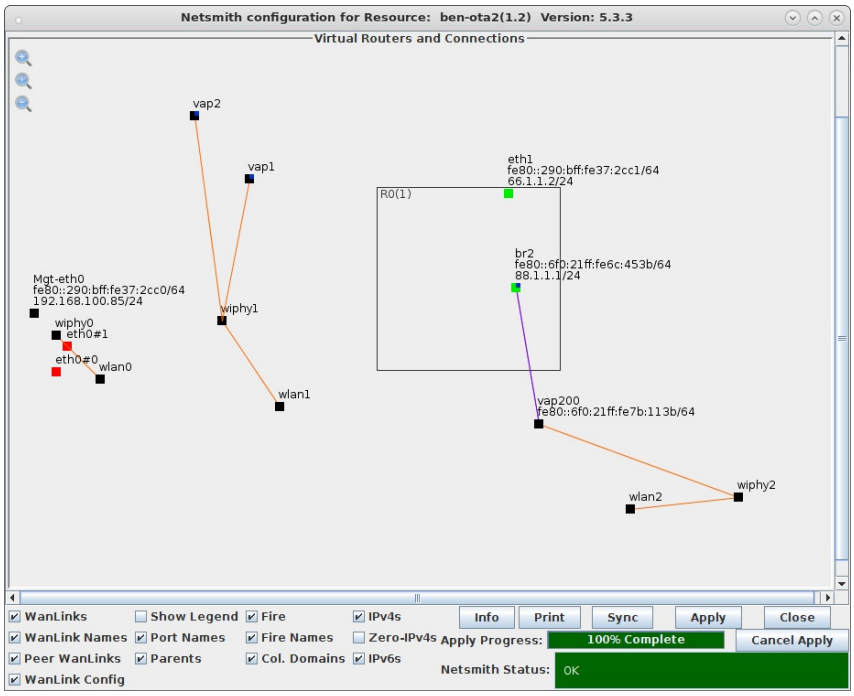
2. In the **Ports** tab you will see the new WiFi vAP:



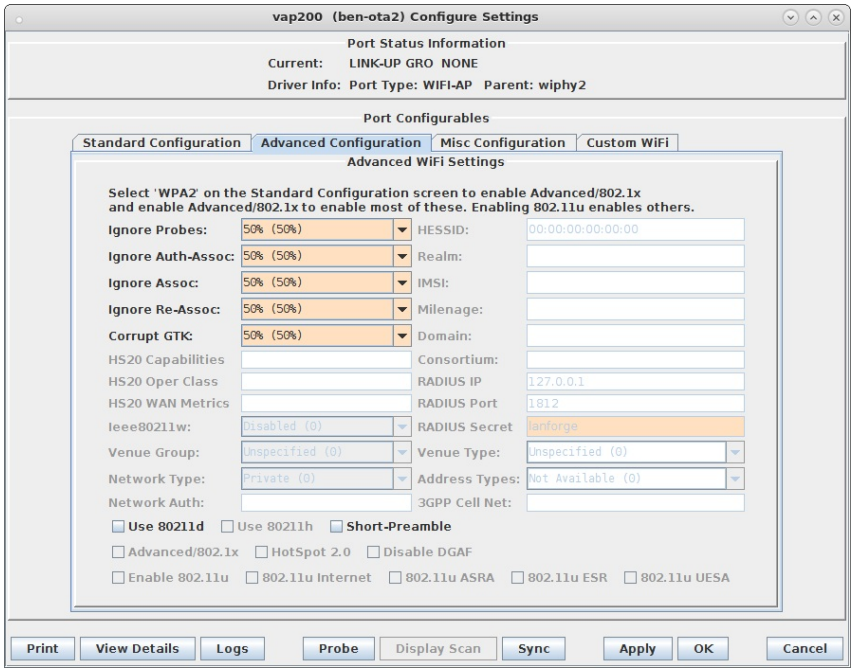
3. Select the **Status** panel in the LANforge GUI, and click the Netsmith button for the appropriate resource. Right-click and select the 'New Bridge' option. In this example, I selected 'br2' as the bridge name. After creating the bridge, click Sync to show the new bridge device. Right-click on br2 and select Modify Port. Add the vAP you just created to the bridge with the Add Ports button and then apply:



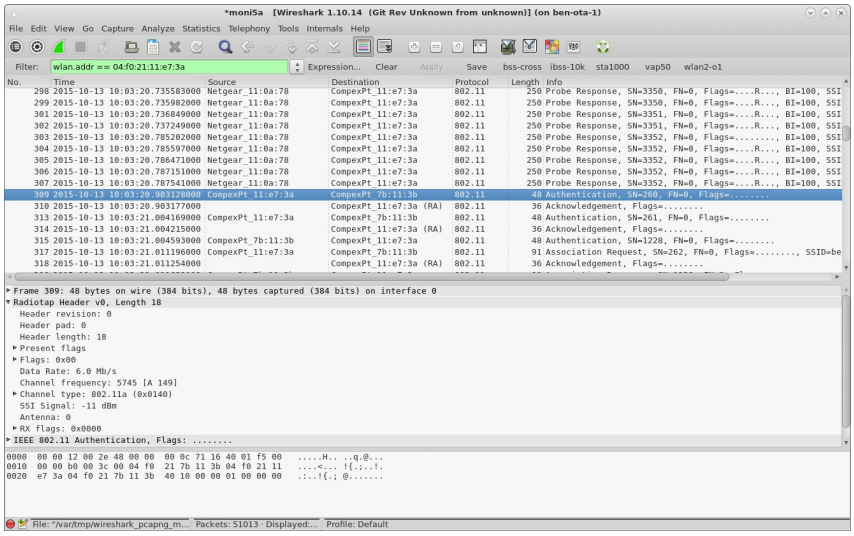
4. Create a virtual router in Netsmith and add br2, and optionally a wired port (eth1) to the router. Double-click the br2 port and configure DHCP to match its IP address. When complete, Netsmith should look something like this:



5. Now, we should have 1 vAP able to accept stations and give out DHCP addresses. For an initial test, make sure the DUT can connect to the vAP and get an IP address. Once that is verified, right-click and choose Port Modify on the vap200 vAP. We will now configure it to not respond to 50% of the management frames sent to it:

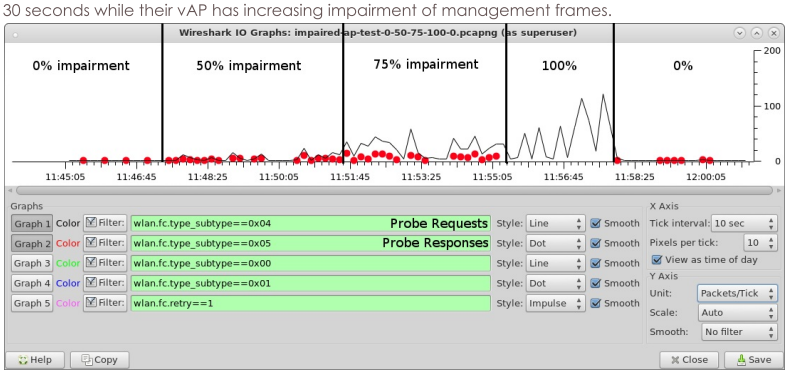


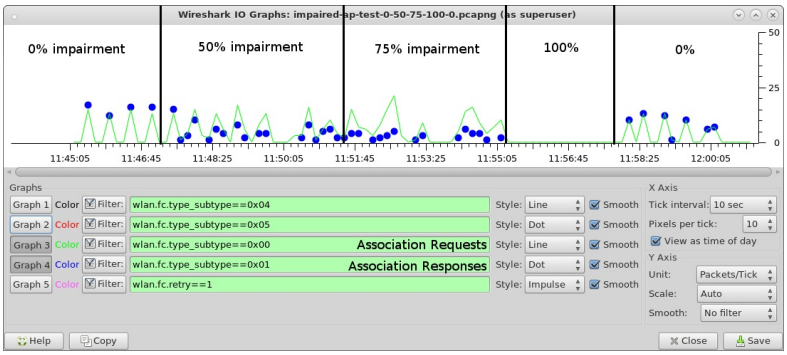
6. In this case, we are using open authentication, but it would also be good to test with encryption (WPA2 PSK, for instance) to make sure that the DUT can handle failures of the 4-way authentication handshake, for instance.
7. To verify the results, use a sniffer to watch the association requests and responses. A LANforge radio configured for monitor mode could verify this, as could third-party sniffers. In the capture below you can see that the station had to make two Authentication requests before the AP would answer (because the AP is set to randomly ignore 50% of the association requests):



A. Also in Wireshark, go to the **Statistics** menu and select **IO Graphs** to display up to 5 graphs based on the available frames in the capture file.

B. The two images below have been annotated to show the behavior of 10 stations being reset every 30 seconds while their vAP has increasing impairment of management frames.

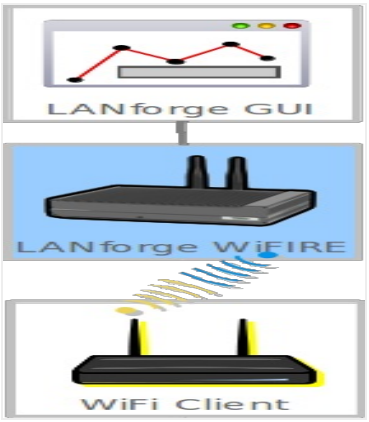


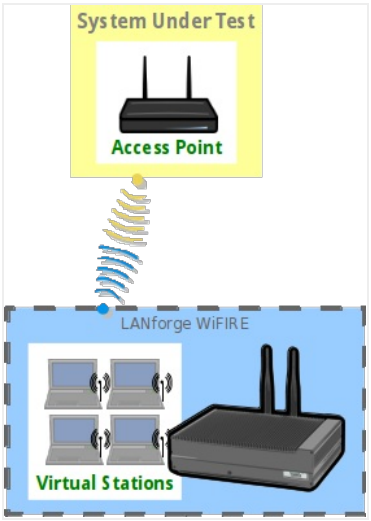


Multiple Stations with a Hunt Script

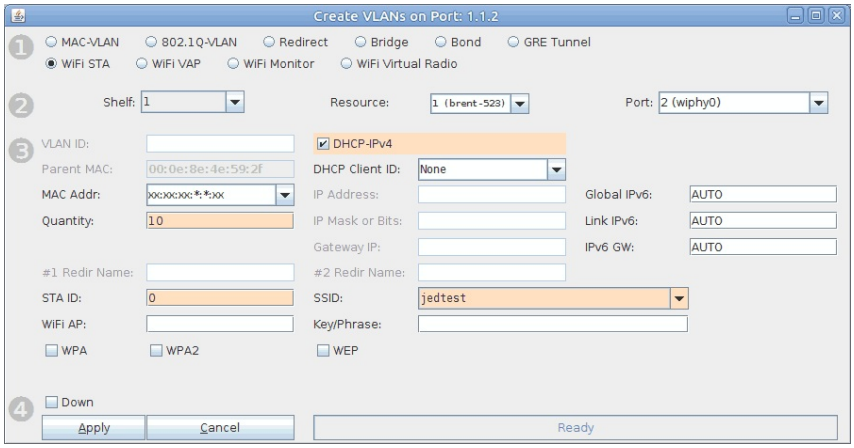
Goal: Do packet sized testing with a hunt script using multiple stations.

We will manipulate the parameters of 10 Layer 3 connections using a single hunt script. The WiFi stations will change packet size as a group by being part of a Test Group. In this scenario, we will create traffic to different upstream destinations using MAC VLANs, and the AP wired up to **eth1** of our LANforge machine. Requires a CT-520 for only stations, or a CT-523 if you also want a WiFi monitor station.





1. Create 10 virtual stations: in the **Port Mgr** tab, highlight radio **wiphy0** and click the Create button. In this scenario, we are using SSID jedtest.



A. Select WiFi STA

B. Check DHCP-IPv4

C. Quantity: 10

D. Station ID: 0

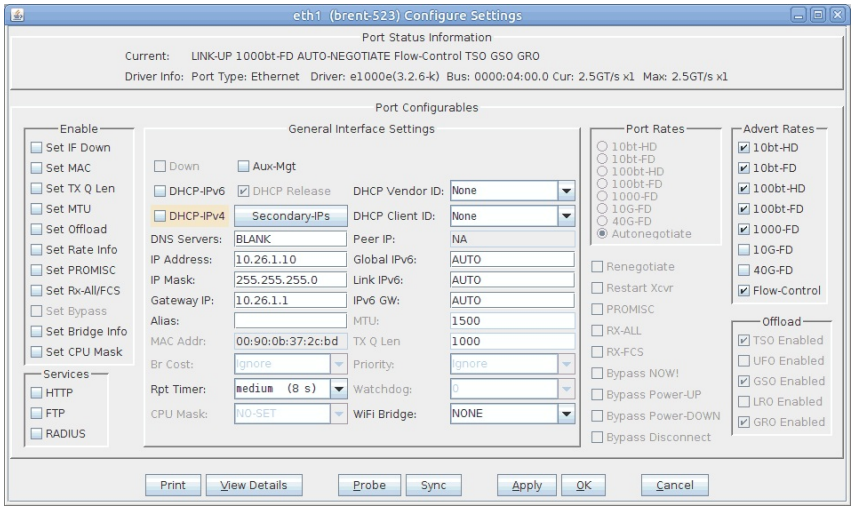
E. SSID: jedtest

F. Click **Apply** and then close the window.

G. You should see stations `sta0 - sta9.`

For more information see creating virtual stations

2. **(Simple Method)** This method only requires setting the IP on eth1. In this scenario, our upstream network is `10.26.1.0/24`. In the **Port Mgr** tab, highlight **eth1** and click **Modify**.



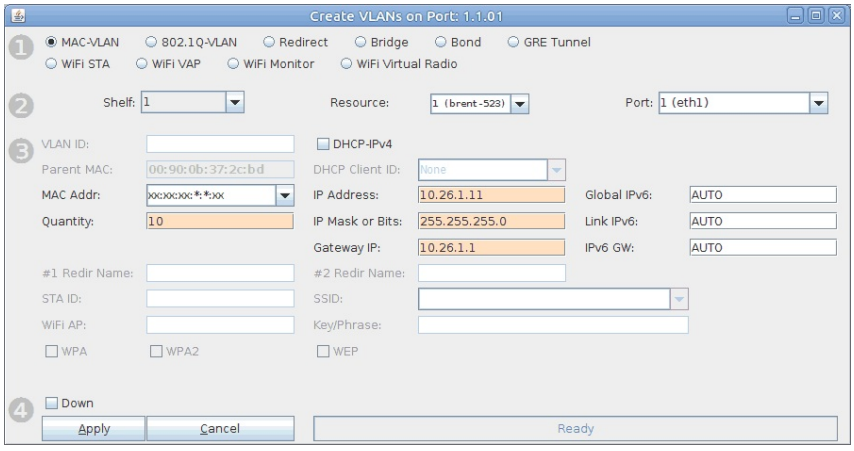
A. IP: 10.26.1.10

B. IP Mask: 255.255.255.0

C. Gateway IP: 10.26.1.1

D. Click **OK**.

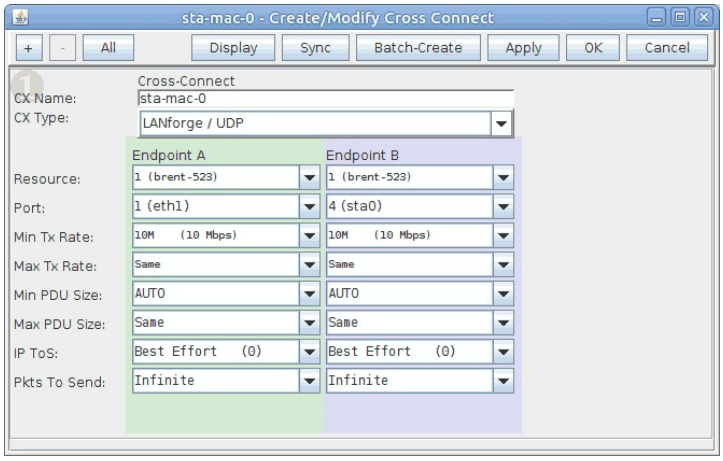
3. **(Optional Advanced Method)** Create ten MAC VLANs on the eth1. In this scenario, our upstream network is `10.26.1.0/24`. In the **Port Mgr** tab, highlight **eth1** and click **Create**.



A. Select MAC-VLAN

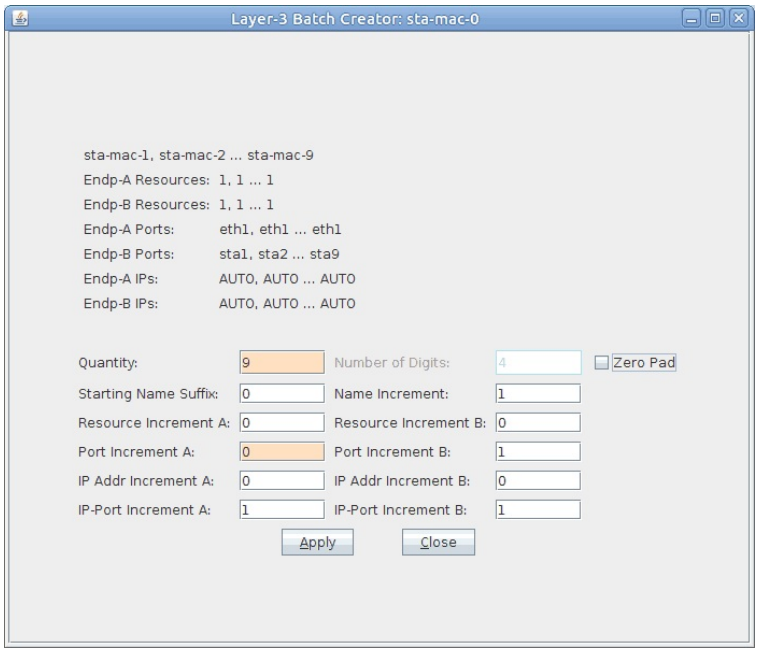
B. Quantity: 10
C. IP: 10.26.1.11
D. IP Mask: 255.255.255.0
E. Gateway: 10.26.1.1
F. Click **Apply** and close the window.
G. You should see 10 MAC VLANs, `eth1#0 - eth1#9`.

4. Create ten Layer-3 cross connects. We will start at 10Mbps transmit on them as a reasonable start. In general hunt scripts start low and try to work their way higher. When using more stations, set a lower starting transmit rate. In the Layer-3 tab, click Create.



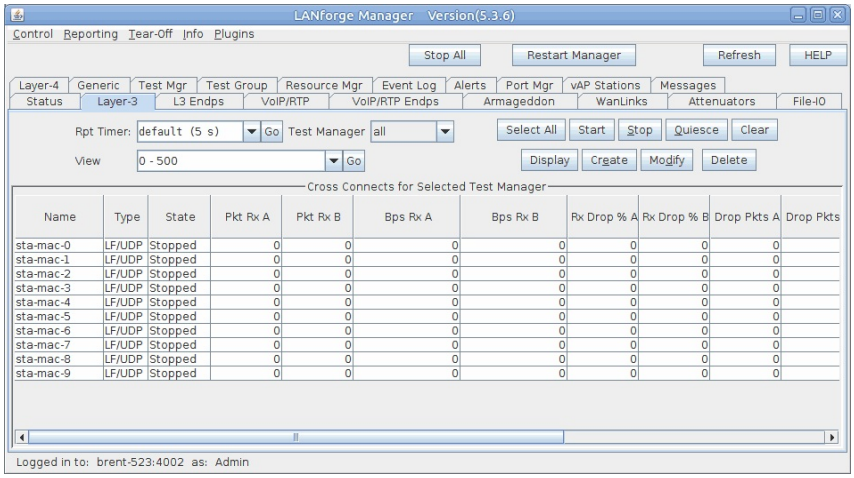
A. Name: **sta-mac-0**
B. Endpoint-A: **eth1** (if using the advanced MAC-VLAN method, set this to **eth1#0**.
C. Endpoint-B: sta0
D. Type: LANforge / UDP
E. Min Tx Rate: 10Mbps (both sides)
F. Click **Apply**. Leave the window open.

5. Create nine more stations. Click **Batch-Create**.



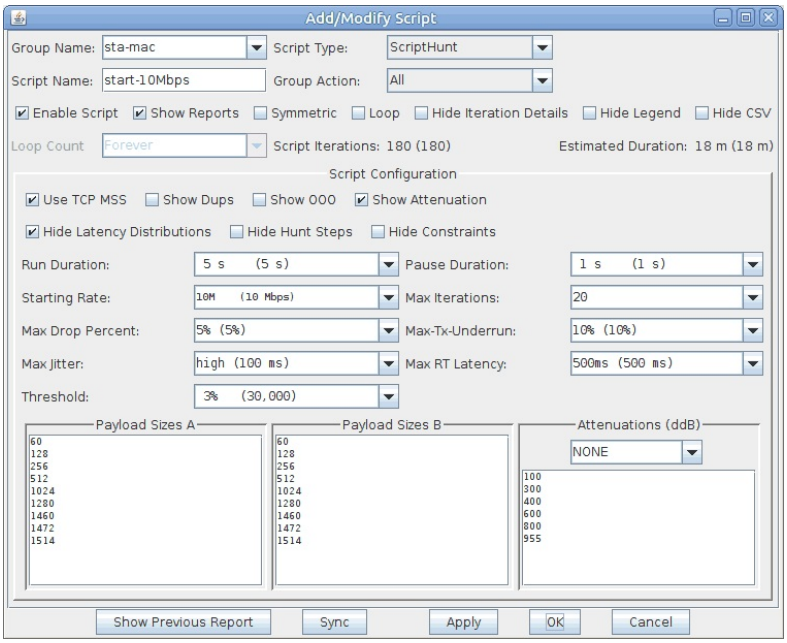
A. Quantity: 9
B. Deselect **Zero Pad**.
C. If only eth1 is used for upstream traffic, set **Port Increment A to 0**. Otherwise leave it at 1.
D. Click **Apply** and close window.
E. Close the Create/Modify Cross Connect window.

6. You will see ten Layer-3 connections in the Layer-3 tab.



7. Create a Test Group. In the Test Group tab, click **Create**.

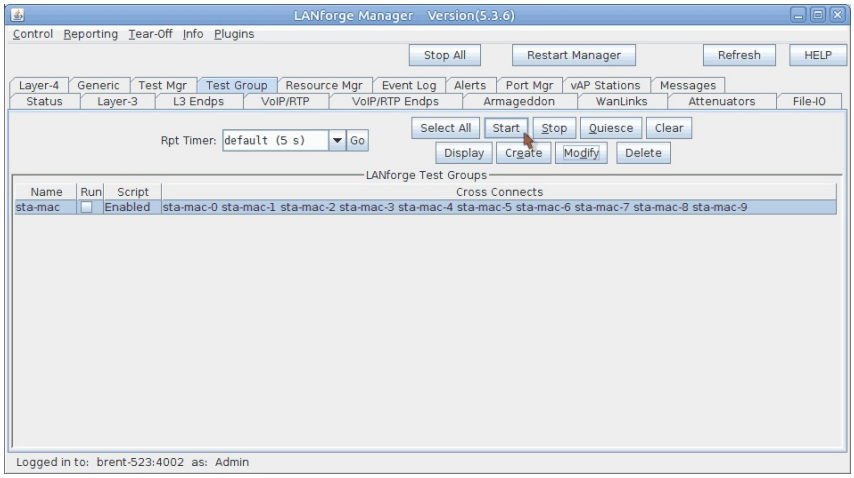
A. Name: **sta-mac**
B. Select **Config As Totals**.
C. Highlight all the **sta-mac-x** connections and click ← **Add Cx**.
D. Click **Apply**.
E. Click **Script**.
F. Configure the Test Group Script.



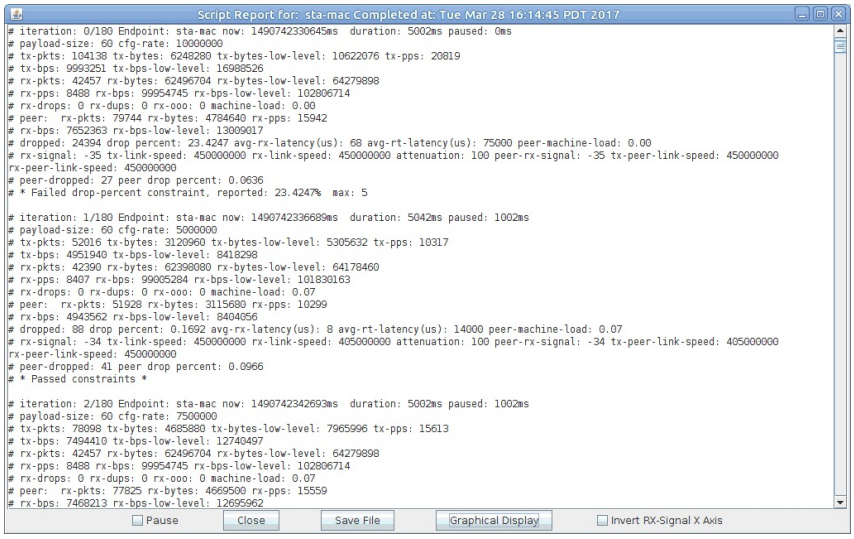
A. Group Name: **sta-mac**
B. Script Type: **ScriptHunt**
C. Script Name: **start-10Mbps**
D. Starting Rate: **10M**
E. Click **OK**.

G. Close the Create Test Group window.

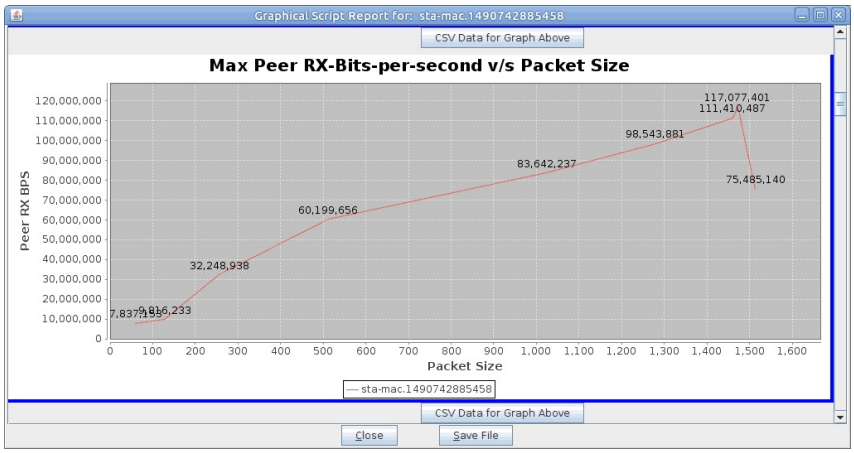
8. Start the test. Highlight the test group and click **Start**.



9. You will see the script report window.



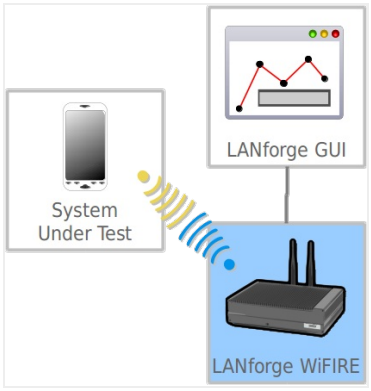
10. When the test is finished, click **Graphical Report** to see graphs.



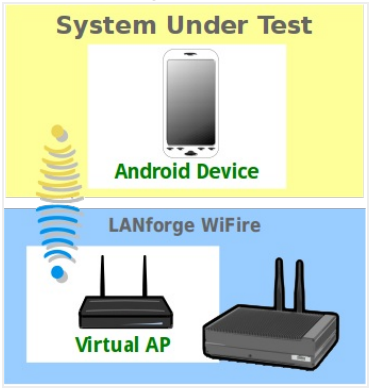
Running UDP Traffic with Android

Goal: Set up Android to be a LANforge resource and then run UDP traffic.

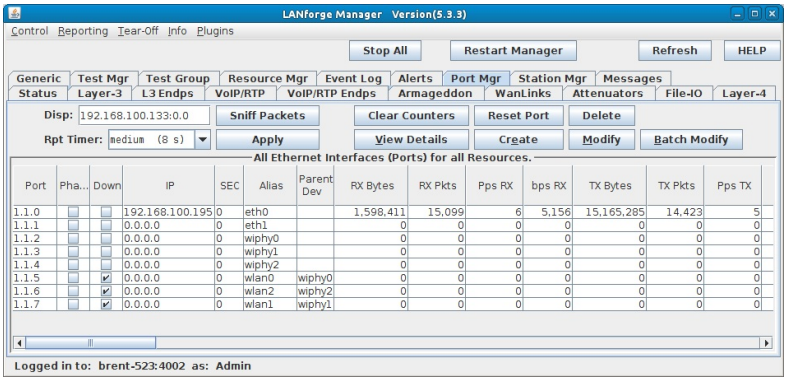
Requires LANforge 5.3.3 or later. Make sure to configure a realm other than 255 for the VAP system, this is because the Android device needs to be in the same realm to be managed. This cookbook will be using realm 195. You can find information on configuring realms using lfconfig starting from step 4 here.



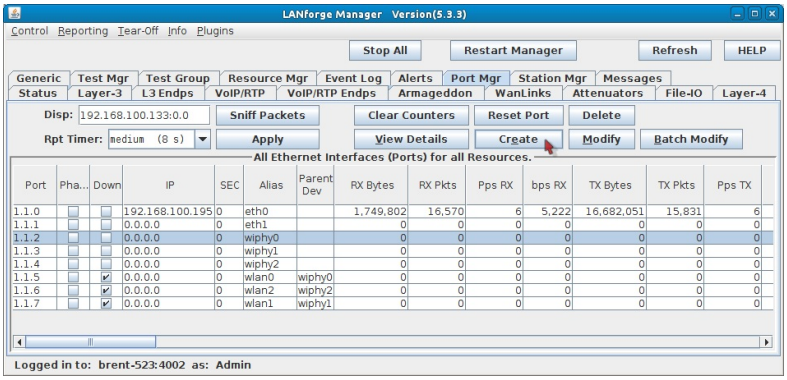
This cookbook will go through installing and configuring LANforge on an Android device, then setting up a Layer-3 UDP connection between the Android device and another LANforge system. Multiple VAPs will be set up to demonstrate roaming. A bridged VAP setup is used (with the bridge as DHCP server) so the device can retain the same IP during roaming, this prevents the device from having to renew its IP saving some time between roams.



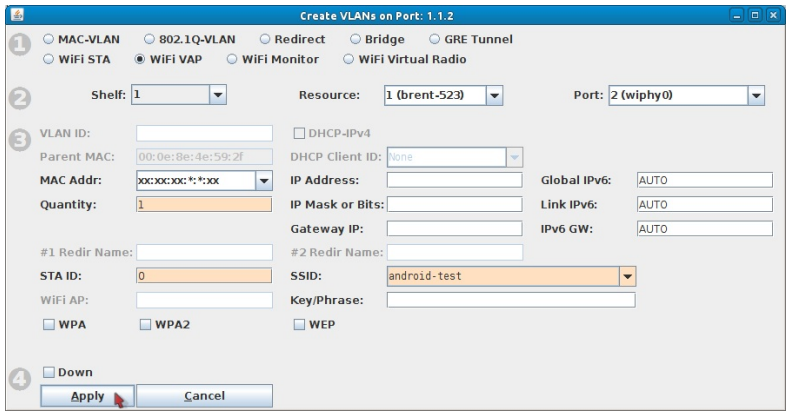
1. First, we will set up the LANforge system so the Android device can connect.
2. Create the first VAP.
    A. Go to the **Port Mgr** tab.



    B. Select wiphy0 and click **Create**.



C. Select **WiFi VAP** and enter in the below values.



A. Quantity `1`

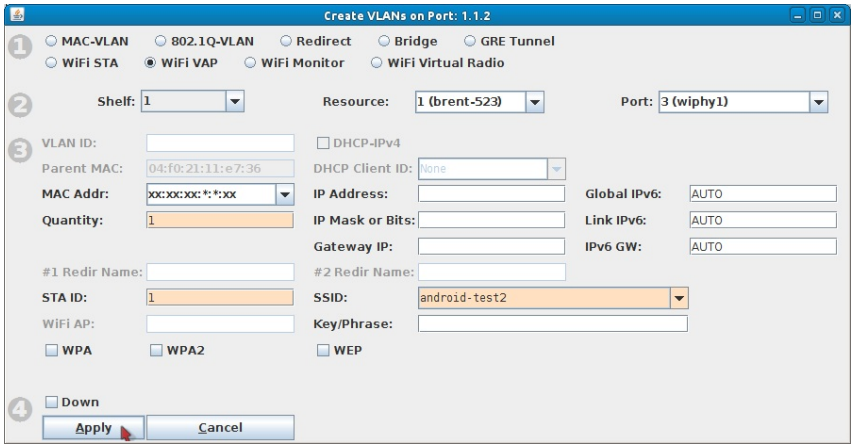
B. STA ID: `0`

C. SSID: `android-test`

D. Click **Apply**.

A. **Note:** we will keep this window open for creating the second VAP.

3. Create the second VAP.



A. Select **wiphy1** from the port drop-down menu.

B. Update the **STA ID** and **SSID** with the below values.

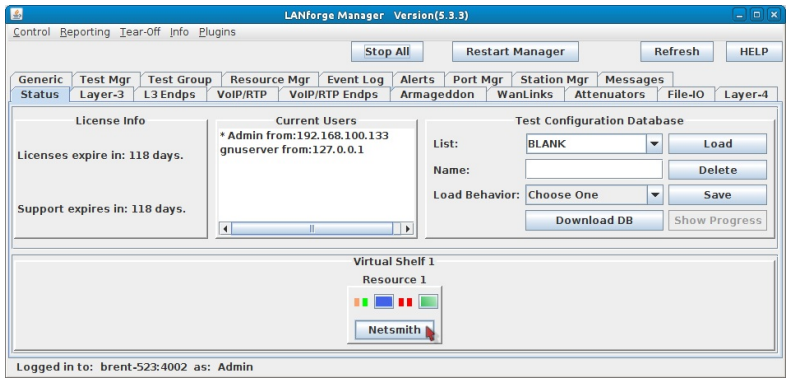
A. STA ID: `1`

B. SSID: `android-test2`

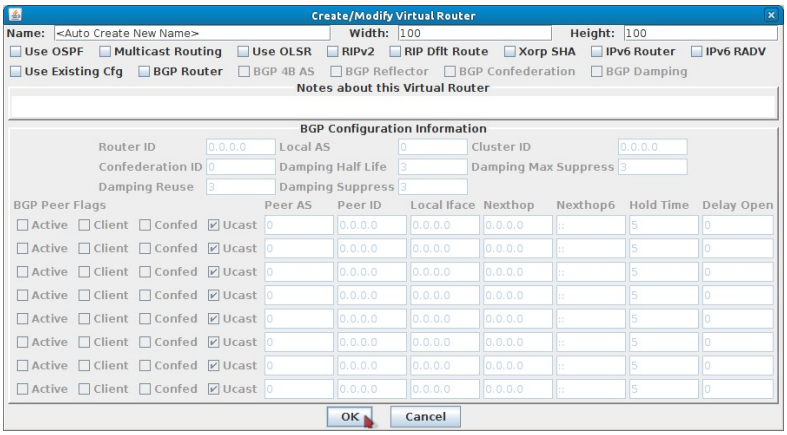
C. Click **Apply** and close the Create Port window.

4. Set up a bridge for both VAPs.

A. Go to the **Status** tab, and click the **Netsmith** button for the AP system (Resource 1 in this example).



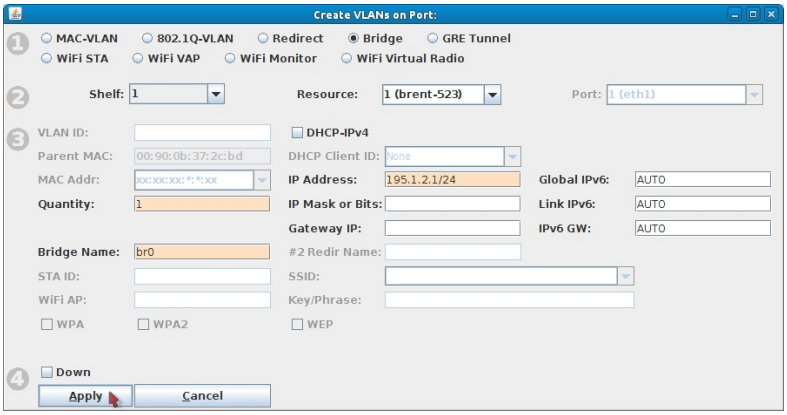
B. Right-click in the Netsmith window and select **New Router**.



A. Click **OK** to accept the default values.

B. Click **Apply** in Netsmith.

C. Right-click in the Netsmith window and select **New Bridge**. Then set the below values.



A. Quantity: `1`

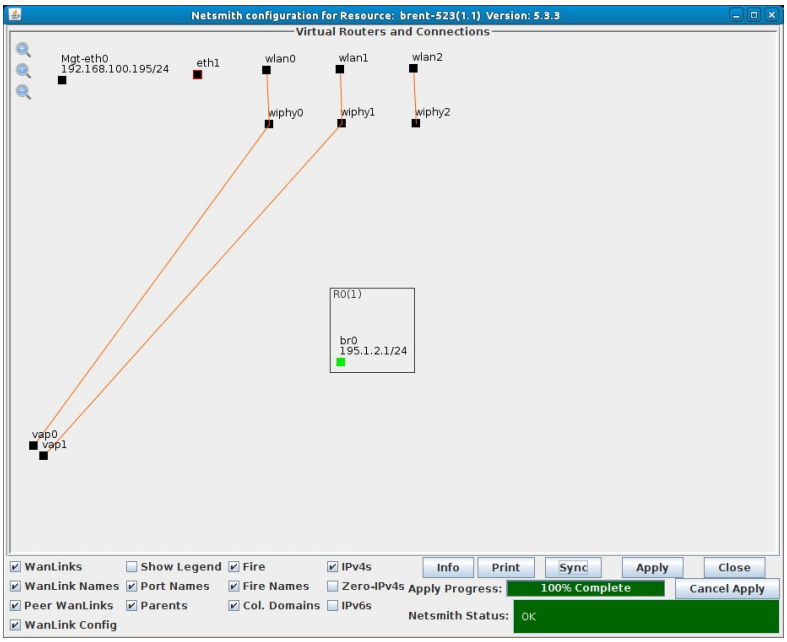
B. Bridge Name: `br0`

C. IP Address: `195.1.2.1/24`

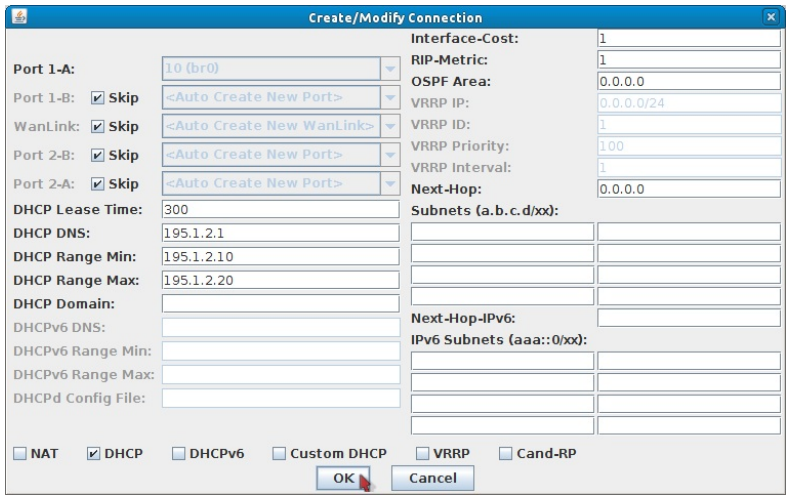
D. Click **Apply** and close the Create Port window.

E. Click **Sync** in Netsmith and the br0 interface should appear.

F. Drag the br0 interface into the virtual router. The interface box should change from black to green.



G. Right click br0 and select modify.



H. Select **DHCP** and use the below values.

A. DHCP Lease Time: `300`

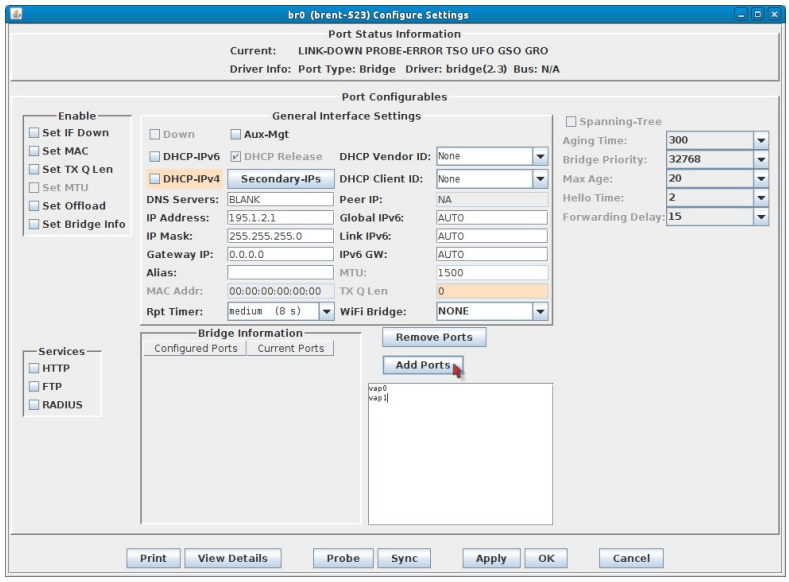
B. DHCP DNS: `195.1.2.1`

C. DHCP Range Min: `195.1.2.10`

D. DHCP Range Max: `195.1.2.20`

I. Click **OK**.

J. Right click br0 and select **Modify Port**.



A. Enter `vap0` and `vap1` into the text box as shown above.

B. Click **Add Ports**.

C. Click **Apply** then **Sync**. The Current Ports column should now show vap0 and vap1.

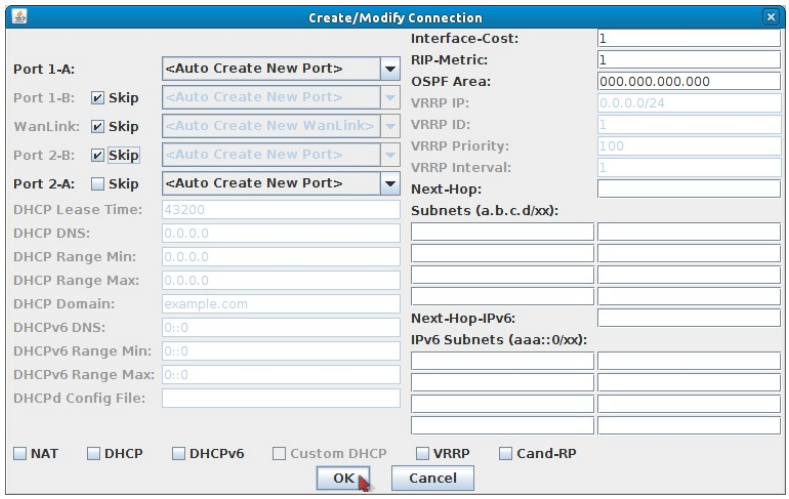
Configured Ports	Current Ports
vap0	vap0
vap1	vap1

D. Click **OK** to close the window.

K. Click **Apply** in Netsmith.

5. Create a redirect-device inside the virtual router. The Android will connect to this port for management purposes.

A. Right-click in the Netsmith window and select **New Connection**.

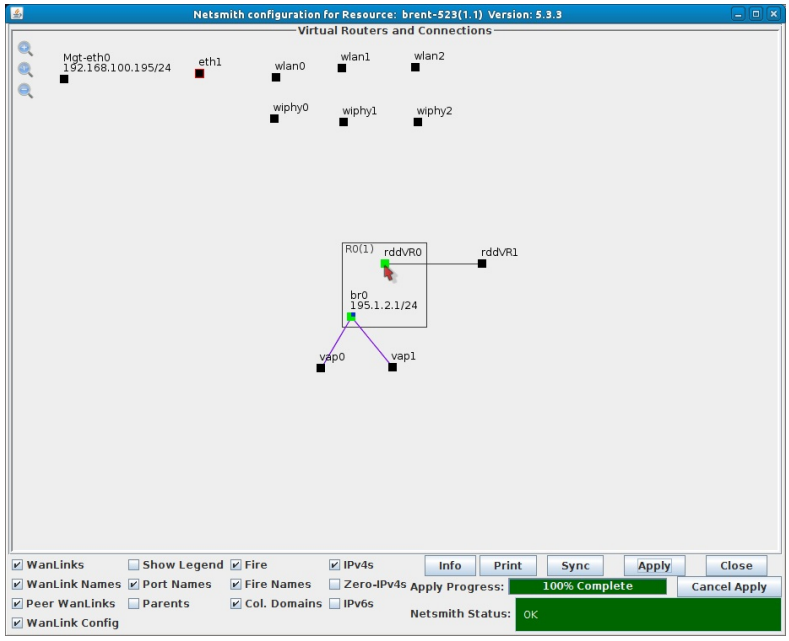


A. Select the **Skip** checkbox for Port 1-B, WanLink, and Port 2-B.

B. Click **OK**

B. Click **Apply** in Netsmith.

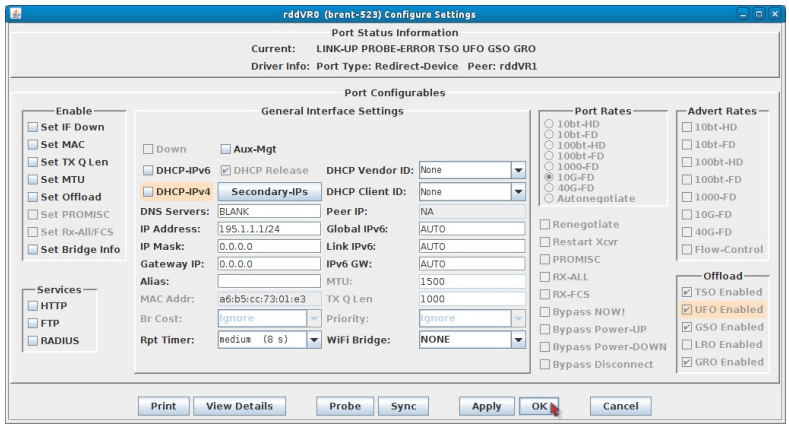
C. Drag one of the rdd interfaces into the virtual router (rddVR0 is used in this example). The interface box should change to green.



A. **Note:** The other rdd (rddVR1 in this case) will not be used and can be ignored.

D. Click **Apply** in Netsmith.

E. Right click rddVR0 and select **Modify Port**.



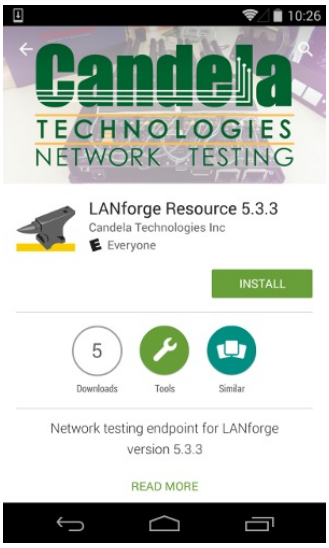
A. Set the **IP Address** to `195.1.1.1/24`

B. Click **OK**.

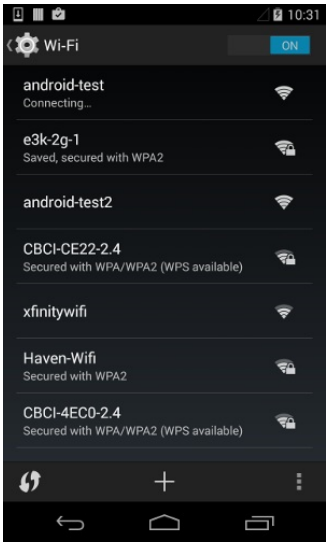
F. Close Netsmith.

6. Set up the Android device.

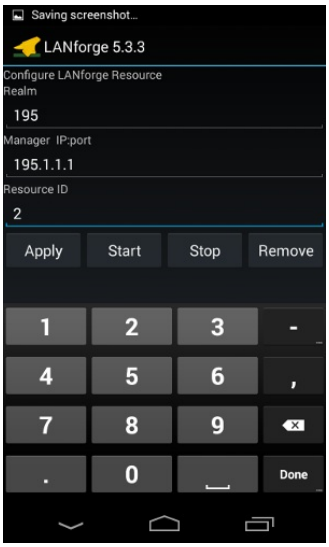
A. Install LANforge Resource from the Google Play Store.



B. Once installed, connect your device to **android-test** using Android's WiFi setup.

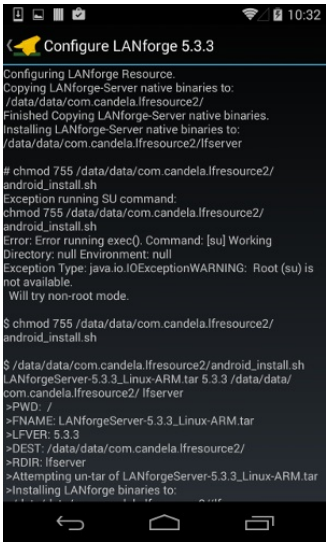


C. Launch the LANforge Resource app and set the values below.

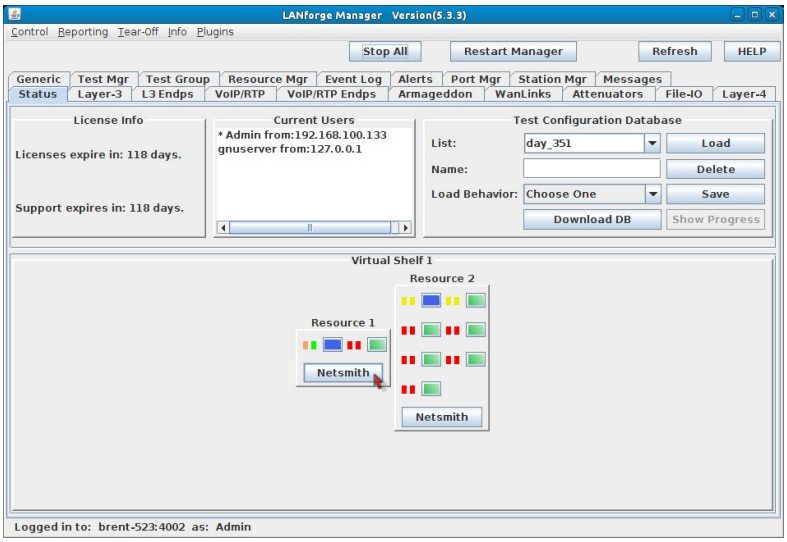


A. Realm `195`
B. Manager IP:port: `195.1.1.1`
C. Resource ID: `2`

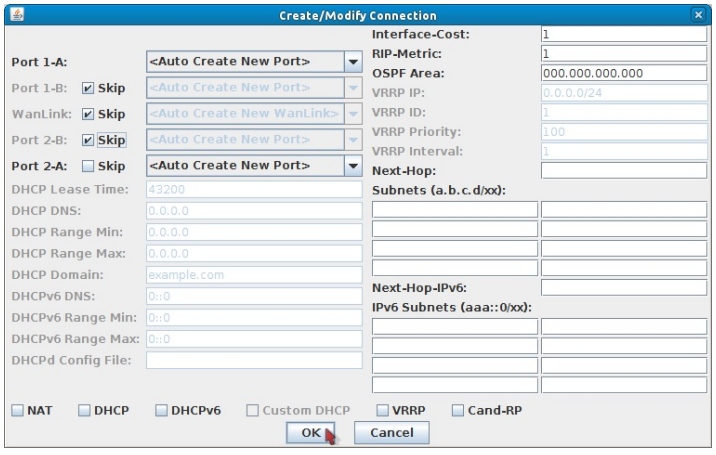
D. Push the **Apply** button to apply changes and start LANforge. After 20-30 seconds the LANforge system should see the Android device connect.



7. Create and run a Layer-3 UDP connection.
   A. Go to the **Status** tab, and click the **Netsmith** button for the AP system (Resource 1 in this example).

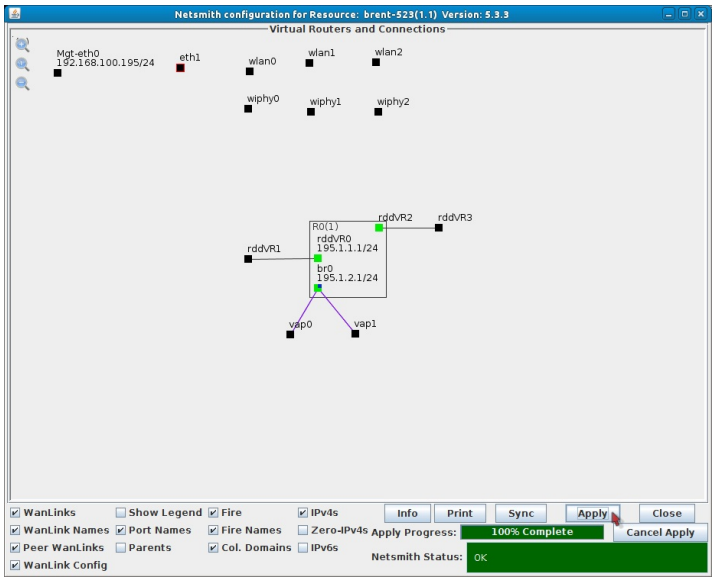


   B. Create a redirect-device inside the virtual router.
      A. Right-click in the Netsmith window and select **New Connection**.



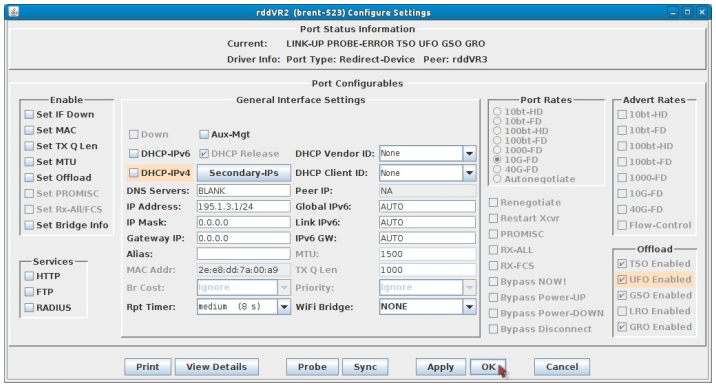
      B. Select the **Skip** checkbox for Port 1-B, WanLink, and Port 2-B.
      C. Click **OK**.
      D. Click **Apply** in Netsmith.

E. Drag rddVR2 into the virtual router. The interface box should change to green.



F. Click **Apply** in Netsmith.

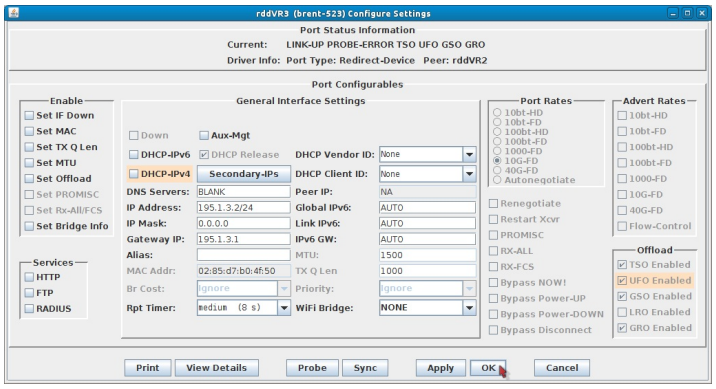
G. Right click rddVR2 and select **Modify Port**.



I. Set the **IP Address** to `195.1.3.1/24`

II. Click **OK**.

H. Right click rddVR3 and select **Modify Port**



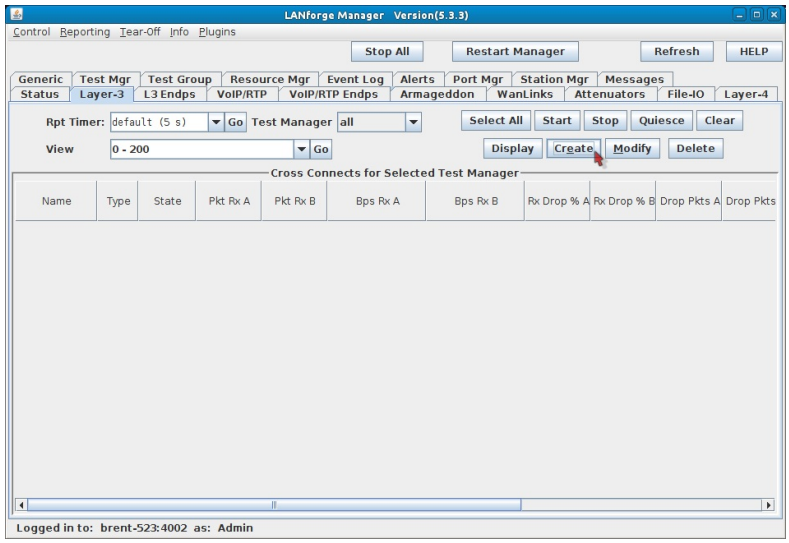
I. Set the **IP Address** to `195.1.3.2/24`

II. Set the **Gateway IP** to `195.1.3.1`

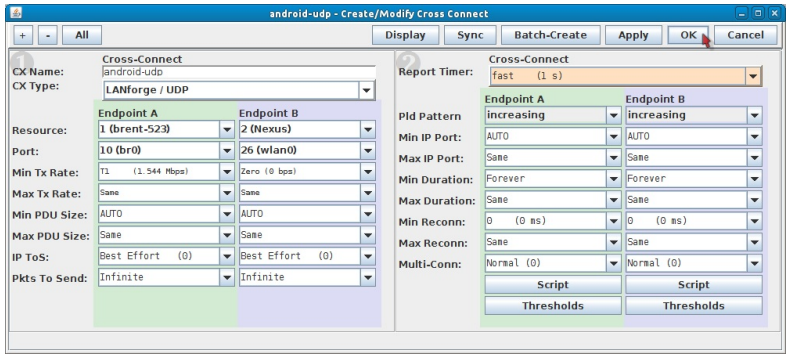
III. Click **OK**.

I. Close Netsmith.

C. Go to the **Layer-3** tab, click **Create**.



D. Set the below values. Note that Endpoint B Resource will differ depending on the Android device type.



A. CX Name: `android-udp`

B. Endpoint A Resource: `brent-523`

C. Endpoint B Resource: `Nexus`

D. Endpoint A Port: `br0`

E. Endpoint B Port: `wlan0`

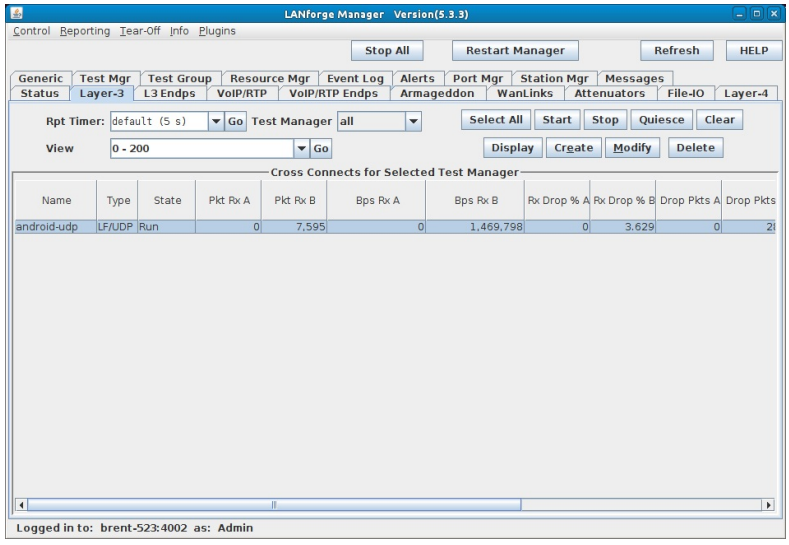
F. Endpoint A Min Tx Rate: `T1 (1.544 Mbps)`

G. Endpoint B Min Tx Rate: `Zero (0 bps)`

H. Report Timer: `fast (1 s)`

I. Click **OK**.

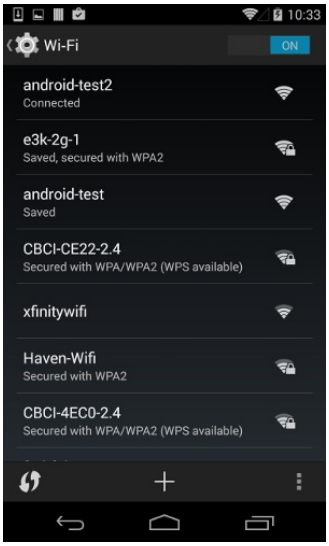
E. Start running traffic.



A. Select the android-udp connection.

B. Click **Start**.

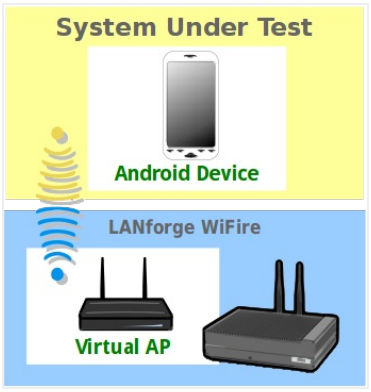
F. Connect your Android device to **android-test2**. Traffic should start flowing again after about 20-30 seconds.

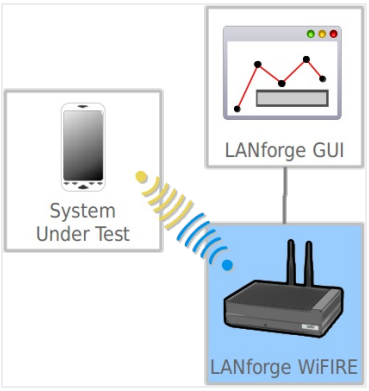


Verifying Android QoS

Goal: Set up Video and Best Effort QoS connections and verify results.

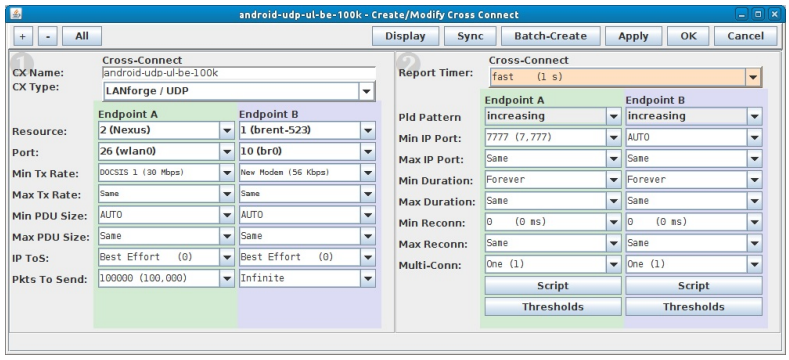
In this example, LANforge-FIRE is used to set up two connections, one running video QoS traffic, the other with Best Effort QoS traffic. The latency and drops will be compared and wireshark will be used to verify QoS information.





1. Connect the Android device to a LANforge system. You can use the cookbook here for tips: Running UDP Traffic with Android
2. Create two connections between the Android device and a LANforge port. One for **Best Effort** QoS traffic, one for **Video** QoS traffic.

A. Creating a Best Effort UDP connection.



A. Set a **name** for the connection (android-udp-ul-be-100k) in this case.

B. Select your **ports**. In this case the connection is between the Android (wlan0) and a bridge (br0).

C. Set **Min Tx Rate** on **Endpoint A** to **DOCSIS 1 (30 Mbps)**.

D. Set **Min Tx Rate** on **Endpoint B** to **New Modem (56 Kbps)**.

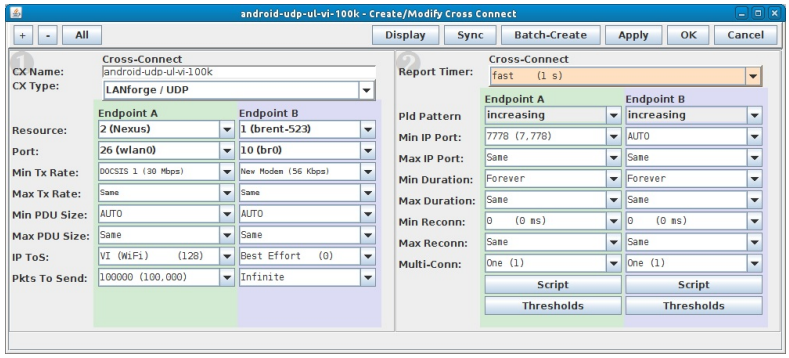
E. Make sure **IP ToS** on **both endpoints** is set to **Best Effort**.

F. Set **Pkts to Send** to **100000** on **Endpoint A**. **Note:** Because Android is being managed in-band, this limit should prevent losing management frames that could contain reporting data.

G. Set **Min IP Port** to **7777** on **Endpoint A**. A custom port is used to help identify the connection type in wireshark.

H. Set **Multi-Conn** to 1 for **both endpoints**. This is so each endpoint is running on its own process.

B. Creating a UDP connection with video QoS traffic.



A. Set a **name** for the connection (android-udp-ul-vi-100k) in this case.

B. Select your **ports**. In this case the connection is between the Android (wlan0) and a bridge (br0).

C. Set **Min Tx Rate** on **Endpoint A** to **DOCSIS 1 (30 Mbps)**.

D. Set **Min Tx Rate** on **Endpoint B** to **New Modem (56 Kbps)**.

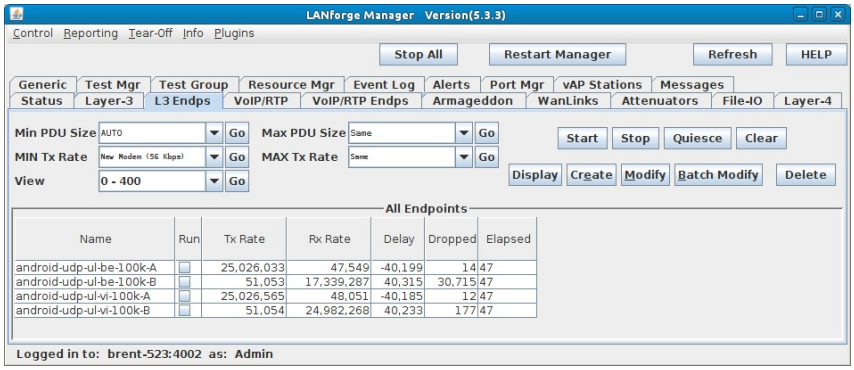
E. Set **IP ToS** on **Endpoint A** to **VI (WiFi)**.

F. Set **Pkts to Send** to **100000** on **Endpoint A**. **Note:** Because Android is being managed in-band, this limit should prevent losing management frames that could contain reporting data.

G. Set **Min IP Port** to **7778** on **Endpoint A**. A custom port is used to help identify the connection type in wireshark.

H. Set **Multi-Conn** to 1 for **both endpoints**. This is so each endpoint is running on its own process.

3. Compare latency and drops.



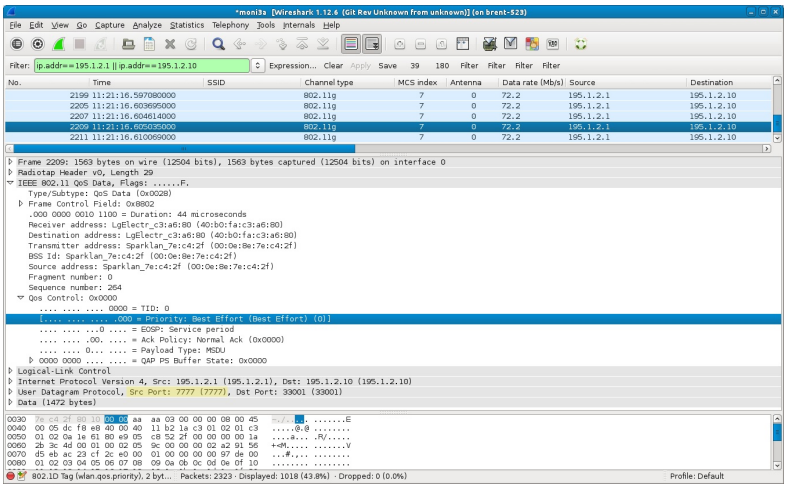
A. The latency can be found under the **Delay** column.

B. The drops can be found under **Dropped** column.

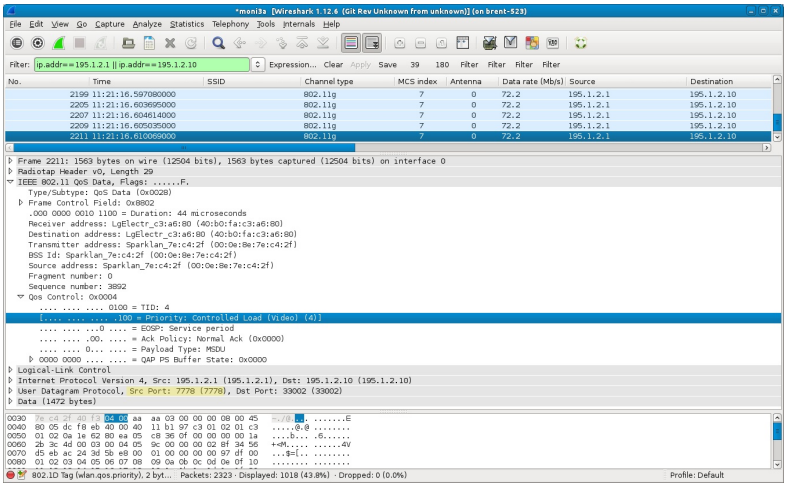
C. If QoS is working properly, you should see **less delay (latency)** and **less drops** for connections using **VI (WiFi) IP ToS**. The delay can be more easily compared if you add endpoint A and B of each connection. As should be expected, the results above show that the particular device is dropping significantly less packets for the connection with video QoS traffic. Because there is a small amount of latency, a major difference isn't shown here.

4. Verify QoS type with wireshark. This is where our custom port setting is helpful.

   A. The packets on port **7777** should show **Best Effort**.



   B. The packets on port **7778** should show **Video**.



ScriptHunt with Test Groups

Goal: Create a Test Group that uses ScriptHunt to maximize the total rate of multiple connections based on desired constraints.

ScriptHunt is a script that runs connections or 'iterates' and changes tx-rates until a maximum acceptable tx-rate is reached. This can be done for multiple payload sizes and attenuations. Using a Test Group along with ScriptHunt allows you to test multiple connections at the same time.

This cookbook assumes ports and connections are already created and configured. The following areas will be covered:

- Creating and Configuring Test Groups

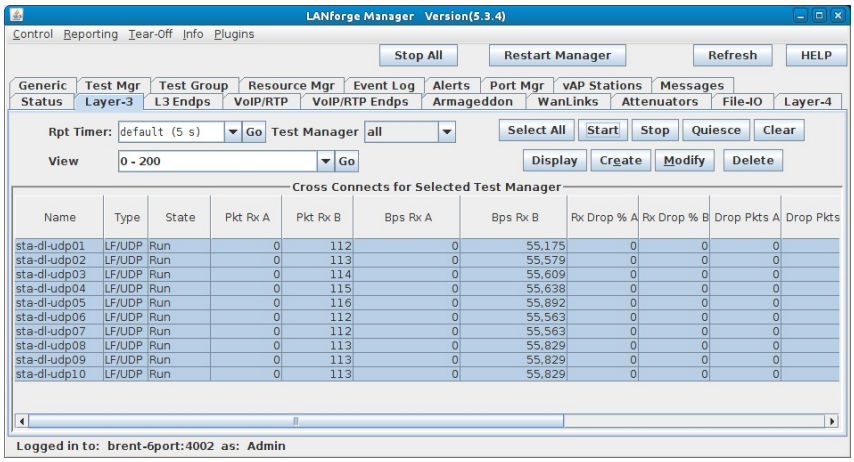
- ScriptHunt setup

- Running ScriptHunt and checking results

- Using additional Test Groups for alternate tests

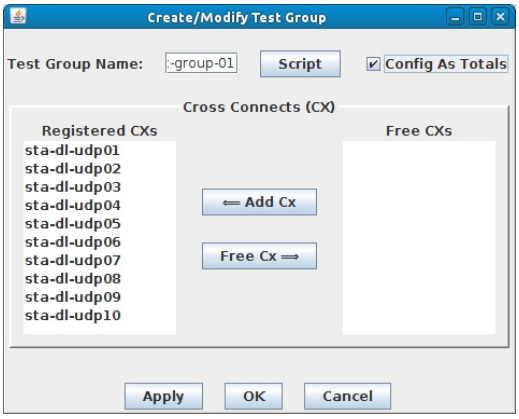
1. It is a good idea to verify connections are running as expected before getting started. The connections

below are testing UDP download on 10 wireless stations.



2. Create and Configure Test Group(s).

   A. In the Test Group tab, click **Create**.



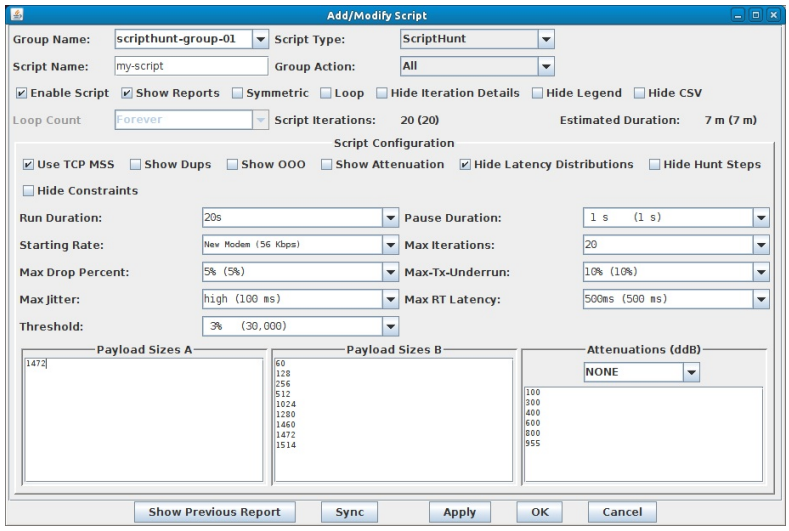
   A. Give the Test Group a **Name**. The name used here is 'scripthunt-group-01'

   B. Under Free CXs, shift-select the connections you want in the group and click **Add Cx**.

   C. Select the **Config As Totals** checkbox. This makes it so the tx-rate is based on a combination of all connections instead of individually.

   D. The Test Group must be created before setting up the script. Click **Apply** to create it.

B. Click **Script** to open the Add/Modify Script window.



A. In the **Script Type** drop-down menu, select **ScriptHunt**.
Note: Most options will be left default for now (all but Run Duration and Payload Sizes A). Below are quick summaries of the drop-down options.

B. **Run Duration**: Determines how long each iteration lasts. This should be about 15-20 seconds minimum. This test will use **20s**.

C. **Pause Duration**: Sets how long the script waits between each iteration.

D. **Starting Rate**: Sets the first tx-rate the script attempts.

E. **Max Iterations**: Allows you to choose a maximum amount of times you want the script to iterate as it adjusts the rate.

F. **Max Drop Percent**: Determines the acceptable range of drops for an iteration. In this case if there are more than 5% drops, the current iteration will fail and the next will attempt to make adjustments so there are less drops.

G. **Max-Tx-Underrun**: With the current setting of 10%, if the actual tx-rate is running 10% slower than the attempted tx-rate, the iteration will fail.

H. **Max Jitter**: Determines the acceptable maximum jitter for an iteration.

I. **Max RT Latency**: This determines the maxiumum acceptable round-trip latency for an iteration.

J. **Threshold**: If the percent difference of tx-rate from iteration to iteration reaches the value here or less, the script will finish unless there are additional payloads/attenuations to test.

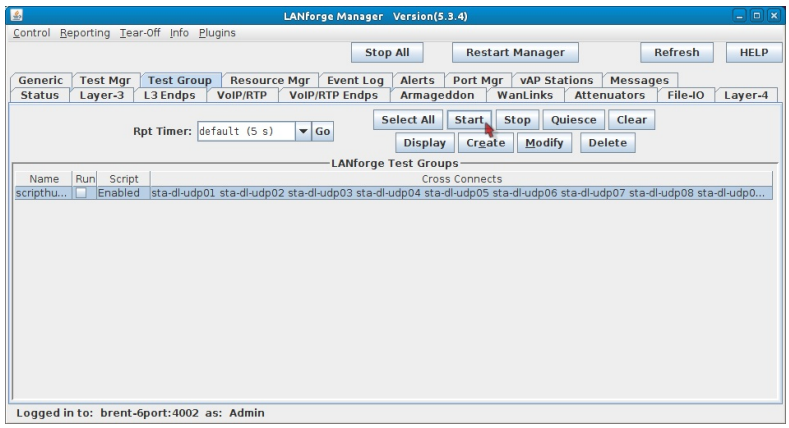
K. ScriptHunt will find the highest acceptable rate for each payload size listed in **Payload Sizes A and B**. The existing values are suggestions, but any desired value(s) can be entered here. If you wish to test Payload Size B, the **Symmetric** checkbox must be enabled. **Attenuations (ddB)** are tested similarly, a Candela attenuator is required and must be connected then selected in the Attenuations drop-down menu. In this specific test, ScriptHunt will just test a payload of 1472 (on endpoint A only) and no attenuations.

L. Click **OK** to add the script.

M. Click **OK** in the Create/Modify Test Group window to close it.

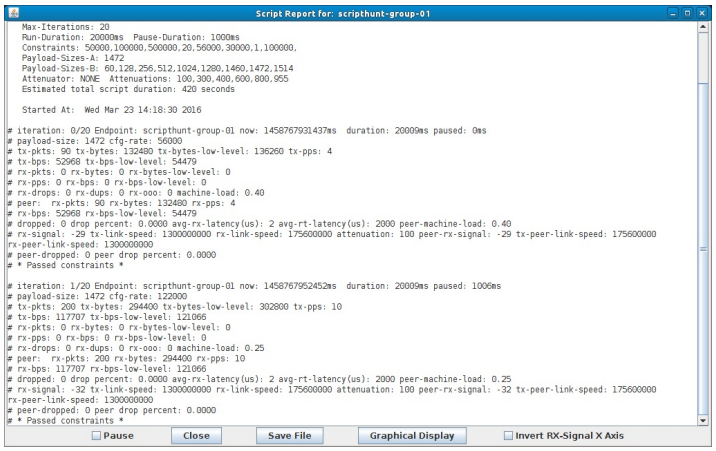
3. Start the Test Group and view the results.

A. Select the Test Group and click **Start**.

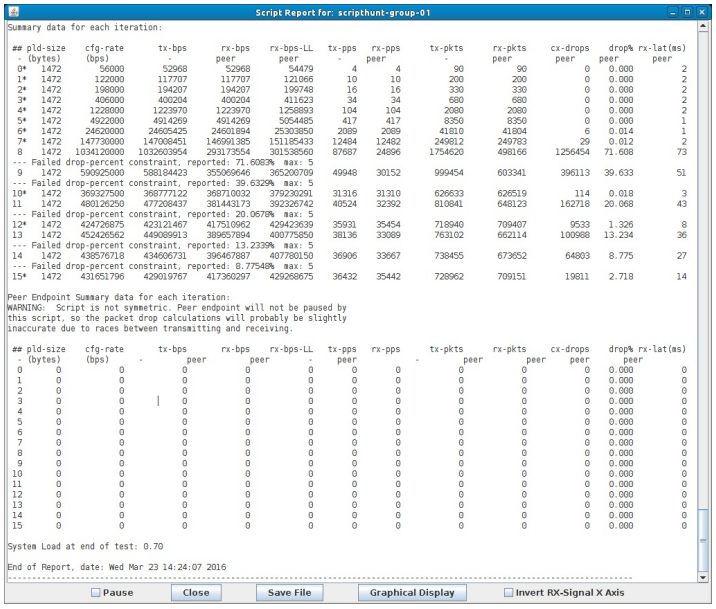


B. Examine the results.

A. After each iteration completes, the results will update with various information on said iteration.

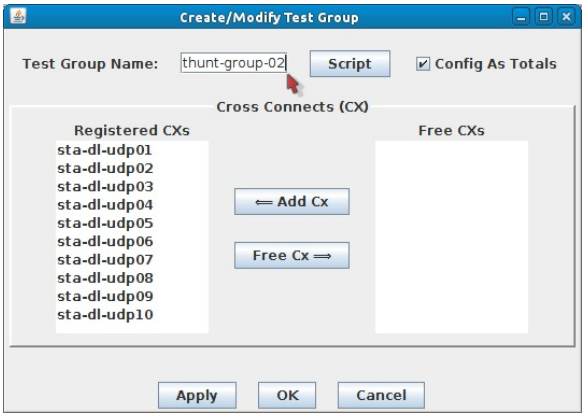


B. When ScriptHunt finishes, the results will show a summary of all iterations. This data shows how high the rates reached and which failure constraints the iterations hit. This particular test reached about 429 Mbps tx-rate within the specified constraints. Notice how the configured rate (cfg-rate (bps) adjusts up and down as the drop percent (drop% peer) goes lower and higher than the 5% constraint set earlier.



4. Create another Test Group with alternate ScriptHunt settings.

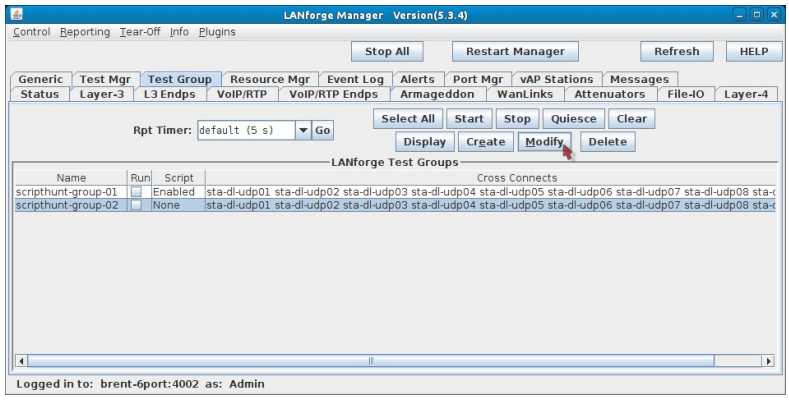
A. We will essentially copy the last Test Group, then change ScriptHunt settings. Select 'scripthunt-group-01' and click **Modify**.



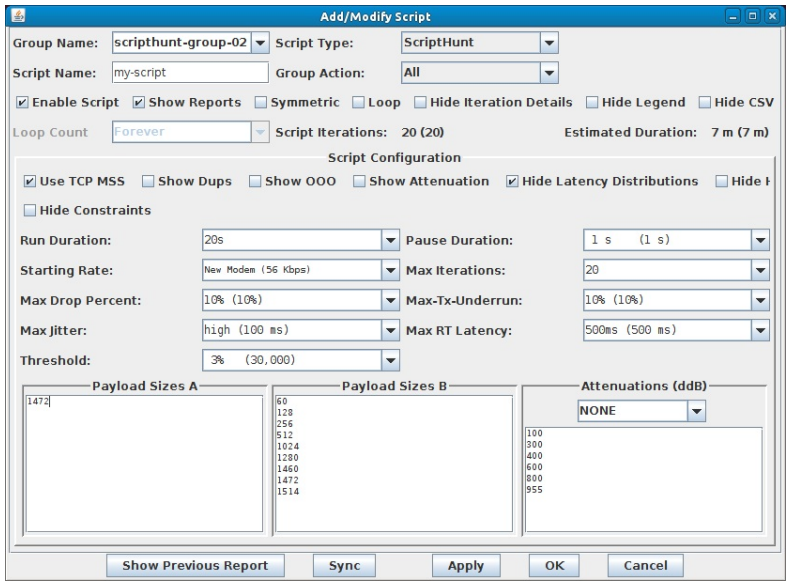
A. Change the **Test Group Name** to 'scripthunt-group-02'.

B. Click **OK**.

B. Select scripthunt-group-02 and click **Modify**.



C. Click **Script**.



A. Set the **Script Type** to **ScriptHunt**.

B. Again, **Run Duration** should be 15-20s minimum. It is set to **20s** for this test.

C. Since the only constraint limiting the first Test Group was **Max Drop Percent**, we'll raise it to **10%** and compare the results.

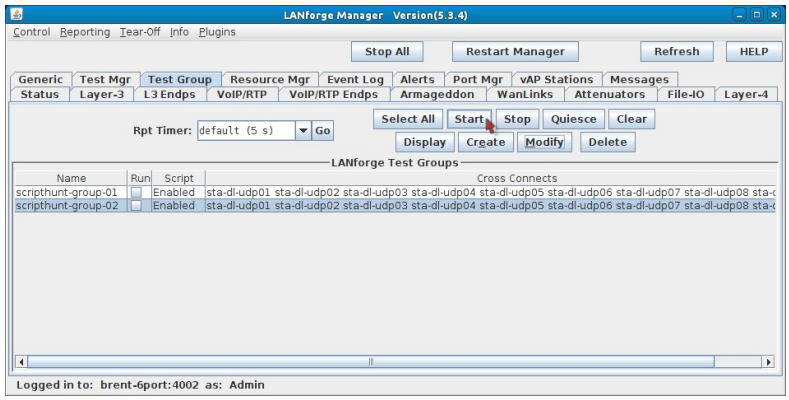
D. Remove all but **1472** in **Payload Sizes A**.

E. Click **OK**.

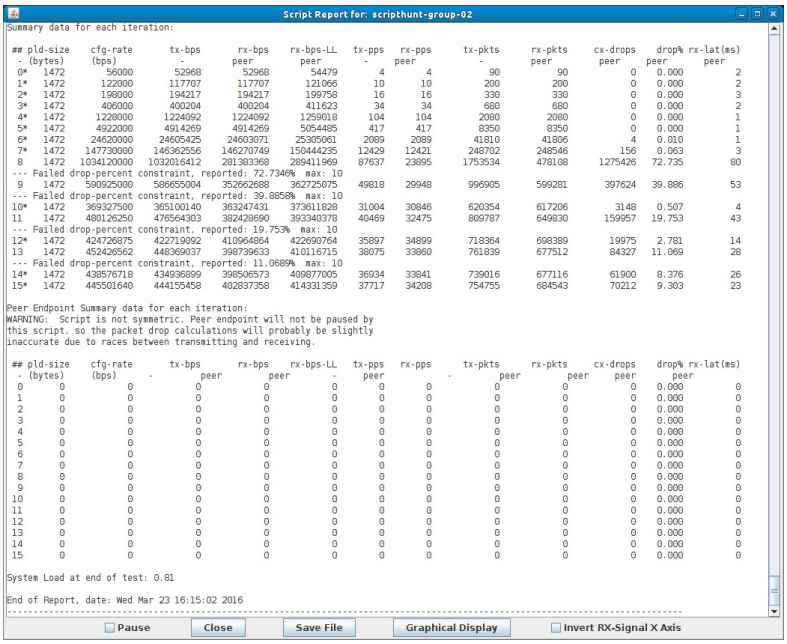
F. Click **OK** in the Create/Modify Test Group window.

D. Be sure 'scripthunt-group-01' is stopped! If you have two different ScriptHunts running on the same connection(s), they will try to fight eachother.

E. Select 'scripthunt-group-02' and click **Start**.



F. Observe the results summary.



Script Report for: scripthunt-group-02

```
Summary data for each iteration:

## pld-size    cfg-rate       tx-bps       rx-bps    rx-bps-LL   tx-pps   rx-pps      tx-pkts      rx-pkts     cx-drops    drop% rx-lat(ms)
  - (bytes)      (bps)            -         peer         peer        -     peer            -         peer         peer     peer         peer
 0*   1472        56000        52968        52968        54479        4        4           90           90            0    0.000            2
 1*   1472       122000       117707       117707       121066       10       10          200          200            0    0.000            2
 2*   1472       198000       194217       194217       199758       16       16          330          330            0    0.000            3
 3*   1472       406000       400204       400204       411623       34       34          680          680            0    0.000            2
 4*   1472      1228000      1224092      1224092      1259018      104      104         2080         2080            0    0.000            1
 5*   1472      4922000      4914269      4914269      5054485      417      417         8350         8350            0    0.000            1
 6*   1472     24620000     24605425     24603071     25305061     2089     2089        41810        41806            4    0.010            1
 7*   1472    147730000    146362556    146270749    150444235    12429    12421       248702       248546          156    0.063            3
 8    1472   1034120000   1032016412    281383368    289411969    87637    23895      1753534       478108      1275426   72.735           80
--- Failed drop-percent constraint, reported: 72.7346%  max: 10
 9    1472    590925000    586655004     35266268     36272507     49818    29948       996905       599281       397624   39.886           53
--- Failed drop-percent constraint, reported: 39.8858%  max: 10
10*   1472    369327500    365100140    363247431    373611828    31004    30846       620354       617206         3148    0.507            4
11    1472    480126250    476564303    382428690    393340378    40469    32475       809787       649830       159957   19.753           43
--- Failed drop-percent constraint, reported: 19.753%  max: 10
12*   1472    424726875    422719092    410964864    422690764    35897    34899       718364       698389        19975    2.781           14
13    1472    452426562    448369037    398739633    410116715    38075    33860       761839       677512        84327   11.069           28
--- Failed drop-percent constraint, reported: 11.0689%  max: 10
14*   1472    438576718    434936899    398506573    409877005    36934    33841       739016       677116        61900    8.376           26
15*   1472    445501640    444155458    402837358    414331359    37717    34208       754755       684543        70212    9.303           23

Peer Endpoint Summary data for each iteration:
WARNING:  Script is not symmetric. Peer endpoint will not be paused by
this script, so the packet drop calculations will probably be slightly
inaccurate due to races between transmitting and receiving.

## pld-size    cfg-rate       tx-bps       rx-bps    rx-bps-LL   tx-pps   rx-pps      tx-pkts      rx-pkts     cx-drops    drop% rx-lat(ms)
  - (bytes)      (bps)         -          peer         peer         -      peer            -          peer         peer     peer         peer
 0        0            0            0            0            0        0        0           0            0            0    0.000            0
 1        0            0            0            0            0        0        0           0            0            0    0.000            0
 2        0            0            0            0            0        0        0           0            0            0    0.000            0
 3        0            0            0            0            0        0        0           0            0            0    0.000            0
 4        0            0            0            0            0        0        0           0            0            0    0.000            0
 5        0            0            0            0            0        0        0           0            0            0    0.000            0
 6        0            0            0            0            0        0        0           0            0            0    0.000            0
 7        0            0            0            0            0        0        0           0            0            0    0.000            0
 8        0            0            0            0            0        0        0           0            0            0    0.000            0
 9        0            0            0            0            0        0        0           0            0            0    0.000            0
10        0            0            0            0            0        0        0           0            0            0    0.000            0
11        0            0            0            0            0        0        0           0            0            0    0.000            0
12        0            0            0            0            0        0        0           0            0            0    0.000            0
13        0            0            0            0            0        0        0           0            0            0    0.000            0
14        0            0            0            0            0        0        0           0            0            0    0.000            0
15        0            0            0            0            0        0        0           0            0            0    0.000            0

System Load at end of test: 0.81

End of Report, date: Wed Mar 23 16:15:02 2016
```

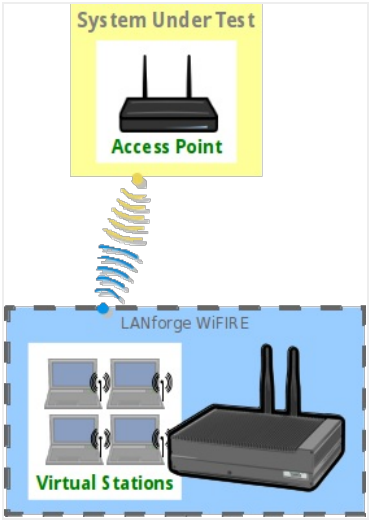
Pause | Close | Save File | Graphical Display | Invert RX-Signal X Axis

A. Compare the two results final tx-bps. The first test group got 429 Mbps at 5% maximum drops. Because the second Test Group's ScriptHunt was set at 10% maximum drops, it was able to achieve a higher tx-bps of 444 Mbps.

WPS Association

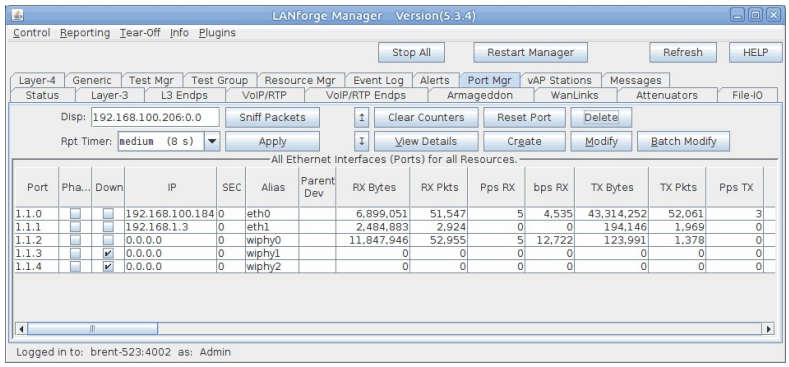
Goal: Associate a station to an AP with the WPS Push and PIN methods.

In this test scenario, a LANforge CT523 is used to simulate a station connecting to a commercial AP with WPS Push mode and also with WPS PIN mode. Currently this test requires the use of a terminal on the LANforge system to run some WPS commands.

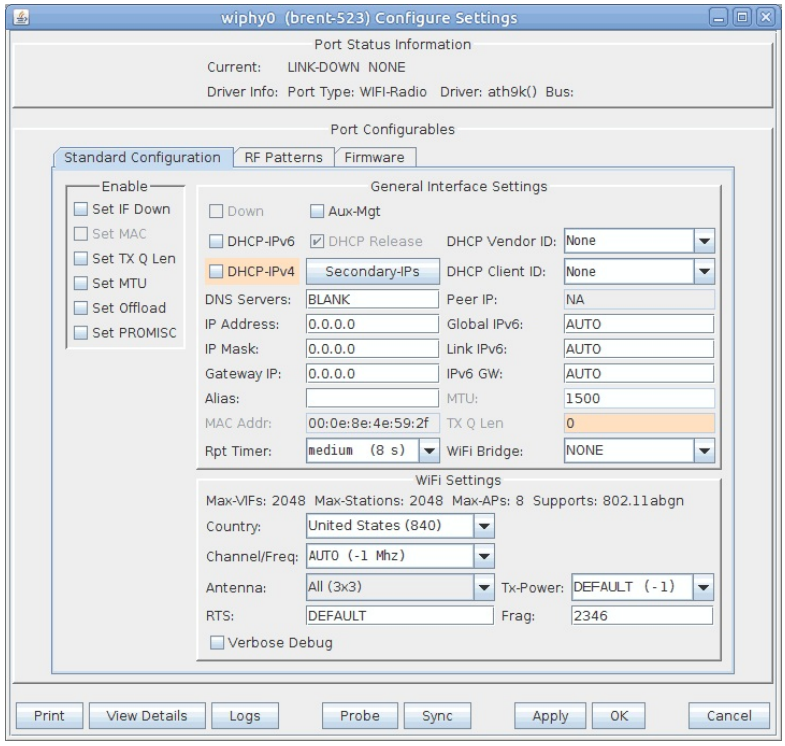


1. Create a virtual wireless station.

A. Go to the **Port Mgr** tab.



B. Check wiphy0 settings by selecting wiphy0 and click **Modify**.

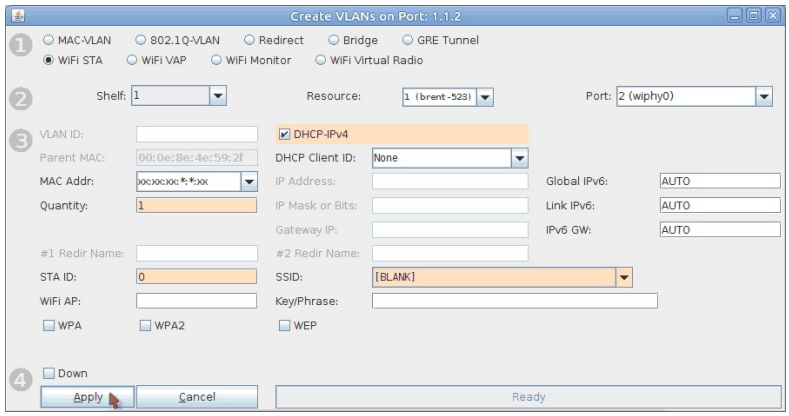


A. If the **Down** checkbox is selected, Click **Set IF Down** on the left, then uncheck the **Down** checkbox.

B. Make sure **Channel/Freq** is set to **AUTO**.

C. Click **OK**.

C. Back in the **Port Mgr** tab, select port wiphy0 and click **Create**.



A. Select **WiFi STA**.

B. Select **DHCP-IPv4**.

C. Set **Quantity** to **1**.

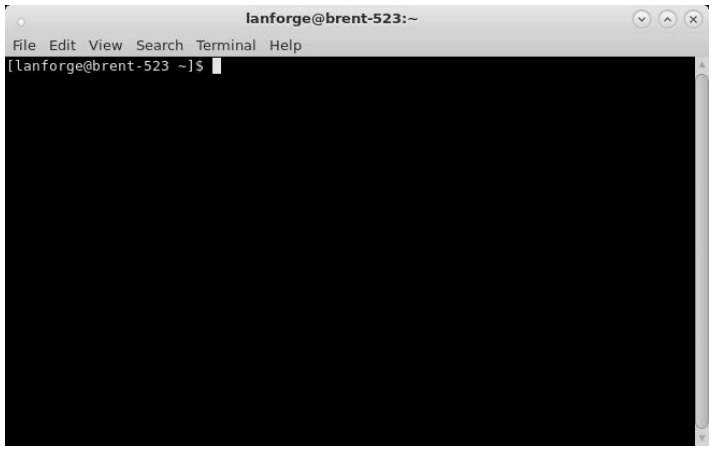
D. Set **STA ID** to **0**.

E. Set **SSID** to **[BLANK]**. Don't forget to add the brackets.

F. Click **Apply**.

2. Create a wpa_supplicant.conf file.

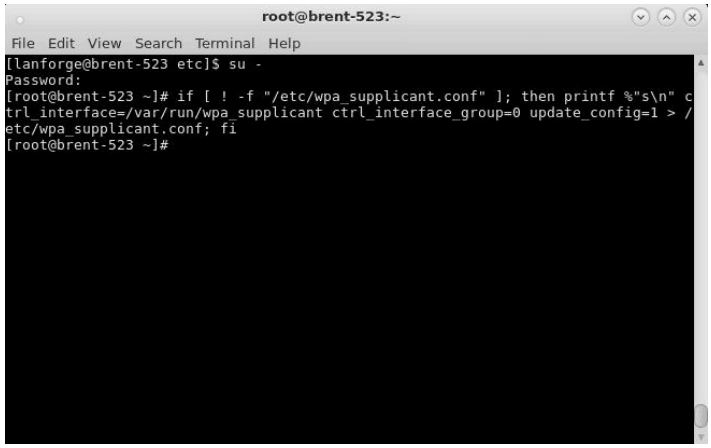
A. Open a terminal on the LANforge system.



B. Create the file wpa_supplicant.conf in /etc that contains the below text. Alternatively, the command in the next step can be used to create the file.
`ctrl_interface=/var/run/wpa_supplicant`
`ctrl_interface_group=0`
`update_config=1`

C. In the open terminal run the below command to create the wpa_supplicant.conf file.
NOTE: Please make sure wpa_supplicant.conf doesn't already exist in the **/etc** directory.

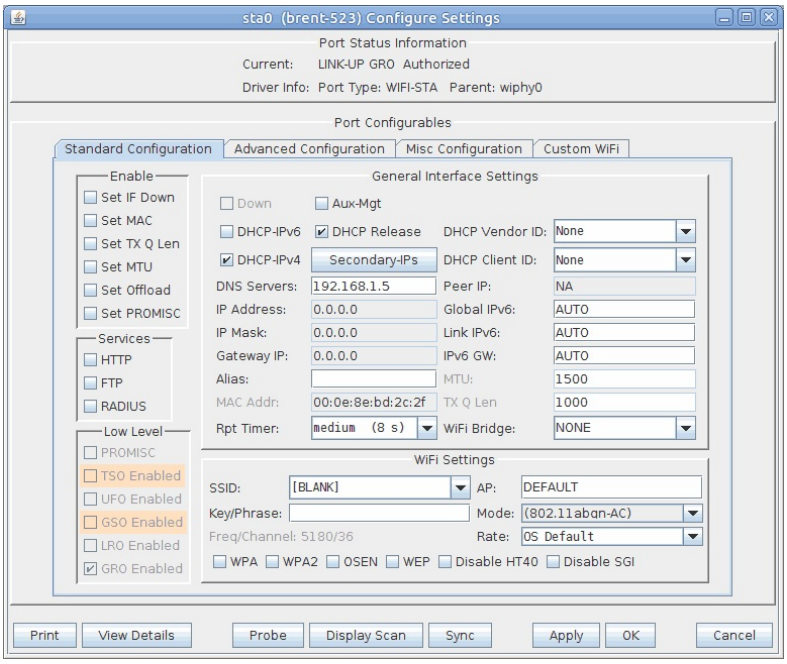
First become root: `su -`
Then run:
`if [ ! -f '/etc/wpa_supplicant.conf' ]; then printf %'s\n' ctrl_interface=/var/run/wpa_supplicant ctrl_interface_group=0 update_config=1 > /etc/wpa_supplicant.conf; fi`



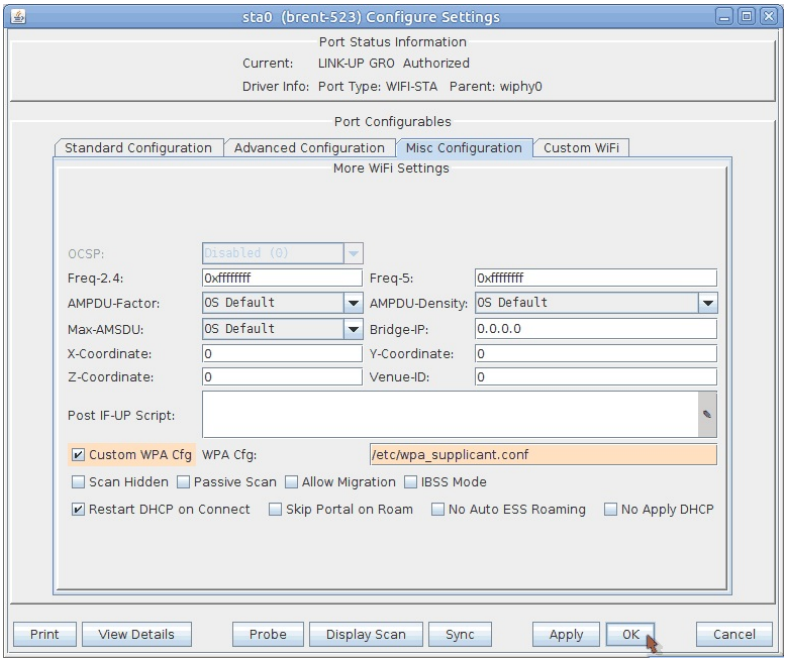
D. Keep in mind the wpa_supplicant.conf file will save AP information for stations. If you want a station to forget the AP information, you'll need to recreate the wpa_supplicant.conf file.

3. Configure sta0 to use a the wpa_supplicant.conf file created in step 2.

A. Select **sta0** in the **Port mgr** tab and click **Modify**.



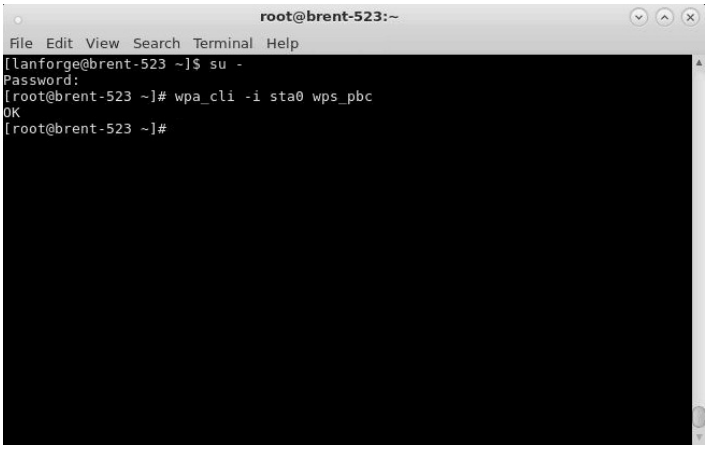
B. In the **Misc Configuration** tab, select **Custom WPA Cfg** then inside the text field for **WPA Cfg**, add the path of the wpa_supplicant.conf file. In this case, we'll be using **/etc/wpa_supplicant.conf**



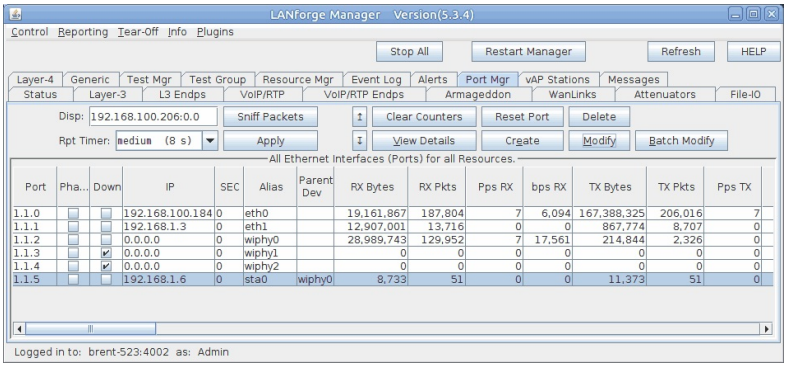
C. Click **OK**.

4. Associate the station using **WPS Push mode**.

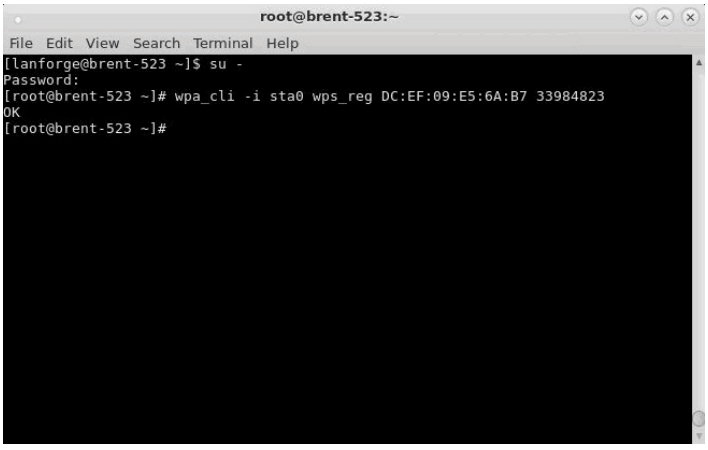
A. In a terminal on the LANforge system, enter in the below command and then push your AP's WPS button. If you aren't root, use `su -`
`wpa_cli -i sta0 wps_pbc`



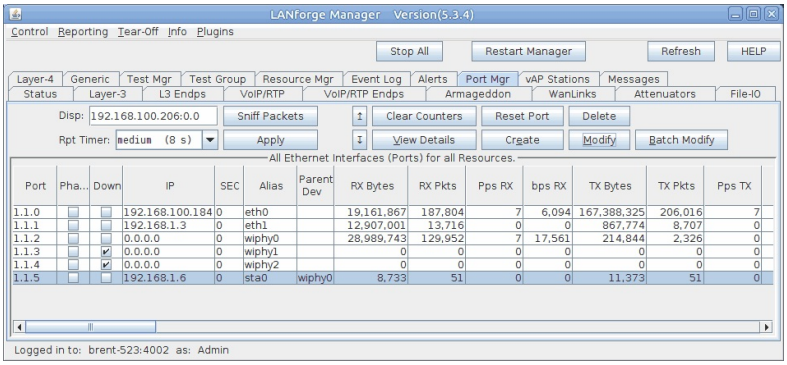
B. Your station should now associate and get an IP.
Note: The SSID field in the sta0 Modify window will remain as [BLANK]. You also may see a different SSID in the View Details window, this is a bug and can be ignored for now as long as the BSSID is correct.



5. Associate the station using **WPS PIN mode**.
   A. Obtain your AP's BSSID and WPS PIN. This information can usually be found on the AP label or in the AP's software.
   B. In a terminal on the LANforge system, enter in the below command. Replace the x's with the BSSID and numbers with the PIN. If you aren't root, use `su -`
   `wpa_cli -i sta0 wps_reg xx:xx:xx:xx:xx:xx 12345678`



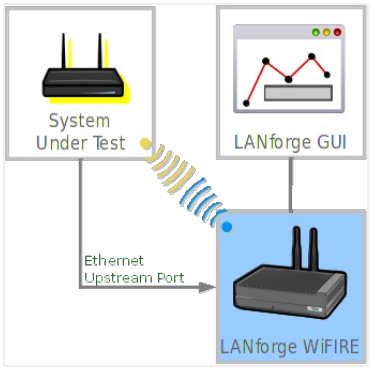
C. Your station should now associate and get an IP.
Note: The SSID field in the sta0 Modify window will remain as [BLANK]. You also may see a different SSID in the View Details window, this is a bug and can be ignored for now as long as the BSSID is correct.

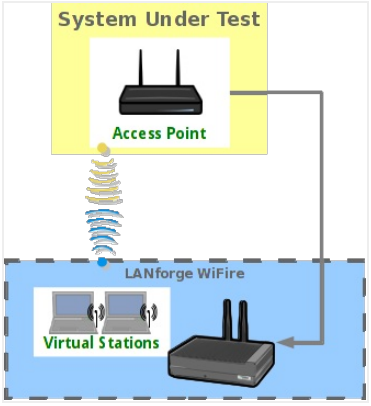


Test WiFi MU-MIMO Download.

Goal: Test WiFi MU-MIMO station Download, one 2x2 station, one 1x1.

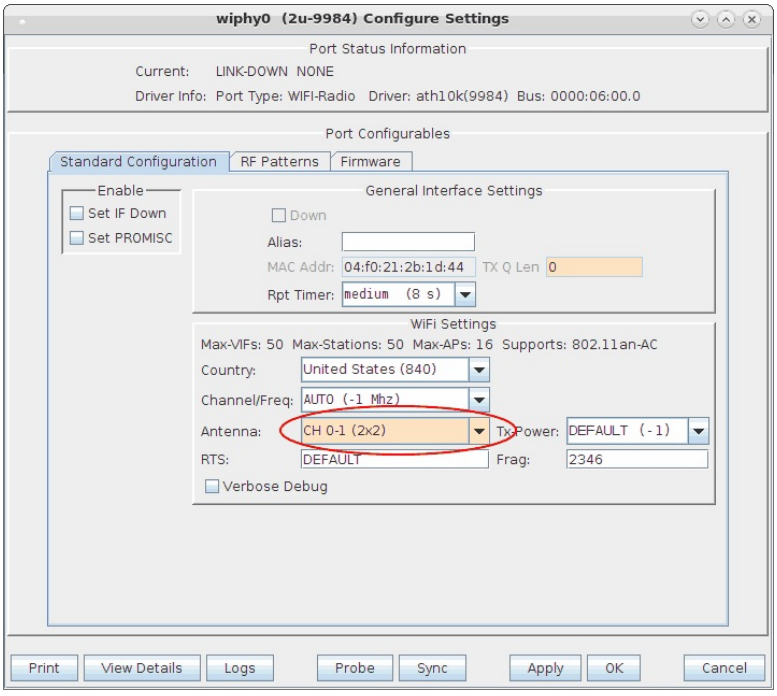
Test WiFi MU-MIMO station Download using two MU-MIMO capable radios. One radio will emulate a 2x2 station, and a second will emulate a 1x1 station. When testing MU-MIMO, only a single station can be used per radio. For additional non-MU-MIMO station emulation, additional radios can be configured for multiple station virtual devices. This example uses a system similar to the LANforge CT525 system. It is configured with 4 radios: Two of the 4x4 MU-MIMO radios are used for MU-MIMO testing. The other two are not used in this test scenario. This procedure should work on any system that can support at least 2 of the 4x4 wave-2 radios. The AP in this test is a Netgear R7800 configured in bridging mode. **This feature requires 2 wave-2 WiFi network cards and LANforge release 5.3.5 or higher.**



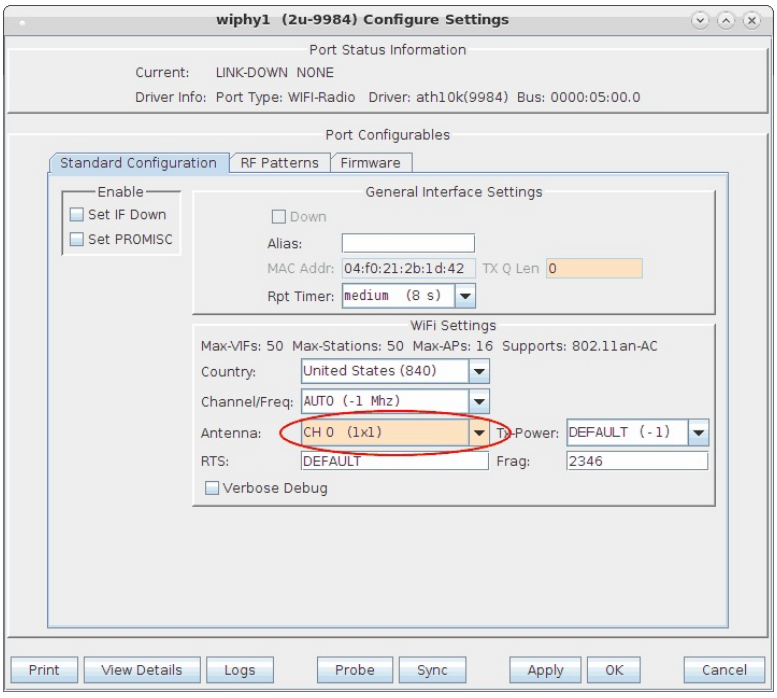


1. Configure Radios and Station devices for MU-MIMO capabilities.

A. Go to the Port Manager, select the **wiphy0** interface, and click **Modify**. Configure the radio for 2x2 MIMO and click Apply.

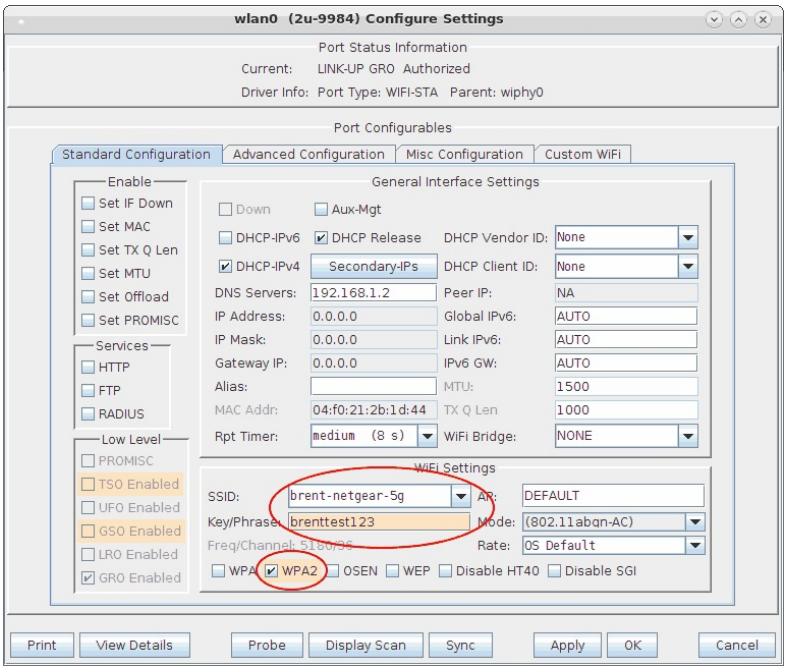


B. Select the **wiphy1** interface, and click **Modify**. Configure the radio for 1x1 MIMO and click Apply.

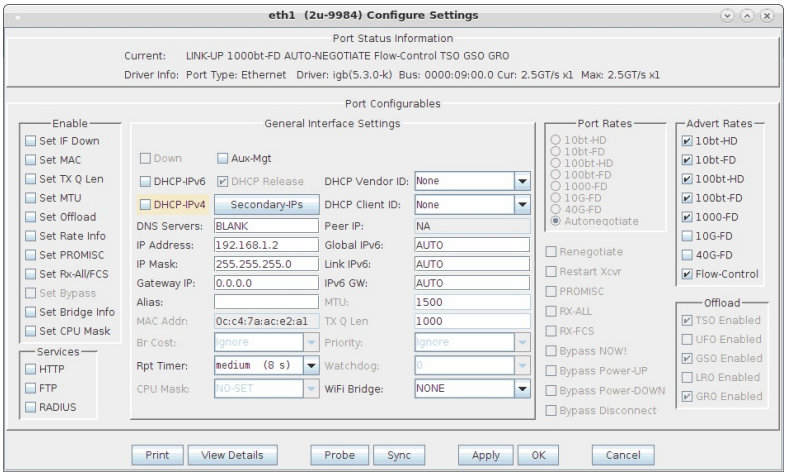


C. For both wiphy0 and wiphy1 ensure that the firmware is configured properly for MU-MIMO. The Port Status Information section at the top should mention the 9984 chipset, as other hardware may not support MU-MIMO. Normally the best option is to go into the **Firmware** tab, click the **Customize Firmware** box, click the top **Firmware Defaults for chipset: 9984** button, and then select **Allow MU-MIMO**. Please note that selecting MU-MIMO disables a feature that allows multiple virtual stations to work properly on a single radio. So, when you are done with MU-MIMO testing, you should probably change this selection back to **Software Decrypt** settings.

D. Select the **wlan0** interface, and click **Modify**. Configure the station for proper SSID, password, etc, and click Apply. Do the same configuration for **wlan1**.

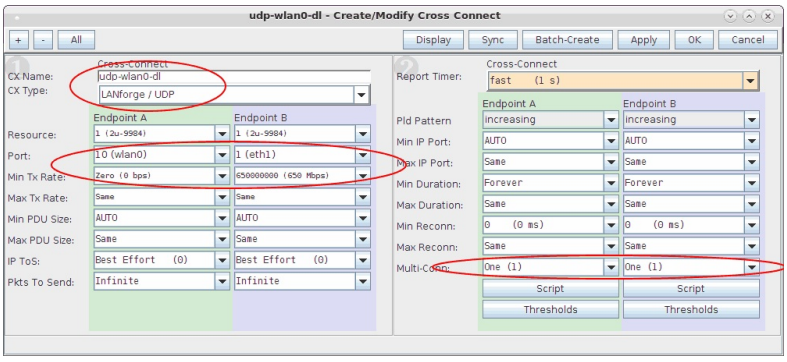


E. In this scenario, we are using eth1 as the upstream port. Ensure it is configured properly. In this example, it is actually configured to serve DHCP using a virtual router and the Netsmith feature in LANforge, but for simplicity, it is normally best if you use the AP as DHCP server or just use fixed IP addresses for eth1 and the wlan interfaces instead of using DHCP.

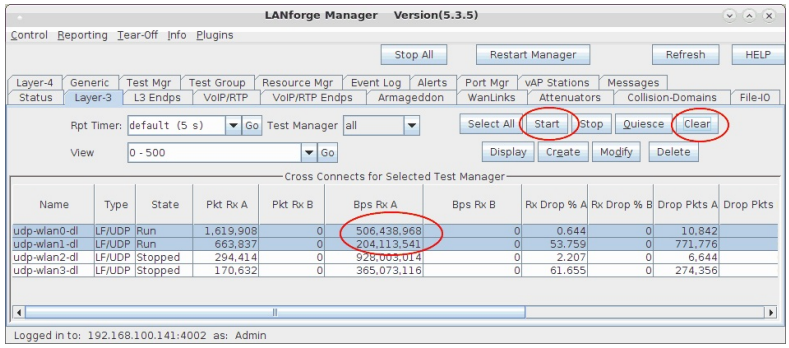


2. Create Layer-3 UDP Download traffic flows.

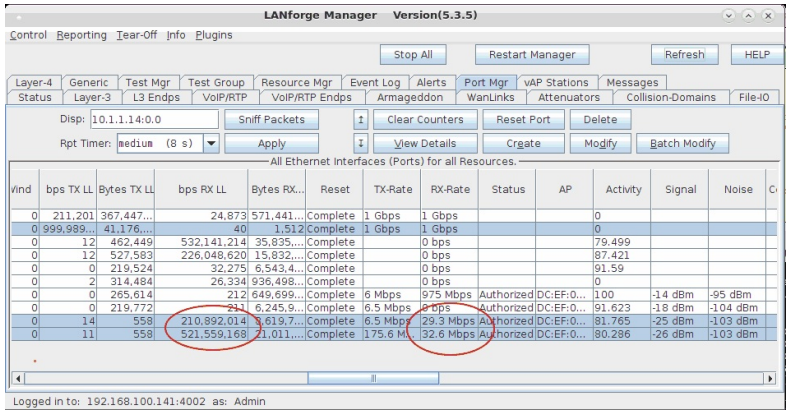
A. Go to Layer-3 tab and click **Create** to build a UDP connection. Select the Protocol, ports, rates, and use Multi-Conn 1 so that separate processes are created for optimal throughput performance. Create a second one for the wlan1 interface, with download speed of about 450Mbps since it is only 1x1 MU-MIMO. You may need to adjust the **+ -** buttons at top left to show the section containing Multi-Conn settings.



B. Start the test by selecting the connections click **Start**. We see about 500Mbps on wlan0 (2x2) and 200Mbps on wlan1 (1x1). For best results, you may need to tune orientation of the first two antenna on the wiphy0 radio and the first antenna on wiphy1. In addition, it can take a short amount of time for the rates to reach maximum throughput, so you may wish to clear the counters after around 15 seconds of running to make sure the averages do not include the initial ramp-up time.



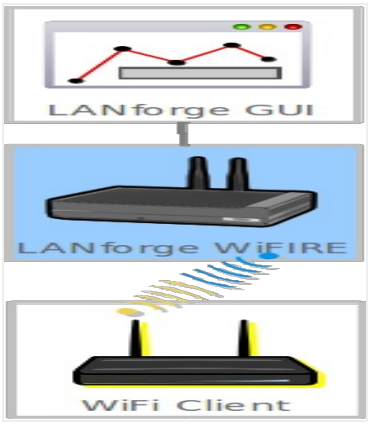
C. It can be a bit difficult to know if MU-MIMO is working properly. In general, if you disable MU-MIMO in the AP, then aggregate throughput should decrease significantly. In addition, the current firmware and/or driver is unable to properly report RX encoding rates for MU-MIMO frames, so it always reports low rates. If you see total throughput that is greater than the reported RX Rate, then likely the system is receiving MU-MIMO frames from the AP.



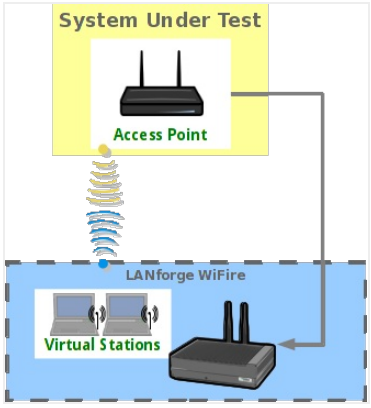
WiFi Capacity Test with Layer 4-7

Goal: Use the WiFi Capacity Test plugin to emulate layer 4-7 traffic from ten virtual stations across an access point and report the results.

Requires 5.2.13 or later. This cookbook will go through setting up a VAP (Virtual Access Point) as an HTTP server, and creating/configuring 10 virtual stations to communicate with the VAP. It will also go through the setup of the WiFi Capacity Test LANforge-GUI plugin to have the virtual stations emulate downloading a file using Layer 4-7 endpoints in LANforge.



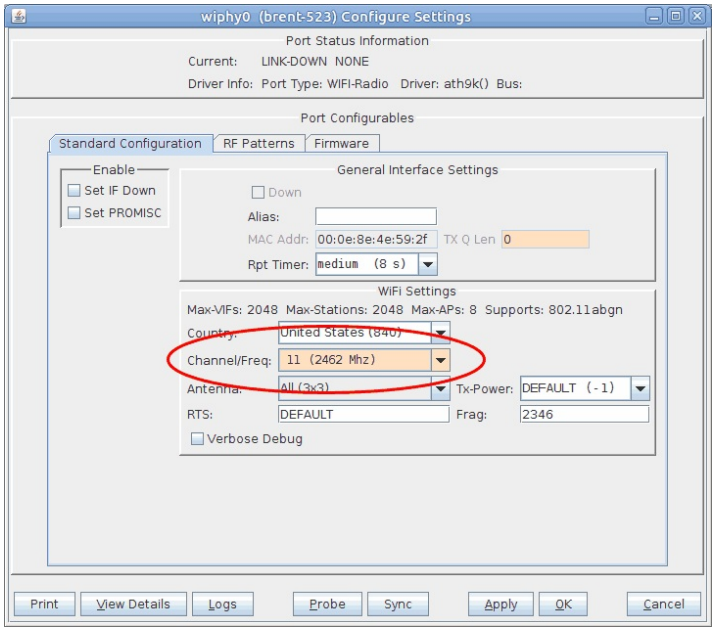
This example uses a LANforge CT523 system but the procedure should also work on a CT521, CT522, CT525 or similar system.



1. Create a VAP.

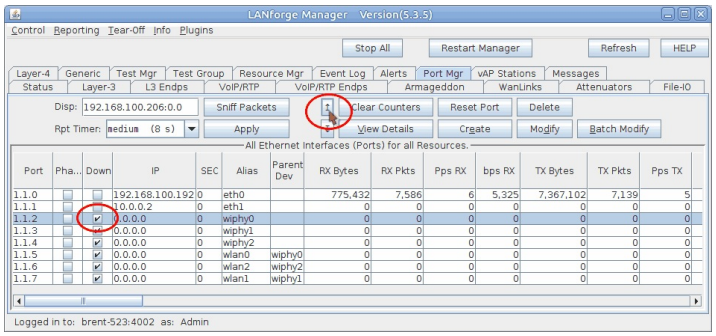
A. Verify the wiphy device used for the VAP is on your preferred channel (this test will use channel 11).

A. In the **Port Mgr** tab of the LANforge Manager, modify the wiphy device that'll be used for the VAP (wiphy0 in this test).



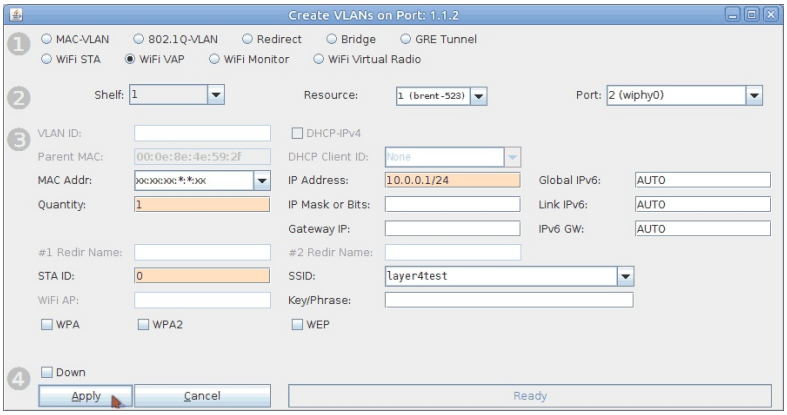
I. Select your preferred channel here.

B. Make sure the wiphy device is up.



I. If the device is down like in the above screenshot, select it and click the **Admin UP** button (also indicated in the above screenshot).

B. In the **Port Mgr** tab, select a wiphy device (wiphy0 in this test) and click **Create**.



A. Select the **WiFi VAP** radio button.

B. Set **Quantity** to **1**.

C. Set **STA ID** to **0**.

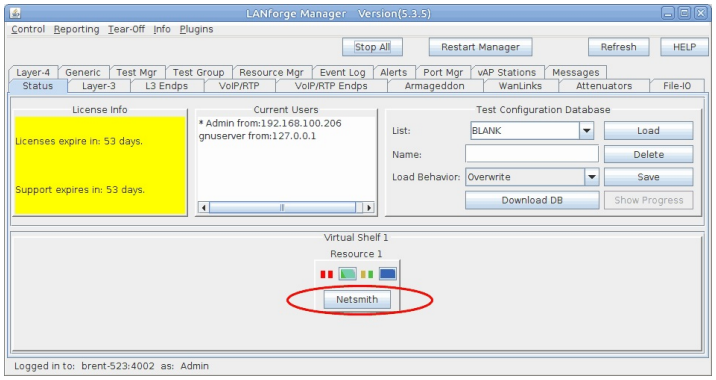
D. Set **IP Address** to **10.0.0.1/24**.

E. Set the **SSID** to **layer4test**.

F. Click **Apply** and close the create port window.

C. Configure the VAP.

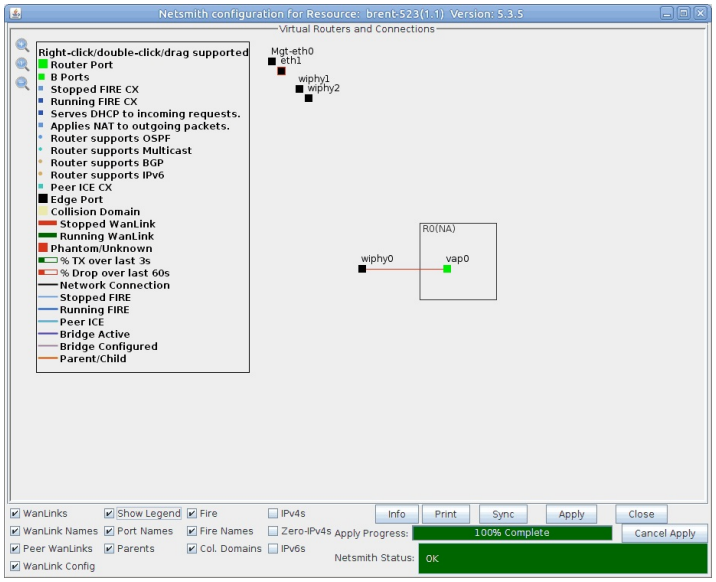
A. Open **Netsmith** from the **Status** tab.



B. In Netsmith, right click and select **New Router**.

C. Click **OK**.

D. Drag vap0 into the virtual router.



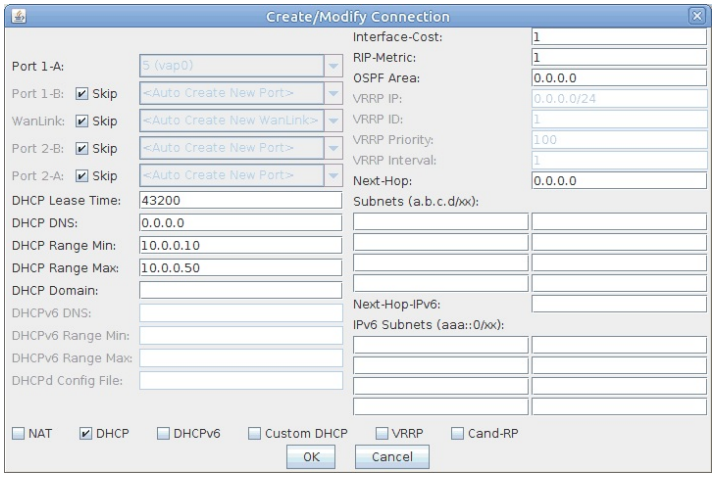
E. Right click vap0 and select **Modify**.

F. Check **DHCP**.

G. Change **DHCP Range Min** to **10.0.0.10**

H. Change **DHCP Range Max** to **10.0.0.50**

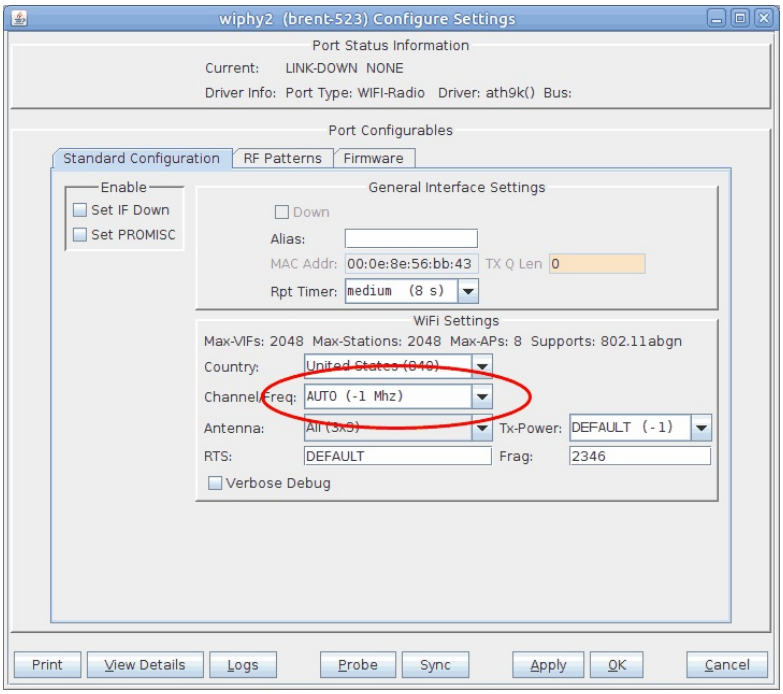
I. Click **OK**.



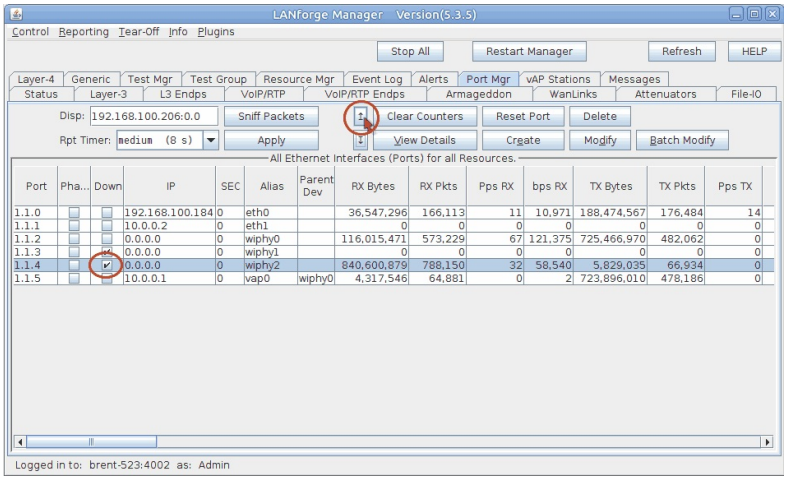
J. Click **Apply** in Netsmith then close the window.

2. Create 10 virtual stations.

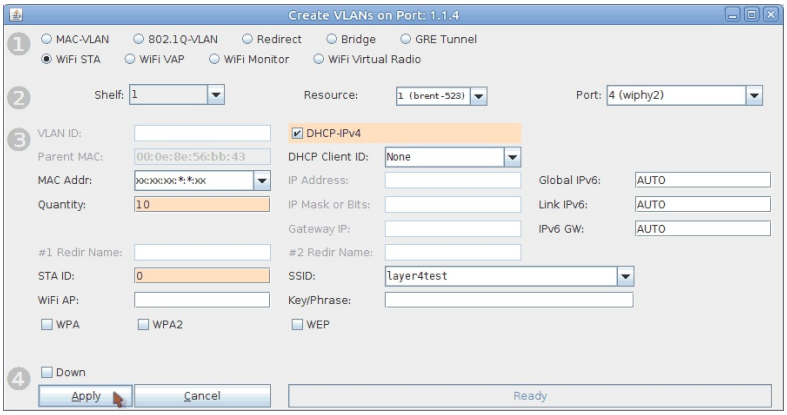
A. Verify the wiphy device used for the stations (wiphy 2 in this test) is on the **AUTO** channel.



B. Make sure wiphy2 is up.



C. In the **Port Mgr** tab, select a wiphy device (wiphy2 in this test) and click **Create**.



A. Select the **WiFi STA** radio button.

B. Set **Quantity** to **10**.

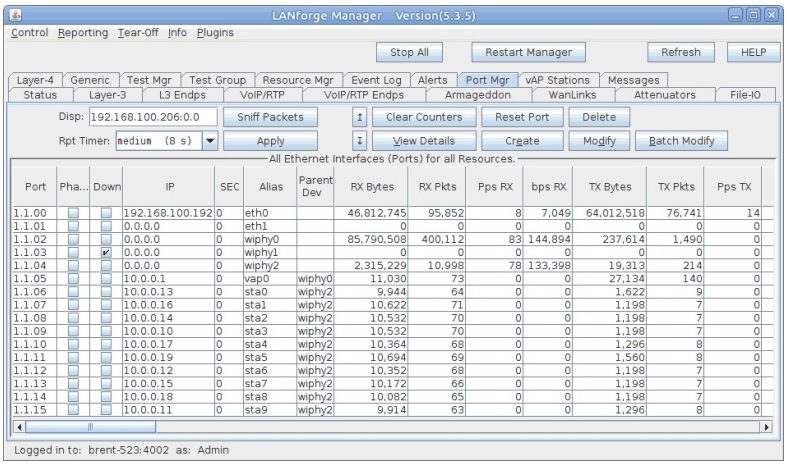
C. Set **STA ID** to **0**.

D. Select **DHCP-IPv4**.

E. Set the **SSID** to **layer4test**.

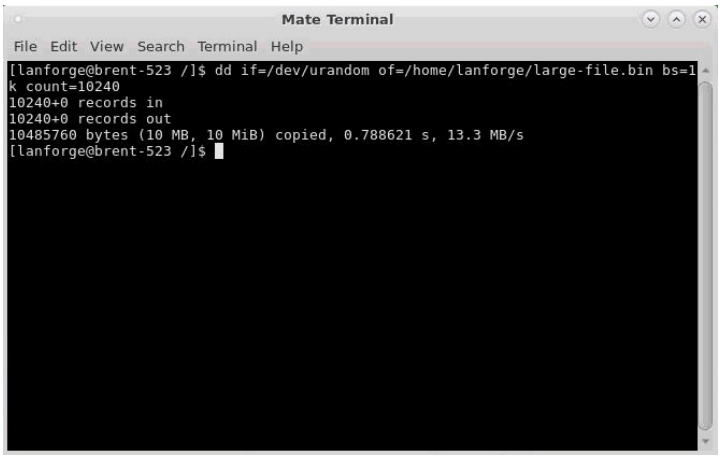
F. Click **Apply** and close the create port window.

D. Make sure the 10 staX ports get IPs.



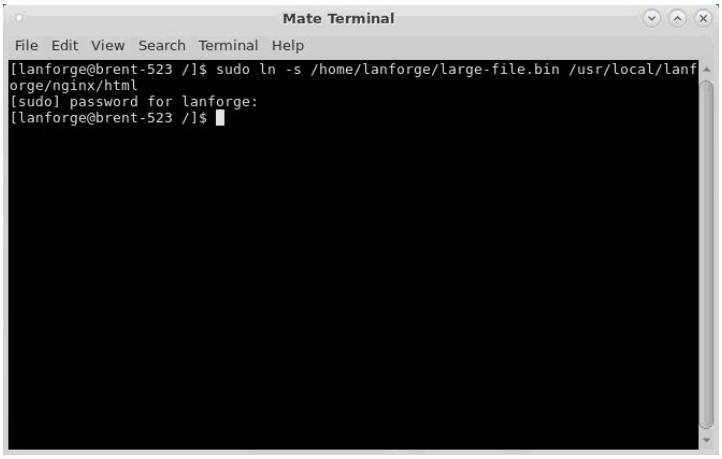
3. Create a file for the layer 4-7 endpoint to use.
   A. In a terminal on the LANforge system, run the below command to generate a 10MB file in **/home/lanforge**.
   Note: The smaller a file is, the harder it is to reach higher rates. Therefore it is recommended to use a larger file for these tests.

```
dd if=/dev/urandom of=/home/lanforge/large-file.bin bs=1k count=10240
```



   B. For the webserver to serve the file we created, it needs to know where to find it. Run the below command in a terminal on the LANforge system to link the file.

```
ln -s /home/lanforge/large-file.bin /usr/local/lanforge/nginx/html
```

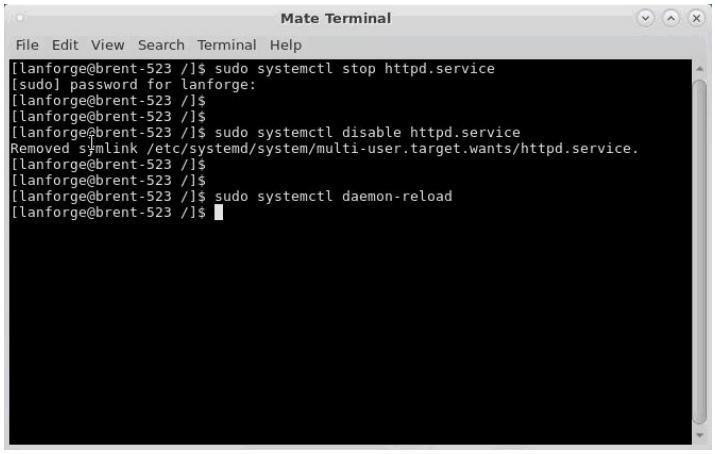


4. Set up the HTTP server on **vap0**.

A. Before starting HTTP on vap0, the Apache service may need to be disabled.

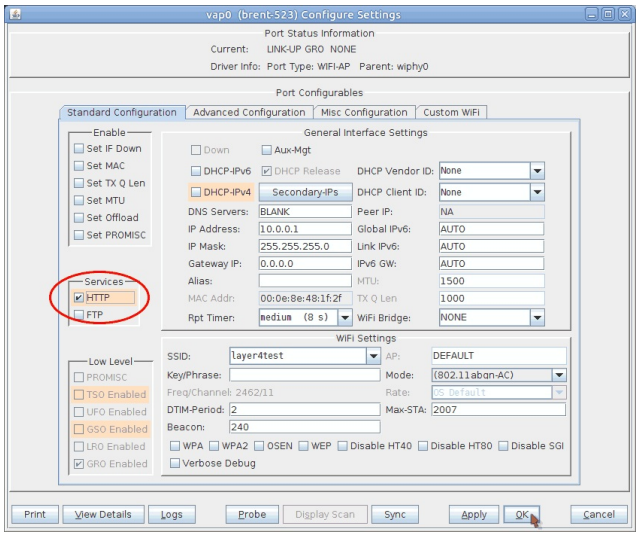
A. Stop and disable httpd (Apache) in the LANforge terminal with the below commands.

```
sudo systemctl stop httpd.service
sudo systemctl disable httpd.service
sudo systemctl daemon-reload
```



B. Modify **vap0** in the LANforge **Port Mgr** tab.

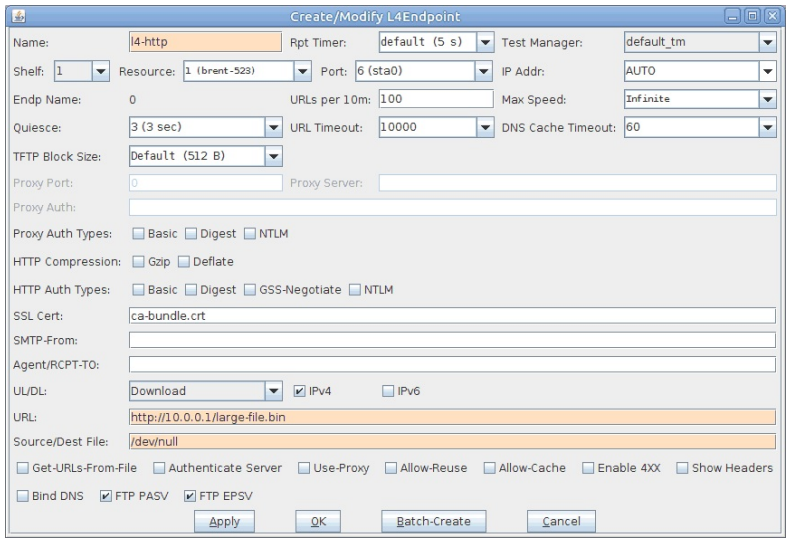
I. Enable the **HTTP** checkbox.



II. Click **OK**.

5. Create a layer 4-7 endpoint. WiFi Capacity will be using this as a template to copy from.

A. In the Layer 4-7 tab, click **Create**.



A. Set the **Name** to **l4-http**

B. Set the **Port** to **sta0**.

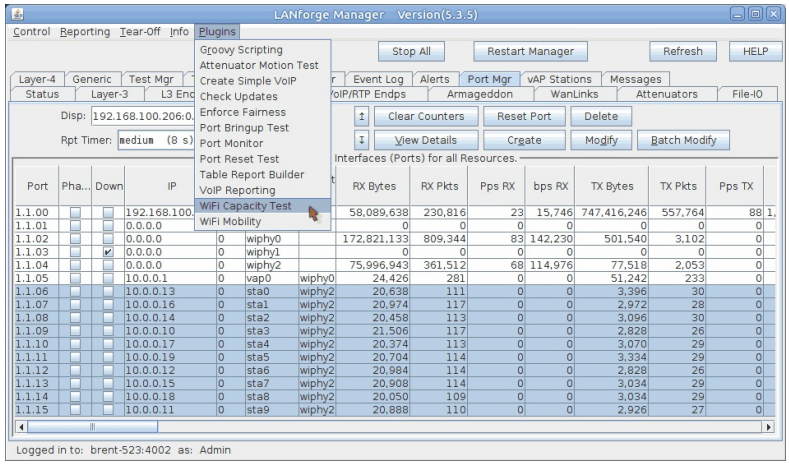
C. The **URL** will point to the VAP's IP: **http://10.0.0.1/large-file.bin**
Note: This is where you can specify an IP of an AP you wish to test. LANforge also supports other layer 4-7 protocols, for more information you can view a tooltip by hovering over the URL text box.

D. Set the **Source/Dest File** to **/dev/null**

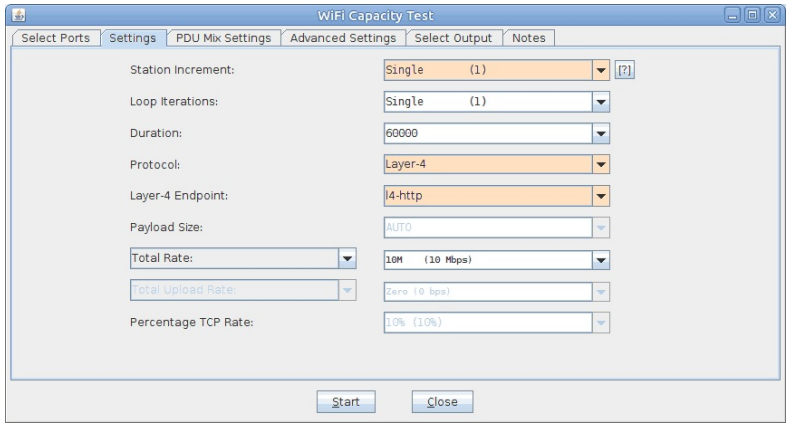
E. Click **OK**.

6. Set up and run a WiFi Capacity test.

A. Select the 10 created stations, then open WiFi Capacity Test from the Plugins menu.



B. Go to the Settings tab.



A. Set **Station Increment** to **1**.

B. Set the Protocol to **Layer 4-7**. **Note:** This should automatically be set if you first select a Layer 4-7 Endpoint.

C. Select your Layer 4-7 Endpoint (**l4-http** in this test). The capacity test will use this as a template for each of the ten stations.

D. **Total Rate** can stay at 10Mbps. **Note:** This rate can represent either upload or download traffic depending on how you have your layer 4-7 endpoint configured.

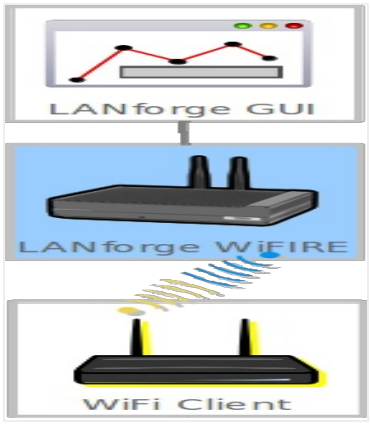
C. Run the Capacity test by clicking **Start**.

A. The test will now make a copy of the selected layer 4-7 endpoint for each station. **Note:** You may notice that URLs per 10m is set to a high rate, this is to ensure the maximum amount of URLs are processed as WiFi Capacity adjusts the Max Speed.

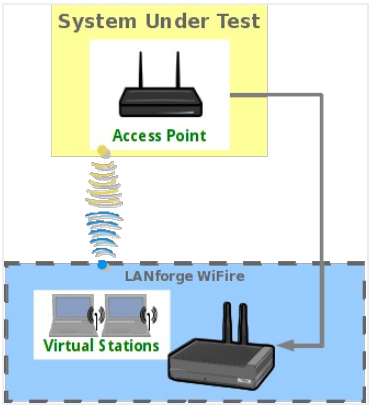
Using Wireshark to Sniff WiFi Monitors

Goal: Sniff wireless traffic from a LANforge radio using Wireshark and a WiFi Monitor port.

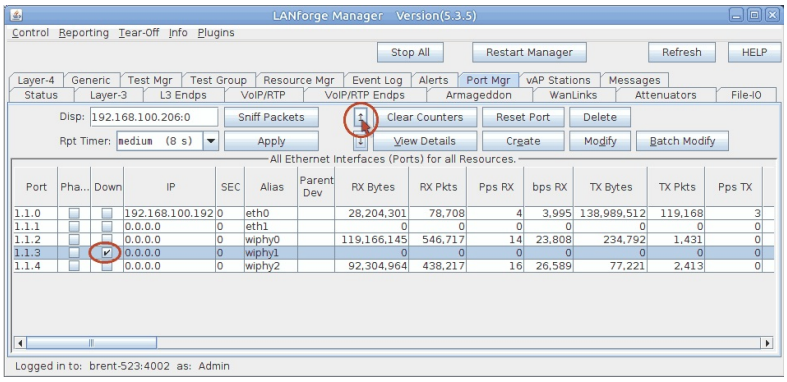
The best way to sniff wireless packets via Wireshark in LANforge is from a monitor port that is on its own radio (no other AP, STAs, etc.). This example will walk through the monitor port creation, sniffing the monitor port, as well as Wireshark filter recommendations.



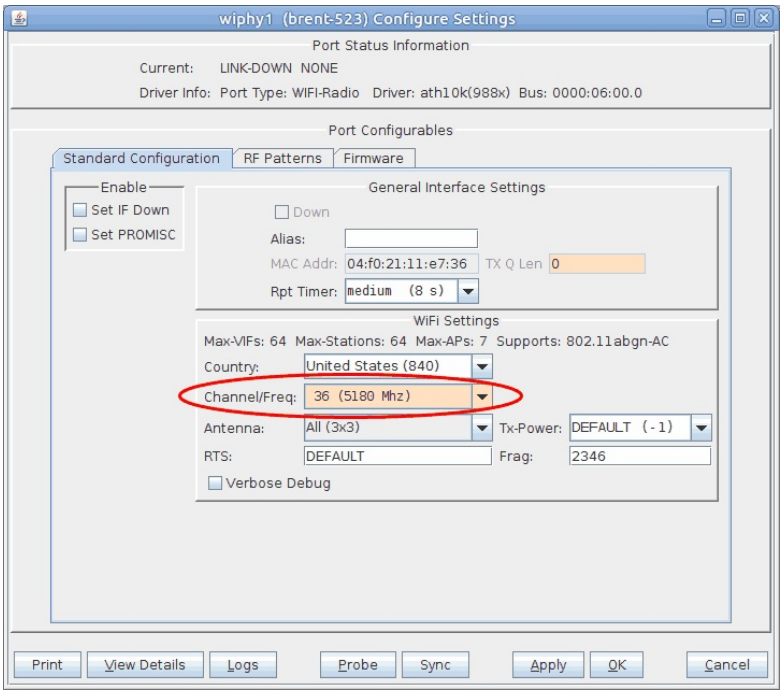
This example uses a LANforge CT523 system but the procedure should work on a CT522, CT525, or similar system.



1. Create a monitor port.
   A. In the **Port Mgr** tab, select a wiphy device that you wish to sniff with (this example will use wiphy1, an ath10k radio).
   B. If the wiphy device is down, click the up arrow to enable it.



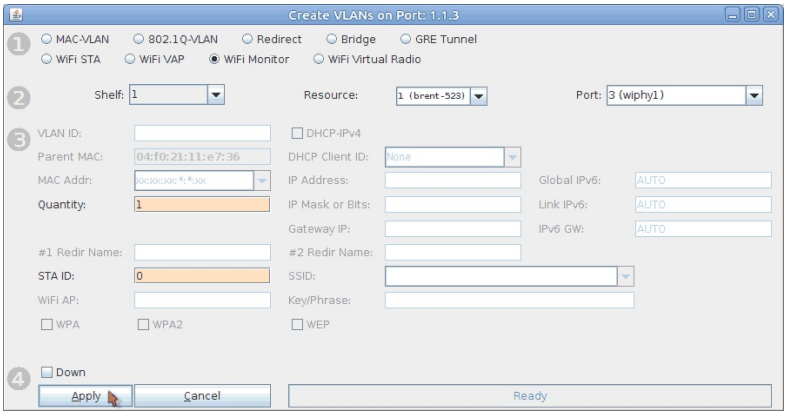
C. Click **Modify**.



A. Select the channel you wish to sniff. Channel 36 will be used for this test.

B. Click **OK**.

D. Back in the **Port Mgr** tab, with the wiphy device still selected, click **Create**.



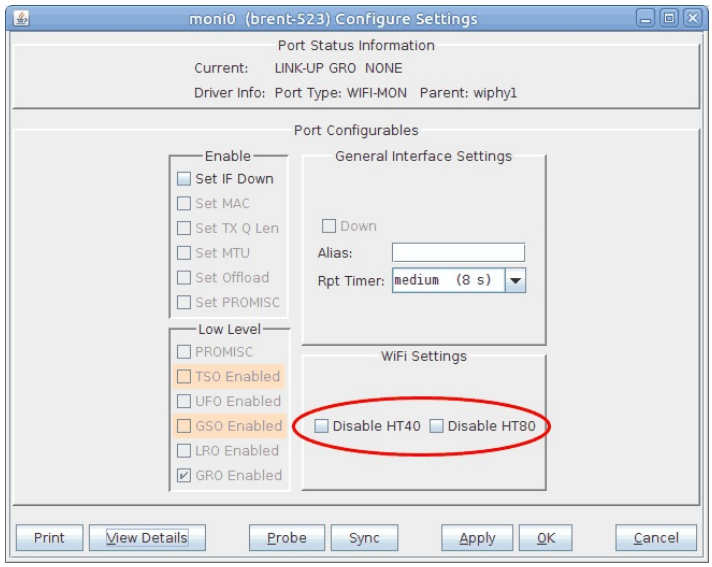
A. Select the **WiFi Monitor** option at the top.

B. Set the **Quantity** to **1**.

C. Set the **STA ID** to **0**.

D. Click **Apply** and close the Create Port window.

E. In the **Port Mgr** tab again, modify **moni0**.



A. You can disable **HT40** and **HT80** here if needed.

B. Click **OK** to close the window.

2. For this current setup, traffic will be generated with a layer 3 UDP connection between two stations.

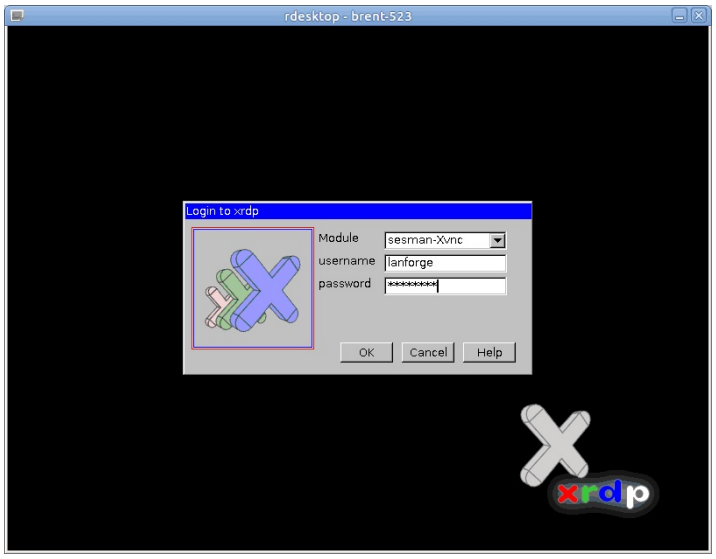
   For more information see Generating Traffic for WLAN Testing

3. Use Wireshark to sniff **moni0**.

   A. If you are running the LANforge GUI from a Windows machine without x server installed, you will need to connect remotely to the LANforge system via **rdesktop** or **vnc**.

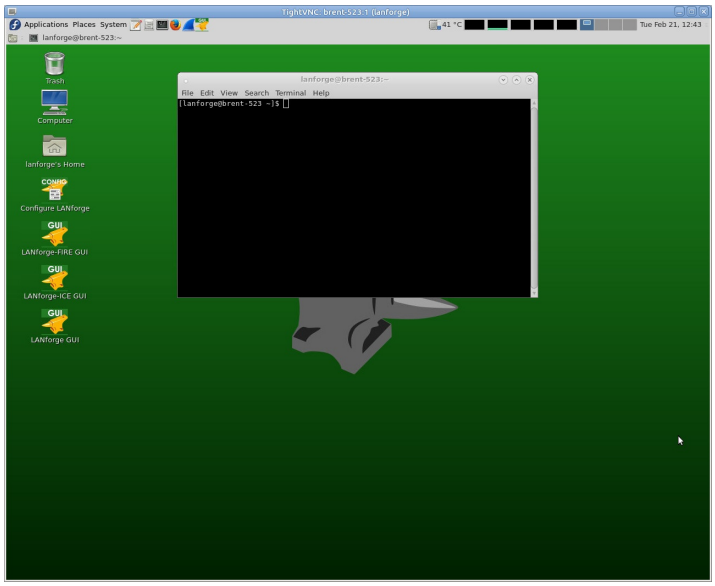
      A. To connect via **rdesktop**, type the following command into a console (replace LANforge-IP with the IP of your LANforge system):

      ```
      rdesktop LANforge-IP
      ```

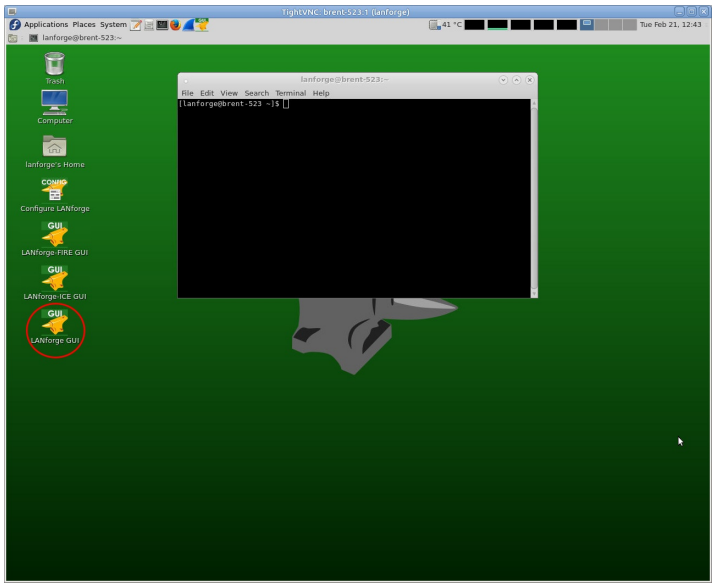


I. The login info is username/password **lanforge/lanforge**

B. To connect via **vnc**, type the following command into a console (replace LANforge-IP with the IP of your LANforge system. **Don't forget to add the ':1' after the IP**):
`vncviewer [LANforge-IP]:1`
The password is **lanforge**.

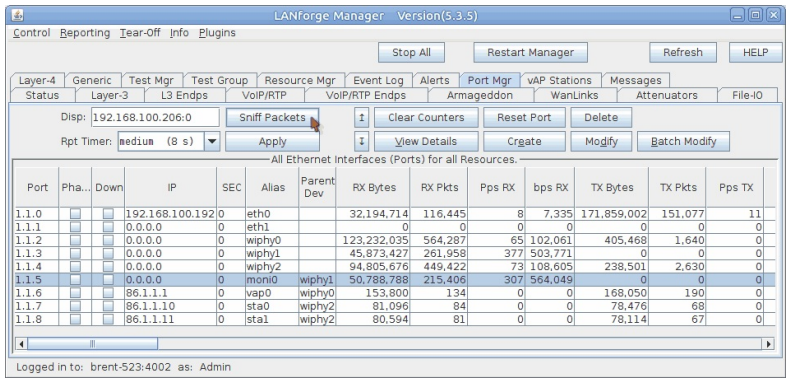


C. Once you have accessed the LANforge system via rdesktop or vnc, open the LANforge GUI with the desktop icon shown below.

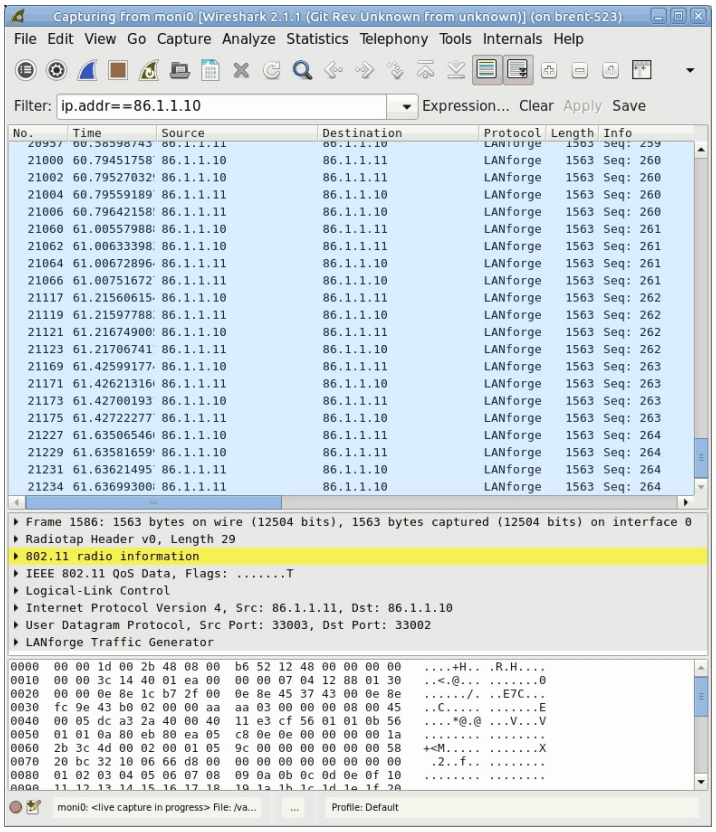


B. Select **moni0** in the **Port Mgr** tab.

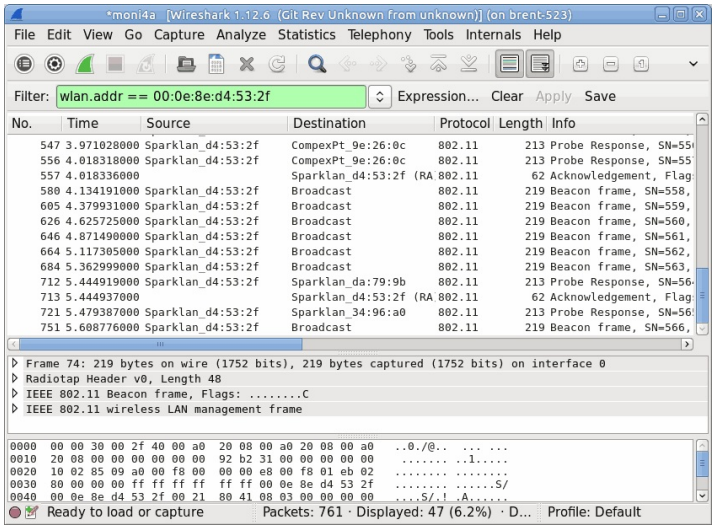
C. Click the **Sniff Packets** button. Wireshark will now open and automatically start scanning for packets. If you get a window that warns about running as user root, click **OK**.



A. To use a filter, simply add the filter constraints to the filter text box as seen below and click **Apply** to the right. The below screenshot has wireshark filtering on a specific IP.



B. If you'd like to only see traffic to/from a single AP use the filter **wlan.addr == [bssid]**



D. There are many filters that can be used in Wireshark. Some handy ones include:
IP: `ip.addr==x.x.x.x`
wlan MAC: `wlan.addr==xx:xx:xx:xx:xx:xx`
Association request `wlan.fc.type_subtype eq 0`
Association response `wlan.fc.type_subtype eq 1`
Probe request `wlan.fc.type_subtype eq 4`
Probe response `wlan.fc.type_subtype eq 5`
Beacon `wlan.fc.type_subtype eq 8`
Authentication `wlan.fc.type_subtype eq 11`
Deauthentication `wlan.fc.type_subtype eq 12`

E. Filters can be combined to specify if packets should match all filters (with **&&**) or any filters (with **||**). For example, if you wanted to view packets that **only contain both** IPs 1.1.1.1 and 2.2.2.2 you could use the following: `ip.addr==1.1.1.1 && ip.addr==2.2.2.2`
Or, if you want to see all packets containing 1.1.1.1 and all packets containing 2.2.2.2, you could use the following: `ip.addr=1.1.1.1 || ip.addr==2.2.2.2`

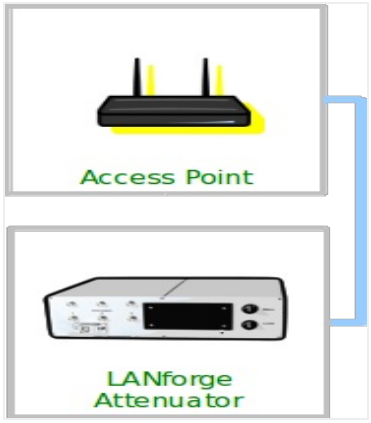
F. You can visit https://wiki.wireshark.org/DisplayFilters for more tips on filters.
A handy 'cheat sheet' with most filters can be found here.

Scripting a Standalone CT703 or CT704 Attenuator

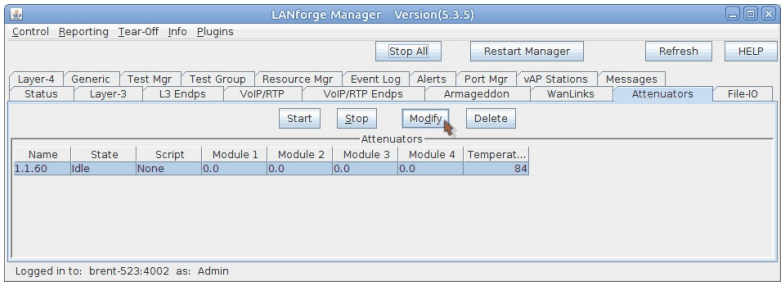
Goal: Operate a CT703 or CT704 WiFi Attenuator with a script.

The LANforge GUI provides scripting support for the CT703 and CT704 Attenuators. This cookbook provides a basic example of how to script the signal strength up and down.

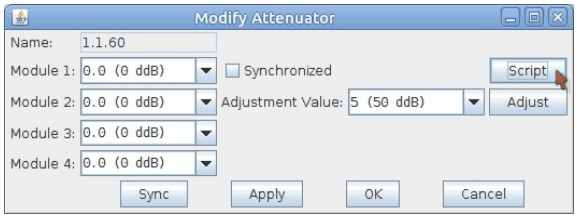
Requires LANforge 5.2.11+ and CT703 or CT704 Attenuator.



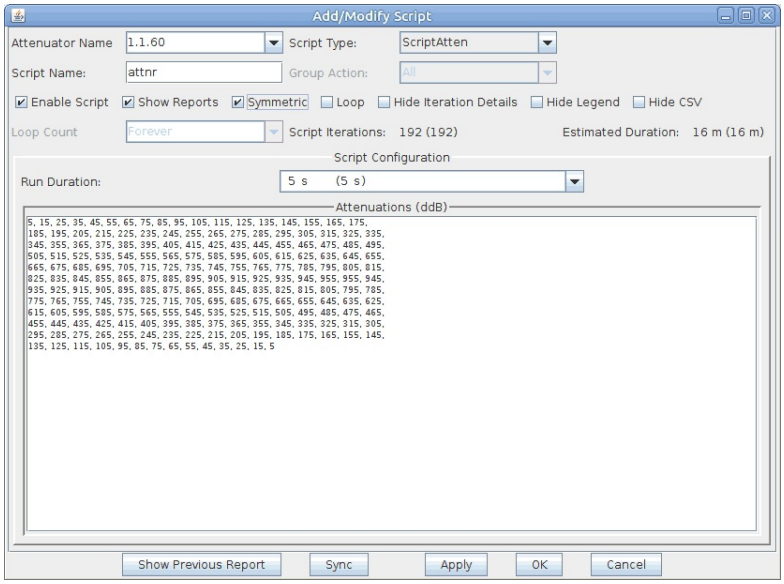
1. Create an Attenuator Script
   A. In the **Attenuators** tab, select your attenuator and click **Modify**



   B. You will see the Modify Attenuator window. Click on the **Script** button.



   C. The Add/Modify Script window opens. In the picture below you see a huge list of numbers (the attenuation levels). These numbers are not pre-populated. You can copy and paste them out of this document or create a series with other commands listed below.



A. In the Script Type pulldown, select **ScriptAtten**

B. Type in the name for the script, this example is called `attnr`

C. Select Symmetric

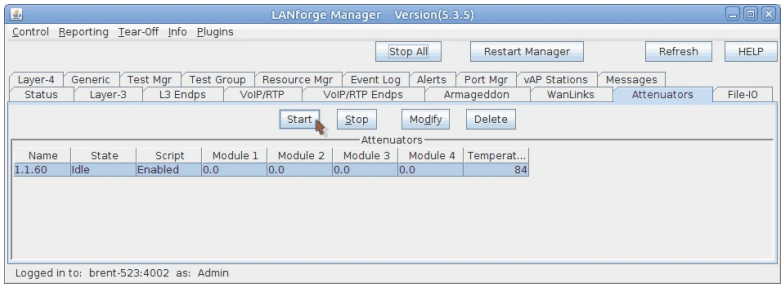
D. Choose **5 s** for Run Duration. This will run the Attenuator at each ddB value for this period of time.

E. Enter the following attenuation values for a 16 minute long test. This will send the attenuator from 5 to 955ddB and back:

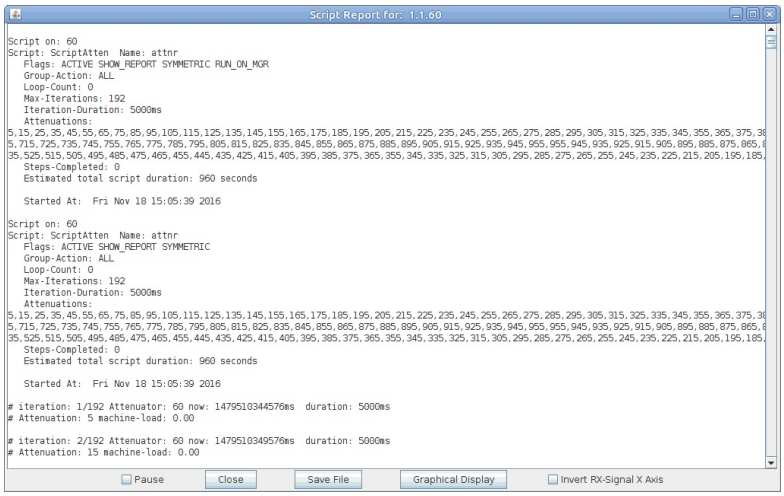
F.
```
5, 15, 25, 35, 45, 55, 65, 75, 85, 95, 105, 115, 125, 135, 145, 155, 165, 175,
185, 195, 205, 215, 225, 235, 245, 255, 265, 275, 285, 295, 305, 315, 325, 335,
345, 355, 365, 375, 385, 395, 405, 415, 425, 435, 445, 455, 465, 475, 485, 495,
505, 515, 525, 535, 545, 555, 565, 575, 585, 595, 605, 615, 625, 635, 645, 655,
665, 675, 685, 695, 705, 715, 725, 735, 745, 755, 765, 775, 785, 795, 805, 815,
825, 835, 845, 855, 865, 875, 885, 895, 905, 915, 925, 935, 945, 955, 955, 945,
935, 925, 915, 905, 895, 885, 875, 865, 855, 845, 835, 825, 815, 805, 795, 785,
775, 765, 755, 745, 735, 725, 715, 705, 695, 685, 675, 665, 655, 645, 635, 625,
615, 605, 595, 585, 575, 565, 555, 545, 535, 525, 515, 505, 495, 485, 475, 465,
455, 445, 435, 425, 415, 405, 395, 385, 375, 365, 355, 345, 335, 325, 315, 305,
295, 285, 275, 265, 255, 245, 235, 225, 215, 205, 195, 185, 175, 165, 155, 145,
135, 125, 115, 105, 95, 85, 75, 65, 55, 45, 35, 25, 15, 5
```

G. ...click **OK**

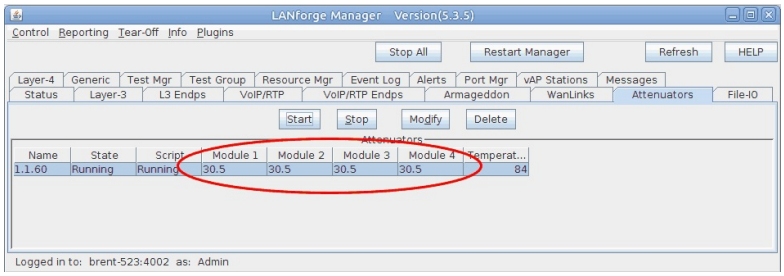
D. In the Attenuators tab, you will notice that your attenuator now reads **Enabled** in the Script column. Select the attenuator and click the **Start** button.



E. The window Script Report will appear. This monitors the script behavior of the attenuator script.



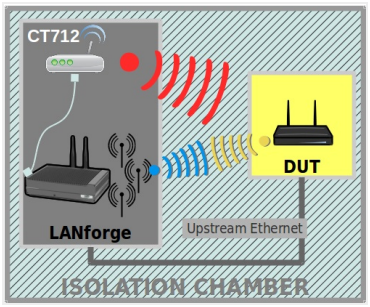
F. Also notice the attenuation values change in the LANforge Manager as the script runs.
Note: These values update every five seconds, so values may differ within the five second window.



Using CT712 RF Noise Generator and RADAR Simulator to test DFS channels

Goal: Generate RF pulses that simulate RADAR and cause an AP using DFS to exercise its detection and response abilities.

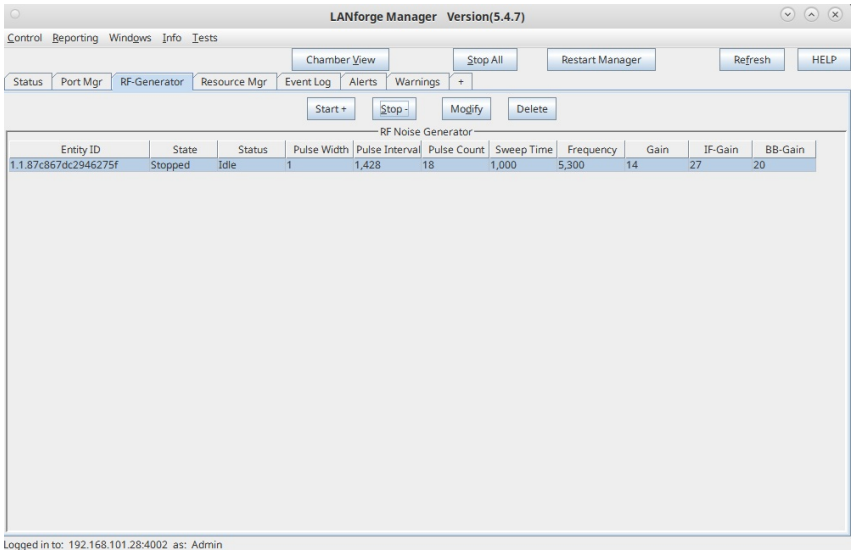
DFS (Dynamic Frequency Selection) allows an AP to use channels in the same spectrum that some RADAR systems use and provide a method of changing channels when RADAR signals are detected. In this example, we will use a LANforge CT712 which is our customized script and integrated GUI control for the HackRF-One Software-Defined Radio (SDR) to generate the RF pulses that simulate RADAR signals and cause a third-party AP under test to detect and change channels. It is recommended that this type of testing be done in a RF isolation chamber to limit interference to production WLAN and RADAR systems in use.



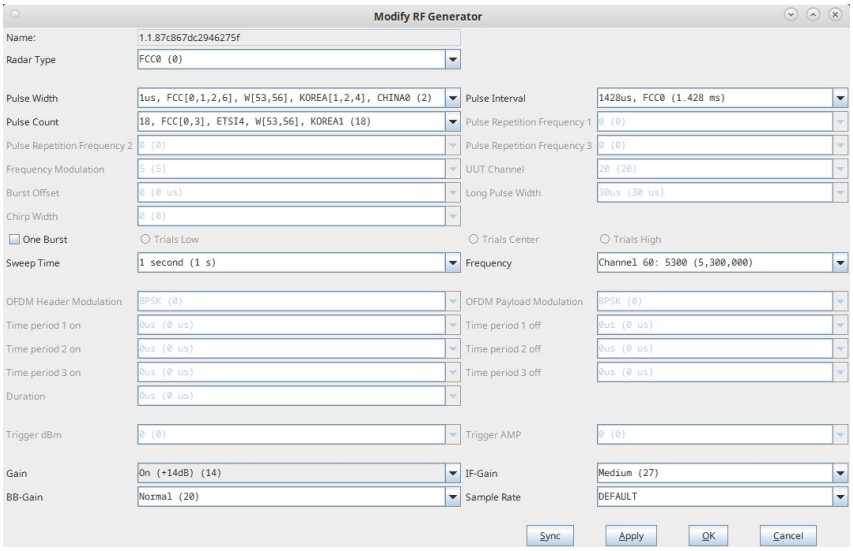
1. Verify a client is connected to a third-party AP using a DFS channel.

   For more information see
   Wikipedia: 5GHz Channels
   WLAN Pros WiFi Channel Reference

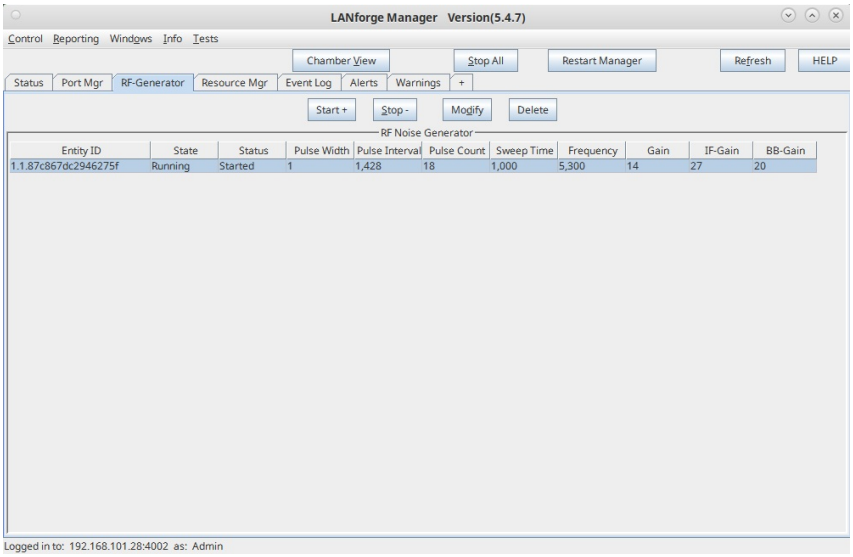
2. On the RF Generator tab, modify the device that corresponds to the CT712 USB device to set the desired channel and pulse characteristics.
   NOTE: The different RADAR types are defined as described in the document FCC DFS Compliance Procedures. RF pulses defined outside of these parameters can be used for other RF noise and interference tests.



3. Select a FCC RADAR type preset button or enter specific values to set the desired pulse characteristics, then set the transmit frequency, then select OK.



4. Highlight the RF Generator entry and select the Start+ button to start transmitting.
   In some versions, you may have to select Start+ twice in order to start transmitting



5. When the CT712 is activated, the Status will show Started and the HackRF-One device's TX LED should show a red light. The AP under test should detect RADAR within 10 seconds, change to an available channel and not be able to use the previous channel again for 30 minutes.

   For more information see
   WiFi Alliance: DFS Best Practices

6. An alternate method of controlling the CT712 HackRF-One device is to use the lf_hackrf_dfs.py script in the /home/lanforge directory:

   A. Open a terminal window on the LANforge system.

   B. cd to /home/lanforge

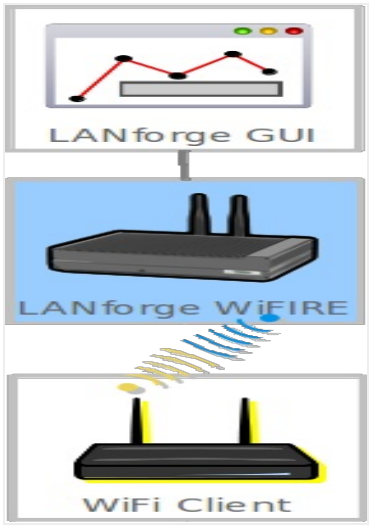
   C. To show the script usage, type: ./lf_hackrf_dfs.py --help

```
usage: lf_hackrf_dfs.py [-h] [--lf_hackrf LF_HACKRF] [--pulse_width PULSE_WIDTH]
                        [--pulse_interval PULSE_INTERVAL] [--pulse_count PULSE_COUNT]
                        [--one_burst] [--bursts BURSTS] [--rf_type RF_TYPE]
                        [--uut_channel UUT_CHANNEL] [--sweep_time SWEEP_TIME] [--freq FREQ]
                        [--daemon DAEMON] [--pid_file PID_FILE] [--gain GAIN] [--if_gain IF_GAIN]
                        [--bb_gain BB_GAIN] [--log_level LOG_LEVEL] [--mgt_pipe MGT_PIPE]
                        [--no_repeat] [--sample_mod SAMPLE_MOD]
                        [--host_perf_adjust HOST_PERF_ADJUST] [--tx_sample_rate TX_SAMPLE_RATE]
           dfs testing , FCC0-6, ETSI0-6
           Note: some systems will need to preface command with sudo nice -19


options:
  -h, --help            show this help message and exit
  --lf_hackrf LF_HACKRF
                        --lf_hackrf { last 4 bytes of serial num from hackrf_info }
  --pulse_width PULSE_WIDTH
                        --pulse_width { usecs } default: 1
  --pulse_interval PULSE_INTERVAL
                        --pulse_interval { usecs } default: 1
  --pulse_count PULSE_COUNT
                        --pulse_count { number } default: 1
  --one_burst           --one_burst store_true default: False
  --bursts BURSTS       --bursts provide a number of bursts
  --rf_type RF_TYPE, --radar_type RF_TYPE
                         Note: .lower used so upper case or lower case may be entered.
                                                to see addition information telnet <lanforge ip> 4003 sh
```

WiFi: Mesh Testing: video demonstration

Goal: Learn about how to use the Chamber View feature to create WiFi mesh test scenarios.

Watch a demonstration of how a Chamber View mesh test works. This scenario requires LANforge version 5.3.9, four isolation chambers, four or more programmable attenuators a DUT and three mesh AP nodes.

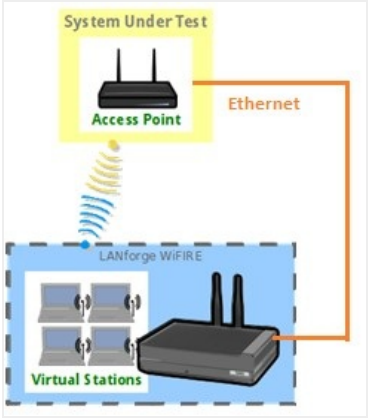


WiFi Mesh demonstration

Diagnostic script for WiFi packet capture files.

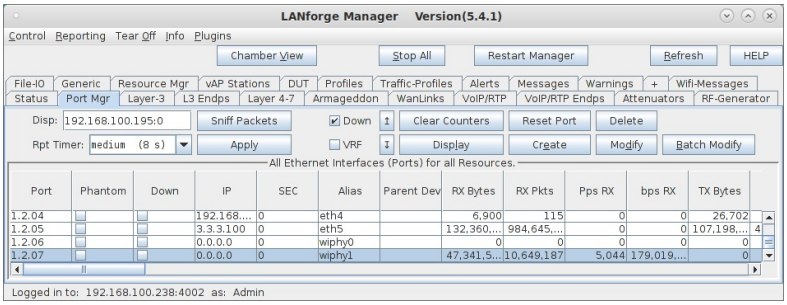
Goal: Use a diagnostic script to parse a WiFi packet capture file and generate histograms, packet loss stats, and other helpful information for diagnosing WiFi behaviour.

LANforge will be used to create a WiFi capture file, and then we will run the diagnostic script to help understand the on-air behaviour.

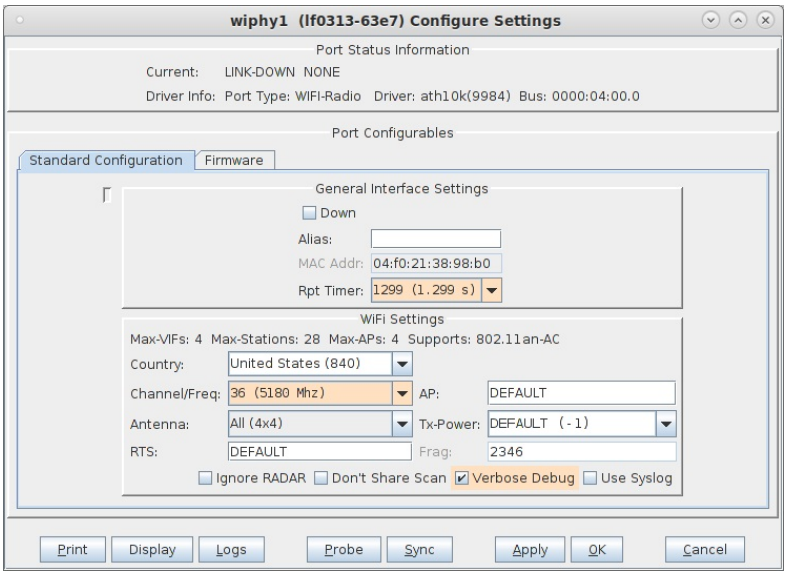


1. Create capture file.
    A. If you already have your own capture file or know how to create them, then skip this section.

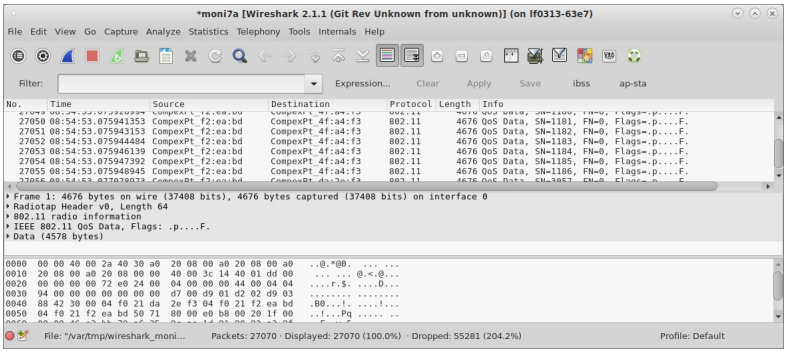
B. Packet captures are created using WiFi Monitor interfaces. LANforge can automatically create and manage these for you. The simplest way to create a capture is to use the Port-Mgr tab in the LANforge GUI.



C. Select WiFi radio that you wish to use and double-click it to bring up the modify window. Set the frequency to match the channel you wish to sniff and click OK to submit the changes and close the window.



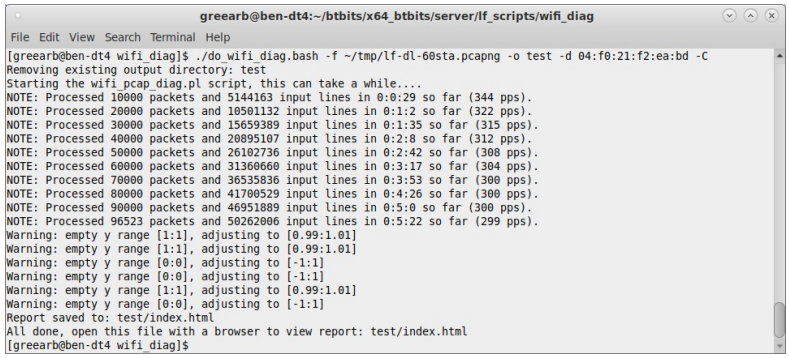
D. Make sure the radio is still selected, and click the 'Sniff Packets' button on the Port Mgr tab. You normally need to be connectedto the LANforge system using remote-desktop or VNC for this to work properly. After you click 'Sniff Packets', a monitor interface will be created and Wireshark will start. When your capture is complete, stop the capture in Wireshark and save the capture file.



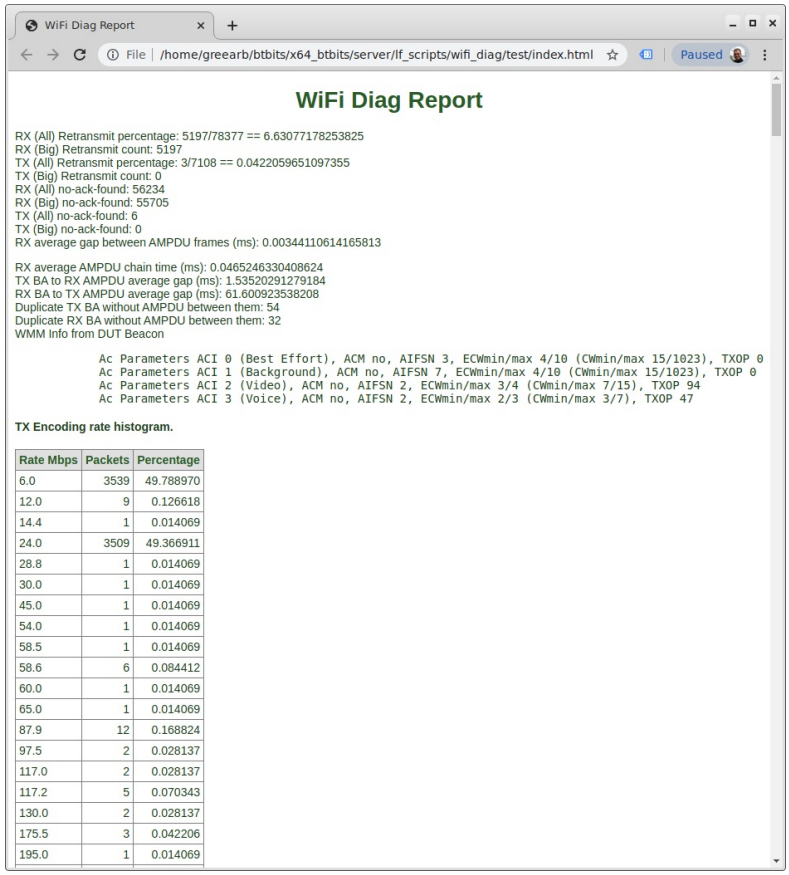
2. Use the do_wifi_diag.bash script to diagnose the capture file.

A. The LANforge scripts package is normally installed in /home/lanforge/scripts You can also clone the repository from github using this link: https://github.com/greearb/lanforge-scripts Assuming it is in the standard location, change to the wifi_diag directory: **cd /home/lanfore/scripts/wifi_diag**. The ./do_wifi_diag.bash script will be used to launch the wifi-diag script.

B. Run the do_wifi_diag.bash script with appropriate arguments to match your device-under-test (DUT) and pcap file. The diagnostic script can process around 300 packets per second on a fast machine, so it can take a while to process a big file.



C. When the test is complete, you can open the [test]/index.html file to view the results, print to PDF, etc.



D. You can find the full report from this example here: examples/wifi-diag-report/index.html You can also view the report in PDF format: examples/wifi-diag-report.pdf

Powersave Test Cases

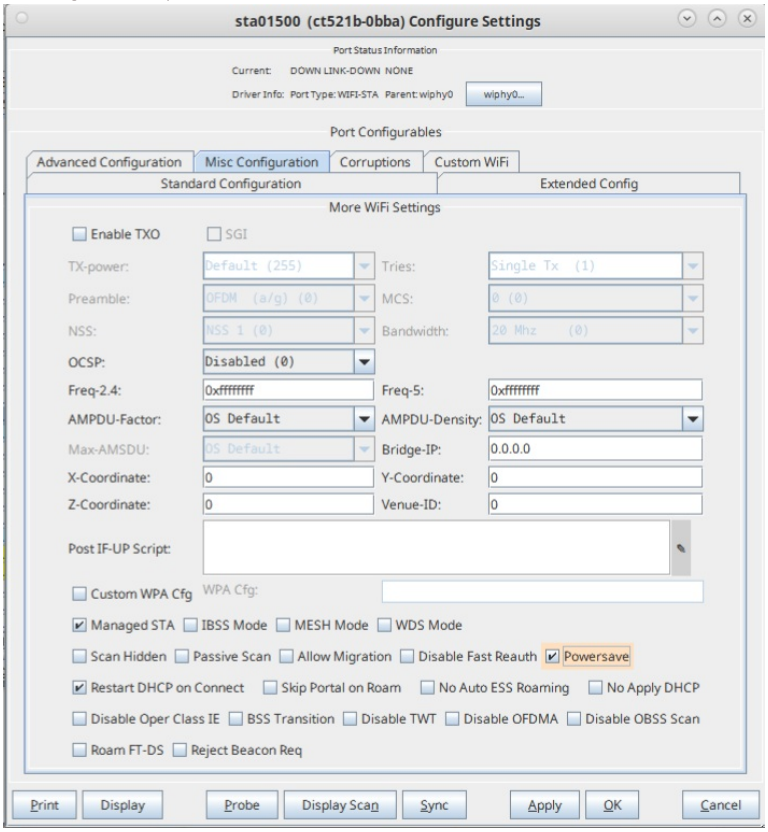
Goal: Test and verify powersave mode on a station using various methods.

DTIM multicast testing:

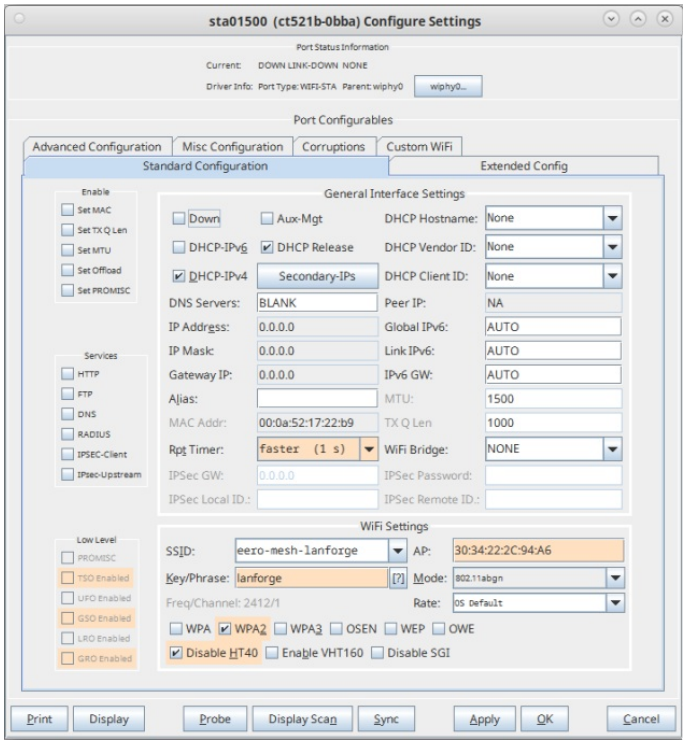
- Set up mcast traffic between upstream port sending to station with parameters of 9kbps Min Speed, 128 Kbps Max Speed and Default Pkt Size using the following cookbook link:

http://www.candelatech.com/cookbook.php?vol=wifire&book=WiFi+Station+Multicast .

- Double-click on the created station in the Port Mgr tab. Enable Powersave on LANforge station (in the Misc Configuration tab)



- Configure station for something easy to sniff (20Mhz a/b/g/n) by clicking the Disable HT40 button in the Standard Configuration tab. Then click Apply and OK to close the Configure Settings window for the station.



- Start sniffer with optional filter to see all frames from AP BSSID or STA BSSID.
- Investigate packet capture:
  - The beacon right before the mcast frame should have the TIM Multicast flag set.
  - Beacon without mcast frame soon after should NOT have the Multicast flag set.
  - DTIM count counts down to zero, only at zero can mcast frames be transmitted.
  - DTIM count should count down, with maximum value being DTIM Period - 1.
  - If AP can change DTIM period, test with multiple DTIM periods.

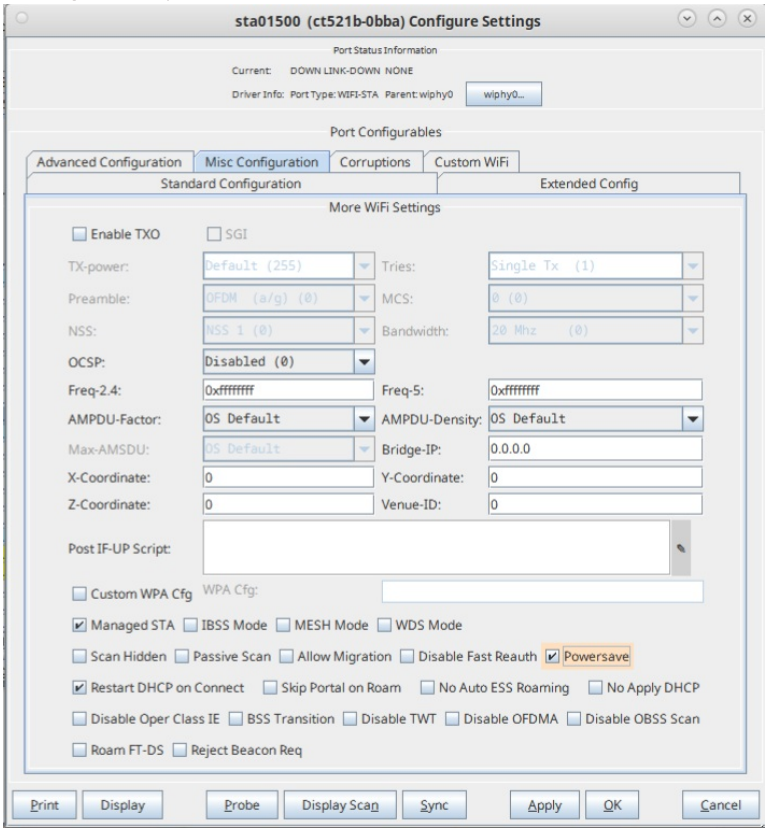
This pcap file from a station on a mtk7921k radio shows proper unicast behavior. This is using the packet filter:

wlan.addr == a8:93:4a:9d:47:a3 | | wlan.addr == 04:f0:21:9a:64:65

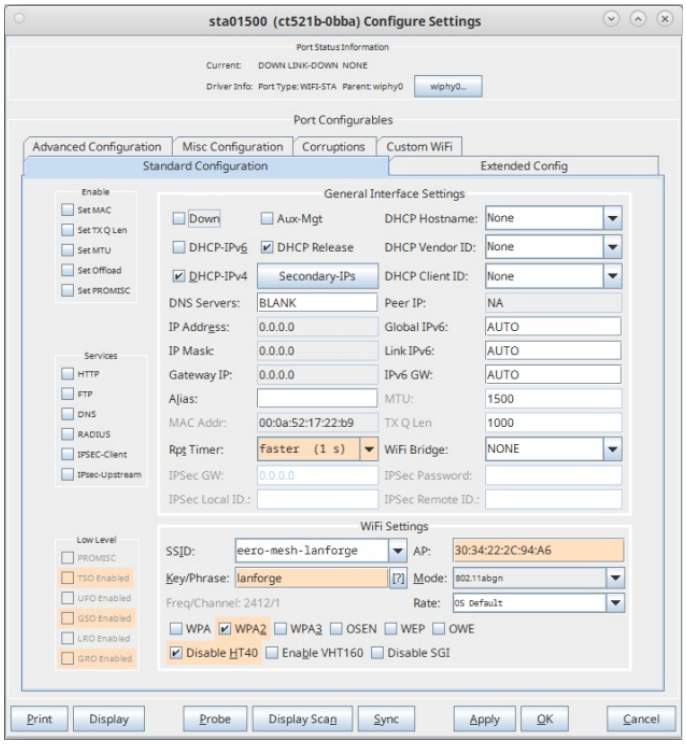
This pcap file shows bad broadcast Powersave behavior. Frame 157 TIM field should indicate Multicast and the broadcast frame 158 should be immediately after frame 157 instead of being 65ms later.

TIM unicast testing (with APs set up for multicast → unicast behaviour):

- Set up download unicast TCP traffic between upstream port and station. Set the Min Speed as 9 Kbps, Max Speed 128 Kbps, and 1400 as the Pkt Size.
- Double-click on the created station in the Port Mgr tab. Enable Powersave on LANforge station (in the Misc Configuration tab)



- Configure station for something easy to sniff (20Mhz a/b/g/n) by clicking the Disable HT40 button in the Standard Configuration tab. Then click Apply and OK to close the Configure Settings window for the station.



- Start sniffer, with optional filter to see all frames from AP BSSID or STA BSSID.

- Investigate packet capture:
  - The beacon should indicate that AID has traffic waiting (and Multicast flag would not be set).
  - STA should send wake-up null-func frame very soon after the beacon is seen (regardless of DTIM count/period, which is not pertinent for unicast frames).
  - AP acks that and proceeds to send queued traffic to STA.
  - STA goes back to sleep after a short period (around 100ms)
  - Average latency for the TCP download frames is around 100ms since frames are held on the AP until the next beacon so that STA can know to wake.

power-save-ax200-asus.pc…

power-save-ax200-sta-998…

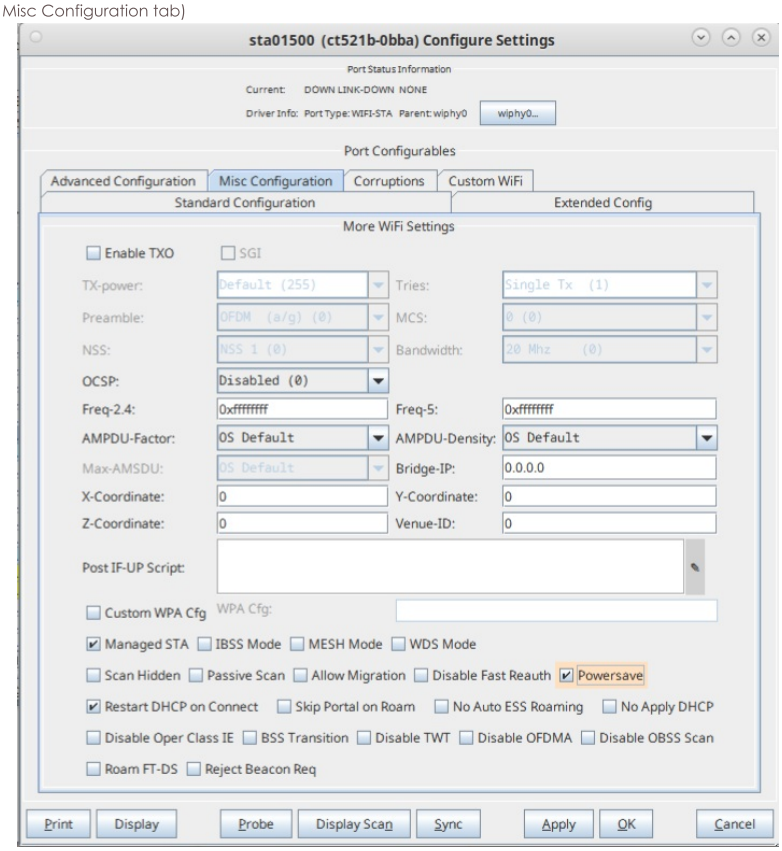
power-save-mtk7921k-sta-…

For the ASUS capture above, use this filter for **ax200: wlan.addr == 50:e0:85:8a:0a:f2 || wlan.addr == F0:2F:74:7C:A3:B0**, for ax200 to LANforge VAP use: **wlan.addr == 50:e0:85:8a:0a:f2 || wlan.addr == wlan.addr == 04:f0:21:9a:64:65** and for the mtk to LANforge VAP, use: **wlan.addr == a8:93:4a:9d:47:a3 || wlan.addr == 04:f0:21:9a:64:65**

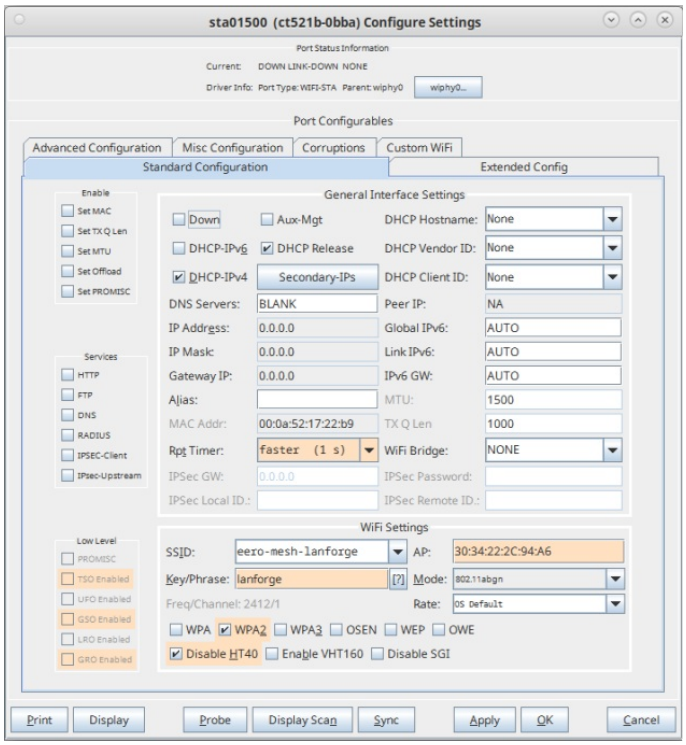
The ax200 capture shows buggy behaviour on the ASUS: ASUS does not ack frames well so ax200 sends lots of retries. The ASUS is doing multicast to unicast behaviour, so the 'multicast' frames are sent as directed unicast frames directly to the station.

Wake/Sleep testing in upload direction

- Set bursty traffic, min of zero, max of 128kbps, in upload direction.
- Double-click on the created station in the Port Mgr tab. Enable Powersave on LANforge station (in the Misc Configuration tab)



- Configure station for something easy to sniff (20Mhz a/b/g/n) by clicking the Disable HT40 button in the Standard Configuration tab. Then click Apply and OK to close the Configure Settings window for the station.

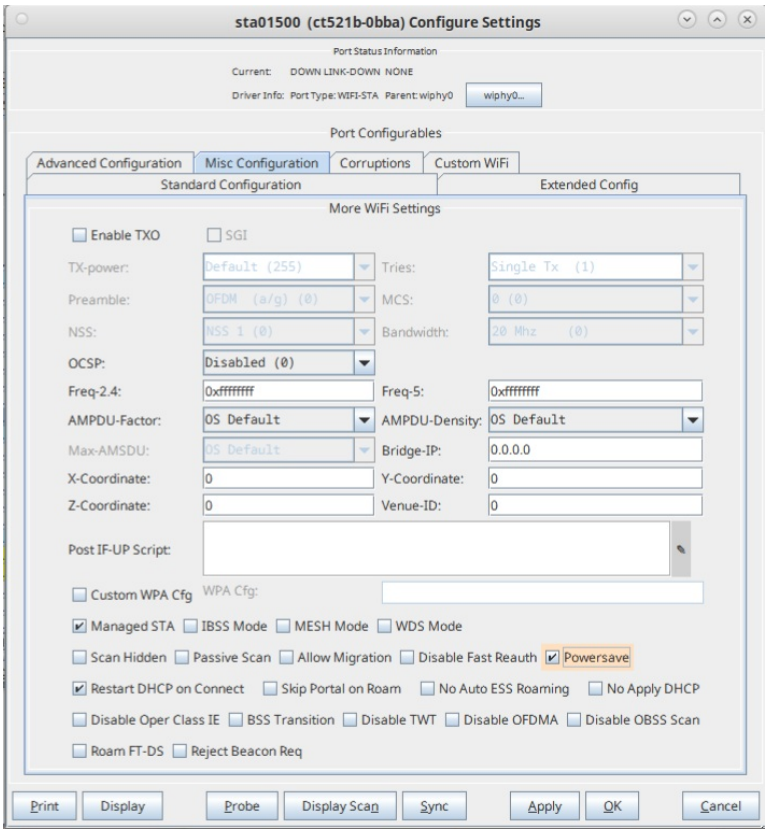


- Start sniffer, with optional filter to see all frames from AP BSSID or STA BSSID.
- Watch for packets from STA indicating it is going awake or asleep (Frame Control Field → Flags → PWR MGT)
- STA should :
  - indicate it is awake before transmitting frames.
  - go to sleep after a period of idle time (about 100ms, I think, but it may depend on the station).
  - not transmit while asleep
- STA may receive multicast packets while 'asleep' using the DTIM logic described above.
- AP should not transmit unicast frames to STA while STA is asleep.

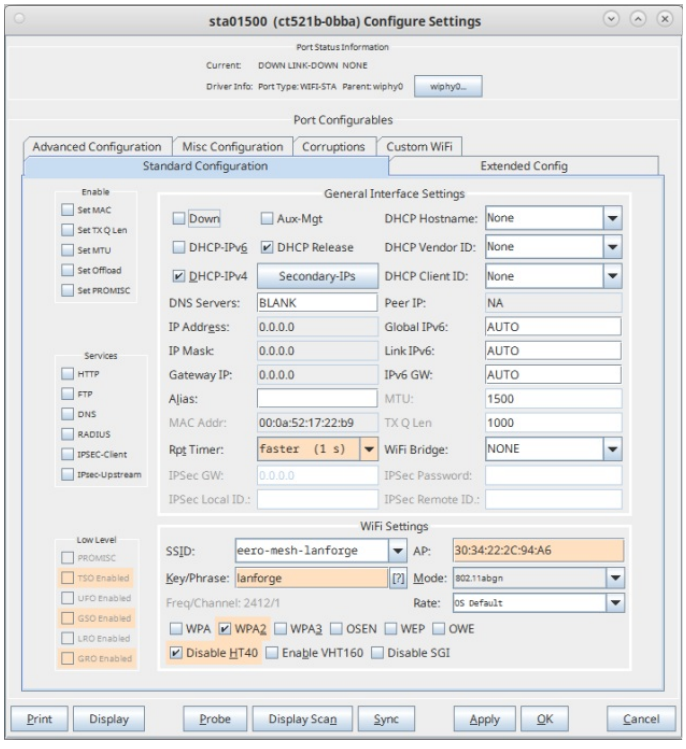
The following capture: shows power-save-mtk7921k-sta-mtk7915-lanforge-vap-unicast-upload.pcapng capture shows mtk7921k station talking to a LANforge mtk7915 VAP. The capture filter is: **wlan.addr == a8:93:4a:9d:47:a3 || wlan.sa == 00:0A:52:46:8D:4**. The capture starts with the TCP traffic idle and the station in sleep mode. The STA then wakes up in frame 884 and it starts sending TCP traffic in frame 888. The TCP traffic has then quiesced for a bit in frame 996 and STA goes back to sleep.

Wake/Sleep testing in download direction

- Configure AP for bridge mode so that NAT is not an issue. Follow the following cookbook link to configure the AP in bridge mode: http://www.candelatech.com/cookbook.php?vol=wifire&book=wifi+VAP+bridge
- Double-click on the created station in the Port Mgr tab. Enable Powersave on LANforge station (in the Misc Configuration tab)



- Configure station for something easy to sniff (20Mhz a/b/g/n) by clicking the Disable HT40 button in the Standard Configuration tab. Then click Apply and OK to close the Configure Settings window for the station.



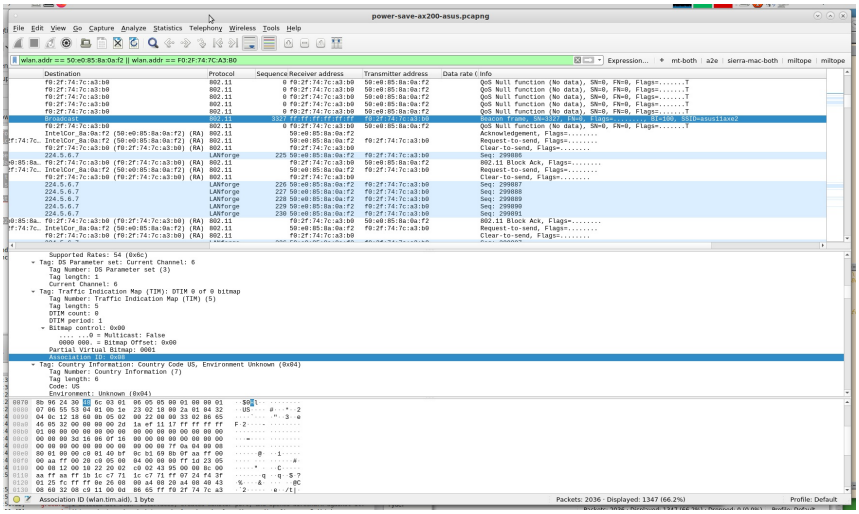
- Start sniffer, with optional filter to see all frames from AP BSSID or STA BSSID.
- Ping from upstream to STA, one packet per second.
- Ping latency should vary from near zero to the beacon interval since AP will store packets between when STA sleeps and wakes again. STA will wake after seeing indication in DTIM that it needs to receive (non multicast) frames.
- Packet loss should not be very high, this indicates AP is storing frames properly and that STA is waking properly.
- Ping from STA to upstream should not have much latency, because STA should wake each time it wants to send ICMP request, and stay awake long enough to receive the response before going back to sleep.

Packet Capture Analysis of Ucast to Mcast Capture with Buggy AP with Bad

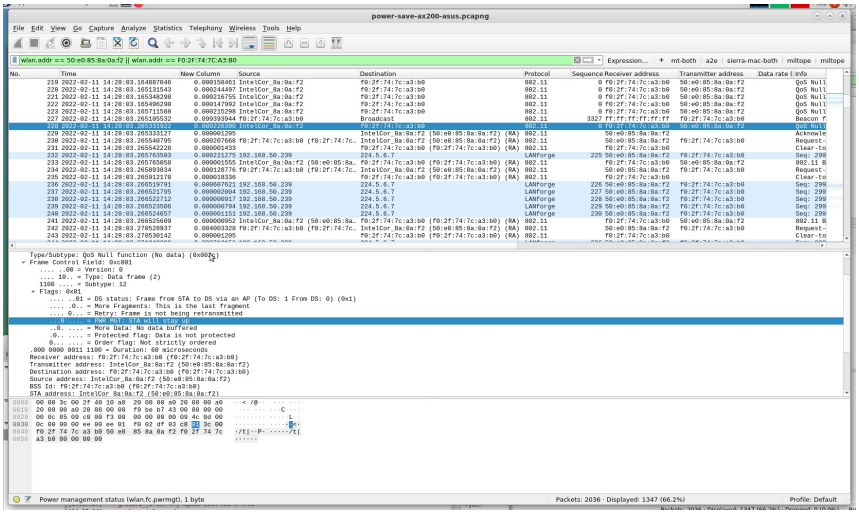
Ack on Frames

See 'power-save-ax200-asus.pcapng' file linked above. The captures 'power-save-ax200-sta-9984-lanforge-vap-unicast.pcapng' and 'power-save-mtk7921k-sta-9984-lanforge-vap-unicast.pcapng' show similar behaviour, but without the buggy lack of ACKs from the AP since a different AP (LANforge on QCA 9984 chipset) was used.

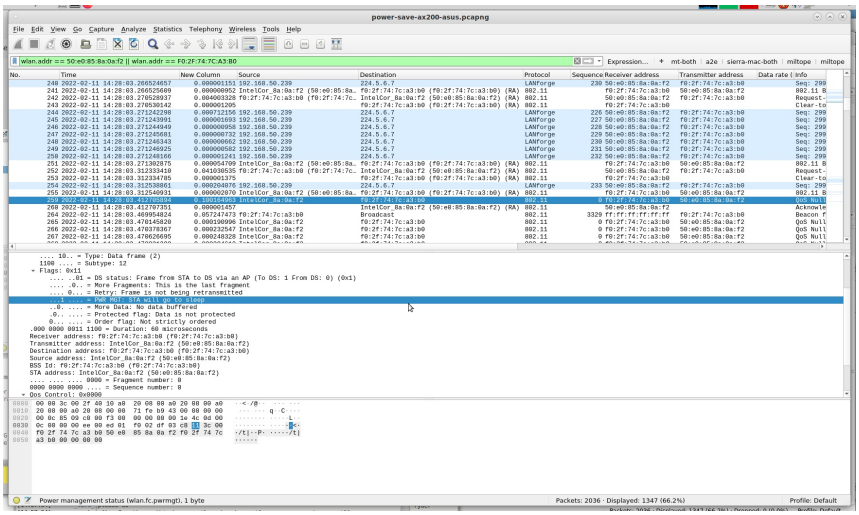
Starting with frame 227, a beacon frame, the AP indicates in the TIM IE that (unicast) data is waiting for the station with Association ID (AID) 0x8. This is the LANforge station in question, where it's AID is seen in the 'Probe' text dump in LANforge. This can also be seen by looking at the association frames, though this is not part of the capture. The TIM does not set the Multicast bit since this AP is converting multicast to unicast.



Frame 228 is from the station. It will always wake up to receive the beacon so that it can check the TIM element to see if it needs to stay awake and tell AP it needs to receive frames. Frame 228 is a few micro-seconds after the beacon, and STA tells the AP that is is waking up (because the AID was set in the TIM), and so AP can send traffic to it:



The multicast frames (converted to unicast in this case) are sent very soon after the STA wakes, see frames 232, 236 - 240. After the burst of multicast frames, the traffic goes quite. The station decides to go back to sleep at frame 259. Notice the interval between frames, and that it has been idle for a bit...

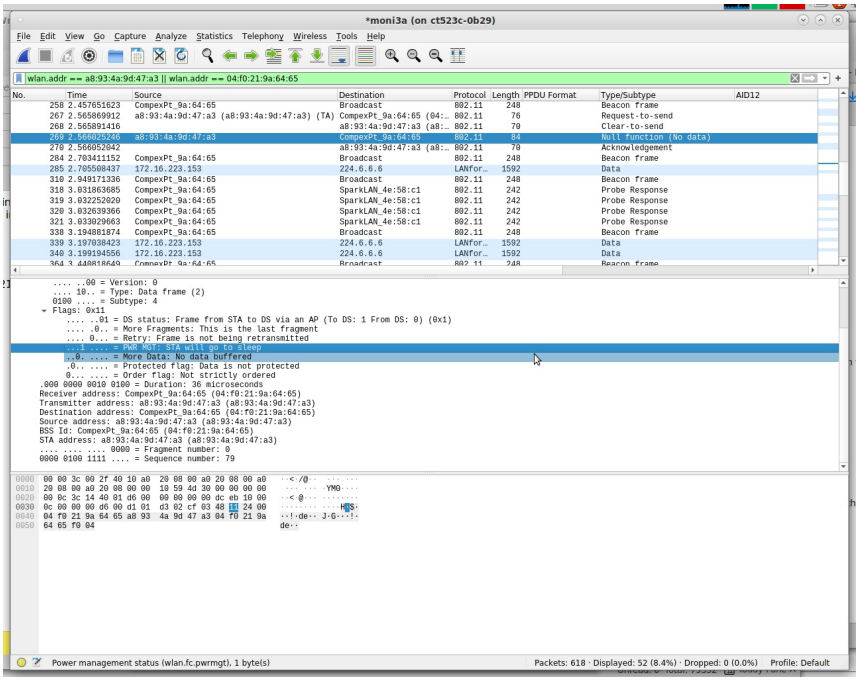


Packet Capture Analysis of Multicast Capture with Powersave

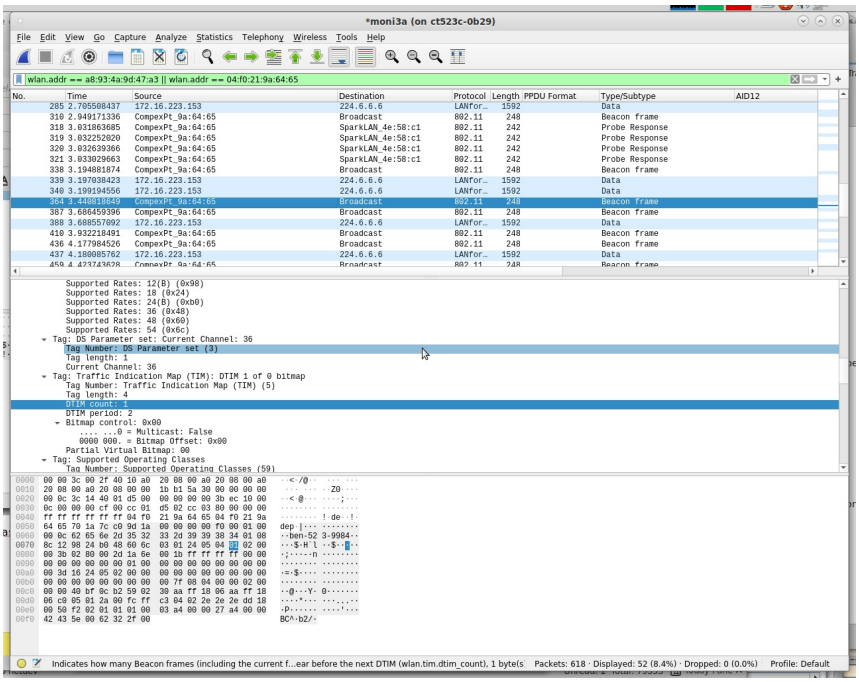
See file 'power-save-mtk7921k-sta-9984-lanforge-vap.pcapng', using filter: wlan.addr == a8:93:4a:9d:47:a3 || wlan.addr == 04:f0:21:9a:64:65

The AP in this setup is LANforge using QCA 9984 chipset radio. The AP is not set to convert multicast to unicast, so we get to inspect how multicast traffic is queued and sent to sleeping stations.

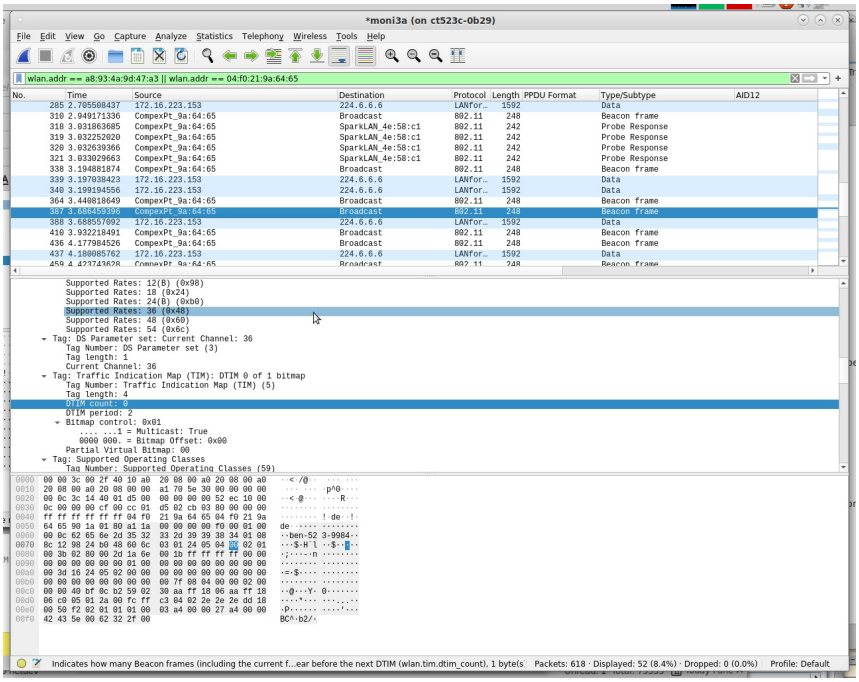
In frame 269, the station goes to sleep. It does not wake up again, at least not often, but it does wake up to hear beacons and pay attention to the DTIM so it can stay awake to receive multicast frames when needed. This was verified by ensuring that the multicast receiver endpoint on the station shows proper amount of received traffic.



Frame 364 shows DTIM with no multicast bit set, indicating no mcast frames are queued. And, DTIM count is 1, which means that stations do not need to wake now, but do need to wake for next DTIM period.



Frame 365 is next beacon, and it shows mcast frames are queued, and the DTIM count indicates that stations should wake NOW to receive the queued frames.

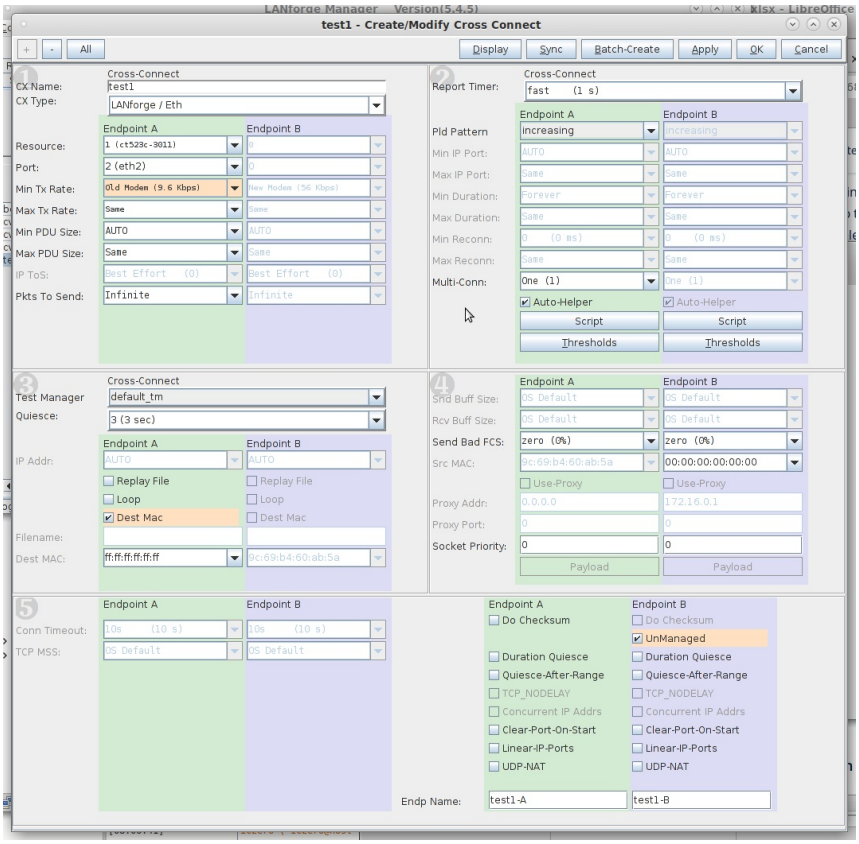


And right after the beacon, the mcast frame is sent, as expected. Later, if station needs to send a frame, it would send a Null-Func frame with the 'STA is awake' flag set, and if AP needed to send a unicast frame, the AID of the STA would be added to the TIM element and so STA would wake for that as well….

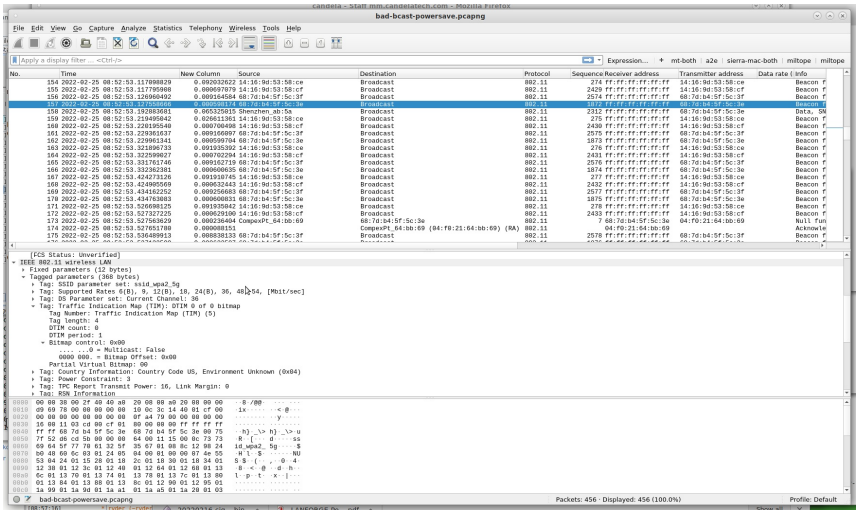
Packet capture for buggy broadcast power-save behavior:

See file linked near the top: bad-bcast-powersave.pcapng

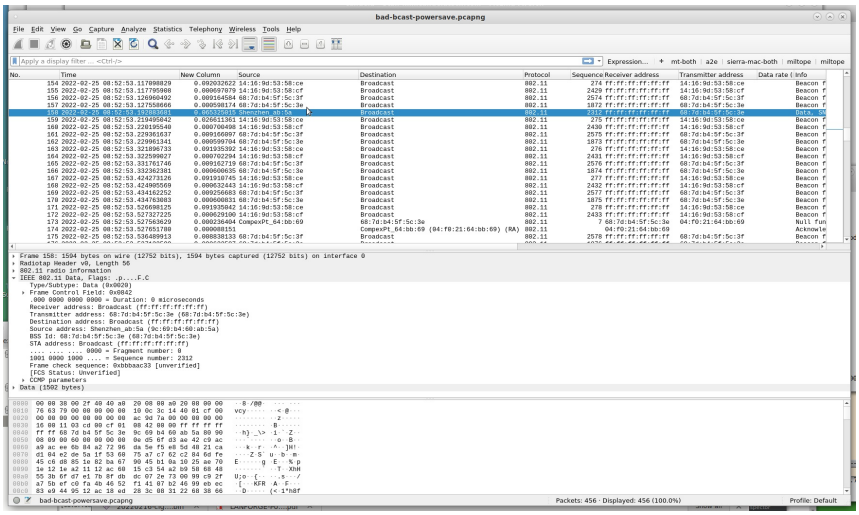
LANforge is configured to generate slow broadcast traffic in downstream direction using LANforge-Eth layer-3 connection type, with Dest Mac specified as all FFs, and 'b' side as un-managed. This causes A side to blindly send out these frames, the B side does not actually attempt to receive it.



This AP is not buffering broadcast packets and sending them after the beacon like it is supposed to. Frame 157 (beacon) does not have Multicast bit set in its TIM:

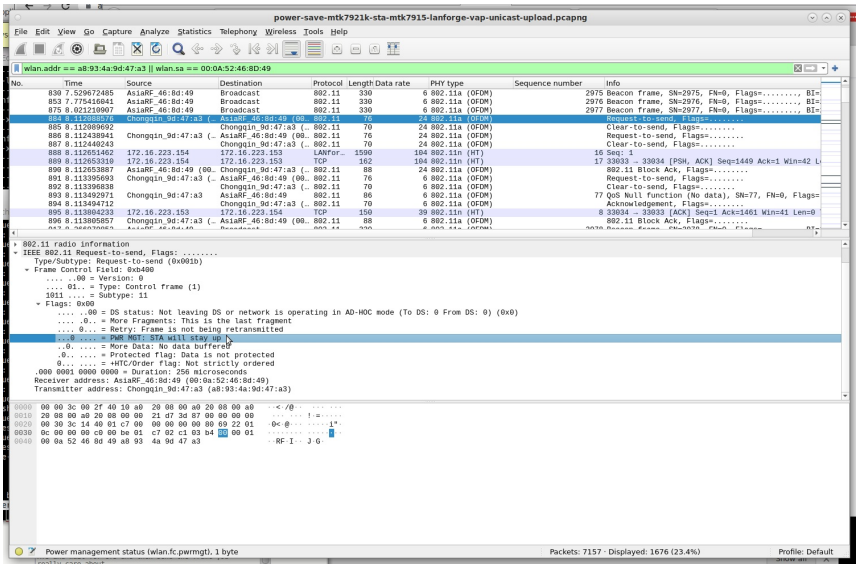


Frame 158 (broadcast frame) arrives 65ms after the beacon, indicating that the AP did not buffer the frame properly.

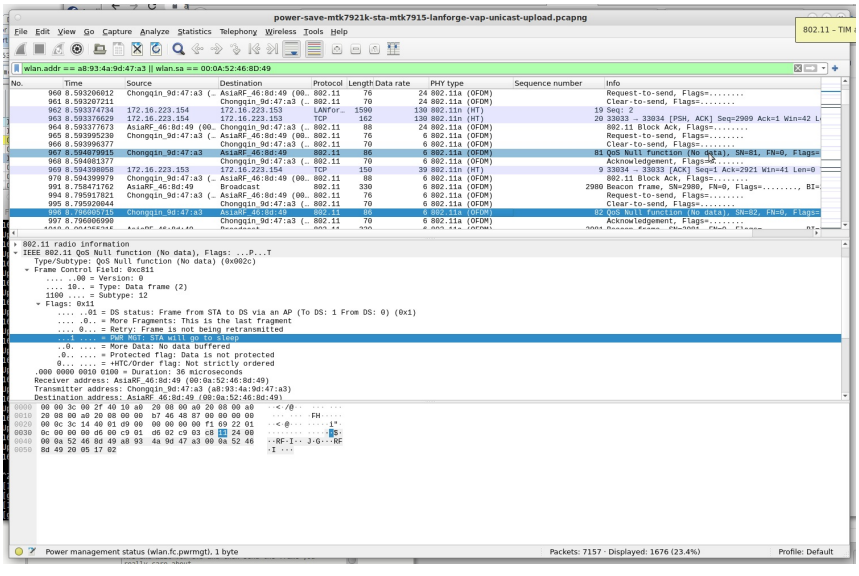


Packet capture for Wake/Sleep testing in upload direction:

See file 'power-save-mtk7921k-sta-mtk7915-lanforge-vap-unicast-upload.pcapng' This capture shows mtk7921k station talking to LANforge mtk7915 VAP. The capture filter is: **wlan.addr == a8:93:4a:9d:47:a3 || wlan.sa == 00:0A:52:46:8D:4**. The capture starts with the TCP traffic idle and the station in sleep mode. Frame 884 is STA waking up, and it starts sending TCP traffic frame 888.



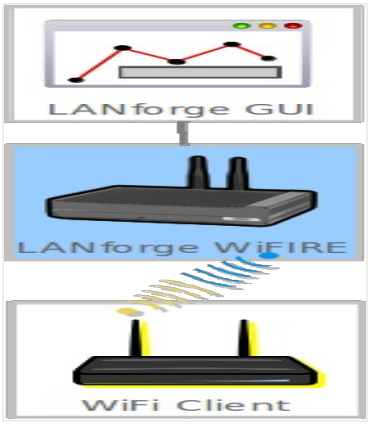
At frame 996, the TCP traffic has quiesced for a bit, and STA goes back to sleep.

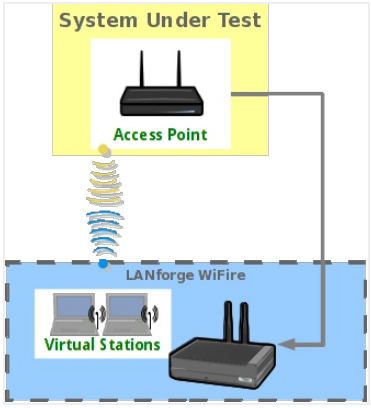


Using LANforge python script to do packet captures of OFDMA and regular WiFi Stations

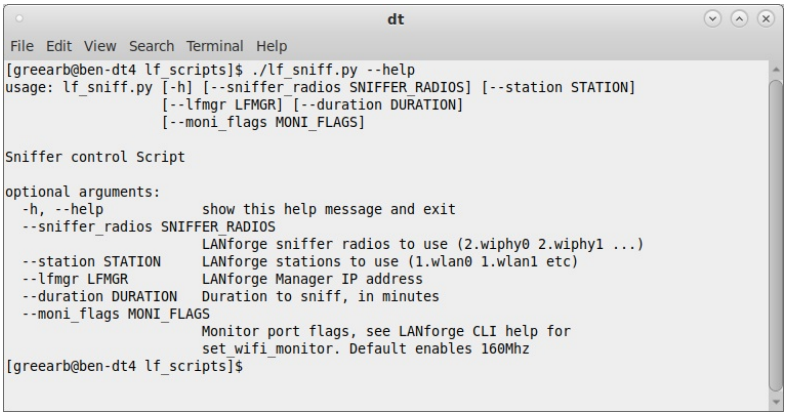
Goal: Sniff wireless traffic to/from LANforge stations using additional LANforge Radios in monitor mode.

This script automates creating WiFi monitor devices correctly configured for the station's channel, and in the case of OFDMA, the AID and BSSID will also be configured in the sniffing radio. This script requires LANforge 5.4.2 or higher and an appropriate set of WiFi radios.





1. Create and configure stations to talk to your DUT. You may create these through Chamber View scenarios or other methods. Please refer to introductory cookbooks for more information on this if you have questions.
2. Run lf_sniff.py script.
   A. Change directory to the /home/lanforge/scripts directory (or other location if you have installed scripts elsewhere), and run the lf_sniff.py script with the --help argument to understand your options..

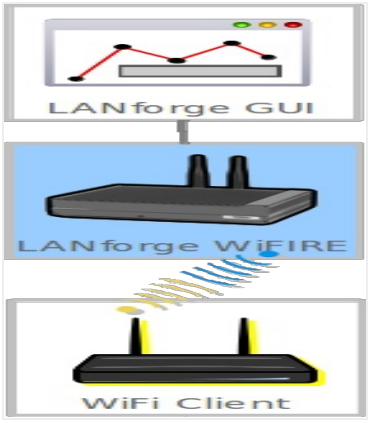


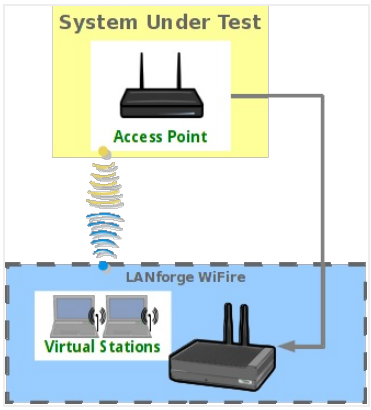
   B. Run the script with arguments for your test case. The basic idea is to provide a list of stations you wish to sniff, and a list of matching radios that you wish to use as sniffers. The script will query the station to find the channel, AID and BSSID and will then create WiFi monitor devices on the sniffer radios. A command-line packet capture tool will then be started and write the captured packets to disk for the requested duration. When the duration is complete, the packet capture process will exit. You can then post-process the packet captures, open them with wireshark, etc.
   Examples
   Sniff sta00000 on LANforge resource 1 using radio wiphy0 on LANforge resource 2:
   `./lf_sniff.py --lfmgr 192.168.100.238 --station "1.sta00000" --sniffer_radios "2.wiphy0" --duration 0.25`

Using LANforge python script to do packet captures then analyze to determine if the traffic was OFDMA, MU-MIMO or Single User

Goal: Sniff wireless traffic to/from LANforge stations using additional LANforge Radios in monitor mode.

This script automates creating WiFi monitor devices correctly configured for the station's channel, and in the case of OFDMA, the AID and BSSID will also be configured in the sniffing radio. The OFDMA vs MU-MIMO analysis is performed by the wifi diagnostics script. Both of these scripts require LANforge 5.4.2 or higher and an appropriate set of WiFi radios.



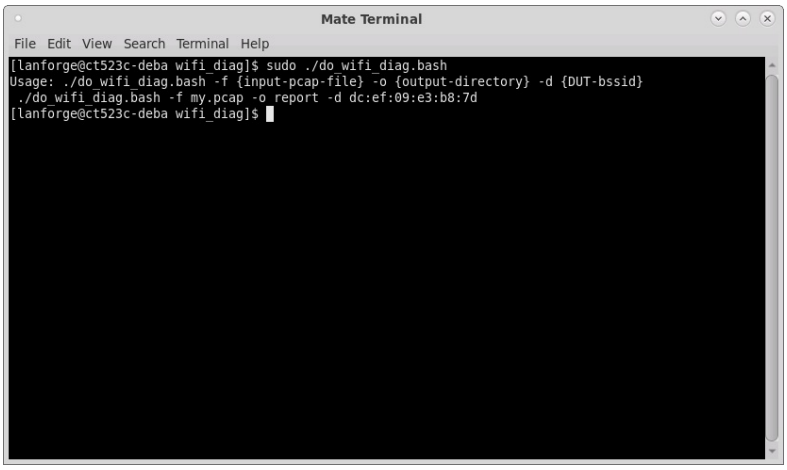


1. Create and configure stations to talk to your DUT. You may create these through Chamber View scenarios or other methods. Please refer to introductory cookbooks for more information on this if you have questions.
2. Run lf_sniff.py script.
   A. Change directory to the /home/lanforge/scripts directory (or other location if you have installed scripts elsewhere), and run the lf_sniff.py script with the --help argument to understand your options..
   B. Run the script with arguments for your test case. The basic idea is to provide a list of stations you wish to sniff, and a list of matching radios that you wish to use as sniffers. The script will query the station to find the channel, AID and BSSID and will then create WiFi monitor devices on the sniffer radios. A command-line packet capture tool will then be started and write the captured packets to disk for the requested duration. When the duration is complete, the packet capture process will exit. You can then post-process the packet captures, open them with wireshark, etc.
   Examples
   Sniff sta00000 on LANforge resource 1 using radio wiphy0 on LANforge resource 2 for 15 seconds:

```
./lf_sniff.py --lfmgr 192.168.100.238 --station "1.sta00000" \
 --sniffer_radios "2.wiphy0" --duration 0.25
```

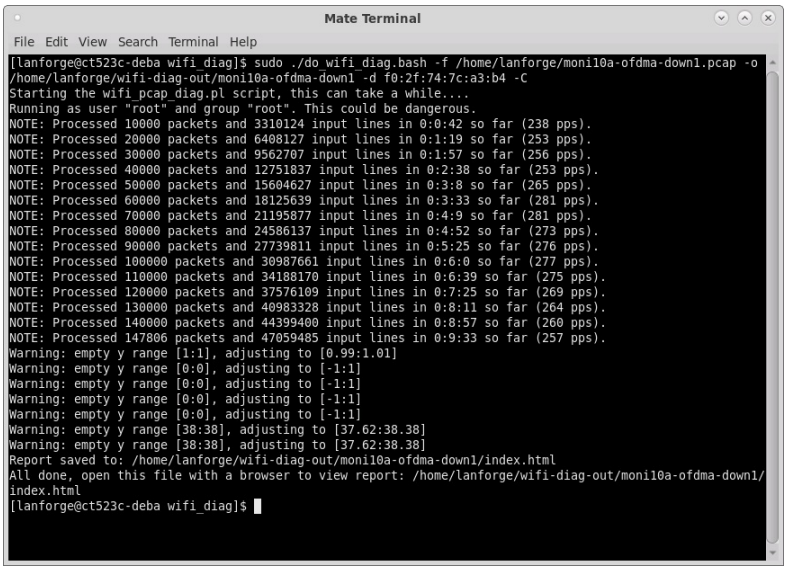
3. Run do_wifi_diag.bash script.

A. Change directory to the /home/lanforge/scripts/wifi_diag directory, and run the bash script without any arguments to view the usage help.



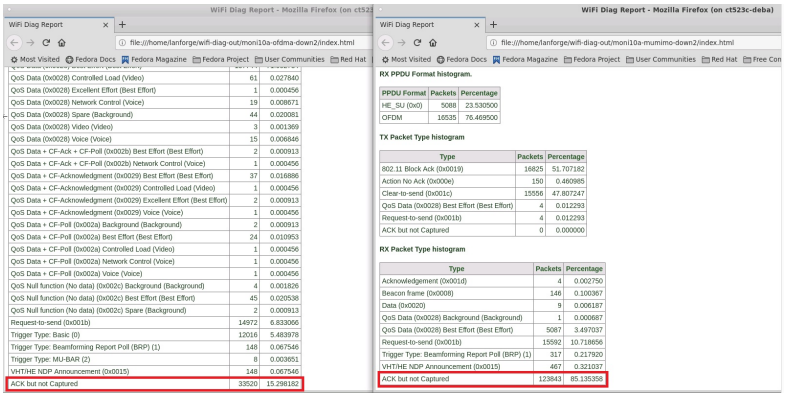
B. Run the bash script with correct location of the pcap file obtained with lf_sniff.py and with the correct BSSID of the AP used as the DUT. Depending on your system processing power and the pcap file size, the wifi diag script will take some time to complete.

C. When the wifi diag script is complete, you can open the index.html file to view the results.



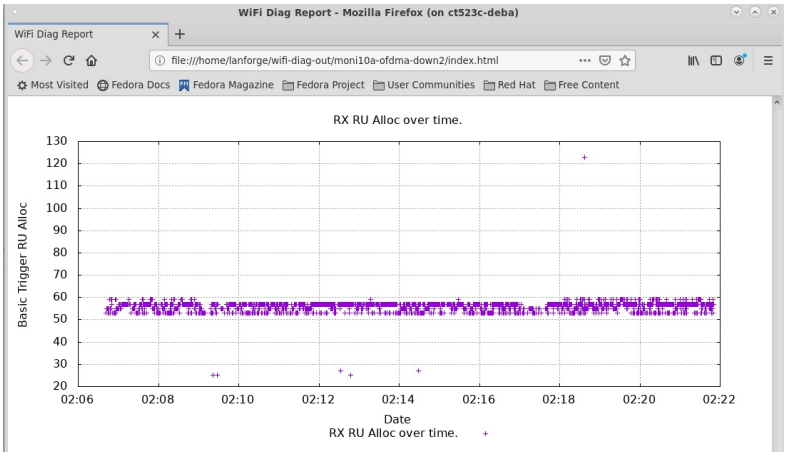
4. Interpreting the report for OFDMA or MU-MIMO operation.

A. To determine if the packet capture contains OFDMA or MU-MIMO operation, scroll down in the report to the table labeled "RX Packet Type Histogram" and the element "ACK but not captured" which is a count of the number of Block ACKs captured without capturing the corresponding Data frames. If the report shows a large percentage of "ACK but not captured", this indicates MU-MIMO operation because the data frames are not able to be captured. If there is a lower percentage, than the traffic is likely OFDMA.



B. "ACK but not captured" is also represented in the graph later in the report labeled "RX Dummy Packets".

C. The graph labeled "RX RU Alloc over time" also indicates OFDMA operation because it shows when frames were assigned a Resource Unit allocation which is the main trait of OFDMA operation.



6GHz Packet Capture

Goal: 6GHz Packet Capture Using LANforge Radios in Monitor Mode.

Depending on the NIC, there are different steps required in order to capture on 6GHz frequencies.
Intel AX210/BE200 - Require a station on the parent interface to scan and determine that it is in a US regulatory domain before a monitor mode interface can be assigned a 6GHz frequency.
Mediatek 7921/7922 - No regulatory domain restriction at this time.

1. Clone and get latest copy of lanforge-scripts.
   A. cd /home/lanforge
   B. git clone https://github.com/greearb/lanforge-scripts
   C. cd /home/lanforge/lanforge-scripts/py-scripts

2. Run lf_sniff_radio.py script.
   A. Run the lf_sniff_radio.py script with the --help argument to understand your options or pip install any missing modules as needed.

```
[root@ibase-i5-f36 py-scripts]# ./lf_sniff_radio.py --help
 usage: ./lf_sniff_radio.py
        --mgr localhost
        --mgr_port 8080
        --radio wiphy0
        --outfile /home/lanforge/test_sniff.pcap
        --duration 1
        --channel 36
        --channel_bw 40
        --center_freq 5190
        --radio_mode AUTO
        --monitor_name Sniffer0

        AX210 sniff command
        ===================

        ./lf_sniff_radio.py
        --mgr 192.168.0.104
        --mgr_port 8080
        --radio wiphy7
        --outfile /home/lanforge/sniff_6G_80.pcap
        --duration 20
        //--channel 1e
        --channel_bw 80
        --channel_freq 5955
        --center_freq 5985
        --radio_mode AUTO
        --monitor_name SNIFF_6G_80
```

B. Run the script with arguments for your test case.
To sniff on 6GHz with an AX210 NIC on resource 1, wiphy2:

```
./lf_sniff_radio.py --mgr 192.168.101.197 --radio "1.wiphy2" \
--outfile sniff_6G-AX210.pcapng --duration 10 --channel 37e --ax210 \
--num_stations 1 --ssid test --ax210_scan_time 20
```

To sniff on 6GHz with an MTK7921/MTK7922 NIC on resource 6, wiphy0:

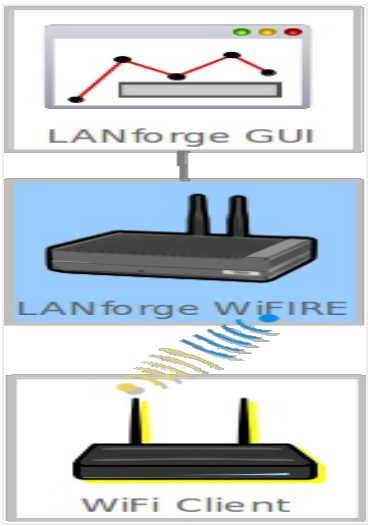
```
./lf_sniff_radio.py --mgr 192.168.100.193 --radio "6.wiphy0" \
--outfile sniff_6G-MTK.pcapng --duration 10 --channel 133e
```

C. For more information, see this guide: sniffer_manual.pdf

WiFi: Inflight Entertainment: video demonstration

Goal: Learn about how to emulate a crowded WiFi environment that might occur in an airplane cabin.

Watch a demonstration of how to model a roaming WiFi system using multiple virtual APs that might be present in a transportation system. This scenario requires LANforge version 5.3.9, four isolation chambers, programmable attenuators, and multiple LANforges for virtual APs and stations and as a content server.

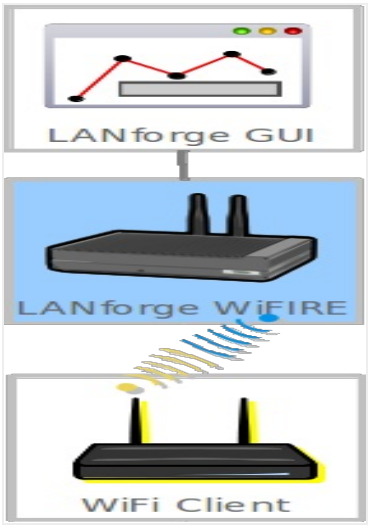


WiFi Transportation demonstration

WiFi: Gaming Test: video demonstration

Goal: Learn about how to combine the WAN emulation and programmable attenuation to emulate different network environments for game consoles.

Watch a demonstration of how to modify the gaming experience using WAN links to drop packets and combining that with attenuating the wifi signal in an isolation chamber. This scenario requires LANforge version 5.3.9, two isolation chambers, one or more programmable attenuators a DUT and three mesh AP nodes.



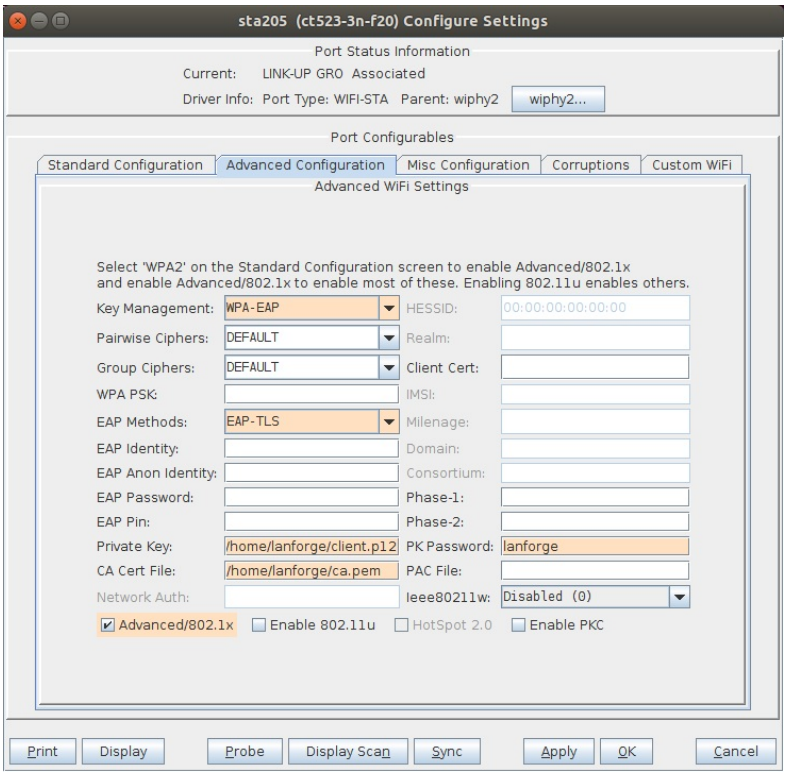
WiFi Gaming Test demonstration

Setting up a RADIUS Server

Goal: To set up a LANforge wireless access point with a local RADIUS server.

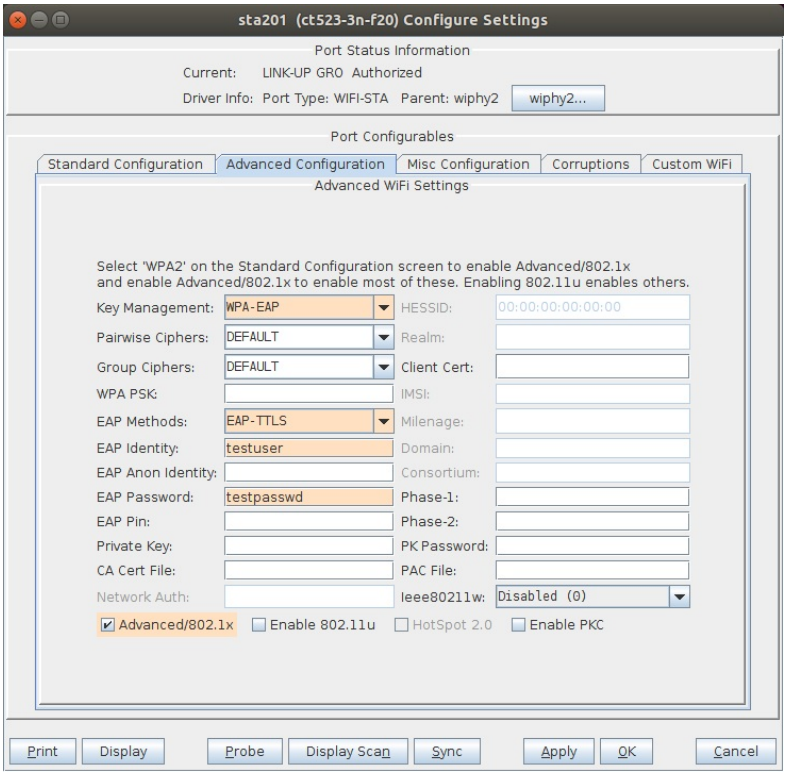
1. **The LANforge auto-install --do_radius option will setup FreeRADIUS on the LANforge system with two example EAP methods, EAP-TLS and EAP-TTLS**
2. **The config files for FreeRADIUS are located in /etc/raddb**
   A. /etc/raddb/certs contains the files necessary for **EAP-TLS**
   B. The LANforge auto-install copies the necessary files into /home/lanforge for use by LANforge wireless clients.

C. For **EAP-TLS**, use client.p12 as the client's Private Key and ca.pem as the client's CA Cert File. The Private Key password is lanforge



D. /etc/raddb/users contains the user and password for **EAP-TTLS**

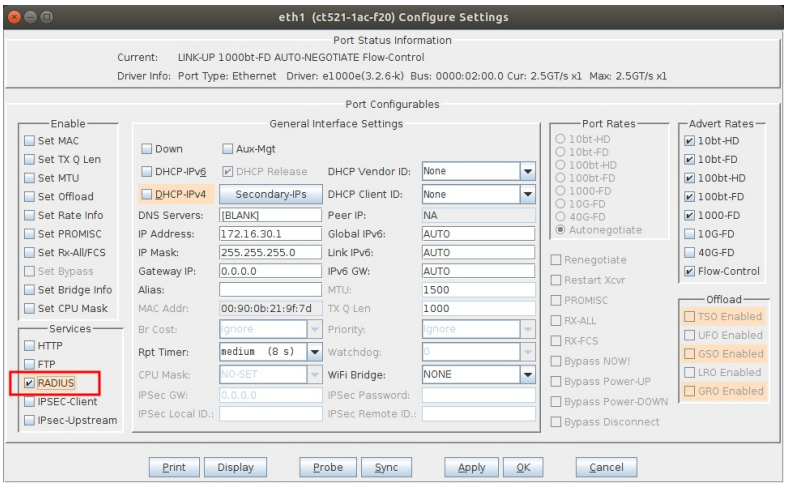
E. The example **EAP-TTLS** user is testuser with password testpasswd. Additional entries can be added to the users file, then restart FreeRADIUS with systemctl restart radiusd.service



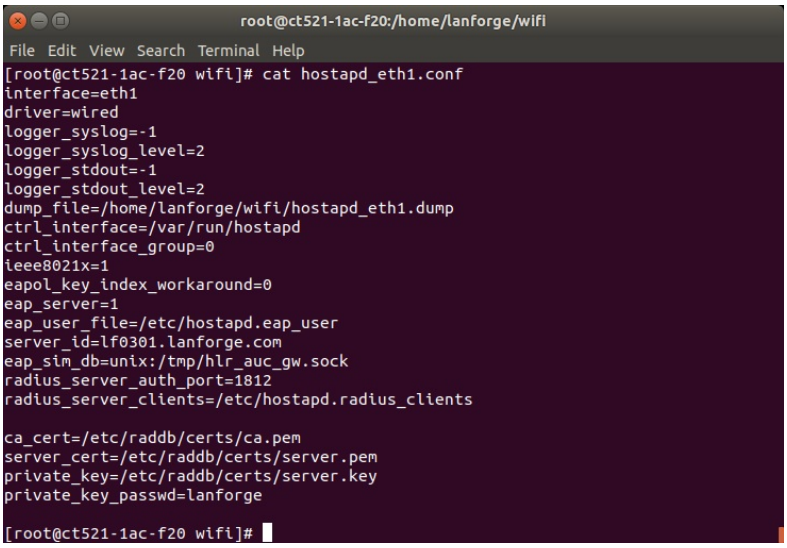
3. **An alternative to FreeRADIUS is to use the hostapd RADIUS server.**

A. Stop the FreeRADIUS service with systemctl stop radiusd.service

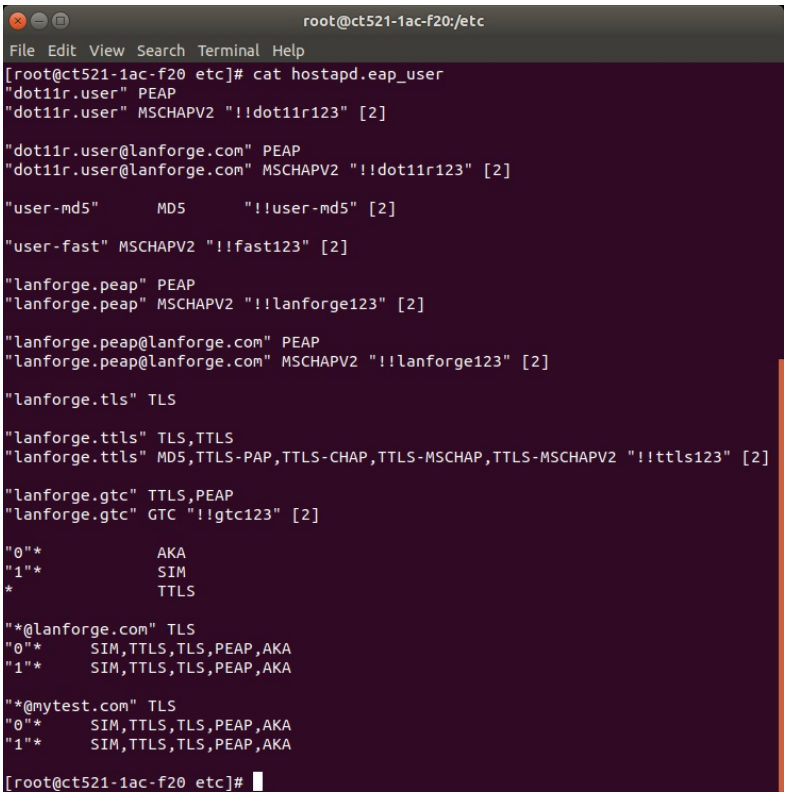
B. Modify the interface to use for the hostapd process and select the RADIUS checkbox.



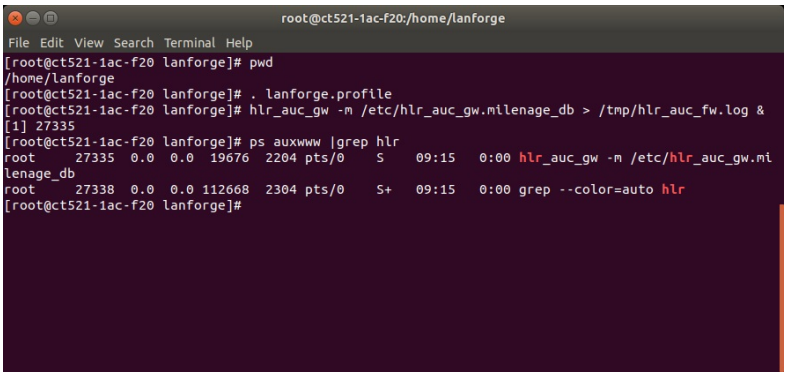
C. Create a hostapd_<port-name>.conf file in the /home/lanforge/wifi directory with the following info.



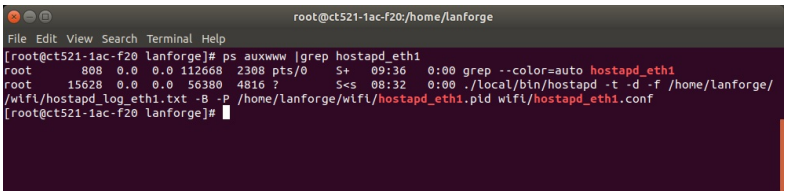
D. Setup the desired EAP methods and passwords in the /etc/hostapd.eap_users file.



E. If using **EAP-SIM** or **EAP-AKA**, verify entries in the /etc/hlr_auc_gw.milenage_db file, then start the HLR tool.

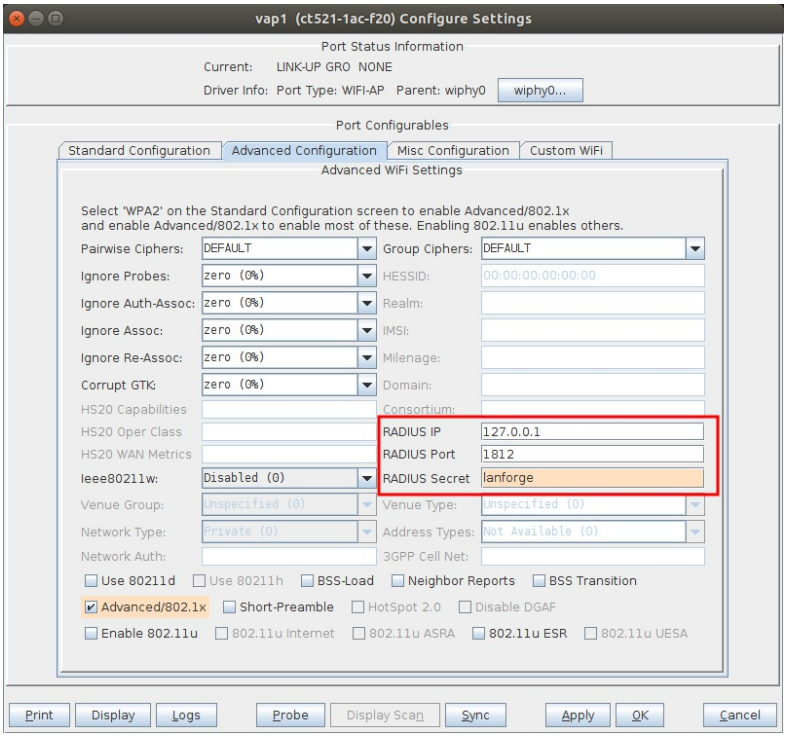


F. Verify the hostapd process is running for the interface selected for the RADIUS server, here it is eth1.



4. **Whether you use FreeRADIUS or hostapd RADIUS, setup your AP with the RADIUS server's IP address and port.**

A. If using a LANforge AP on the same system as the RADIUS server, then the AP will address the RADIUS server at localhost or 127.0.0.1 with port 1812.



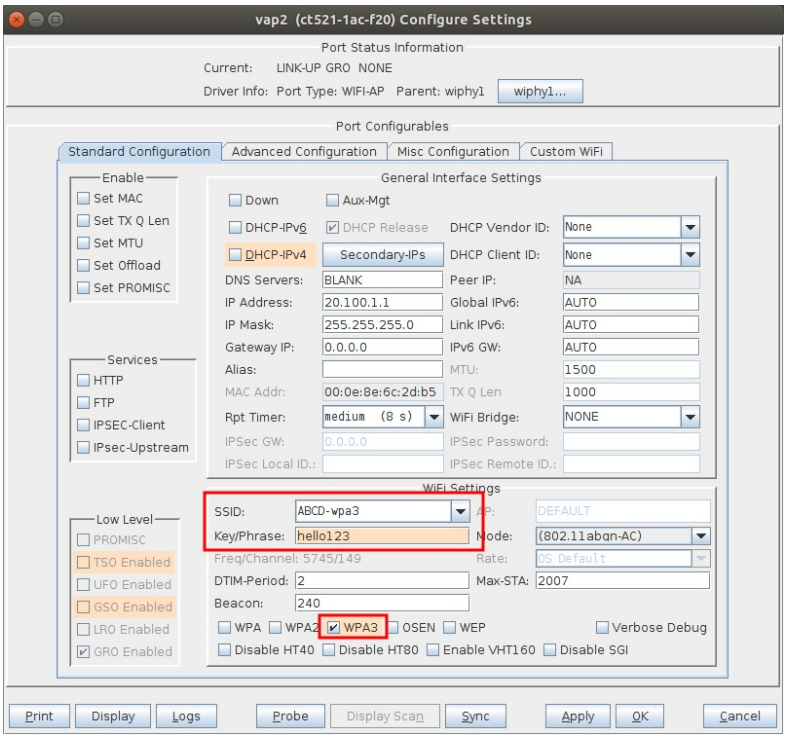
B. If using an external AP or WLAN Controller, then configure the device to address the RADIUS server on the network connected to a LANforge interface configured for RADIUS.

Setting up WPA3

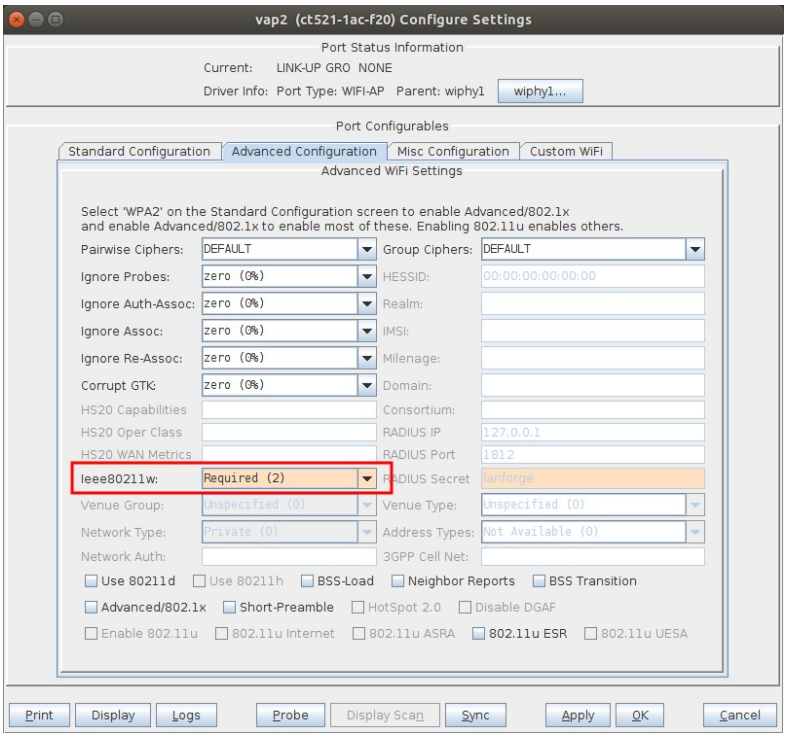
Goal: To set up LANforge wireless access points and clients with WPA3.

This example will cover WPA3-Personal, WPA3-Enterprise and OWE.
For an introduction or review of WPA3, see Hemant Chaskar's WLPC video.

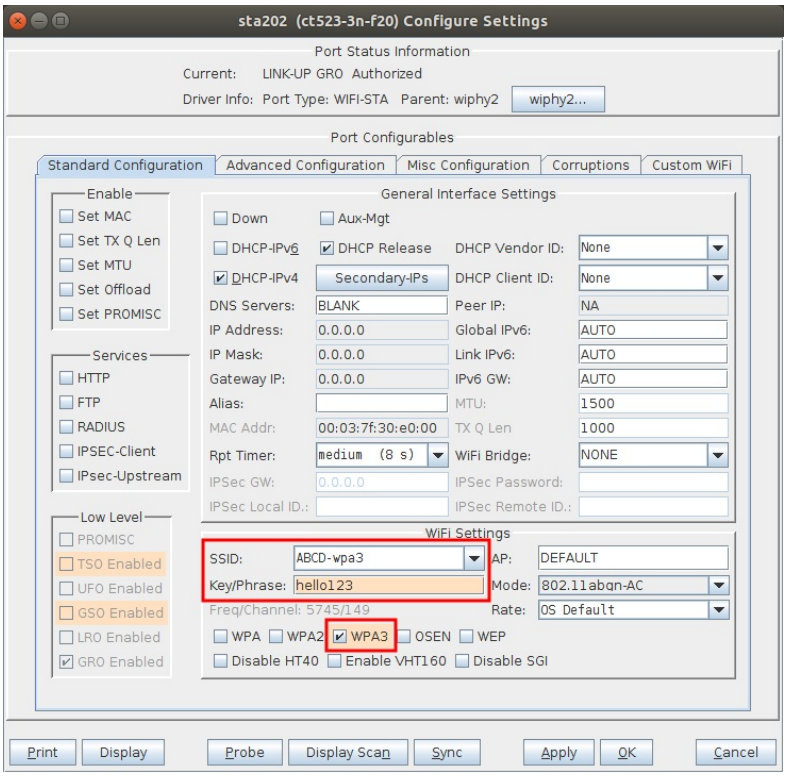
1. **WPA3-Personal for a VAP and a STA client.**
   A. Setup the VAP with SSID, WPA3 security and a PSK.



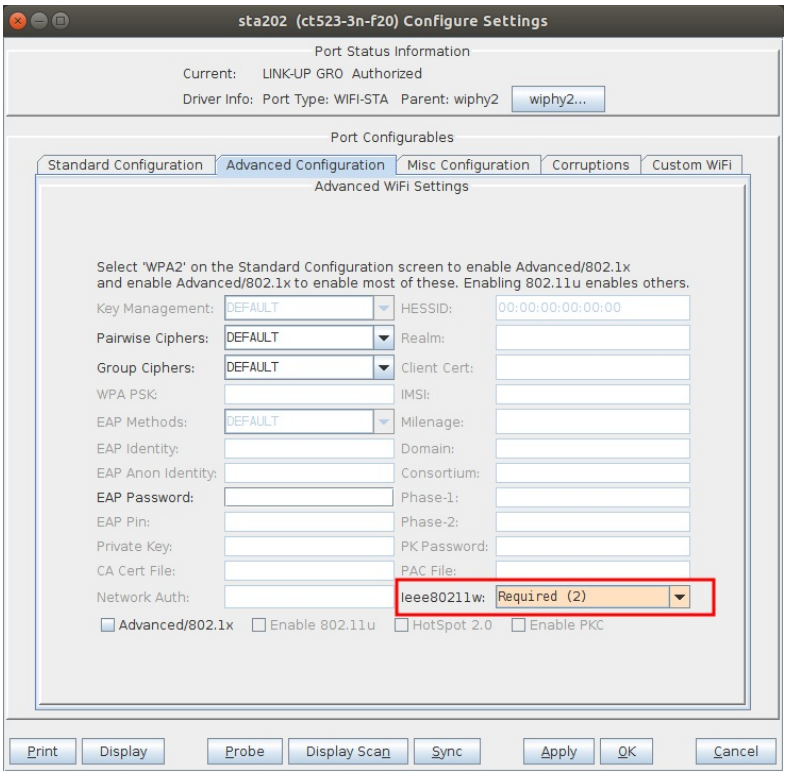
B. Setup the VAP with 11w PMF option Required.



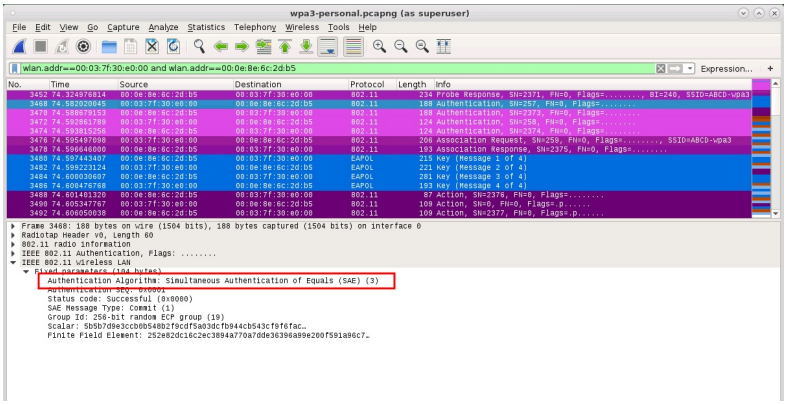
C. Setup the STA with SSID, WPA3 security and a PSK.



D. Setup the STA with 11w PMF option Required.



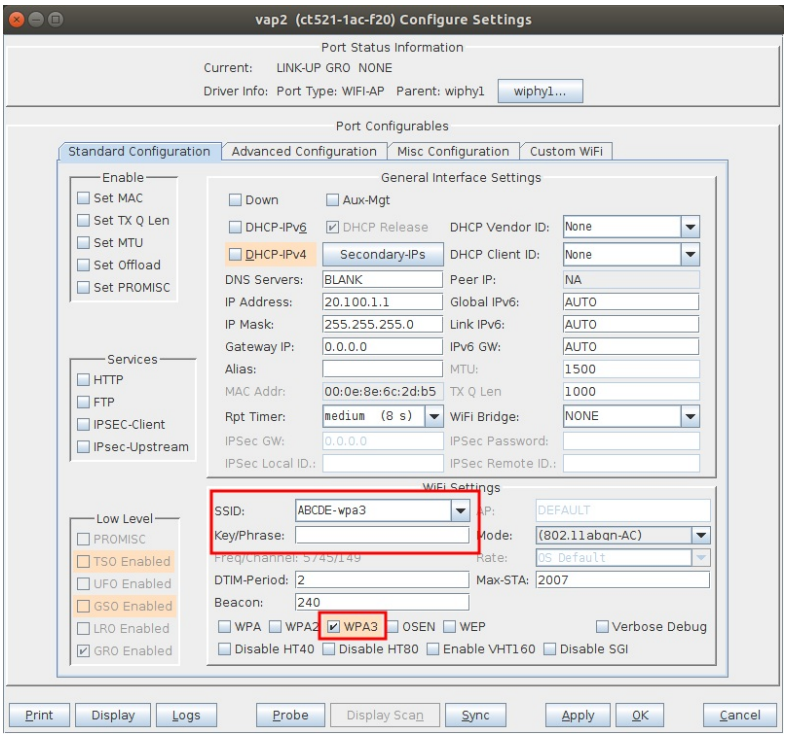
E. A capture of the association.



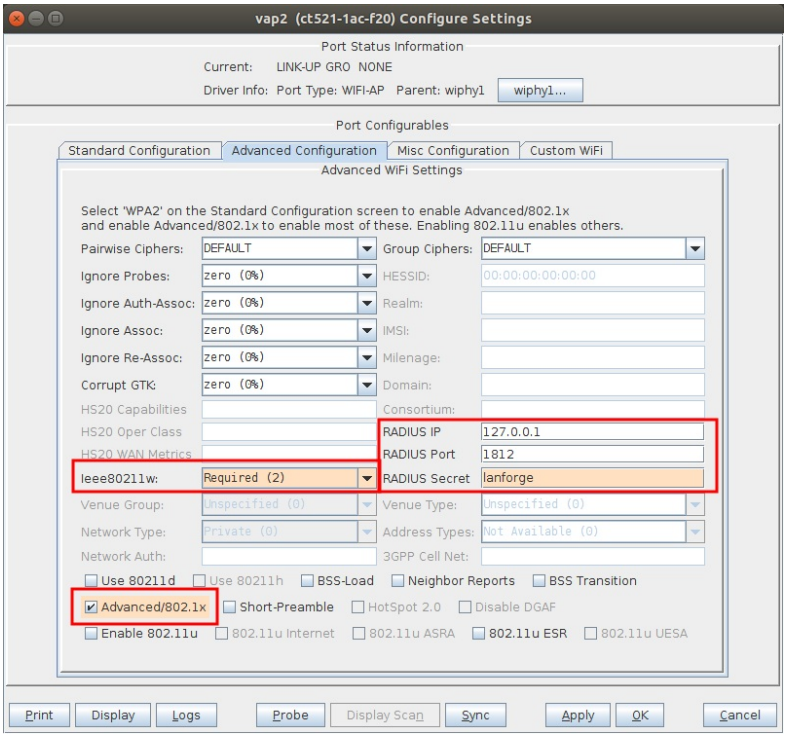
2. **WPA3-Enterprise for a VAP and a STA client.**

A. Setup a RADIUS server for the VAP. This example uses a LANforge hostapd RADIUS server on the same system as the VAP.

B. Setup the VAP with WPA3 security and no PSK on the standard configuration screen.



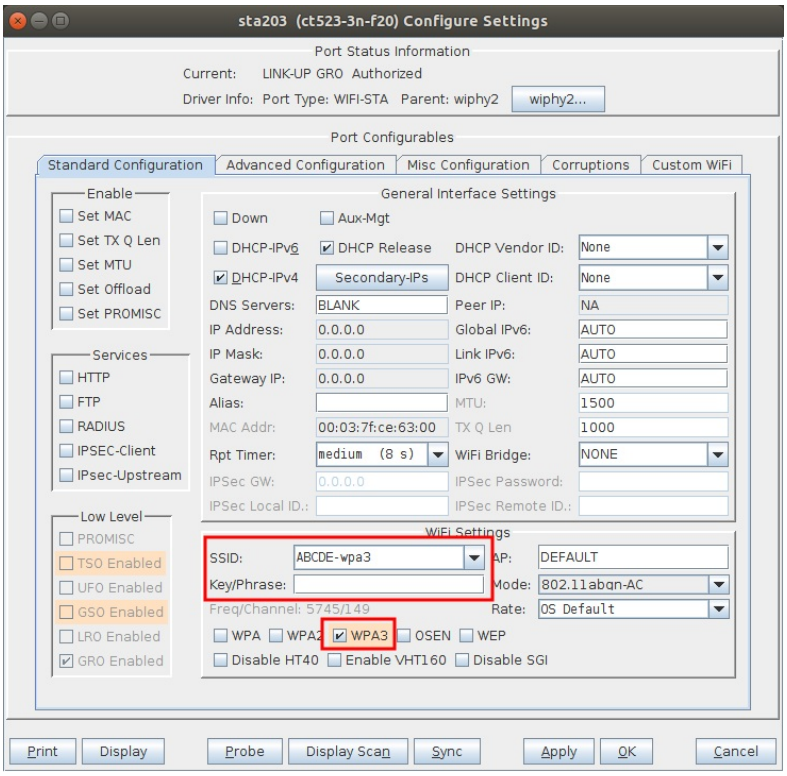
C. Setup the VAP with 11w PMF option Required and select the checkbox for Advanced/802.1X which will also inform the VAP where its RADIUS server is located. In this example the LANforge hostapd RADIUS server is on the localhost.



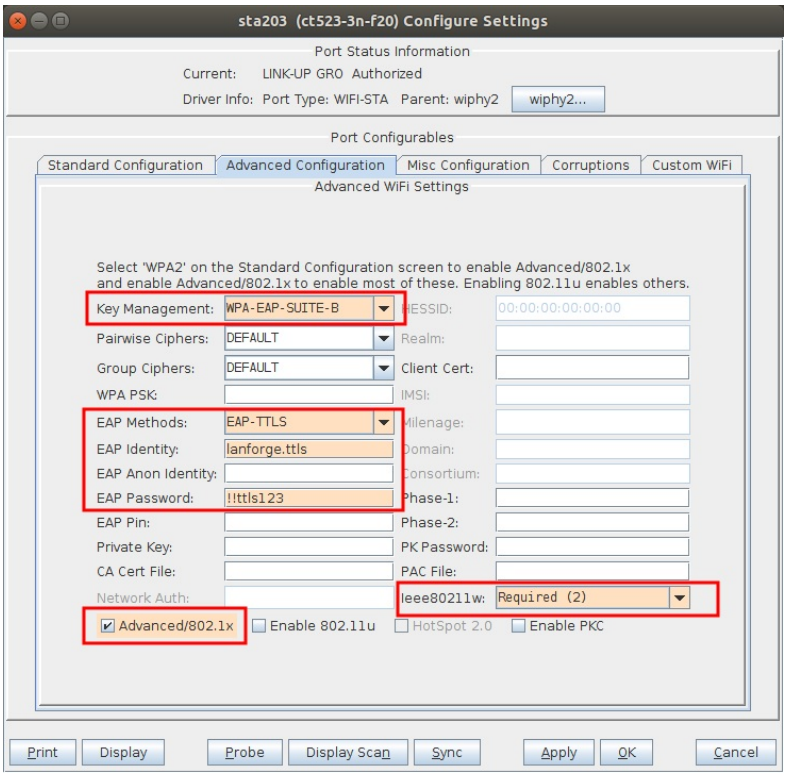
D. After enabling Advanced/802.1X, the VAP is automatically configured for both WPA-EAP-SUITE-B and WPA-EAP-SUITE-B-192 as shown in the back-end configuration for the VAP.

```
cat /home/lanforge/wifi/hostapd_vap2.conf
...
wpa_key_mgmt=WPA-EAP-SUITE-B WPA-EAP-SUITE-B-192
...
```

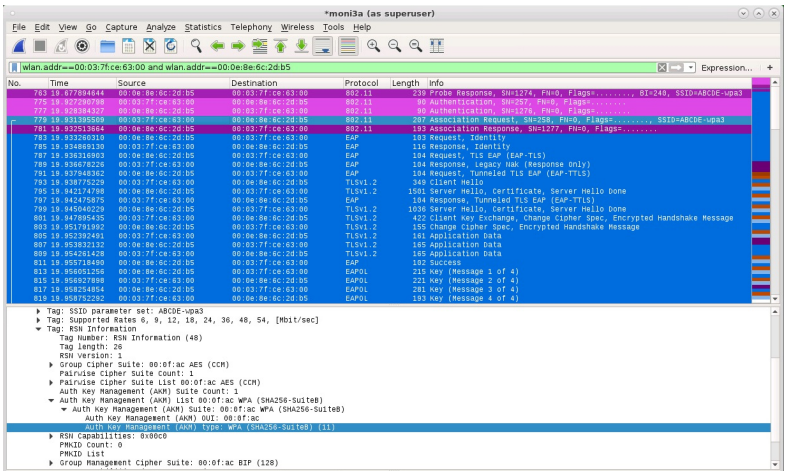
E. Setup the STA with WPA3 security and no PSK on the standard configuration screen.



F. Setup the STA with 11w PMF option Required and select the checkbox for Advanced/802.1X which allows choosing the Key Management scheme and EAP Method. Here the STA is setup to use WPA-EAP-SUITE-B with EAP-TTLS and a user identity and password that were configured with the RADIUS server setup.

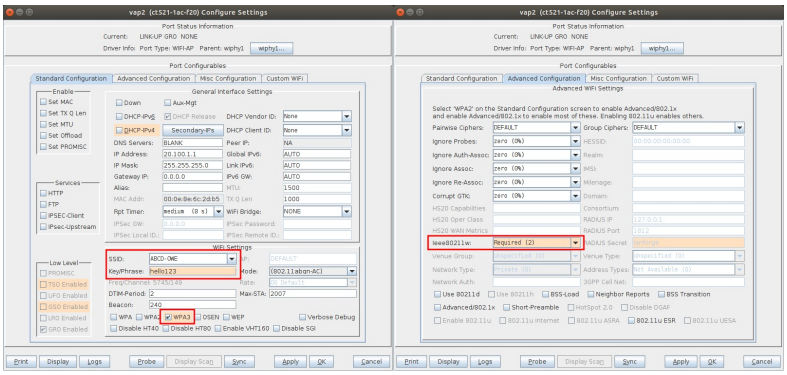


G. A capture of the association.



3. **WPA3 OWE - Opportunistic Wireless Encryption.**

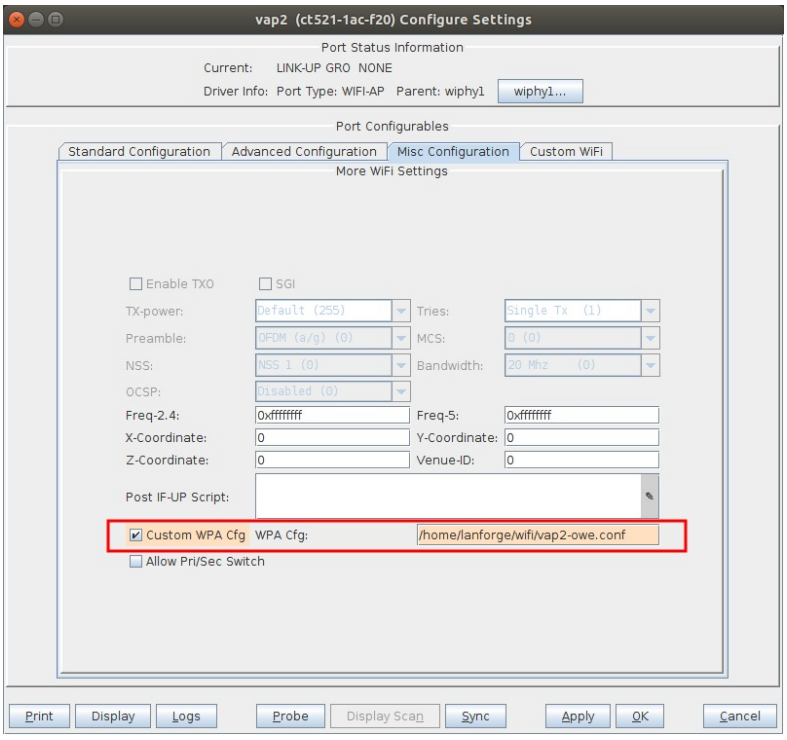
   A. Setup the VAP with WPA3 security and a PSK on the standard configuration, then select option 11w PMF option Required on the advanced configuration, then admin up the VAP.



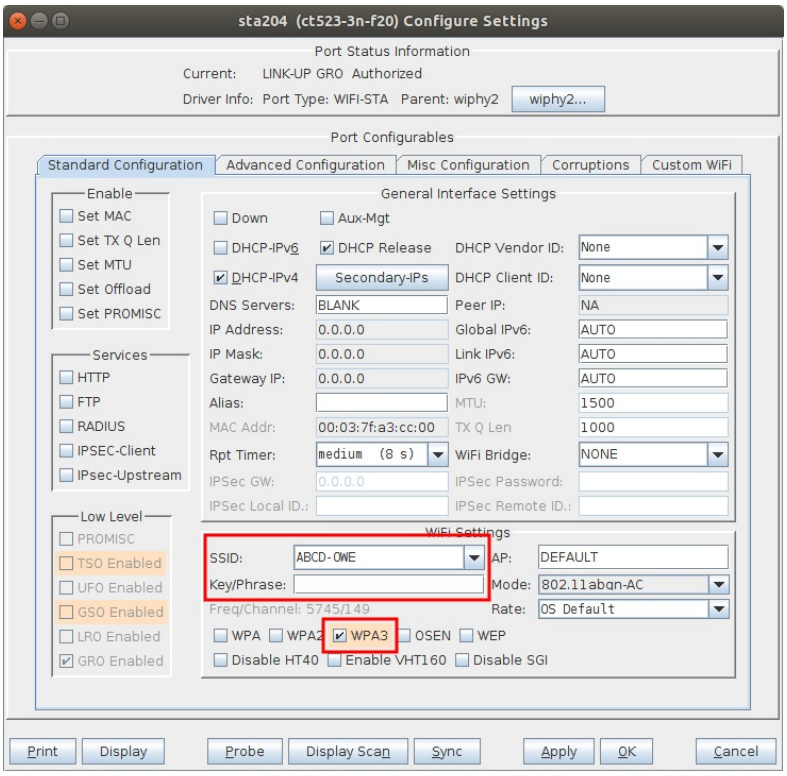
   B. Copy the back-end config file for the VAP to a new filename and edit the wpa_key_mgmt from SAE to OWE.

```
cd /home/lanforge/wifi
cp hostapd_vap2.conf vap2-owe.conf
vi vap2-owe.conf
wpa_key_mgmt=OWE
```

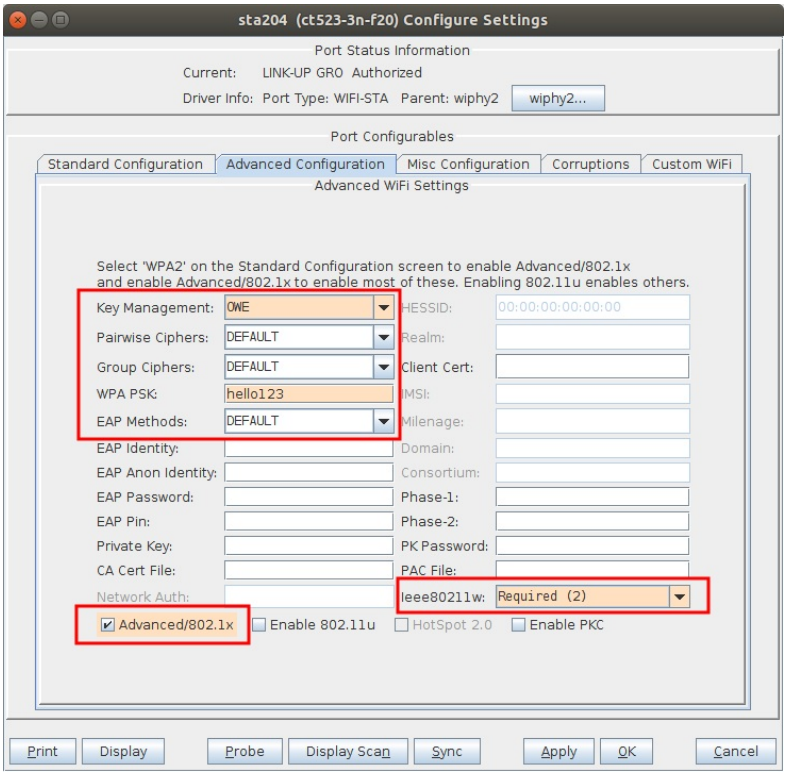
   C. Modify the VAP and select Custom WPA Cfg on the Misc Configuration screen then type in the location of the new VAP config file.



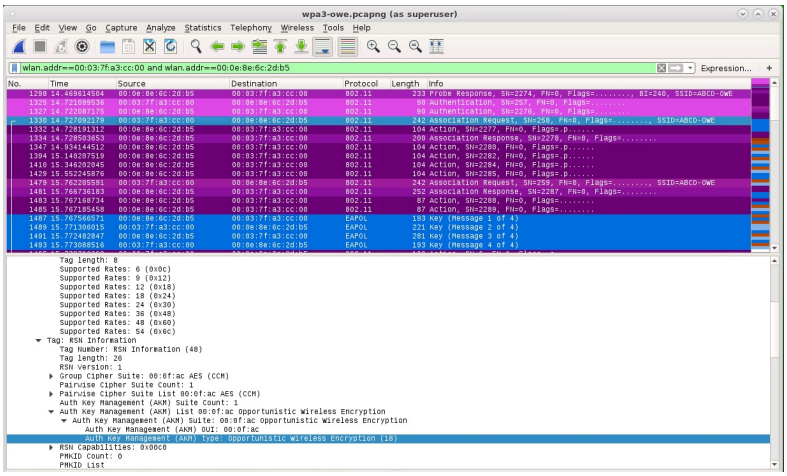
D. Setup the STA with WPA3 security and no PSK on the standard configuration screen.



E. Setup the STA with 11w PMF option Required and select the checkbox for Advanced/802.1X which allows choosing the Key Management scheme. Here the STA will use OWE and a WPA PSK.



F. A capture of the association.



Wifi Roaming with Opportunistic Key Caching (OKC)

Goal: Show how LANforge can emulate an OKC VAP or OKC STA then observe the different results when OKC is enabled or not.

Opportunistic Key Caching (OKC) is a fast roaming solution that is one predecessor to 802.11r Fast BSS Transition. OKC is also referred to as Proactive Key Caching (PKC). Here we will demonstrate the following four scenarios with OKC:

1. OKC on VAP and NOT on STA
2. OKC on both VAP and STA
3. OKC disabled on both VAP and STA
4. OKC disabled on VAP but enabled on STA

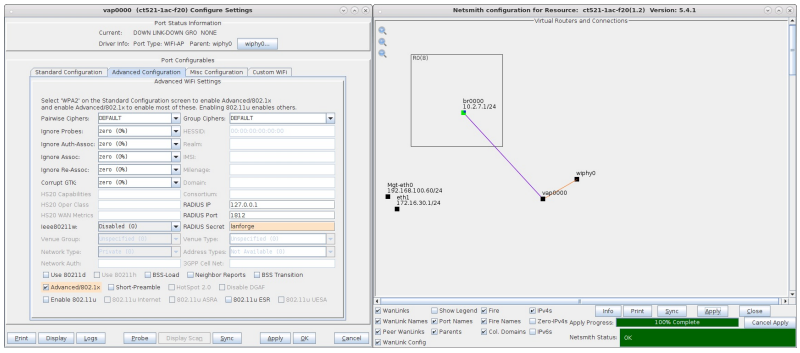
OKC Scenarios

OKC On VAP Only

STA roam result: Full RADIUS authentication plus 4-way handshake.

Using OKC on a VAP requires setting up a custom configuration file in LANforge to utilize the Multiple BSSID feature. In this scenario, the STA is not configured to use OKC and must do a full RADIUS authentication plus 4-way handshake when roaming to the next BSSID.

1. Setup a VAP using RADIUS and EAP-TTLS with a bridge in a virtual router.
   See Setting up a RADIUS Server for help with this step.

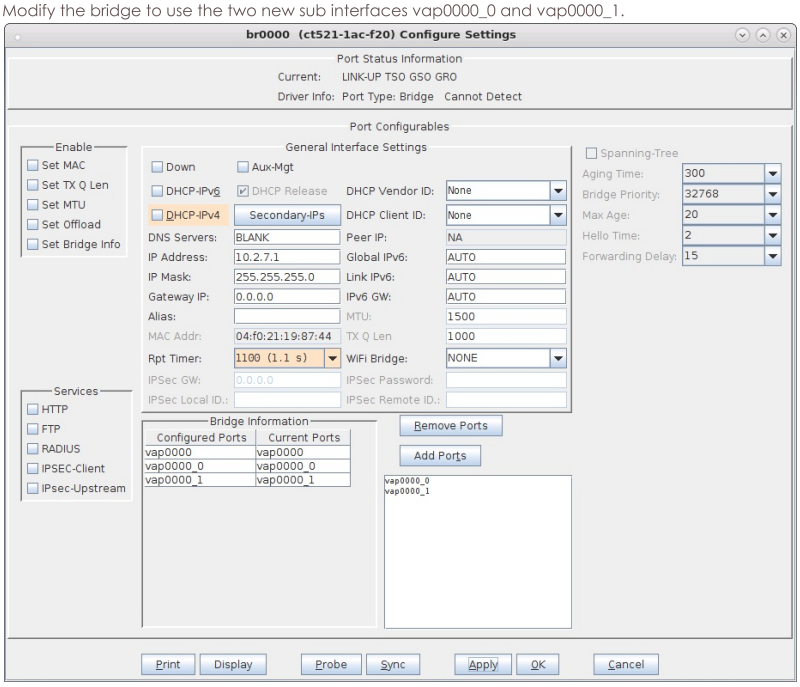


2. Add the following to the custom config section which will create two BSSIDs on the same hostapd process which is required for OKC to work on hostapd:

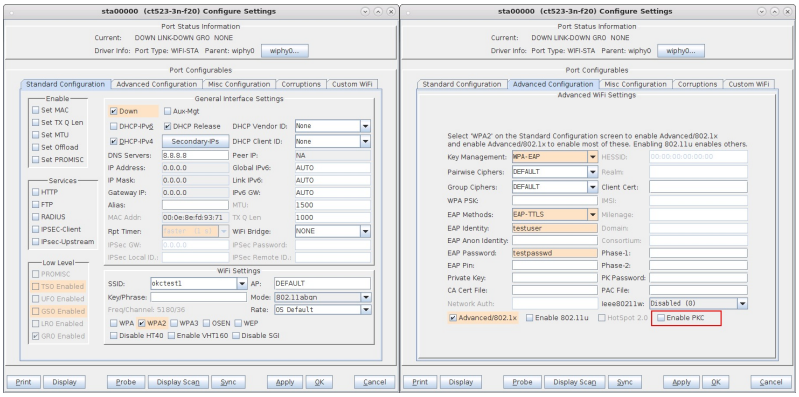
```
bss=vap0000_0
ssid=okctest1
bssid=04:f0:21:19:88:44
ieee8021x=1
own_ip_addr=127.0.0.1
auth_server_addr=127.0.0.1
auth_server_port=1812
```

```
auth_server_shared_secret=lanforge
wpa=2
wpa_pairwise=TKIP CCMP
rsn_pairwise=CCMP
wpa_key_mgmt=WPA-EAP WPA-EAP-SHA256
bss_load_update_period=100
chan_util_avg_period=600
rrm_neighbor_report=1
rrm_beacon_report=1
bss_transition=1
okc=1

bss=vap0000_1
ssid=okctest1
bssid=04:f0:21:19:89:44
ieee8021x=1
own_ip_addr=127.0.0.1
auth_server_addr=127.0.0.1
auth_server_port=1812
auth_server_shared_secret=lanforge
wpa=2
wpa_pairwise=TKIP CCMP
rsn_pairwise=CCMP
wpa_key_mgmt=WPA-EAP WPA-EAP-SHA256
bss_load_update_period=100
chan_util_avg_period=600
rrm_neighbor_report=1
rrm_beacon_report=1
bss_transition=1
okc=1
```

3. Reset the VAP to use the new configuration.
4. Modify the bridge to use the two new sub interfaces vap0000_0 and vap0000_1.



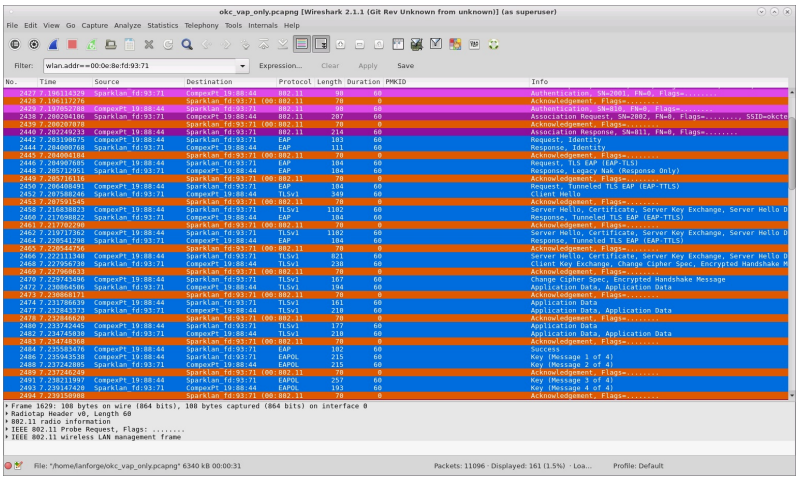
5. Modify a STA so that it is configured to connect to the SSID with 802.1X authentication for EAP-TTLS and with PKC disabled.



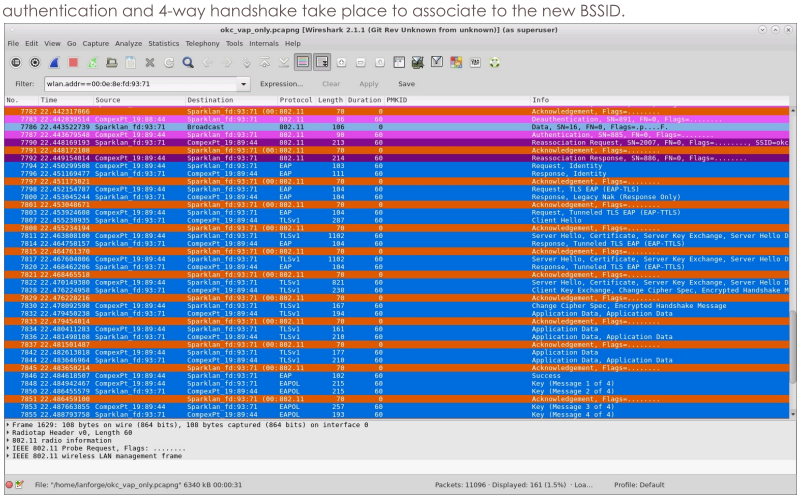
6. Start a packet capture then admin the STA up.
7. Use `wpa_cli` to force the STA to roam with the following terminal commands:

```
# cd /home/lanforge
# . lanforge.profile
# wpa_cli -i sta00000 scan
# wpa_cli -i sta00000 roam <next BSSID>
```

8. In the packet capture, the initial RADIUS authentication and 4-way handshake are shown:



9. Then the STA sends a Reassociation Request which is missing the PMKID and another full RADIUS authentication and 4-way handshake take place to associate to the new BSSID.

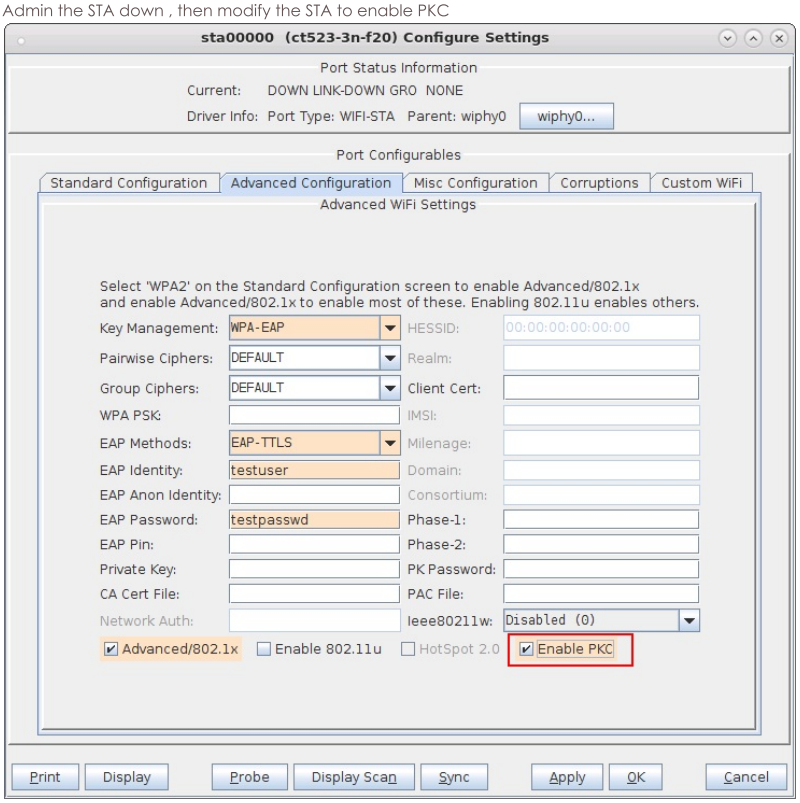


OKC On Both VAP and STA

STA roam result: PMKID is sent, then only 4-way handshake is required.

When both VAP and STA are using OKC, the STA sends its calculated PMKID in the Reassociation Request to the target AP which means the full RADIUS is not needed and only a 4-way handshake is sufficient to connect to the new VAP.

1. Admin the STA down , then modify the STA to enable PKC

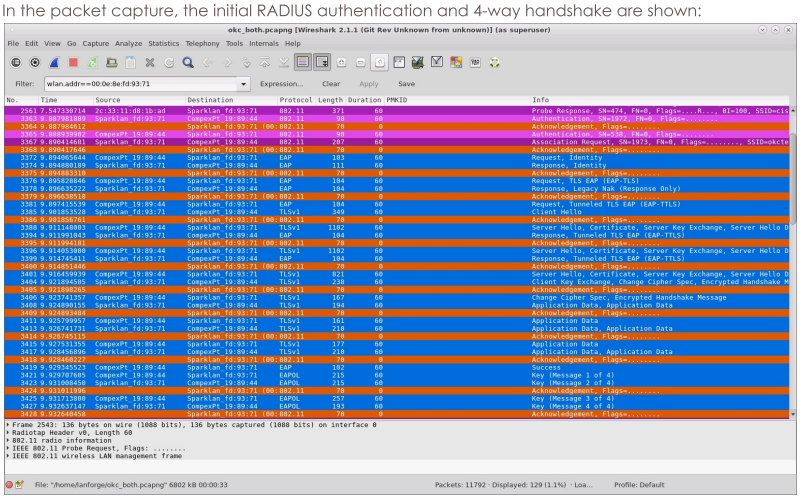


2. Start a packet capture then admin the STA up.

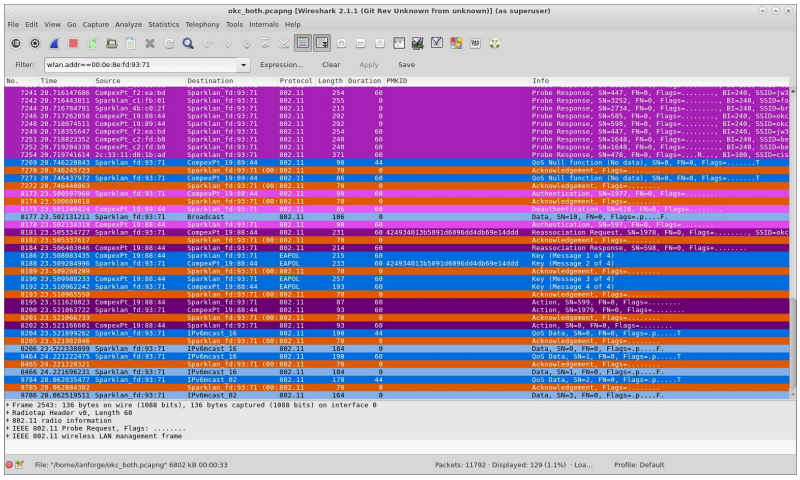
3. Use `wpa_cli` to force the STA to roam with the following terminal commands:

```
# wpa_cli -i sta00000 scan
# wpa_cli -i sta00000 roam <next BSSID>
```

4. In the packet capture, the initial RADIUS authentication and 4-way handshake are shown:



5. Then the STA sends a Reassociation Request which includes its PMKID and only the 4-way handshake is required to associate to the new BSSID.



OKC Disabled On VAP and STA

STA roam result: Full RADIUS authentication plus 4-way handshake.

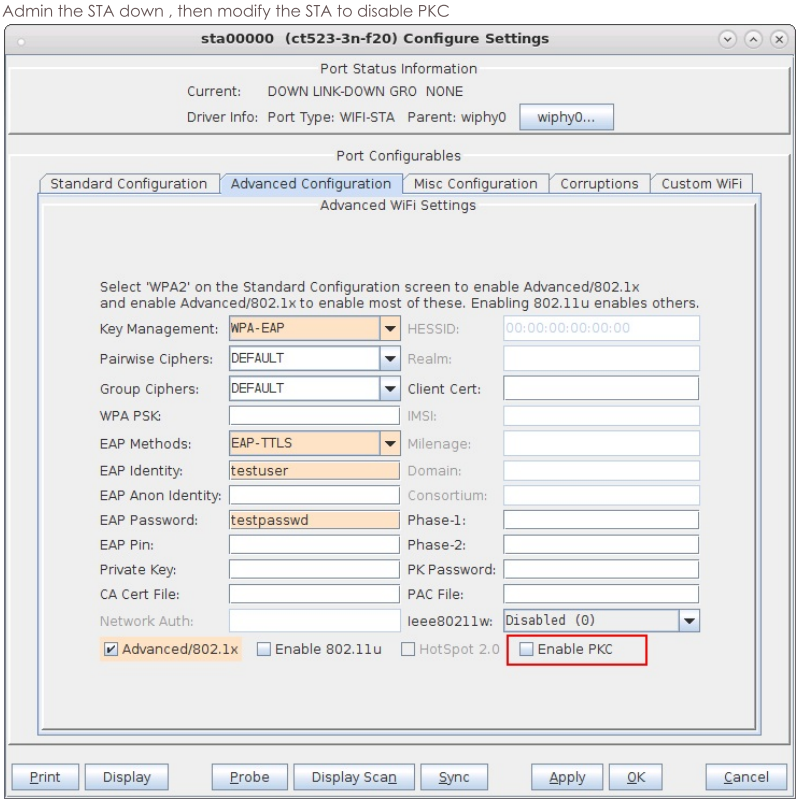
Because neither is using OKC, a full RADIUS authentication plus 4-way handshake is required when the STA roams to the new VAP.

1. To disable OKC on the VAP, comment out the okc=1 lines in the VAP custom configuration, then reset the VAP.

```
bss=vap0000_0
ssid=okctest1
bssid=04:f0:21:19:88:44
ieee8021x=1
own_ip_addr=127.0.0.1
auth_server_addr=127.0.0.1
auth_server_port=1812
auth_server_shared_secret=lanforge
wpa=2
wpa_pairwise=TKIP CCMP
rsn_pairwise=CCMP
wpa_key_mgmt=WPA-EAP WPA-EAP-SHA256
bss_load_update_period=100
chan_util_avg_period=600
rrm_neighbor_report=1
rrm_beacon_report=1
bss_transition=1
#okc=1

bss=vap0000_1
ssid=okctest1
bssid=04:f0:21:19:89:44
ieee8021x=1
own_ip_addr=127.0.0.1
auth_server_addr=127.0.0.1
auth_server_port=1812
auth_server_shared_secret=lanforge
wpa=2
wpa_pairwise=TKIP CCMP
rsn_pairwise=CCMP
wpa_key_mgmt=WPA-EAP WPA-EAP-SHA256
bss_load_update_period=100
```

```
chan_util_avg_period=600
rrm_neighbor_report=1
rrm_beacon_report=1
bss_transition=1
#okc=1
```

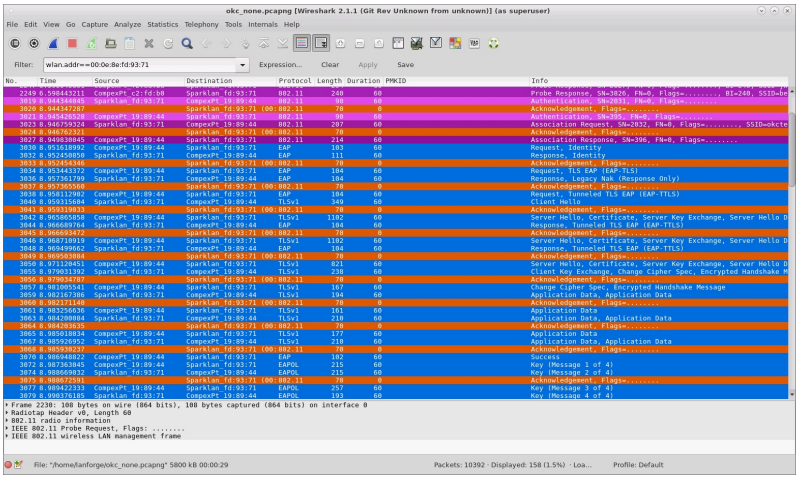
2. Admin the STA down , then modify the STA to disable PKC



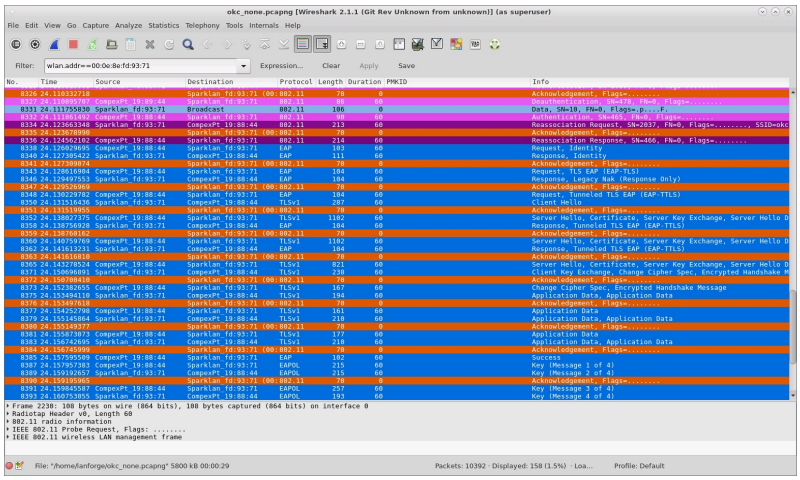
3. Start a packet capture then admin the STA up.
4. Use `wpa_cli` to force the STA to roam with the following terminal commands:

```
# wpa_cli -i sta00000 scan
# wpa_cli -i sta00000 roam <next BSSID>
```

5. In the packet capture, the initial RADIUS authentication and 4-way handshake are shown:



6. Then the STA sends a Reassociation Request which is missing the PMKID and another full RADIUS authentication and 4-way handshake take place to associate to the new BSSID.

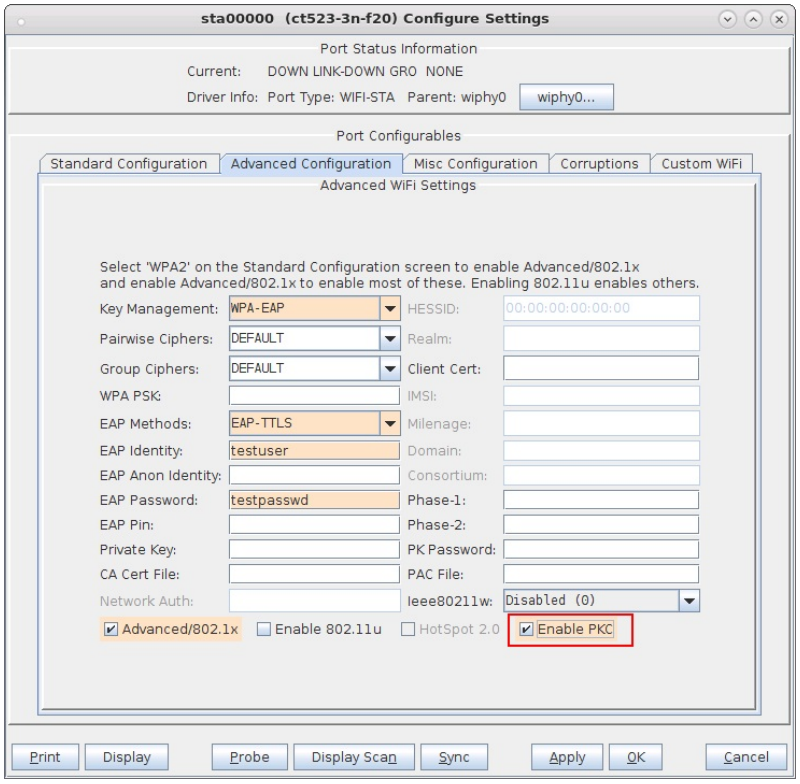


OKC Disabled On VAP and Enabled On STA

STA roam result: PMKID is sent, then full RADIUS authentication plus 4-way handshake.

If just the STA is using OKC, it will send its calculated PMKID in a Reassociation Request to the target AP, but the AP ignores it and the STA must perform a full RADIUS authentication plus 4-way handshake.

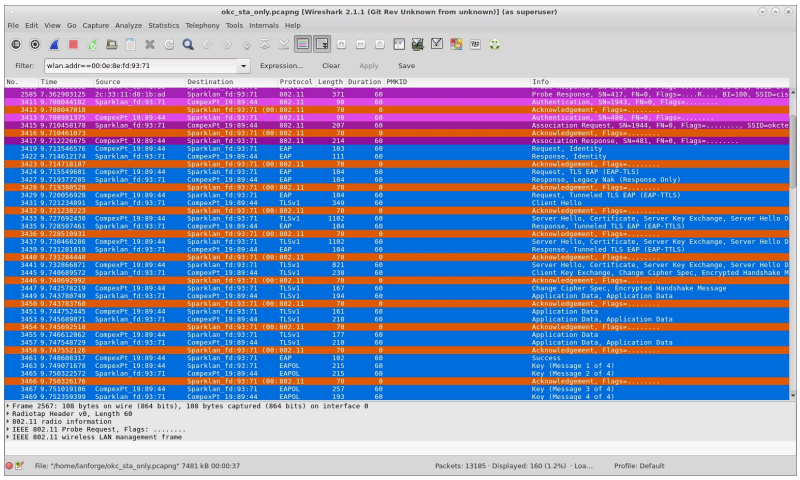
1. Admin the STA down , then modify the STA to enable PKC



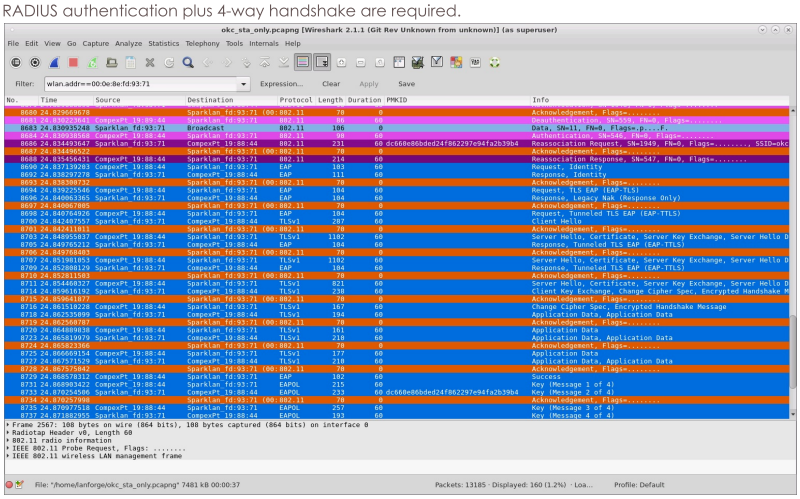
2. Start a packet capture then admin the STA up.
3. Use `wpa_cli` to force the STA to roam with the following terminal commands:

```
# wpa_cli -i sta00000 scan
# wpa_cli -i sta00000 roam <next BSSID>
```

4. In the packet capture, the initial RADIUS authentication and 4-way handshake are shown:



5. Then the STA sends a Reassociation Request which includes its PMKID but the VAP ignores it and a full RADIUS authentication plus 4-way handshake are required.

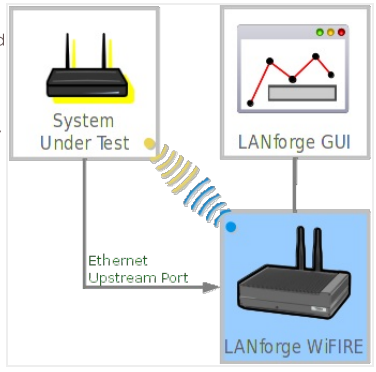


Using iperf3 to Generate Traffic

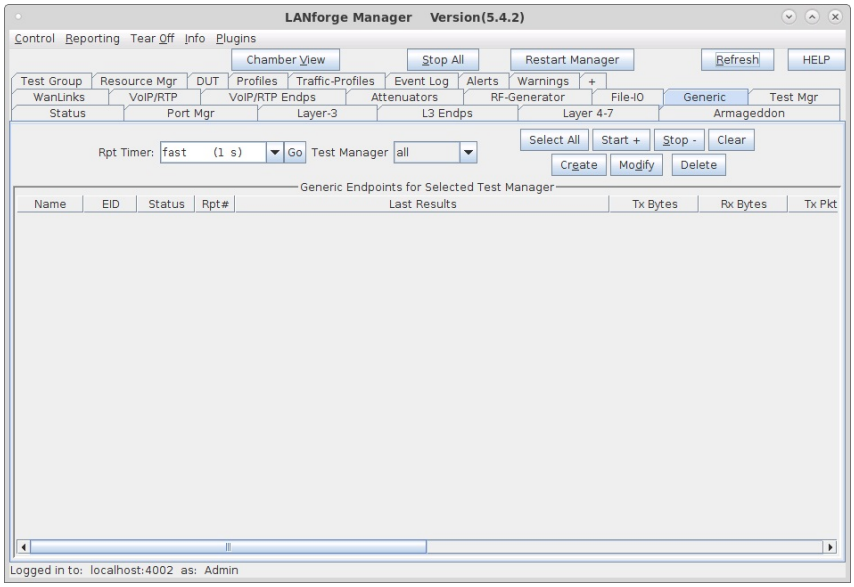
Goal: Set up virtual stations using a LANforge system, connect them to an AP under test, set up iperf3, and run tests.

In this test scenario a LANforge system is used to create both the wireless stations and iperf3 server. The test is then configured to use iperf3 to generate traffic in both the download and upload directions.

Although LANforge Generic endpoints prompt the use of iperf3, iperf2 could be used instead with a manual configuration. This example however, focuses on how to use iperf3.

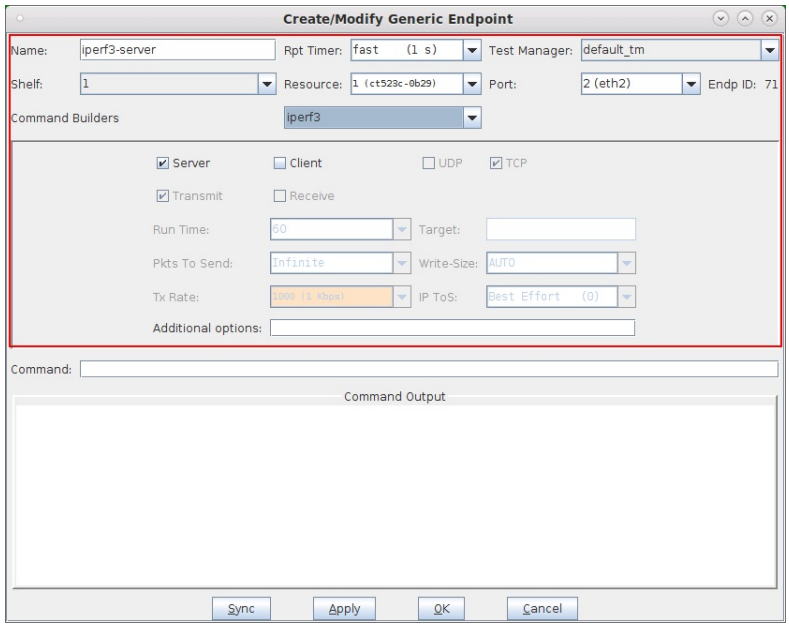


1. **Select the Generic tab from the main GUI window.**

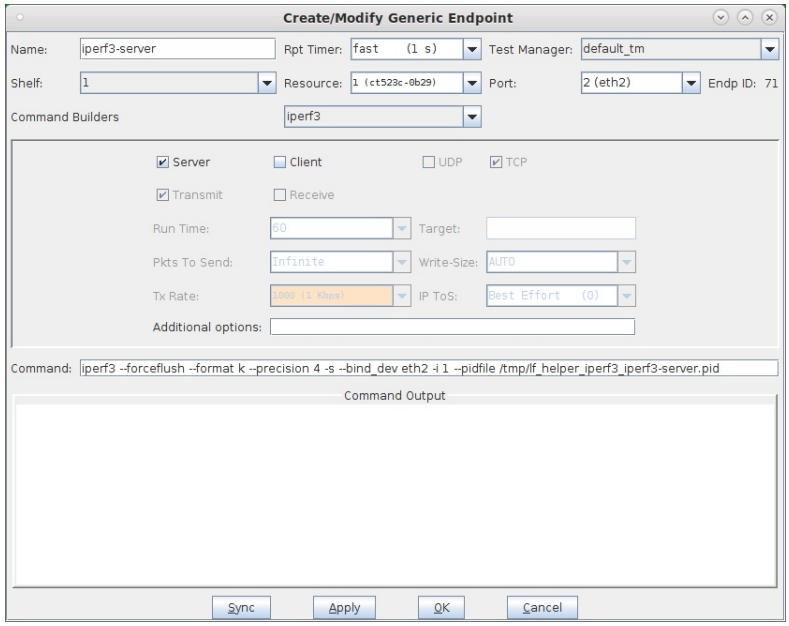


2. **Select the Create button to create the iperf3 server.**

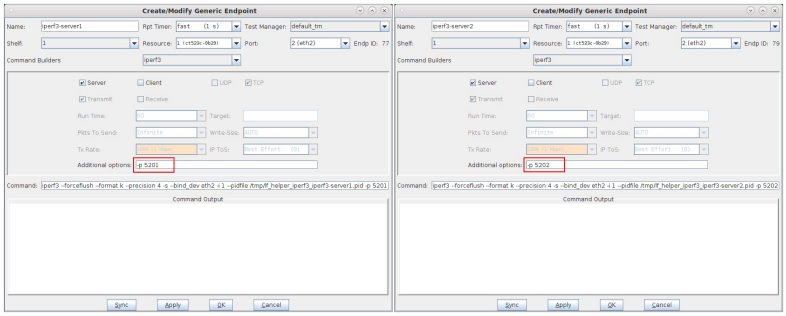
   A. Use the top section to name the connection and choose the port/interface, then use the Command Builder for iperf3 to choose the Server option. The command line will be filled in when the Apply button is selected.



   B. For a single iperf3 test at a time setup one server.

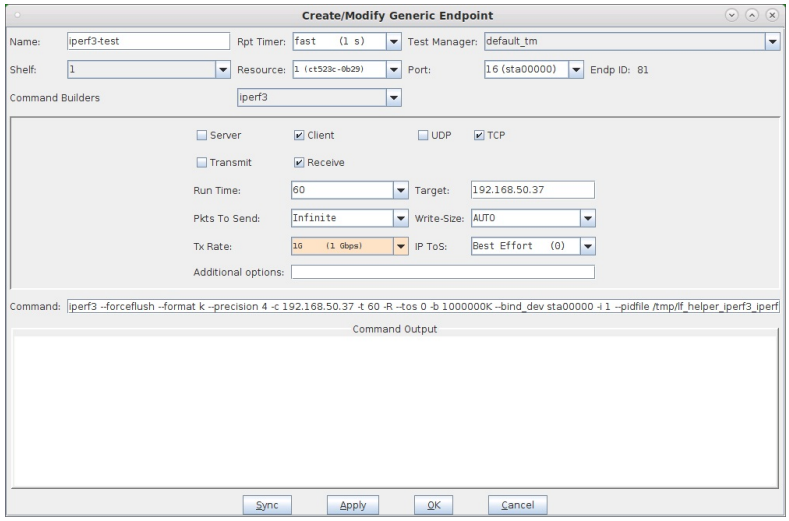


C. For multiple iperf3 tests at a time, setup multiple servers, each with a unique IP port number by using the -p option.

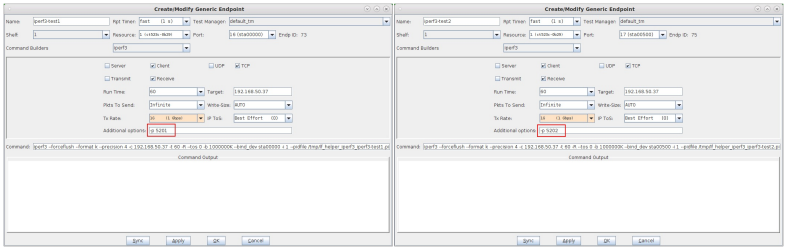


3. **Create the iperf3 client.**

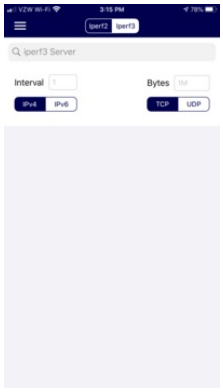
A. For a single iperf3 test at a time setup one client.



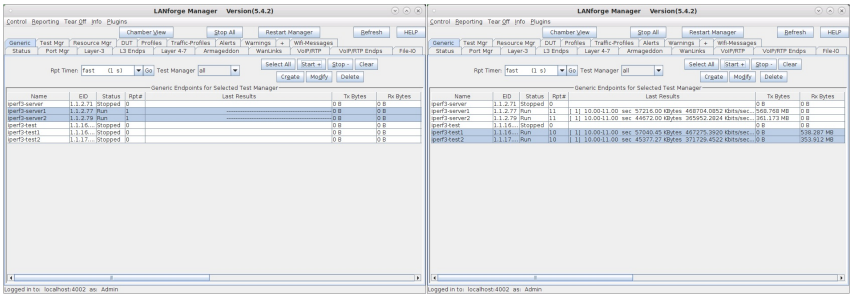
B. For multiple iperf3 tests at a time, setup multiple clients using the corresponding IP ports as the servers.



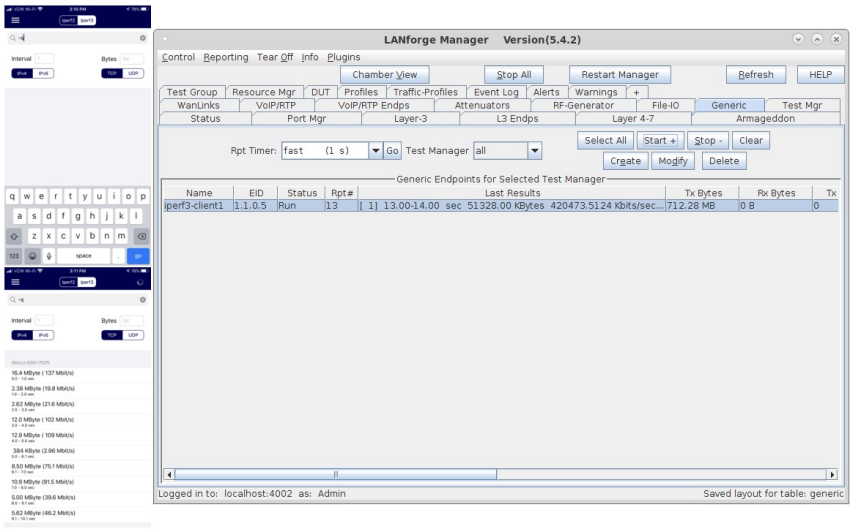
4. **If you want to run iperf3 on a mobile device, run it in server mode.**
One free and simple mobile application for iperf3 is provided by he.net



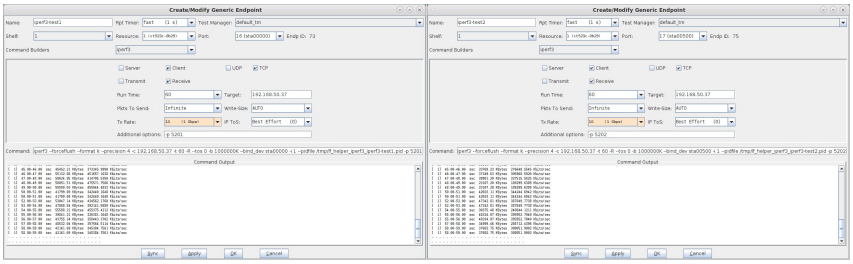
5. **To start any iperf3 test, start the server or servers first, then start the clients.**



6. **For mobile devices, start the iperf3 server on the mobile devices first, then start the clients.**



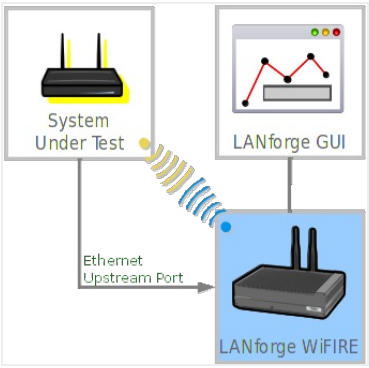
7. **Modify the endpoint to view the per iteration text results.**



Using Flent to Generate Traffic

Goal: Set up virtual stations using a LANforge system, connect them to an AP under test, set up Flent, and run tests.

In this test scenario a LANforge system is used to create both the wireless stations and Flent test setup. The tests can then be configured to use Flent to generate upload or download traffic.



FLENT - FLExible **N**etwork **T**ester is a set of python scripts that use various open-source tools to generate traffic and collect test data. Flent is developed and maintained by Toke Høiland-Jørgensen (@toke_dk)

One advantage to using Flent on LANforge is that a single LANforge box can send traffic to itself which eliminates the need for multiple systems to run client/server test traffic against a DUT.

1. **Install packages required to run Flent on Fedora-30 (also works on Fedora-36).**
   Additional packages may need to be installed. See flent.org for more info.

A. Install automake and texinfo

```
sudo dnf install automake texinfo autoconf autotools-dev kernel-devel
```

B. Flent

```
git clone https://github.com/tohojo/flent.git
cd flent
git checkout v1.3.2
sudo python3 setup.py install
```

C. Flent GUI - **only necessary to view flent output graphs**

```
sudo dnf install python3-matplotlib python3-matplotlib-qt5
sudo dnf install python3-qt5 python3-PyQt5 python3-QtPy
```

D. Netperf/Netserver

```
git clone https://github.com/HewlettPackard/netperf.git
cd netperf
sudo ./autogen.sh
sudo ./configure --enable-demo=yes
sudo make
sudo make install
```

E. Fping

```
git clone https://github.com/schweikert/fping.com
cd fping
git checkout 3.6
sudo ./autogen.sh
sudo ./configure
sudo make
sudo make install
```

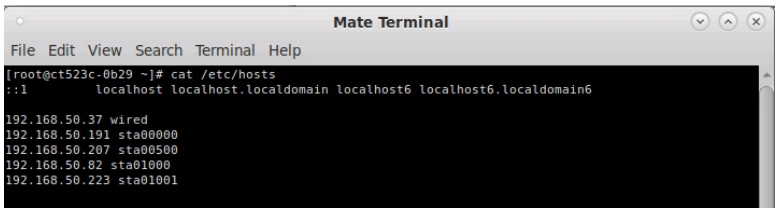
F. Iperf2 - **Required for the --enhanced option which will display enhanced output in reports.**

```
dnf install iperf
```

2. **To view a list of possible Flent tests, or run Flent by command line, open a terminal window.**

```
[lanforge@ct523c-0b29 ~]$ flent --list-tests
Available tests:
bursts                      :  Latency measurements under intermittent UDP bu
bursts_11e                  :  802.11e Latency measurements under intermitten
cisco_5tcpup                :  RTT Fair Realtime Response Under Load
cisco_5tcpup_2udpflood      :  Cisco 5TCP up + 2 6Mbit UDP
cubic_bbr                   :  Cubic VS BBR smackdown
cubic_cdg                   :  Cubic VS CDG smackdown
cubic_dctcp                 :  Cubic VS DCTCP smackdown
cubic_ledbat                :  Cubic VS Ledbat smackdown
cubic_ledbat_1              :  Cubic vs LEDBAT upload streams w/ping
cubic_reno                  :  Cubic VS Reno smackdown
cubic_westwood              :  Cubic VS Westwood
dslreports_8dn              :  8 down - dslreports dsl test equivalent
http                        :  HTTP latency test
http-1down                  :  HTTP get latency with competing TCP download s
http-1up                    :  HTTP get latency with competing TCP upload str
http-rrul                   :  HTTP get latency with competing RRUL test
iterated_bidirectional      :  Iterated TCP bidirectional transfers example
ledbat_cubic_1              :  Cubic vs LEDBAT upload streams w/ping
```

3. **Enter interface names to be used in Flent tests into the /etc/hosts file.**
   This is necessary for some of the Flent tools that can only use an interface name rather than IP address.



4. **To run Flent tests by command line:**

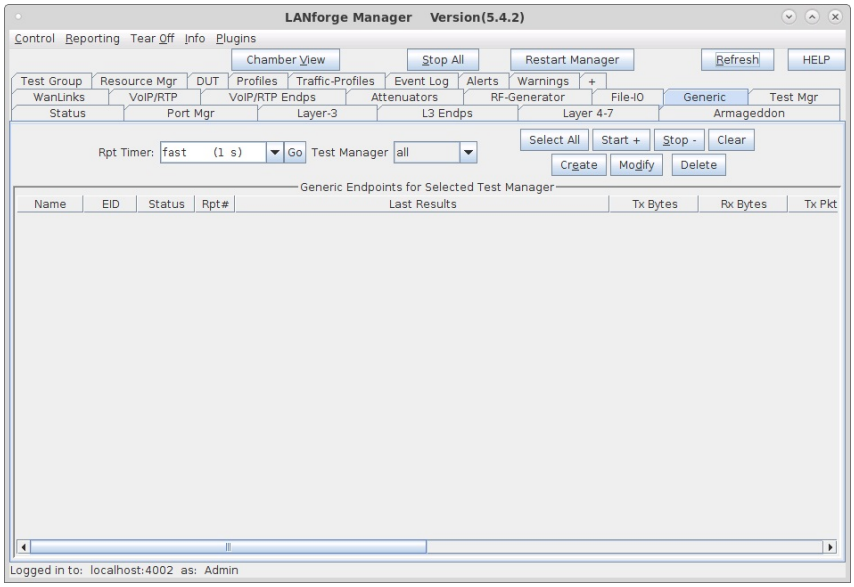
   A. Run netserver. This will start the netserver program which is the traffic server for the other Flent tools specifically netperf.

```
[lanforge@ct523c-0b29 ~]$ netserver
Starting netserver with host 'IN(6)ADDR_ANY' port '12865' and family AF_UNSPEC
```

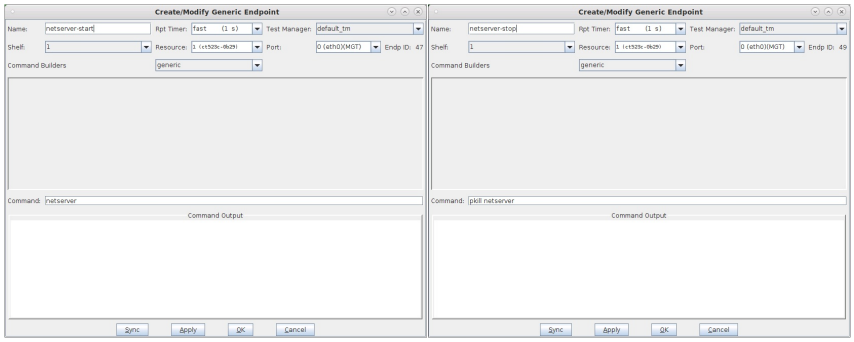
   B. Run one of the available tests. Because LANforge is using vrf, use the **vrf_exec.bash** script to specify which interface to run the test. The direction of the test is from the perspective of the wired interface. Use the **--swap-up-down** option to change the direction to the host interface which is the wireless station sta00000 in this example.

```
[lanforge@ct523c-0b29 ~]$ sudo ./vrf_exec.bash eth2 flent -x -H sta00000 tcp_download --swap-up-down
[sudo] password for lanforge:
Started Flent 1.9.9-git-d06e0b6 using Python 3.7.6.
Starting tcp_download test. Expected run time: 70 seconds.
Data file written to ./tcp_download-2020-04-15T113910.591355.flent.gz

Summary of tcp_download test run from 2020-04-15 18:39:10.591355

                          avg       median          # data pts
 Ping (ms) ICMP :        23.20        23.10 ms            350
 TCP download   :       671.49       672.59 Mbits/s       350
[lanforge@ct523c-0b29 ~]$
```

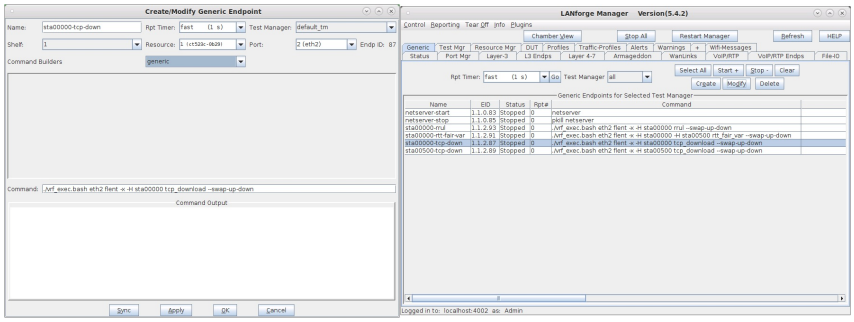
5. **To run Flent tests from the LANforge GUI, select the Generic tab from the main GUI window.**



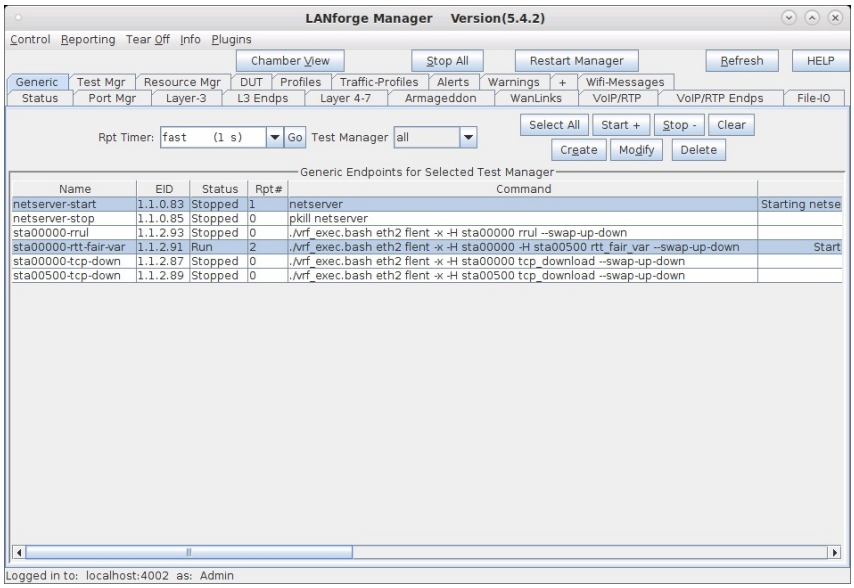
6. **Create the Flent netserver.**
   Here it is useful to create one Generic endpoint to start netserver and one to stop it because the netserver program runs in the background once it is started.



7. **Create the Flent test or tests.**

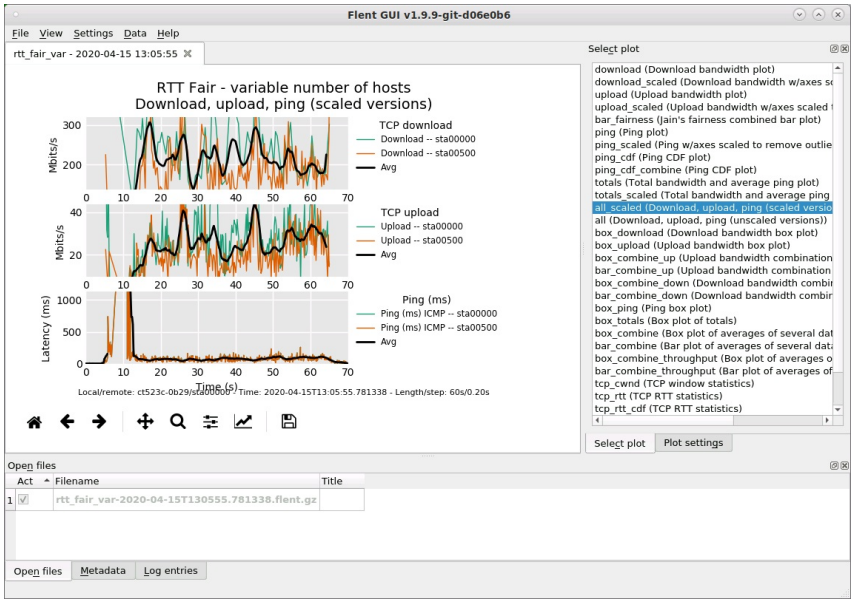


8. **To start any Flent test, start the netserver first, then start the tests one at a time.**



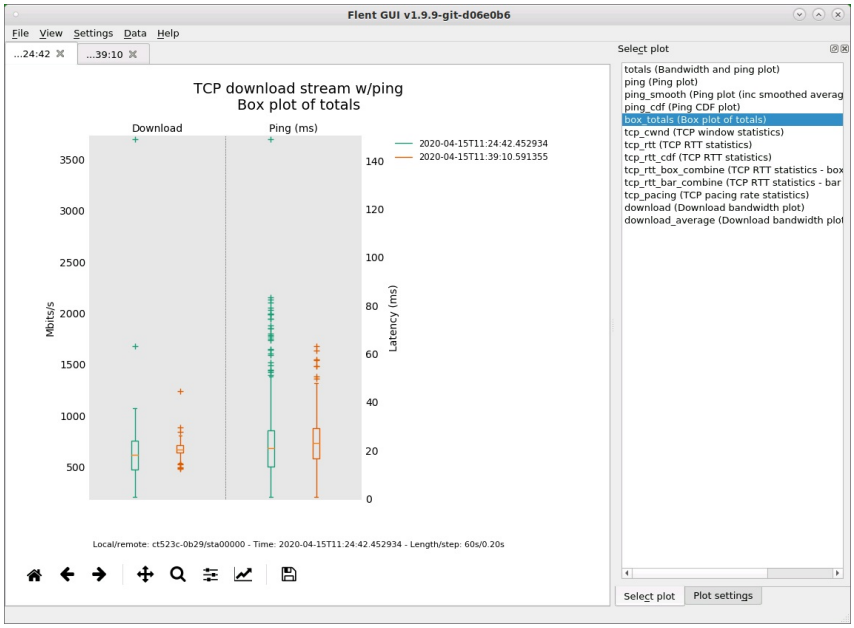
9. **Use the Flent GUI to view Flent test results.**

```
[lanforge@ct523c-0b29 ~]$ flent-gui rtt_fair_var-2020-04-15T130555.781338.flent.gz
```



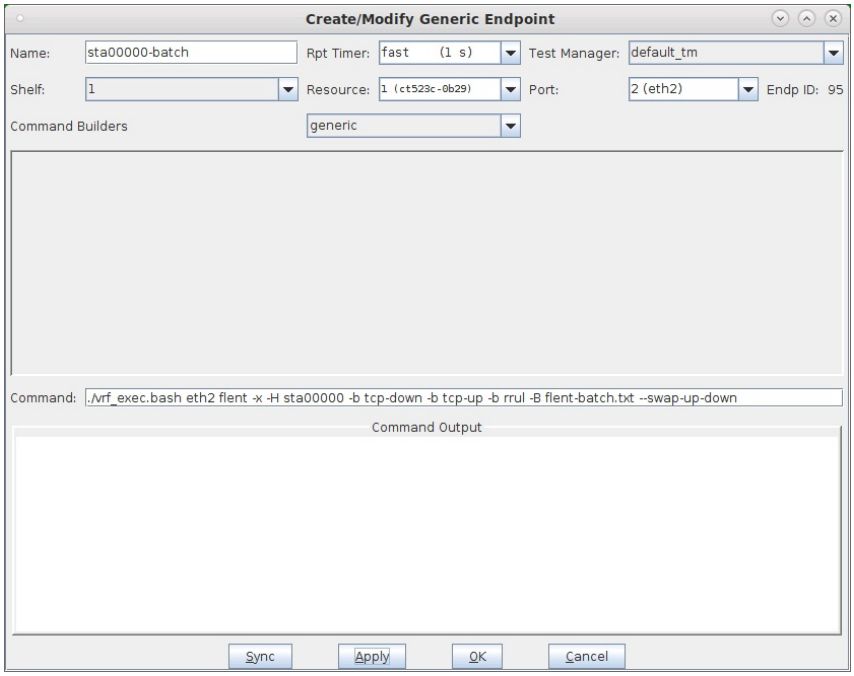
10. **Use the Flent GUI to compare Flent test results.**

```
[lanforge@ct523c-0b29 ~]$ flent-gui tcp_download-2020-04-15T112442.452934.flent.gz t
```

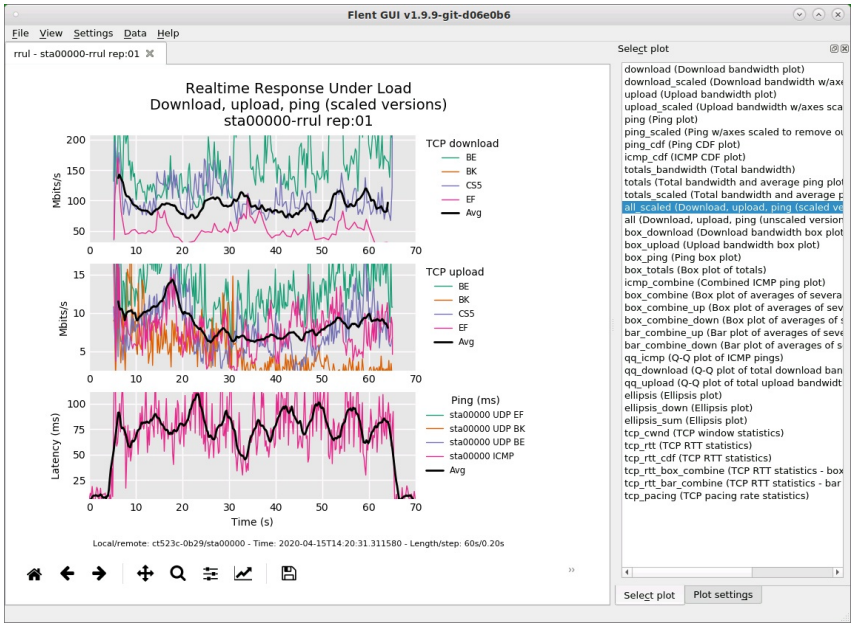


11. **Flent batch files can be setup to run consecutive and repeated tests.**

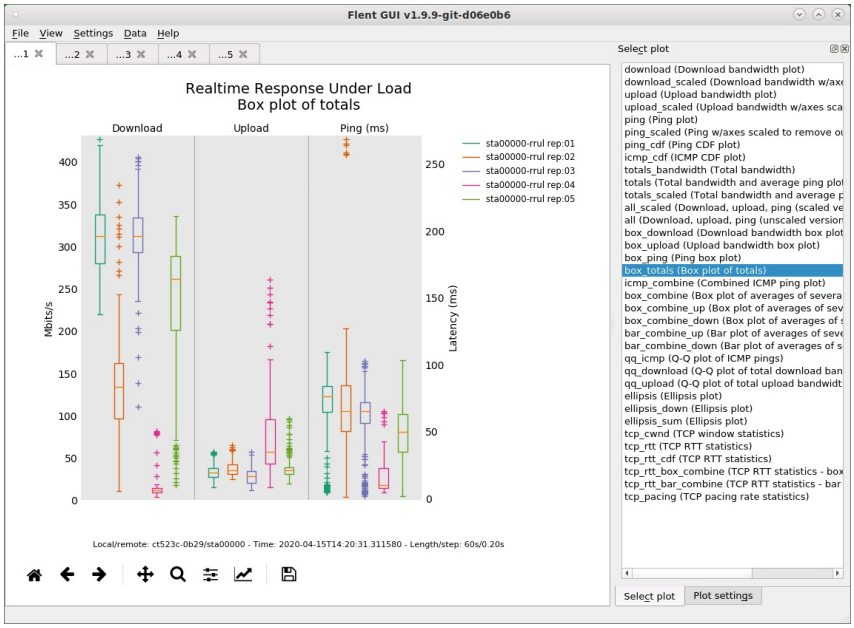
```
[lanforge@ct523c-0b29 ~]$ cat flent-batch.txt

[Batch::tcp-down]
test_name = tcp_download
hosts = sta00000
title = sta00000-tcp-down rep:${repetition}
filename_extra = sta00000-tcp-down-${repetition}
repetitions = 5
[Batch::tcp-up]
test_name = tcp_upload
hosts = sta00000
title = sta00000-tcp-up rep:${repetition}
filename_extra = sta00000-tcp-up-${repetition}
repetitions = 5
[Batch::rrul]
test_name = rrul
hosts = sta00000
title = sta00000-rrul rep:${repetition}
filename_extra = sta00000-rrul-${repetition}
repetitions = 5\
```



12. **Flent batch results for rrul.**
    The **rrul** test is the Realtime Response Under Load test which attempts to completely use up the link with four upstream TCP connections and four downstream TCP connections each with a different IP ToS setting plus ping and UDP to measure latency all running simultaneously for the purpose of exposing bufferbloat in the network under test.



13. **Flent batch results for five rrul repetitions.**



Using own_ie_override for Custom RSN Information Elements of Beacon Frames

Goal: Manually override certain RSN-related information elements of Beacon frames sent by a LANforge system in AP Mode for testing WPA2 authentication.

In this test scenario a LANforge system acts as a WiFi access point configured to use WPA2 authentication. Beacon frames sent by this AP contain information elements about RSN encryption. The `own_ie_override` field in `hostapd.conf` provides a built-in way to override certain parts of these IEs for testing purposes, and may be configured through LANforge **Custom WiFi** parameters. Listed below are several example test cases provided in the hostap repository.

---

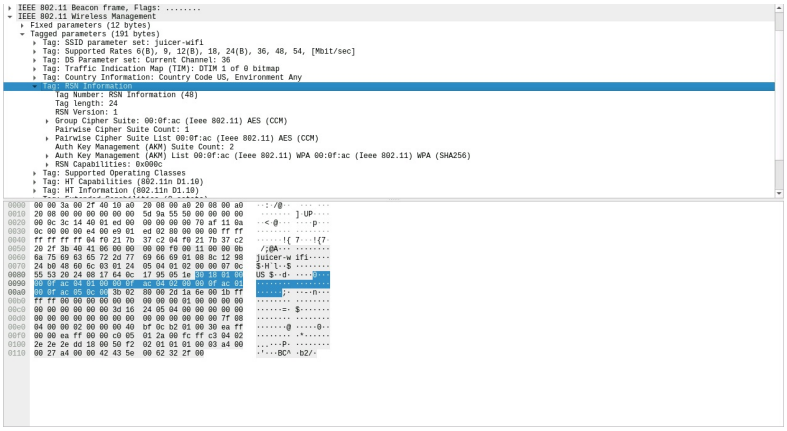
1. **Initial Setup for WPA2-Authentication Testing.**
   A. Set up a virtual AP for testing.
   In this test, it is named `vap0000` on parent device `wiphy0`.
   For more information see Create vAP in Bridge Mode
   B. On a separate radio, create a station to authenticate with `vap0000`:
   In the **Port Manager** tab, select `wiphy1` and click **Create**; select **WiFi STA**, then click **Apply**.
   In this test, the station is named `wlan1` on parent device `wiphy1`.
   For more information see Generating Traffic for WLAN Testing
   C. Configure `vap0000` and `wlan1` to use WPA2-PSK encrypted authentication.
   For more information see WPA2 Authentication Test Scenario
   D. Configure `vap0000` and `wlan1` with **SSID** `test-wpa2-psk` and **Keyphrase** `qwertyuiop`.
   E. Create a Monitor Port on its own radio to sniff wireless packets.
   In this test, the monitor port is named `moni3a`.
   For more information see Using Wireshark to Sniff WiFi Monitors

2. **Control (No Change):**
   A. Configure **Custom WiFi** in `vap0000`:
   Select `vap0000` and click **Modify**.
   Navigate to the **Custom WiFi** tab.
   Ensure that no `own_ie_override` parameter is set in **User-Specified supplicant/hostapd configuration text**.
   Click **Apply** then **OK**.
   B. Set the vAP down and back up to allow changes to take effect:
   In the **Port Manager** tab, select `vap0000`.
   Admin all selected interfaces **DOWN** (CTRL-PLUS).
   Admin all selected interfaces **UP** (CTRL-MINUS).
   C. Sniff packets to observe the authentication behavior:
   On the observation system in the **Port Manager** tab, select only `moni3a`:
   Click **Sniff Packets**.

D. Reset the station to force re-authentication:
In the Port Manager tab, select only `wlan1`.
Click **Reset Port**.

E. Observe the results, which should be similar to the following:
- Packets are not malformed.
- The station `wlan1` succeeds in authenticating with `vap0000`.
- No RSN Information Element is found in Beacon frames sent by `vap0000`.

F. Example results:



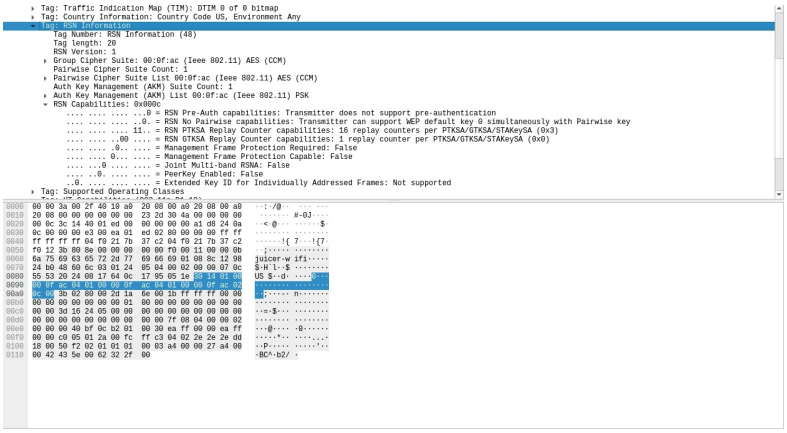
3. **The RSN element used normally by hostapd:**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`own_ie_override=30140100000fac040100000fac040100000fac020c0`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000`.
- RSN Information Element is present in Beacon frames sent by `vap0000`.

D. Example results:



4. **No RSN Capabilities field:**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`own_ie_override=30120100000fac040100000fac040100000fac02`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000`.
- Beacon frames sent by `vap0000` are recognizably malformed.

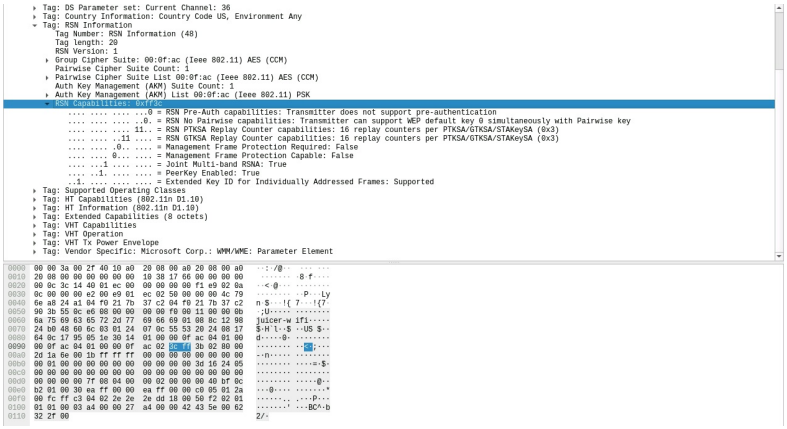
5. **Reserved RSN Capabilities bits set:**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`own_ie_override=30140100000fac040100000fac040100000fac023cff`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000` with `CTRL-MSG: NETWORK NOT FOUND`.
- RSN Information Element is present in Beacon frames sent by `vap0000`.
- Beacon frames sent by `vap0000` are not malformed.

D. Example results:



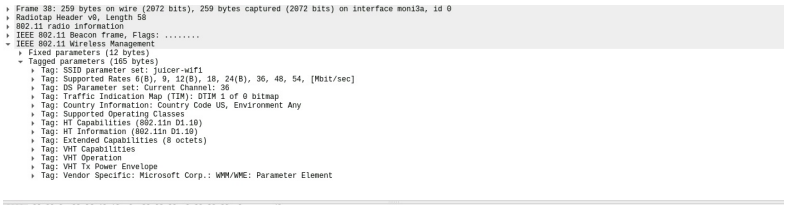
6. **Truncated RSN Capabilities field:**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`own_ie_override=30130100000fac040100000fac040100000fac023c`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000` with `CTRL-MSG: NETWORK NOT FOUND`.
- RSN Information Element is not present in Beacon frames sent by `vap0000`.
- Beacon frames sent by `vap0000` are not malformed.

D. Example results:



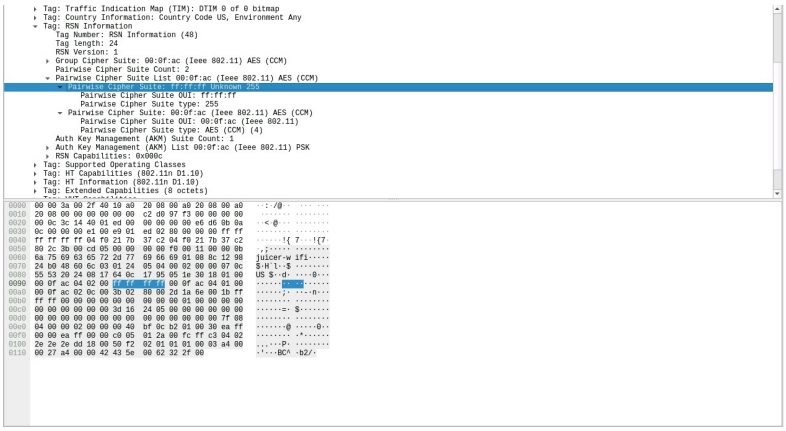
7. **Extra pairwise cipher suite (unsupported):**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`own_ie_override=30180100000fac040200ffffffff000fac040100000fac020c00`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000` with `CTRL-MSG: NETWORK NOT FOUND`.
- RSN Information Element is present in Beacon frames sent by `vap0000`.
- Beacon frames sent by `vap0000` are not malformed.

D. Example results:



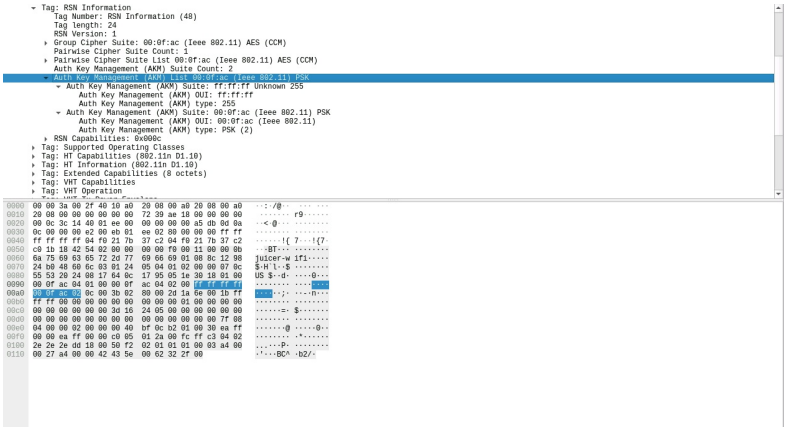
8. **Extra AKM suite (unsupported):**

   A. Configure **Custom WiFi** in `vap0000`:
   Select `vap0000` and click **Modify**.
   Navigate to the **Custom WiFi** tab.
   In the **User-Specified supplicant/hostapd configuration text** field, write:
   `own_ie_override=30180100000fac040100000fac040200ffffffff000fac020c00`.
   Click **Apply** then **OK**.

   B. Reset ports and sniff packets:
   Repeat steps B through D of Step 2.

   C. Observe the results, which should be similar to the following:
   - The station `wlan1` fails to authenticate with `vap0000` with `CTRL-MSG: NETWORK NOT FOUND`.
   - RSN Information Element is present in Beacon frames sent by `vap0000`.
   - The RSN IE in a Beacon frame sent by `vap0000` contains two Pairwise Cipher Suite fields rather than one.
   - Beacon frames sent by `vap0000` are not malformed.

   D. Example results:



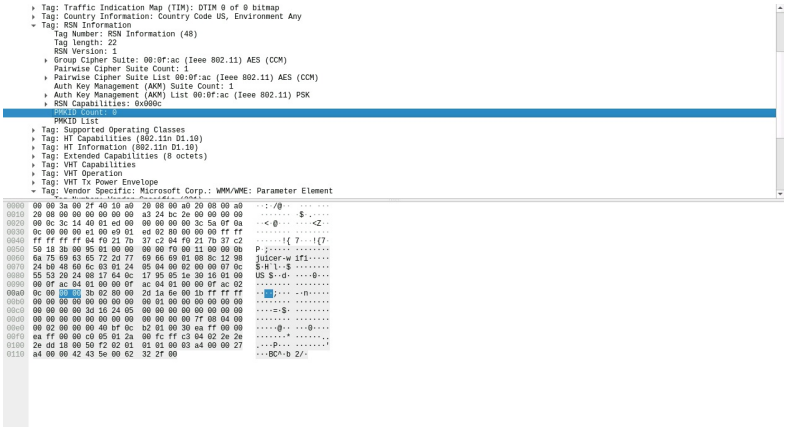
9. **PMKIDCount field included:**

   A. Configure **Custom WiFi** in `vap0000`:
   Select `vap0000` and click **Modify**.
   Navigate to the **Custom WiFi** tab.
   In the **User-Specified supplicant/hostapd configuration text** field, write:
   `own_ie_override=30160100000fac040100000fac040100000fac020c000000`.
   Click **Apply** then **OK**.

   B. Reset ports and sniff packets:
   Repeat steps B through D of Step 2.

   C. Observe the results, which should be similar to the following:
   - The station `wlan1` fails to authenticate with `vap0000` with `CTRL-MSG: NETWORK NOT FOUND`.
   - RSN Information Element is present in Beacon frames sent by `vap0000`.
   - The RSN IE in a Beacon frame sent by `vap0000` contains a PMKIDCount field.
   - Beacon frames sent by `vap0000` are not malformed.

D. Example results:



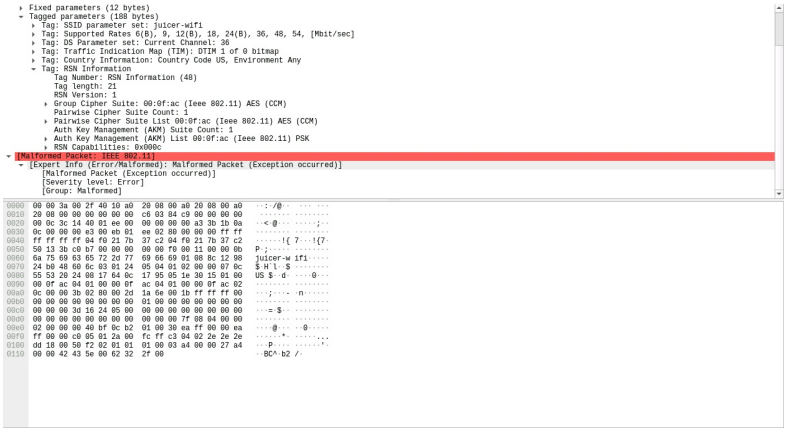
10. **Truncated PMKIDCount field:**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`own_ie_override=30150100000fac040100000fac040100000fac020c0000`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000` with `CTRL-MSG: NETWORK NOT FOUND`.
- The RSN IE in a Beacon frame sent by `vap0000` is present, but incomplete.
- Beacon frames sent by `vap0000` are recognizably malformed.

D. Example results:



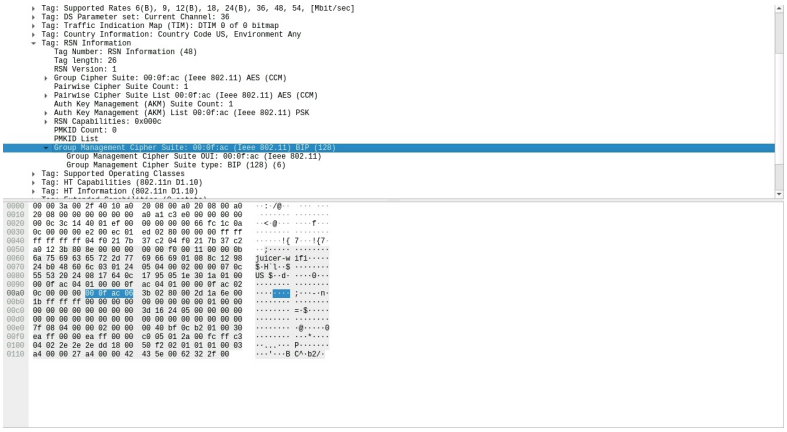
11. **Unexpected Group Management Cipher Suite with PMF disabled:**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`own_ie_override=301a0100000fac040100000fac040100000fac020c000000000fac06`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000` with `CTRL-MSG: NETWORK NOT FOUND`.
- The RSN IE in a Beacon frame sent by `vap0000` contains a Group Management Cipher field.
- Beacon frames sent by `vap0000` are not malformed.

D. Example results:



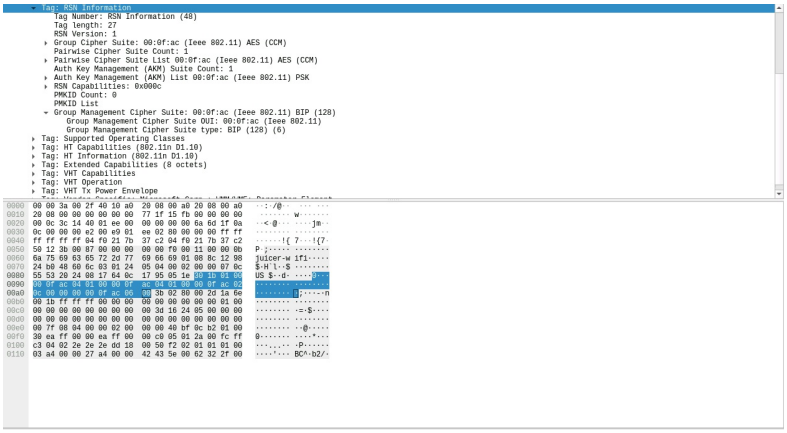
12. **Extra octet after defined fields (future extensibility):**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`own_ie_override=301b0100000fac040100000fac040100000fac020c000000000fac0600`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000` with `CTRL-MSG: NETWORK NOT FOUND`.
- The RSN IE in a Beacon frame sent by `vap0000` contains a Group Management Cipher field.
- Beacon frames sent by `vap0000` are not malformed.

D. Example results:



Corrupting EAPOL-Key 3/4 Handshake Message RSNE WPA-Key Information Elements

Goal: Manually override RSN-related information elements of 3 of 4-way EAPOL authentication handshake messages sent by a LANforge system in AP Mode for testing purposes.

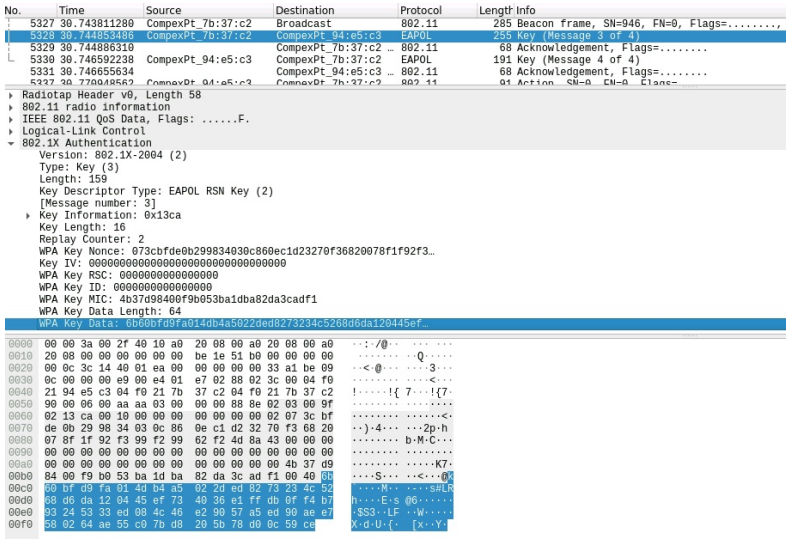
In this test scenario a LANforge system acts as a WiFi access point configured to use WPA2 authentication. The `rsne_override_eapol` field in LANforge Custom WiFi Parameters provides means to corrupt or customize certain information elements (IEs) in the third of four messages comprising EAPOL authentication "handshake". These IEs contain information about RSN encryption, including Pre-Shared Key (PSK) information required by 802.11 protocol for authentication success. Thus, such IEs may be overriden for the purpose of testing behavior under faulty authentication. Listed below is an example test case, of which documentation may be found in the hostap repository.

---

1. **Initial Setup for WPA2-Authentication Testing.**
   A. Set up a virtual AP for testing.
   In this test, it is named `vap0000` on parent device `wiphy0`.
   For more information see Create VAP in Bridge Mode
   B. On a separate radio, create a station to authenticate with `vap0000`:
   In the **Port Manager** tab, select `wiphy1` and click **Create**; select **WiFi STA**, then click **Apply**.
   In this test, the station is named `wlan1` on parent device `wiphy1`.
   For more information see Generating Traffic for WLAN Testing
   C. Configure `vap0000` and `wlan1` to use WPA2-PSK encrypted authentication.
   For more information see WPA2-Authentication Test Scenario
   D. Configure `vap0000` and `wlan1` with **SSID** `test-wpa2-psk` and **Keyphrase** `qwertyuiop`.
   E. Create a Monitor Port on its own radio to sniff wireless packets.
   In this test, the monitor port is named `moni3a`.
   For more information see Using Wireshark to Sniff WiFi Monitors

2. **Control (No Change):**
   A. Configure **Custom WiFi** in `vap0000`:
   Select `vap0000` and click **Modify**.
   Navigate to the **Custom WiFi** tab.
   Ensure that no `rsne_override_eapol` parameter is set in **User-Specified supplicant/hostapd configuration text**.
   Click **Apply** then **OK**.
   B. Set the vAP down and back up to allow changes to take effect:
   In the **Port Manager** tab, select `vap0000`.
   Admin all selected interfaces **DOWN** (CTRL-PLUS).
   Admin all selected interfaces **UP** (CTRL-MINUS).
   C. Sniff packets to observe the authentication behavior:
   On the observation system in the **Port Manager** tab, select only `moni3a`:
   Click **Sniff Packets**.
   D. Reset the station to force re-authentication:
   In the **Port Manager** tab, select only `wlan1`.
   Click **Reset Port**.
   E. Observe the results, which should be similar to the following:
   - Packets are not malformed.
   - The station `wlan1` succeeds in authenticating with `vap0000`.
   - RSN Information Element is found in `EAPOL-Key Message 3 of 4` sent by `vap0000` with WPA-Key-Data field.

F. Example results:



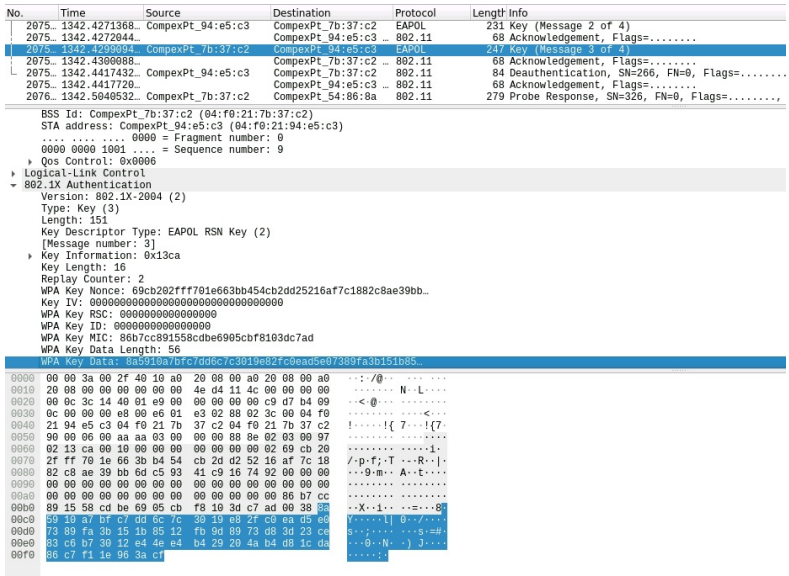
3. **RSNE Mismatch in EAPOL-Key Message 3/4:**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`rsne_override_eapol=30140100000fac040100000fac040100000fac020c80`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000`.
- The WPA-Key-Data field `EAPOL-Key Message 3 of 4` sent by `vap0000` is changed.
- The frame following `EAPOL-Key Message 3 of 4` sent by `vap0000` has type `DEAUTH`.

D. Example results:



Corrupting EAPOL-Key 3/4 Handshake Message RSNXE WPA-Key Information Elements

Goal: Manually override RSN-Extension-Element (RSNXE) of 3 of 4-way EAPOL authentication handshake messages sent by a LANforge system in AP Mode for testing purposes.

The `rsnxe_override_eapol` field in LANforge **Custom WiFi Parameters** provides capability to corrupt specific information elements of **EAPOL-Key Message 3 of 4** of WPA3 authentication sequence using SAE with encrypted management frames (802.11w). The Robust Security Network Extension Element (RSNXE), used to communicate and confirm certain aspects of security negotiation such as "SAE-hash-to-element", must be consistent between Beacon frames and EAPOL-Key messages; this method corrupts this RSNXE IE, provoking authentication failure with a distinctive response message: `WPA: RSNXE mismatch between Beacon/ProbeResp and EAPOL-Key msg 3/4`.

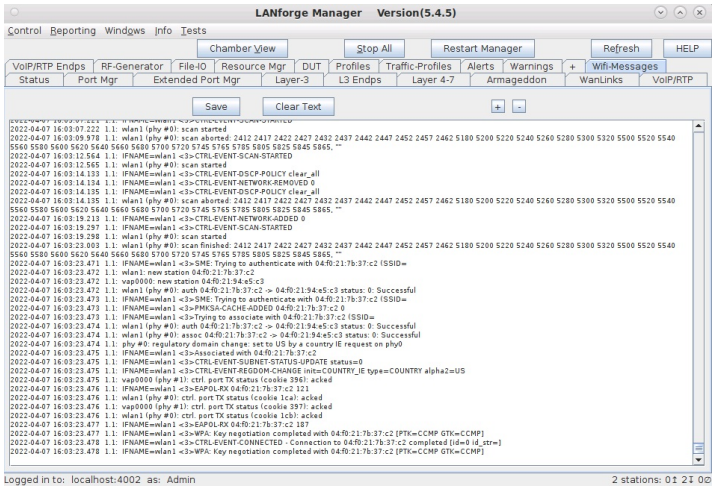
---

1. **Initial Setup for WPA3-Authentication Testing with Simultaneous Authentication of Equals (SAE).**
   The setup requires AP and station NIC drivers capable of enabling SAE encryption (this example uses MediaTEK radios with ath10k(988x) driver), enabling encrypted management frames (`802.11w`), enabling WPA3 and disabling WPA2-PSK authentication in both station and AP.
   A. Set up a virtual AP for testing.
      In this test, it is named `vap0000` on parent device `wiphy0`.
      For more information see Create VAP in Bridge Mode
   B. On a separate radio, create a station to authenticate with `vap0000`:
      In the **Port Manager** tab, select `wiphy1` and click **Create**; select **WiFi STA**, then click **Apply**.
      In this test, the station is named `wlan1` on parent device `wiphy1`.
      For more information see Generating Traffic for WLAN Testing
   C. Configure `vap0000` and `wlan1` to use **WPA3-SAE** encrypted authentication.
      Ensure that `802.11w` is enabled, since it is required for WPA3.
      For more information see Setting up WPA3
   D. Configure `vap0000` and `wlan1` with **SSID** `test-wpa2-psk` and **Keyphrase** `qwertyuiop`.
   E. Create a Monitor Port on its own radio to sniff wireless packets.
      In this test, the monitor port is named `moni3a`.
      For more information see Using Wireshark to Sniff WiFi Monitors

2. **Control (No Change):**
   A. Configure **Custom WiFi** in `vap0000`:
      Select `vap0000` and click **Modify**.
      Navigate to the **Custom WiFi** tab.
      Ensure that no `sae_commit_override` parameter is set in **User-Specified supplicant/hostapd configuration text**.
      Click **Apply** then **OK**.
   B. Set the vAP down and back up to allow changes to take effect:
      In the **Port Manager** tab, select `vap0000`.
      Admin all selected interfaces **DOWN** (CTRL-PLUS).
      Admin all selected interfaces **UP** (CTRL-MINUS).
   C. Sniff packets to observe the authentication behavior:
      On the observation system in the **Port Manager** tab, select only `moni3a`:
      Click **Sniff Packets**.
   D. Observe the results, which should be similar to the following:
      - The station `wlan1` succeeds in authenticating with `vap0000`.
      - LANforge **WiFi Messages** shows `WPA: Key negotiation completed`.

E. Example results:

A. Behavior in LANforge **Wifi Messages**: Control Test



3. **RSNXE Mismatch in EAPOL-Key Message 3/4:**

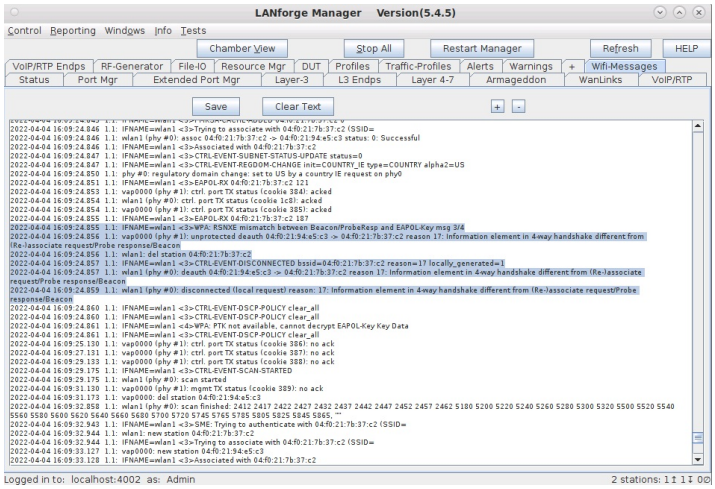
A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write:
`rsnxe_override_eapol=F40100`.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- Message 2/4 shows only encrypted in Wireshark due to having enabled `802.11w`.
- The station `wlan1` fails to authenticate with `vap0000`.
- LANforge **Wifi-Messages** recognizes `WPA:RSNXE mismatch between Beacon/ProbeResp and EAPOL-Key msg 3/4` and gives `CTRL-EVENT-DISCONNECTED for reason 17: Information element in 4-way handshake different from (Re-)associate request/Probe response/Beacon`.
- `Deauthentication` management frame is sent by the station with `Reason code: Information element in 4-way Handshake different from (Re)Association Request/Probe Response/Beacon frame (0x0011)`.
- Compare **EAPOL-Key Message 3 of 4** and **BEACON** RSNXE information for mismatch.

D. Example results:

A. Behavior in LANforge **Wifi Messages**: RSNXE Override Test



B. **Deauthentication**: RSNXE Override Test

```
Type/Subtype: Deauthentication (0x000c)
Frame Control Field: 0xc000
    .... ...0 = Version: 0
    .... 00.. = Type: Management frame (0)
    1100 .... = Subtype: 12
  Flags: 0x00
      .... ..00 = DS status: Not leaving DS or network is operating in AD-HOC mode (To DS: 0 From DS: 0) (0x0)
      .... .0.. = More Fragments: This is the last fragment
      .... 0... = Retry: Frame is not being retransmitted
      ...0 .... = PWR MGT: STA will stay up
      ..0. .... = More Data: No data buffered
      .0.. .... = Protected flag: Data is not protected
      0... .... = Order flag: Not strictly ordered
.000 0000 0011 1100 = Duration: 60 microseconds
Receiver address: 04:f0:21:7b:37:c2
Destination address: 04:f0:21:7b:37:c2
Transmitter address: 04:f0:21:94:e5:c3
Source address: 04:f0:21:94:e5:c3
BSS Id: 04:f0:21:7b:37:c2
    .... .... .... 0000 = Fragment number: 0
    0001 0000 1000 .... = Sequence number: 264
IEEE 802.11 Wireless Management
  Fixed parameters (2 bytes)
    Reason code: Information element in 4-Way Handshake different from (Re)Association Request/Probe Response/Beacon frame (0x0011)
```

```
0000  00 00 44 00 2f 40 40 a0  20 08 00 a0 20 08 00 a0   ··D·/@@·  ···· ····
0010  20 08 00 a0 20 08 00 00  0b 3a 77 fb 00 00 00 00    ···· ···  ·:w·····
0020  00 0c 3c 14 40 01 ee 00  00 00 00 00 00 00 00 00   ··<·@··· ········
0030  0b 3a 77 fb 00 00 00 00  00 00 01 01 ee 00 e3 01   ·:w····· ········
0040  e3 02 e4 03 c0 00 3c 00  04 f0 21 7b 37 c2 04 f0   ······<· ··!{7···
0050  21 94 e5 c3 04 f0 21 7b  37 c2 80 10 11 00        !·····!{ 7·····
```

C. **EAPOL-Key Message 3 of 4**: RSNXE Override Test

```
Frame 263: 289 bytes on wire (2312 bits), 289 bytes captured (2312 bits) on interface moni3a, id 0
Radiotap Header v0, Length 68
802.11 radio information
IEEE 802.11 QoS Data, Flags: ......F.
Logical-Link Control
802.1X Authentication
    Version: 802.1X-2004 (2)
    Type: Key (3)
    Length: 183
    Key Descriptor Type: EAPOL RSN Key (2)
    [Message number: 3]
  Key Information: 0x13c8
      .... .... .... .000 = Key Descriptor Version: Unknown (0)
      .... .... .... 1... = Key Type: Pairwise Key
      .... .... ..00 .... = Key Index: 0
      .... .... .1.. .... = Install: Set
      .... .... 1... .... = Key ACK: Set
      .... ...1 .... .... = Key MIC: Set
      .... ..1. .... .... = Secure: Set
      .... .0.. .... .... = Error: Not set
      .... 0... .... .... = Request: Not set
      ...1 .... .... .... = Encrypted Key Data: Set
      ..0. .... .... .... = SMK Message: Not set
    Key Length: 16
    Replay Counter: 2
    WPA Key Nonce: 6502b4fda34d00be8fba0cb671f6034263d9c8a92c21a73c…
    Key IV: 00000000000000000000000000000000
    WPA Key RSC: 0000000000000000
    WPA Key ID: 0000000000000000
    WPA Key MIC: 31e19714b1e9446765b97589060f326e
    WPA Key Data Length: 88
    WPA Key Data: 33dbdd5179a576dbc3f7d2b58baf36b897159e50d4f4ead7…
```

```
0000  00 00 44 00 2f 40 40 a0  20 08 00 a0 20 08 00 a0   ··D·/@@·  ···· ····
0010  20 08 00 a0 20 08 00 00  9e 01 77 fb 00 00 00 00    ···· ···  ··w·····
0020  00 0c 3c 14 40 01 e7 00  00 00 00 00 00 00 00 00   ··<·@··· ········
0030  9e 01 77 fb 00 00 00 00  00 00 01 01 e6 00 e2 01   ··w····· ········
0040  df 02 dc 03 88 02 24 00  04 f0 21 94 e5 c3 04 f0   ····· $· ··!·····
0050  21 7b 37 c2 04 f0 21 7b  37 c2 10 00 07 00 aa aa   !{7···!{ 7·····
0060  03 00 00 00 88 8e 02 03  00 b7 02 13 c8 00 10 00   ···· ··· ···· ····
0070  00 00 00 00 00 00 02 65  02 b4 fd a3 4d 00 be 8f   ·····e ····M···
0080  ba 0c b6 71 f6 03 42 63  d9 c8 a9 2c 21 a7 3c 0d   ···q··Bc ···,!·<·
0090  4e 1c a1 59 af 0c 0f 00  00 00 00 00 00 00 00 00   N··Y···· ········
00a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ···· ···· ····
00b0  00 00 00 00 00 00 00 31  e1 97 14 b1 e9 44 67 65   ·····1 ·····Dge
00c0  b9 75 89 06 0f 32 6e 00  58 33 db dd 51 79 a5 76   ·u···2n· X3··Qy·v
00d0  db c3 f7 d2 b5 8b af 36  b8 97 15 9e 50 d4 f4 ea   ·····6 ····P···
00e0  d7 58 47 72 85 ea 2f 00  25 f4 d9 e0 3e 72 d5 f3   ·XGr··/· %···>r··
00f0  57 3c 6f 2d fa 34 42 22  fd c7 53 ef 12 7a 2d ad   W<o-·4B" ··S··z--
0100  4f 08 a8 51 68 5e 0d 78  5a a7 d2 2b 31 35 ba ea   O··Qh^·x Z··+15··
0110  66 ee 01 42 6a 6e d2 69  88 fd e0 55 92 2e ef 05   f··Bjn·i ···U····
0120  91                                                ·
```

D. **BEACON**: RSNXE Override Test

```
Tagged parameters (248 bytes)
  Tag: SSID parameter set: juicer-wifi
  Tag: Supported Rates 6(B), 9, 12(B), 18, 24(B), 36, 48, 54, [Mbit/sec]
  Tag: DS Parameter set: Current Channel: 36
  Tag: Traffic Indication Map (TIM): DTIM 0 of 0 bitmap
  Tag: Country Information: Country Code US, Environment Any
  Tag: RSN Information
    Tag Number: RSN Information (48)
    Tag length: 20
    RSN Version: 1
  Group Cipher Suite: 00:0f:ac (Ieee 802.11) AES (CCM)
    Pairwise Cipher Suite Count: 1
  Pairwise Cipher Suite List 00:0f:ac (Ieee 802.11) AES (CCM)
    Auth Key Management (AKM) Suite Count: 1
  Auth Key Management (AKM) List 00:0f:ac (Ieee 802.11) SAE (SHA256)
  RSN Capabilities: 0x00cc
      .... .... .... ...0 = RSN Pre-Auth capabilities: Transmitter does not support pre-authentication
      .... .... .... ..0. = RSN No Pairwise capabilities: Transmitter can support WEP default key 0 simultaneously with Pairwise key
      .... .... .... 11.. = RSN PTKSA Replay Counter capabilities: 16 replay counters per PTKSA/GTKSA/STAKeySA (0x3)
      .... .... ..00 .... = RSN GTKSA Replay Counter capabilities: 1 replay counter per PTKSA/GTKSA/STAKeySA (0x0)
      .... .... .1.. .... = Management Frame Protection Required: True
      .... .... 1... .... = Management Frame Protection Capable: True
      .... ...0 .... .... = Joint Multi-band RSNA: False
      .... ..0. .... .... = PeerKey Enabled: False
      ..0. .... .... .... = Extended Key ID for Individually Addressed Frames: Not supported
  Tag: Supported Operating Classes
  Tag: HT Capabilities (802.11n D1.10)
  Tag: HT Information (802.11n D1.10)
  Tag: Extended Capabilities (10 octets)
  Tag: VHT Capabilities
  Tag: VHT Operation
  Tag: VHT Tx Power Envelope
  Ext Tag: HE Capabilities (IEEE Std 802.11ax/D3.0)
  Ext Tag: HE Operation (IEEE Std 802.11ax/D3.0)
  Ext Tag: MU EDCA Parameter Set
  Tag: Vendor Specific: Microsoft Corp.: WMM/WME: Parameter Element
```

```
0000  00 00 44 00 2f 40 40 a0  20 08 00 a0 20 08 00 a0   ··D·/@@·  ···· ····
0010  20 08 00 a0 20 08 00 00  9c df 5d fb 00 00 00 00    ···· ···  ····]···
0020  00 0c 3c 14 40 01 f2 00  00 00 00 00 00 00 00 00   ··<·@··· ········
0030  9c df 5d fb 00 00 00 00  00 00 01 01 f1 00 ea 01   ··]····· ········
0040  e6 02 eb 03 80 00 00 00  ff ff ff ff ff ff 04 f0   ········ ········
0050  21 7b 37 c2 04 f0 21 7b  37 c2 d0 04 47 40 93 2f   !{7···!{ 7···G@·/
0060  66 00 00 00 f0 00 11 00  00 0b 6a 75 69 63 65 72   f······ ··juicer
0070  2d 77 69 66 69 01 08 8c  12 98 24 b0 48 60 6c 03   -wifi··· ··$·H`l·
0080  01 24 05 04 00 02 00 00  07 0c 55 53 20 24 08 17   ·$····· ··US $··
0090  64 0c 17 95 05 1e 30 14  01 00 00 0f ac 04 01 00   d····0· ········
00a0  00 0f ac 04 01 00 00 0f  ac 08 cc 00 3b 02 80 00   ········ ····;···
00b0  2d 1a 6e 00 03 ff ff ff  ff 00 00 00 00 00 00 00   -·n····· ········
00c0  00 01 00 00 00 00 00 00  00 00 00 00 3d 16 24 05   ········ ····=·$·
00d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00e0  00 00 00 00 7f 0a 04 00  00 02 00 00 00 40 00 40   ········ ·····@·@
00f0  bf 0c b1 59 8a 31 aa ff  00 00 aa ff 00 00 c0 05   ···Y·1·· ········
```

Overriding SAE-Commit Message of WPA3-Authentication Sequence

Goal: Corrupt specific IEs of SAE-Commit messages to provoke WPA3-EAPOL authentication failure.

The `sae_commit_override` field in LANforge **Custom WiFi Parameters** provides means to corrupt or customize certain information elements (IEs) of **SAE-Commit** messages of WPA3 authentication sequence using SAE-encrypted management frames (802.11w). **Scalar** and **Finite-Field** elements may be overriden with arbitary hex strings, provoking authentication failure. Below are documented example test cases and their expected behavior.

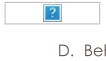
---

1. **Initial Setup for WPA3-Authentication Testing with Simultaneous Authentication of Equals (SAE).**
   The setup requires AP and station NIC drivers capable of enabling SAE encryption (this example uses MediaTEK radios with ath10k(988x) driver), enabling encrypted management frames (`802.11w`), enabling WPA3 and disabling WPA2-PSK authentication in both station and AP.
   A. Set up a virtual AP for testing.
      In this test, it is named `vap0000` on parent device `wiphy0`.
      For more information see Create VAP in Bridge Mode
   B. On a separate radio, create a station to authenticate with `vap0000`:
      In the **Port Manager** tab, select `wiphy1` and click **Create**; select **WiFi STA**, then click **Apply**.
      In this test, the station is named `wlan1` on parent device `wiphy1`.
      For more information see Generating Traffic for WLAN Testing
   C. Configure `vap0000` and `wlan1` to use **WPA3-SAE** encrypted authentication.
      Ensure that `802.11w` is enabled, since it is required for WPA3.
      For more information see Setting up WPA3
   D. Configure `vap0000` and `wlan1` with **SSID** `test-wpa2-psk` and **Keyphrase** `qwertyuiop`.
   E. Create a Monitor Port on its own radio to sniff wireless packets.
      In this test, the monitor port is named `moni3a`.
      For more information see Using Wireshark to Sniff WiFi Monitors

2. **Control (No Change):**
   A. Configure **Custom WiFi** in `vap0000`:
      Select `vap0000` and click **Modify**.
      Navigate to the **Custom WiFi** tab.
      Ensure that no `sae_commit_override` parameter is set in **User-Specified supplicant/hostapd configuration text**.
      Click **Apply** then **OK**.
   B. Set the vAP down and back up to allow changes to take effect:
      In the **Port Manager** tab, select `vap0000`.
      Admin all selected interfaces **DOWN** (CTRL-PLUS).
      Admin all selected interfaces **UP** (CTRL-MINUS).
   C. Sniff packets to observe the authentication behavior:
      On the observation system in the **Port Manager** tab, select only `moni3a`:
      Click **Sniff Packets**.
   D. Observe the results, which should be similar to the following:
      - Packets are not malformed.
      - The station `wlan1` succeeds in authenticating with `vap0000`.
      - RSN Information Element is found in `EAPOL-Key Message 3 of 4` sent by `vap0000` with WPA-Key-Data field.
   E. Example results and expected behavior:
      A. **SAE-Commit** Message: Control Test

      B. **SAE-Confirm** Message: Control Test

      C. Behavior in LANforge **Wifi Messages**: Control Test (1)

      D. Behavior in LANforge **Wifi Messages**: Control Test (2)

3. **SAE Commit Override:**

A. Configure **Custom WiFi** in `vap0000`:
Select `vap0000` and click **Modify**.
Navigate to the **Custom WiFi** tab.
In the **User-Specified supplicant/hostapd configuration text** field, write (with no line breaks):
`sae_commit_override=13ffbad00d215867a7c5ff37d87bb9bdb7cb116e520f71e8d7a794ca2606d537ddc6c099c40e7a25372b80a8fd443cd7dd222c8ea21b8ef372d4b3e316c26a73fd999cc79ad483eb826e7b3893ea332da68fa13224bcdeb4fb18b0584dd100a2c514`.
Note the recognizable '`bad00d2`" in this hex.
Click **Apply** then **OK**.

B. Reset ports and sniff packets:
Repeat steps B through D of Step 2.

C. Observe the results, which should be similar to the following:
- The station `wlan1` fails to authenticate with `vap0000`.
- The station `wlan1` cycles between scanning and association attempts.
- LANforge Wifi-Messages shows `CTRL-EVENT-SSID-TEMP-DISABLED` for `reason=CONN_FAILED`.
- No longer **Confirm** message is visible in the authentication sequence, rather **Deauthentication**.

D. Example results and expected behavior:

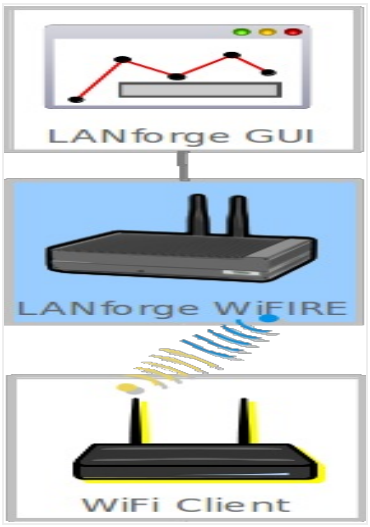
A. **SAE-Confirm** Message: Override Test

B. Behavior in LANforge **Wifi Messages**: Override Test

WiFi: Inflight Entertainment: video demonstration

Goal: Learn about how to emulate a crowded WiFi environment that might occur in an airplane cabin.

Watch a demonstration of how to model the inflight entertainment experience using 200 or more virtual stations simultaneously streaming video. This scenario requires LANforge version 5.3.9, six CT521a, and one CT523c as a controller, for virtual APs and as a content server.

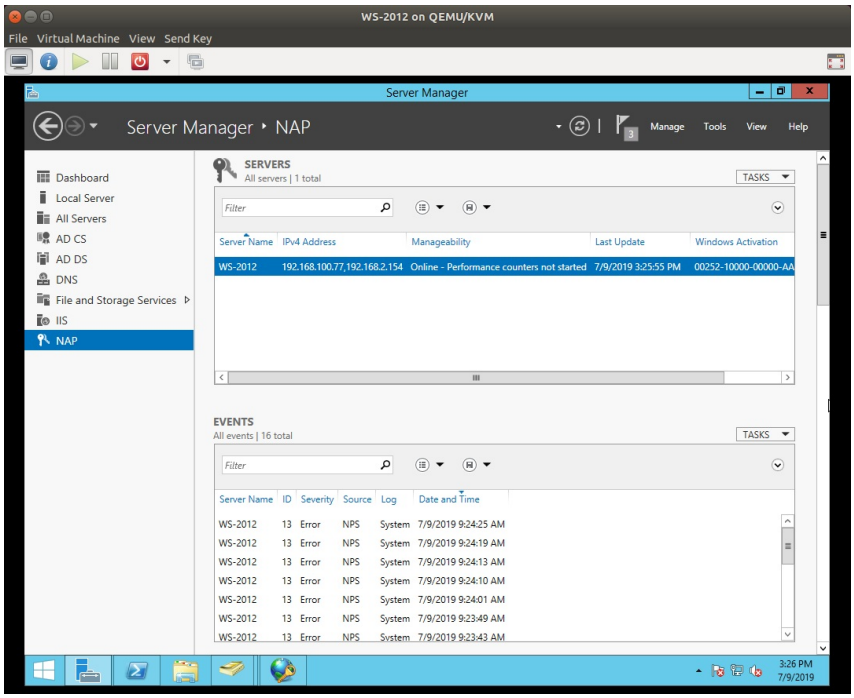


WiFi Inflight Entertainment demonstration

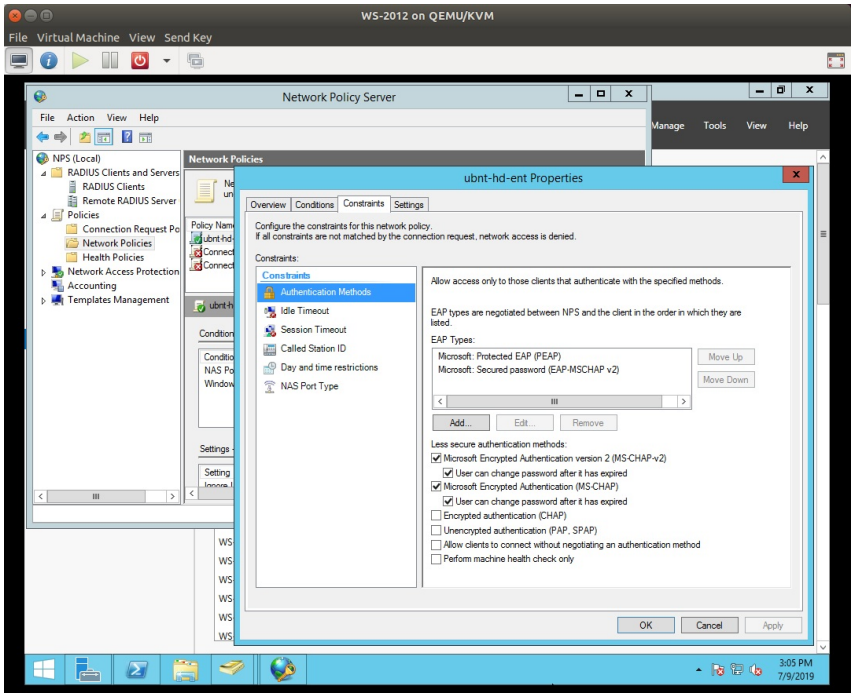
EAP-PEAP AND MSCHAPv2 Configuration

Goal: To set up a lanforge wireless station with EAP-PEAP AND MSCHAPv2 authentication.

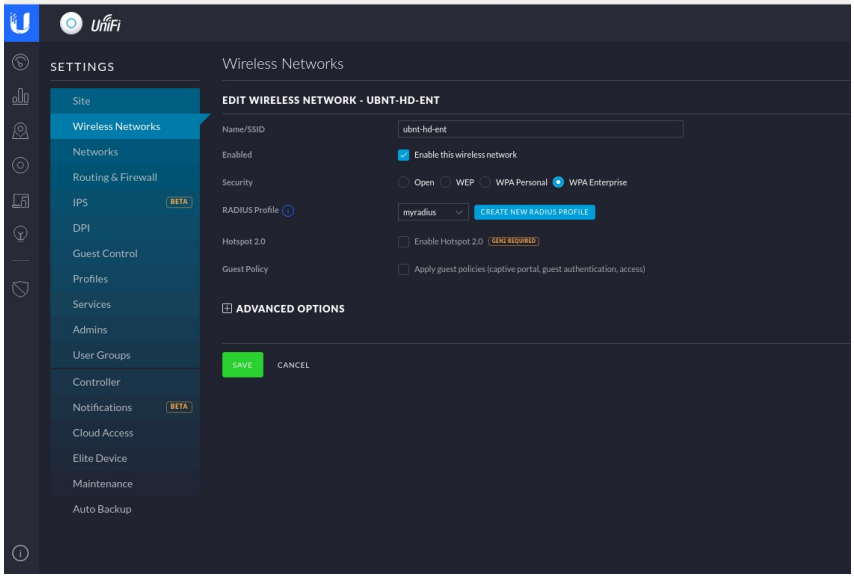
1. **RADIUS Server is a Windows Server 2012 VM using its internal RADIUS Server specifically on interface 192.168.2.154.**



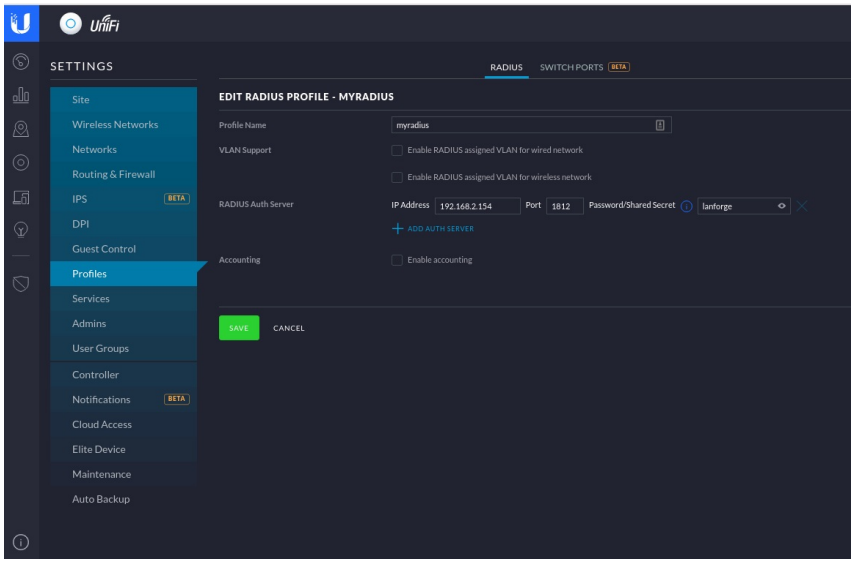
2. **RADIUS Server has both EAP-PEAP and MSCHAPv2 configured for use.**



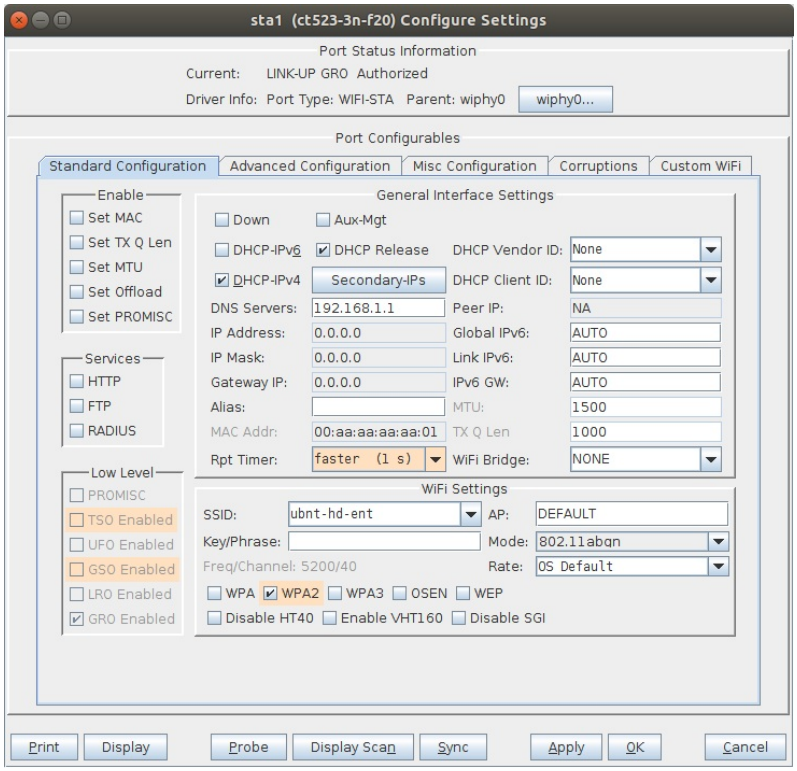
3. **AP is a Ubiquiti HD AP with SSID ubnt-hd-ent with WPA Enterprise security.**



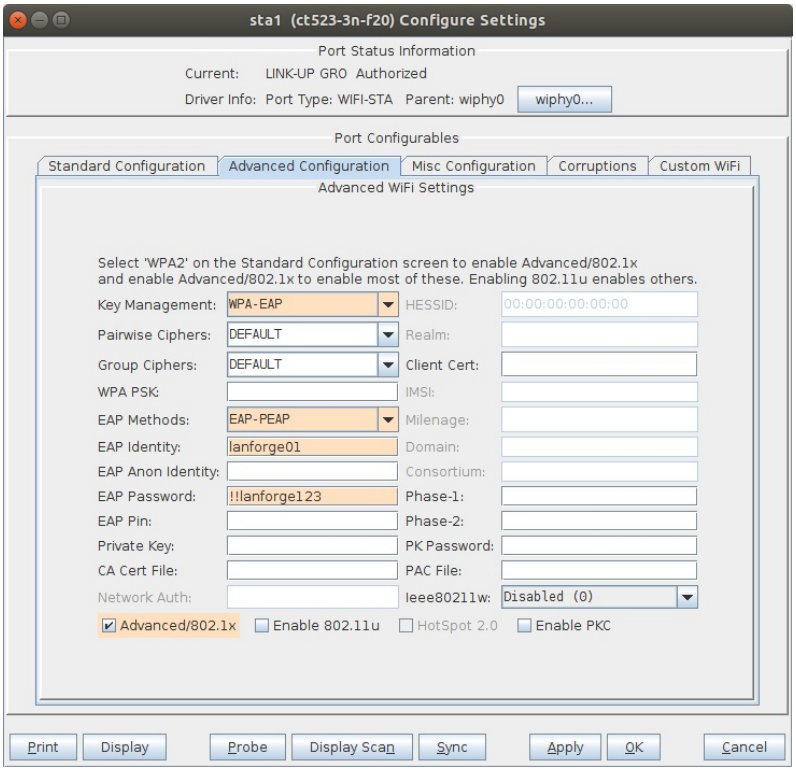
4. **The AP controller points to the RADIUS server at 192.168.2.154.**



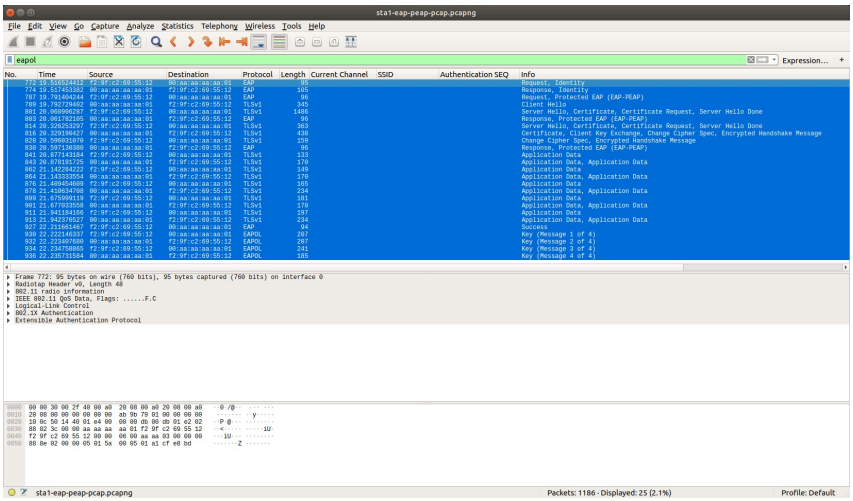
5. **LANforge STA1 basic configuration:**



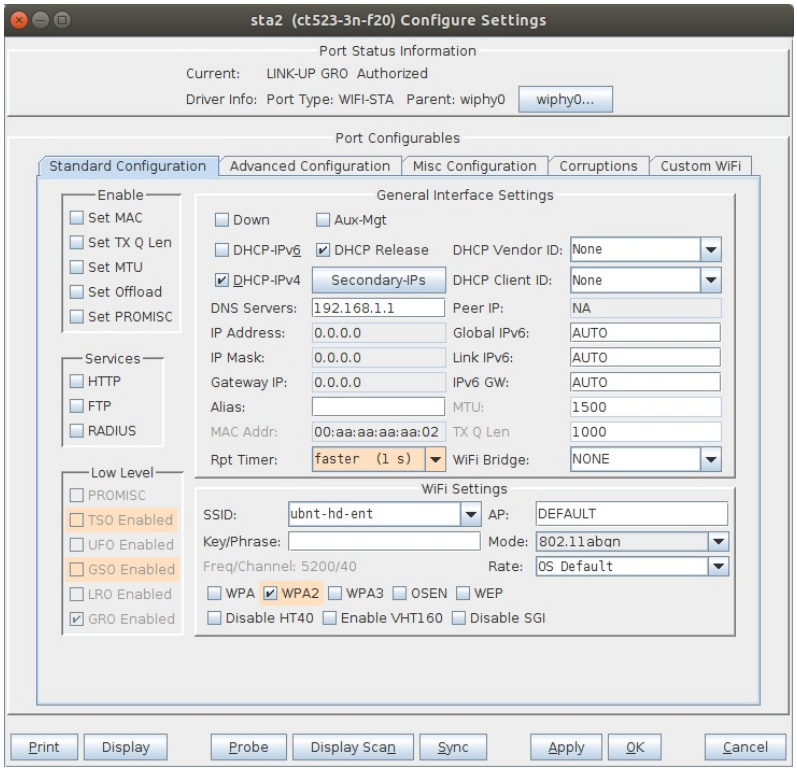
6. **LANforge STA1 advanced configuration is using EAP-PEAP.**



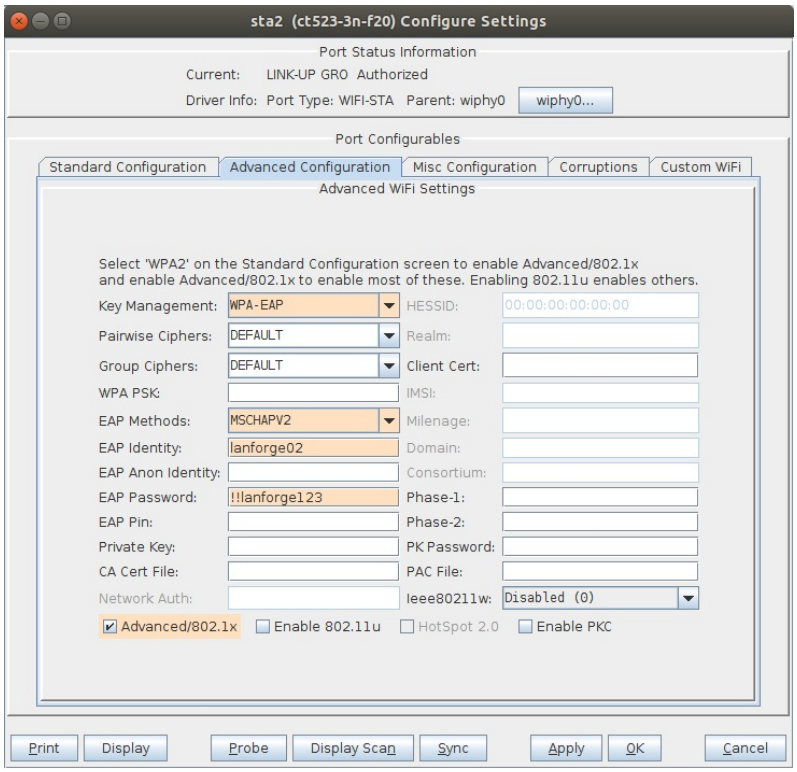
7. **Wireshark capture of STA1 EAPOL authentication.**



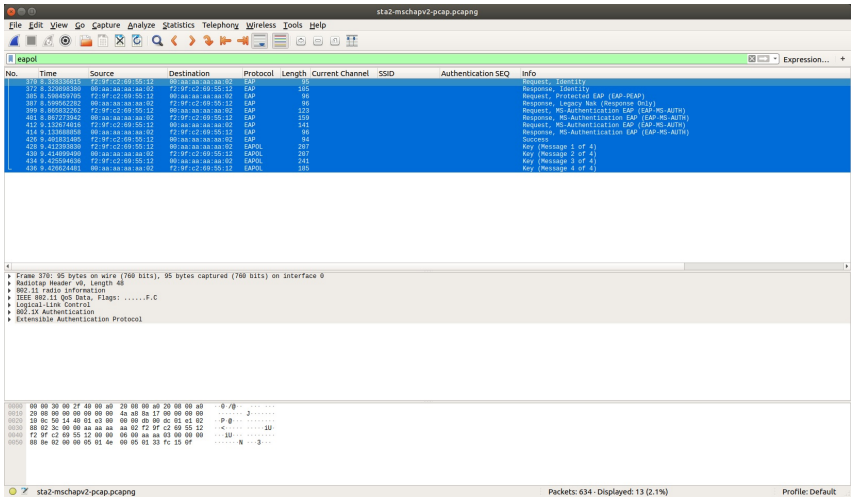
8. **LANforge STA2 basic configuration:**



9. **LANforge STA2 advanced configuration is using MSCHAPv2.**



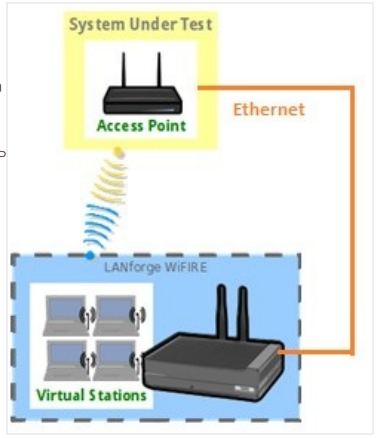
10. **Wireshark capture of STA2 EAPOL authentication.**



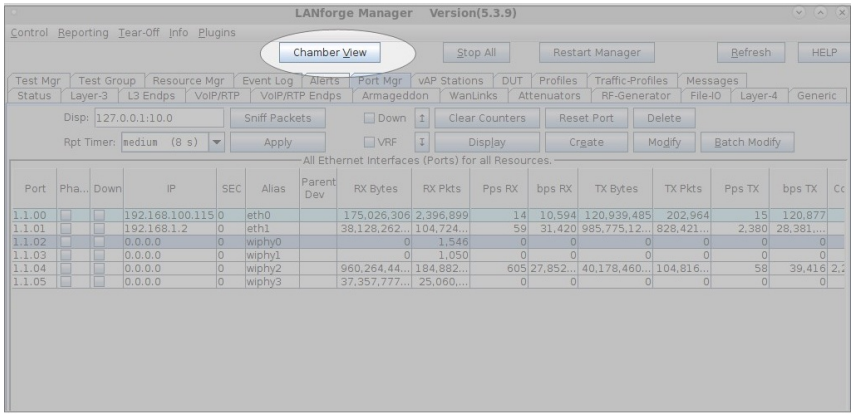
Basic AP Testing with LANforge (using Chamber View)

Goal: Set up virtual stations using a LANforge system, connect them to an AP under test, send traffic, and make measurements.

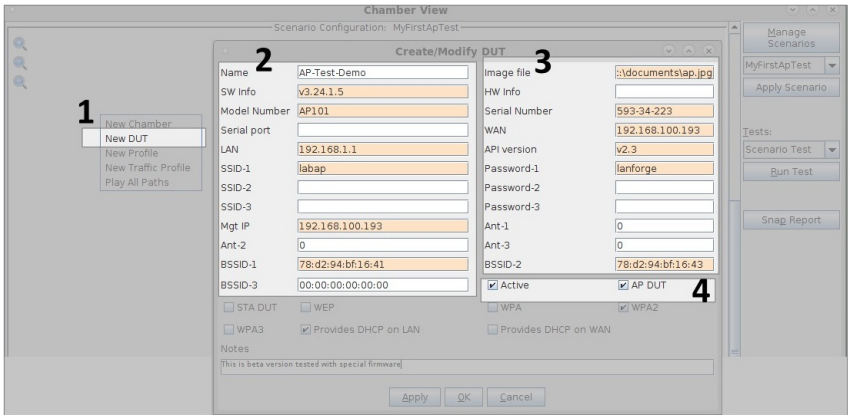
In this test scenario a LANforge system is used to create 5 virtual WiFi stations and configured to connect to a real WiFi AP under test. The AP's ethernet interface is connected to a LANforge ethernet interface allowing the LANforge system to create both the wireless stations and ethernet server. The test is then configured to send simple upstream and downstream UDP traffic between the server and the 5 WiFi stations through the AP under test.



1. **Click on the Chamber View button in the LANforge GUI to launch the Chamber View screen.**

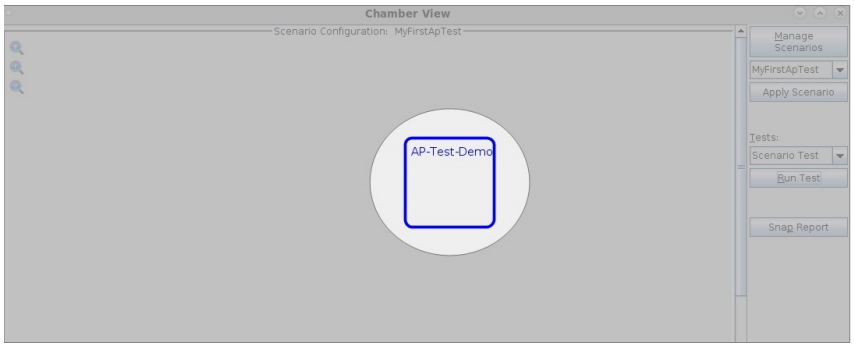


2. **Configure an AP under test (DUT).**

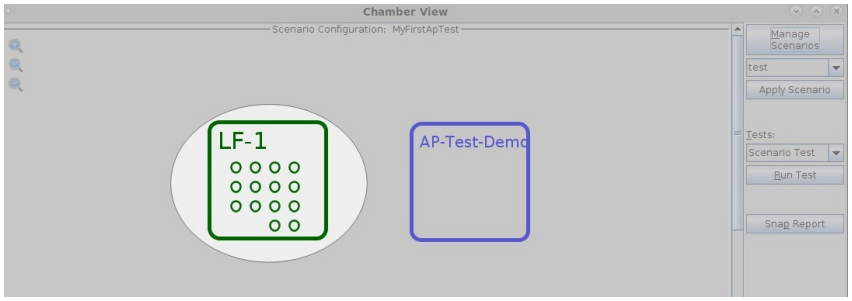


3. **To configure the DUT:**
   A. Right click anywhere on the canvas in Chamber View and select "New DUT" from the menu.
   B. Enter all the known details about the AP under test including the SSID (multiple SSIDs if the AP has multiple SSIDs), BSSID (multiple BSSIDs if the AP has multiple radios with the same SSID).
   C. If the AP has security enabled, enter the security information.
   D. Set the DUT to Active by checking the 'Active' checkbox.
   E. Select the 'AP DUT' checkbox to indicate that this DUT is an Access Point.

4. **Once OK is clicked, check to make sure the DUT appears in Chamber View as shown below.**

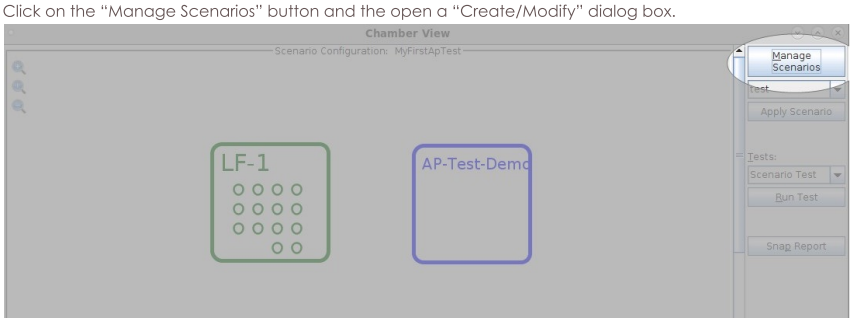


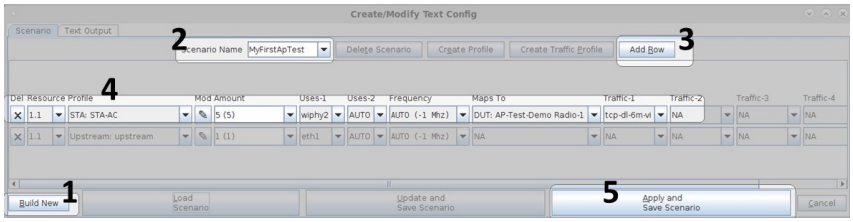
5. **Select the LANforge system to be used in the test.**



6. All active LANforge systems will automatically appear in Chamber View. Select the unit being used in the test and drag to the middle of the canvas next to the AP under test.The small circles inside the LANforge box represent the SMA connectors. The columns of circles represent the individual radios in the system For instance, in the screenshot about LF-1 has 4 columnsof circle indicating 4 radios. The first two are 3x3 radios and the second two are 4x4 radios with 4 circles each.
7. **Create the test scenario**

   Click on the "Manage Scenarios" button and the open a "Create/Modify" dialog box.





8. In the Create/Modify dialog box

    A. Click on “Build New” button to create a new scenario.
    B. Enter a scenario name e.g: MyFirstApTest (no spaces allowed)
    C. Click “Add Row” button to add one or more rows.

9. **Create 5 WiFi Stations, map them to the AP under test and create traffic streams.**

    Row1 is used to create the 5 WiFi stations. Do the following in Row1:

    A. The “Resource” indicates the ID of the LANforge system being used in the test. Since we are using LF-1 for this test, select resource “1.1”.
    B. For the “Profile” column, pick “STA:STA-AC” profile which is one of the existing default station profiles that will create 11ac WiFi stations. The user can select from existing profiles are create new profiles by clicking on the “Create Profile” button at the top.
    C. In the “Amount” drop down, enter 5 to indicate the 5 stations being created.
    D. The “Uses-1” drop down indicates the primary hardware interface on which the user intends tocreate these clients. Here, select the right radio interface on which the WiFi stations are to be created.
    E. The “Uses-2” is for bridge mode and it can be left as “AUTO” for this scenario.
    F. Under the Frequency section, the user can select the channel on which to create the stations. If left to Auto, the stations will scan across all channels and find the target AP. So in this case we leave this setting to “AUTO”
    G. The “Maps To” represents which target AP the Stations should connect to. Since we have already configured the DUT in step1, we can simply select the DUT.
    H. For traffic, the GUI allows user to run up to 5 traffic streams on each client. In this case we are choosing one of the default traffic profiles “tcp-dl-6m-vi” which is a 6Mbps downstream tcp traffic stream. Once this is selected, the test will create 5 traffic streams of this type, one for each station. Users can create their own custom traffic streams by clicking on the “Create Traffic Profile” button.

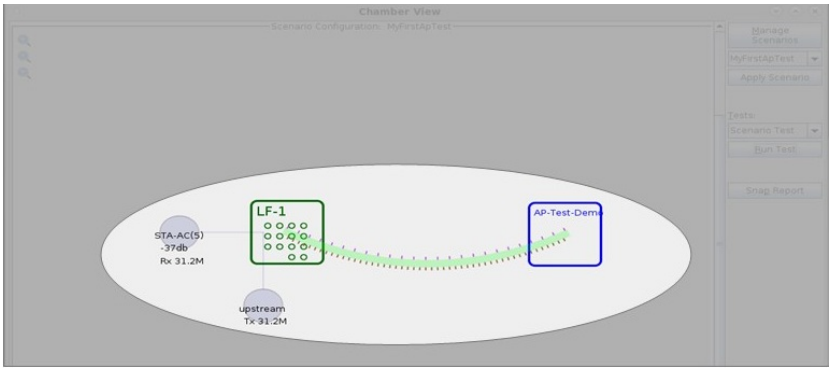
10. **Create an upstream server**

    Add a new row to create an upstream server. For this new row.

    A. Select resource “1.1” as the upstream server is being created on the same LANforge system.
    B. Select the profile as “Upstream” to represent an Upstream endpoint.
    C. Set Amount to 1 as we are creating only 1 upstream server
    D. Set “Uses-1” to eth1 as we are creating the server on “Eth1” interface on the LANforge system.
    E. Now click on “Apply and Save Scenario” to save the test scenario.

11. **Load and Run Test Scenario**

    The next step is to load the scenario and run the test.



    A. Once the scenario is applied, click on the “Build Scenario” button and this action will configure the 5 stations and the traffic streams.
    B. Then in order to run the test scenario, select the “Scenario Test” option from the Tests drop down and click on the “Run Test” button.
    C. The test then starts running and the 5 stations will connect to the AP under test.

D. Once the connections are complete (indicated by the thick green line between the LANforge system and the AP under test", the traffic streams are started (indicated by the dotted lines on both sides of the thick green line).

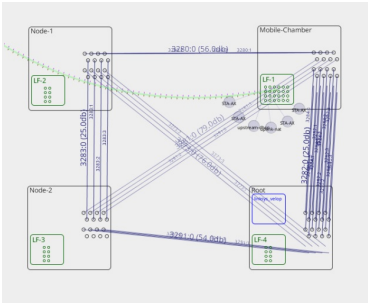
E. In the above example the Upstream shows a Tx stat of 31Mbps representing a total of about 30Mbps of downstream traffic (5 stations each doing 6Mbps) and the stations show an aggregate receive of about the same 31Mbps

F. "STA-AC (5)" indicates the 5 802.11ac stations connected to the AP under test and the -37db indicated the current signal strength the LANforge stations are seeing from the AP.

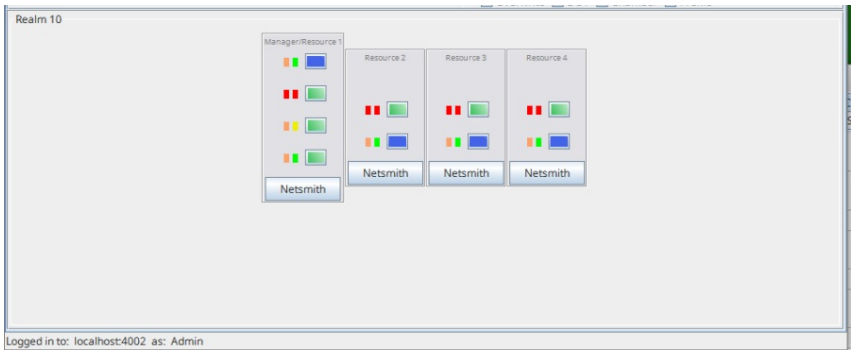
Setting up Attenuator Connections in LANforge

Goal: Configure the connections between chambers in Chamber View to match the physical cables connecting RF chambers in actuality.

This Chamber View setup will replicate the physical cabled attenuator setup configuration. The Chamber View connections created with this cookbook will allow accurate frequency attenuation in the physical multi-chamber setup during tests such as TR-398, Mesh and other LANforge tests. The physical chamber used in the example below is a 4-RF Chamber setup.

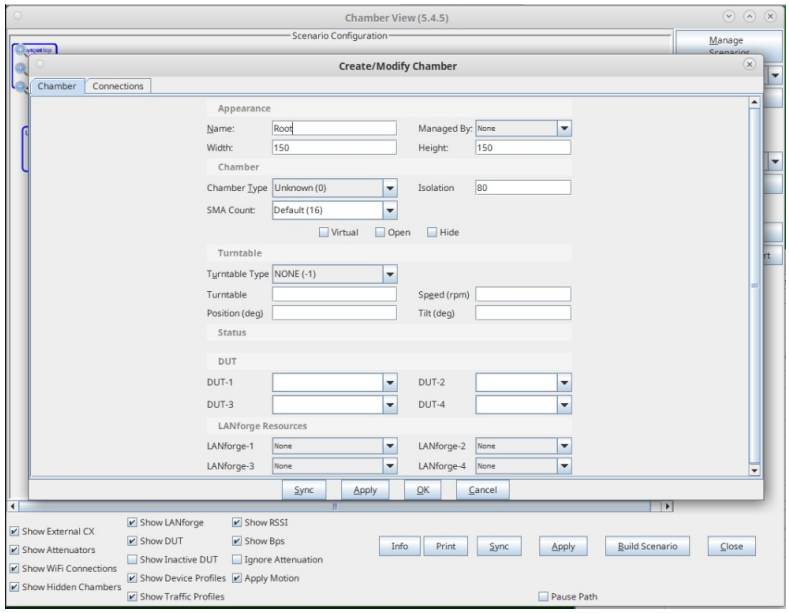


1. If the final testbed to be setup has multiple LANforges, check if they need to be clustered and cluster them. Please contact support@candelatech.com to help configure the cluster. Example below is the status tab of the manager of a set of clustered systems.

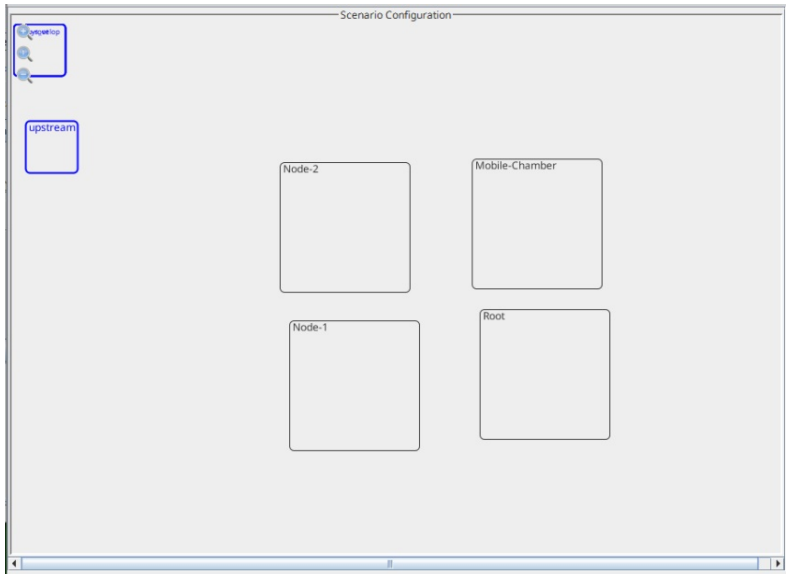


2. Create chambers in Chamber View that match your testbed setup. The example below is for 4 chambers. If your Chamber View scenario is all setup and ready for attenuator connection setup, skip to step 3.

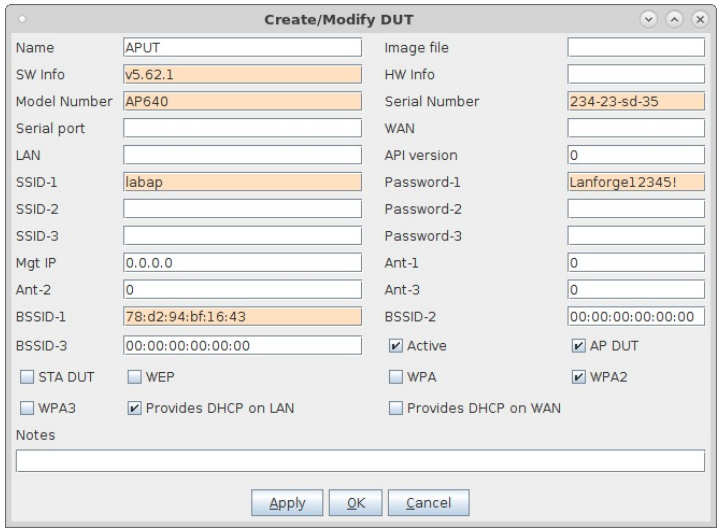
A. Open Chamber View by clicking on the **Chamber View** button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. Right-click in Chamber View to create various objects (and the chambers for this example).



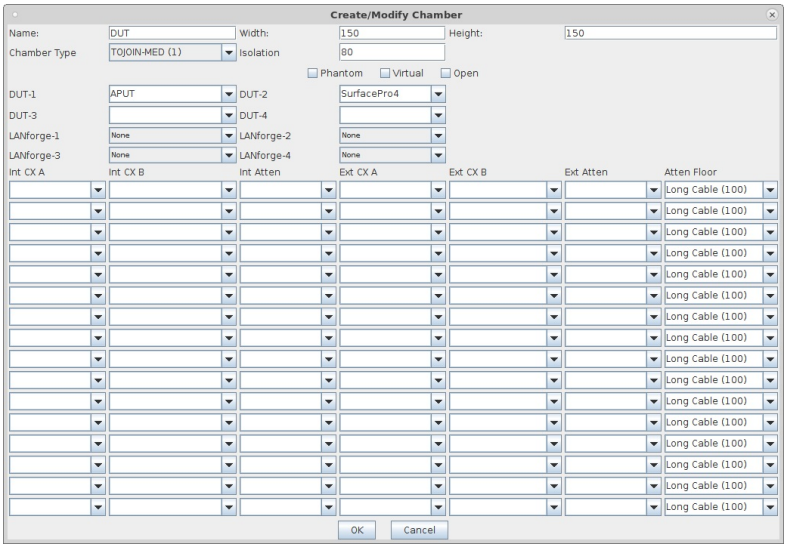
B. Once the 4 chambers are created, the Chamber View window should look similar to below.



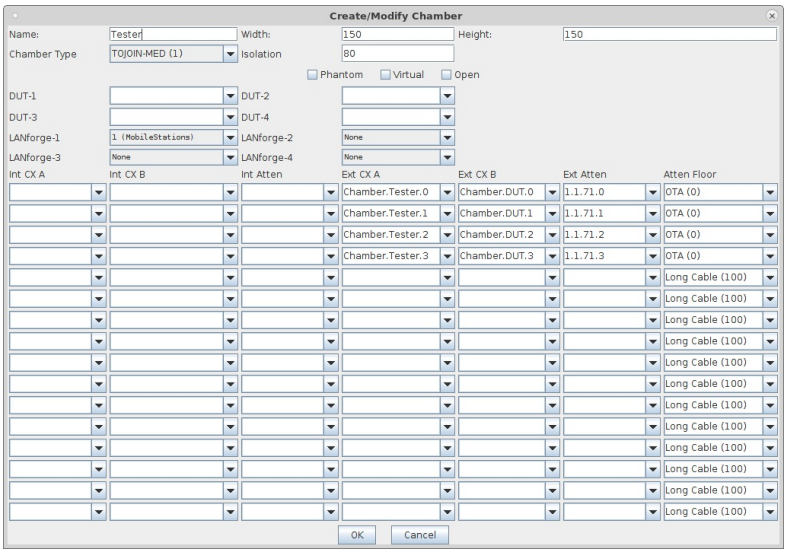
3. Add LANforge(s) and DUT(s) in their appropriate chambers.
   A. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



B. Add the DUT to the chamber object by double-clicking the specified chamber that requires the DUT and selecting the DUT from the dropdown under the DUT section of the window. If there are no chambers, a fake chamber a chamber can be made, but the test will not be isolated and may not function as desired.

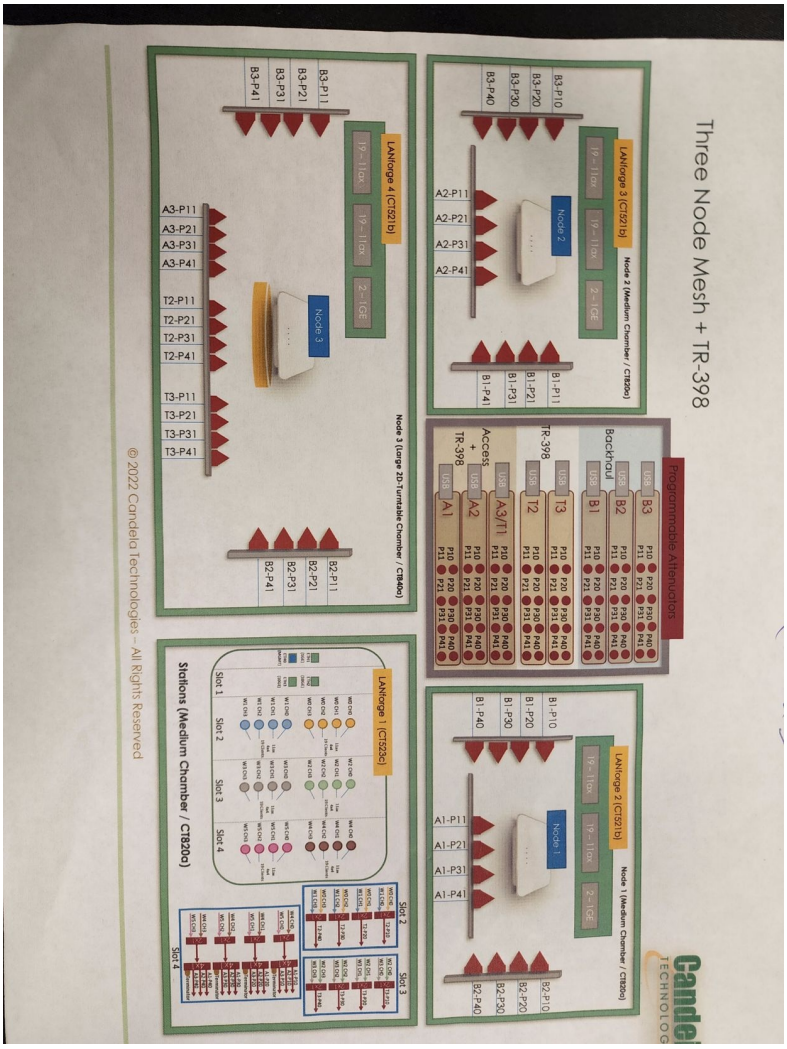


C. Add the LANforge to a chamber object. Double-click the specified chamber that requires the LANforge and select the hostname from the dropdown under the LANforge Resources tab.

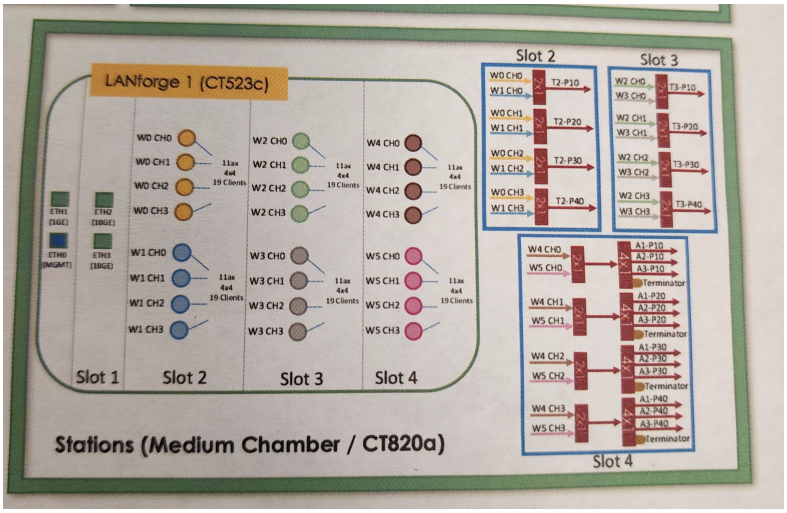


4. Gather notes for your attenuator connections.

A. Looking at the attenuator diagram of the testbed setup, identify where each attenuator goes to. Below is an example of an attenuator diagram of a 4-chamber setup. Another name for the Node-3 Chamber is the Root chamber.

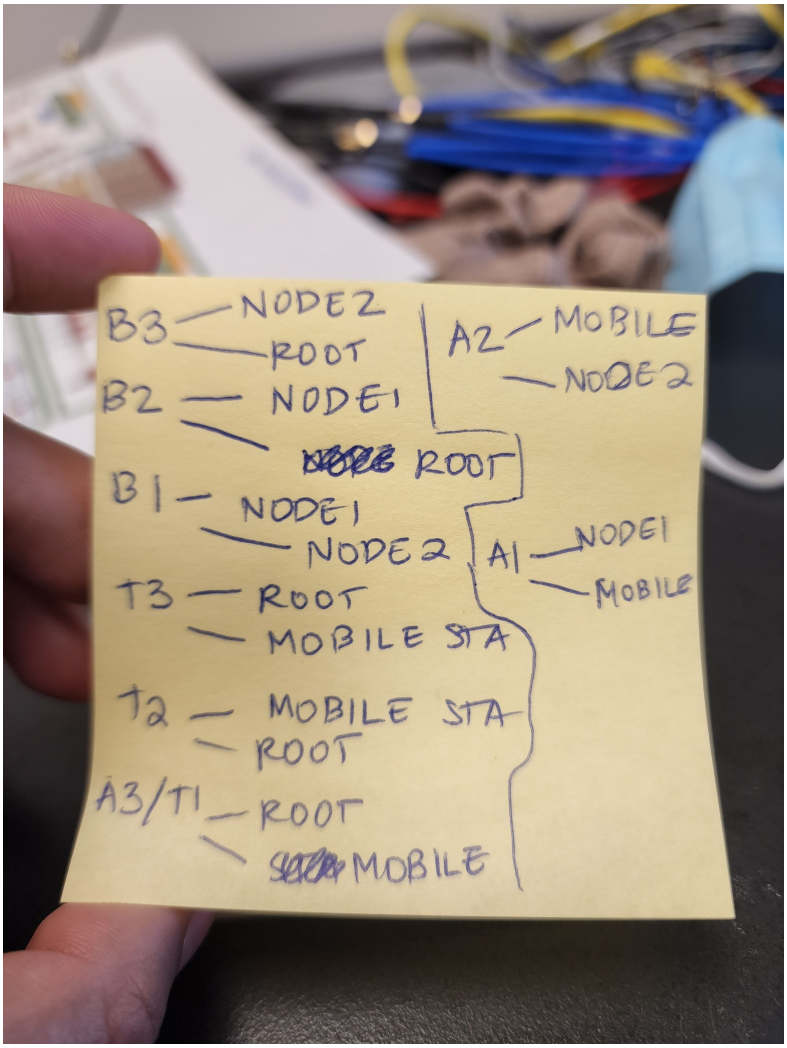


B. Below is a closer look at the bottom right box of the attenuator diagram in the step above.

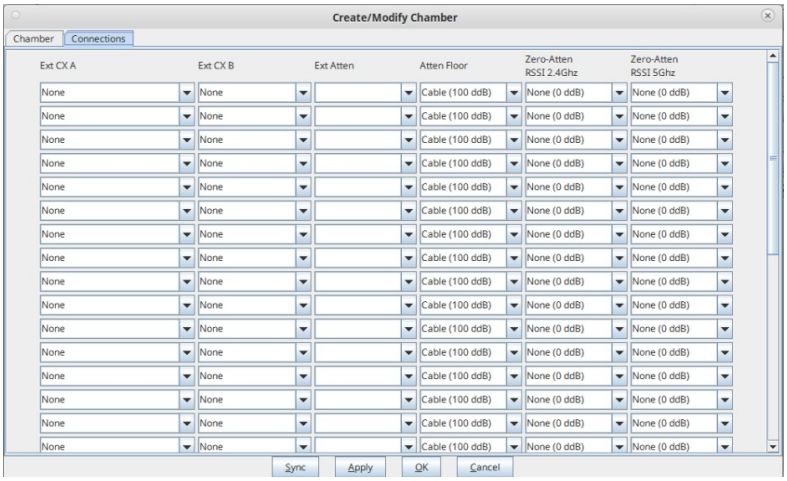


C. Gather the serial numbers of the attenuators on the stack and make sure the attenuators are stacked in the same order as on the paper.

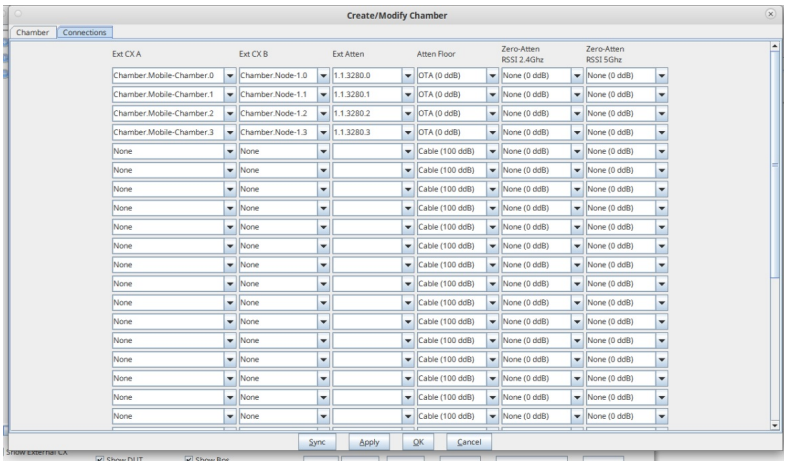
D. Trace each attenuator out to the end chambers. In this example, the attenuators are stacked in the same order as the diagram, so we can assume that each name on the diagram (such a B3) corresponds to the according attenuator in the physical stack. On the diagram, B3 is both in Node 2 and Node 3/Root. On the sticky note, B3 has lines to both Root and Node 2.



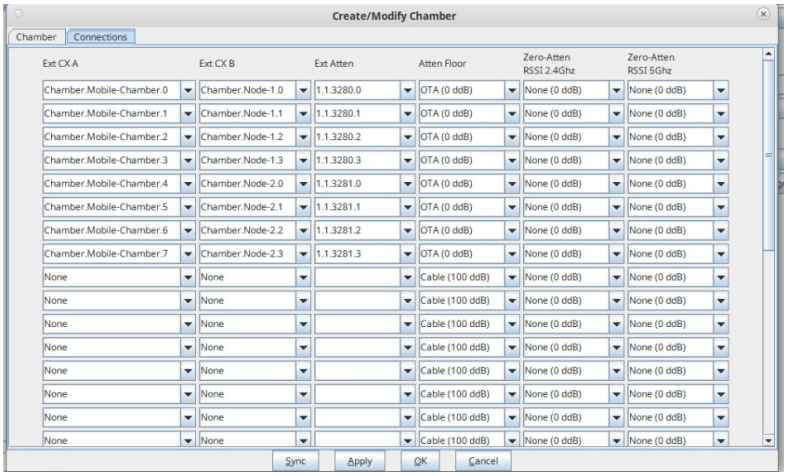
5. Create all the connections for each chamber.
   A. Pick the first attenuator to start connections on. For example. we are first starting with attenuator A1, which goes to both Node-1 Chamber and Mobile Clients Chamber. In order to create the lines between those 2 chambers in Chamber View, the connections can be placed in either chamber (but not both). In this example, the connections will be placed in the Mobile Stations chamber, under the Connections tab.



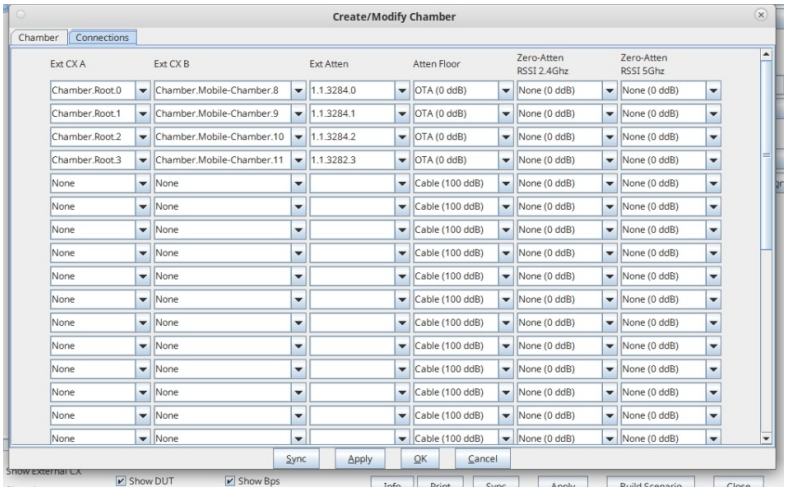
B. Determine the number of SMA cables used for each chamber. For A1, all 8 of the attenuator SMA ports are used, 4 on Mobile Stations, 4 on Node-1. Add those ports to the connections tab. A side is Mobile Client since it is the chamber open for editing, B is Node-1. First, the port number at the end after the chamber name (Chamber.Mobile-Chamber.0), as long as that port is only used once in any of the connections. Our 4 ports on both A and B sides are 0-3, because those are the first 4 ports that haven't been used yet. None of the attenuator ports haven't been used yet either, so those are 0-3 (serial no 3280, attenuator A1). The atten floor is 0db (OTA) because at some point in thse connections, there is 1 over the air connection. If the attenuator and chambers are strictly cables only, then the atten floor would be cabled. Click Apply to save data and OK to save data only and close window.



C. Once the first set of ports are used for a chamber, do not reuse those ports (ports 0-3 were used above for Mobile Stations, Node 1 and Attenuator A1). Attenuator A2 (serial 3281) is the next attenuator to be entered in. In the notes gathered, A2 is between Mobile-Stations and Node 2. Each of the chambers has 4 connections to the attenuator, so those are also added to 'Connections' tab within the Mobile Stations chamber. Those connections can also be added to the Node 2 chamber instead, (but not in both). Notice on the A-side, the next port added was Chamber.Mobile-Chamber.4, since 0-3 are already used up. Since none of the SMA ports are used in Node-2, the SMA ports start at 0 and go til 3. None of the attenuator 3281's SMA ports are used so far, ports 0 to 3 are used. Click Apply to save data and OK to save data only and close window.



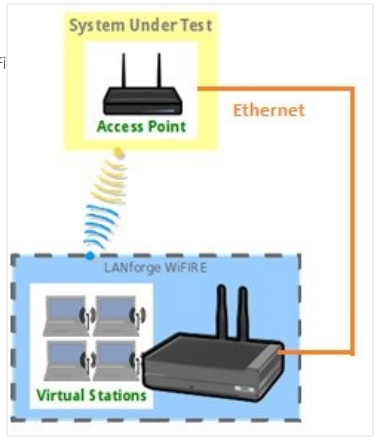
D. Last example is connections for attenuator A3/T1, which goes from root to mobile chamber. These connections are being added in the Root chamber now. Since no ports (in all chambers of all connections) are used from the Root chamber, ports 0-3 are used on A side. B side (mobile) already has 0-7 ports used in the Mobile chamber, so this example starts with port 8. Click Apply to save data and OK to save data only and close window.



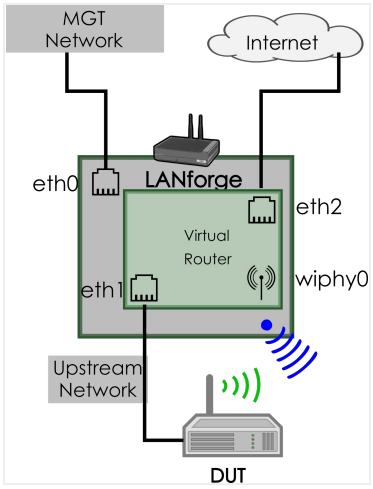
Advanced AP Testing with LANforge (using Chamber View)

Goal: Provide VLANs and upstream router for AP, as well as stations for complete end-to-end testing.

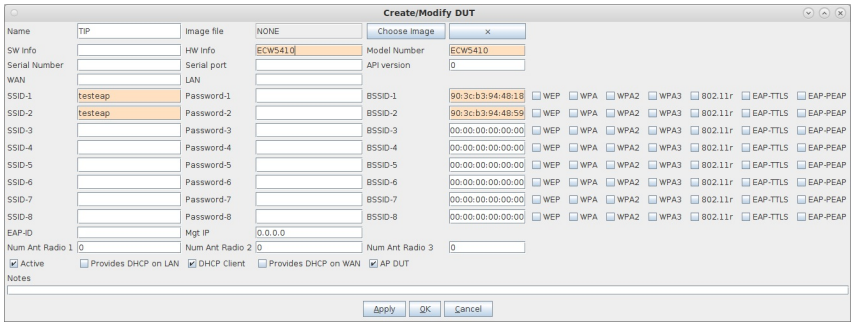
In this test scenario a LANforge system will provide a DHCP server with 802.1q vlans, provide NATed access to the internet for the AP, as well as create stations that will connect to the WiFi AP under test. The AP's Ethernet interface is connected to a LANforge Ethernet interface allowing the LANforge system to create both the wireless stations and Ethernet server. This is an advanced topic, so some steps will be glossed over. This requires LANforge version 5.4.3 or higher.



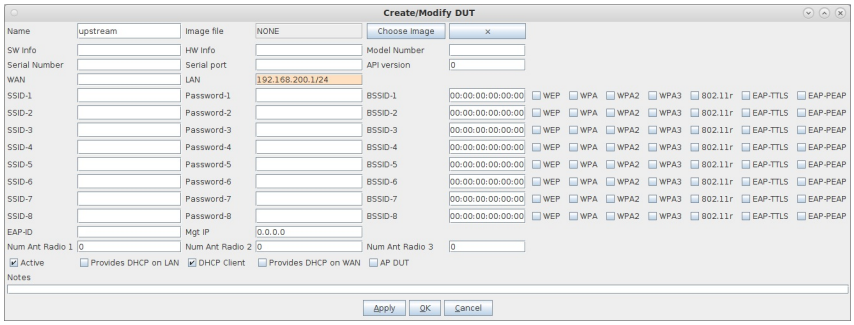
This scenario is built with these components



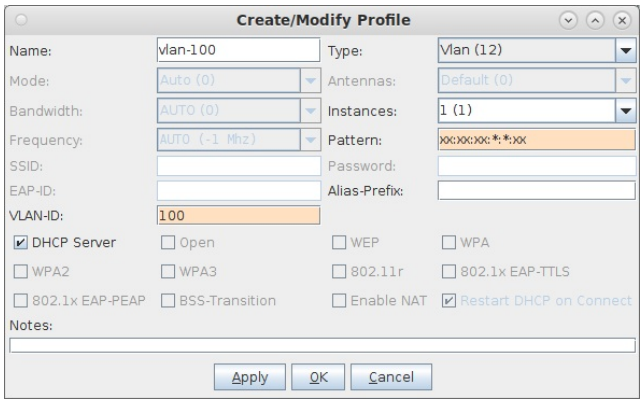
1. **Click on the Chamber View button in the LANforge GUI to launch the Chamber View screen.**
2. **Configure an AP under test (DUT).**



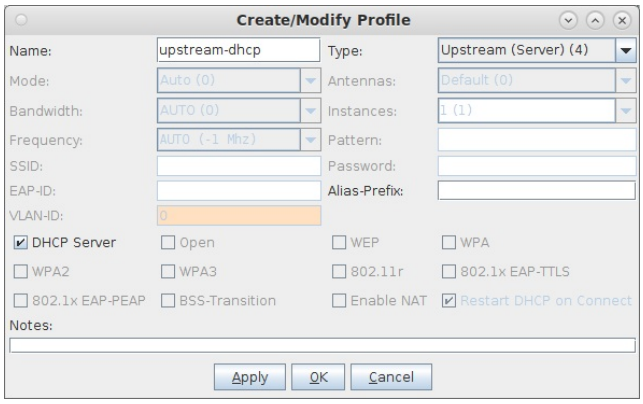
3. **Configure an Upstream (DUT).** This represents the upstream route for the AP's path to the Internet. The LANforge Ethernet port (eth2 in this example) that provides uplink to the internet must be configured with a static IP address (not via DHCP).



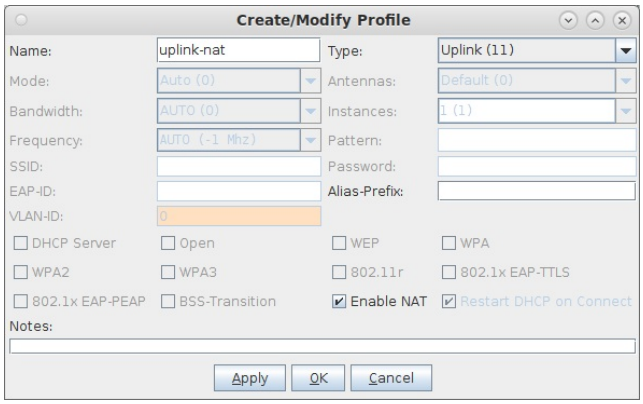
4. **Profiles used in this scenario include VLAN.**



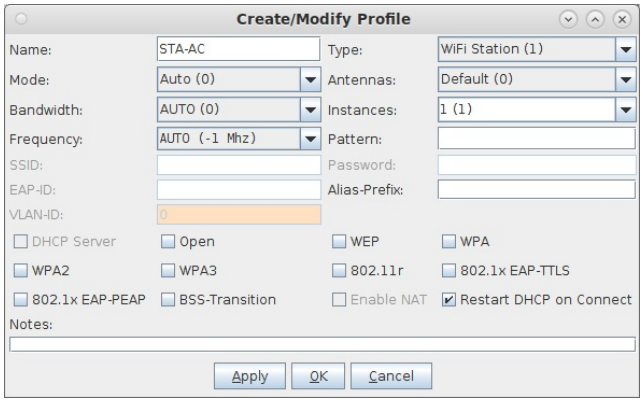
5. **Profiles used in this scenario include Upstream.**



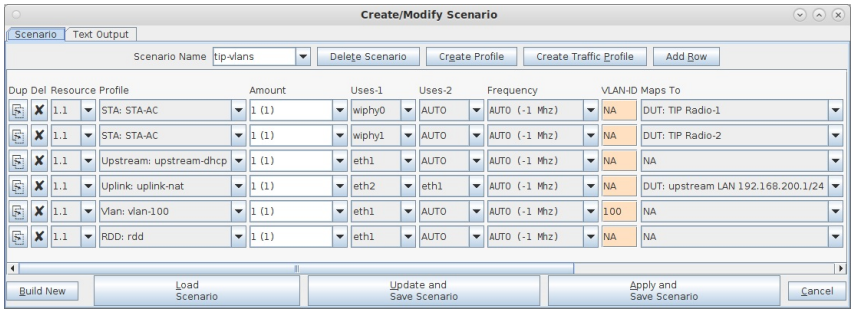
6. **Profiles used in this scenario include Uplink.**



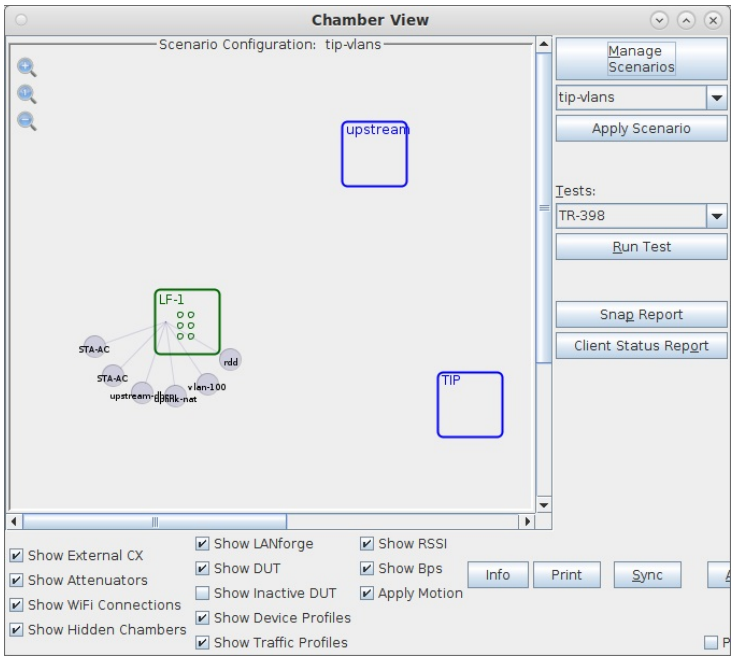
7. **Profiles used in this scenario include Stations.**



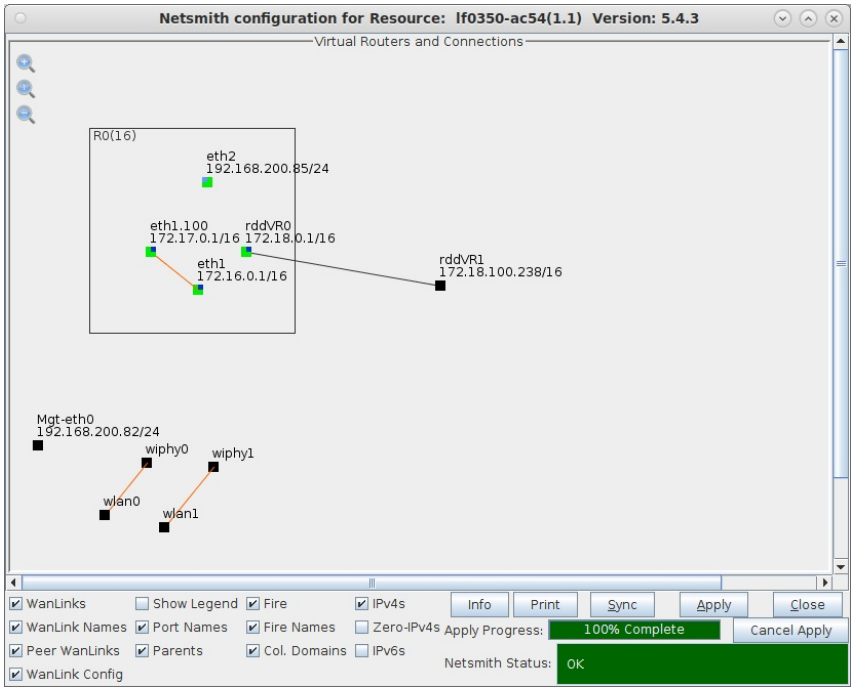
8. **Create a Chamber View Scenario.** The first two rows specify the radios used for virtual stations. The third row is for the Upstream port. This is what the AP connects to with its WAN port. The Uplink-Nat row, associated with eth2, indicates that the eth2 port routes the APs traffic to the internet. The Maps-to portion of this line is important. The secondary port 'eth1' mapping associates this uplink port with the virtual-router that will hold eth2. The VLAN line creates an 802.1q VLAN on port eth1. Additional VLANs can be added as needed. The final RDD line creates a virtual Ethernet pair and adds to the virtual router associated with eth1. The RDD port outside of the virtual router can be used as an upstream port that is able to generate traffic across eth1 and any VLANs created.



9. **Chamber-view screen with LANforge, AP, and Upstream.** When satisfied with the scenario, click Build Scenario and the ports and virtual router will be created and configured.



10. **Netsmith virtual router.** The step above creates the Netsmith configuration. Eth2 is our uplink port, it enables NAT for anything routing out of it. Eth1 and the vlan eth1.100 provide DHCP leases to the DUT and DUT's stations. The rddVR1 port is a convenient 'upstream' port for generating traffic to/from the stations across the DUT. The IP addresses for eth1, eth1.100 and rddVR* are auto-created by the scenario building logic. The wlan0 and wlan1 are stations that can be associated with the AP.

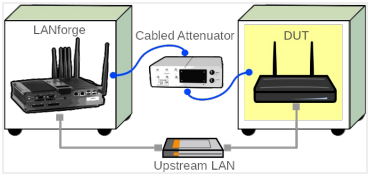


11. At this point, your AP should be able to get a DHCP address from the LANforge virtual router and connect to the Internet. You can run various tests using the pull-down test selector in Chamber View, do manual testing, or launch fully automated scripted tests against the system.

Testing Receiver Sensitivity for a WiFi Device

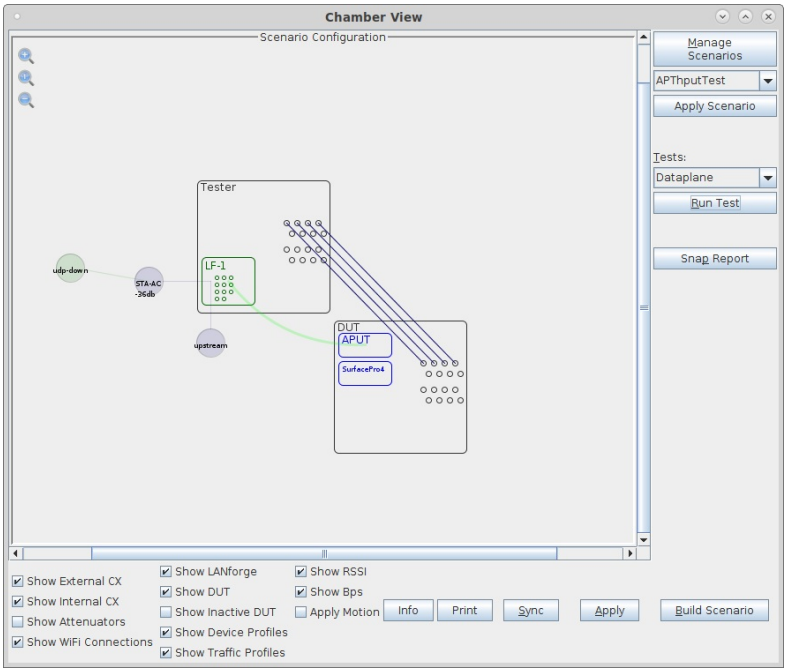
Goal: Setup and run a Receiver Sensitivity test for an AP using the LANforge CT523c or similar system in order to test how well the AP can receive packets with different MCS at different RF Signal levels.

In this test scenario, the LANforge CT523c is used to generate packets at specific MCS and spatial streams at a range of attenuation. This example assumes you are familar with Chamber View, and that you have a LANforge system with a wave-2 WiFi Card, a programmable attenuator like the CT704b and two isolation chambers like the CT820a. The AP should be in one chamber, the LANforge system is in the other chamber, and the Attenuator is cabled between them. This feature is in LANforge version 5.3.9 and higher.

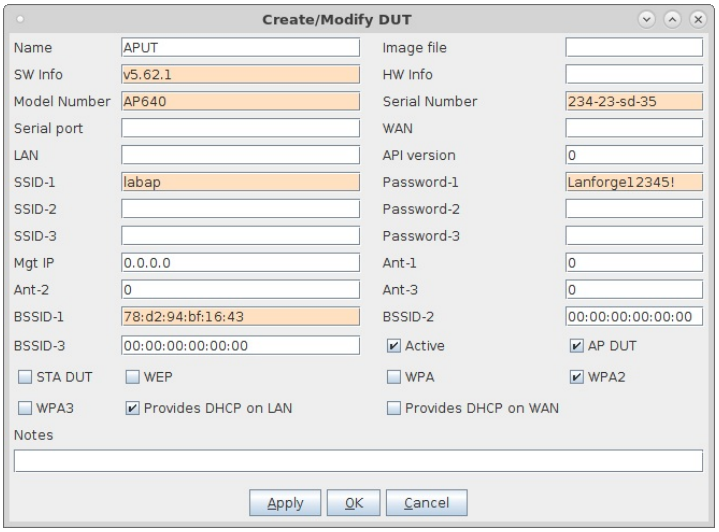


1. Configure Chamber View for Dataplane and Similar Tests.

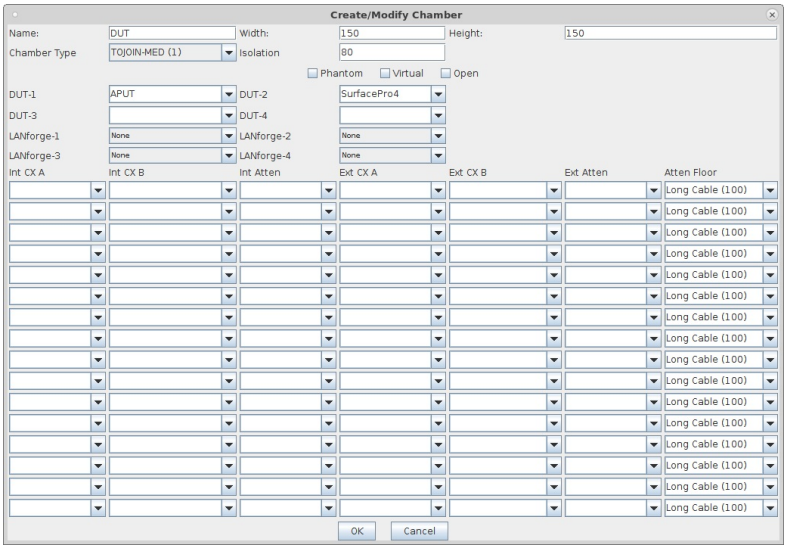
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber-View to create various objects.



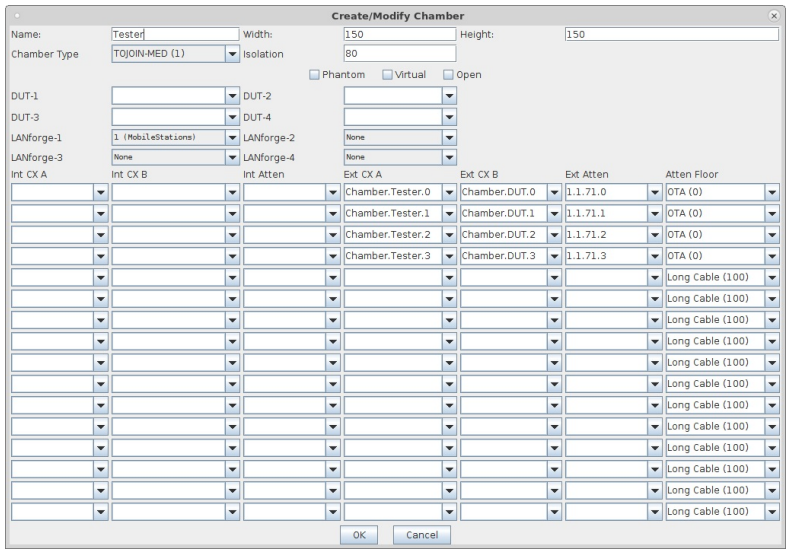
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



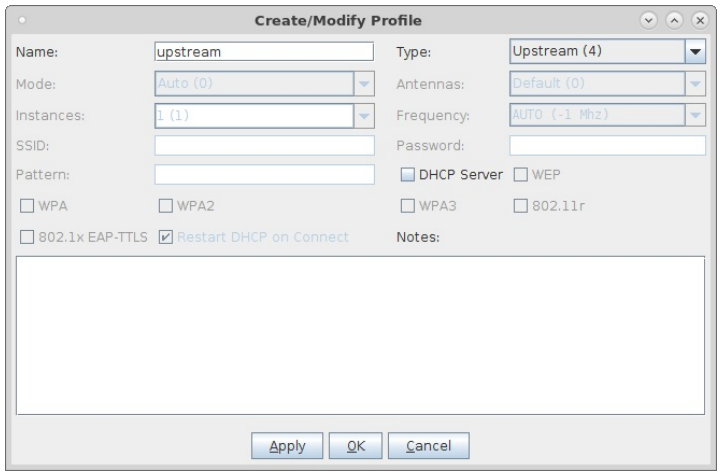
C. Create a chamber object to hold the DUT, and add the DUT to that chamber. If you have no chambers, you can create a fake chamber, but your test will not be isolated and may not function as desired.



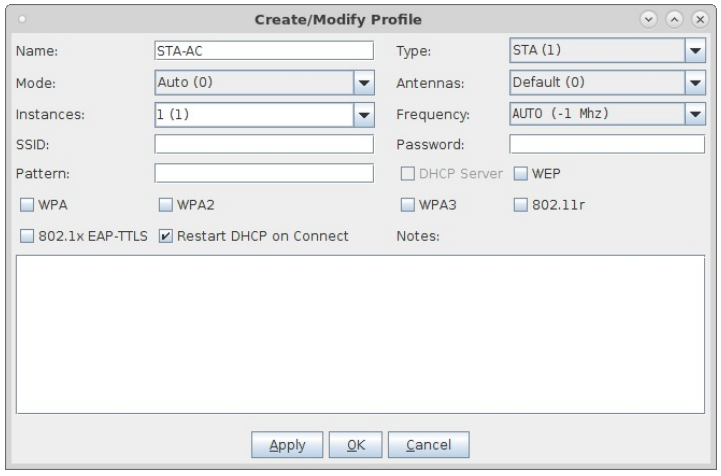
D. Create a chamber object to hold the LANforge system, and add the LANforge to it. Add connections from this chamber to the DUT chamber, specifying the proper Attenuator modules. Please view our other cookbook on setting up attenuator connections in LANforge.



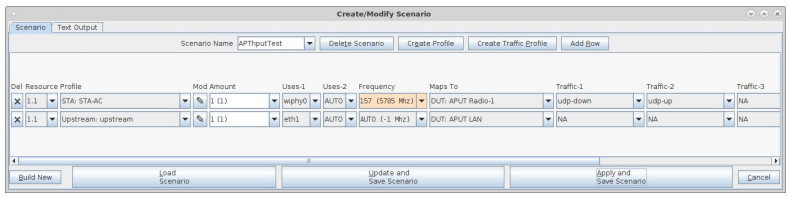
E. Configure an Upstream profile using eth1 on the LANforge system.



F. Configure an STA profile on the LANforge system.

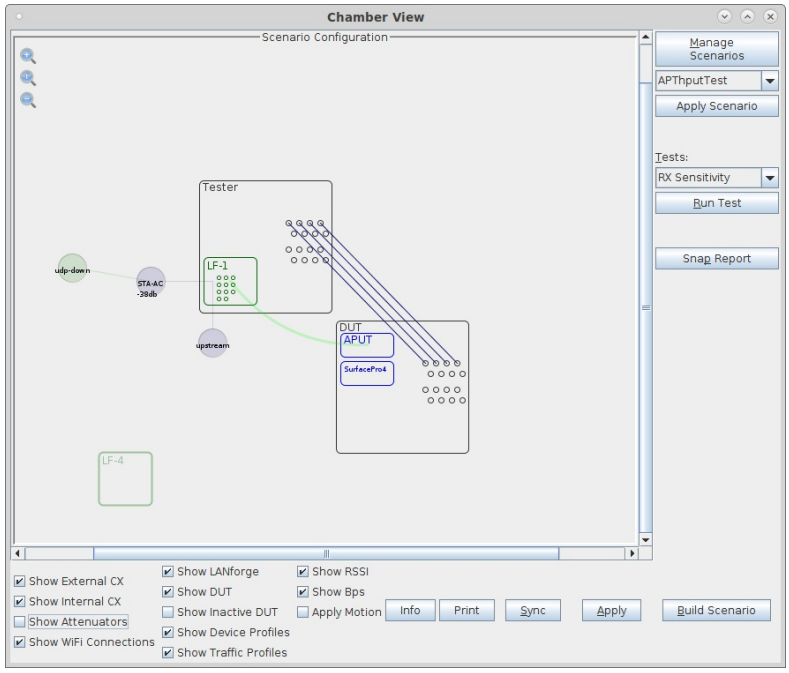


G. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).

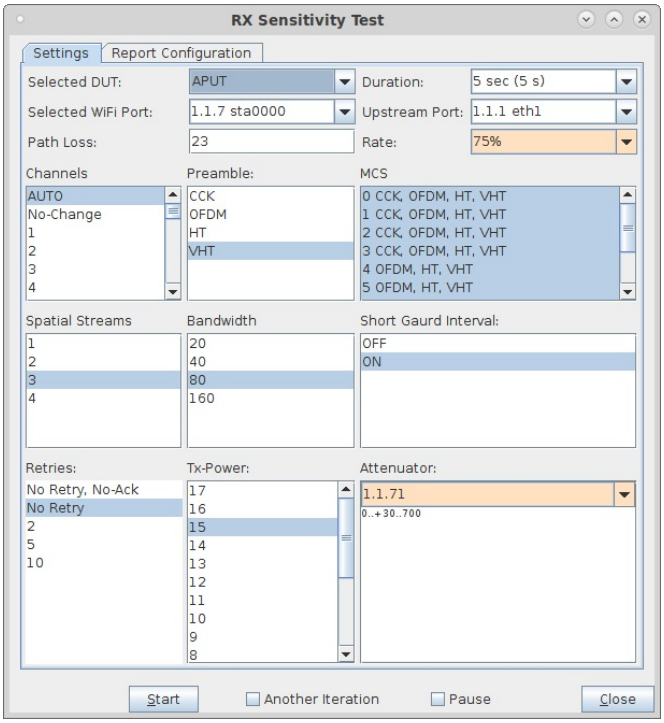


2. Use Chamber View to run a Receiver Sensitivity test.

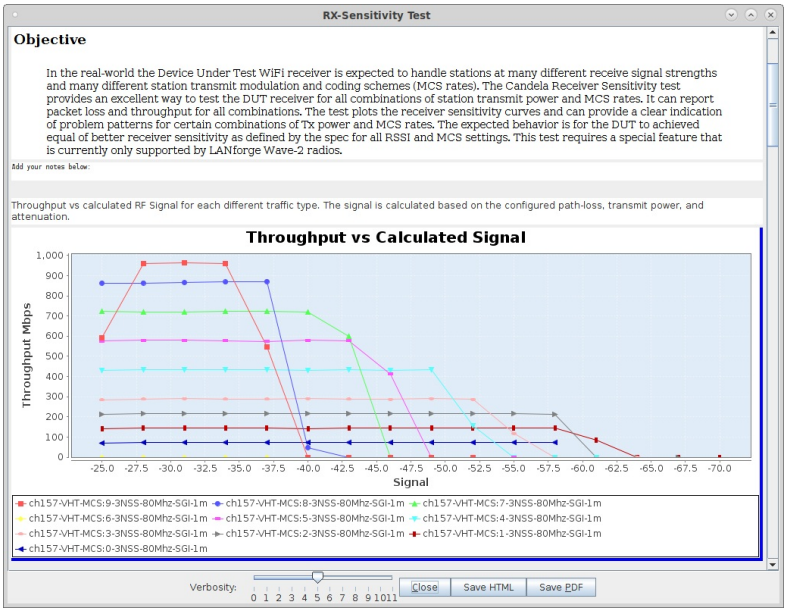
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario or create a new scenario as needed. Apply the Scenario, then Build the scenario.



B. Select the **RX Sensitivity** test and click **Run Test**. You should see the RX Sensitivity Test configuration window pop up. It will remember the last configuration for most fields. Select the DUT and WiFi station device, and select the combinations of traffic types you wish to send.:



C. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs.

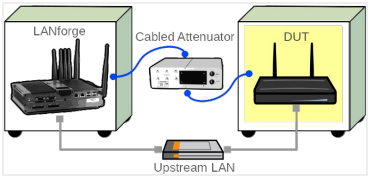


D. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example RX Sensitivity Report

Testing AP Dataplane throughput at different packet sizes

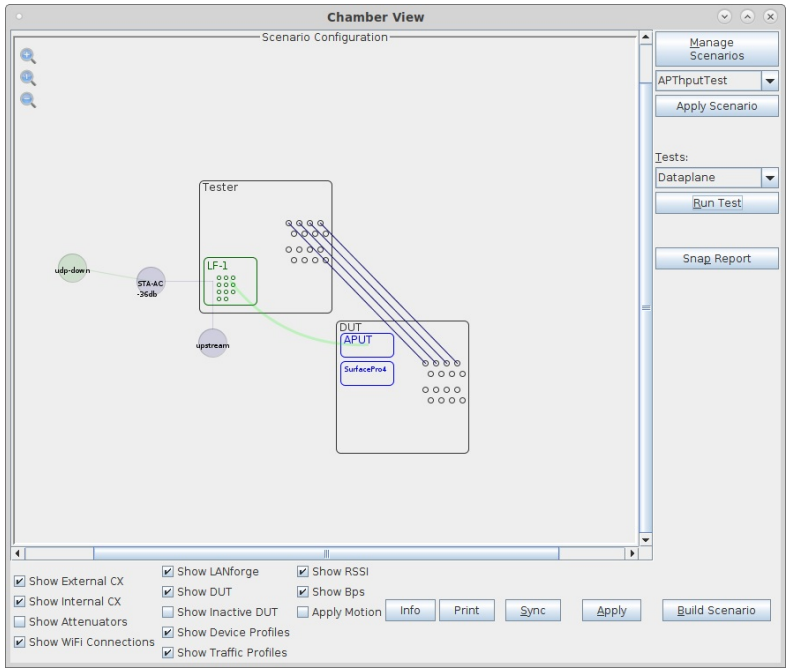
Goal: Setup and run a Dataplane test for an AP using the LANforge CT523c or similar system in order to test how well the AP can handle sending and receiving packets with different packet sizes.

In this test scenario, the LANforge CT523c is used to generate packets of different sizes in the upstream and downstream direction through an AP. This example assumes you have some experience with Chamber View, and that you have a LANforge system, a programmable attenuator like the CT704b and two isolation chambers like the CT820a. The AP should be in one chamber, the LANforge system is in the other chamber, and the attenuator is cabled between them. In this example, the Attenuator is left at an optimal configuration, but you can also use this same Dataplane test to generate a report at different RF signal levels using the Attenuator. This feature requires LANforge version 5.3.9 or higher.

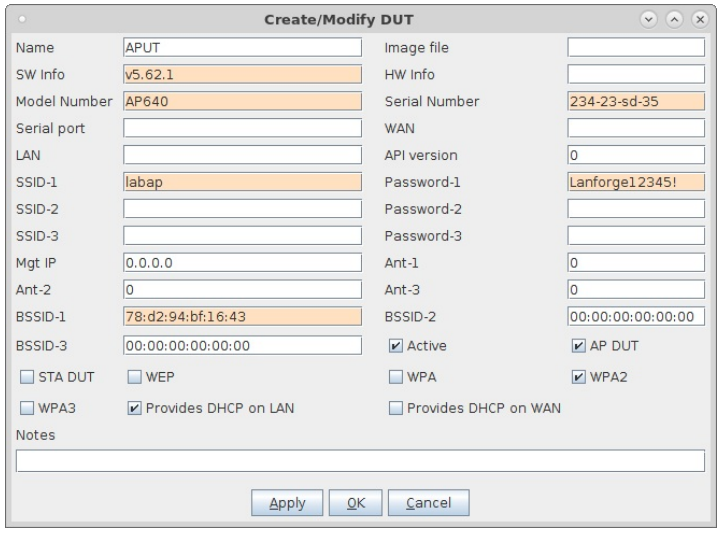


1. Configure Chamber View for Dataplane and Similar Tests.

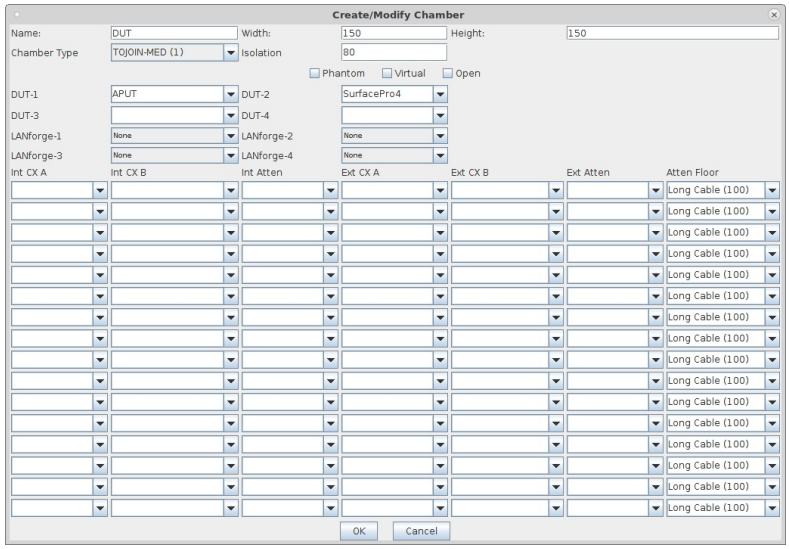
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects.



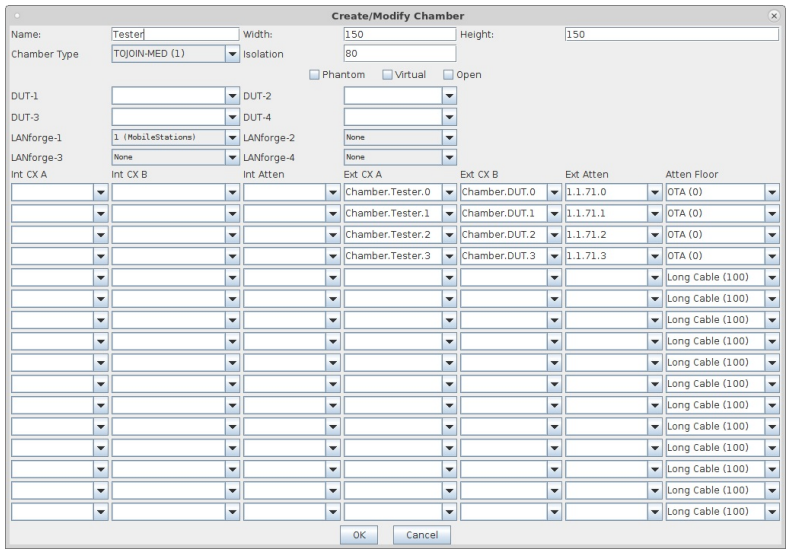
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



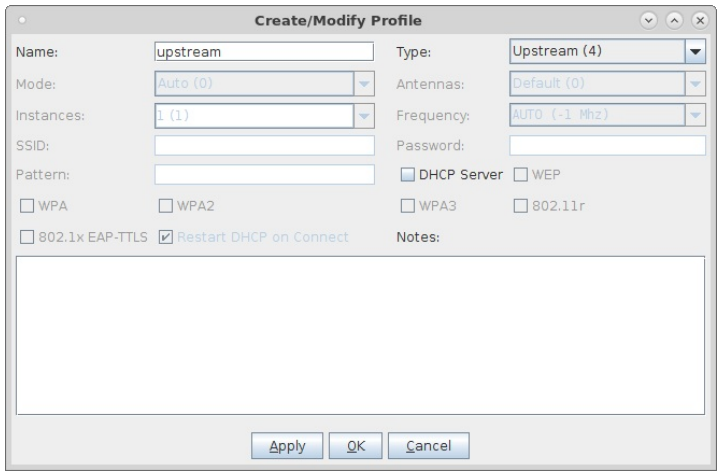
C. Create a chamber object to hold the DUT, and add the DUT to that chamber. If you have no chambers, you can create a fake chamber, but your test will not be isolated and may not function as desired.



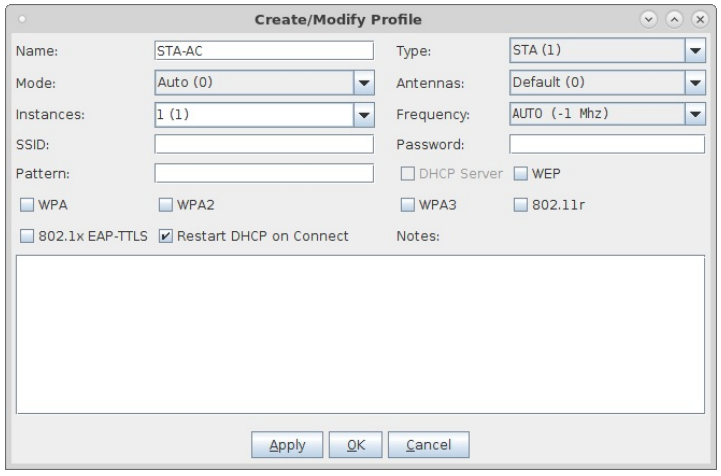
D. Create a chamber object to hold the LANforge system, and add the LANforge to it. Add connections from this chamber to the DUT chamber, specifying the proper Attenuator modules. Please view our other cookbook on setting up attenuator connections in LANforge.



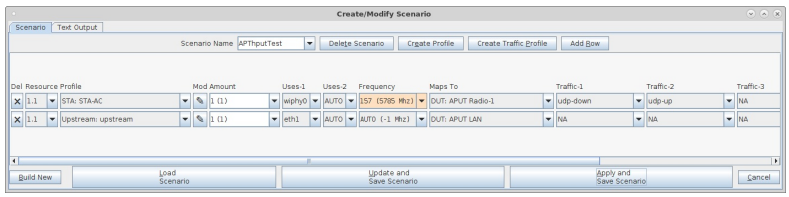
E. Configure an Upstream profile using eth1 on the LANforge system.



F. Configure an STA profile on the LANforge system.

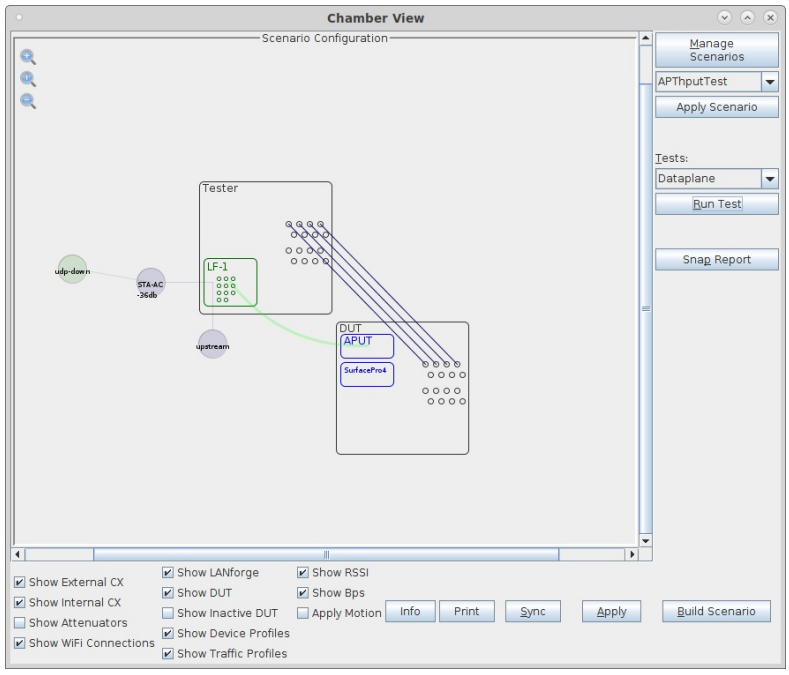


G. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).

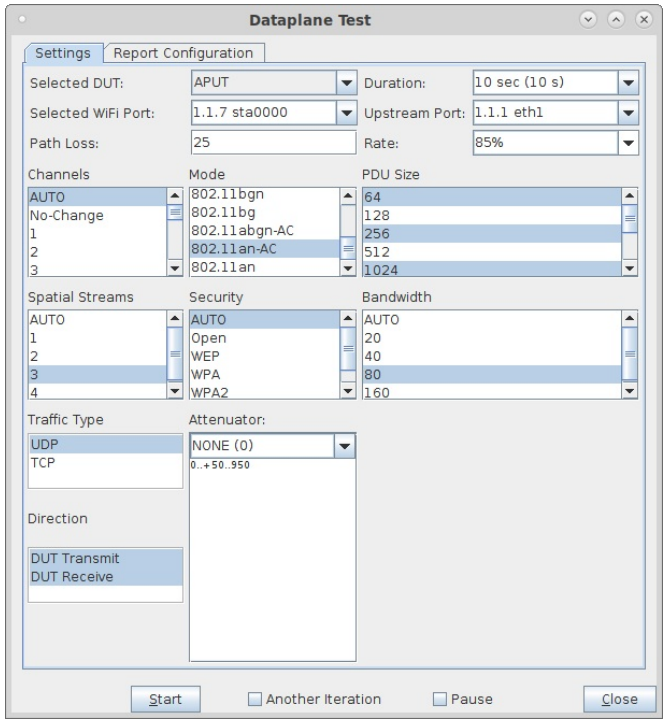


2. Use Chamber View for Dataplane test.

A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario. Apply the Scenario, then Build the scenario.



B. Select the **Dataplane** test and click **Run Test**. You should see the Dataplane Test configuration window pop up. It will remember the last configuration for most fields. Select the DUT and WiFi station device, and select the combinations of traffic types and packet sizes you wish to send.:



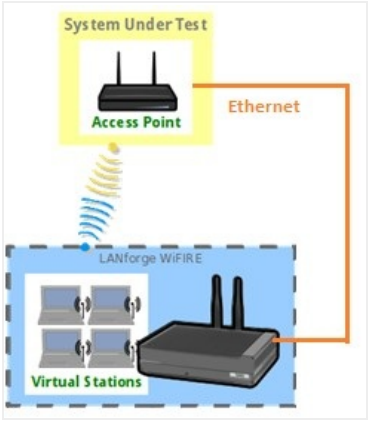
C. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs.

D. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example Dataplane Report.

Testing AP Dataplane throughput at different orientation

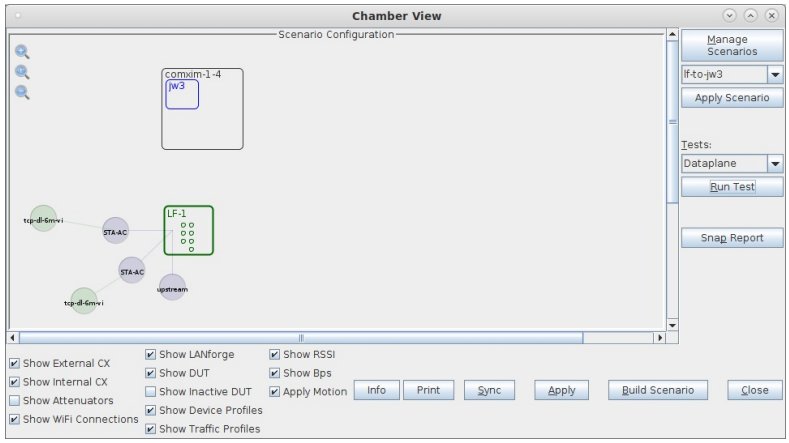
Goal: Setup and run a Dataplane test for an AP using the LANforge CT522 or similar system in order to test how well the AP can handle sending and receiving packets at different rotations.

In this test scenario, the LANforge CT522 is used to generate packets in the upstream and downstream direction through an AP at different AP orientations. An affordable stand-alone turn-table is used to automatically rotate the AP to the desired orientation. This example assumes you have some experience with Chamber View, and that you have a LANforge system and turn-table. Using chambers will make the test perform more consistently, but is not required for this test. This feature requires LANforge version 5.4.1 or higher.

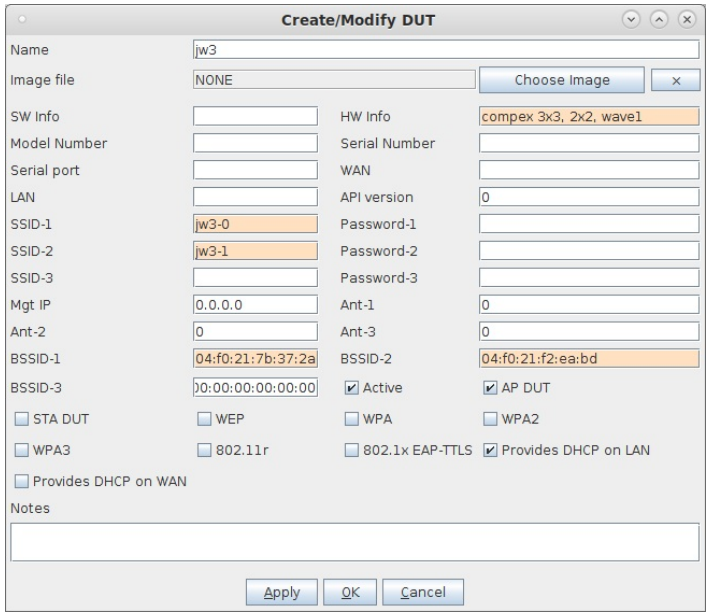


1. Configure Chamber View for Dataplane and Similar Tests.

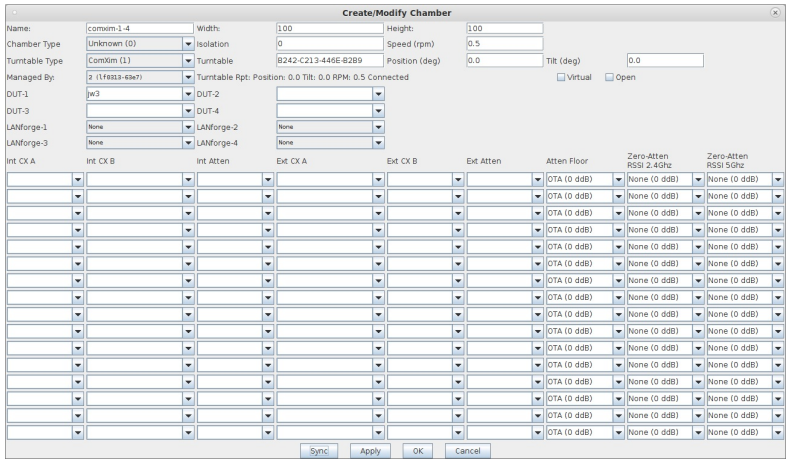
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects.



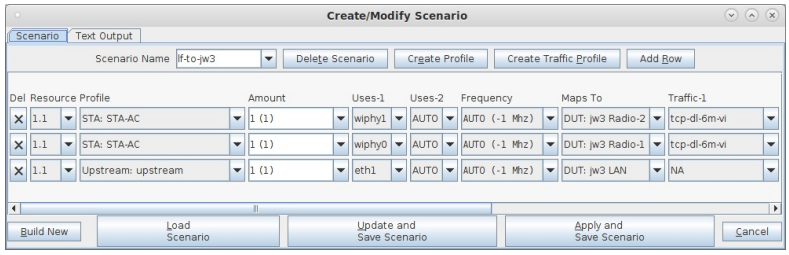
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



C. If using the 'comxim' stand-alone turn-table, the chamber object should automatically be created. For other real chambers, you may have to create the chamber object and configure it to be able to communicate to the chamber control API. This example uses the stand-alone turntable configured with a fake chamber object:

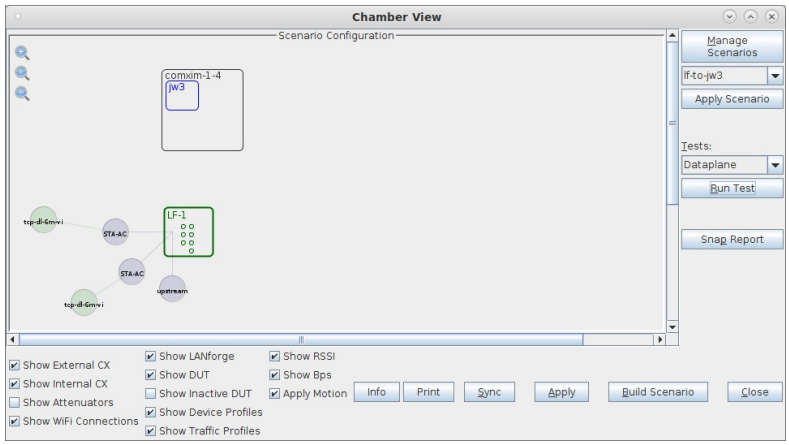


D. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).

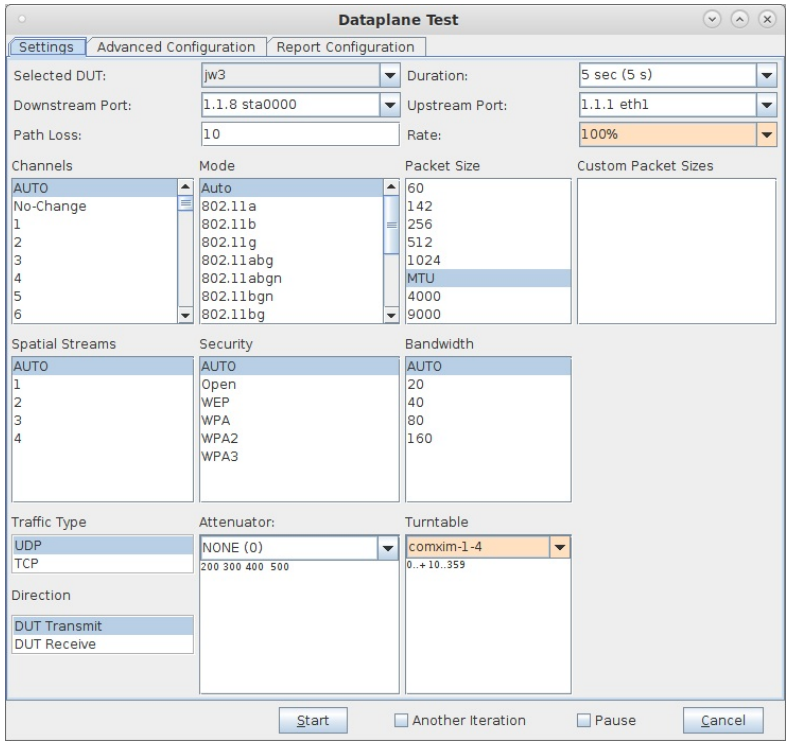


2. Use Chamber View for Dataplane test.

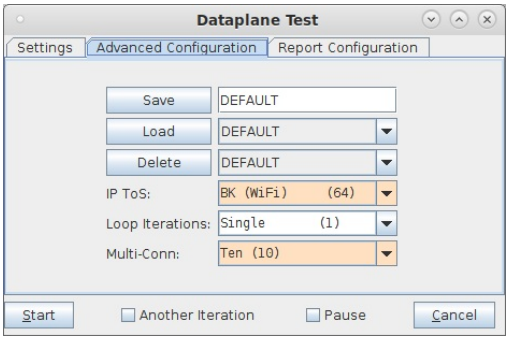
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario. Apply the Scenario, then Build the scenario.



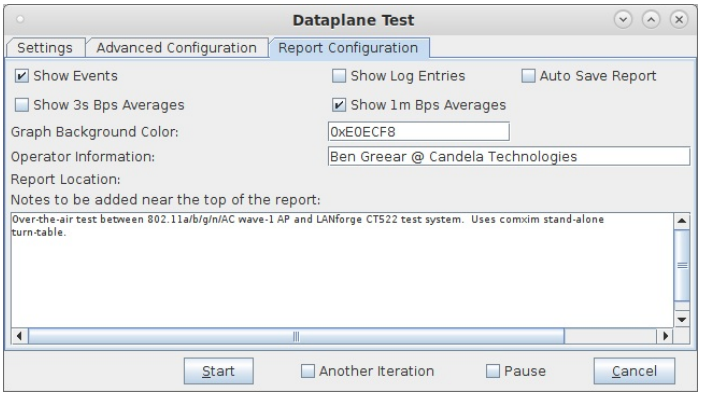
B. Select the **Dataplane** test and click **Run Test**. You should see the Dataplane Test configuration window pop up. It will remember the last configuration for most fields. Select the DUT and WiFi station device, and select the combinations of traffic types and turn-table degrees that you wish to test. The degrees show in this image will go from zero to 359 in steps of 10 degrees. The mouse-over tooltip for the turntable configuration entry field has the details on the available syntax.:



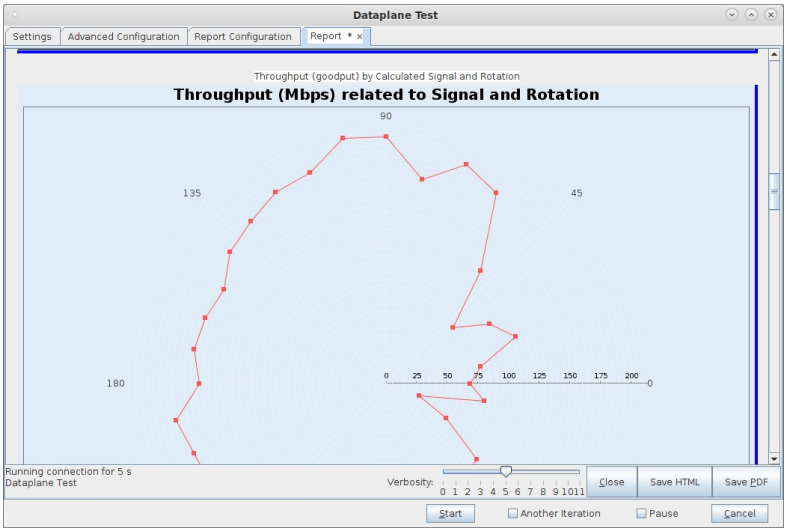
C. You may wish to save/restore configurations or make some advanced configuration on the 'Advanced Configuration' tab.



D. The Report Configuration tab lets you input the operator information, notes about this test setup, and more.



E. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs.



F. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example Dataplane Rotation Test Report.

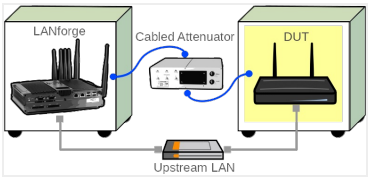
3. Stand-alone Turn Table Information.

A. We built a cable stand for the stand-alone turn-table to help keep cables out of the way while rotating.

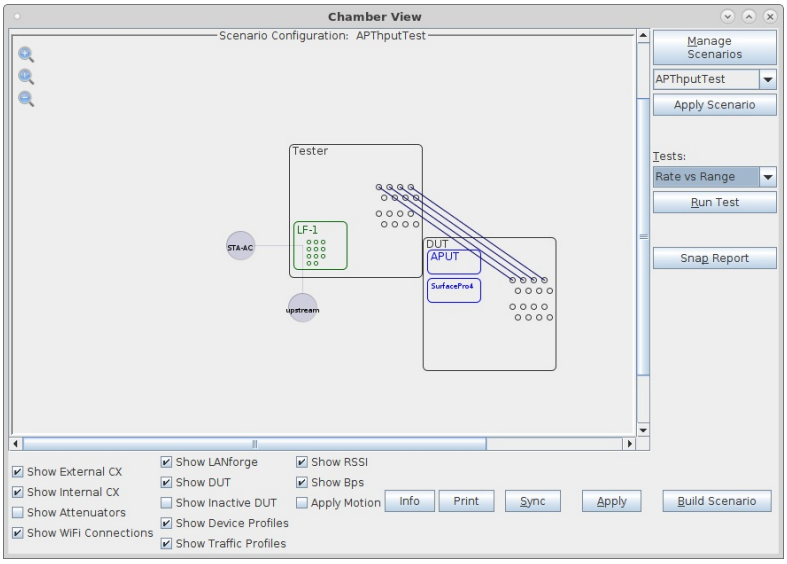
Testing Rate vs Range throughput for a WiFi Device

Goal: Setup and run a Rate vs Range test for an AP using the LANforge CT523c or similar system in order to test how well the AP can transmit packets at different signal levels. This is a good test of the AP's rate-control logic, as well as tx power and general ability to deal with various RF conditions. This emulates a throughput test as the user walks away from the AP.

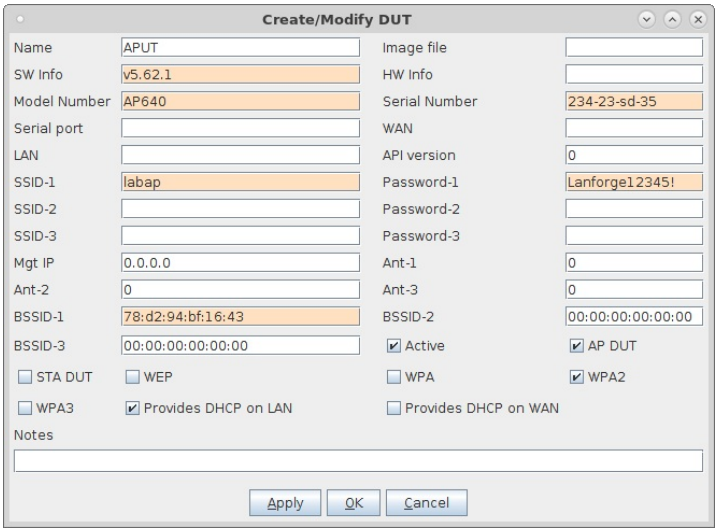
In this test scenario, the LANforge CT523c is used to generate packets on the Ethernet port towards the wired side of the AP. The AP will then transmit the frames to the LANforge WiFi station. This example assumes you have some experience with Chamber View, and that you have a LANforge system, a programmable attenuator like the CT704b and two isolation chambers like the CT820a. The AP should be in one chamber, the LANforge system is in the other chamber, and the Attenuator is cabled between them. This feature is in LANforge version 5.3.9 and higher.



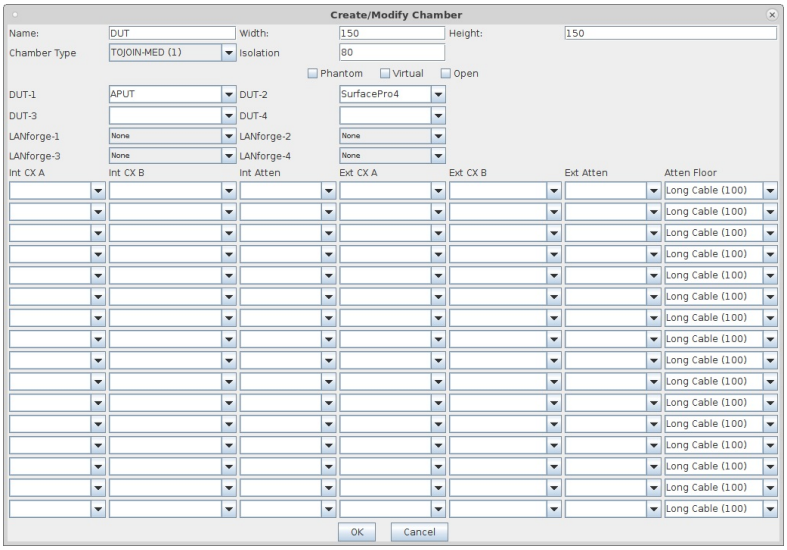
1. Configure Chamber View for Rate vs Range and Similar Tests.
    A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects.



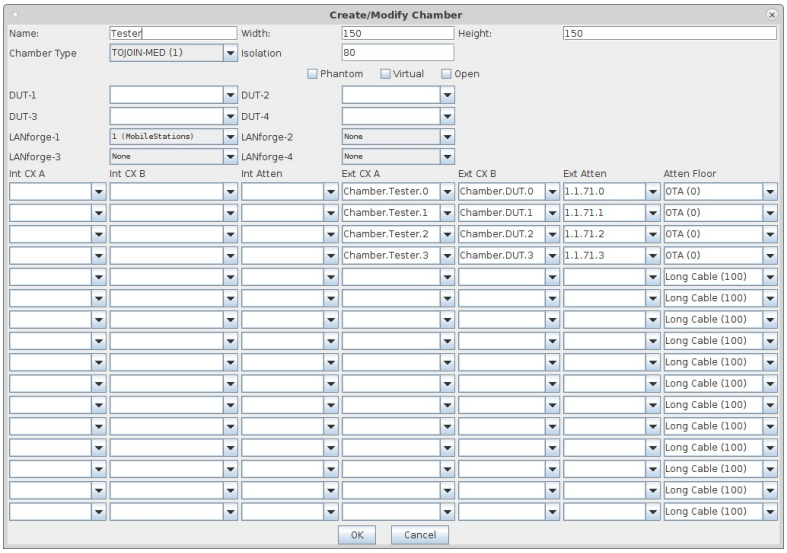
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



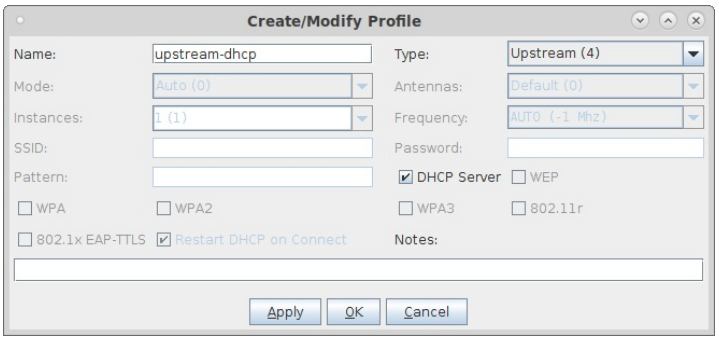
C. Create a chamber object to hold the DUT, and add the DUT to that chamber. If you have no chambers, you can create a fake chamber, but your test will not be isolated and may not function as desired.



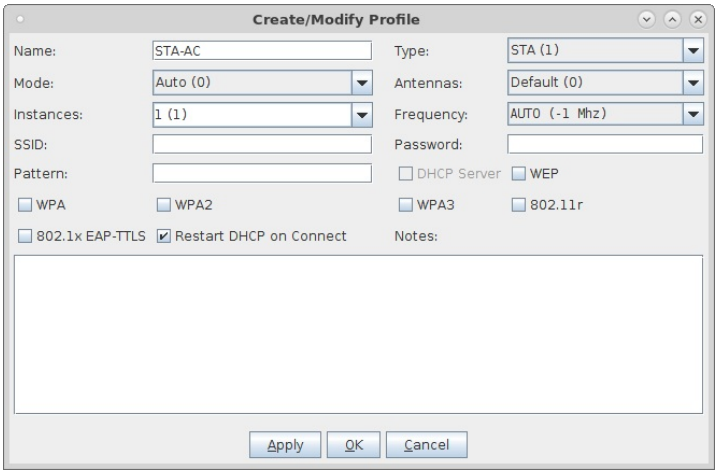
D. Create a chamber object to hold the LANforge system, and add the LANforge to it. Add connections from this chamber to the DUT chamber, specifying the proper Attenuator modules. Please view our other cookbook on setting up attenuator connections in LANforge.



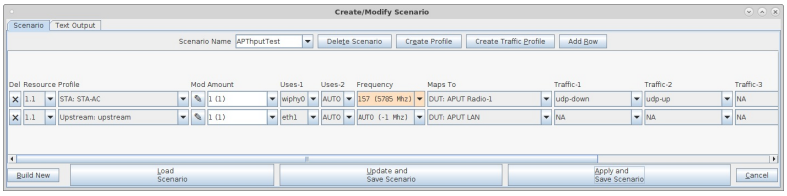
E. Configure an Upstream profile using eth1 on the LANforge system.



F. Configure an STA profile on the LANforge system.

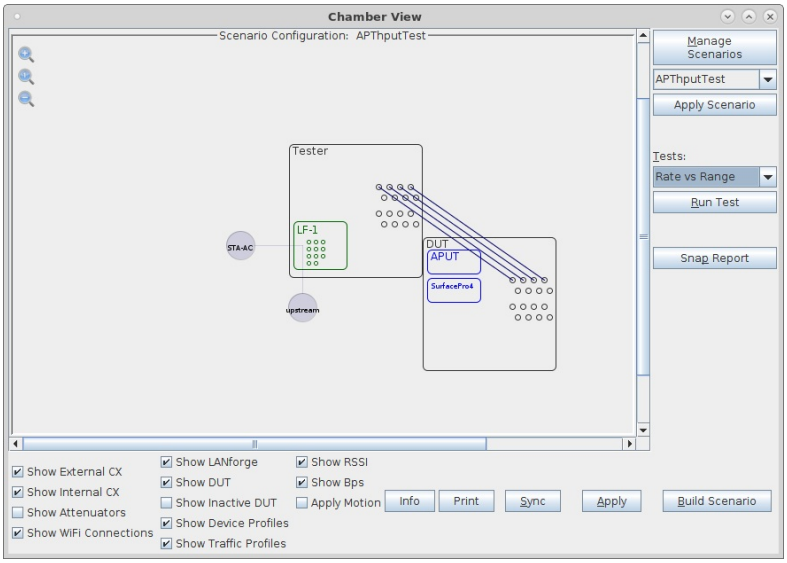


G. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).

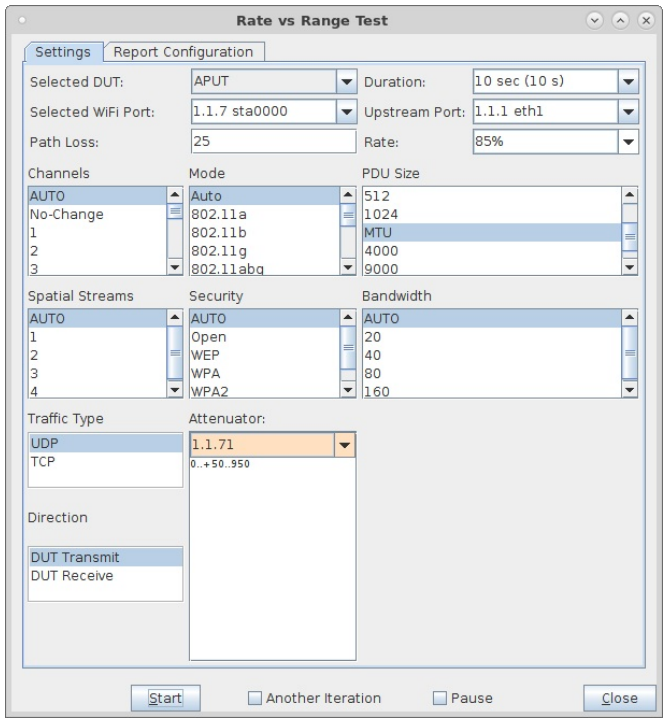


2. Use Chamber View to run a Rate vs Range test.

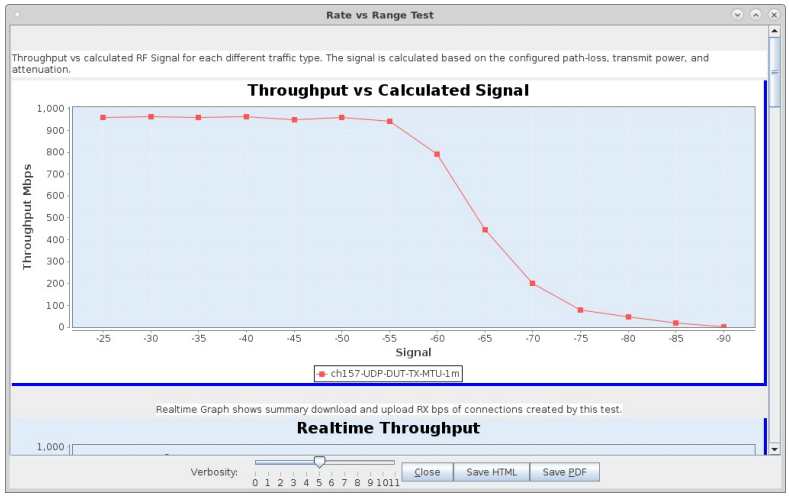
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario or create a new scenario as needed. Apply the Scenario, then Build the scenario.



B. Select the **Rate vs Range** test and click **Run Test**. You should see the Rate vs Range Test configuration window pop up. It will remember the last configuration for most fields. Select the DUT and WiFi station device, and select the combinations of traffic types you wish to send. Be sure to select the attenuator and configure the attenuation steps. In this case, we have antenna over-the-air connection inside one of the chambers, with total path-loss at zero attenuation of about 25db. One interesting way to use this tool is to select the 'Another Iteration' checkbox. When the curren test is complete, you will see a popup message notifying completeness. You can then reconfigure the DUT (by changing firmware versions, or some other configuration), and then re-run the test. The second test will be displayed on the same graphs, so it is easy to compare the difference. This particular test is not using that feature, however:



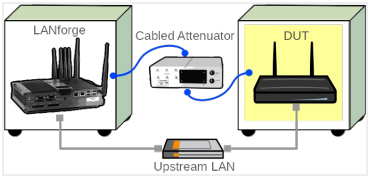
C. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs.



D. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example Rate vs Range Report

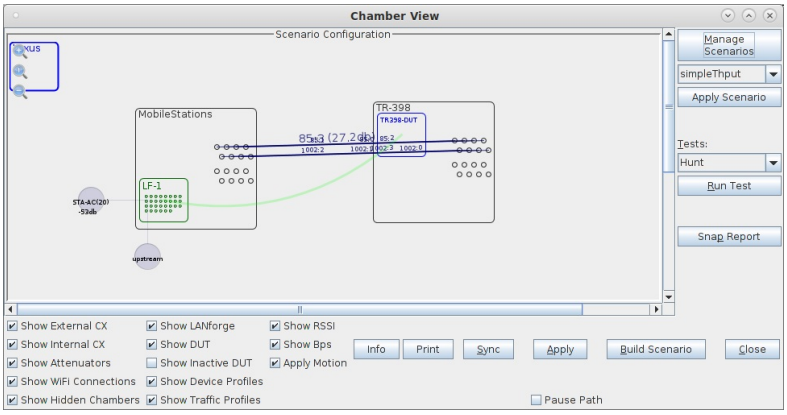
Testing Latency at different packet sizes using the Hunt test

Goal: Setup and run a Hunt test for an AP using the LANforge CT523c or similar system in order to test Latency at the highest throughput rate supported by the DUT.

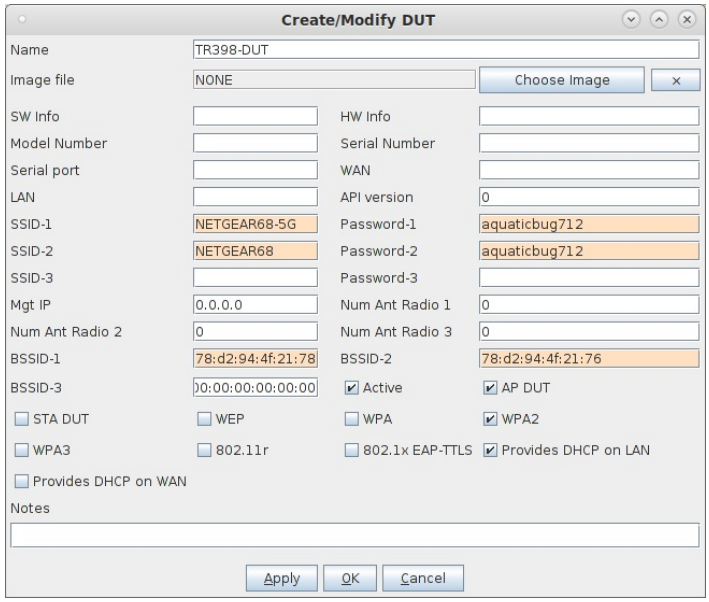


In this test scenario, the LANforge CT523c is used to generate packets of different sizes in the downstream direction through an AP. The test hunts to find the best throughput, and then re-runs the traffic test configured to a percentage of the best actual throughput. Latency graphs show how well the DUT performs. This guide assumes you have some experience with Chamber View, and that you have a LANforge system, a programmable attenuator like the CT704b and two isolation chambers like the CT820a. The AP should be in one chamber, the LANforge system is in the other chamber, and the attenuator is cabled between them. In this example, the Attenuator is left at an optimal configuration, but you can also use this same Hunt test to generate a report at different RF signal levels using the Attenuator. This feature requires LANforge version 5.4.2 or higher.

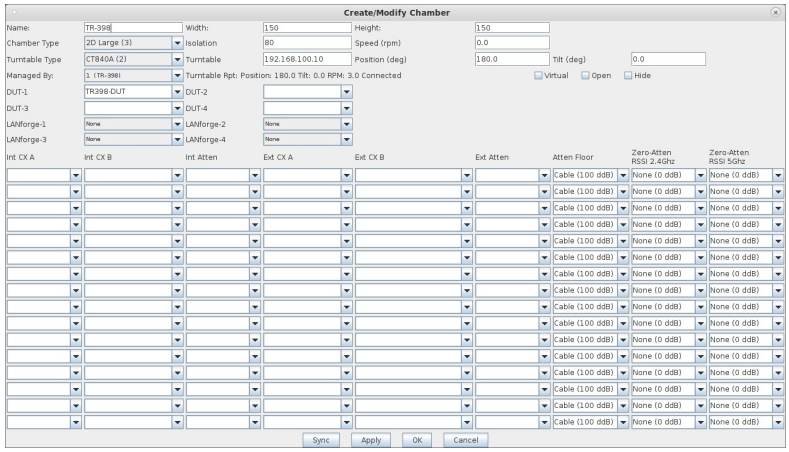
1. Configure Chamber View for Hunt and Similar Tests.
    A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects.



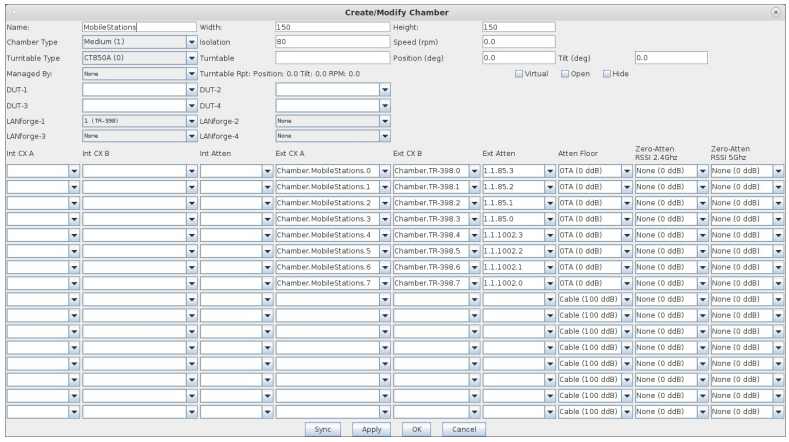
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



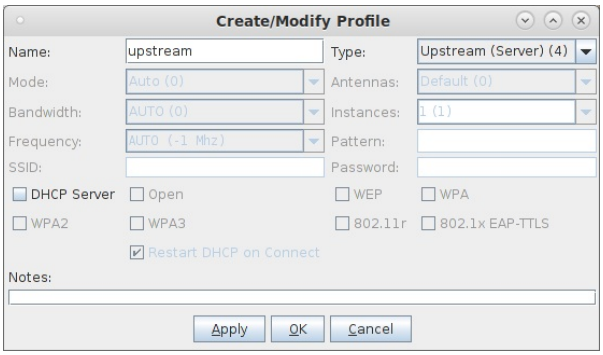
C. Create a chamber object to hold the DUT, and add the DUT to that chamber. If you have no chambers, you can create a fake chamber, but your test will not be isolated and may not function as desired.



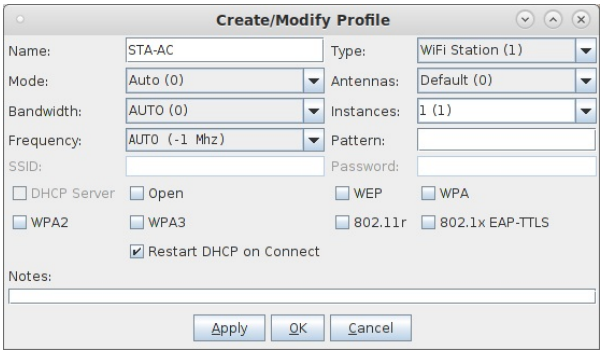
D. Create a chamber object to hold the LANforge system, and add the LANforge to it. Add connections from this chamber to the DUT chamber, specifying the proper Attenuator modules. Please view our other cookbook on setting up attenuator connections in LANforge.



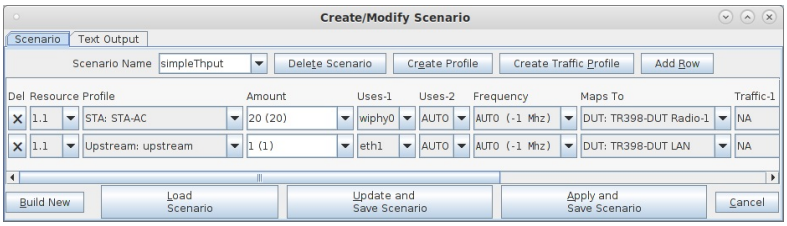
E. Configure an Upstream profile using eth1 on the LANforge system.



F. Configure an STA profile on the LANforge system.

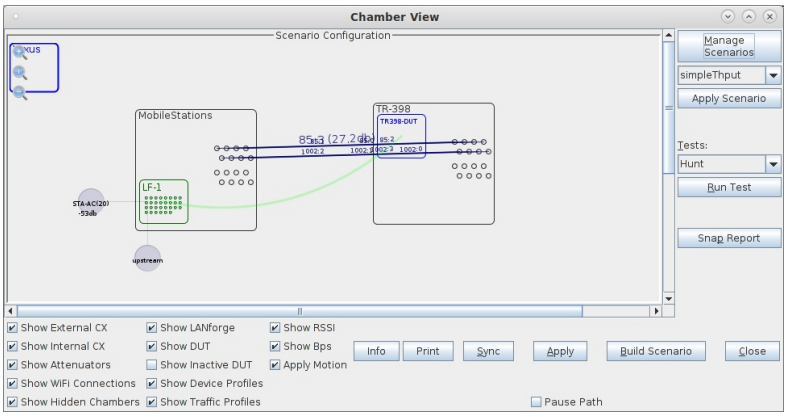


G. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).

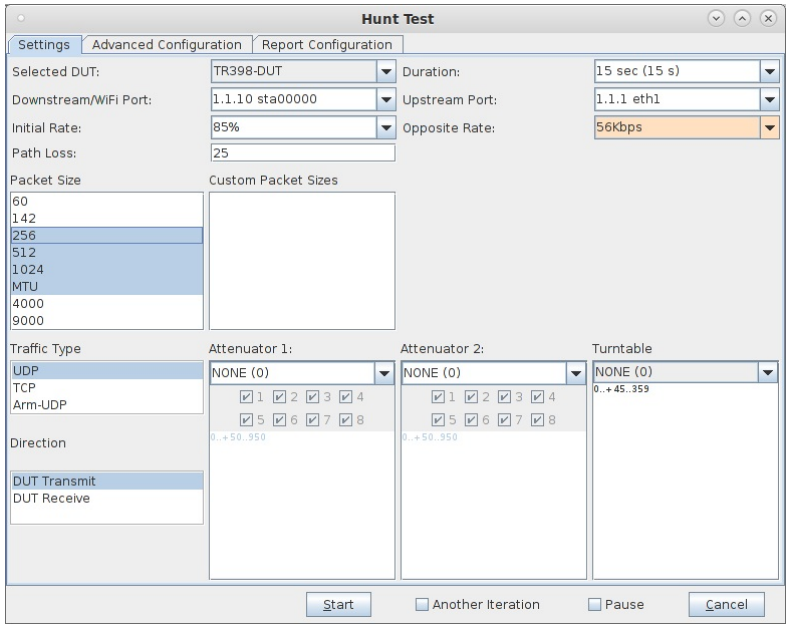


2. Use Chamber View for Hunt test.

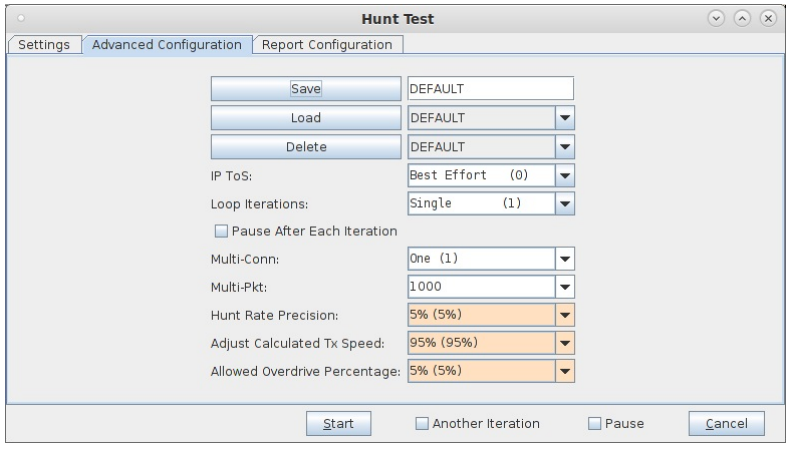
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario. Apply the Scenario, then Build the scenario.



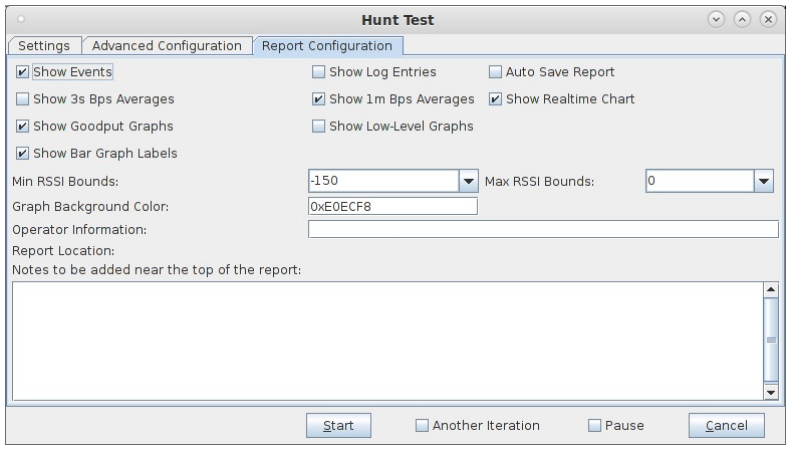
B. Select the **Hunt** test and click **Run Test**. You should see the Hunt Test configuration window pop up. It will remember the last configuration for most fields. Select the DUT and WiFi station device, and select the combinations of traffic types and packet sizes you wish to send.



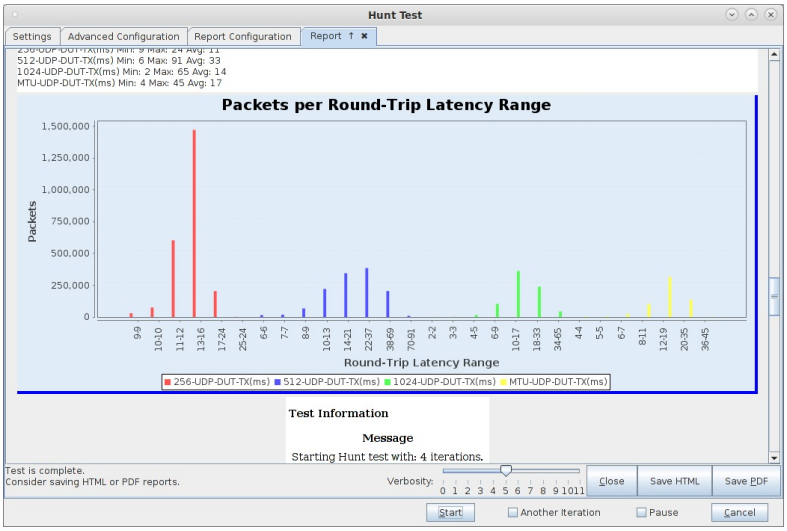
C. The Advanced Configuration screen can tune the hunt criteria, save/restore configuration and other options.



D. The Report Configuration screen can tune how the reports are generated.



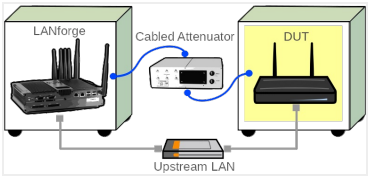
E. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs.



F. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example Hunt Report.

Testing Station Capacity and throughput for a WiFi Device

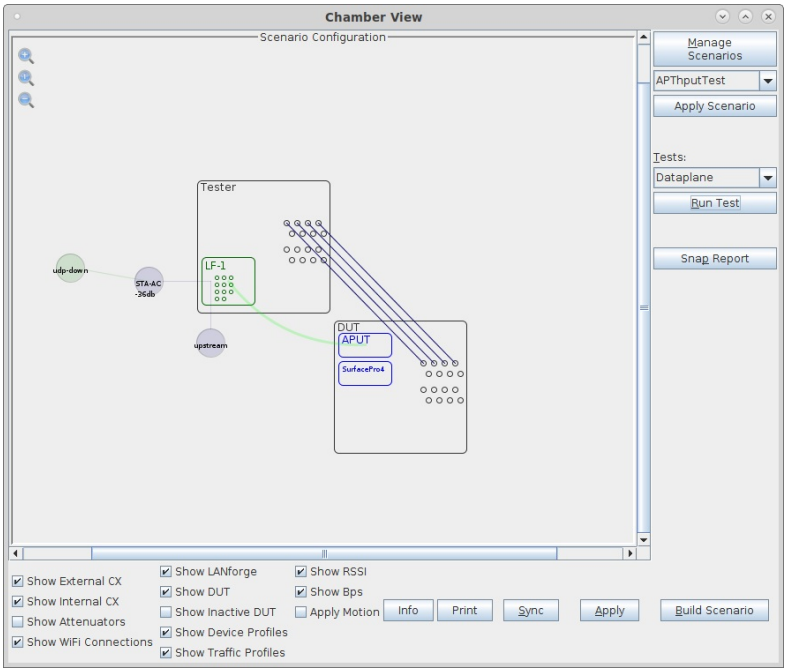
Goal: Setup and run a WiFi Capacity Test test for an AP using the LANforge CT523c or similar system in order to test how well the AP can handle different amounts of active stations. This is a good test of the AP's scalability and stability.



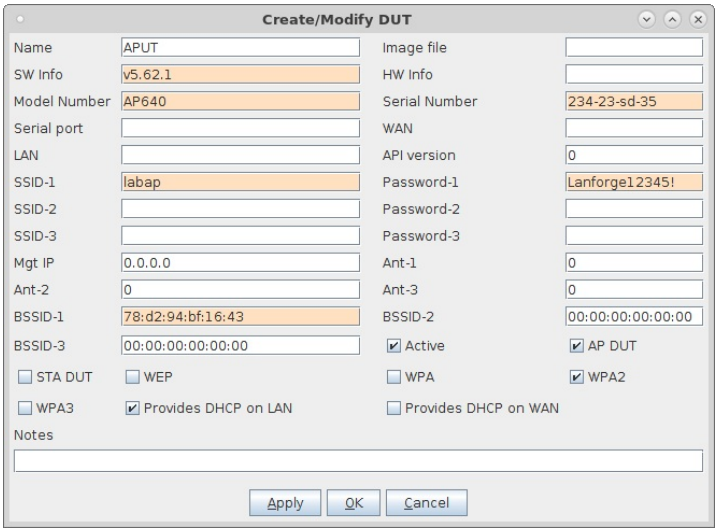
In this test scenario, the LANforge CT523c is used to create 64 stations and generate packets on the Ethernet port towards the wired side of the AP. The AP will then transmit the frames to the LANforge WiFi stations. LANforge will bring up stations in configured amounts, run a throughput test, and then bring up the next set of stations and repeat until the test is complete. This example assumes you have some experience with Chamber View, and that you have a LANforge system and two isolation chambers like the CT820a. The AP should be in one chamber, the LANforge system is in the other chamber, and the Attenuator (if using one) is cabled between them. Set the Attenuator to desired state before starting the test. Chambers and attenuators are not required for this test, but running inside isolation chambers will usually give you better and more reliable test results. This feature is in LANforge version 5.3.9 and higher.

1. Configure Chamber View for WiFi Capacity Test and Similar Tests.

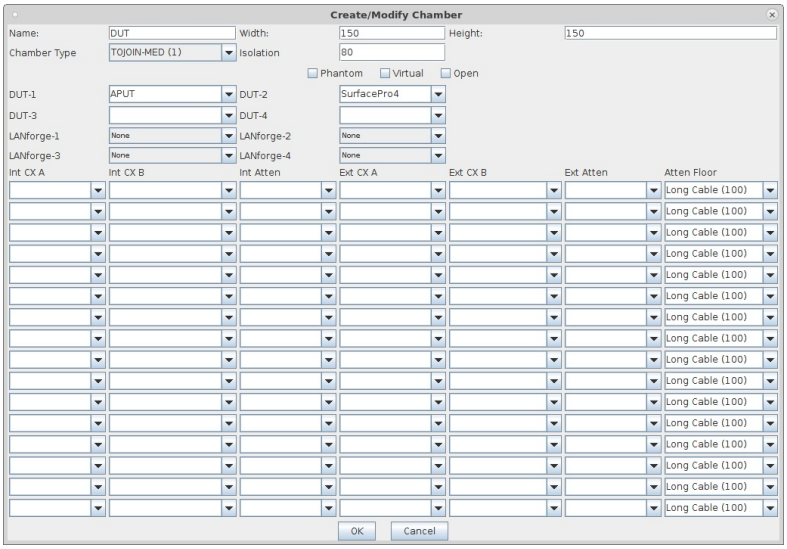
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects.



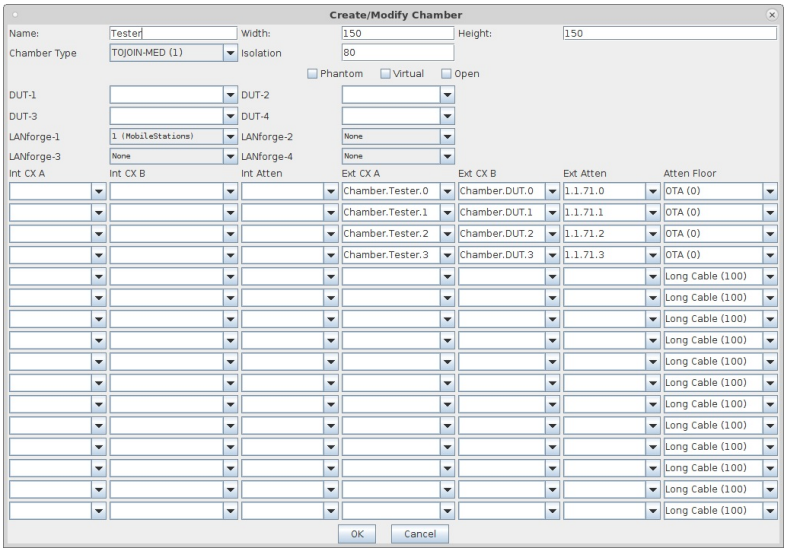
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



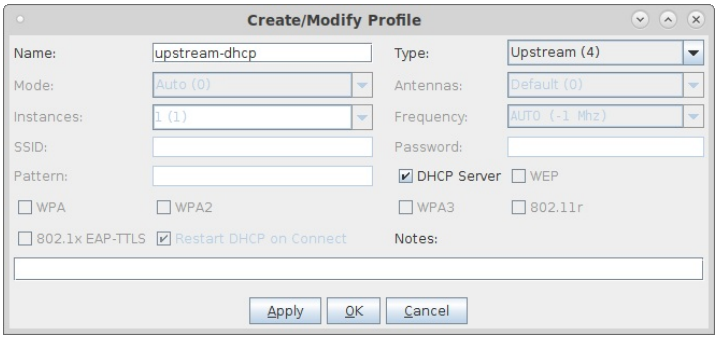
C. Create a chamber object to hold the DUT, and add the DUT to that chamber. If you have no chambers, you can create a fake chamber, but your test will not be isolated and may have decreased performance due to outside RF interference.



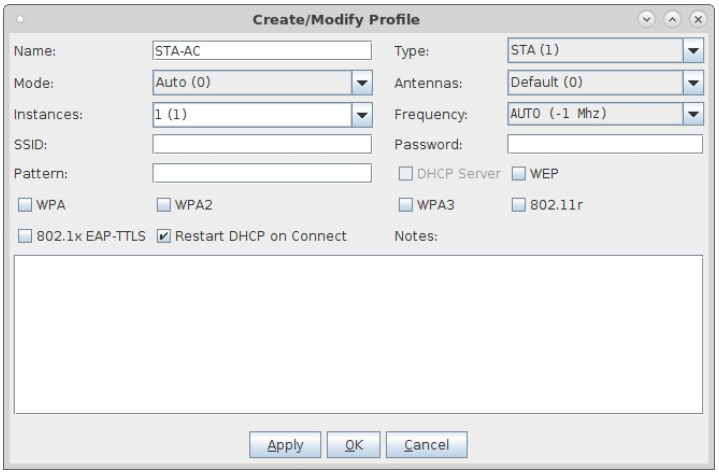
D. Create a chamber object to hold the LANforge system, and add the LANforge to it. Add connections from this chamber to the DUT chamber, specifying the proper Attenuator modules. Please view our other cookbook on setting up attenuator connections in LANforge.



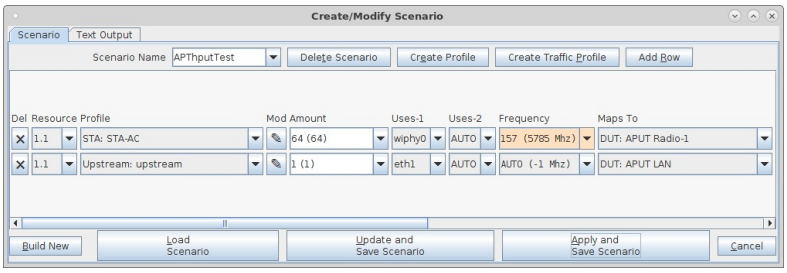
E. Configure an Upstream profile using eth1 on the LANforge system. Notice the dhcp-server checkbox is selected, since in this test case our AP is not configured as a DHCP server.



F. Configure an STA profile on the LANforge system.

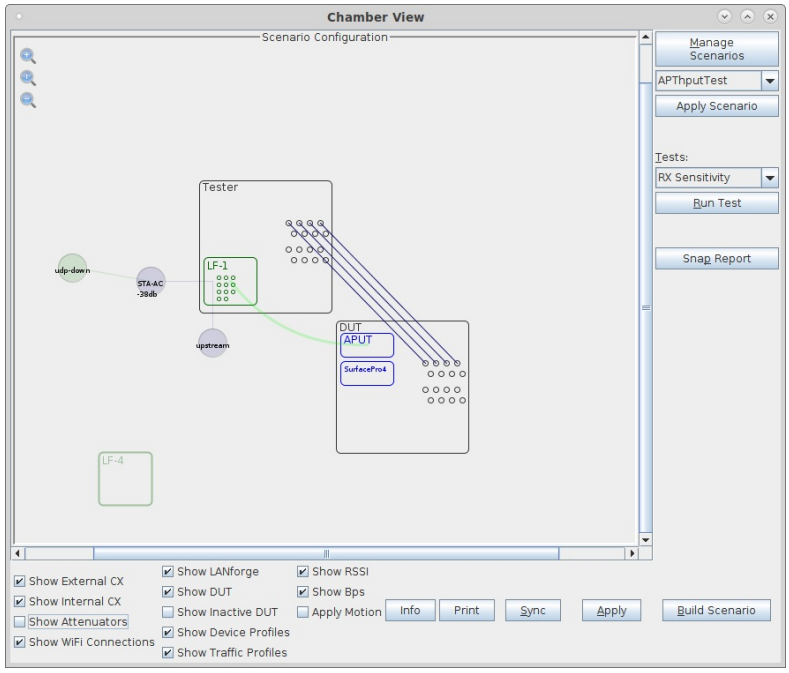


G. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT). Choose the appropriate quantity of stations for your test case.

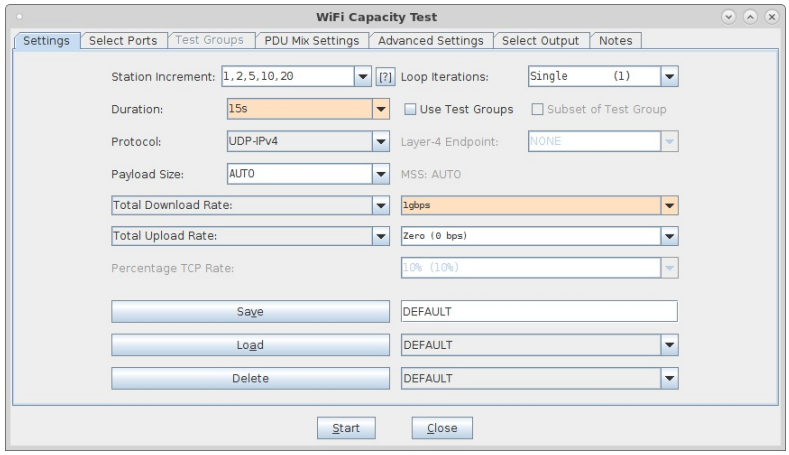


2. Use Chamber View to run a WiFi Capacity Test.

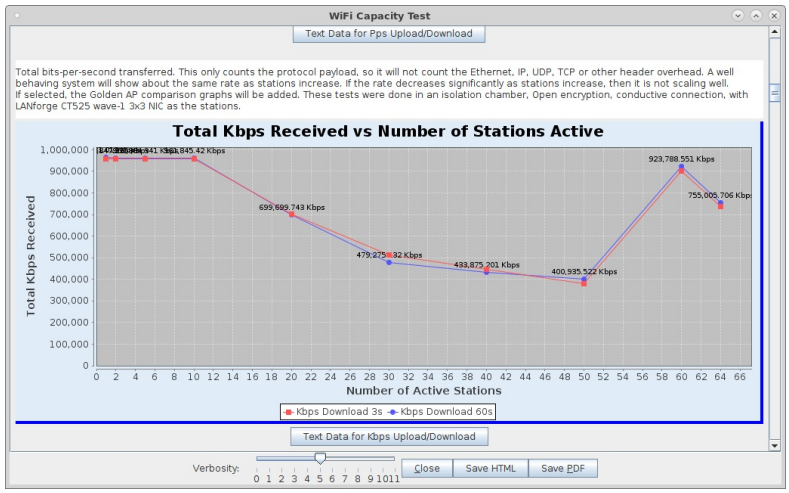
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario or create a new scenario as needed. Apply the Scenario, then Build the scenario.



B. Select the **WiFi Capacity** test and click **Run Test**. You should see the WiFi Capcity Test configuration window pop up. The Ports are normally selected properly based on the Scenario, but you may need to adjust them depending on your goals and scenario. Select the desired throughput and any other configuration changes from the defaults:



C. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs. In this particular test, the results are a it unexpected. Normally rate starts off high and slowly decreases after 5 or 10 stations are active. In this case, we see a falloff after 10, but then it recovers at 60.

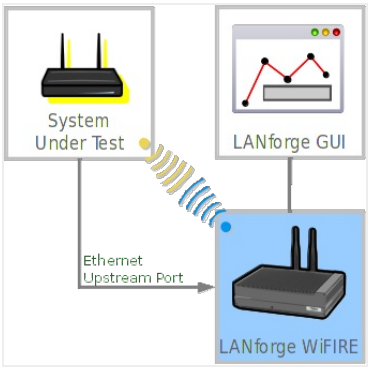


D. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example WiFi Capacity Test Report.

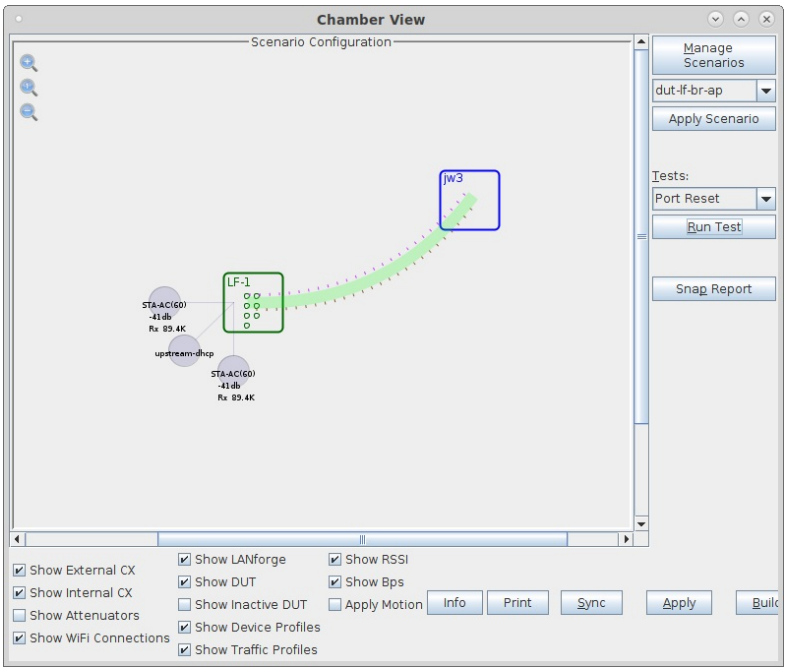
Testing Station Associate and Disassociate for a WiFi Device

Goal: Setup and run a Port Reset test for an AP using the LANforge CT523c or similar system in order to test how well the AP can handle stations connecting and disconnecting many times. This is a good test of the AP's management plane stability.

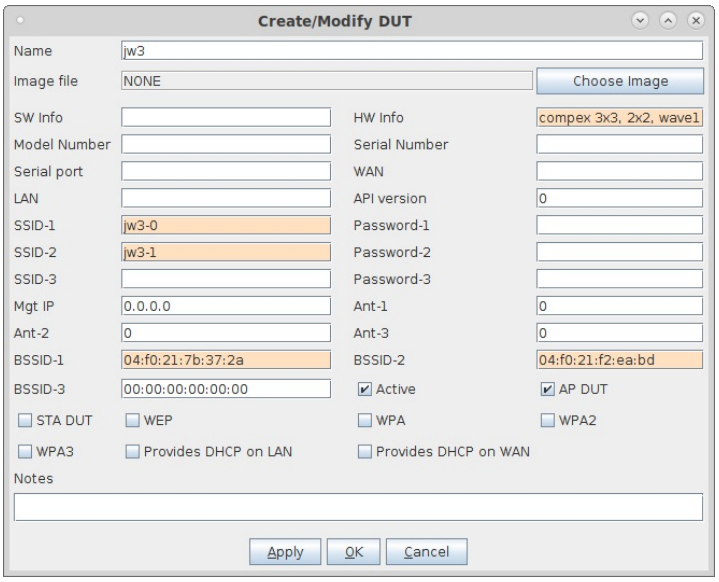
In this test scenario, the LANforge CT522 is used to create 120 stations and then have them connect and disconnect to the AP. The test will count the number of connections and related events. This example assumes you have some experience with Chamber View, and that you have a LANforge system. A programmable attenuator and two isolation chambers would add the ability to test station reconnects at different RF signal levels, but this test normally runs fine withour chambers or attenuators. This feature is in LANforge version 5.3.9 and higher.



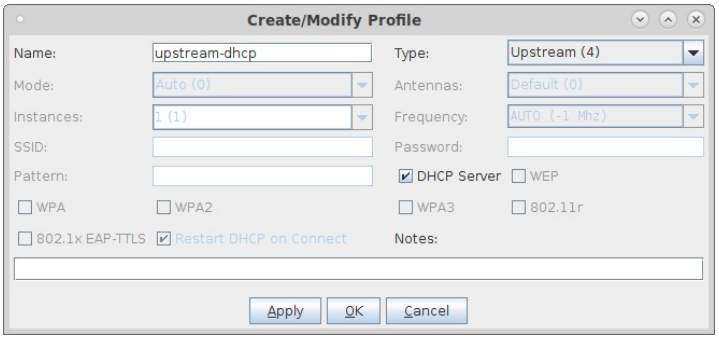
1. Configure Chamber View for Port Reset and Similar Tests.
    A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects. If you do not have chambers or attenuators, just create the DUT object and skip the chamber setup.



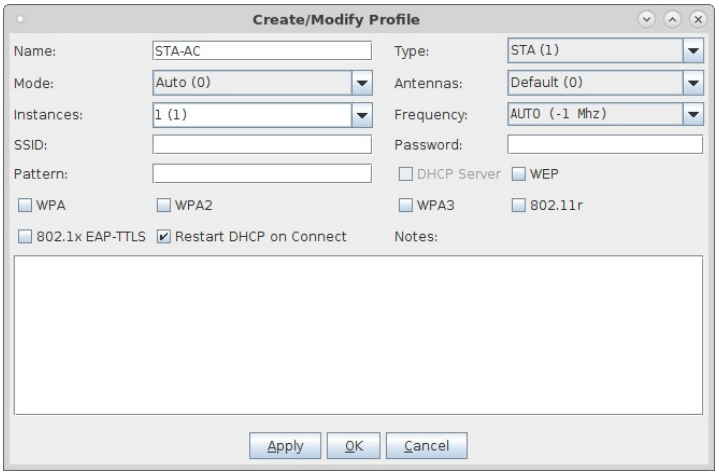
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



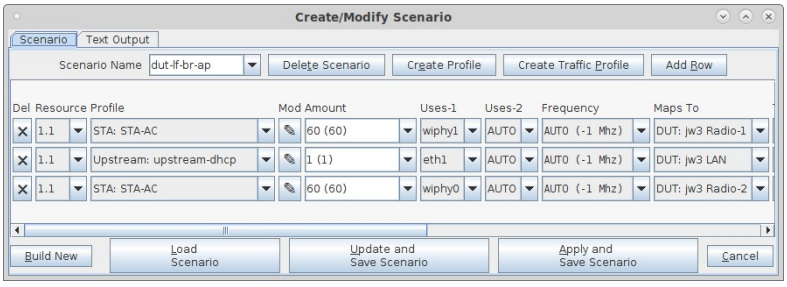
C. Configure an Upstream profile using eth1 on the LANforge system.



D. Configure an STA profile on the LANforge system.

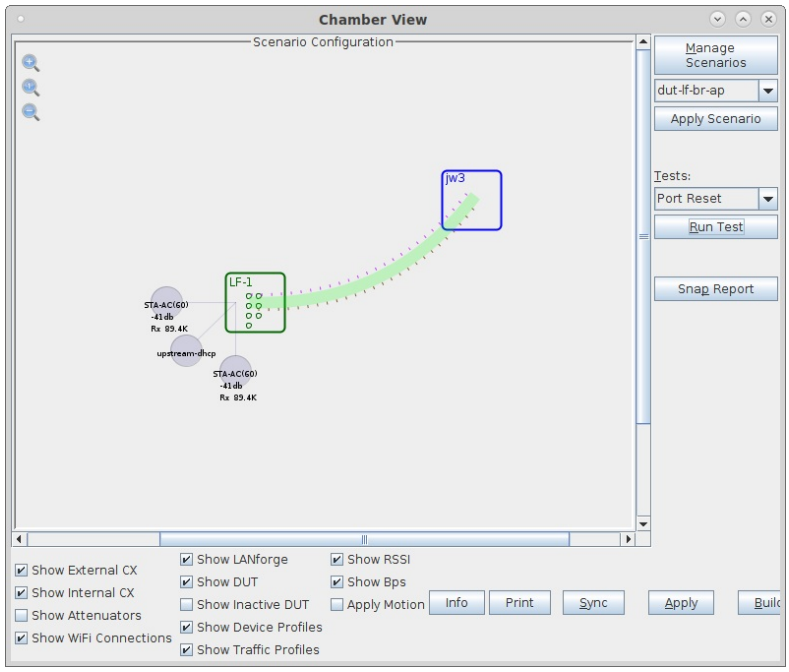


E. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).

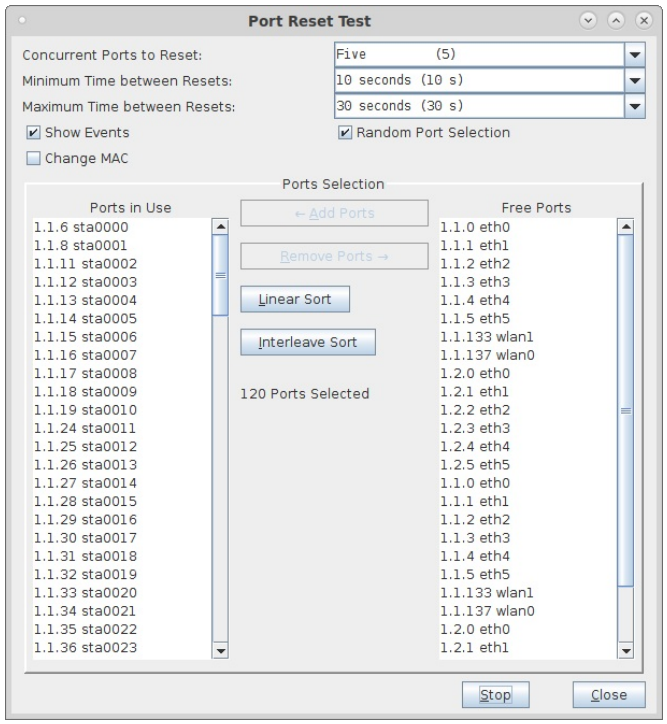


2. Use Chamber View to run a Port Reset test.

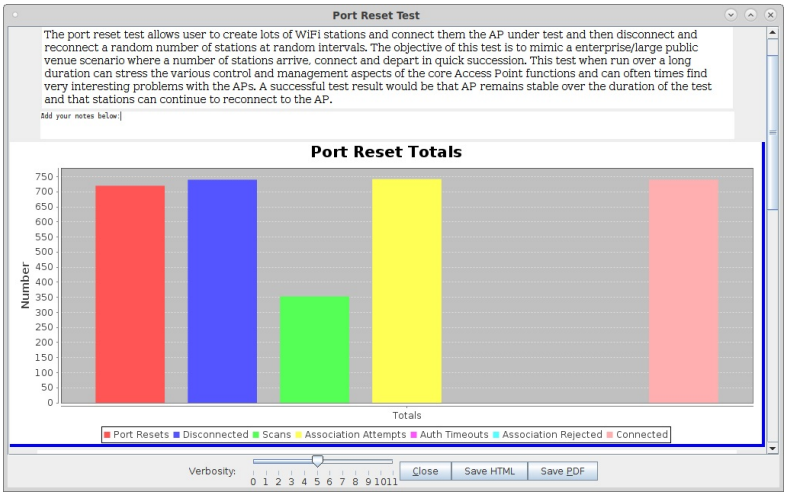
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario or create a new scenario as needed. Apply the Scenario, then Build the scenario.



B. Select the **Port Reset** test and click **Run Test**. You should see the Port Reset Test configuration window pop up. By default, all of the stations will be selected to use in the reset test. You may adjust the selection at this time. If you want each station to act like a new device when it resets, select the 'Change MAC' checkbox. Make any other configuration changes:



C. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs.

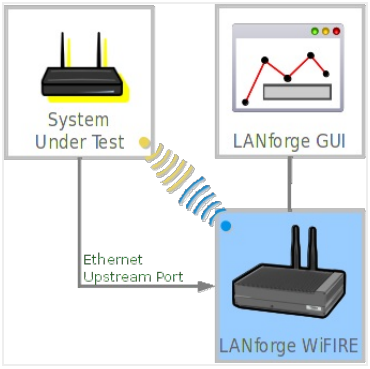


D. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example Rate vs Range Report

Comparing performance of different AP configurations

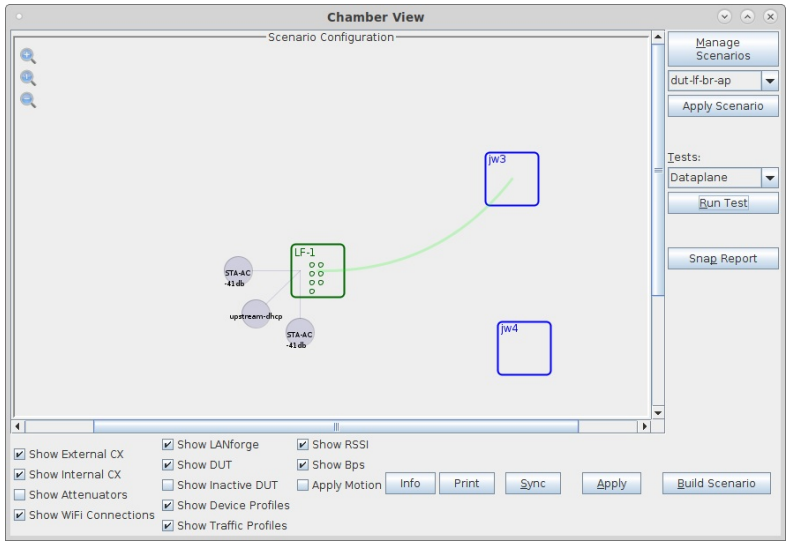
Goal: Compare dataplane throughput for several different APs set up in a similar manner. This allows comparing different hardware performance, and a similar test case can compare different firmware/software versions or other configuration changes.

In this test scenario, the LANforge CT522 is used to create a station and generate packets at different packet sizes to and from a series of APs. We will run one set of iterations of the dataplane test, pause it, make changes to use a different SSID, and then restart the test. The result is a report showing the differences in performance of the different APs under test. This specific test case is run over-the-air in an office. It would perform better and be more repeatable if RF chambers like the CT820a were used.

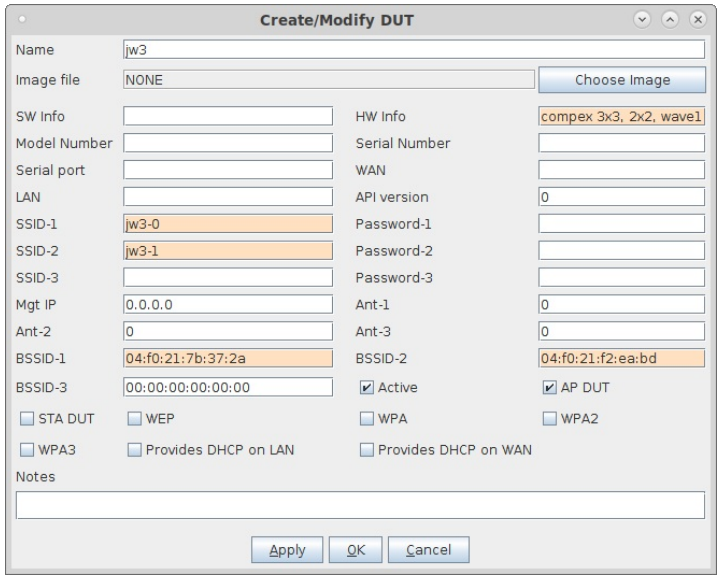


1. Configure Chamber View for DUT testing.

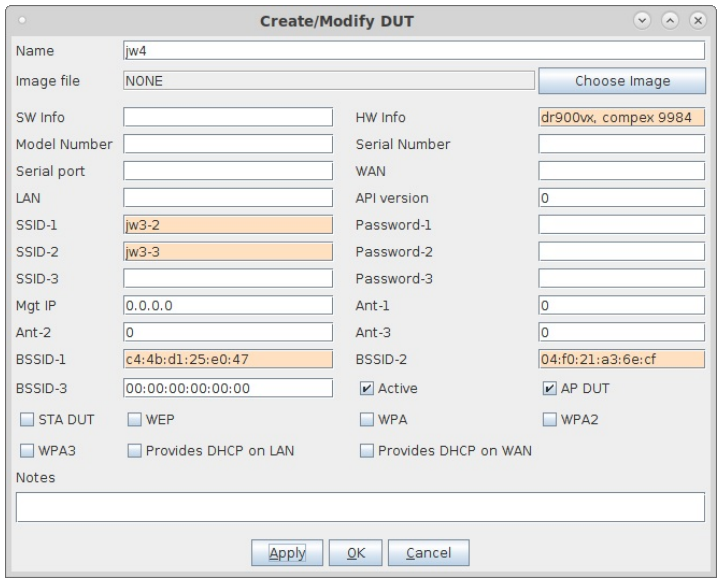
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber-View to create various objects.



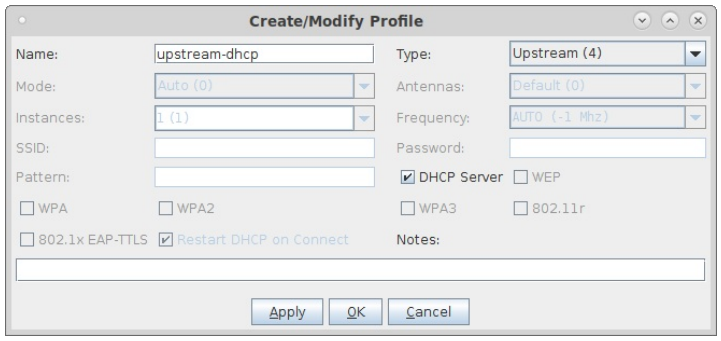
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP. Create additional DUTs as needed.



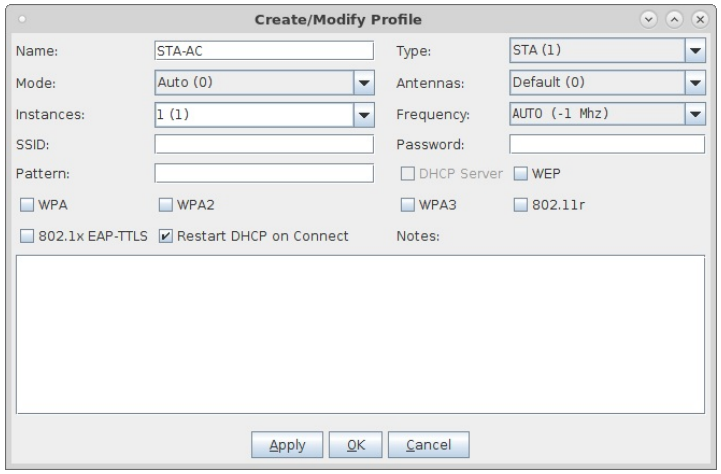
C. This example uses a second DUT as well.



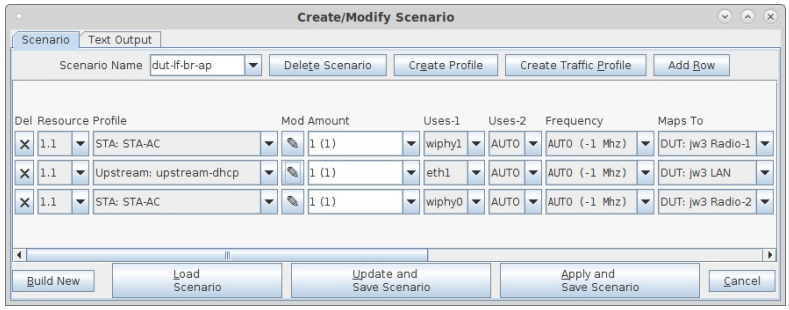
D. Configure an Upstream profile using eth1 on the LANforge system.



E. Configure an STA profile on the LANforge system.

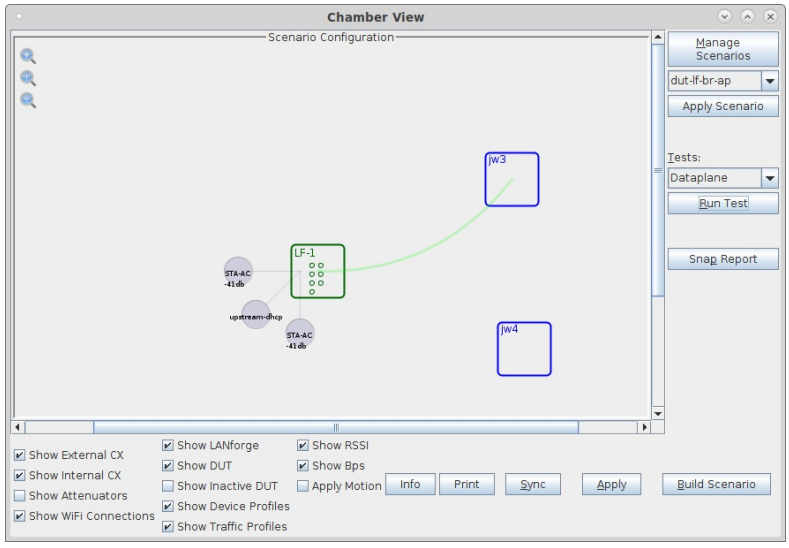


F. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT). Please note that we will manually configure the station to connect to the second DUT as part of the test steps below.

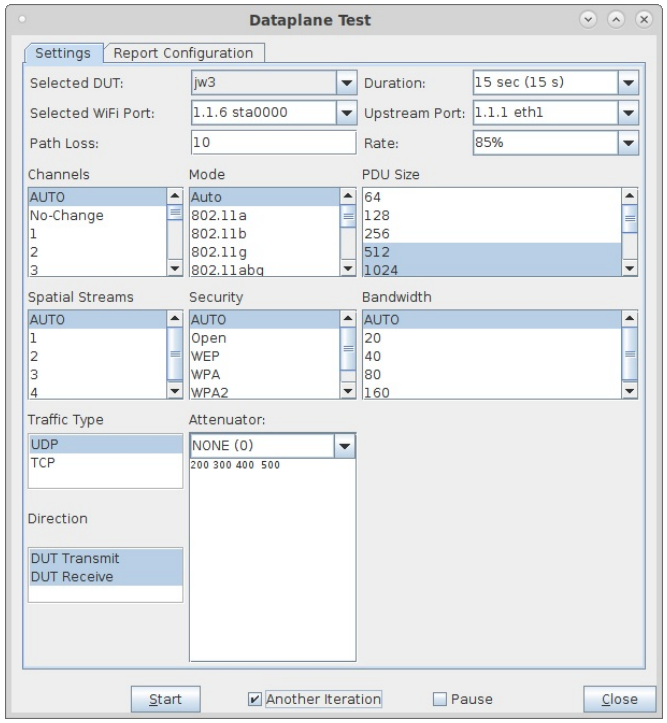


2. Use Chamber View to run a Dataplane comparison test.

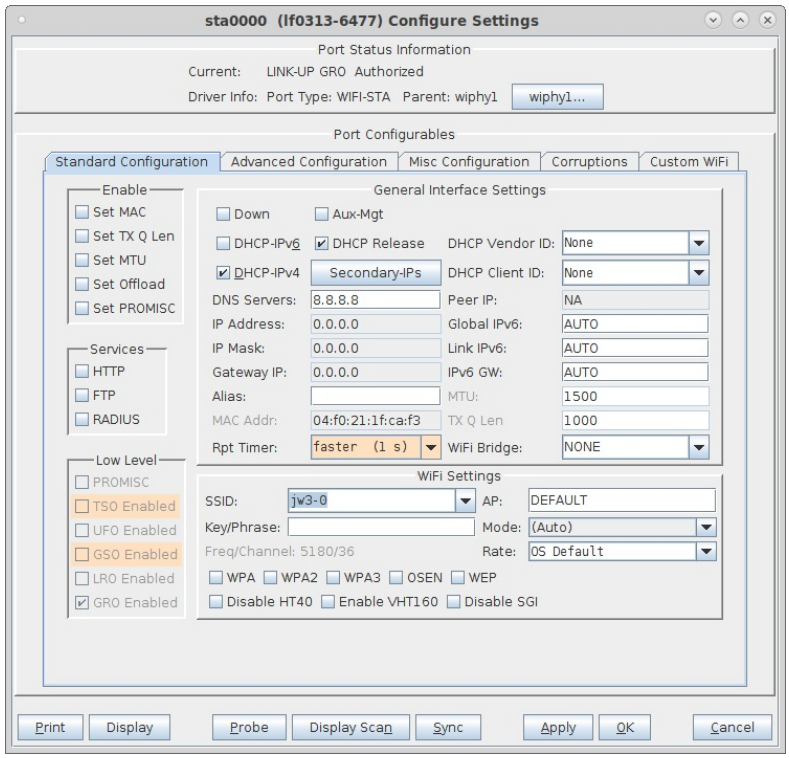
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario or create a new scenario as needed. Apply the Scenario, then Build the scenario.



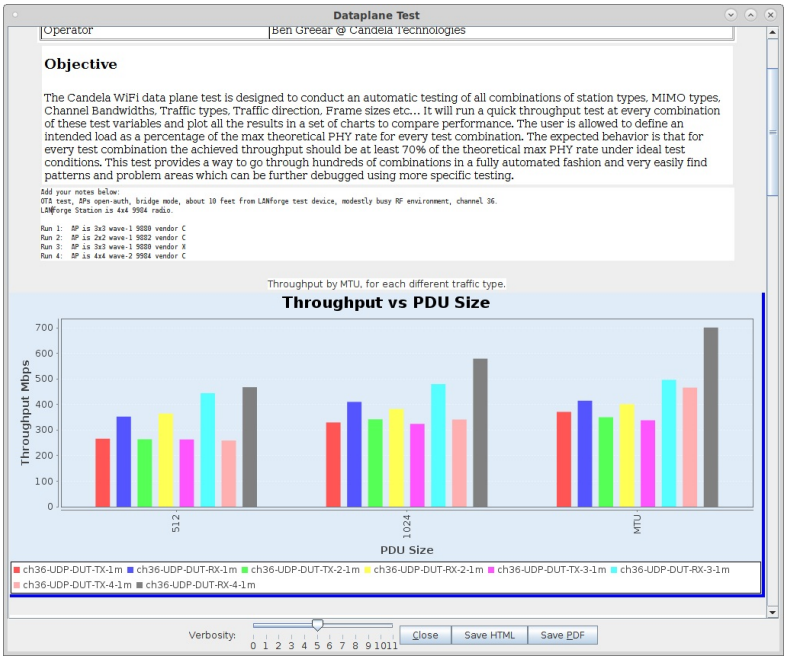
B. Select the **Dataplane** test and click **Run Test**. You should see the RX Sensitivity Test configuration window pop up. It will remember the last configuration for most fields. Select the DUT and WiFi station device, and select the combinations of traffic types you wish to send. Select the **Another Iteration** checkbox to allow comparison graphs.:



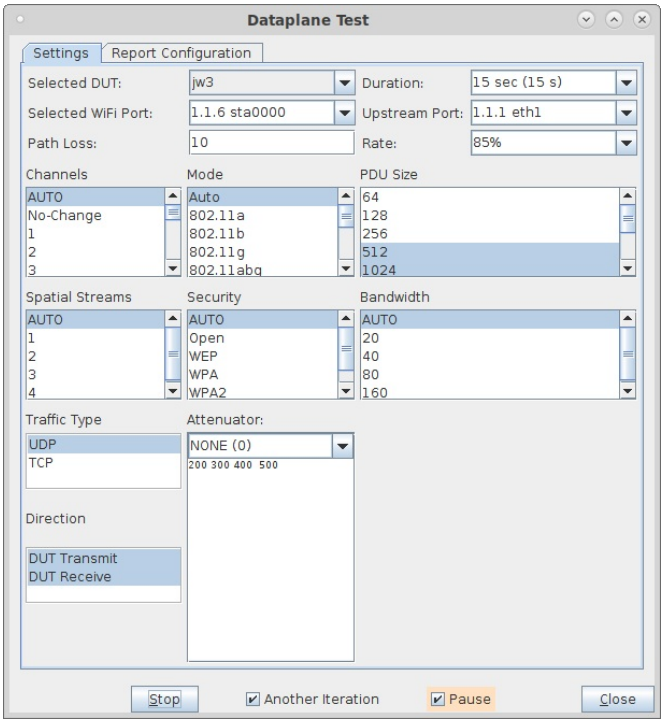
C. We have 4 total BSSIDs that we wish to test. The Scenario creation logic will have chosen one of the BSSIDs for the station, but we will need to override that for each of our comparison runs. Go to the Port-Mgr tab in the LANforge-GUI, double-click the station, and make sure that its SSID is correct. In case you are using different passwords you could change that at this time as well. When complete, click Apply. You can leave the window open as you will need it in future steps:



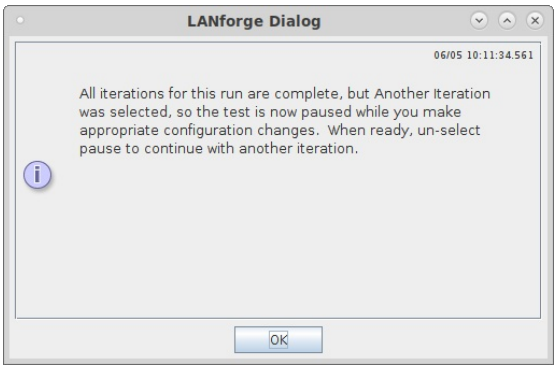
D. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs.



E. When the first comparison run is complete, a popup window will be shown, and the **Paused** checkbox will be selected. Select new BSSID by reconfiguring the station with a new BSSID, and when the reconfiguration is complete, un-select the Paused checkbox on the Dataplane test to do the next comparison run. Continue to run new comparisons in this manner until the full test is complete. When complete, unselect the **Another Iteration**, and then unselect 'Paused* to have the test complete:



F. This is the pause message, it is just informational and you can close it after it pops up.

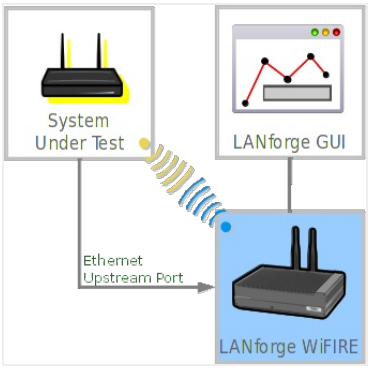


G. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example Dataplane Comparison Report

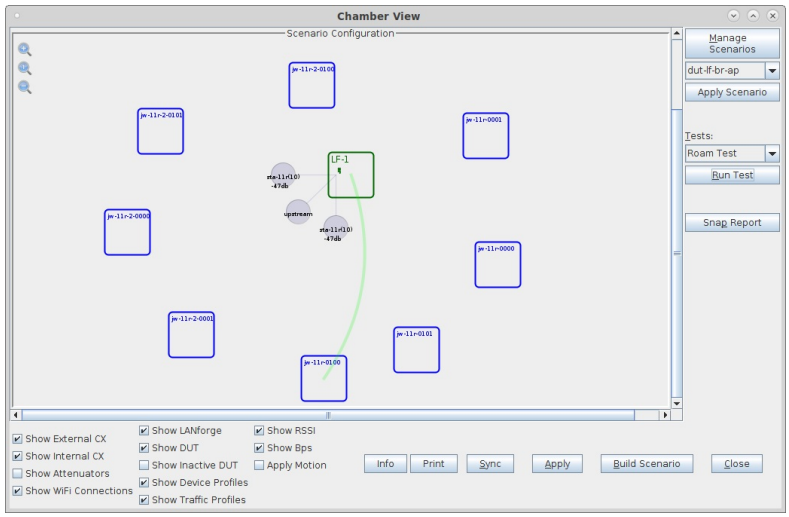
Testing Station Roaming with 802.11r, 802.11k and Protected Management Frames

Goal: Setup and run a Station Roam test against a cluster of APs supporting 802.11k, 802.11r, and Protected Management Frames (PMF/MFP). This test uses a 2-radio LANforge CT522 system, but other similar systems will work as well. This tests AP functionality and stability over many roam attempts. This is a good test of the AP's management plane stability, and may also be a good controller test in case the AP system uses a controller.

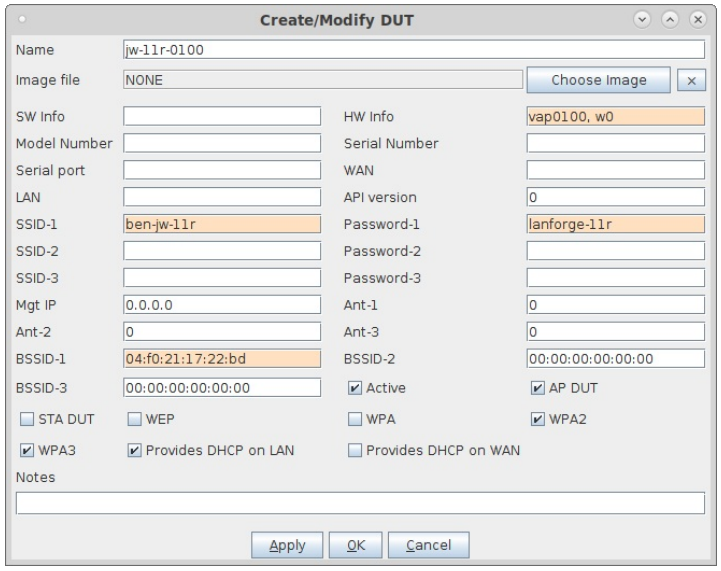
In this test scenario, the LANforge CT522 is used to create 2 stations and then have them roam between the APs. The test will count the number of successful roams as well as various different failure cases. This example assumes you have some experience with Chamber View, and that you have a LANforge system and properly configured AP cluster. A programmable attenuator and two isolation chambers would add the ability to test station roams at different RF signal levels, but this test normally runs fine withour chambers or attenuators. This feature is in LANforge version 5.3.9 and higher.



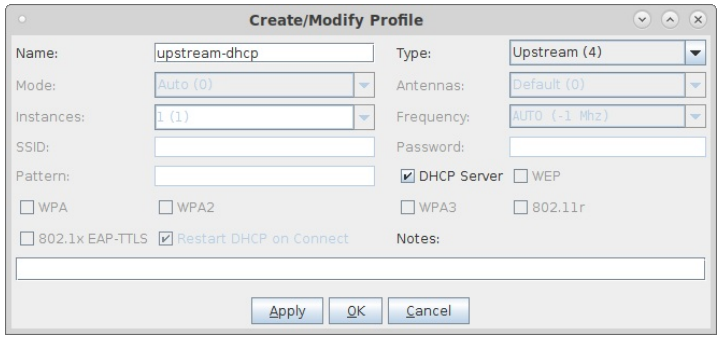
1. Configure Chamber View for Station Roaming and Similar Tests.
   A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects. If you do not have chambers or attenuators, just create the DUT object and skip the chamber setup. This image shows the completed setup.



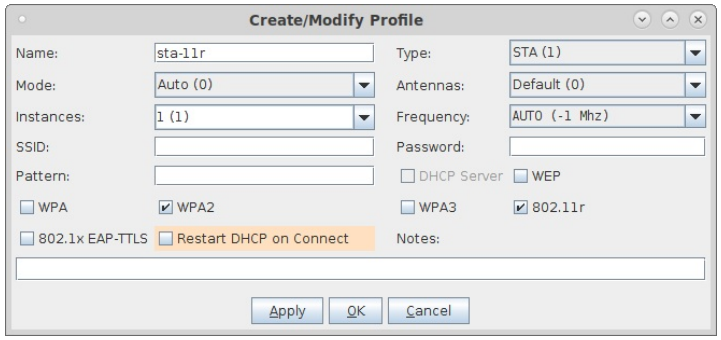
B. Create a Device Under Test (DUT) Profile for each of your APs. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP. The authentication information and BSSID should be the same for all APs in the cluster, so probably you just need to change the name and BSSID for each of your DUTs and click save. To aid the visual representation of the roaming, consider putting the DUTs in a circle around the LANforge system as shown in the image above.



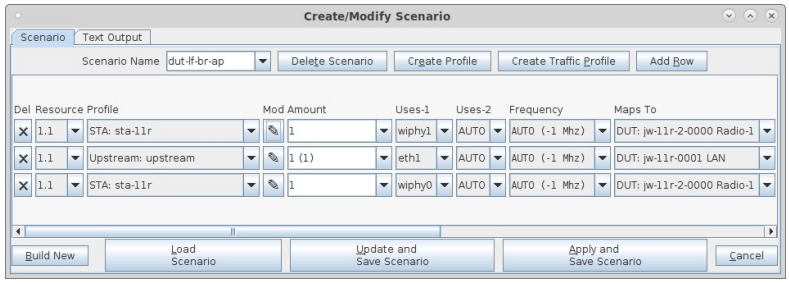
C. Configure an Upstream profile using eth1 on the LANforge system.



D. Configure an STA profile supporting 802.11r on the LANforge system. Roaming tests normally should de-select the Restart DHCP on Connect behaviour, as shown.

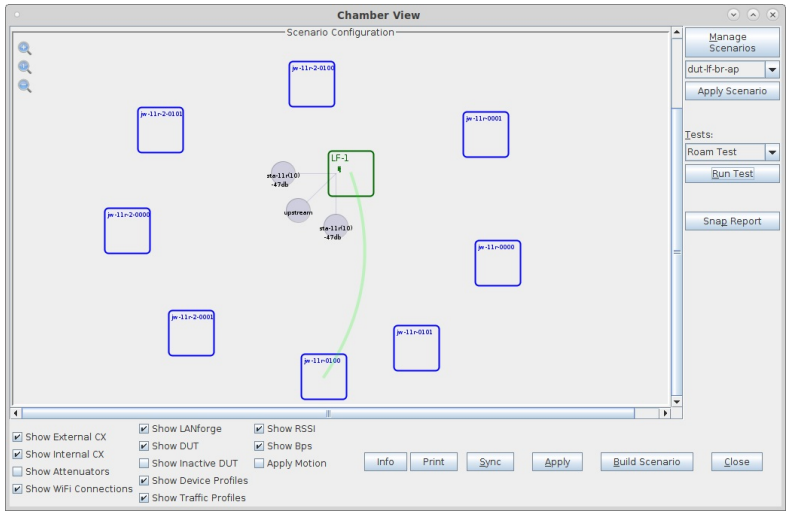


E. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radios and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT). You can map to any of the DUT APs in the cluster and the system will still be able to roam to the others.



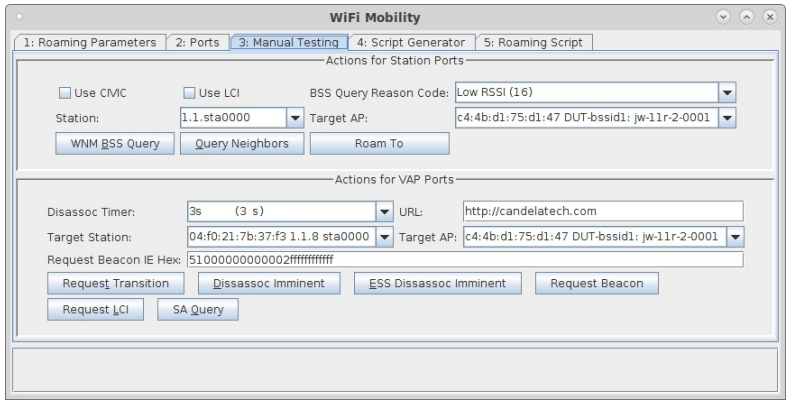
2. Use Chamber View to run a Station Roam test.

A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario or create a new scenario as needed. Apply the Scenario, then Build the scenario.

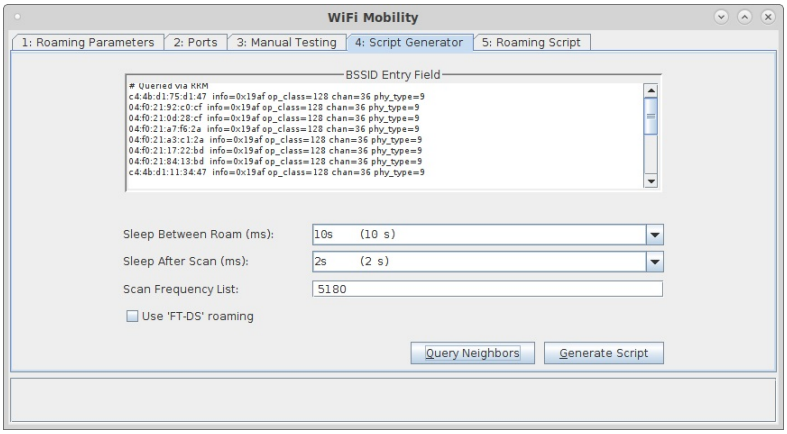


B. Select the **Roam Test** test and click **Run Test**. You should see the Wifi Mobility Test configuration window pop up. You can normally just use the defaults in the Roaming Parameters tab, so that tab is ignored in this example. By default, all of the stations will be selected to use in the roam test. You may adjust the selection at this time.:

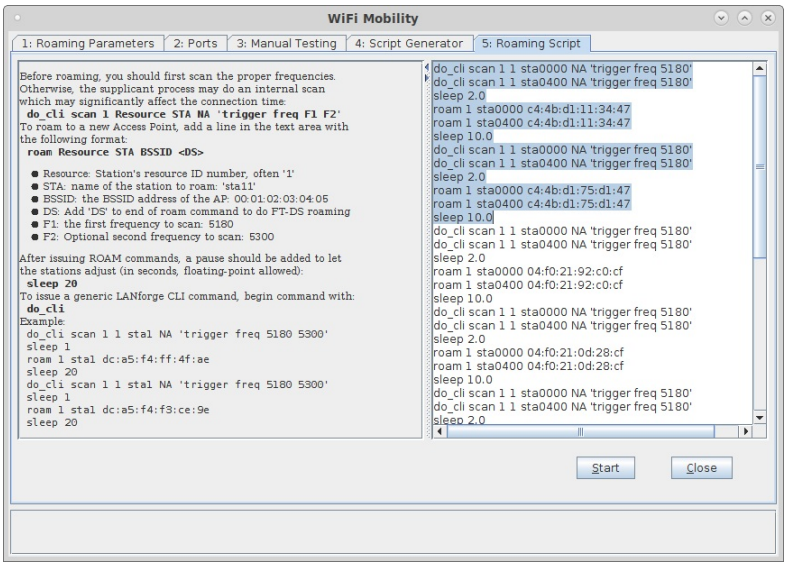
C. You can do some manual testing, including some 802.11k/v features on the Manual Testing tab, but in this example, we will not be using those features.



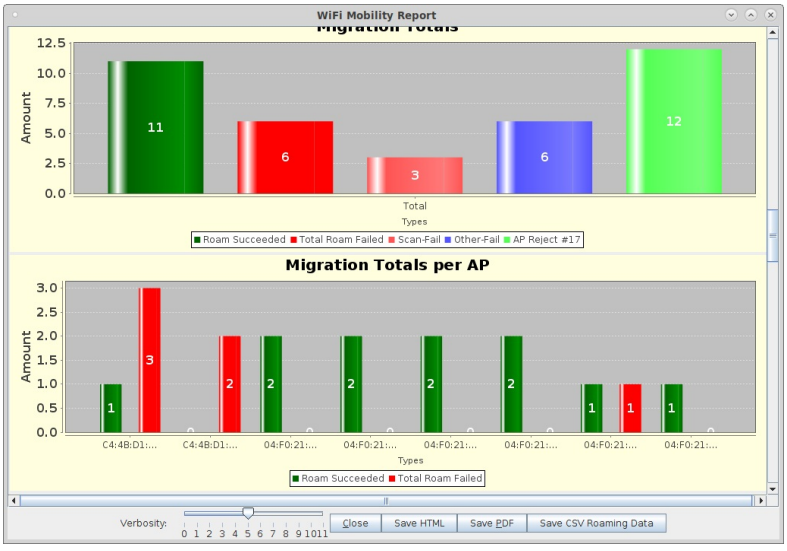
D. The Script Generator tab is used to create the roaming script. If your APs support Neighbor Report Requests, then you can click the Query Neighbors button and it should populate the BSSID Entry Field as well as the Scan Frequency List. If your stations have not been brought up yet, then the Query Neighbors button will cause them to associate. It make take a small bit of time until the neighbor report becomes available, please click the Query Neighbors button again after 10 or so seconds. Double-check that all of your APs are found and that the frequency list is the expected value. If your AP does not support Neighbor Query, then manually enter the BSSIDs and frequencies.



E. When the BSSIDs have been discovered to manually entered, click the Generate Script button. The generated script describes the roaming behaviour. You may also edit or paste in your own script, and you may save the generated script text for later use.



F. To start the test, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report window will be created and will be updated as the test runs. The AP cluster used in this example has some issues and is rejecting a lot of roam attempts with 'code-17', which indicates the AP thinks it has no more capacity. Hopefully your system works better!

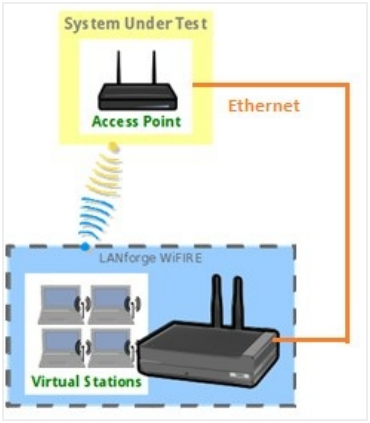


G. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example Roam Report

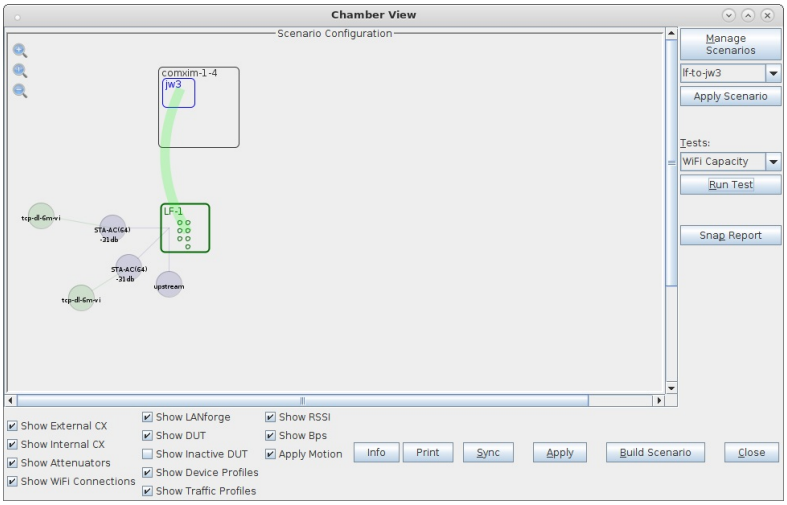
Automate WiFi Capacity and other GUI tests.

Goal: Use a command-line script to have the LANforge-GUI run the WiFi Capacity test and generate a pdf automatically.

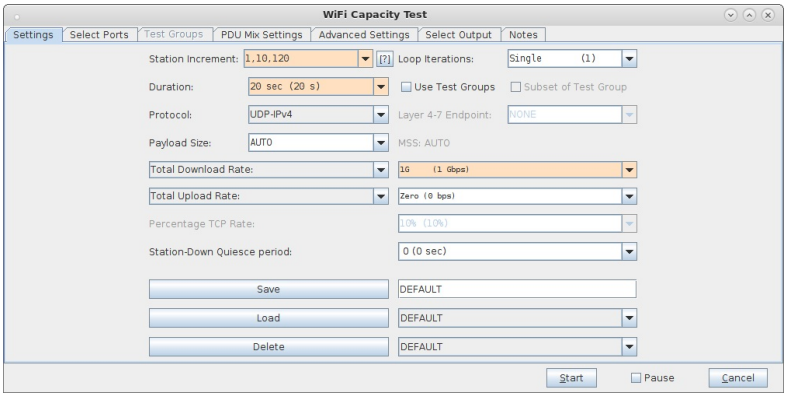
In this test scenario, a script is used to bring up the WiFi Capacity test with a pre-configured configuration. The capacity test is then started and a report is generated. All of this is automated, and other tests such as Dataplane are also supported. This feature requires LANforge version 5.4.1 or higher.



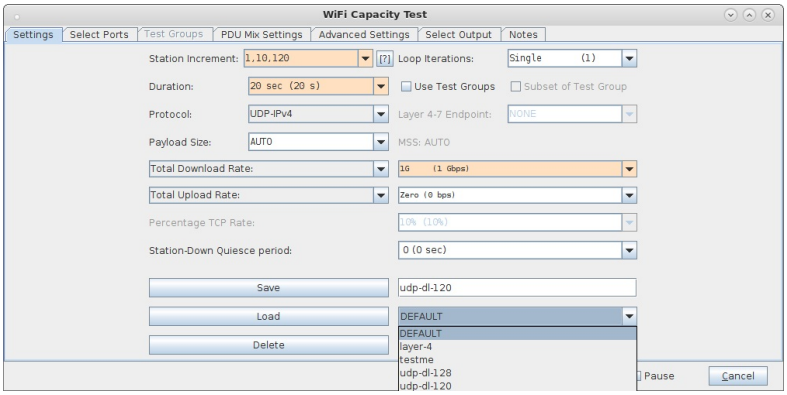
1. Configure WiFi Capacity Test for automated run.
   A. For this to work, the LANforge GUI must be started with the **-cli-socket 3990** argument. This causes it to open a socket to listen for text commands.
   B. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Create an appropriate scenario and DUT if you have not already done so. Other cookbook examples have more details of how to do this, please see those if you are unfamiliar with Chamber View.



   C. Select WiFi Capacity test, and click **Run Test** to configure it as desired.

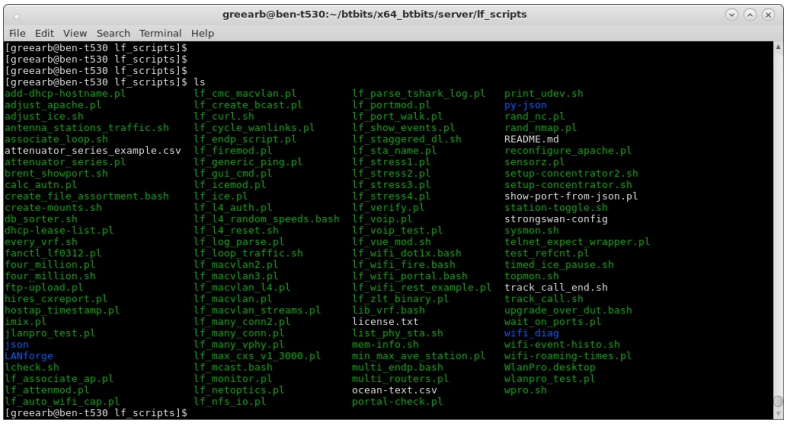


D. Enter a name in the 'Save' field, click save, and make sure it shows up as a loadable configuration. In this case, we are saving the configuration as 'udp-dl-120'

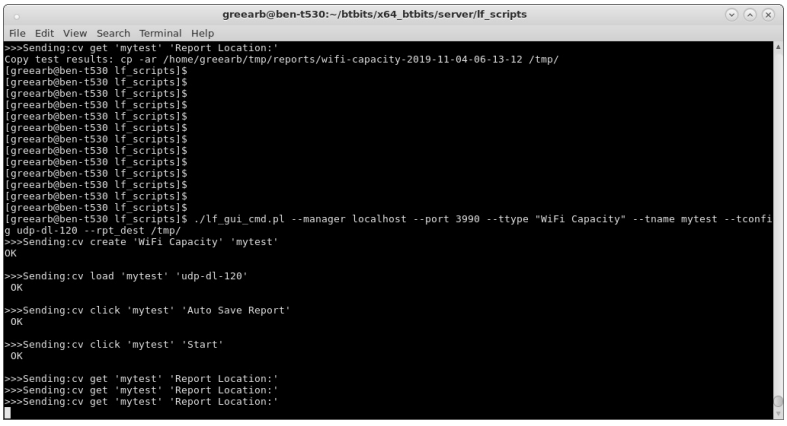


2. Use the lf_gui_cmd.pl script to launch the WiFi Capacity Test.

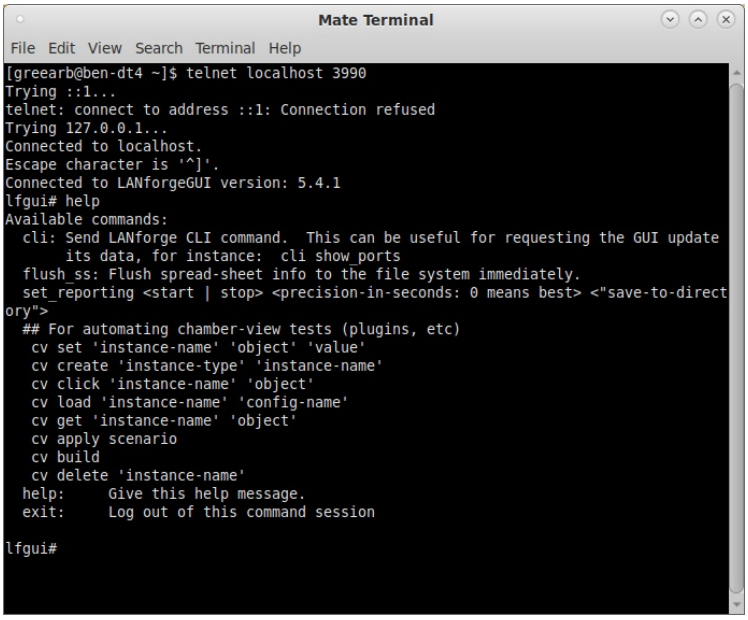
A. Open an ssh session or terminal window and log into the LANforge system, or some other system with the LANforge scripts/ repository. On a LANforge system, this will usually be /home/lanforge/scripts In this case, the directory name is called lf_scripts



B. Run the lf_gui_cli_cmd.pl script with appropriate arguments. Use --help for details. Once you run this, the WiFi Capacity test should be automatically opened and the test will be started. The script will end when the capacity test has completed. You may copy the results to some easily found location, such as a web server directory.



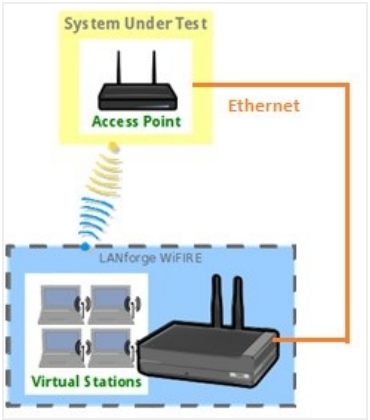
C. For details on what GUI-CLI commands are supported, please see the screen-shot below and look at the contents of the lf_gui_cmd.pl script.



AP-Auto Test Suite Setup and basic AP client connectivity testing.

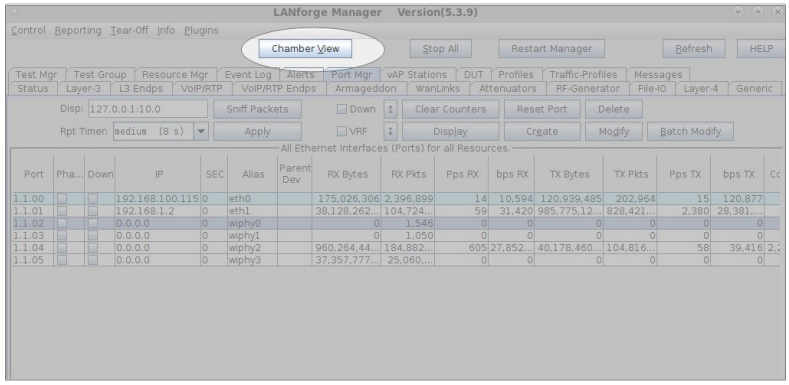
Goal: Setup AP-Auto preliminaries and run a basic client connectivity test for an AP using the LANforge CT523c or similar system. The AP-Auto test is similar to the TR-398 test, but is designed to be functional with a minimum amount of test equipment. A 2-radio LANforge system and DUT is all that is required to run these tests.

In this test scenario, a 6-radio LANforge CT523 is used to create stations and run the AP-Auto Basic Client Connectivity test. This example assumes you have some experience with Chamber View, and that you have a LANforge system and a DUT AP. The AP and LANforge may be in chambers, but that is not required. This feature requires LANforge version 5.4.2 or higher.

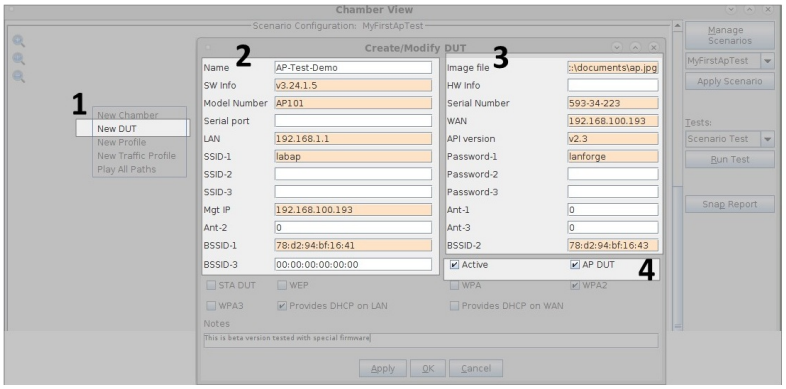


1. Configure Chamber View for AP-Auto and Similar Tests.

A. Click on the **Chamber View** button in the LANforge GUI to launch the Chamber View screen.



B. Configure an AP under test (DUT).



C. To configure the DUT:

A. Right click anywhere on the canvas in Chamber View and select **New DUT** from the menu.

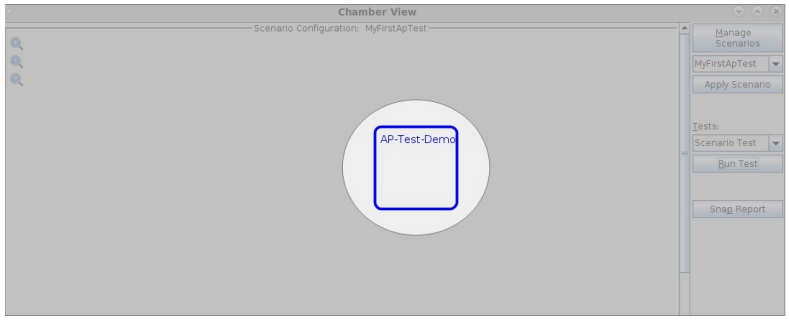
B. Enter all the known details about the AP under test including the SSID (multiple SSIDs if the AP has multiple SSIDs), BSSID (multiple BSSIDs if the AP has multiple radios with the same SSID).

C. If the AP has security enabled, enter the security information.

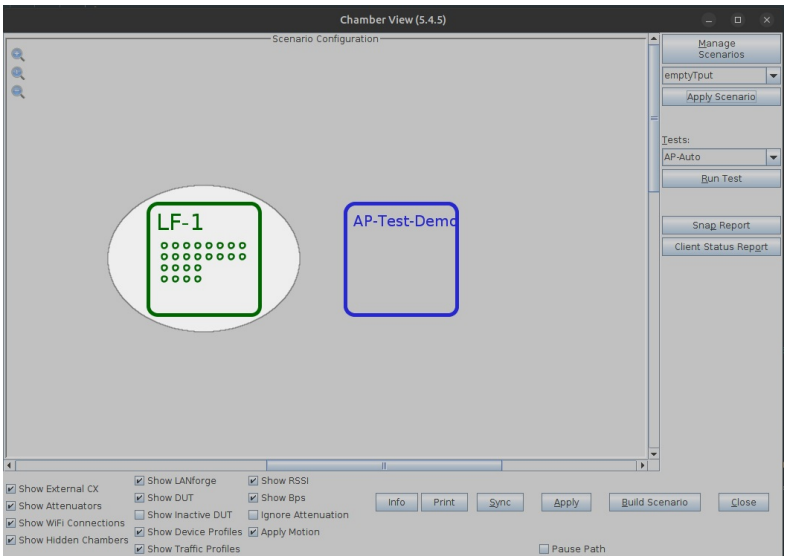
D. Set the DUT to Active by checking the **Active** checkbox.

E. Select the **AP DUT** checkbox to indicate that this DUT is an Access Point.

D. Once **OK** is clicked, check to make sure the DUT appears in Chamber View as shown below.



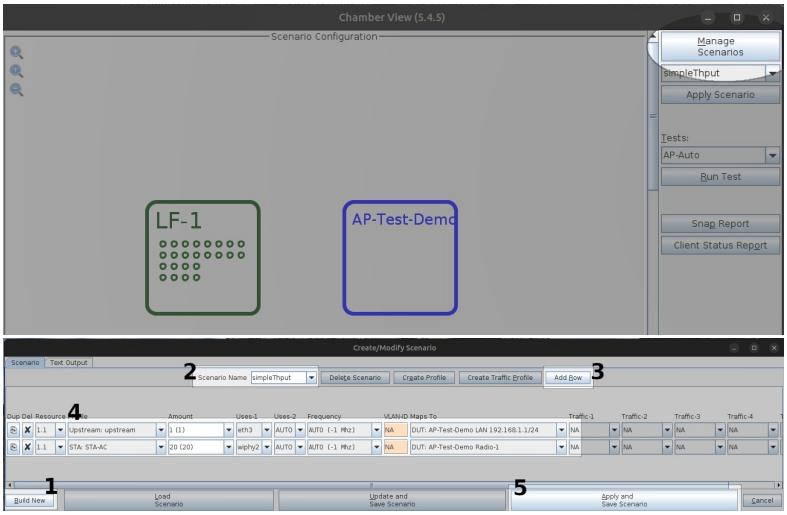
E. Select the LANforge system to be used in the test.



F. All active LANforge systems will automatically appear in Chamber View. Select the unit being used in the test and drag to the middle of the canvas next to the AP under test. The small circles inside the LANforge box represent the SMA connectors. The columns of circles represent the individual radios in the system. For instance, in the screenshot above LF-1 has 8 columns of circle indicating 8 radios. The first four are 4x4 radios and the next four are 2x2 radios with 2 circles each.

G. Create the test scenario

Click on the **Manage Scenarios** button and then open a Create/Modify dialog box.



H. In the Create/Modify dialog box

A. Click on **Build New** button to create a new scenario.

B. Enter a scenario name e.g: 'SimpleThput' (no spaces allowed)

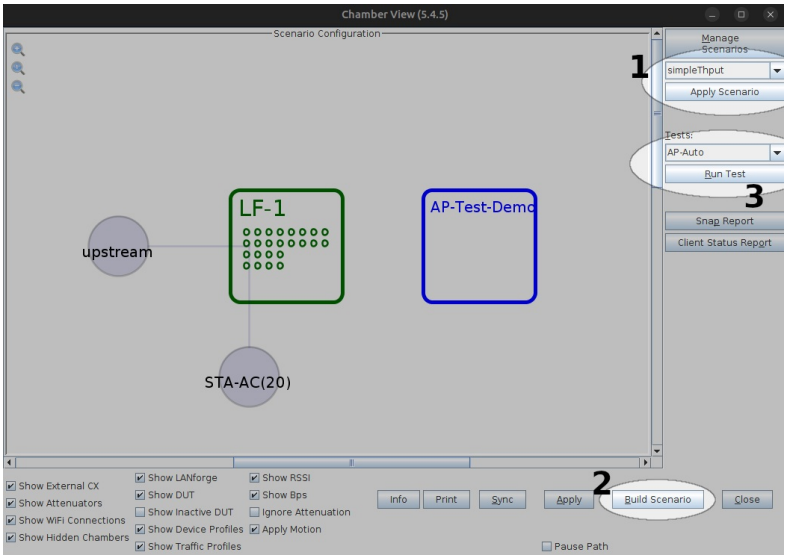
C. Click **Add Row** button to add one or more rows.

D. Add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).

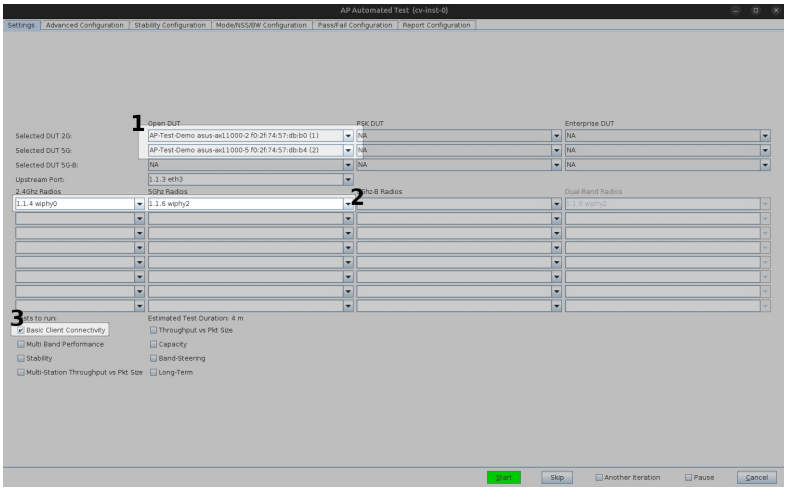
E. Choose **Apply and Save Scenario**

2. Configure AP-Auto settings and run Basic Client Connectivity test.

A. Ensure that the correct scenario is chosen, then click **Build Scenario**



B. Select the AP-Auto test and click **Run Test**. You should see the AP-Auto Test configuration window pop up. It will remember the last configuration for most fields, and you can also save and load configurations in the Advanced Configuration tab.



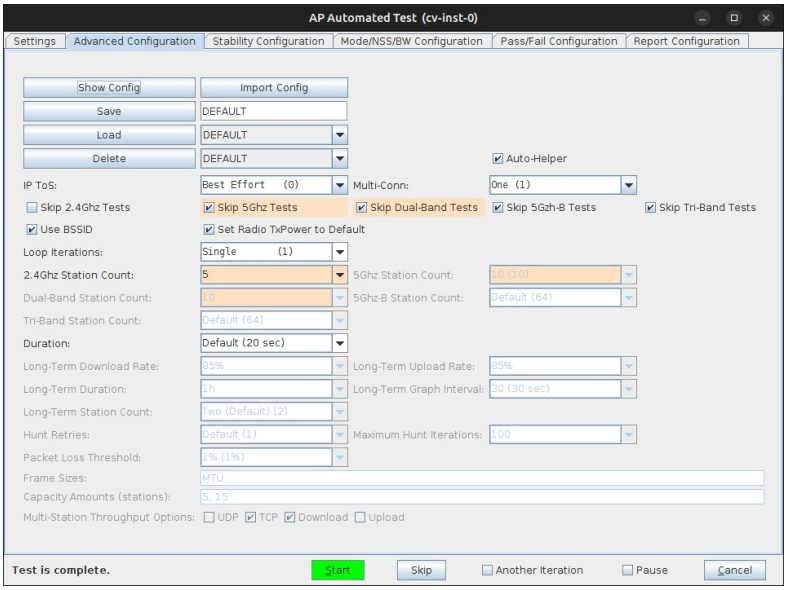
C. In the AP-Auto Settings Tab

A. Select the **DUT 2G** and **DUT 5G** SSIDs. This test requires that Open or PSK SSIDS are filled out. For SSIDs that are not enabled, leave as 'NA'.

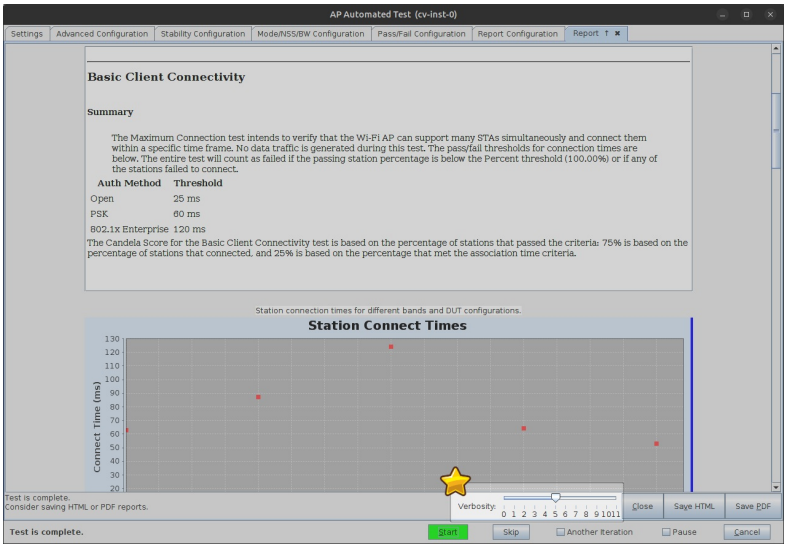
B. Select the LANforge radios to be used in this test. You need at least one 2.4Ghz radio and one 5Ghz radio for full functionality.

C. At the bottom, select the **Basic Client Connectivity** test checkbox.

D. Located Next to the Settings Tab, the Advanced Configuration tab lets you save and restore test configurations and also tune the behaviour of the automated tests. Station counts and skipping the single-band tests are the main options that affect this test.



E. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report tab will be created and will be updated as the test runs.



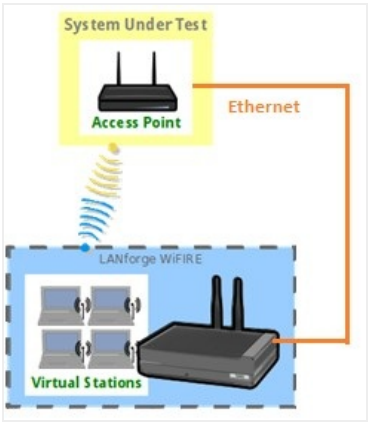
F. Consider changing the test result verbosity level by adjusting the **Verbosity** slider. Maximizing it will show all generated figures and data. The verbosity level also affects the length of the saved report.

G. At the end of the test, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click **Save PDF** and the browser will be directed to open the pdf file directly. Please see this example AP-Auto Basic Client Connectivity Report.

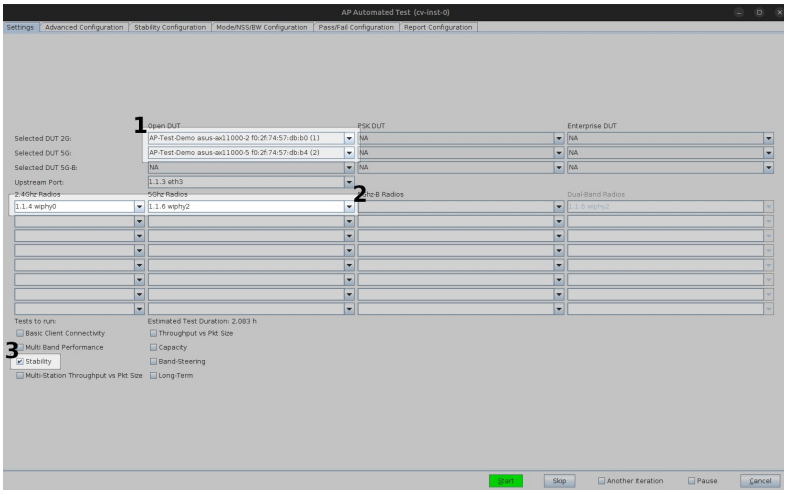
Testing AP Stability with the AP-Auto Automated Test Suite

Goal: Run an AP-Auto test for an AP using the LANforge CT523c or similar system in order to test how well the AP can handle station stability. The AP-Auto test is similar to the TR-398 test, but is designed to be functional with a minimum amount of test equipment. A 2-radio LANforge system and DUT is all that is required to run these tests.

In this test scenario, the LANforge CT523 is used to create stations and run the Stability test. This example assumes you have some experience with Chamber View, and that you have a LANforge system and a DUT AP. The AP and LANforge may be in chambers, but that is not required. This feature requires LANforge version 5.4.2 or higher.



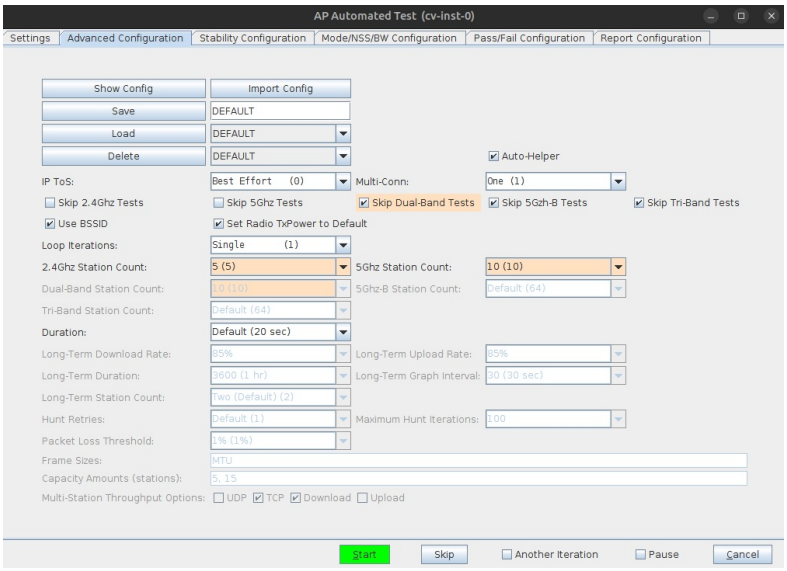
1. If you haven't setup or performed AP-Auto tests on your LANforge system, please refer to the AP-Auto Test Suite Setup guide for quick setup.
2. Running the AP-Auto Stability Test:
   A. Open the AP-Auto Test window.



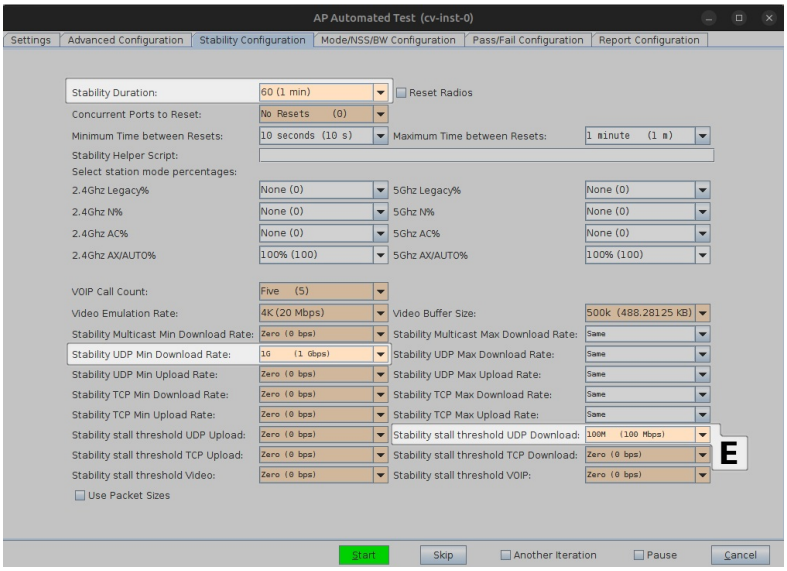
   B. In the AP-Auto Settings Tab:

      A. Select the **DUT 2G** and **DUT 5G** SSIDs. This test requires that Open or PSK SSIDS are filled out.
      B. Select the LANforge radios to be used in this test. You need at least one 2.4Ghz radio and one 5Ghz radio for full functionality.
      C. At the bottom, select the **Stability** test checkbox.

C. Your Advanced Configuration tab should look similar to the following:



D. Most of the settings for the Stability test are located in the Stability Configuration tab. Below is an example configuration for running a UDP Download Stability test for 60 seconds.



E. Note that the value chosen in the Stability stall threshold UDP Download field will directly impact the Pass/Fail outcome of the test. Decreasing this threshold will increase the likelihood of a PASS result, and increasing it will have the inverse effect.

F. When the configuration is complete, click the **Start** button (which will change to **Stop** once start is clicked) to start the test. An interactive report tab will be created and will be updated as the test runs.

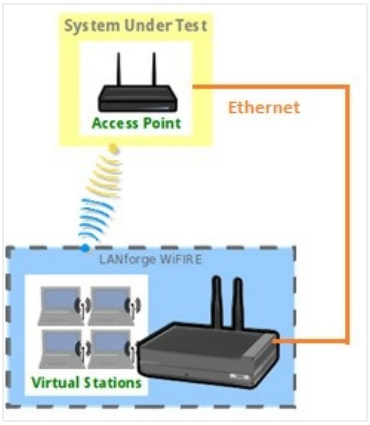
G. You can change the test result verbosity level by adjusting the **Verbosity** slider. Maximizing it will show all generated figures and data. The verbosity level also affects the length of the saved report.

H. At the end of the test, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click **Save PDF** and the browser will be directed to open the pdf file directly. Please see this example AP-Auto Mix Stability Report.

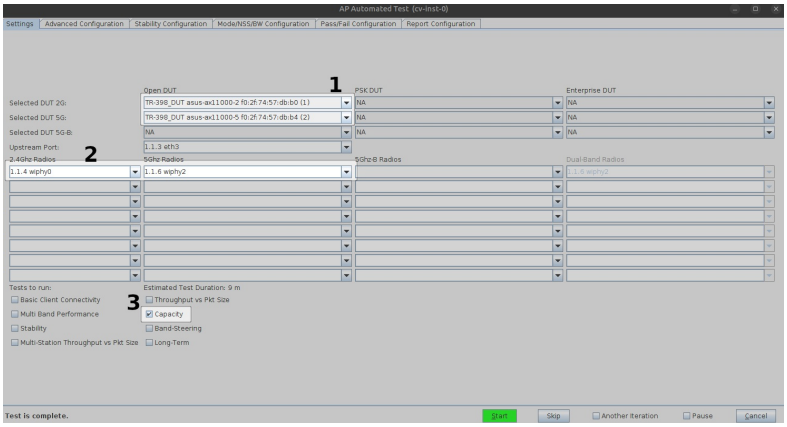
Testing AP Capacity Performance with the AP-Auto Automated Test Suite

Goal: Run an AP-Auto test for an AP using the LANforge CT523c or similar system in order to test how well the AP can handle many stations simultaneously. The AP-Auto test is similar to the TR-398 test, but is designed to be functional with a minimum amount of test equipment. A 2-radio LANforge system and DUT is all that is required to run these tests.

In this test scenario, the LANforge CT523 is used to create stations and run the Capacity test. This example assumes you have some experience with Chamber View, and that you have a LANforge system and a DUT AP. The AP and LANforge may be in chambers, but that is not required. This feature requires LANforge version 5.4.2 or higher.



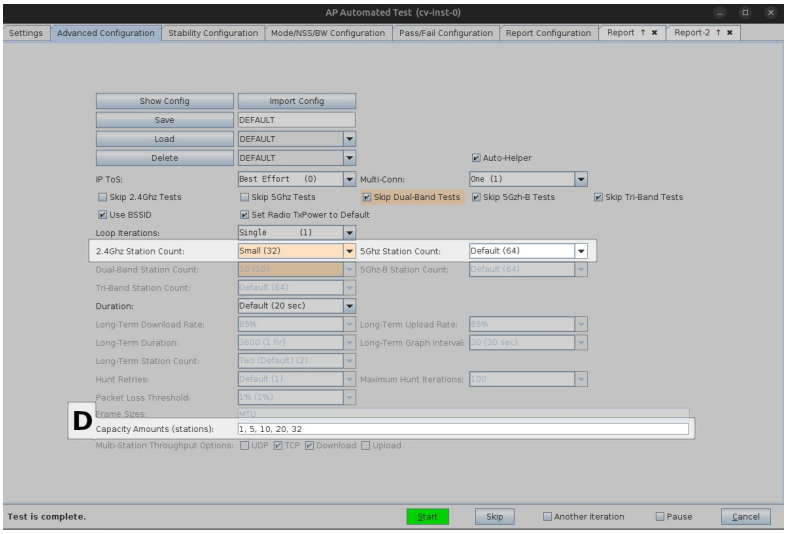
1. If you haven't setup or performed AP-Auto tests on your LANforge system, please refer to the AP-Auto Test Suite Setup guide for quick setup.
2. Running the AP-Auto Capacity Test:
    A. Open the AP-Auto Test window.



    B. In the AP-Auto Settings Tab:

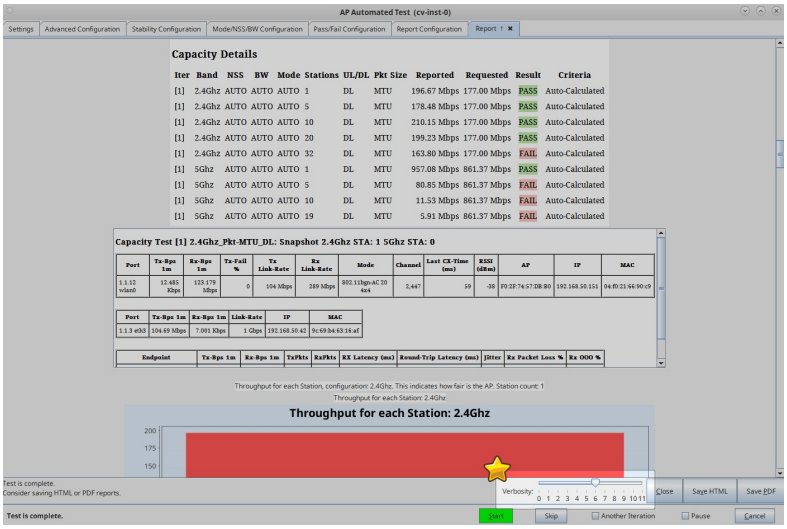
        A. Select the **DUT 2G** and **DUT 5G** SSIDs. This test requires that Open or PSK SSIDS are filled out.
        B. Select the LANforge radios to be used in this test. You need at least one 2.4Ghz radio and one 5Ghz radio for full functionality.
        C. At the bottom, select the **Capacity** test checkbox.

C. Your Advanced Configuration tab should look similar to the following:



D. You can change the number of stations that are brought up in each capacity trial by changing **Capacity Amounts (stations)**. None of the capacity amount values should exceed the station count of any enabled test band.

E. When the configuration is complete, click the **Start** button (which will change to **Stop** once start is clicked) to start the test. An interactive report tab will be created and will be updated as the test runs.



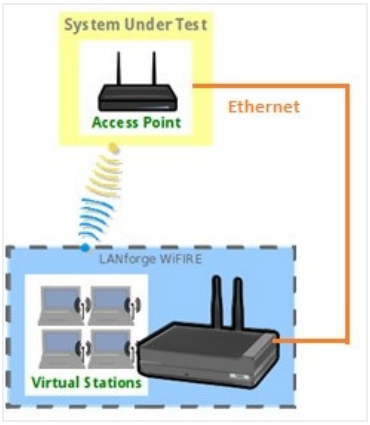
F. You can change the test result verbosity level by adjusting the **Verbosity** slider. Maximizing it will show all generated figures and data. The verbosity level also affects the length of the saved report.

G. At the end of the test, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click **Save PDF** and the browser will be directed to open the pdf file directly. Please see this example AP-Auto Capacity Report.

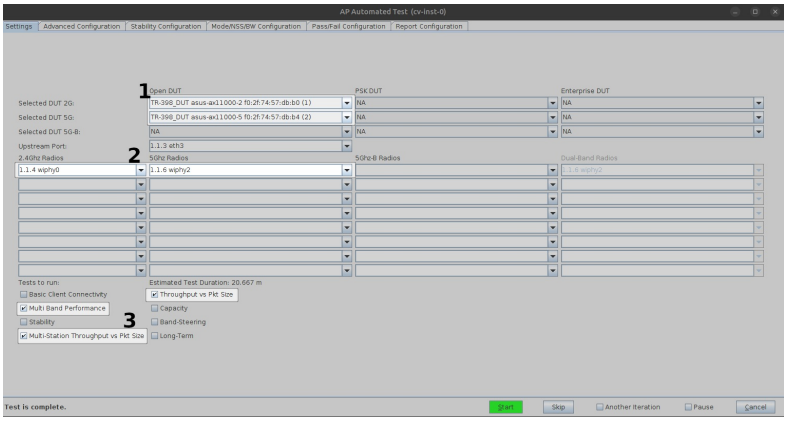
Testing AP Throughput with the AP-Auto Automated Test Suite

Goal: Run an AP-Auto test for an AP using the LANforge CT523c or similar system in order to test how well the AP can provide throughput in various scenarios. The AP-Auto test is similar to the TR-398 test, but is designed to be functional with a minimum amount of test equipment. A 2-radio LANforge system and DUT is all that is required to run these tests.

In this test scenario, the LANforge CT523 is used to create stations and run a series of Throughput tests. This example assumes you have some experience with Chamber View, and that you have a LANforge system and a DUT AP. The AP and LANforge may be in chambers, but that is not required. This feature requires LANforge version 5.4.2 or higher.



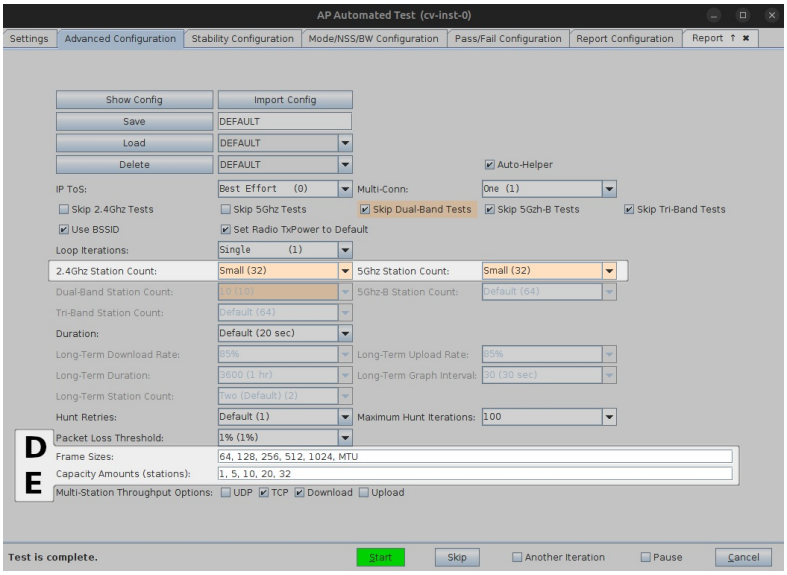
1. If you haven't setup or performed AP-Auto tests on your LANforge system, please refer to the AP-Auto Test Suite Setup guide for quick setup.
2. Running the AP-Auto Throughput Tests:
    A. Open the AP-Auto Test window.



    B. In the AP-Auto Settings Tab:

        A. Select the **DUT 2G** and **DUT 5G** SSIDs. This test requires that Open or PSK SSIDS are filled out.
        B. Select the LANforge radios to be used in this test. You need at least one 2.4Ghz radio and one 5Ghz radio for full functionality.
        C. At the bottom, select the **Multi Band Performance**, **Multi-Station Throughput vs Pkt Size**, and **Throughput vs Pkt Size** test checkboxes.

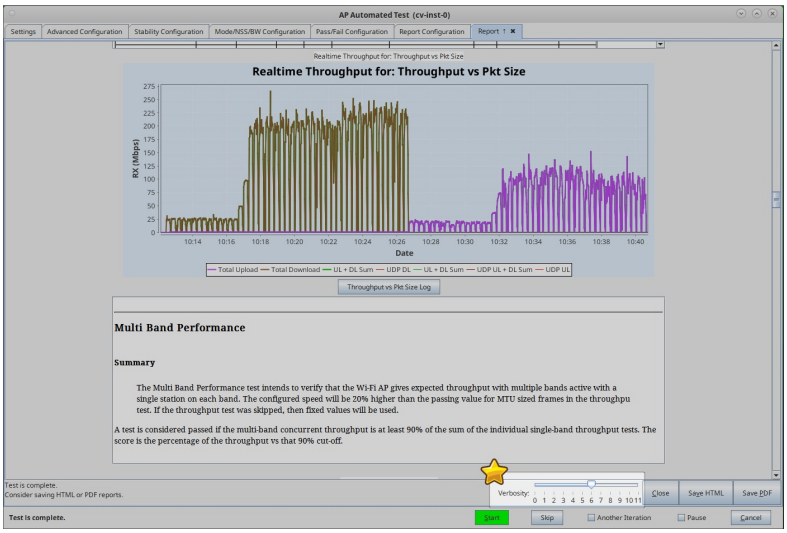
C. Your Advanced Configuration tab should look similar to the following:



D. You can specify the frame sizes to be evaluated in the throughput hunt tests by changing the **Frame Sizes** field.

E. You can also change the number of stations that are brought up in the Multi-Station Throughput vs Pkt Size test by changing **Capacity Amounts (stations)**. Note that no station amount value may exceed the station count of any enabled test band.

F. When the configuration is complete, click the **Start** button (which will change to **Stop** once start is clicked) to start the test. An interactive report tab will be created and will be updated as the test runs.



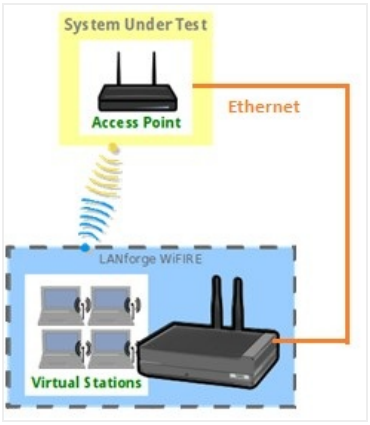
G. You can change the test result verbosity level by adjusting the **Verbosity** slider. Maximizing it will show all generated figures and data. The verbosity also affects the length of the saved report.

H. At the end of the test, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click **Save PDF** and the browser will be directed to open the pdf file directly. Please see this example AP-Auto Throughput Report.

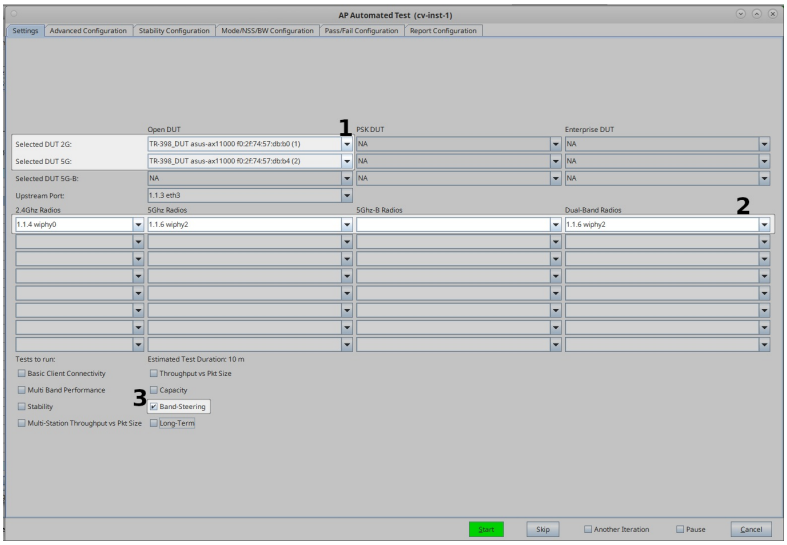
Testing AP Band Steering with the AP-Auto Automated Test Suite

Goal: Run an AP-Auto test for an AP using the LANforge CT523c or similar system in order to test how successfully an AP conducts band steering. The AP-Auto test is similar to the TR-398 test, but is designed to be functional with a minimum amount of test equipment. A 2-radio LANforge system and DUT is all that is required to run these tests.

In this test scenario, the LANforge CT523 is used to create stations and run the Band Steering test. This example assumes you have some experience with Chamber View, and that you have a LANforge system and a DUT AP. The AP and LANforge may be in chambers, but that is not required. This feature requires LANforge version 5.4.2 or higher.



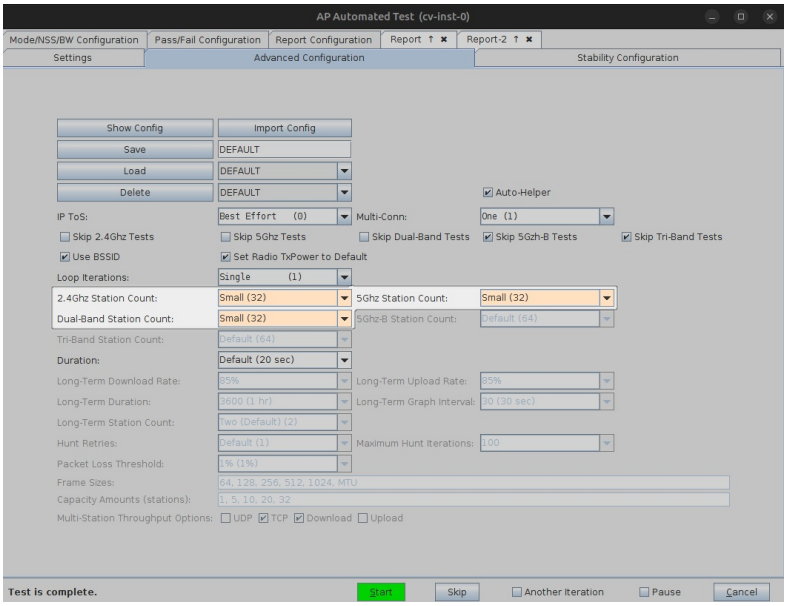
1. If you haven't setup or performed AP-Auto tests on your LANforge system, please refer to the AP-Auto Test Suite Setup guide for quick setup.
2. Running the AP-Auto Band Steering Test:
   A. Open the AP-Auto Test window.



   B. In the AP-Auto Settings Tab:

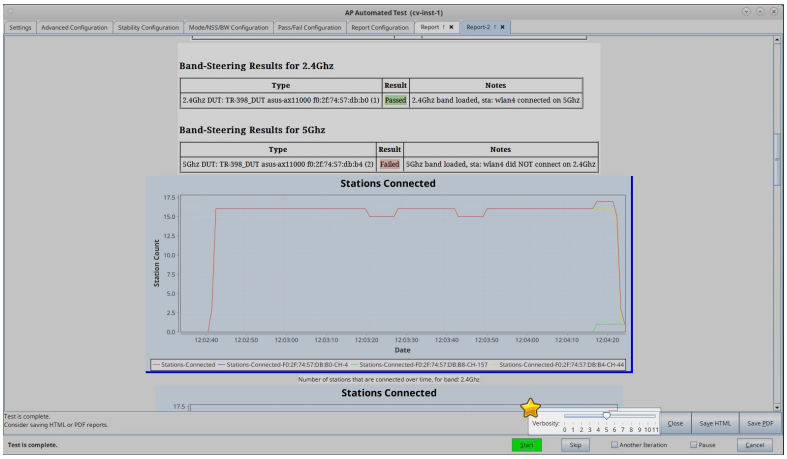
      A. Select the **DUT 2G** and **DUT 5G** SSIDs. **This test requires that** each DUT use one shared SSID across all bands. Most DUTs will default to this behavior once they have been configured for band steering.
      B. Select the LANforge radios to be used in this test. You need at least one 2.4Ghz radio, one 5Ghz radio, and one Dual-Band radio for full functionality.
      C. At the bottom, select the **Band-Steering** test checkbox.

C. Your Advanced Configuration tab should look similar to the following:



D. Modifying the band station count fields will have the most direct effect on the 'PASS/FAIL' results of this test (E.g. increasing station count on the loaded band might increase likelihood of DUT steering new clients to the less crowded band).

E. When the configuration is complete, click the **Start** button (which will change to **Stop** once start is clicked) to start the test. An interactive report tab will be created and will be updated as the test runs.



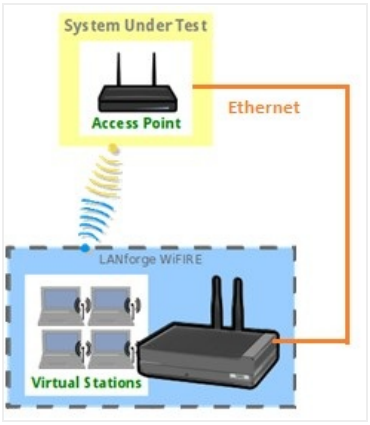
F. You can change the test result verbosity level by adjusting the **Verbosity** slider. Maximizing it will show all generated figures and data. The verbosity also affects the length of the saved report.

G. At the end of the test, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click **Save PDF** and the browser will be directed to open the pdf file directly. Please see this example AP-Auto Band-Steering Report.

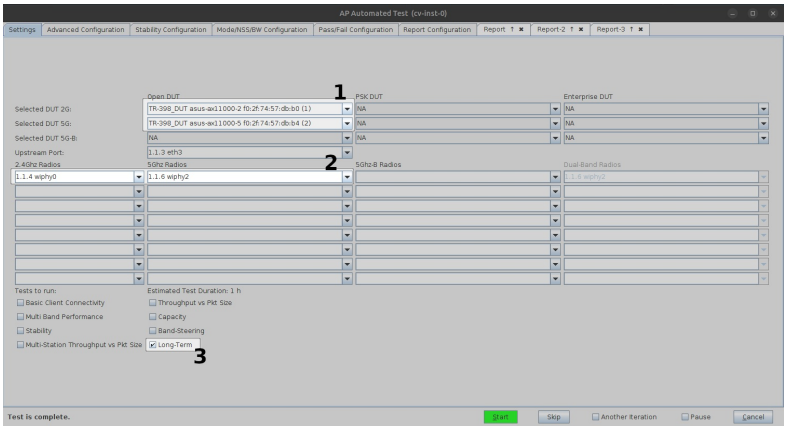
Testing AP Long-Term Performance with the AP-Auto Automated Test Suite

Goal: Run an AP-Auto test for an AP using the LANforge CT523c or similar system in order to test how well the AP can long-term traffic. The AP-Auto test is similar to the TR-398 test, but is designed to be functional with a minimum amount of test equipment. A 2-radio LANforge system and DUT is all that is required to run these tests.

In this test scenario, the LANforge CT523 is used to create stations and run the Long-Term test. This example assumes you have some experience with Chamber View, and that you have a LANforge system and a DUT AP. The AP and LANforge may be in chambers, but that is not required. This feature requires LANforge version 5.4.2 or higher.



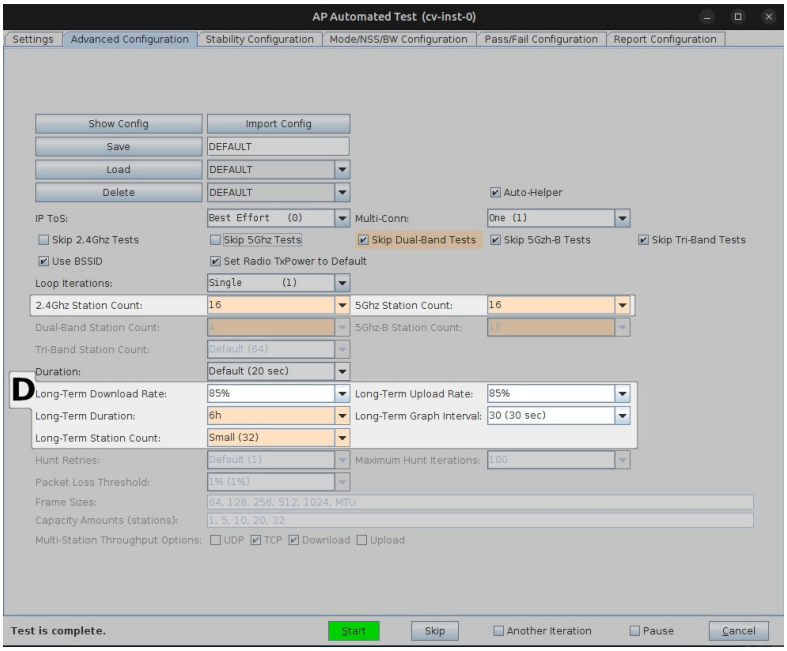
1. If you haven't setup or performed AP-Auto tests on your LANforge system, please refer to the AP-Auto Test Suite Setup guide for quick setup.
2. Running the AP-Auto Long-Term Test:
    A. Open the AP-Auto Test window.



    B. In the AP-Auto Settings Tab:

        A. Select the **DUT 2G** and **DUT 5G** SSIDs. This test requires that Open or PSK SSIDS are filled out.
        B. Select the LANforge radios to be used in this test. You need at least one 2.4Ghz radio and one 5Ghz radio for full functionality.
        C. At the bottom, select the **Long-Term** test checkbox.

C. Your Advanced Configuration tab should look similar to the following:



D. Highlighted are the Long-Term test settings. Note, stations are brought up on multiple bands concurrently in this test and so the **Long-Term Station Count** value will determine the sum total of stations across all bands.

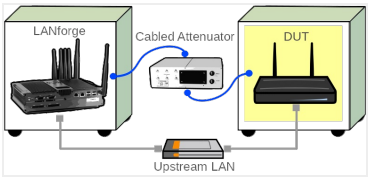
E. When the configuration is complete, click the **Start** button (which will change to **Stop** once start is clicked) to start the test. An interactive report tab will be created and will be updated as the test runs.

F. You can change the test result verbosity level by adjusting the **Verbosity** slider. Maximizing it will show all generated figures and data. The verbosity level also affects the length of the saved report.

G. At the end of the test, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click **Save PDF** and the browser will be directed to open the pdf file directly. Please see this example AP-Auto Long-Term Report.

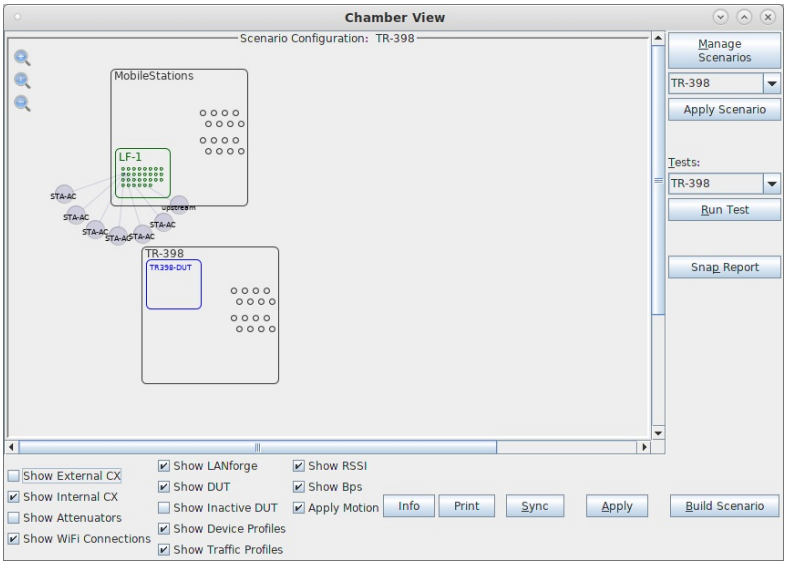
Testing AP with automated TR-398 scenario

Goal: Setup and run a TR-398 test for an AP using a 6-radio version of the LANforge CT523c or similar system in order to test how well the AP can handle the various test cases specified in the TR-398 test document.

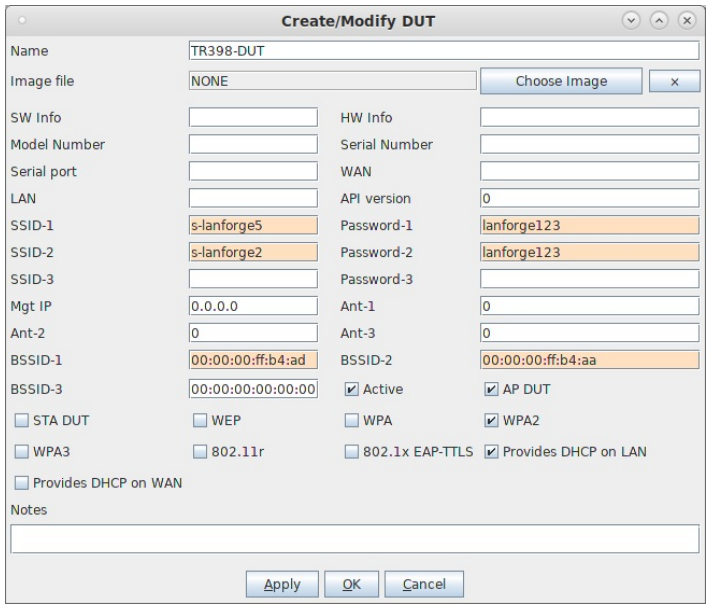


In this test scenario, the LANforge CT523c is used to emulate different station and AP scenarios and generate and receive traffic through an AP. This example assumes you have some experience with Chamber View, and that you have an appropriate LANforge system, programmable attenuators like the CT714 and two isolation chambers like the CT820a and CT840a. The AP should be in the chamber with the 2D turntable, the LANforge system is in the other chamber, and the attenuators are cabled between them. This feature requires LANforge version 5.4.1 or higher and the LANforge TR-398 automation software license.

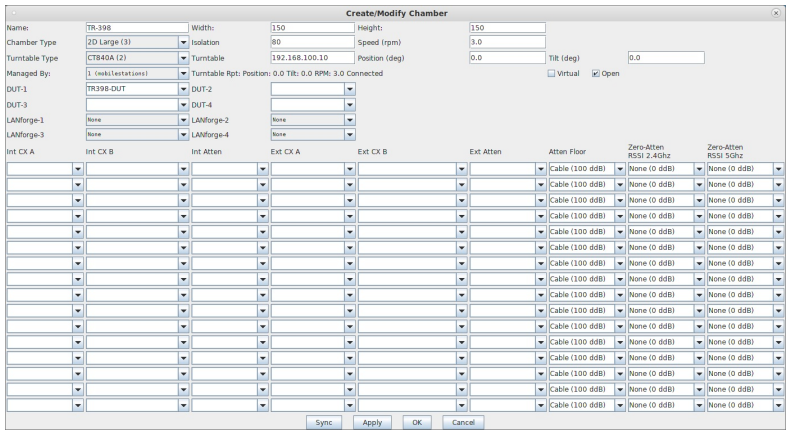
1. Configure Chamber View for TR-398 and Similar Tests.
    A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects. In this example, the 'Mobilestations' chamber holds the LANforge, and the TR-398 chamber holds the Device Under Test. Your configuration should look like this when these steps are complete.



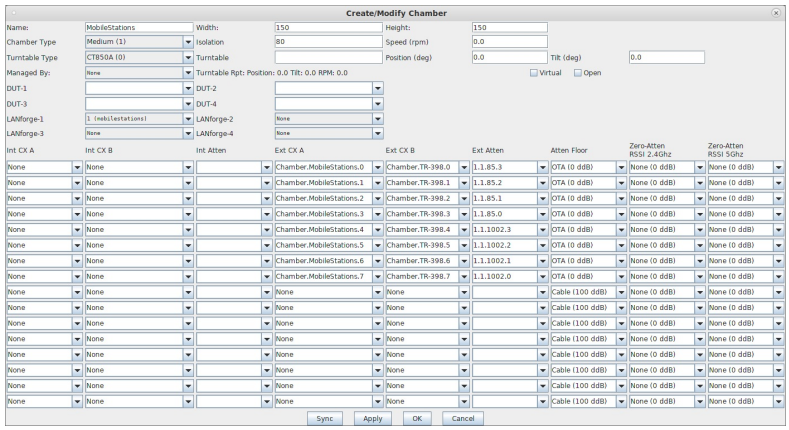
B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.



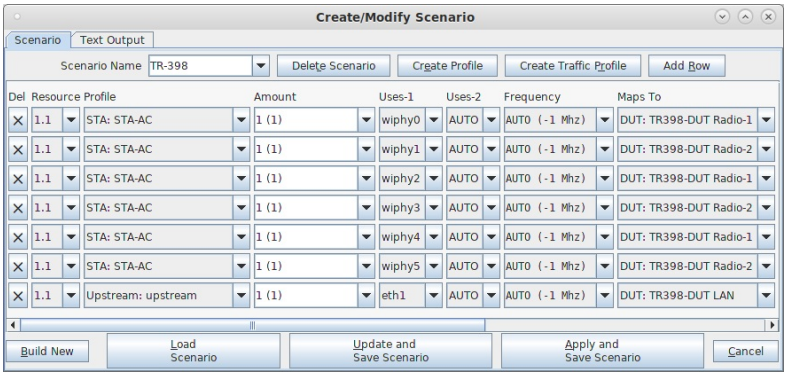
C. Create a chamber object to hold the DUT, and add the DUT to that chamber. If you have no chambers, you can create a fake chamber, but your test will not be isolated and may not function as desired. The turntable configuration is different for different models of chambers, this example (bare IP address) is for the CT840a chamber.



D. Create a chamber object to hold the LANforge system, and add the LANforge to it. Add connections from this chamber to the DUT chamber, specifying the proper Attenuator modules. Please note we use the 'OTA' attenuation floor since we have OTA connection between DUT and antennas inside the DUT chamber. Please view our other cookbook on setting up attenuator connections in LANforge.

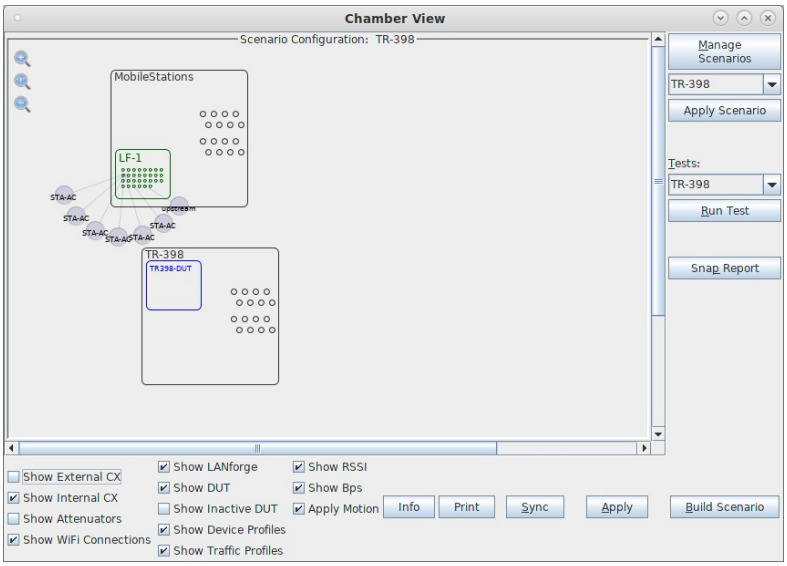


E. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).

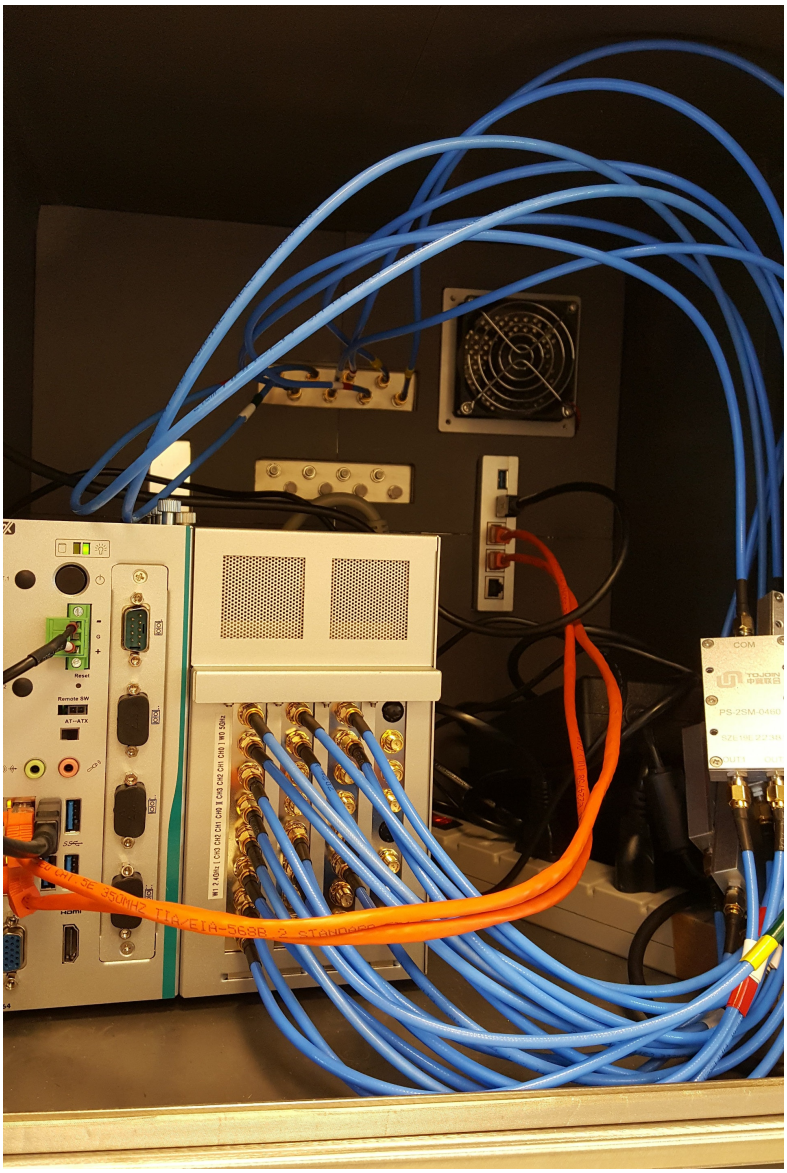


2. Use Chamber View for TR-398 test.

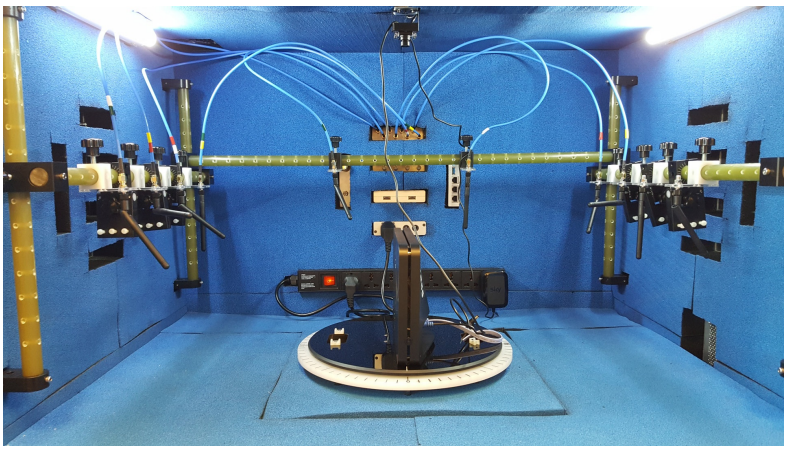
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario. Apply the Scenario, then Build the scenario.



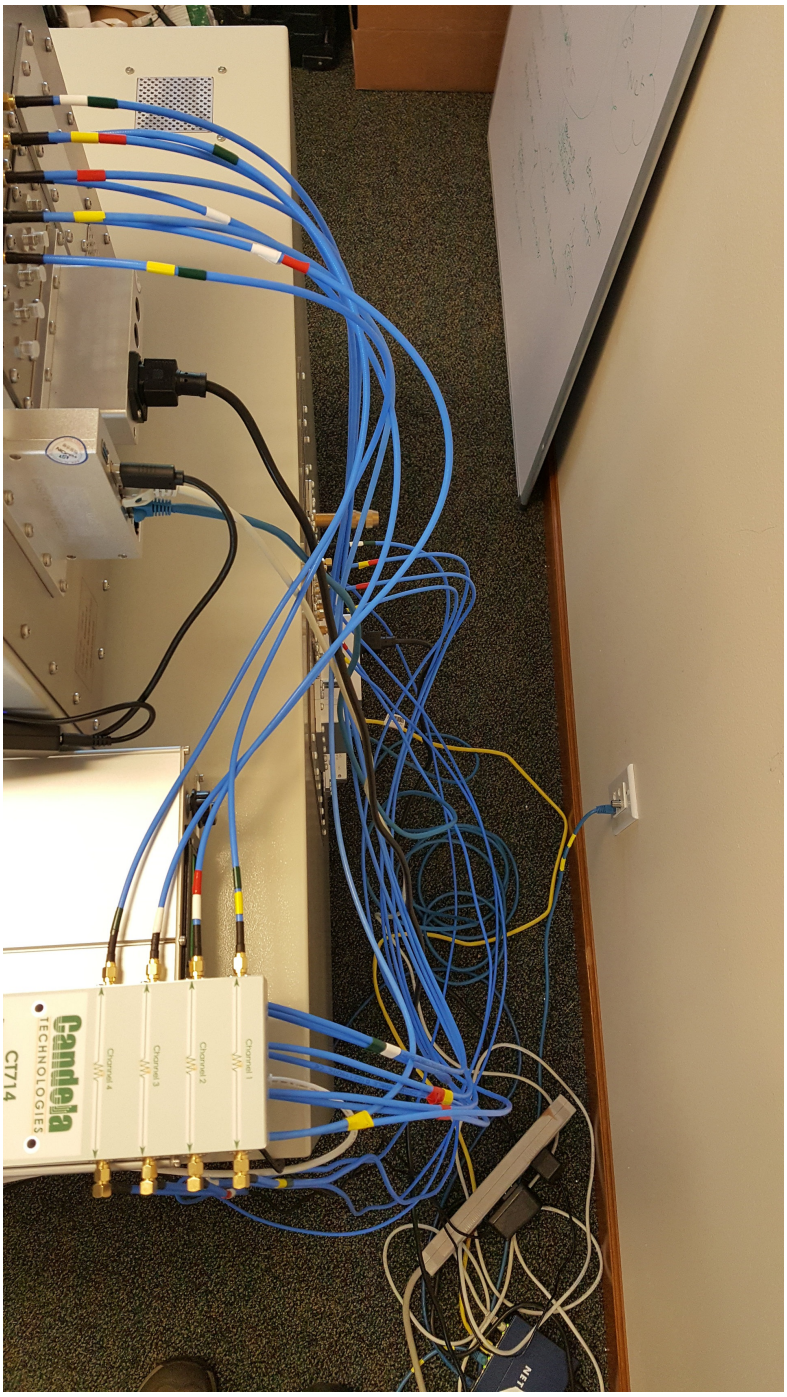
B. To help make the cable routing easier to understand, you may wish to color-code the cables with tape or other similar markings.. Inside of the LANforge chamber:



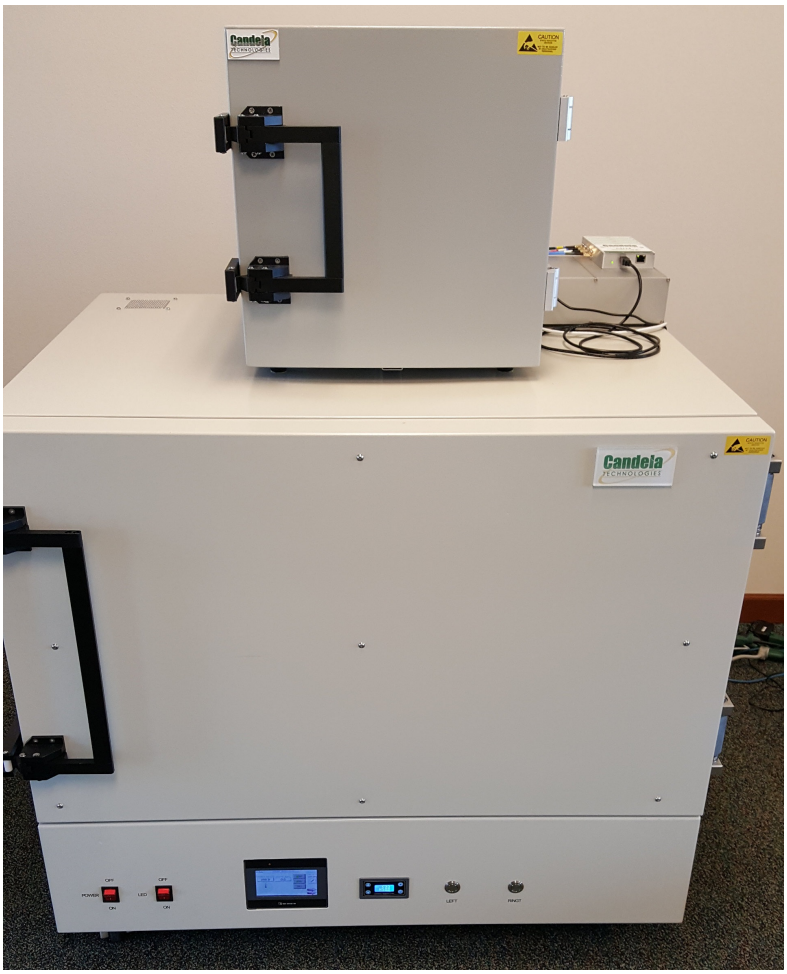
C. Inside of the DUT chamber:



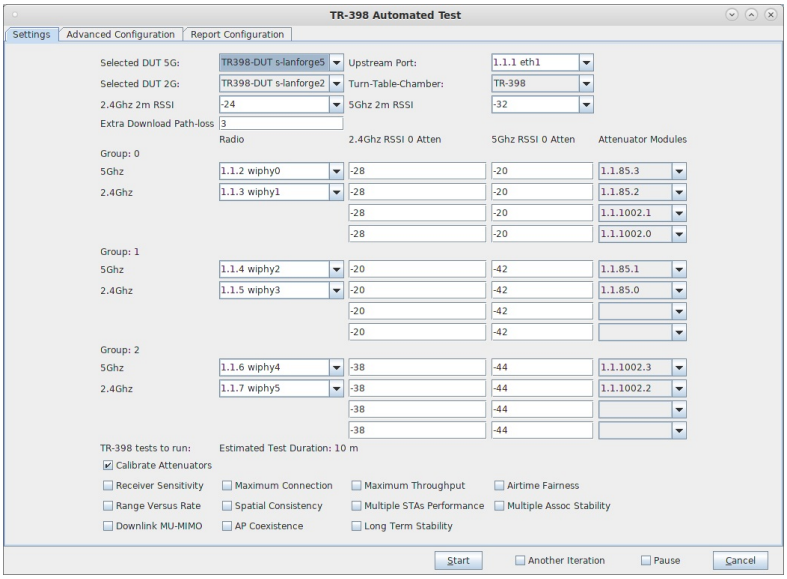
D. Outside, back view:



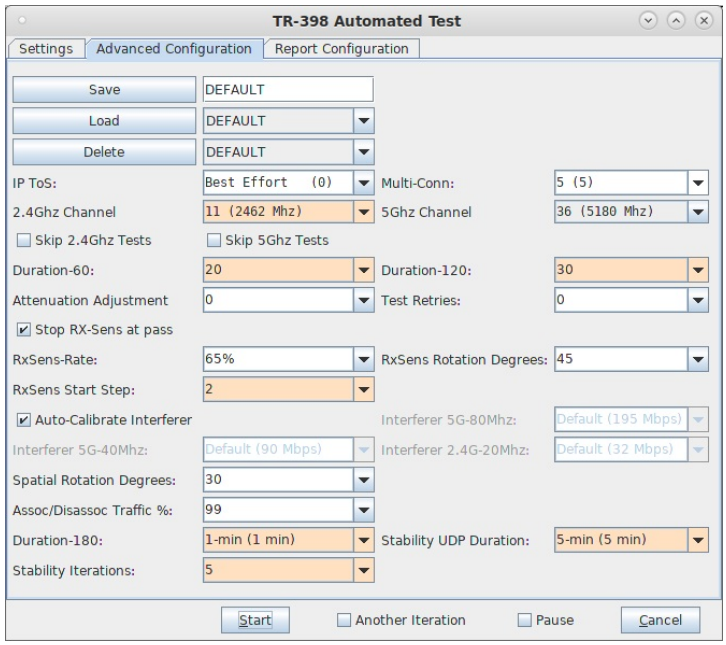
E. Front view, all closed up and ready to test:



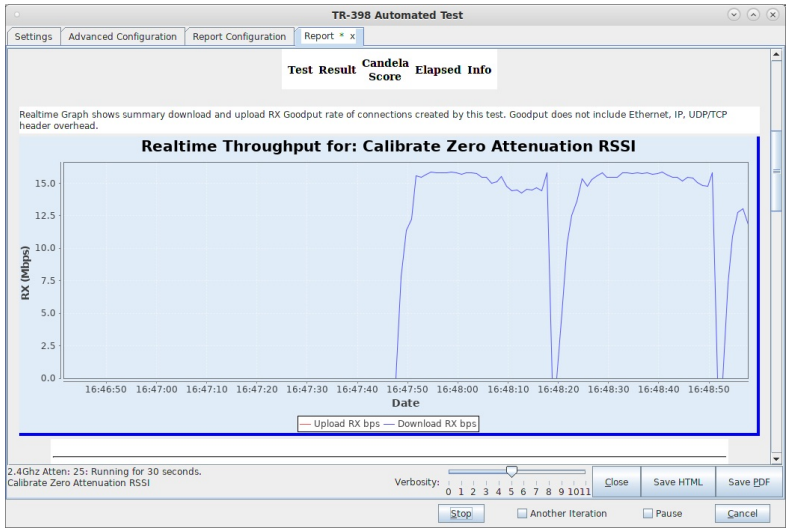
F. Select the **TR-398** test and click **Run Test**. You should see the TR-398 Test configuration window pop up. It will remember the last configuration for most fields. Select the 2.4 and 5Ghz DUT, turn-table chamber, and carefully associate the radios with correct attenuator modules. For the first run, we will just run the calibration test to auto-configure the path loss by adjusting the attenuators and testing RSSI:



G. Go to the Advanced Configuration tab andmake sure the channels are configured properly, and tune any other settings that you wish to modify.:



H. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report tab will be created and will be updated as the test runs. For the first run, we will just run the calibration test to auto-configure the path loss by adjusting the attenuators and testing RSSI. If the calibration was acceptable, click Start again to save the calculated zero-RSSI attenuation values for future test runs in which you may decide to skip re-running the attenuation:

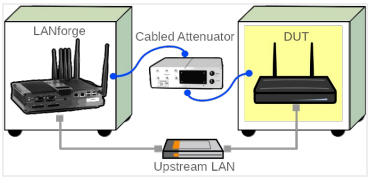


I. After the calibration step is complete, select the desired set of test cases and click Start. The test may run for many hours. If you have the webcam configured, you can watch the rotational tests happen using the 'camorama' program (in Linux), or your own preferred USB webcam program. If using the LANforge machine to view camarama, please set the size to small so that it uses less CPU. When you click start, the Report tab is created, you can view that tab to watch the progress.

J. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example TR-398 Test Report.

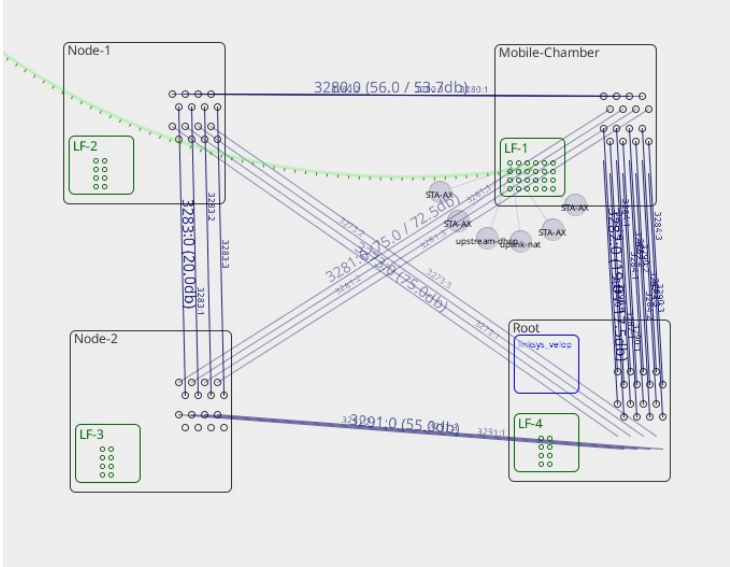
Calibrating TR-398 Issue-2 and Mesh on a 4-Chamber Setup

Goal: Setup and run a TR-398 Issue 2 test for an AP using a 6-radio version of the LANforge CT523c or similar system in order to test how well the AP can handle the various test cases specified in the TR-398 Issue 2 test document.

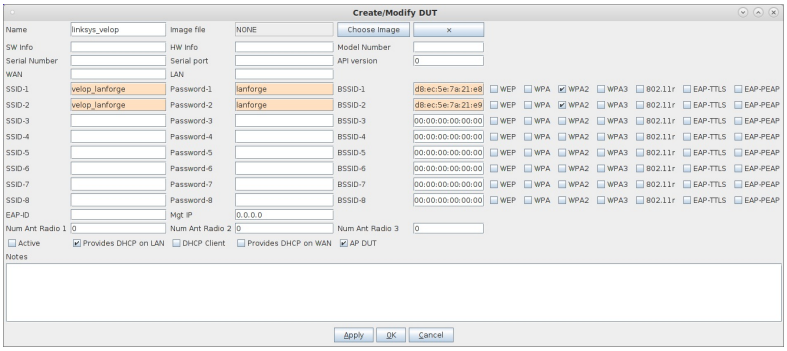


In this test scenario, a LANforge cluster (of a 1 523c and 3 521as) is used to emulate different station and AP scenarios and generate and receive traffic with an AP. This example assumes user has some experience with Chamber View, and has an appropriate LANforge system (fit for TR-398 Issue 2), programmable attenuators like the CT714 and some isolation chambers like the CT820a and CT840a. Please contact support@candelatech.com for assistance in setting up the TR-398 testbed.

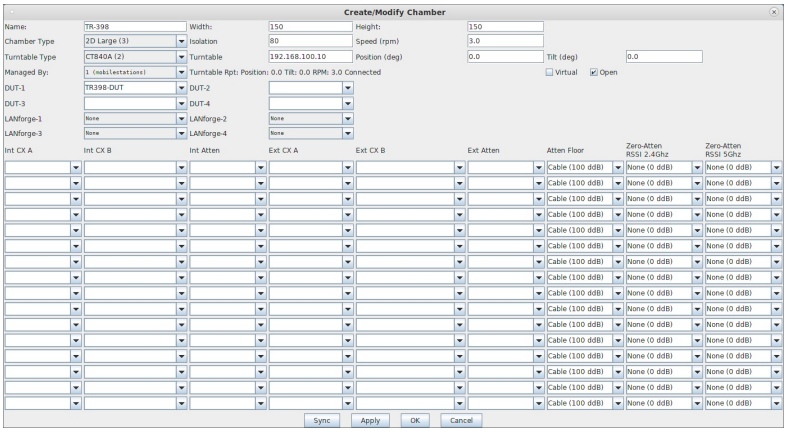
1. Configure Chamber View for TR-398 and Similar Tests.
   A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, please skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects. This cookbook will use the Chamber View scenario seen below.



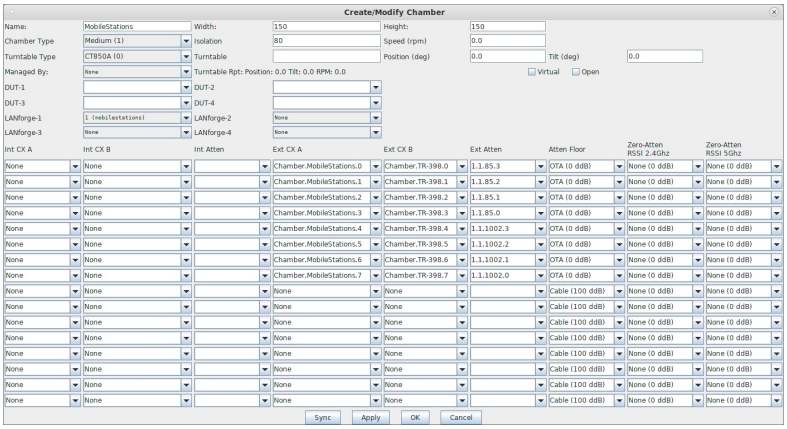
   B. Create a Device Under Test (DUT) Profile that matches your AP. The BSSID is important to configure so that LANforge knows when it is connected to the correct AP. In a mesh scenario, you will create 3 DUT objects, one for each of the mesh AP chambers.



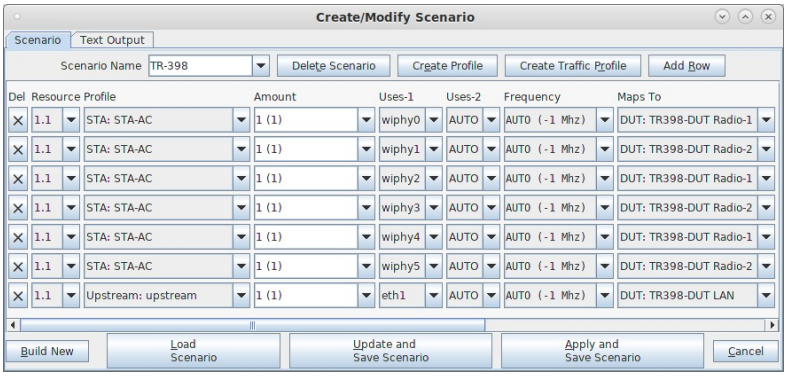
C. Create a chamber object to hold the DUT, and add the DUT to that chamber. If you have no chambers, you can create a fake chamber, but your test will not be isolated and may not function as desired. The turntable configuration is different for different models of chambers, this example (bare IP address) is for the CT840a chamber.



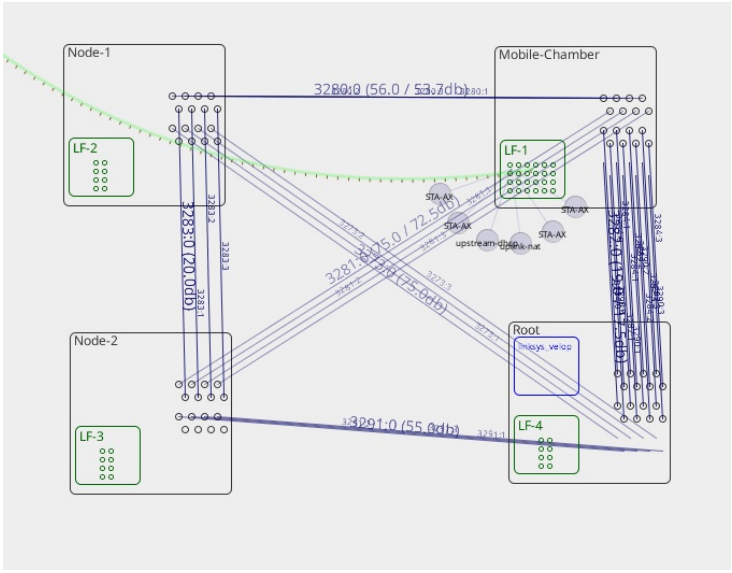
D. Create a chamber object to hold the LANforge system, and add the LANforge to it. Add connections from this chamber to the DUT chamber, specifying the proper Attenuator modules. Please note we use the 'OTA' attenuation floor since we have OTA connection between DUT and antennas inside the DUT chamber. Please view our other cookbook on setting up attenuator connections in LANforge.



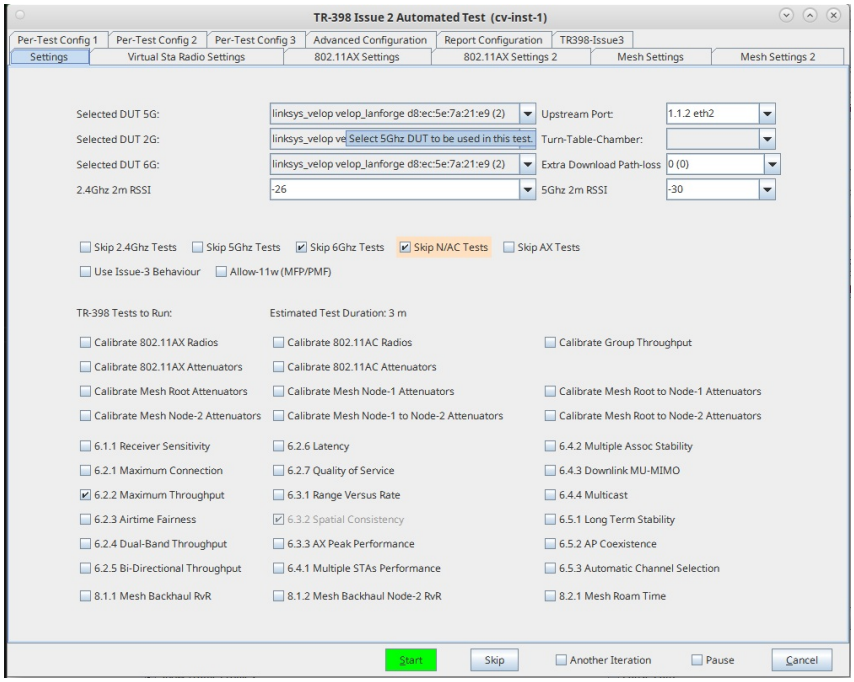
E. Configure a Chamber View Scenario and add the STA profile (mapped to desired wiphyX radio and DUT). Add an upstream profile mapped to DUT LAN side (or possibly WAN side if that is more appropriate for your DUT).



2. For TR398 tests, the DUT AP's front should face the antennas for the Group-1 stations. Or, optionally, one can use the Advanced configuration tab to specify the default turntable angle for non rotational tests.
3. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario. Apply the Scenario, then Build the scenario.

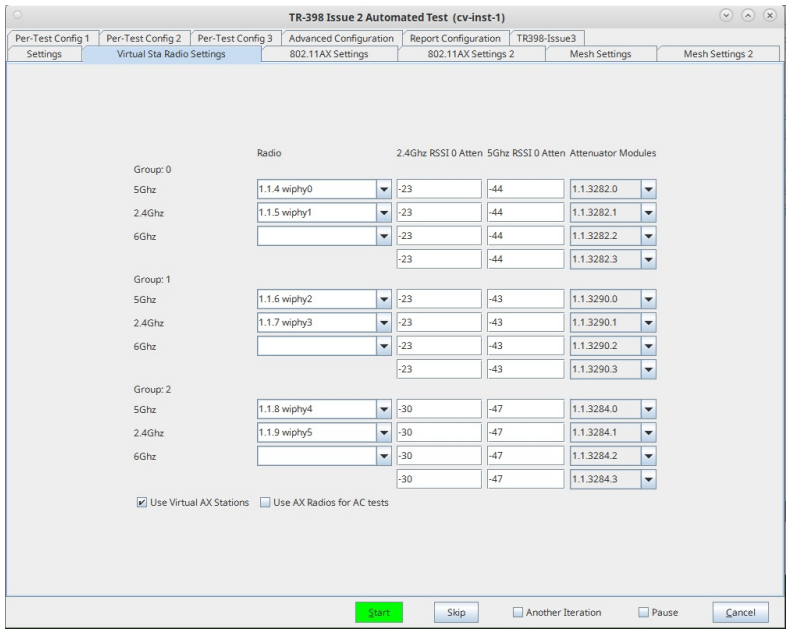


4. Select the **TR-398-Issue-2** test from Tests dropdown in the right panel. Then click Run Test. The following window below should pop up. In the Settings tab, set slots 'Selected DUT 5G', 'Selected DUT 2G', and 'Selected DUT 6G' the according BSSIDs found in the 'Root' Chamber. Also set the upstream port. In this example, the upstream port is the eth2 port on the LANforge in Mobile Stations chamber.

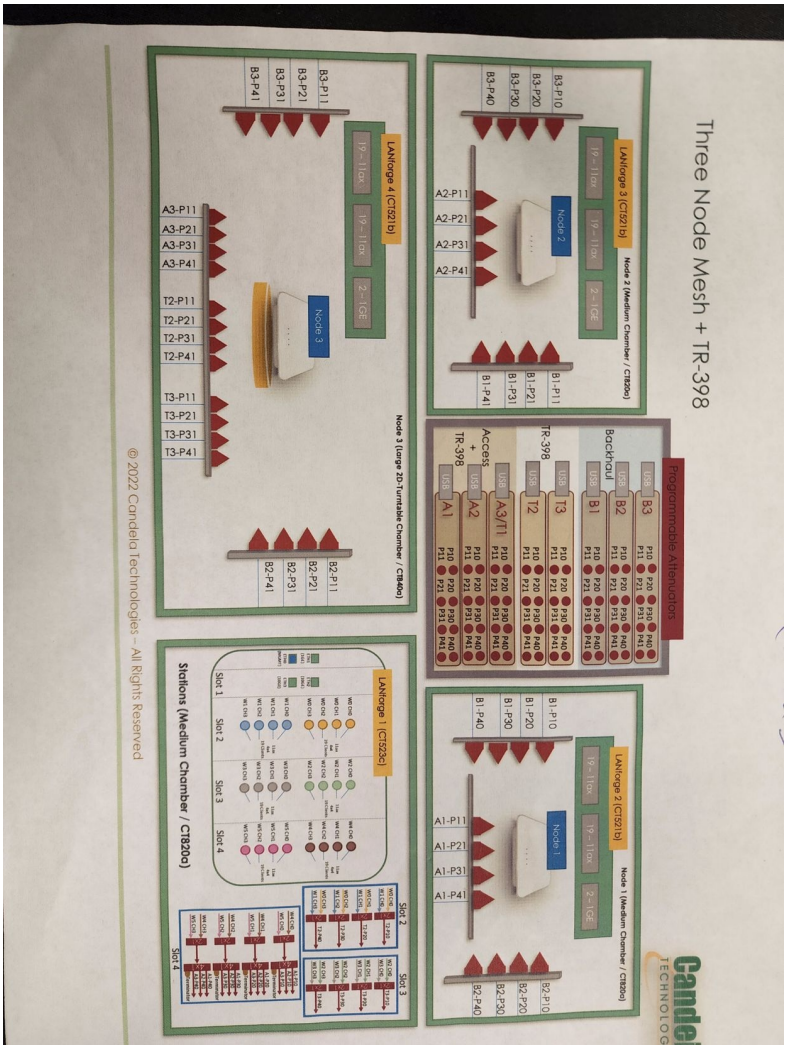


5. Configure the Virtual Sta Radio Settings tab.

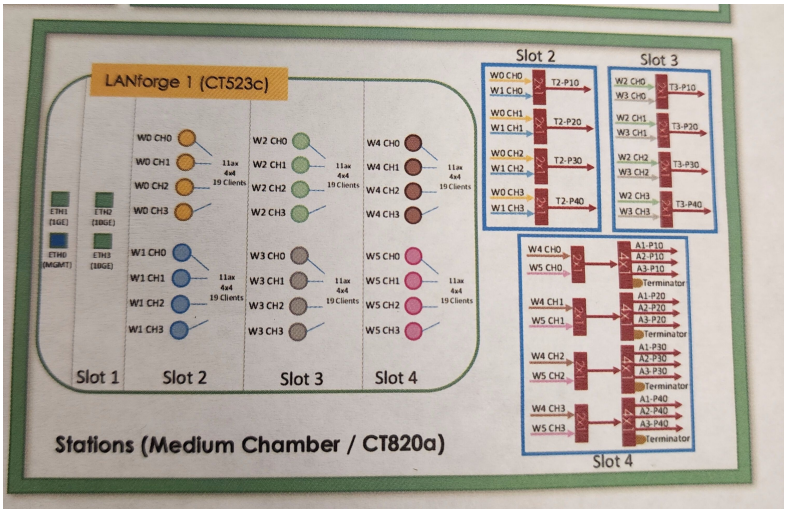
A. Select the Virtual Sta Radio Settings tab. This tab is for radios that can have virtual stations on them (either AC, N, some AX). Notice that there are 3 groups here, this will be the 3 slots of radios (Slots 2, 3, 4) in the Mobile Clients Chamber. Visit the pictures in the next 2 steps to see close ups of the attenuator diagram that has the slots. Slot 2 (Group 0) has W0 ad W1 (wiphy0 and wiphy1). Slot 3(Group 1) has W2 and W3 (wiphy2 and wiphy3). Slot 4 (Group 2) has W4 and W5 (wiphy4 and wiphy5). Leave the '2.4Ghz RSSI 0 Atten' and '5Ghz RSSI 0 Atten' blank, those will be auto calculated when we calibrate the attenuators. Lastly, fill out the attenuator modules column (with the correct serial numbers that relate to each group). All these attenuators will be the TR-398 section of the attenuator stack. In this case, those are attenuators T1, T2 and T3. These attenuators are also all connecting the **Node 3/Root** chamber to the **Mobile Clients** chamber. Match the according attenuator to each group on the attenuator diagram. Fill out all the dropdowns under 'Attenuator Modules' for ports 0-3 on the matching attenuator. Select the 'Use Virtual AX Stations' checkbox.



B. Below is the attenuator diagram of the 4-chamber setup in the example. Another name for the Node-3 Chamber is the Root chamber.



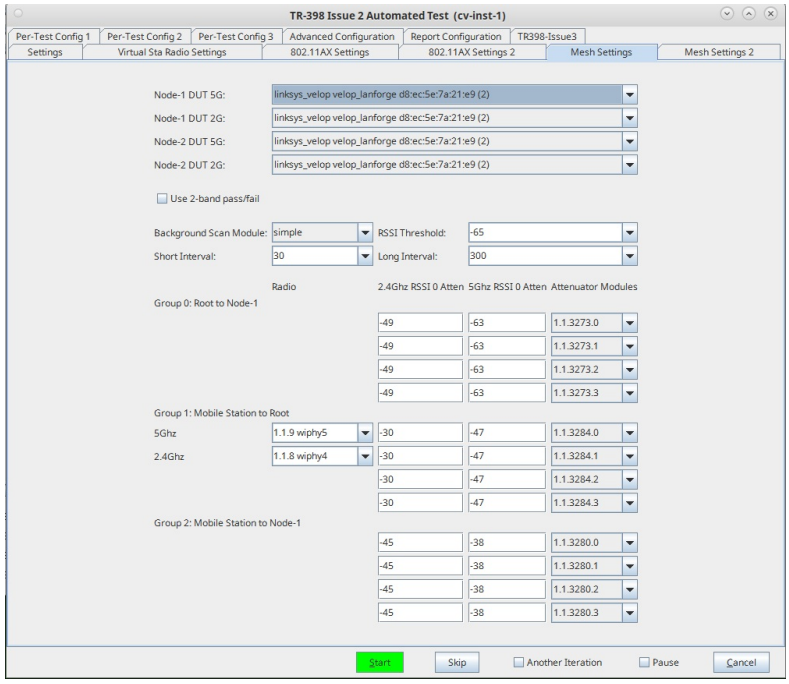
C. Below is a closer look at the bottom right box of the attenuator diagram in the step above.



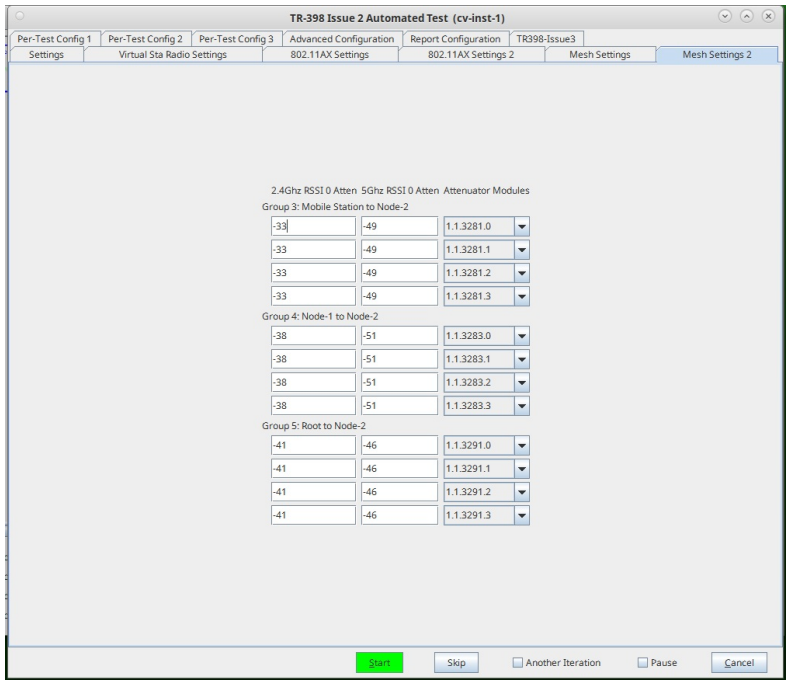
D. Below are the serial numbers of the attenuator stack. They are stacked in the same order as the attenuator diagram.

6. Configure the 802.11 AX Settings (1 & 2) tabs. Please contact support@candelatech.com for assistance in filling out these tabs. These settings only apply to AX capable radios that do not support virtual stations.
7. Configure the Mesh Settings and Mesh Settings 2 tabs. If the test bed being setup has no need for Mesh, please skip this step.

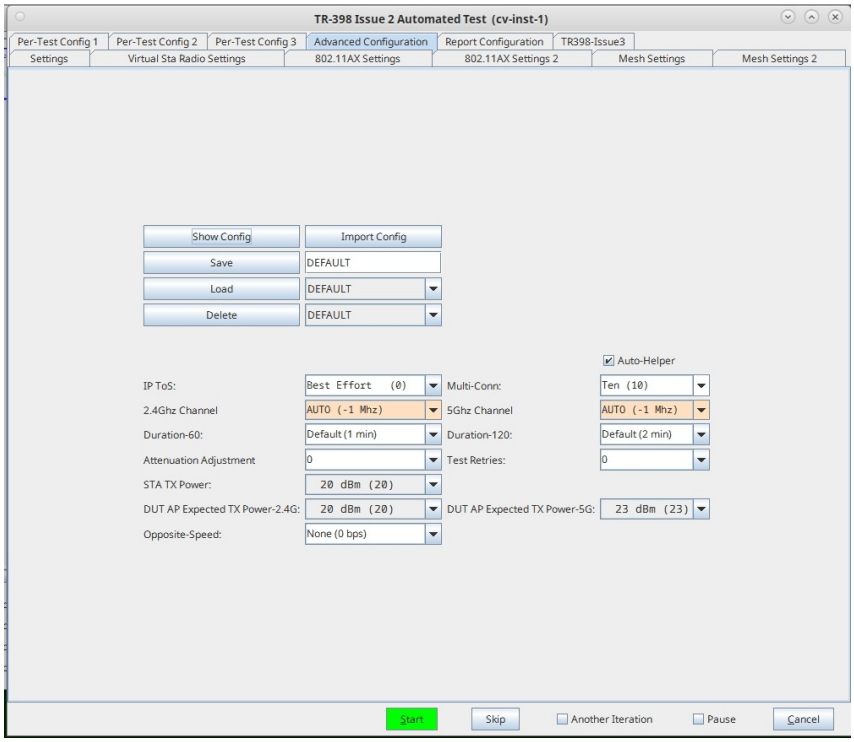
A. Select the Mesh Settings tab. The picture below is a final version of the calibrated mesh setup. In an un-calibrated version (yours), all the rows in the 'RSSI 0 Atten' columns will be empty. Fill out select the 2.4Ghz and 5Ghz BSSIDs for both Node 1 and Node 2 Chambers in the top section. Fill out the 'Atten Modules' column for each group (using ports 0-3 for each attenuator). To find out the correct attenuator, trace the attenuator diagram to find out which attenuator connects which 2 Chambers. In Group 1 (in this section), use the attenuator and radios used in 'Group 2' of the Virtual Sta Radio Settings tab. This attenuator is both TR398 and Access (A3/T1), so it belongs in both Virtual Sta Radio Settings and Mesh Settings



B. Fill out the Mesh Settings 2 tab similarly to Mesh Settings 1

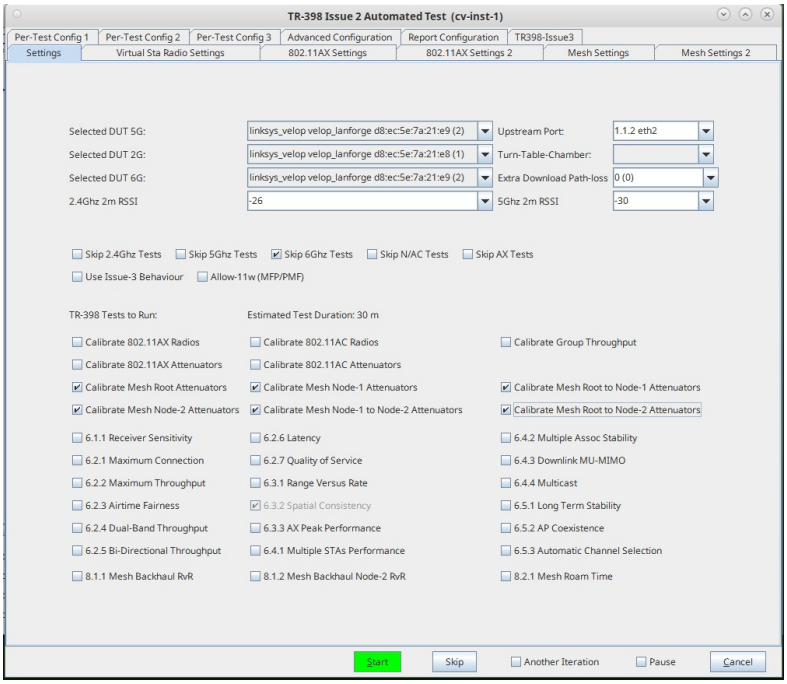


8. Click on the Advanced Configuration tab. Set 2.4GHz channel and 5GHz channel to '-1'/ AUTO. AUTO won't work for all cases (like the ap-coex test), but will work for most of them.

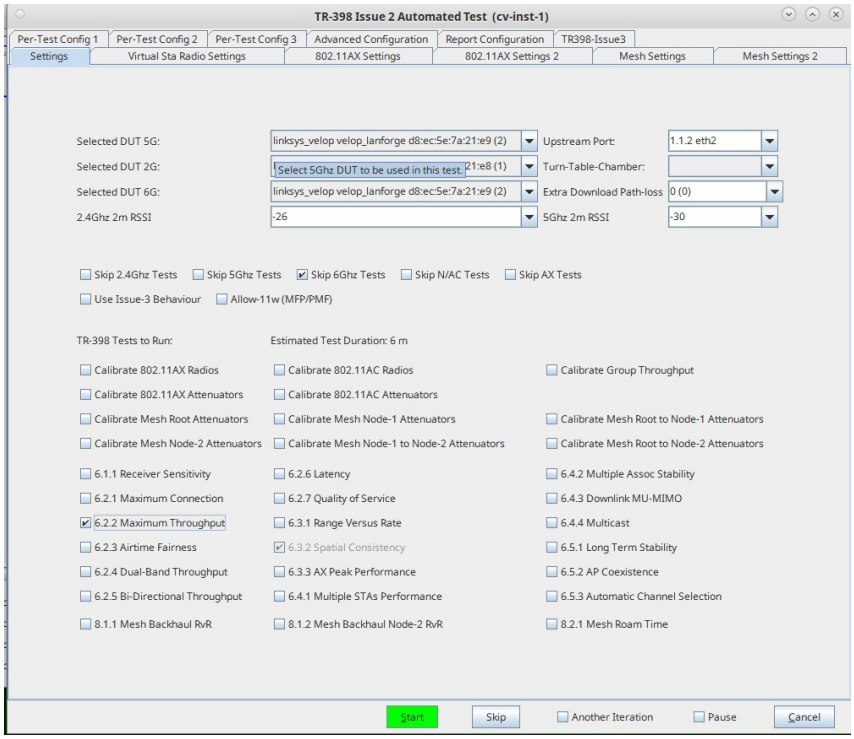


9. Calibrate Zero Attenuation RSSI for all chamber to chamber connections. This also verifies attenuators are connected and functioning as expected.
    A. Start by setting the LANforge in the center of all the chambers, with all the antennae pointing directly to the LANforge. We are calculating 0 atten against the LANforge.
    B. Calibrate the 802.11AC attenuators. Only select this checkbox to ensure that the calibration goes OK.

    C. In the image below, run each of the checkboxed boxes **one** at a time, until all are run.



10. At this point, all the 0 Atten values should be filled out (In Virtual Sta Radio Settings and Mesh Settings 1 & 2 tabs).
11. Run a TR-398 Issue 2 throughput test. Select the 6.2.2 Maxiumum Throughput checkbox to run the test. After verifying the throughput test is working as expected, select and run other tests as desired.

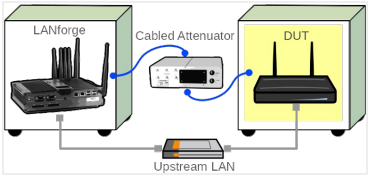


12. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. Another option is to click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this passing example TR-398 Issue 2 Maximum Throughput Test Report .

Advanced configuration options for TR-398 Issue-2/3 and Mesh on a 4-Chamber Setup

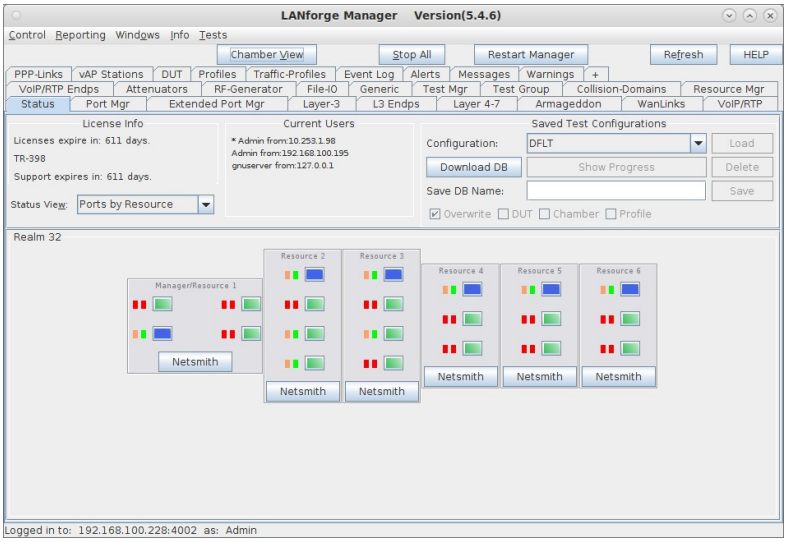
Goal: Explore advanced configuration options for a TR398 issue-3 mesh testbed. The testbed in this example supports both 32 real tri-band radios as well as 6 virtual-station dual-band radios. See the TR398 testbed page for more information.

In this test scenario, a LANforge cluster (of three 523c and 3 521b systems) is used to emulate different station and AP scenarios and generate and receive traffic with a set of 3 meshed APs. This example assumes user has some experience with Chamber View and TR398, and has an appropriate LANforge TR398 testbed. Please contact support@candelatech.com for assistance in setting up the TR-398 testbed.

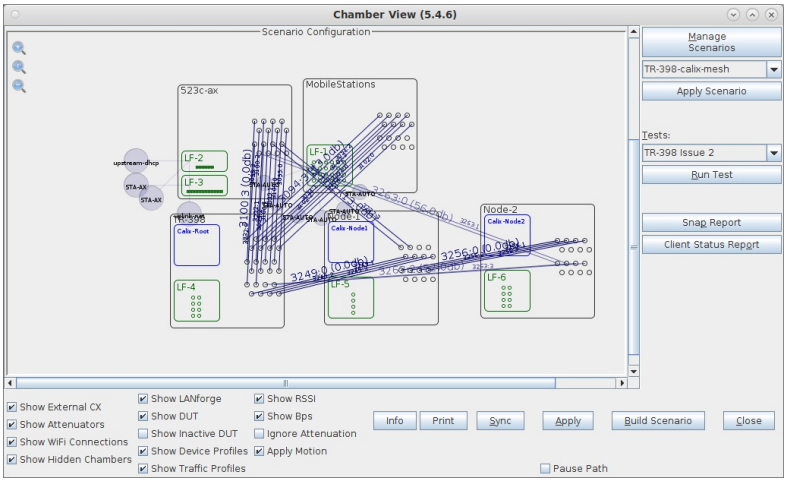


1. System configuration overview.

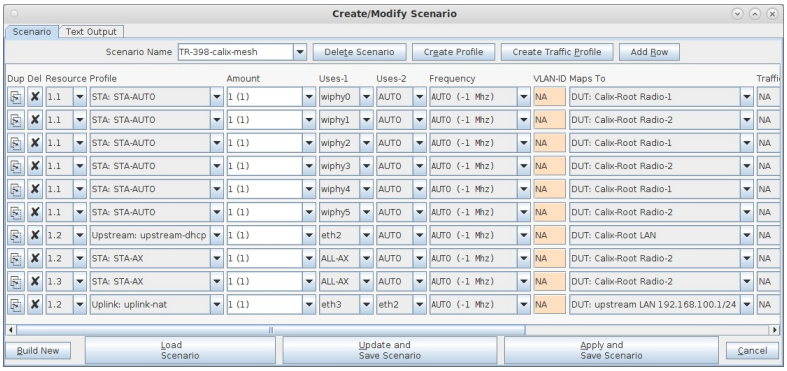
A. The status panel shows all 6 LANforge systems connected and available.



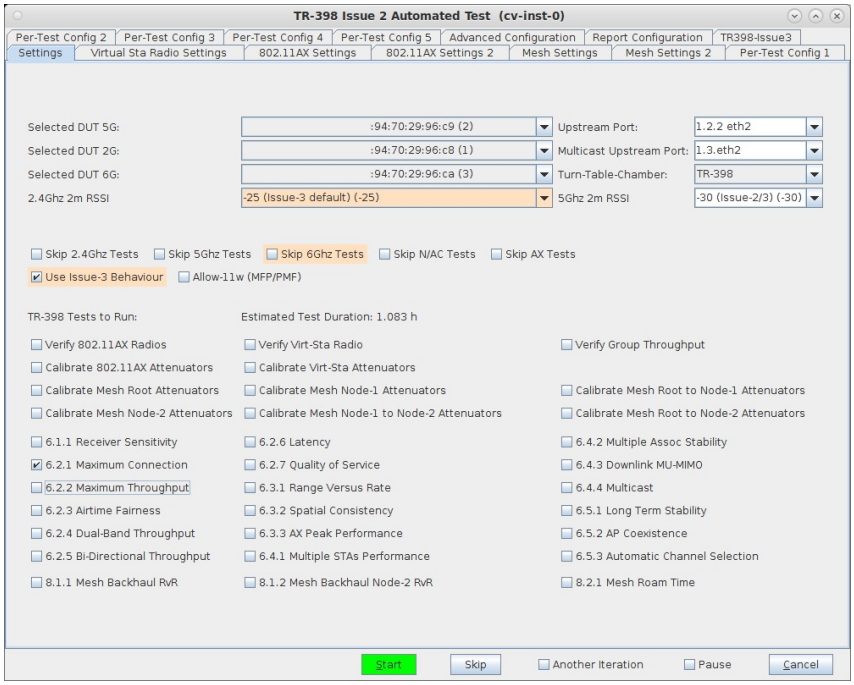
B. The Chamber View window shows the RF chambers, attenuated paths, APs, and LANforge systems.



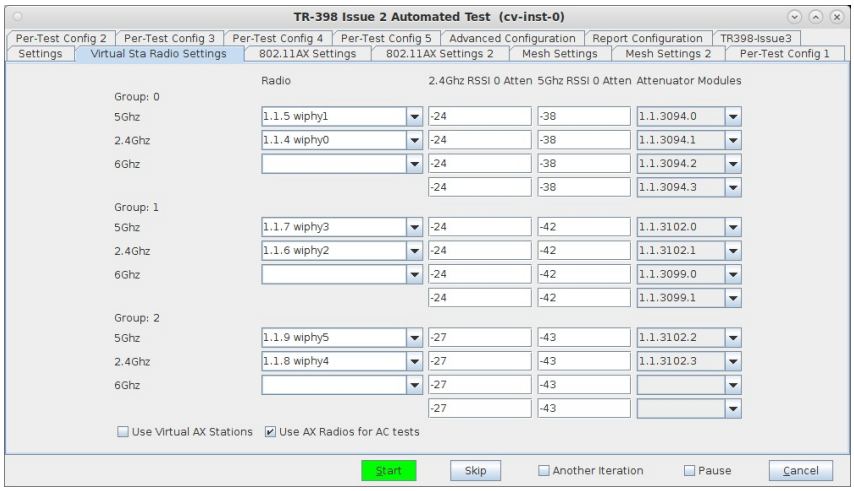
C. This setup uses LANforge as the upstream (WAN) port for the APs. This allows the testbed to be isolated from the lab network, and also facilitates running the APs in either bridge or routed mode.



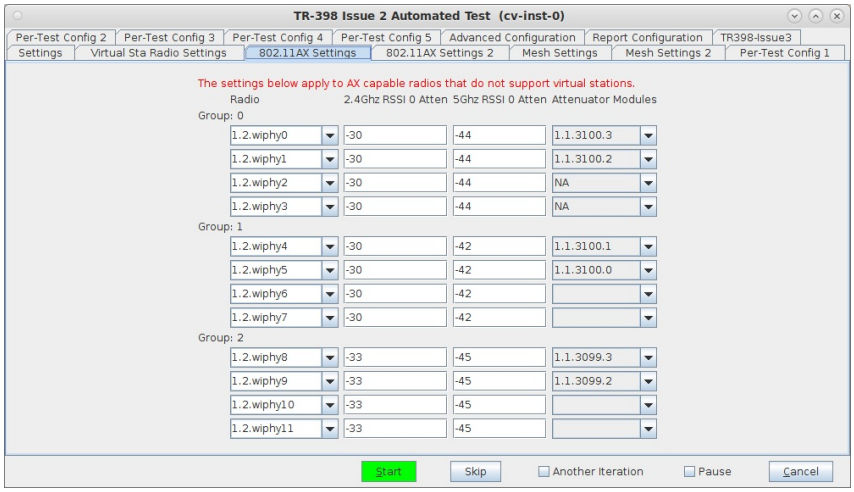
2. Launch the TR398 issue 2 test. Unless otherwise specified, this test is configured for TR398 issue3. The settings may be changed to run the TR398 issue 2 tests as user prefers.



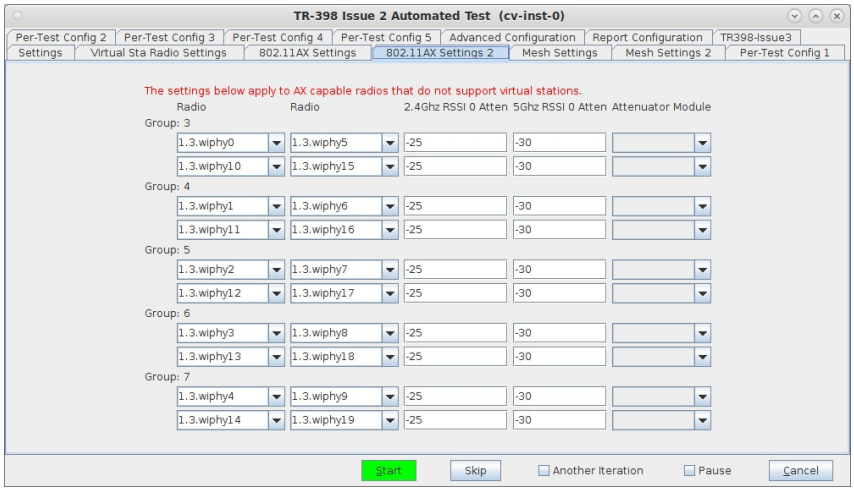
3. The Virtual Sta Radio settings tab is for the radios supporting virtual stations (MTK7915 4x4 dual-band AX in this example). The zero-attenuation RSSI values are calibrated during testbed configuration. See TR398 calibration cookbook for instructions on how to do this.



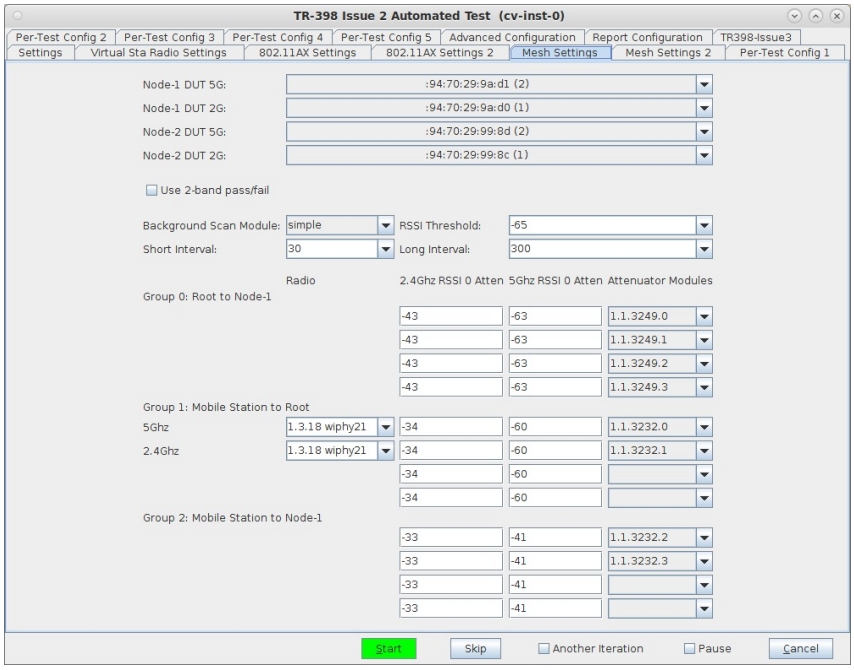
4. The 802.11AX Settings tab is for configuring non-virtual-station radios, Intel AX210 in this case.



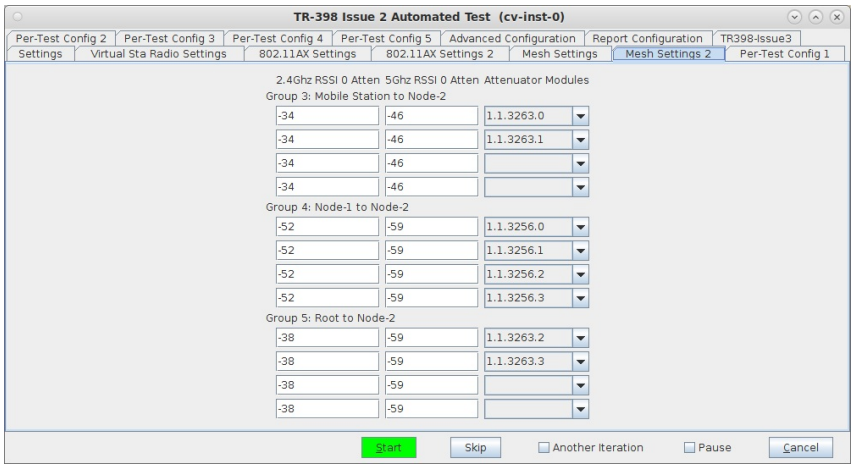
5. The 802.11AX Settings 2 tab is for configuring additional non-virtual-station radios, Intel AX210 in this case. Some testbeds will have different attenuators for groups 2-7. The testbed in this example uses splitter-combiners so that all radios in groups 2-7 go through the same attenuator, so there are no attenuators configured on this second AX settings tab.



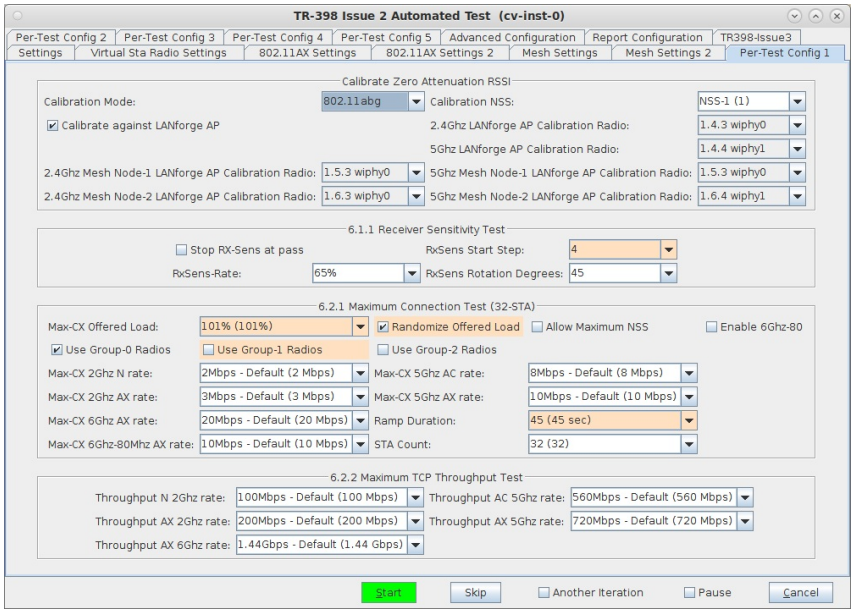
6. The Mesh Settings tab has information needed to run the Mesh test cases. This includes the DUT information for the node-1 and node-2 mesh chambers, attenuator mappings between chambers, and some scanning configuration.



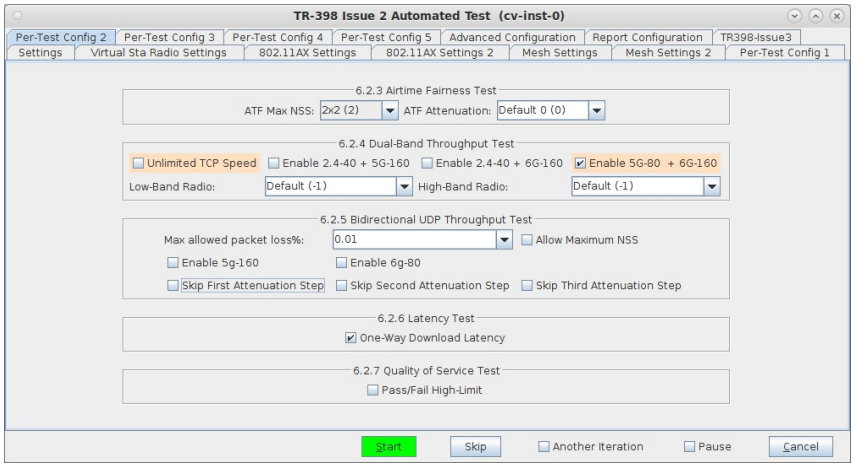
7. The Mesh Settings 2 tab has additional mapping information needed for a 3-node mesh testbed.



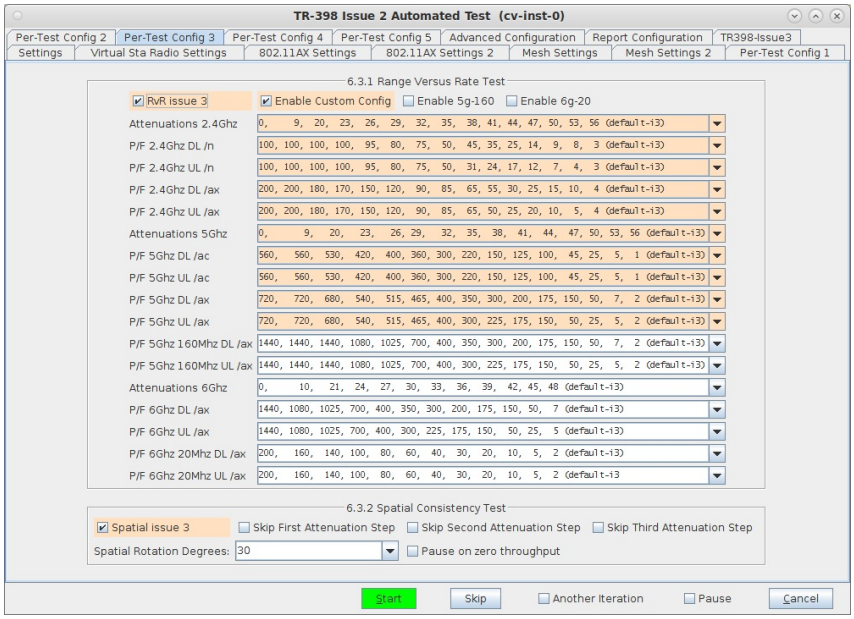
8. The Per-Test Config tabs provide tunable settings for individual test cases. Each entry field will have a tool-tip to explain what it does in more detail. The sections will only be active if the respective test case is selected on the Settings tab.



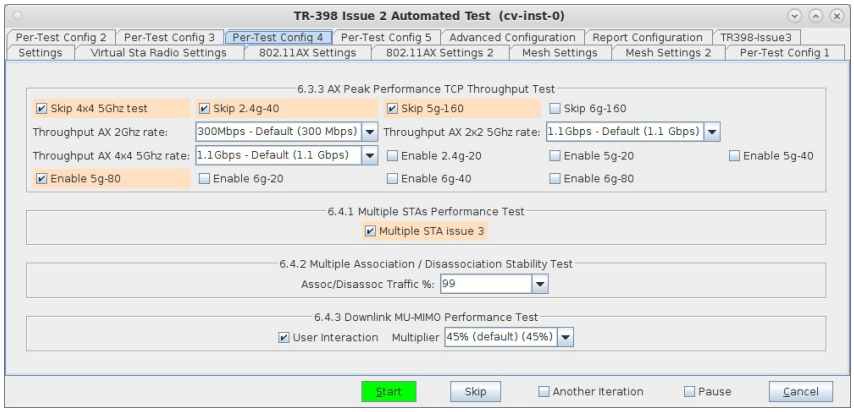
9. The Per-Test Config 2 tab.



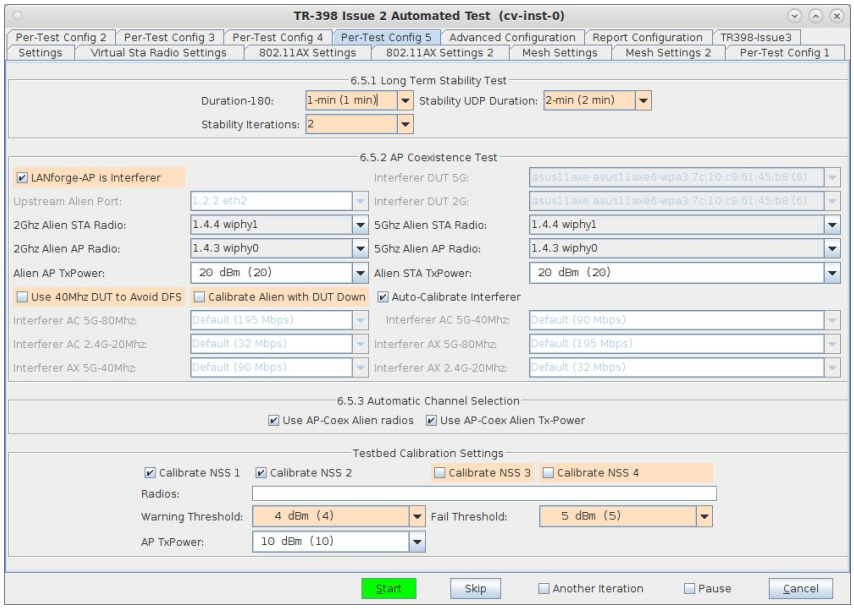
10. The Per-Test Config 3 tab allows setting specific attenuations and pass/fail for the RvR and spatial consistency tests.



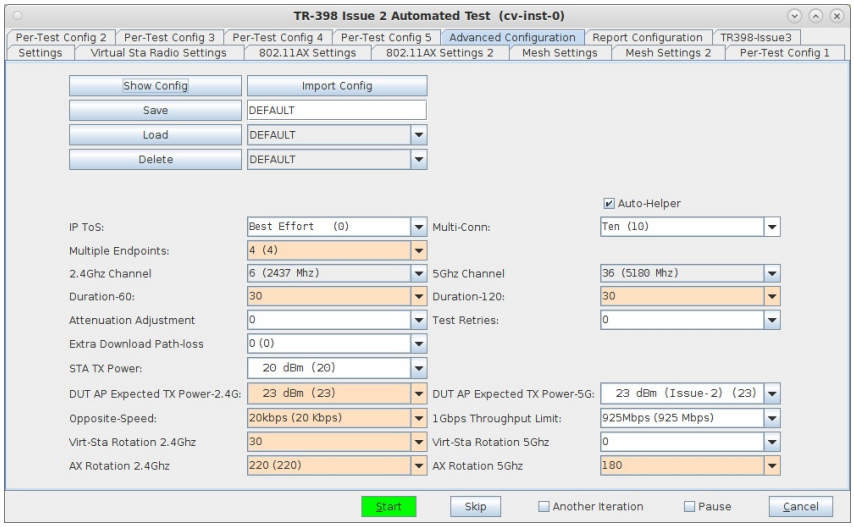
11. The Per-Test Config 4 tab allows tuning the peak-performanc tests and others.



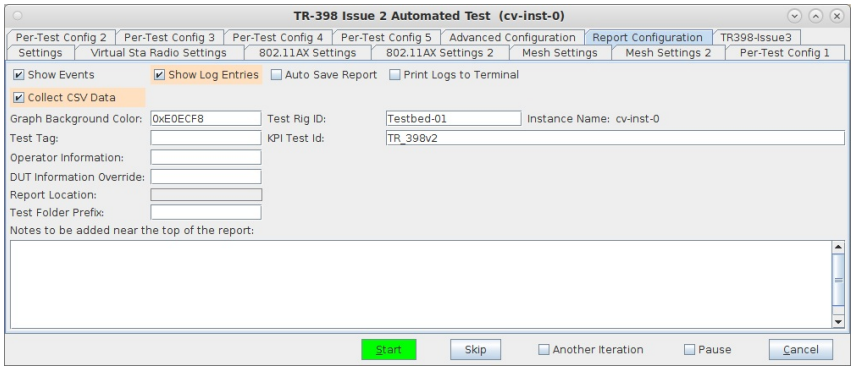
12. The Per-Test Config 5 tab.



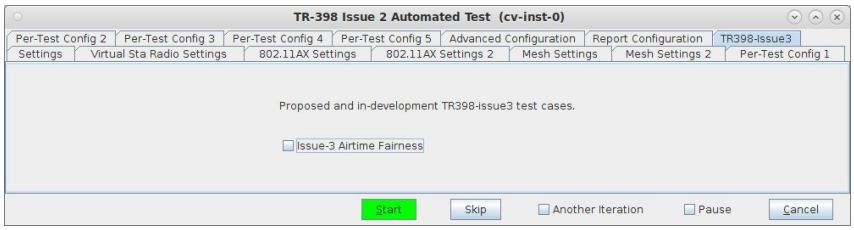
13. The Advanced Configuration tab has settings that apply to many tests. The top section is for saving and loading test configurations. The Import Config window will let one paste in a full or partial test configuration file to apply those changes easily. The Rotation fiels at the bottom allow the user to specify optimal rotations to run tests which do not explicitly control the turntable. Use the Rate vs Orientation test to find optimal rotations.



14. The Report Configuration tab allows customizing the report generated by the TR398 test cases..



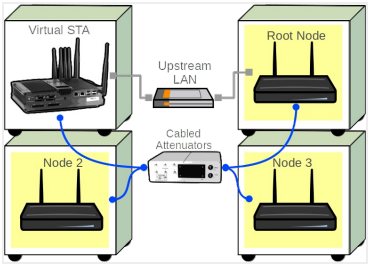
15. The Issue-3 page is for experimental test cases. At time of writing, this tab is not needed. The ATF test case will be executed if the 'Use Issue-3 Behaviour' checkbox is selected on the Settings page.



Testing Mesh APs with automated Mesh scenario

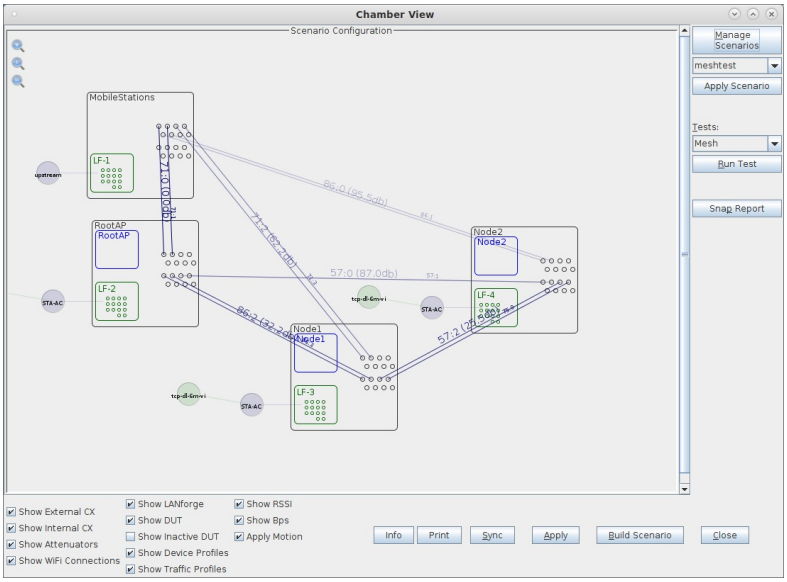
Goal: Setup and run a series of tests in different mesh topologies using the LANforge CT523c or similar system..

In this test scenario, the LANforge CT523c is used to emulate a set of stations that move to different locations relative to a set of mesh APs. Throughput tests are run at each requested topology. A set of 4 chambers are used. One holds the station emulator, the other three hold the mesh APs. Programmable attenuators are used to adjust the signal strenght between APs and the station chamber to emulate different topologies. This example assumes you have some experience with Chamber View, and that you have an appropriate LANforge system, programmable attenuators like the CT714 and four isolation chambers like the CT820a. This feature requires LANforge version 5.4.1 or higher.

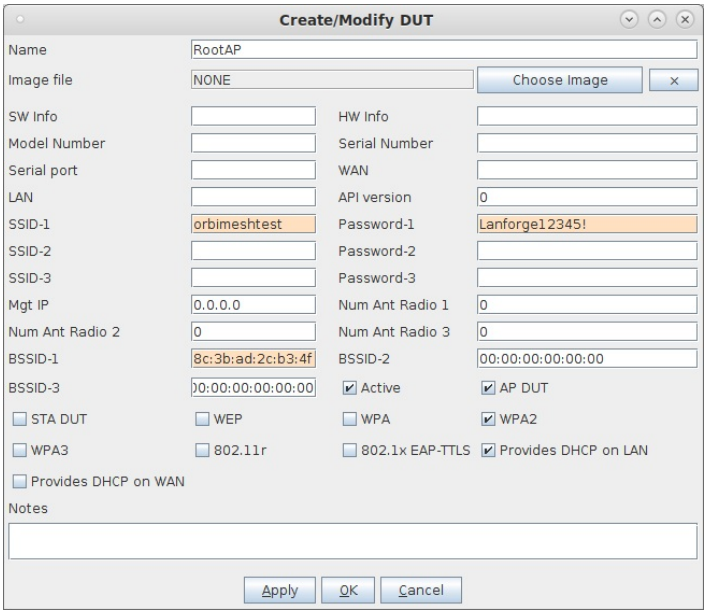


1. Configure Chamber View for Mesh and Similar Tests.

A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. If you have an appropriate scenario already created, then skip to the next section, otherwise you will need to build a scenario that matches your system. You can right-click in Chamber View to create various objects. In this example, the 'Mobilestations' chamber holds the LANforge, and the other chambers hold mesh APs and additional LANforge traffic generating systems. In this example, the LANforge systems in the AP chambers are not used. Your configuration should look like this when these steps are complete.

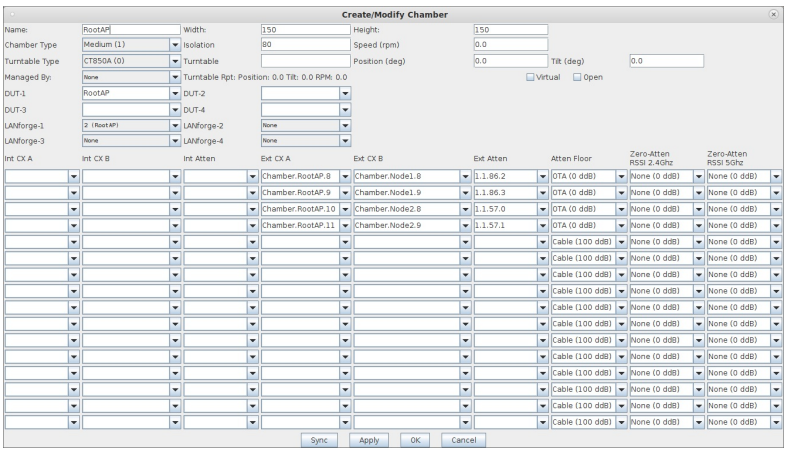


B. Create a Device Under Test (DUT) Profile that matches each of your APs. The BSSID is important to configured so that LANforge knows when it is connected to the correct AP.

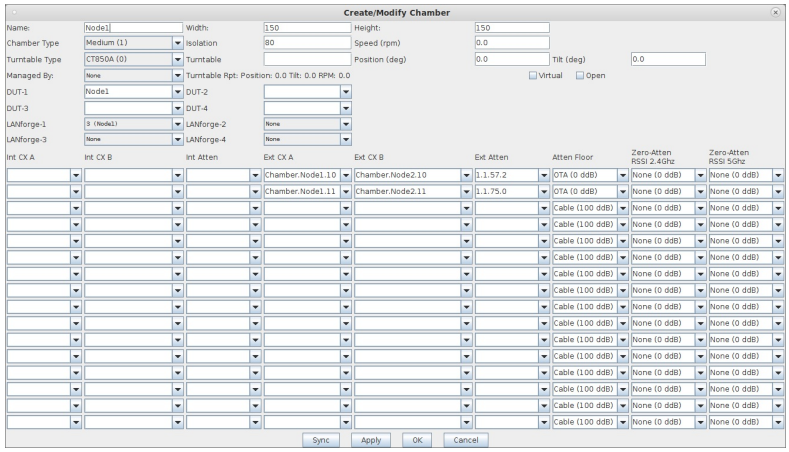


C. Create chamber objects to hold the mobile stations and DUTs, and add the LANforge and DUTs to the chambers. If you have no chambers, you can create a fake chamber, but your test will not be isolated and may not function as desired. Here is an example of the mobile stations chamber. Please note we will auto-calibrate the Zero-Attenuation values for the station chamber.

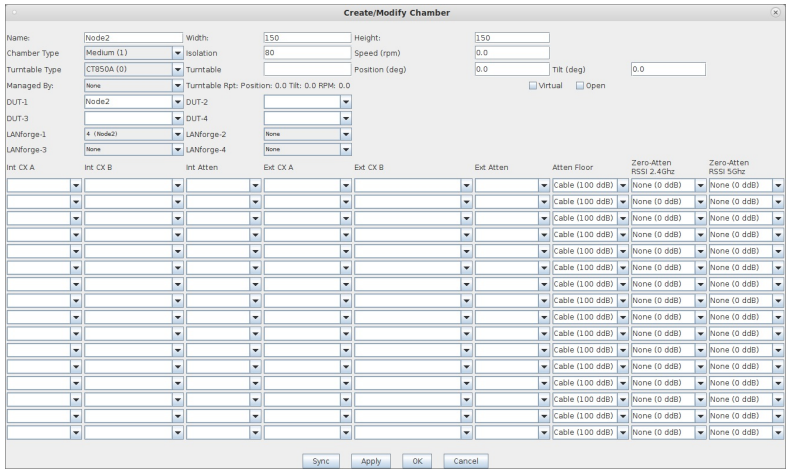
D. Here is an example of the RootAP chamber



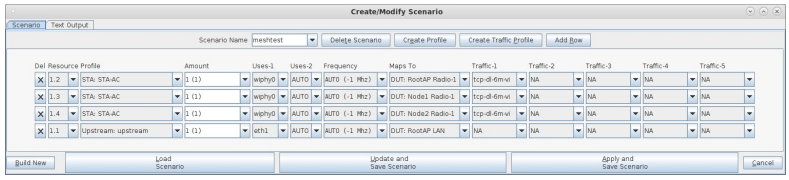
E. Here is an example of the Node-1 chamber



F. Here is an example of the Node-2 chamber

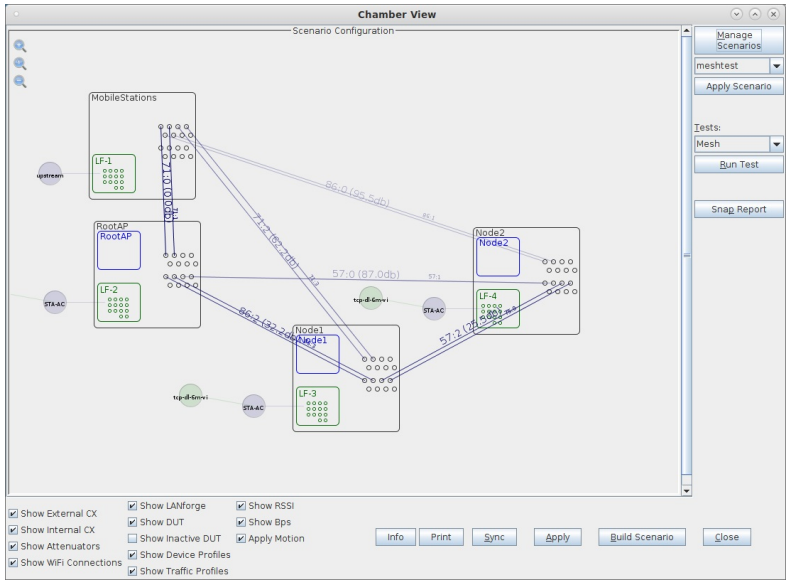


G. Configure a Chamber View Scenario. This is somewhat optional since the mesh automation logic will create it's own temporary scenario to run the automation tests.

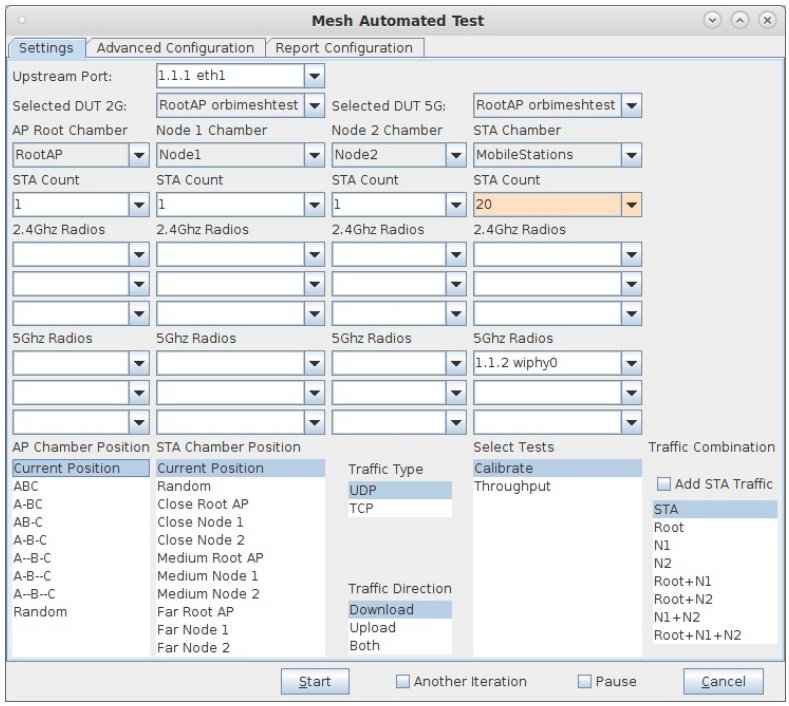


2. Use Chamber View to start the Mesh test.

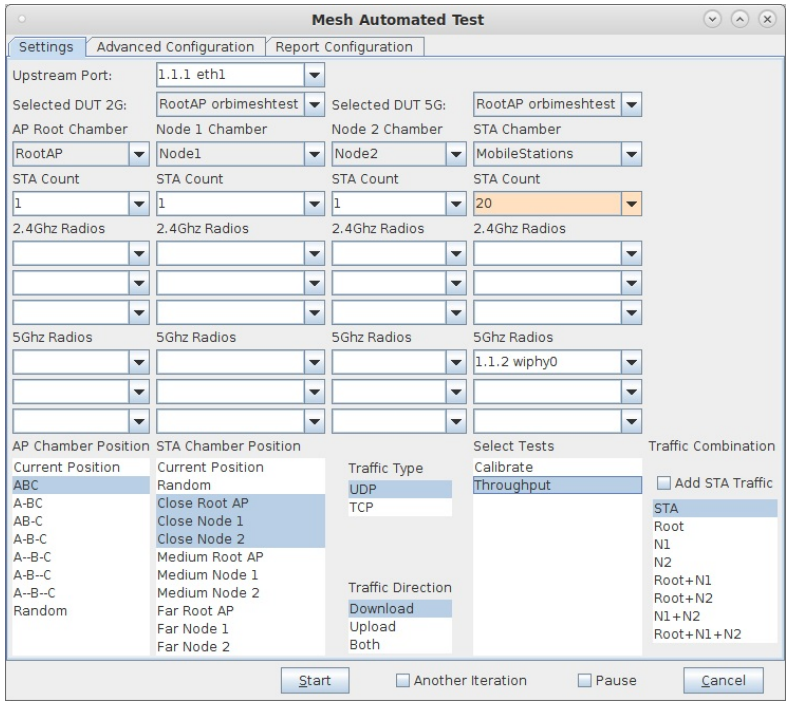
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Load appropriate scenario. Apply the Scenario, then Build the scenario.



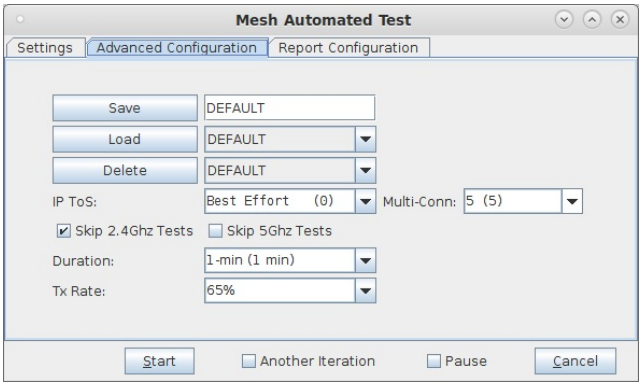
B. Select the **Mesh** test and click **Run Test**. You should see the Mesh Test configuration window pop up. It will remember the last configuration for most fields. Select the resources, topologies and other test config to be used for this test. If you have not run the calibration step previously, run that test first:



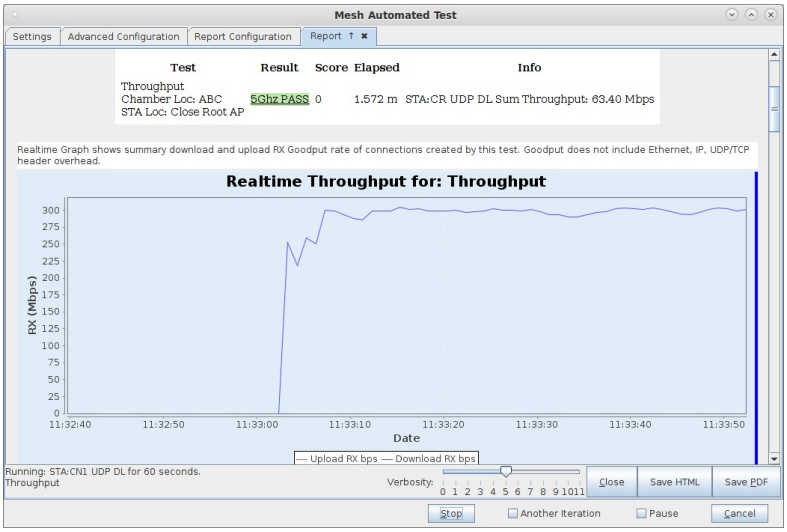
C. Once the calibration step is complete, reconfigure to run the throughput tests. In this case, we are using the ABC topology where all three nodes are close to each other. The stations will do throughput tests at each of three positions: Close to Root AP, Close to Node 1, and Close to Node 2. We will generate UDP download traffic. No stations will be created on the LANforge systems inside the AP chambers in this test.:



D. Configure the appropriate settings on the Advanced tab. In this case, I am disbling 2.4Ghz and doing a 5Ghz only test.



E. When the configuration is complete, click the **Start** button (which will change to 'Stop' once start is clicked) to start the test. An interactive report tab will be created and will be updated as the test runs. For each topology, throughput graphs and topology snapshot will be generated.



F. When the test is complete, click the **Save HTML** button to save an HTML report and generate the PDF. The PDF file will be linked from the HTML page. You can also click 'Save PDF' and the browser will be directed to open the pdf file directly. Please see this example Mesh Test Report.

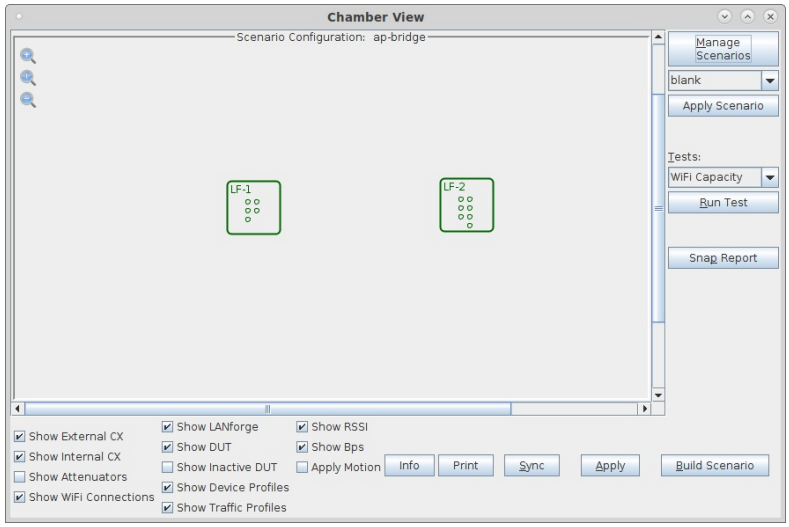
LANforge as Access Point

Goal: Create four LANforge APs in bridged mode using Chamber View

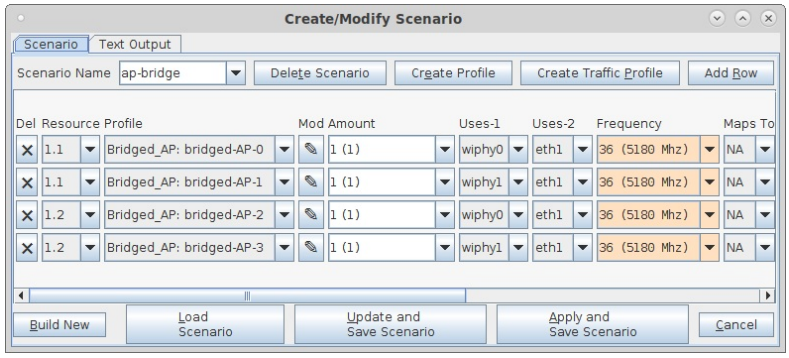
In this test scenario, two LANforge CT522 systems are used to create 4 APs. The APs could be used for testing client devices such as phones. This is a simple example with no authentication.

1. Configure Chamber View to create Access Points.

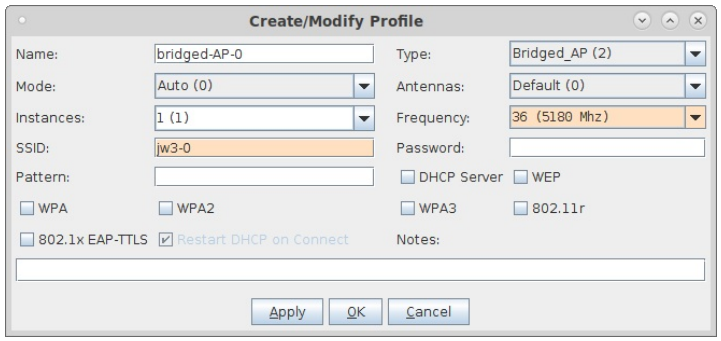
A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. You can right-click in Chamber View to create various objects. The LANforge system(s) should show up as green boxes in Chamber View.



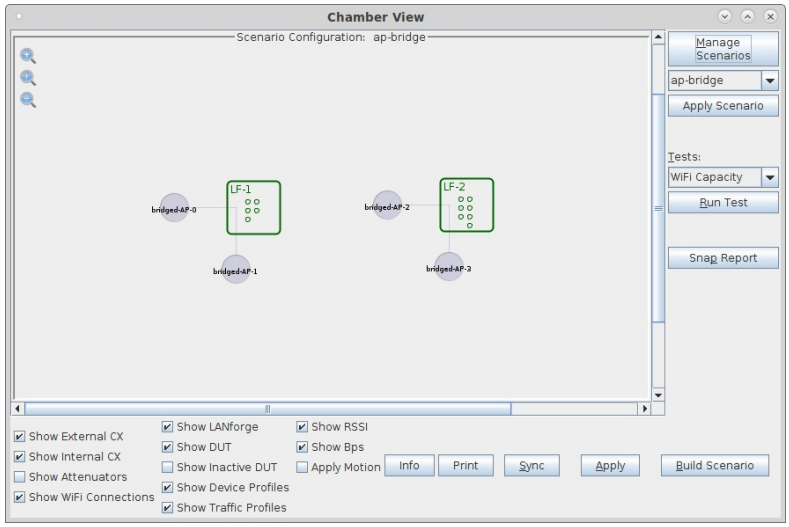
B. Configure a Chamber View Scenario and add the AP profiles.



C. This example uses 4 different bridged-AP profiles. Each profile is the same except that it has a different SSID.



D. Once you have saved and selected the Scenario, click **Apply Scenario** and then click **Build Scenario**. The APs will be created, bridge devices will be created and will contain the APs and the Ethernet ports selected in the scenario. The Access Point devices will be started as part of the build process, so the system is now ready to be used. You can also make further modifications to the AP configuration by modifying the vap interfaces in the Port-Mgr tab of the LANforge GUI.

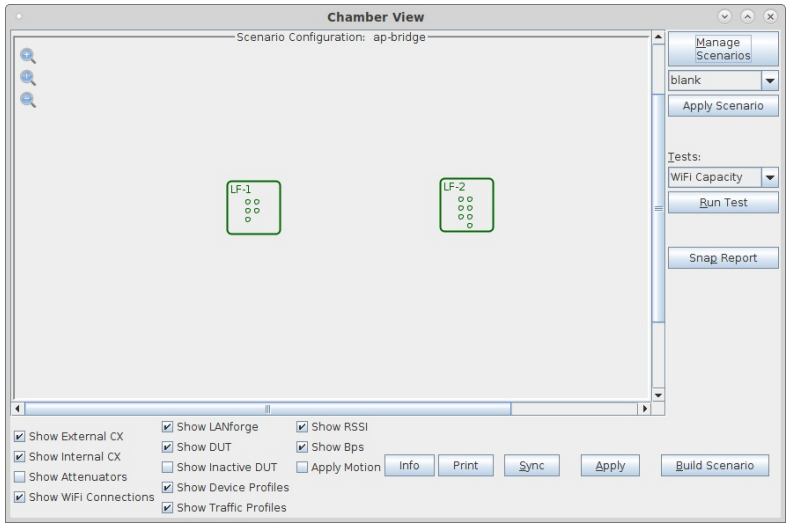


LANforge as 802.11k/v/r Access Point Cluster

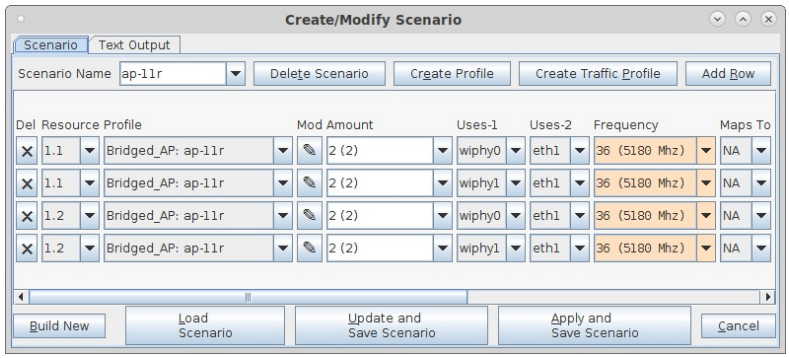
Goal: Create 8 LANforge APs supporting 802.11k, v, and r in bridged mode using Chamber View

In this test scenario, two LANforge CT522 systems are used to create 8 APs. The APs can be used for 802.11k/v/r roaming and related testing. No external radius server is needed. The 'eth1' interfaces on the two LANforges should be connected to the same LAN. NOTE: As of this writing, there is a bug when 802.11w (MFP) is enabled. We are not currently clear whether it is an AP issue or a Station issue.

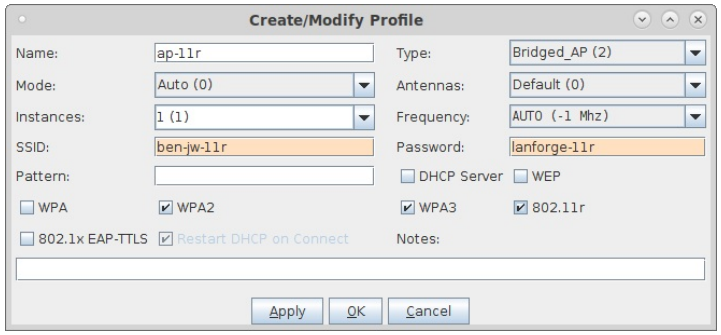
1. Configure Chamber View to create 802.11r Access Points.
   A. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. You can right-click in Chamber View to create various objects. The LANforge system(s) should show up as green boxes in Chamber View.



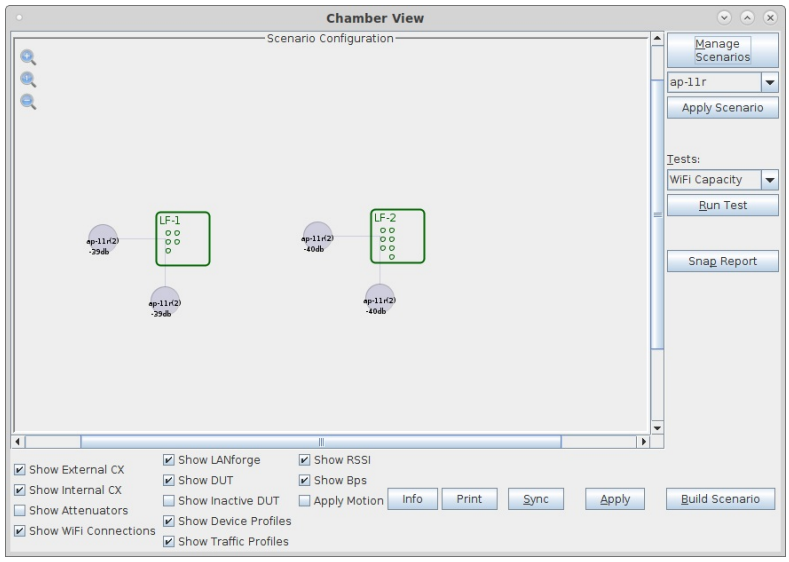
B. Configure a Chamber View Scenario and add the AP profiles.



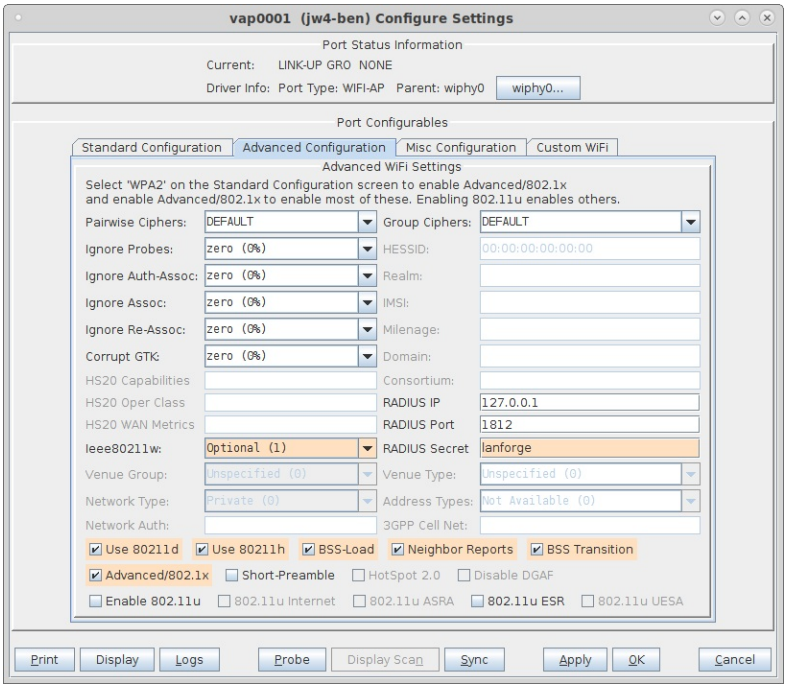
C. This example uses one 802.11r AP profile for all APs.



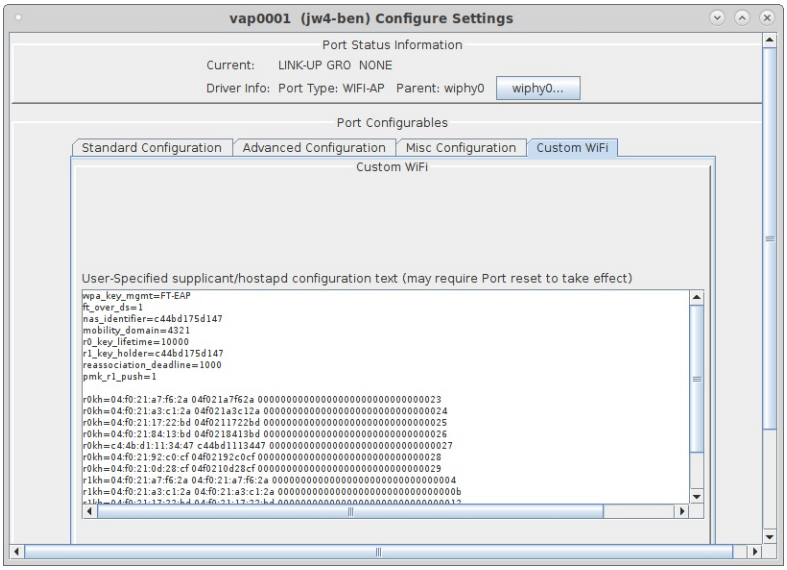
D. Once you have saved and selected the Scenario, click **Apply Scenario** and then click **Build Scenario**. The APs will be created, bridge devices will be created and will contain the APs and the Ethernet ports selected in the scenario. A radius server will be created and started. The Access Point devices will be started as part of the build process, so the system is now ready to be used. You can also make further modifications to the AP configuration by modifying the vap interfaces in the Port-Mgr tab of the LANforge GUI.



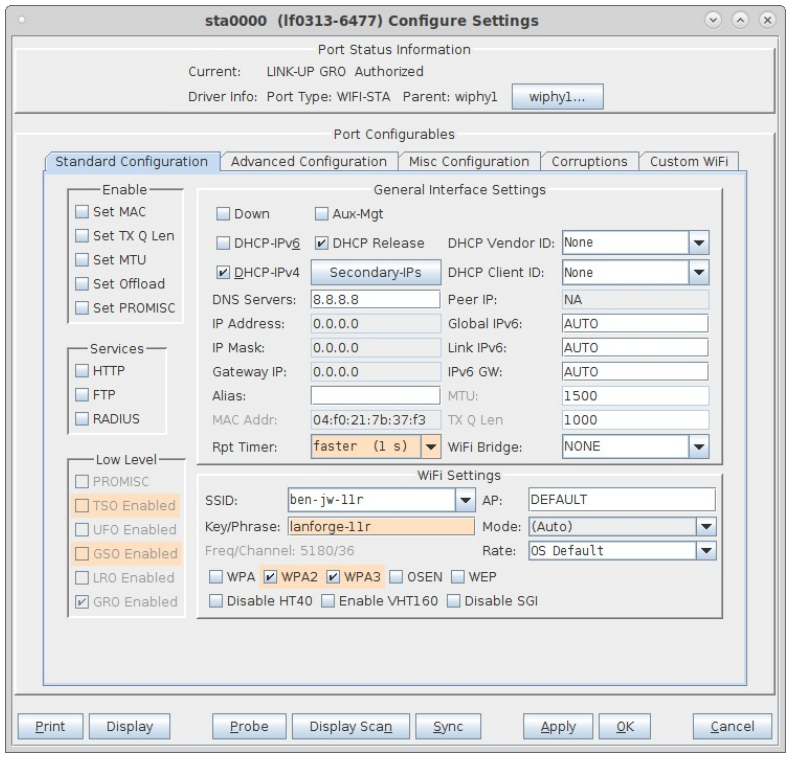
E. To give you some idea of the underlying configuration, please see this VAP configuration window.



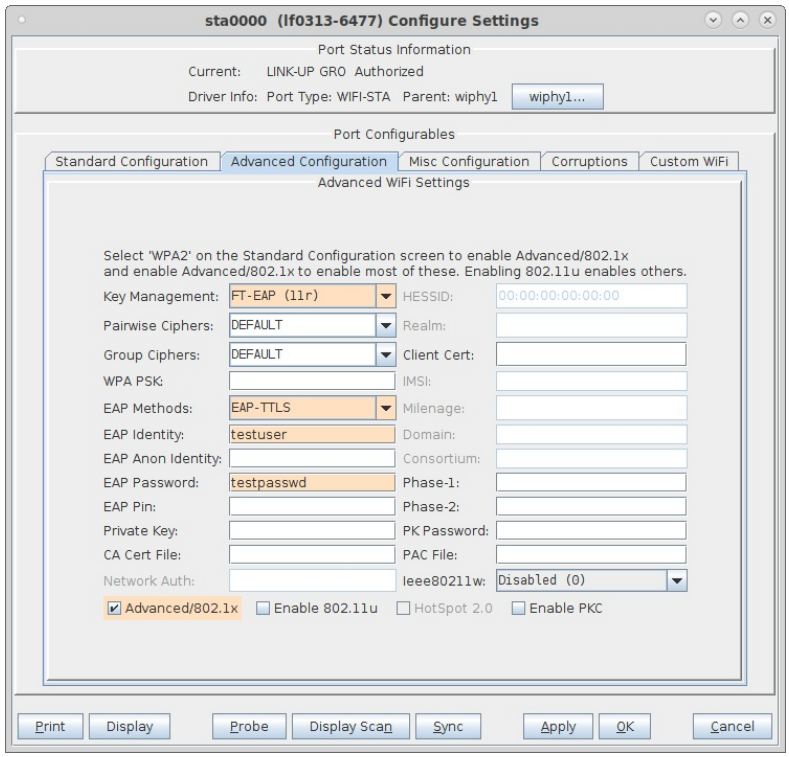
F. And the 'custom' magic that makes the .11r cluster talk to itself.



G. Normally you would configure your own Station device to connect to this AP cluster. In this case, LANforge stations were used. Here is a screenshot of the config window to give some idea of how to configure your own stations.



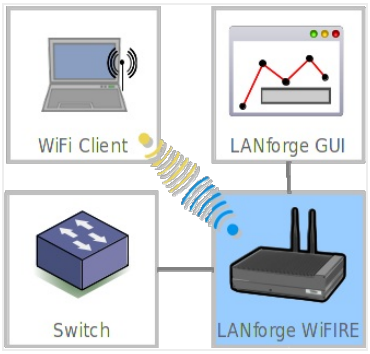
H. The Station advanced screen shows the EAP-TTLS config and key management. Note that 802.11w is disabled in this test to work around some bug.



CI/CD Automated Testing of Access Points

Goal: Setup a framework to automatically find new builds, install them on target AP, run regression tests, and report values. Generate graphs of key data over all runs for historical view. Candela support is available to help customize scripts and fully implement this for your particular environment.

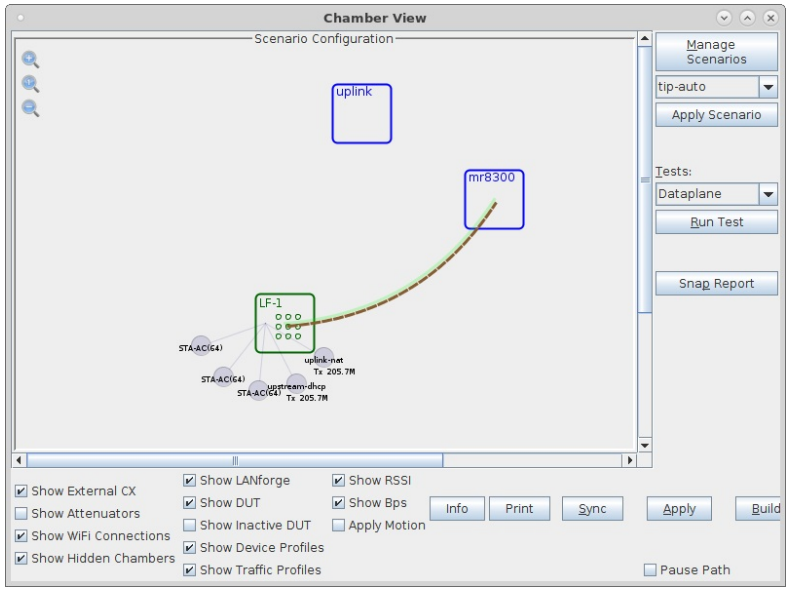
In this test scenario, a LANforge CI/CD package (ct523c, RF chamber, Test Controller and other bits and pieces) consumes AP firmware build images, runs automated tests, and reports the values. A test-orchestrator machine queries the build system to find new builds. Work-items are created when new images are found. A test-bed controller looks for images that match the hardware in the test bed. The test-bed controller updates the AP with new firmware, reboots and configures it. The test-bed controller then automates control of the LANforge system to run a series of tests and then gatherers the results. When the tests are finished, they are uploaded back to the test-orchestrator. The test-orchestrator will regenerate historical graphs when new results are found, and upload the finished reports to a web server.



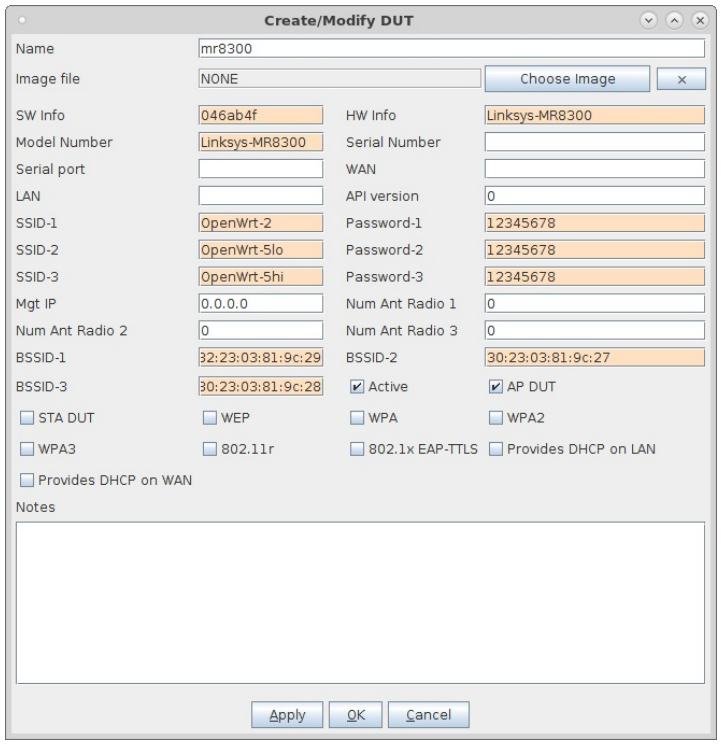
---

1. Configure Systems for automated testing. In this example, there are three main systems: Test-Orchestrator (TO), Test-Bed Controler (TB), and LANforge system (LF). The AP under test is referred to as DUT. Many of the details are left out of this document, your favorite search engine or network administrator should be able to help show you the details.
   A. The automation uses ssh and scp to do its work. So, first you have to make sure that the TB can log into the TO with ssh without requiring password. This example uses the user 'lanforge' on the TO and LF machines.
   B. The testbed machine user must be able to run'sudo' without needing a password.
   C. In order to read console logs, the TO should act as the serial console server for the LANforge and AP DUT. In order to have the lanforge user on the TB system access the serial ports, you must change the permissions:
   `sudo chmod a+rwx /dev/ttyUSB*`
   This command must be run after the serial consoles are connected, or at least re-run once all of the /dev/ttyUSB* devices are found.
   D. Install additional software packages needed by the automation scripts. `sudo pip3 install pexpect-serial`

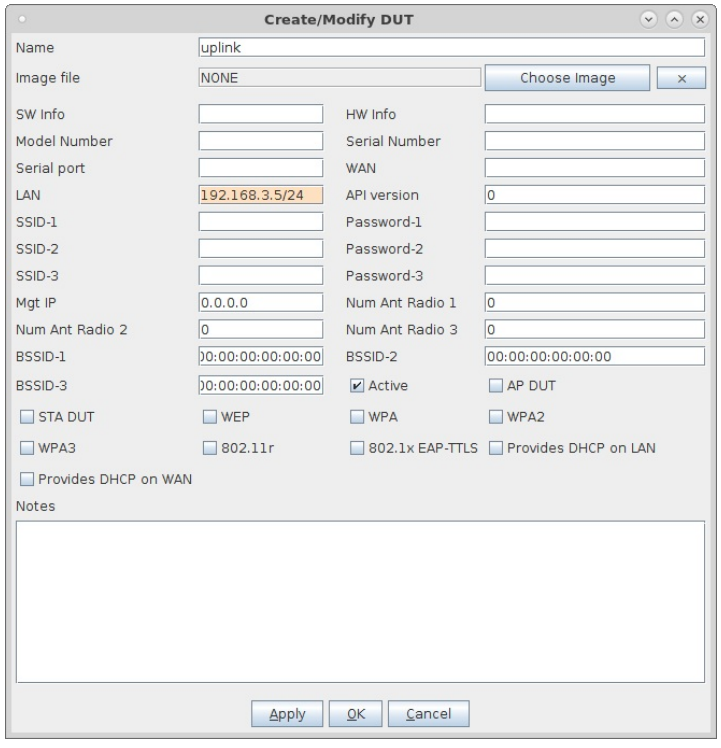
2. Configure LANforge to be used as an automated testbed.
   A. Clean up any old config and create a chamber view scenario and DUT.



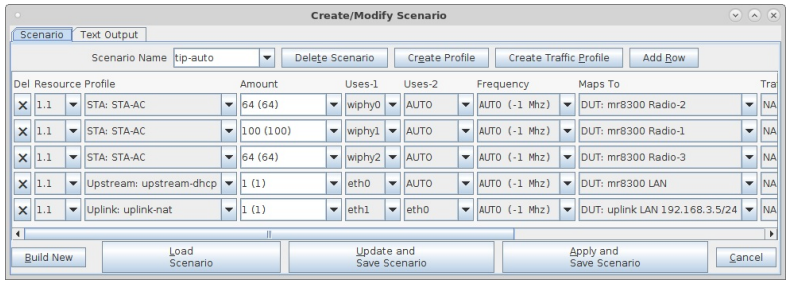
B. In this example, the AP DUT is called 'mr8300'.



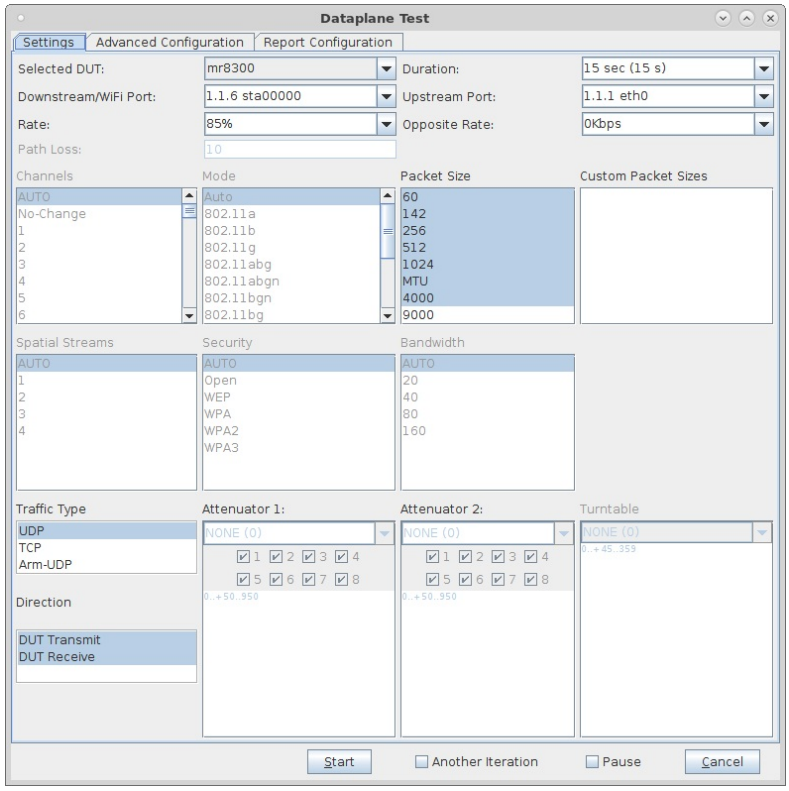
C. This configuration uses some advanced virtual-router features in LANforge to have LANforge serve the DUT DHCP and provide a private NAT'ed network for the DUT the stations connected to it. The uplink DUT is part of how this is configured. The LAN address is the gateway for the virtual router. In this case, the LANforge machine's DUT uplink port is eth1, and it is cabled to a network that uses 192.168.3.5 as the gateway/router. Other test beds may use different Ethernet ports to provide these features.



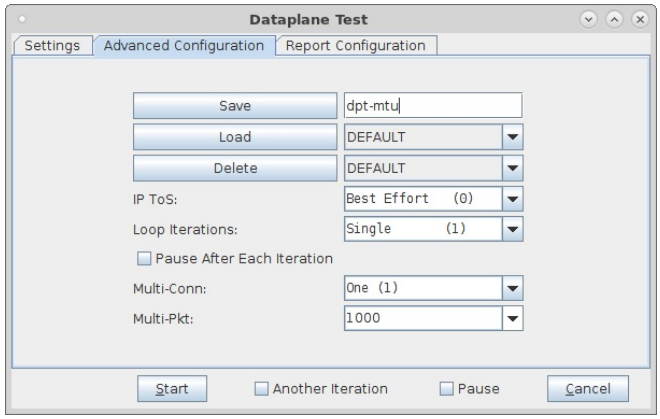
D. The Chamber View scenario creates stations on the available radios, provides the Upstream port (from perspective of the DUT), as well as uplink + NAT for the interface that routes DUT traffic upstream of LANforge.



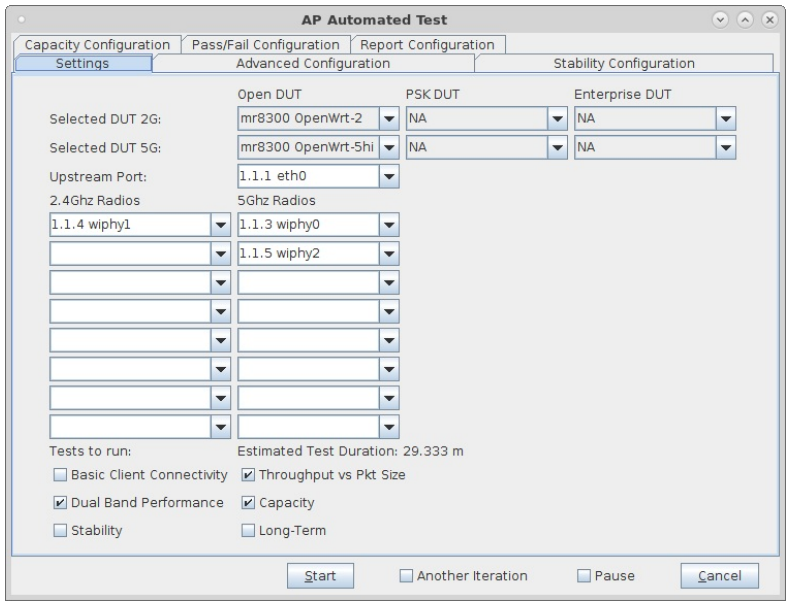
E. Once chamber-view is set up, you need to create some test configurations for the various automated tests you wish to run. In this example, one test we use is the Dataplane test to test throughput at different packet sizes.



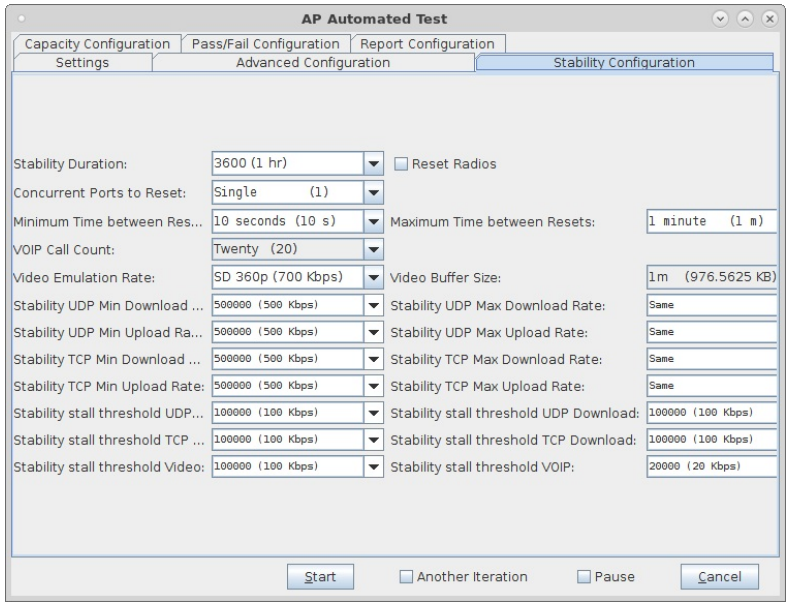
F. On the Dataplane Advanced tab, select a name and save the config. We will use that name later when configuring the automated tests.



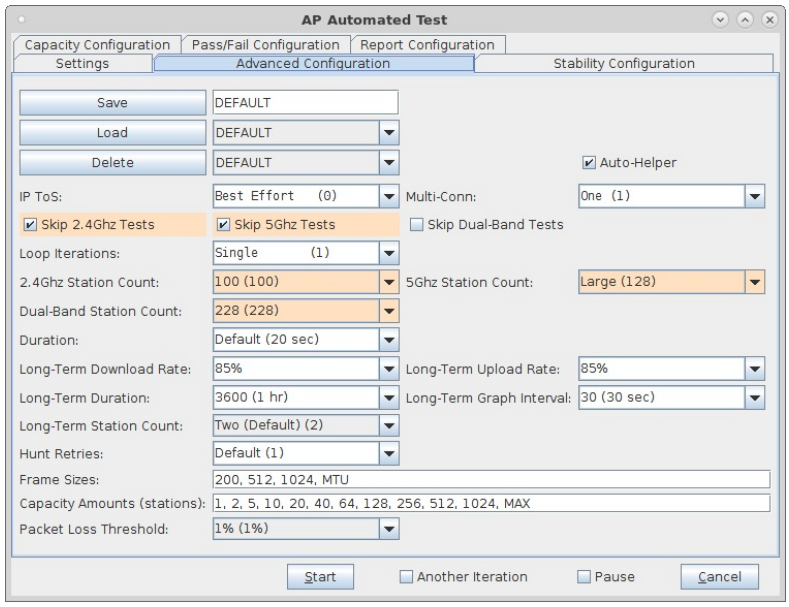
G. Configured the AP-Auto test as well.



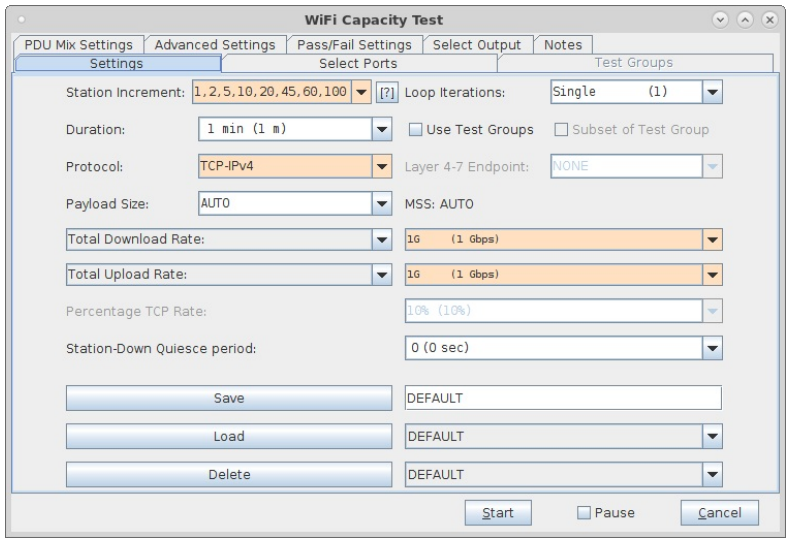
H. Configure the AP-Auto Stability configuration.



I. Configure the AP-Auto Advanced configuration, and save the configuration when done.

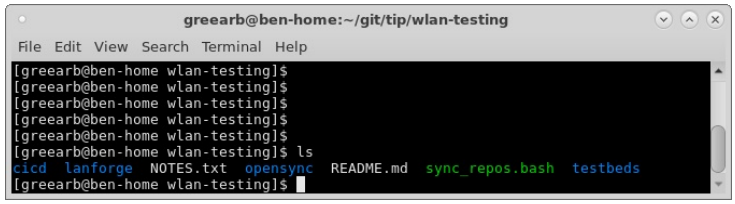


J. Configure a capacity-test, and save the configuration.

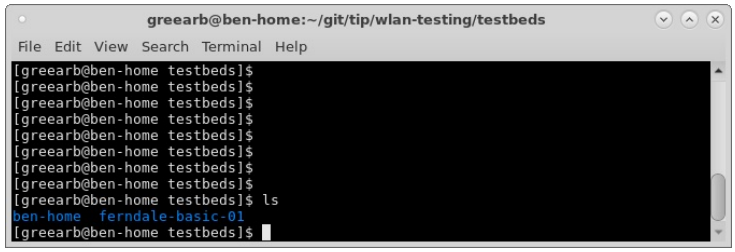


3. Configure Testbed Contoller. This system will use the lanforge-scripts project (installed at /home/lanforge/scripts on LANforge systems). You will probably want to have a separate code repository for the Testbed automation. Candela can help you build scripts specific to your environment. This example shows how it was implemented for one project.

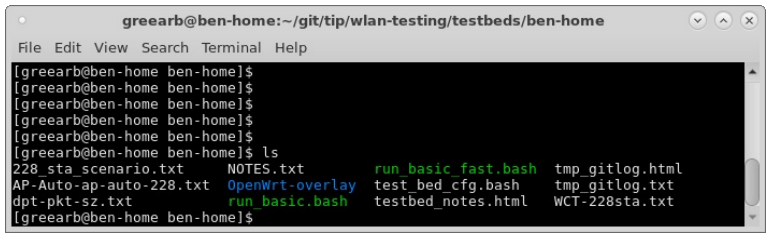
   A. Create a directory structure that looks like this. The 'cicd' directory will contain glue logic to talk to the Test Orchestrator and to the LANforge automation. The testbeds directory holds information for each testbed in your configuration. The lanforge directory has a sub-directory called 'lanforge-scripts' that points to the lanforge-scripts project.



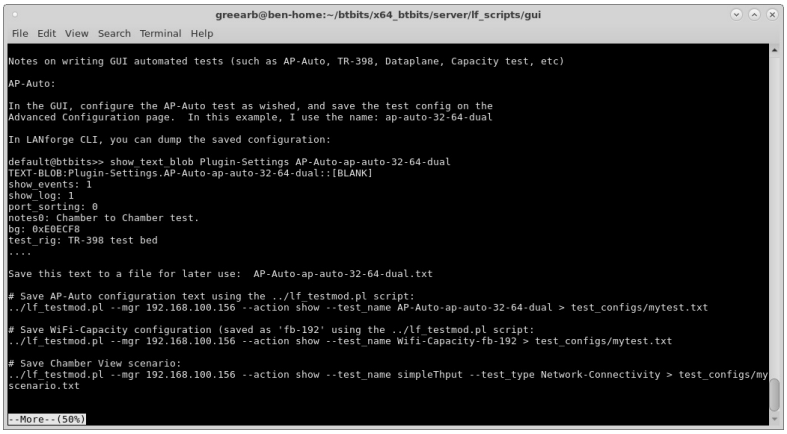
   B. Inside the testbeds directory will be a sub-directory for each testbed in the project. In this example, we have two testbeds. We will focus on the one called 'ben-home'



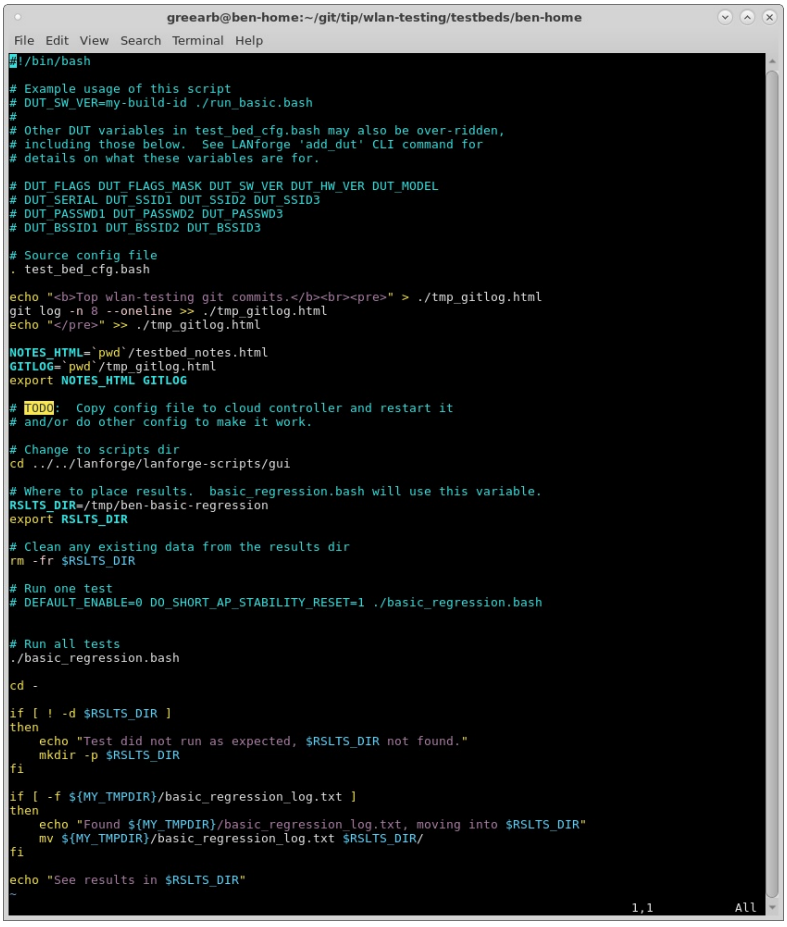
   C. The ben-home testbed directory contains the Chamber View scenario, AP Auto, Dataplane and capacity configuration files. The automation will apply the OpenWrt-overlay on top of the base OpenWrt image after upgrading the software on the AP. This is an optional step and may not be useful for other DUTs.



D. The text configuration files were created by using the lf_testmod.pl script from the lanforge-scripts project. Please see the scripts/gui/README.txt for details.



E. The run_basic.bash script is a helper script that sets some environment variables and then calls the lanforge-scripts/gui/basic_regression.bash script.



F. The run_basic_fast.bash script disables most of the tests and just runs a few specific ones.

greearb@ben-home:~/git/tip/wlan-testing/testbeds/ben-home

File Edit View Search Terminal Help

```
#!/bin/bash

# Example usage of this script
# DUT_SW_VER=my-build-id ./run_basic.bash
#
# Other DUT variables in test_bed_cfg.bash may also be over-ridden,
# including those below.  See LANforge 'add_dut' CLI command for
# details on what these variables are for.

# DUT_FLAGS DUT_FLAGS_MASK DUT_SW_VER DUT_HW_VER DUT_MODEL
# DUT_SERIAL DUT_SSID1 DUT_SSID2 DUT_SSID3
# DUT_PASSWD1 DUT_PASSWD2 DUT_PASSWD3
# DUT_BSSID1 DUT_BSSID2 DUT_BSSID3

# Source config file
. test_bed_cfg.bash

echo "<b>Top wlan-testing git commits.</b><br><pre>" > ./tmp_gitlog.html
git log -n 8 --oneline >> ./tmp_gitlog.html
echo "</pre>" >> ./tmp_gitlog.html

NOTES_HTML=`pwd`/testbed_notes.html
GITLOG=`pwd`/tmp_gitlog.html
export NOTES_HTML GITLOG

# TODO:  Copy config file to cloud controller and restart it
# and/or do other config to make it work.

# Change to scripts dir
cd ../../lanforge/lanforge-scripts/gui

# Where to place results.  basic_regression.bash will use this variable.
RSLTS_DIR=/tmp/ben-basic-regression-fast
export RSLTS_DIR

# Clean any existing data from the results dir
rm -fr $RSLTS_DIR

# Run a subset of available tests
# See 'Tests to run' comment in basic_regression.bash for available options.

#DEFAULT_ENABLE=0 WCT_DURATION=20s DO_SHORT_AP_BASIC_CX=1 DO_WCT_BI=1 ./basic_regression.bash

DEFAULT_ENABLE=0 WCT_DURATION=20s DO_SHORT_AP_BASIC_CX=1 DO_WCT_BI=0 ./basic_regression.bash


# Run all tests
#./basic_regression.bash

cd -

if [ ! -d $RSLTS_DIR ]
then
    echo "Test did not run as expected, $RSLTS_DIR not found."
    mkdir -p $RSLTS_DIR
fi

if [ -f ${MY_TMPDIR}/basic_regression_log.txt ]
then
    echo "Found ${MY_TMPDIR}/basic_regression_log.txt, moving into $RSLTS_DIR"
    mv ${MY_TMPDIR}/basic_regression_log.txt $RSLTS_DIR/
fi

echo "See results in $RSLTS_DIR"
```

42,1 All

G. Both of the scripts use the test_bed_cfg.bash file to configure the details about the device-under-test and the LANforge system. This configuration will allow the automation to properly do its work.

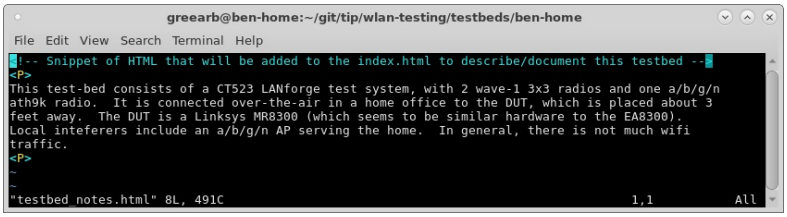
greearb@ben-home:~/git/tip/wlan-testing/testbeds/ben-home

File Edit View Search Terminal Help

```
# Example test-bed configuration

# Scripts should source this file to set the default environment variables
# and then override the variables specific to their test case (and it can be done
# in opposite order for same results
#
# After the env variables are set,
# call the 'lanforge/lanforge-scripts/gui/basic_regression.bash'
# from the directory in which it resides.

PWD=`pwd`
AP_SERIAL=${AP_SERIAL:-/dev/ttyUSB0}
LF_SERIAL=${LF_SERIAL:-/dev/ttyS1}
LFPASSWD=${LFPASSWD:-r}  # Root password on LANforge machine
AP_AUTO_CFG_FILE=${AP_AUTO_CFG_FILE:-$PWD/AP-Auto-ap-auto-228.txt}
WCT_CFG_FILE=${WCT_CFG_FILE:-$PWD/WCT-228sta.txt}
DPT_CFG_FILE=${DPT_CFG_FILE:-$PWD/dpt-pkt-sz.txt}
SCENARIO_CFG_FILE=${SCENARIO_CFG_FILE:-$PWD/228_sta_scenario.txt}

# LANforge target machine
LFMANAGER=${LFMANAGER:-192.168.3.190}

# LANforge GUI machine (may often be same as target)
GMANAGER=${GMANAGER:-192.168.3.190}
GMPORT=${GMPORT:-3990}
MY_TMPDIR=${MY_TMPDIR:-/tmp}

# Test configuration (10 minutes by default, in interest of time)
STABILITY_DURATION=${STABILITY_DURATION:-600}
TEST_RIG_ID=${TEST_RIG_ID:-Ben-Home-OTA}

# DUT configuration
#DUT_FLAGS=${DUT_FLAGS:-0x22}  # AP, WPA-PSK
DUT_FLAGS=${DUT_FLAGS:-0x2}  # AP, Open
DUT_FLAGS_MASK=${DUT_FLAGS_MASK:-0xFFFF}
DUT_SW_VER=${DUT_SW_VER:-OpenWrt-Stock}
DUT_HW_VER=Linksys-MR8300
DUT_MODEL=Linksys-MR8300
DUT_SERIAL=${DUT_SERIAL:-NA}
DUT_SSID1=${DUT_SSID1:-OpenWrt-2}
DUT_SSID2=${DUT_SSID2:-OpenWrt-5lo}
DUT_SSID3=${DUT_SSID3:-OpenWrt-5hi}
DUT_PASSWD1=${DUT_PASSWD1:-12345678}
DUT_PASSWD2=${DUT_PASSWD2:-12345678}
DUT_PASSWD3=${DUT_PASSWD3:-12345678}
DUT_BSSID1=32:23:03:81:9c:29
DUT_BSSID2=30:23:03:81:9c:27
DUT_BSSID3=30:23:03:81:9c:28

export LF_SERIAL AP_SERIAL LFPASSWD
export AP_AUTO_CFG_FILE WCT_CFG_FILE DPT_CFG_FILE SCENARIO_CFG_FILE
export LFMANAGER GMANAGER GMPORT MY_TMPDIR
export STABILITY_DURATION TEST_RIG_ID
export DUT_FLAGS DUT_FLAGS_MASK DUT_SW_VER DUT_HW_VER DUT_MODEL
export DUT_SERIAL DUT_SSID1 DUT_SSID2 DUT_SSID3
export DUT_PASSWD1 DUT_PASSWD2 DUT_PASSWD3
export DUT_BSSID1 DUT_BSSID2 DUT_BSSID3

~
```

1,1 All

H. The testbed_notes.html is a snippet of HTML that will be added to reports to describe this test bed.

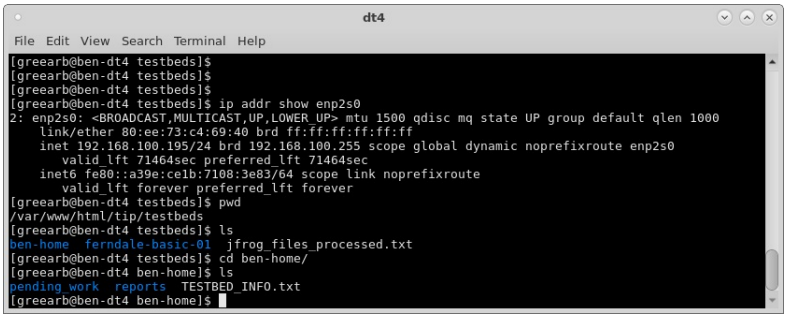


I. The cicd directory contains a directory for each of the test-beds, named similar to the testbeds directory. In this case, we are using the ben-home testbed. This directory will hold a file called TESTBED_INFO.txt. This describes the DUT hardware platform so that the test orchestrator knows what testbed can handle which binary images (this project is doing builds for multiple hardware platforms.)

J. The cicd directory also contains a script that queries the test orchestrator. The code in it is beyond the scope of this guide, but roughly speaking, it queries the test orchestrator's web page to find new work items, and when found, it will execute the test and upload results back to the orchestrator. An example run of this tool is:
`../testbed_poll.pl --jfrog_passwd secret --url http://192.168.100.195/tip/testbeds/ben-home/pending_work/`
For full automation, this program should run in a loop, waiting 2 minutes between tries to give the orchestrator time to notice the results and remove the old work-item.

4. Configure test orchestrator. This system is visible to all the test beds. It could be a VM running in a cloud, or some other system running in your lab. It will have a web server, and directories dedicated to the automation.

A. Create a directory structure that looks like this. A copy of the cicd/[testbeds] should be placed here. In particular, the TESTBED_INFO.txt is important.



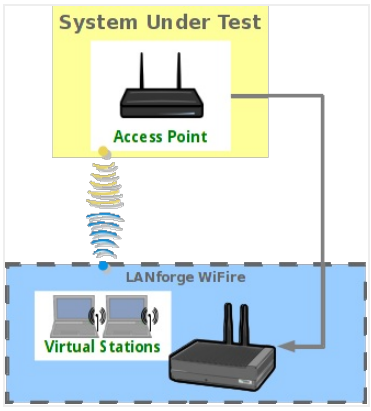
B. The orchestrator will poll the build artifact system to find new builds and build work-items. It will place the work-items in the test-beds that it wishes to execute the tests.

C. The testbed controller will find this work item, do the work, and then upload the results to the location specified in the work-item. The test-orchestrator will then generate summary reports for all available results, including comparison graphs for the different test runs. It will upload these results to some web page so that users can view the end results.

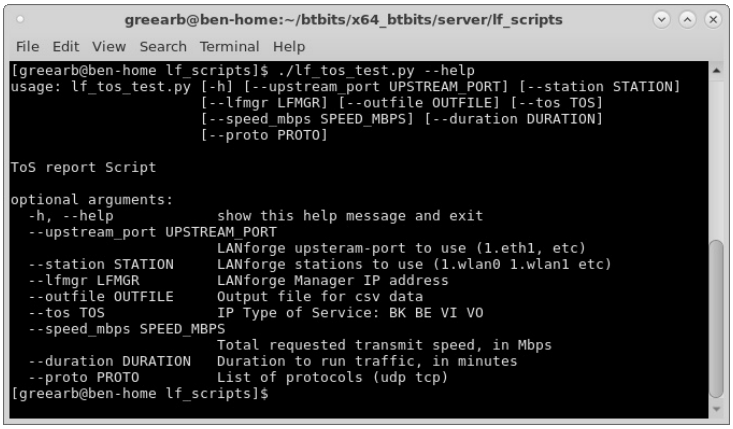
Using LANforge python script to test QoS on WiFi Networks

Goal: Create data connections with different QoS settings and report on throughput and latency

This script automates creating data connections with BK, BE, VI, VO and/or other QoS settings. It then runs the tests for a specified amount of time and reports on latency and throughput. The report is an xlsx spreadsheet that can be opened in your favorite spread-sheet tool. This script requires LANforge 5.4.2 or higher.

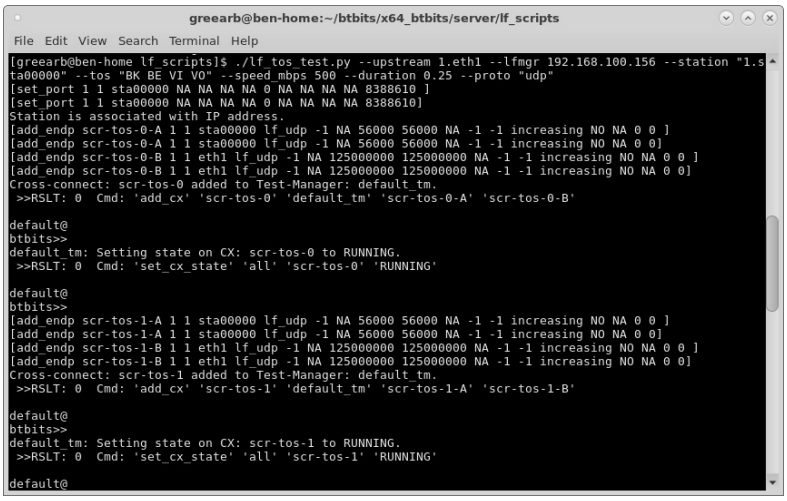


1. Create and configure stations to talk to your DUT. You may create these through Chamber View scenarios or other methods. Please refer to introductory cookbooks for more information on this if you have questions.
2. Run lf_tos_test.py script.
    A. Change directory to the /home/lanforge/scripts directory (or other location if you have installed scripts elsewhere), and run the lf_tos_tst.py script with the --help argument to understand your options..

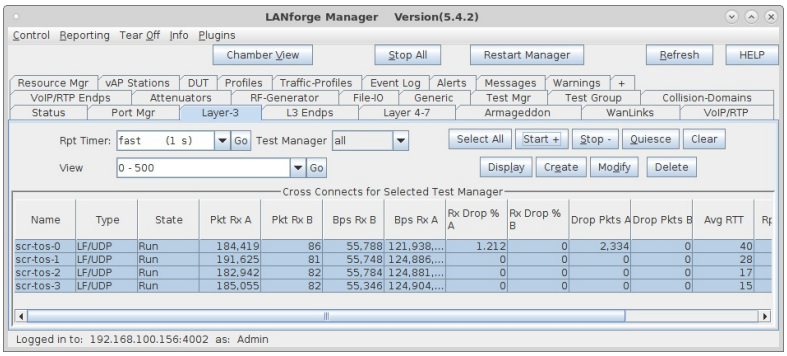


B. Run the script with arguments for your test case. The basic idea is to provide a list of stations you wish to use for traffic generation, the upstream port, the type of service settings to use, requested bandwidth, etc.
Examples
Run BE, BK, VI VO traffic between eth1 and sta0000, 500Mbps total traffic requested, with 15 second duration:

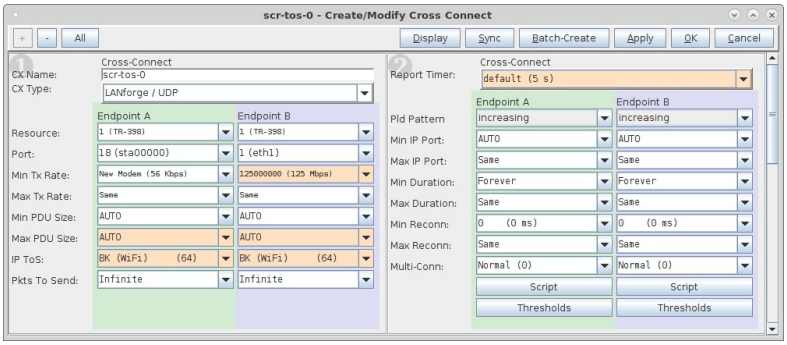
```
./lf_tos_test.py --upstream 1.eth1 --lfmgr 192.168.100.156 --station "1.sta00000" \
 --tos "BK BE VI VO" --speed_mbps 500 --duration 0.25 --proto "udp"
```



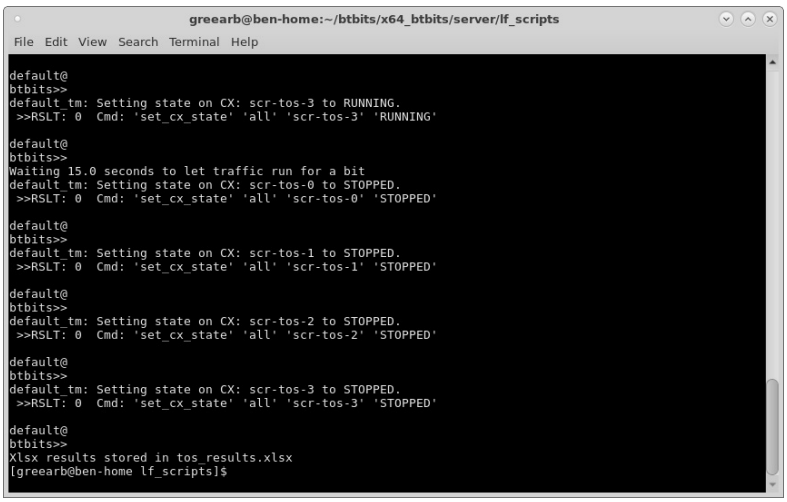
C. The auto-created data connections can be found in the Layer-3 tab in the LANforge GUI.



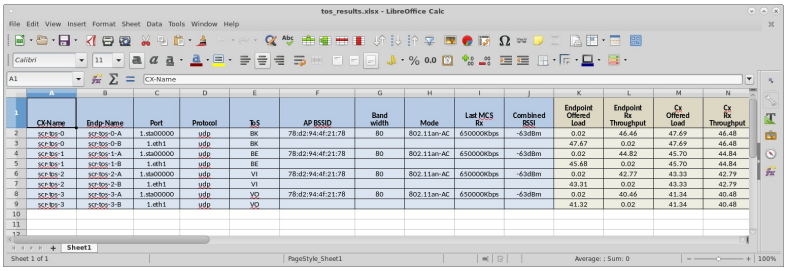
D. Here are the details of one of the connections.



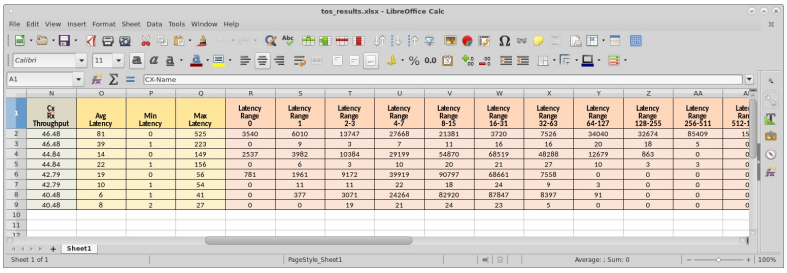
E. The script will run for the requested duration and then stop the traffic and generate the xlsx report.



F. You can open the xlsx report in your favorite spread sheet tool. Here is the first part of the document.



G. The more interesting part of the report is the throughput and latency results. The latency-range columns show the amount of packets received in certain latency ranges. This gives you an idea of jitter and other latency related information. The average latency is a good way to get a quick idea of the latency.

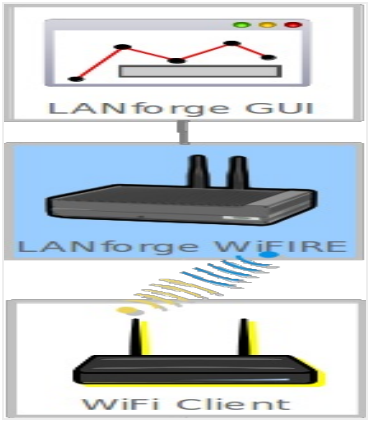


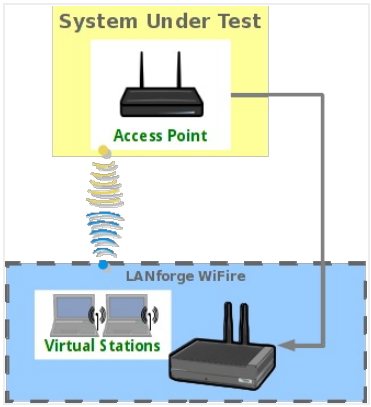
H. For more details, see the spread-sheet result from this test.
There is also a more advanced script that does similar work, see: ./lf_tos_plus_test.py --help, example output.

Using LANforge python script to generate a mix of traffic and capture it.

Goal: Create WiFi stations with specified a/b/g/n/AC/AX modes and spatial streams, data connections with different QoS, packet size and offered load settings and report on throughput and latency. Captures will be made of the offered traffic and also the RF packet capture for processing by external tools.

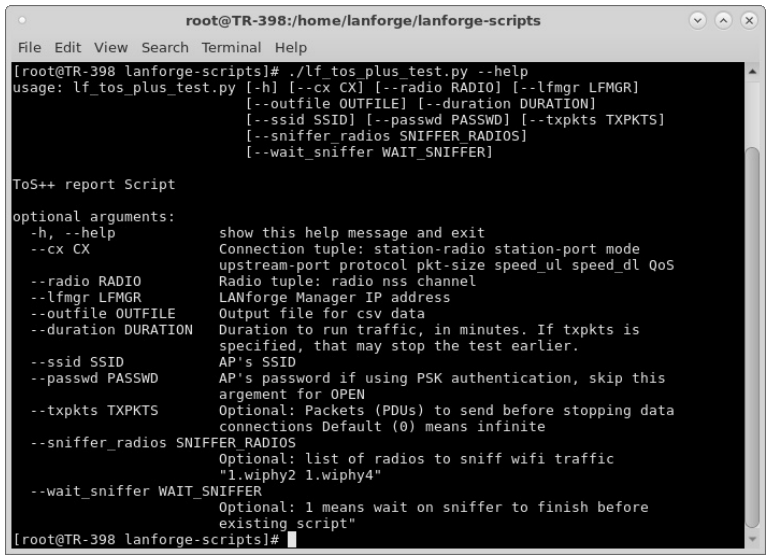
This script automates creating data connections with BK, BE, VI, VO and/or other QoS settings, as well as packet sizes, protocol type, and offered load. It then starts packet captures on the Ethernet port and RF monitor interfaces. The traffic and packet captures run for a specified amount of time and XLSX and CSV reports are generated to report latency and throughput. The packet captures are placed into a directory for post-processing by user-supplied tools. This script requires LANforge 5.4.2 or higher.





1. This test does not require much initial configuration. In this example, I have removed all existing WiFi station and data connection configuration before running the script.

2. Run lf_tos_plus_test.py script.
    A. Change directory to the /home/lanforge/scripts directory (or other location if you have installed scripts elsewhere), and run the lf_tos_plus_tst.py script with the --help argument to understand your options..



B. Run the script with arguments for your test case. The basic idea is to provide a list of connection definitions you wish to use for traffic, along with some additional test-wide configuration.

--cx

Use multiple '--cx' arguments to create as many data connections as you wish.
Each cx argument takes a list of configuration input:
`station-radio station-port mode upstream-port protocol pkt-size speed_ul speed_dl QoS`
For instance:
`--cx "1.wiphy0 1.wlan0 an 1.eth1 udp 1024 10000 500000000 BK"`
The example above creates a wlan0 interface in /AN mode on radio wiphy0. The eth1 port is the upstream side of the data connection. The protocol type is UDP, with 1024 byte payloads. Upload speed is 10kbps, download speed is 500Mbps, and ToS/QoS is Background.

--ssid

The --ssid specifies the SSID for the AP under test.

--passwd

The --passwd specifies the DUT password (PSK). If this option is not specified, then OPEN authentication will be used.

--radio

The '--radio' option allows one to configure the number of spatial streams and channel for a radio used in this test. In this example, we are forcing the radio to use 2 spatial streams and using channel 0 (which means use ANY available channel).
Example: `--radio "1.wiphy0 2 0"`

--dur

The --dur option determines the duration in minutes for the traffic connections and packet capture.

--txpkts

The --txpkts option configures data connections to generate a certain amount of PDUs and then quiesce the test. The test will complete when all connections are finished or the duration timer has expired, whichever is less.

--sniffer_radios

The --sniffer_radios allows you to select a list of radios to be used as packet capture devices. The first radio selected will capture on the first WiFi interface configured in the --cx section, the second for the second --cx, etc. The script automatically takes care of configuring the sniffer for the proper channel, and in the case of OFDMA radios, it will be configured for the specific AID so that it can capture OFDMA traffic.

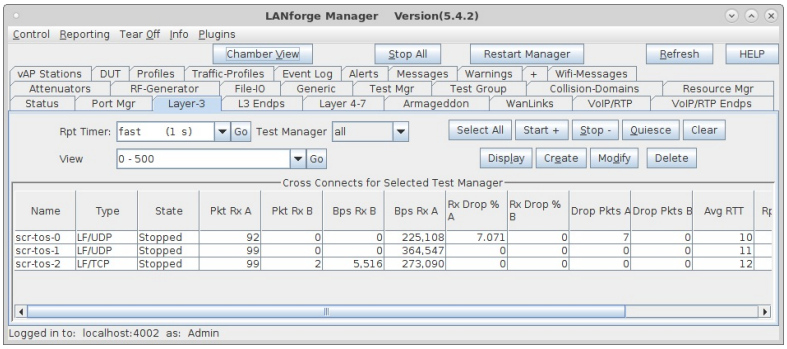
--wait_sniffer

The --wait_sniffer option causes the script to pause until the sniffer programs have completed before existing the script. This makes automation easier since you then know the captures are complete when the script exits.

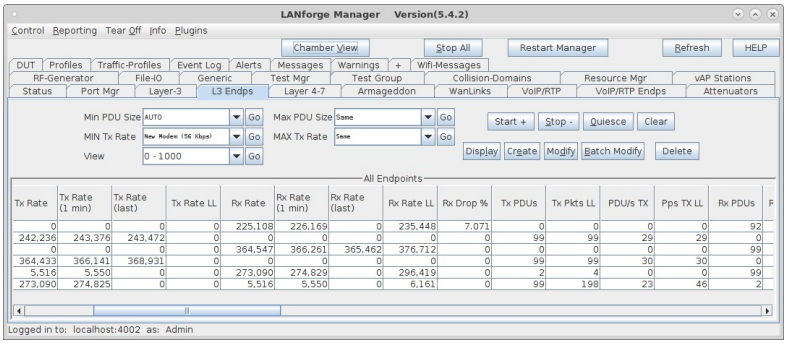
The command used in this example is:

```
./lf_tos_plus_test.py --dur 0.5 --lfmgr 192.168.100.156 --ssid NETGEAR68-5G --
passwd aquaticbug712 --radio "1.wiphy0 2 0" --txpkts 99
--cx "1.wiphy0 1.wlan0 an 1.eth1 udp 1024 10000 500000000 BK"
--cx "1.wiphy0 1.wlan0 an 1.eth1 udp MTU 10000 500000000 VI"
--cx "1.wiphy0 1.sta0 anAC 1.eth1 tcp 1472 56000 2000000 BK"
--sniffer_radios "1.wiphy2" --wait_sniffer 1
```

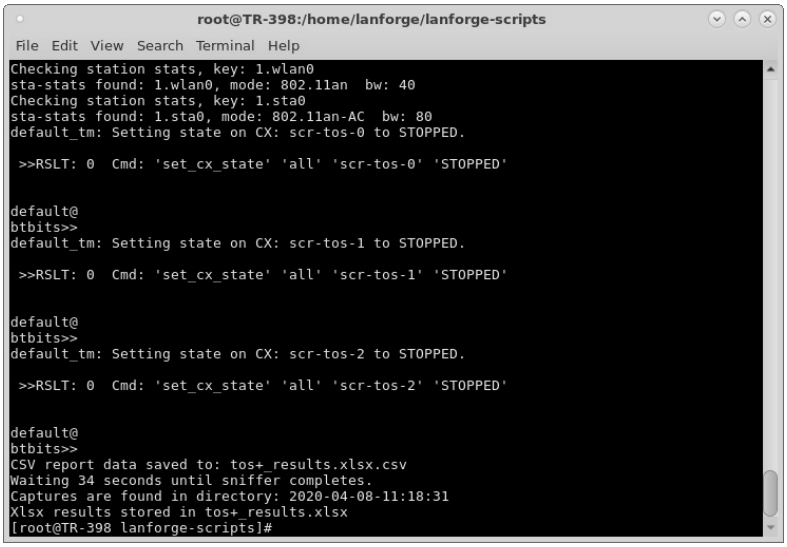
C. The auto-created data connections can be found in the Layer-3 tab in the LANforge GUI.



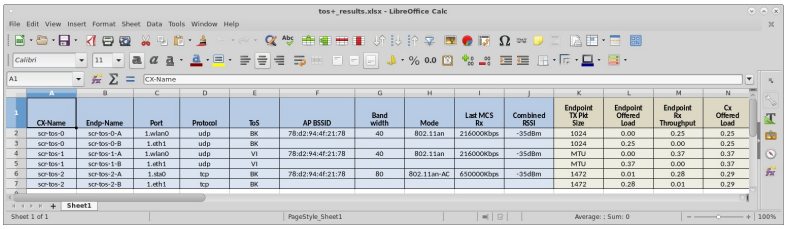
D. Here are the details of the endpoints. Notice each sent 99 pdus as requested with the --txpkts option.



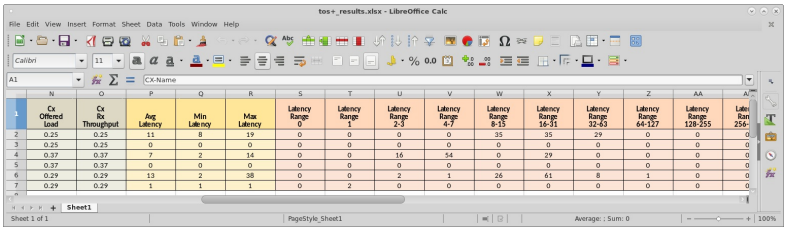
E. The script will run for the requested duration and then stop the traffic and generate the xlsx report.



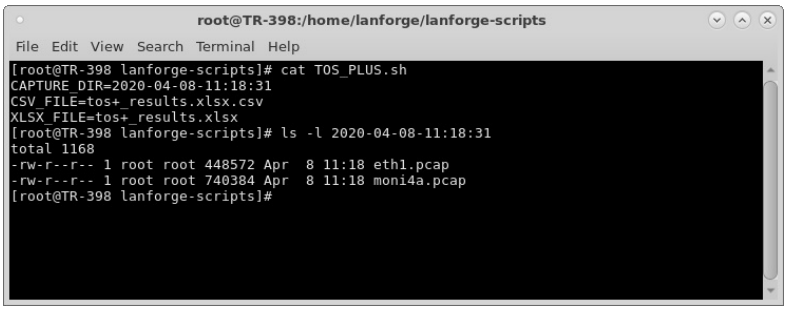
F. You can open the xlsx report in your favorite spread sheet tool. Here is the first part of the document.



G. The more interesting part of the report may be the throughput and latency results. The latency-range columns show the amount of packets received in certain latency ranges. This gives you an idea of jitter and other latency related information. The average latency is a good way to get a quick idea of the latency. In this case, we are only generateing 99 packets, so the report is of limited use. When sending more data, this data becomes more useful.



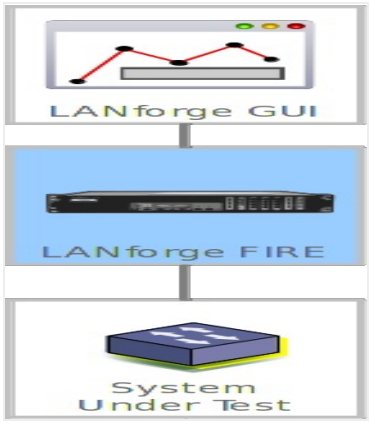
H. This script is designed to work well with automation. It creates a file called TOS_PLUS.sh that holds some shell variables defining the name of the XLSX and CSV files, as well as the capture directory. The capture files may be opened with wireshark or your other favorite packet analyzer, as well as post-processed by other automated tools. For instance, you can find a LANforge packet in the eth1 capture and the same packet in the RF monitor pcap file and compare the latency. If you are capturing on multiple radios at once, you may wish to merge the captures with the mergecap tool or similar tool.



Watch ports and stations with the Monitor Port plugin.

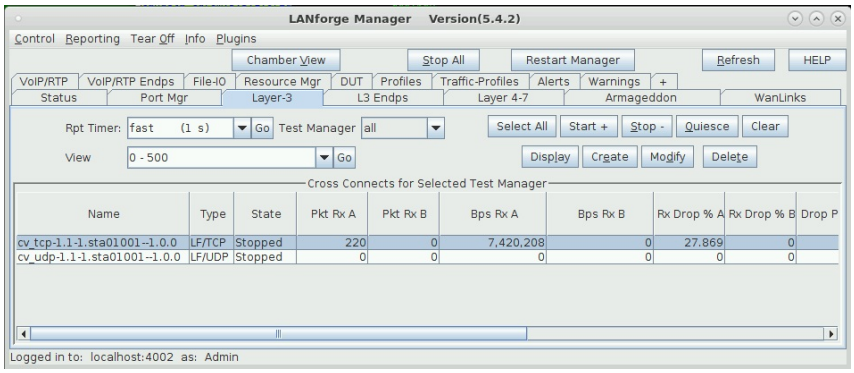
Goal: Display a bandwidth meter for ports and stations.

You can display a series of meters for one or more ports with a nice demo window for traffic speed. For this demonstration, we have configured a station, an upstream port, and a Layer-3 connection to generate traffic.



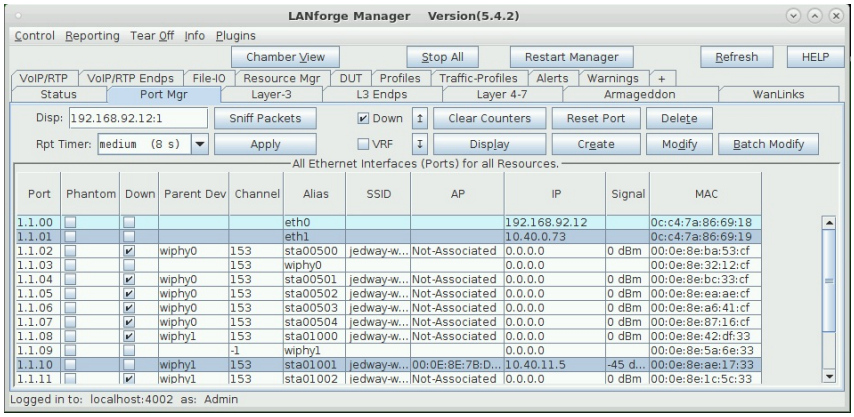
1. ### Single port display

2. The simplest example for port monitoring is showing only one station.
3. We will start our Layer-3 connection.

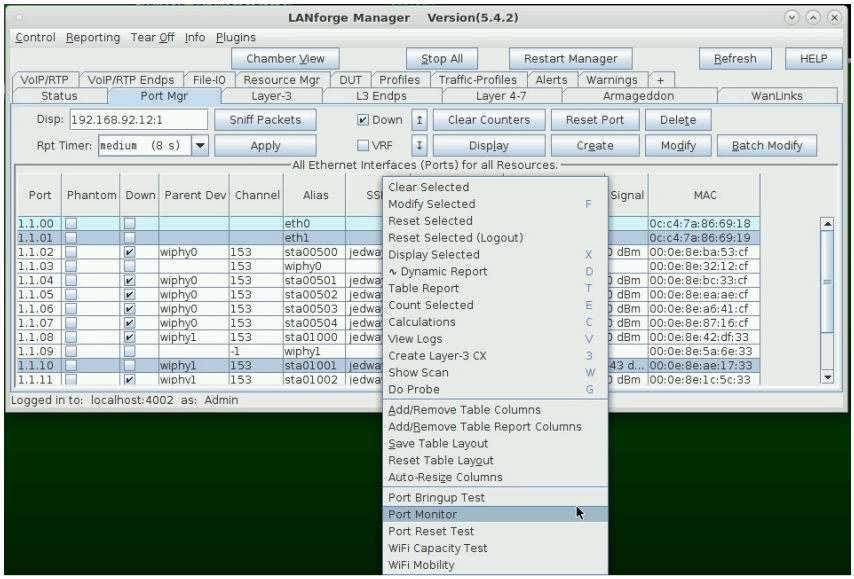


For more information see Generating Traffic to a Switched Network

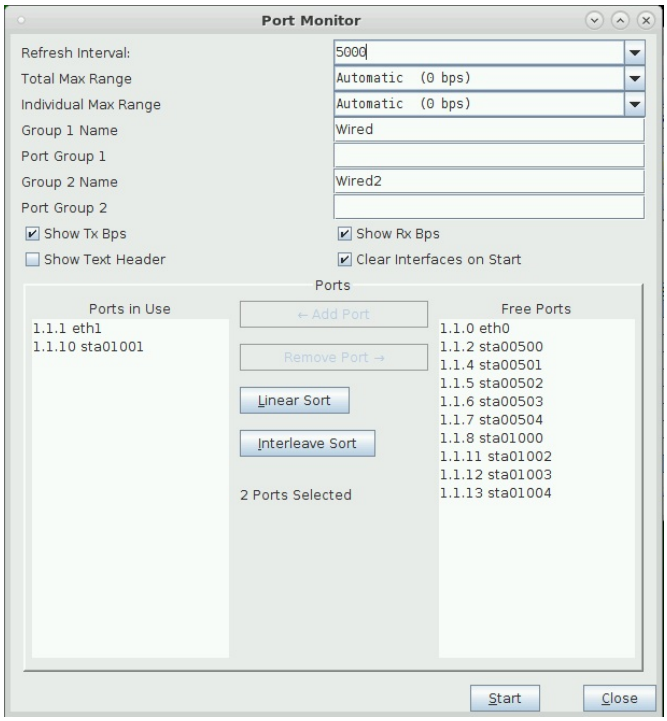
4. After starting the Layer-3 connection, click on the Port Mgr tab; you will see your connection ports highlighted.



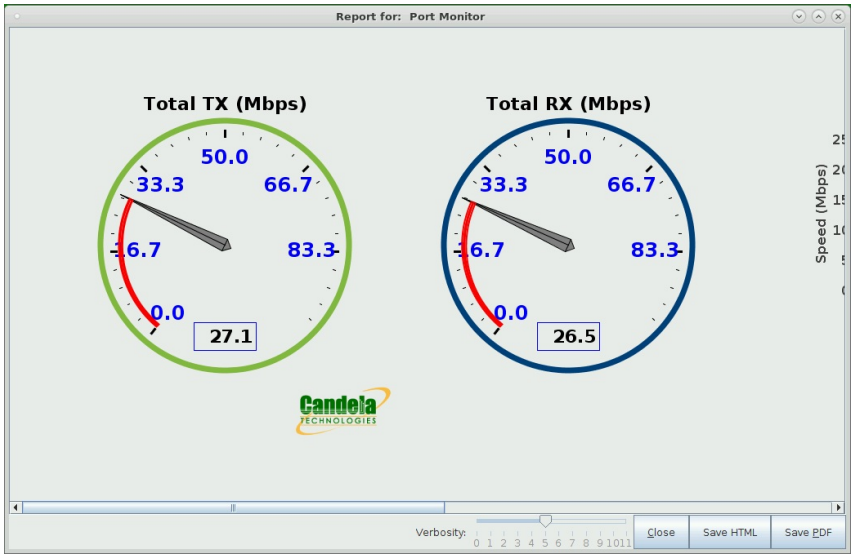
5. Right click on one of the selected ports, and select **Port Monitor**.



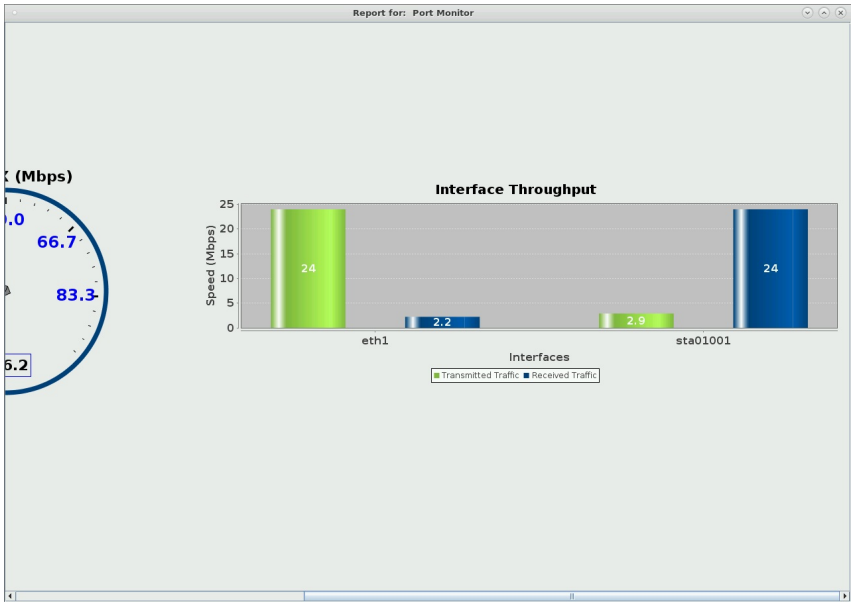
6. You will see your two ports in the port selector. Click **Start** to begin the monitor.



7. The monitor display was designed to be a full-screen window. You'll see it scroll off to the right. Maximize the window to see more.



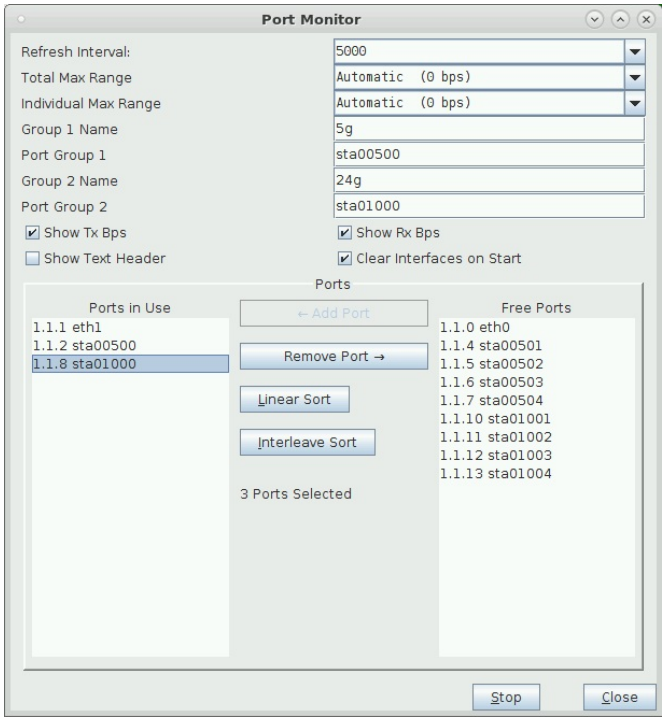
8. The bar chart on the right side shows port totals.



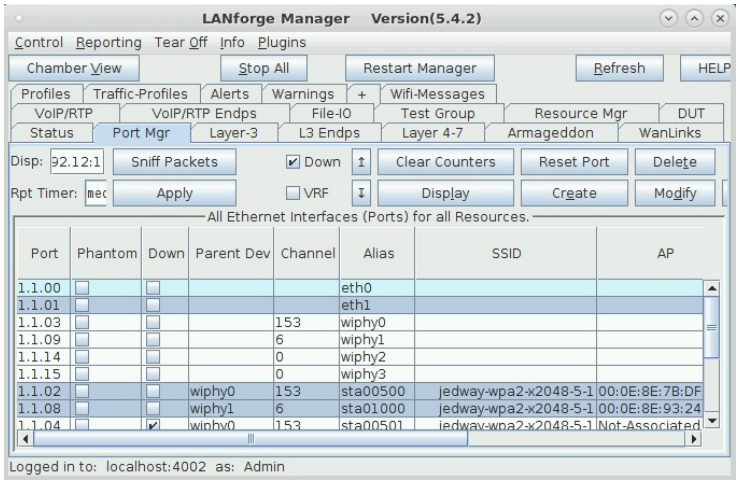
9.

Displaying Groups of Ports

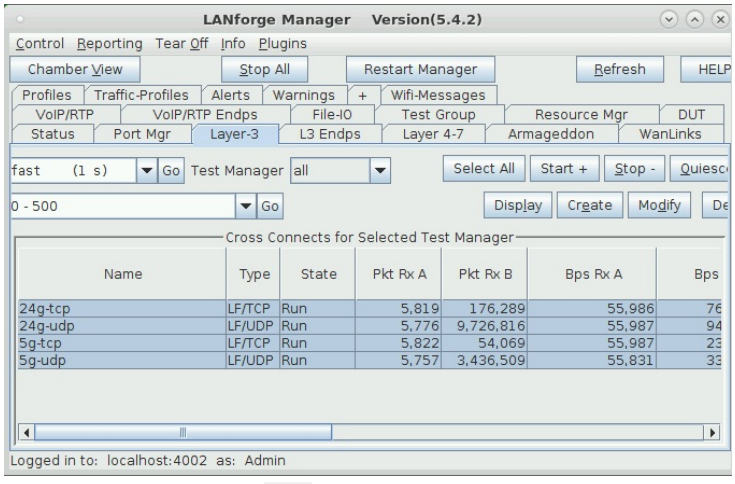
10. You can use the Port Group 1 and Port Group 2 fields in the settings window to display items. The window pictured here shows a station on a 2.4Ghz band and another station on a 5Ghz band.



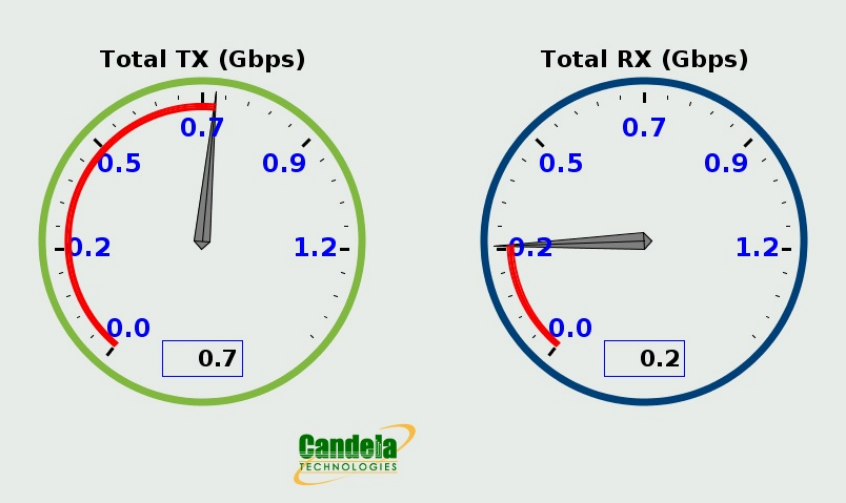
11. The ports for these settings show different channels



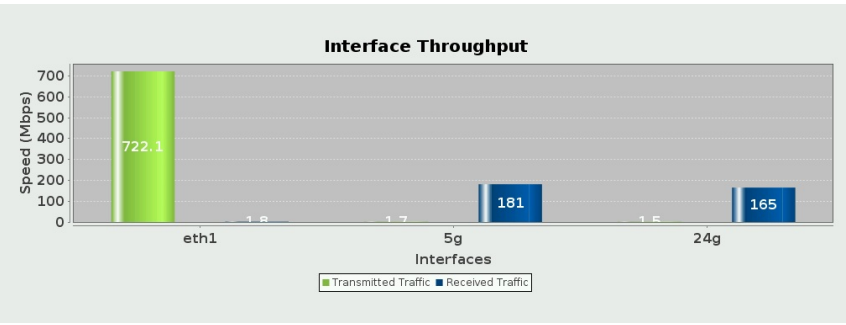
12. The connections for these ports are both attempting 300Mbps download



13. We see a lot of TX from `eth1` in the dial, but not much recieved.

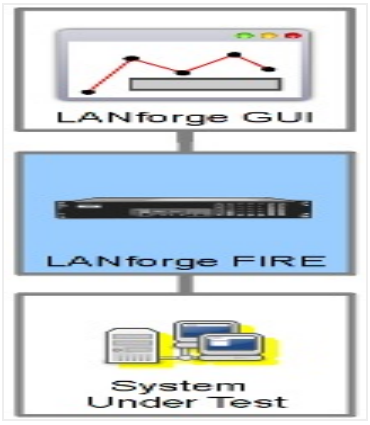


14. We can compare the 2.4Ghz ports and the 5Ghz ports in the bar charts



CI/CD Lights-Out Chamber Setup

Goal: Assemble a LANforge and a device under test (DUT) to operate in an unattended lab setup.



The CI/CD lights-Out chamber is composed of a CT840a chamber, a CT523c LANforge and a test-controller[1], that connects to them via serial and Ethernet.

1) A test controller is a Linux system that can be remotely accessed, and does not need LANforge installed.

Inside the chamber we have:

- a remotely controlled power switch
- a powered USB hub for connecting serial ports of the LANforge and DUT
- a LED lamp
- a USB camera connected to the LANforge machine
- a short table to place above the LANforge for the DUT
- the LANforge machine
- the DUT

Once assembled, this setup can perform a battery of connectivity and traffic tests that do not require a programmable attenuator. Requires LANforge 5.4.2.

1. ## Parts review

Let's review the parts the chamber setup requires:

A. Cables include:

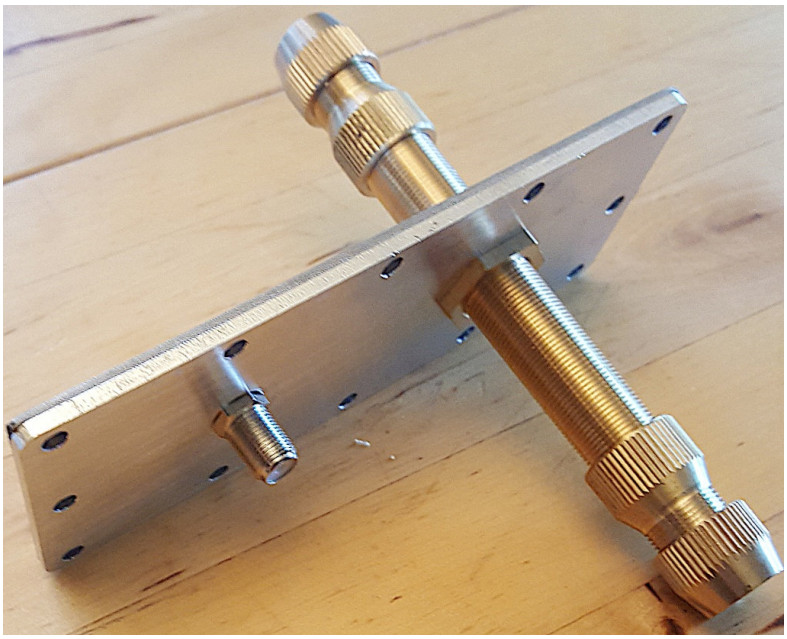
   A. 3 cat5e cables
   B. 2 cat6 cables
   C. 1 USB serial adapter
   D. 1 DB9 female-female cable
   E. Also shown are annenas, with-pin

B. You will also get two small monitor stands and a network power switch. You will be using the legs of both, but discarding one of the stand tops.

C. You will be getting a LED lamp, a USB camera, 8 port network switch, USB hub, camera clamp, USB A-A cables.

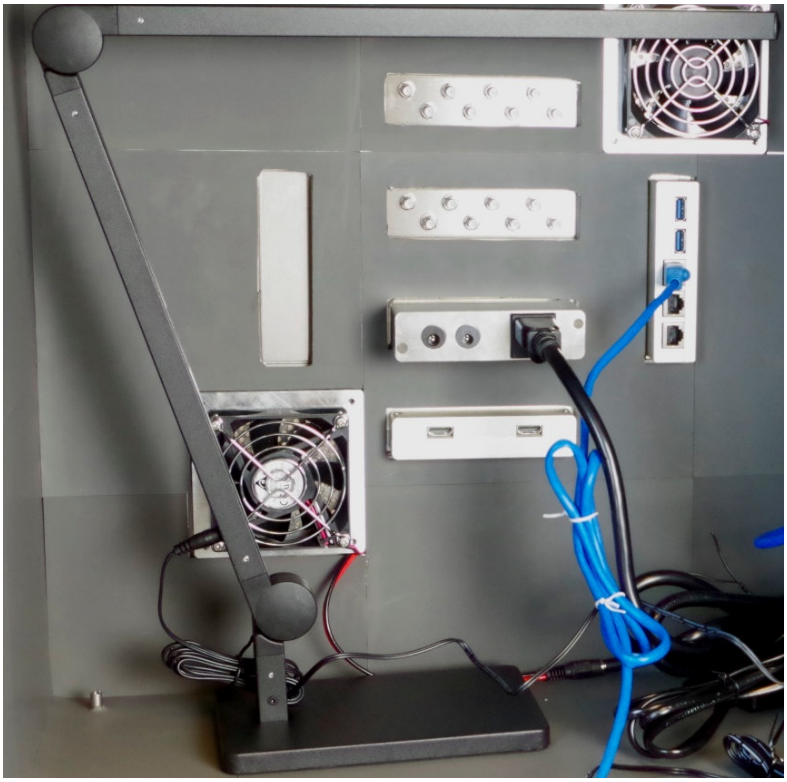
D. Your chamber will come with a universal power strip, AC power cord, fan AC-DC power adapter, and a printed test report. This assembly guide does not use the universal power strip. You might find a use for it.

E. Your chamber will also come with a brass pipe mounted to a steel plate. This is a fiber tube that you can pass fiber optic cabling through. For this setup, **you can refrain from installing the fiber-pipe**. It is not necessary because we are using copper Ethernet cabling.



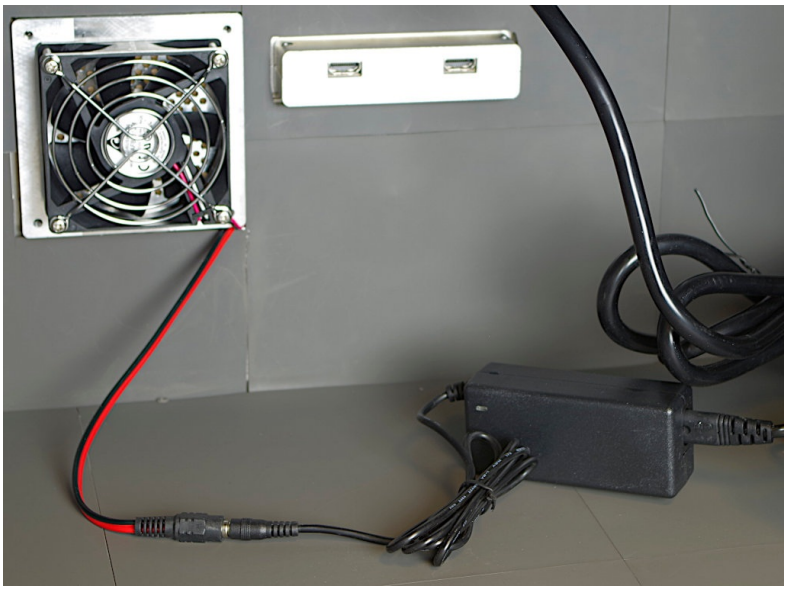
2. **First items**

A. Place the lamp in the chamber



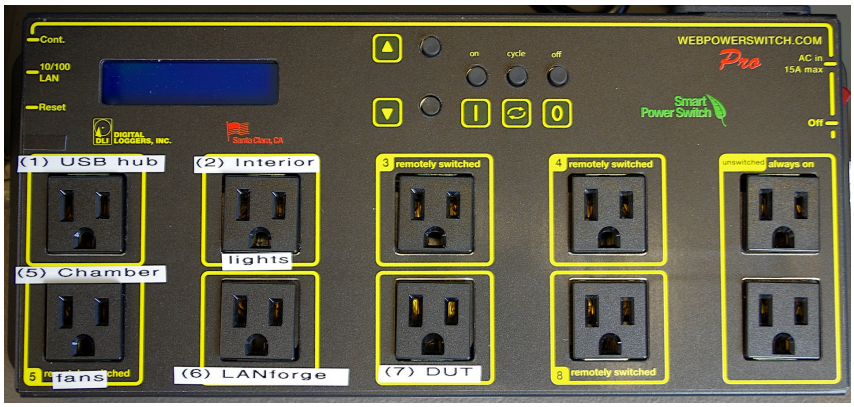
B. Place the power distribution unit (PDU) in the chamber on it's side.

C. Plug the DC barrel connector for the fans into the fan power supply



D. Plug the USB hub USB cable into back of the chamber. The bottom USB port is chamber `USB 1` near the top ethernet filter port `3`. The top USB port is chamber `USB 2`.

3. ## Power Switch



A. Port 1: USB Hub

B. Port 2: Interior Light

C. Port 5: 5 Chamber Fans

D. Port 6: LANforge system

E. Port 7: DUT

F. Others are un-labeled. If you want to add an Ethernet switch in here, we suggest plugging it into one of the **always on** ports on the right side.

4. ## Assemble the USB camera

A. Your camera clamp and USB camera. Your USB camera might be manual-focus.

B. Screw the bolt of the clamp into the tripod mount of the camera

C. Tighten the clamp to the arm of the lamp near the top joint. The USB cable should be plugged into the LANforge when it is added. If you plug it into the USB hub, only the test-controller will be able to use it.

5. **Check espresso levels in human system**

6. **Power cables**

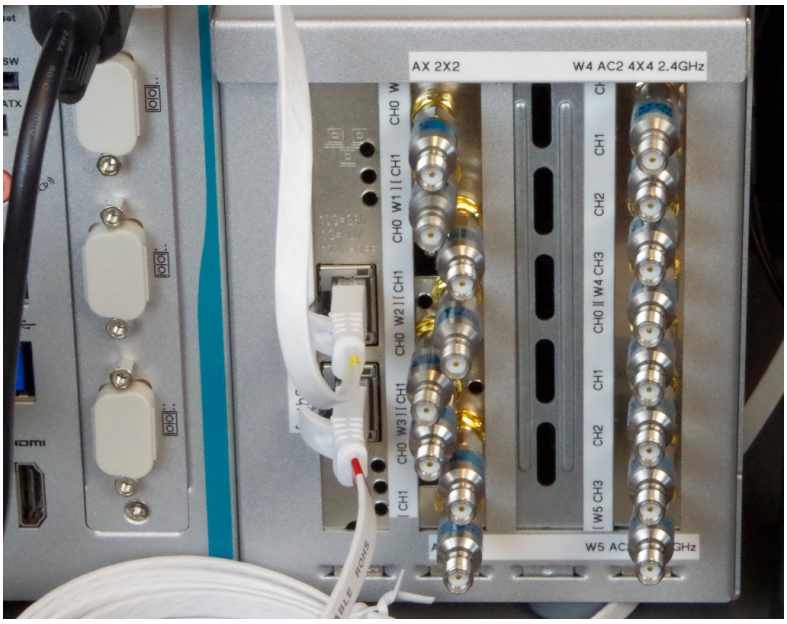
A. Plug the PDU cable into the rear of the chamber
B. Plug the USB hub power into port 1
C. Plug the lamp cable into port 2
D. Plug the chamber fan AC cable into port 5
E. Plug the LANforge power-supply into port 6
F. You will probably plug in your DUT power supply last, into port 7

7. ## Inline Attenuators

A. This is an SMA connector chart. Make sure your antennas and in-line attenuators have the correct pins

B. Your 16 LANforge SMA terminals are SMA-Female.

C. Screw 16 SMA-Male/SMA-Female inline attenuators onto your CT523c.



8. ## Antennas

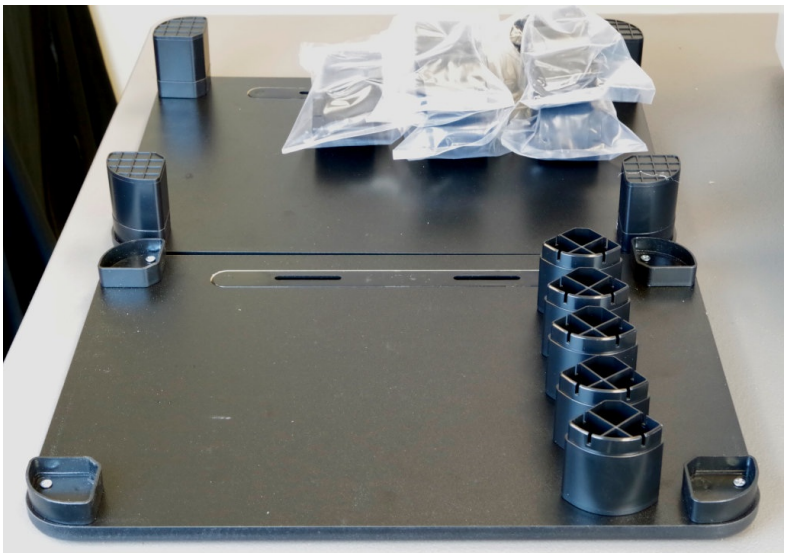
A. Check that you have SMA-Male antennas (antennas are with-pin).

B. Screw on your antennas. You will not leave them straight like this.

C. Bend the antennas various ways to ensure they provide diversity. Not providing diverse antenna orientation means your equipment will not reach desired MCS rates.

9. **Shelf for DUT**

A. You are provided two monitor stand kits as to combine into a shelf for the DUT to rest above the LANforge CT523c. You are going to use the legs from the second kit to extend the legs of the first kit.



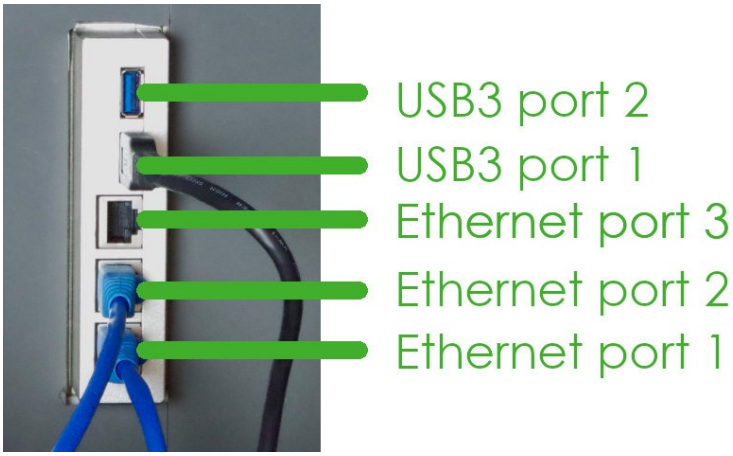
B. Use a utility knife or a screw driver to separate any feet from legs sections you do not need.

C. Here is a shelf with five segments per leg. Depending on the size of the rubber feet on your CT523c, it might be just tall enough. The other photos show a table with six segments per leg. Looks like Batman likes my work.

D. This table clears the LANforge unit well.

10. **Network and Serial Cables**

A. The network ports on the chamber are passive Ethernet RF filters. The do not have activity lights, they do not require power.

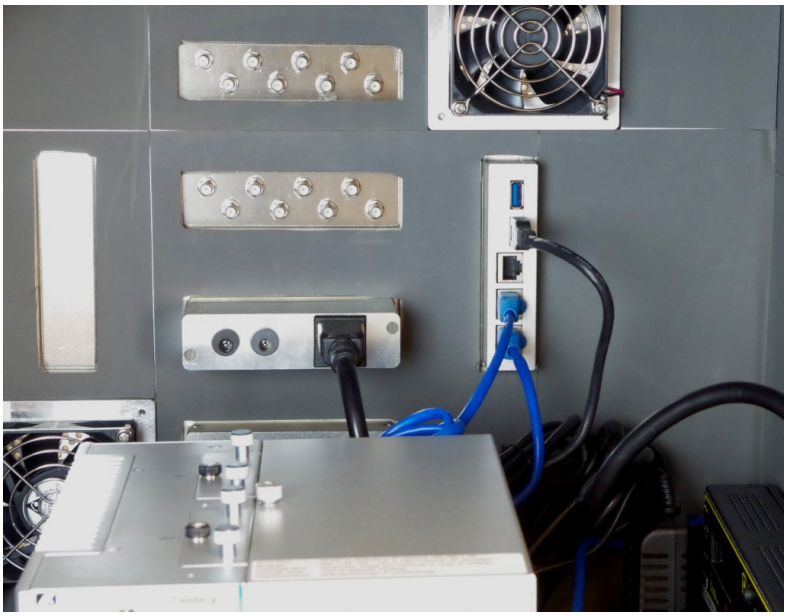


A. The top USB3 port is USB 2
B. The bottom USB3 port is USB 1
C. The top Ethernet port is 3
D. The middle Ethernet port is 2
E. The bottom Ethernet port is 1

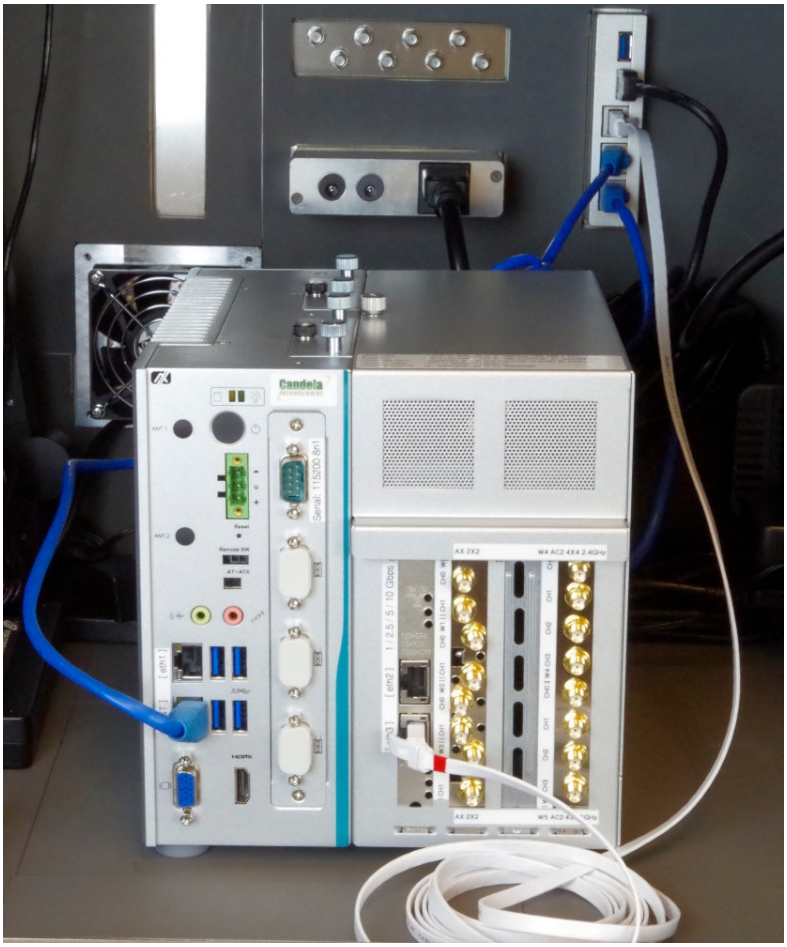
B. Cable the PDU network to chamber Ethernet port `1`

C. Place the LANforge CT523c in the chamber and attach a CAT5e cable to the management port labeled `[ MGT ]`

D. Plug the management port cable into the chamber port `2`



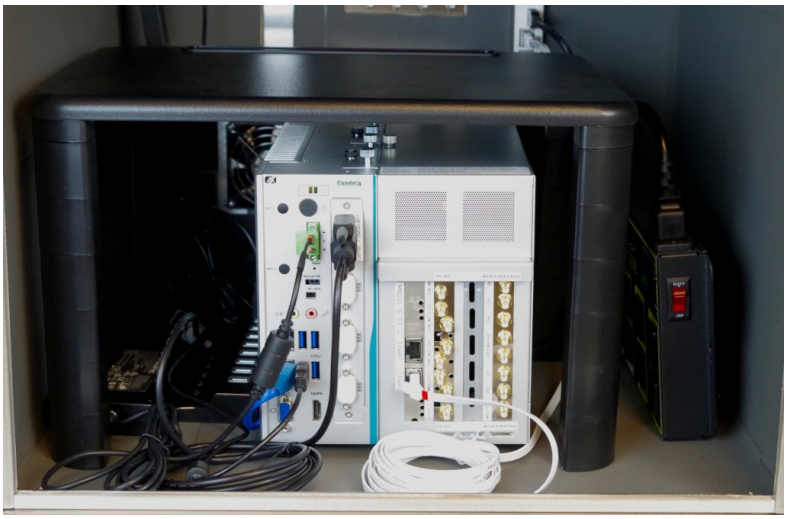
E. Use a CAT6 cable to connect the LANforge `[ eth3 ]` port to the chamber `3` port. This represents your WAN connection.



F. Connect the DB9 female-female serial cable to the CT523c serial port. It is labled `Serial: 115200 8n1`. Connect the USB serial adapter to the other end of the DB9 cable.

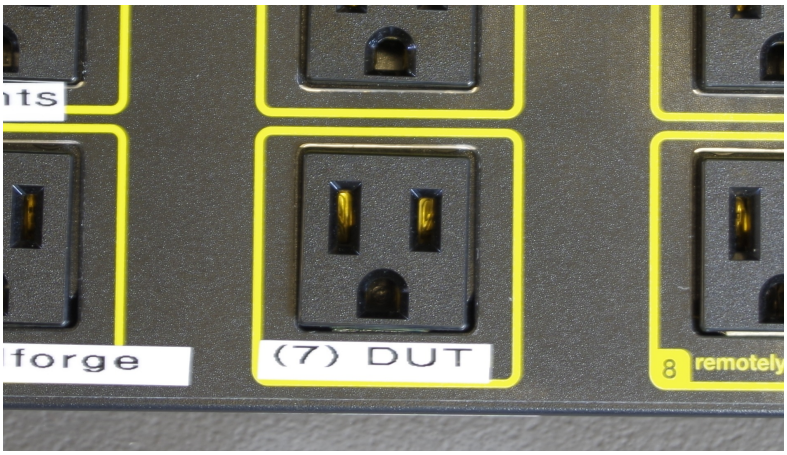
G. Plug the USB end of the USB serial adapter into the USB hub. Your test controller will be able to login to the LANforge for network configuration and debugging.

H. Plug in power to the CT523c. It is a green DC connector. Place the table over the CT523c.

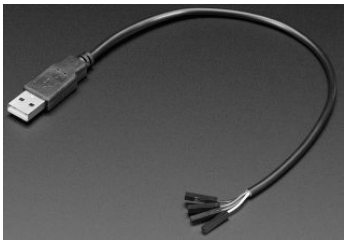


I. You may place the DUT on the table. Use a CAT6 cable to connect the DUT to the LANforge `[ eth2 ]` port. LANforge will serve DHCP on `[ eth2 ]` for the DUT and its connected stations, and will NAT and route packets out of `[ eth3 ]`.

J. Plug your DUT power to into PDU port 7



K. Your DUT probably has a serial connector. Cable your DUTs serial cable into the USB hub.



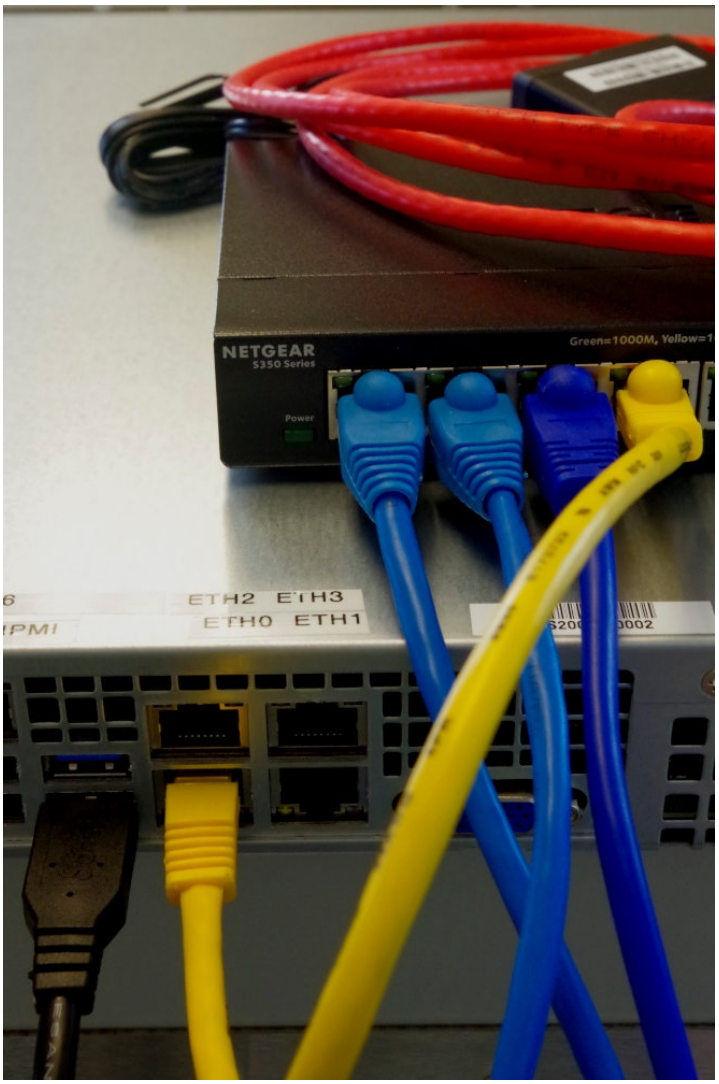
A. The test controller (aka Jump Host) pictured here is a 1U rack unit.

A. Your rack KVM will use the left USB2 ports and VGA port of the test-controller.
B. Connect chamber port `USB 1` to a USB3 port on the test-controller.
C. You might have to use an extra USB3 hub to control more than two CT820a chambers.
D. Cable chamber ethernet `1` to switch, this is your PDU
E. Cable chamber ethernet `2` to switch, this is your LANforge management port.
F. The picture shows chamber ethernet `3` connected to the switch. This connection is at your discretion. You might have a different WAN upstream network to attach to chamber ethernet `3`
G. The USB ports may be renamed each time the system restarts. To fix this, you can create an **/etc/udev/rules.d/81-usb-serial.rules** file that defines the USB ports by name using the serial-number of the USB cable if it supports it, or the path (effectively port to which the USB cable is connected).

```
#LF on cable with serial number
SUBSYSTEM=="tty", ENV{ID_SERIAL_SHORT}=="AK066NLY", SYMLINK+="ttyLF1",
MODE="0666"
# AP
# In case we have something w/out a serial number
SUBSYSTEM=="tty", DEVPATH=="1-2.1.4", SYMLINK+="ttyAP1", MODE="0666"
```

You can find the appropriate information with the **udevadm info -n /dev/ttyUSB0** command.

B. In the picture the red cable represents the control network. The yellow cable is your connection to your test controller `ETH0`



Create Python Scripts Utilizing the Realm Library

Goal: Create a python script to create stations and Layer-3 cross connects

Using the `realm.py` library we will write a script that will allow us to automate the creation of stations and Layer-3 cross connects. We will also be able to start and stop traffic over the cross connects using the script. We will be referencing the script, test_ipv4_variable_time.py, as an example throughout this cookbook. Requires LANforge 5.4.2.

---

1. ### Starting the script

   A. Setting up inheritance for our object

      A. In order for our script to be platform independent we will need to `import sys`. Then use

      ```
      if 'py-json' not in sys.path:
          sys.path.append(os.path.join(os.path.abspath('..'), 'py-json'))
      ```

      B. When creating our object we will need to import the LFCliBase module from the LANforge module using `from LANforge.lfcli_base import LFCliBase`

      C. After importing LFCliBase we can create our Class and inherit from LFCliBase

B. Setting up the main method

A. The main method will typically follow a pattern:

I. First, the creation of a list of stations. This can be done in many ways.
Example:

```
station_list = LFUtils.port_name_series(prefix_="sta",
    start_id_=0,
    end_id_=4,
    padding_number_=10000)
```

II. Following the station list, we can initialize our object:

```
ip_var_test = IPV4VariableTime(lfjson_host, lfjson_port,
    number_template="00",
    sta_list=station_list,
    name_prefix="var_time",
    ssid="testNet",
    password="testPass",
    resource=1,
    security="wpa2",
    test_duration="5m",
    side_a_min_rate=256,
    side_b_min_rate=256)
```

III. After our object has been initialized we can begin the testing process. The preferred order for running our tests is to:

i. Call `cleanup()` to prevent stations, cross-connects, and endpoints within our list from having creation issues if anything exists with the same name.

ii. Call the `build()` method in our class to setup the basic versions of the stations, cross-connects, and endpoints.

iii. Call the `start()` method that will start the test itself, as well as any bring up any stations and start traffic on cross-connects that need it.

iv. Call the `stop()` method to stop the traffic and bring down any stations that are up.

v. Verify that the tests passed using our inherited passes() method.

vi. After verifying a pass we can then call our cleanup function again to clean up everything we worked with.

C. Example Main Method

```
def main():
    lfjson_host = "localhost"
    lfjson_port = 8080
    station_list = LFUtils.portNameSeries(prefix_="sta", start_id_=0, end_id_=4, padd
    ip_var_test = IPV4VariableTime(lfjson_host, lfjson_port, number_template="00", st
        name_prefix="var_time",
        ssid="testNet",
        password="testPass",
        resource=1,
        security="wpa2", test_duration="5m",
        side_a_min_rate=256, side_b_min_rate=256),
    ip_var_test.cleanup(station_list)
    ip_var_test.build()
    if not ip_var_test.passes():
        print(ip_var_test.get_fail_message())
        exit(1)
    ip_var_test.start(False, False)
    ip_var_test.stop()
    if not ip_var_test.passes():
        print(ip_var_test.get_fail_message())
        exit(1)
    time.sleep(30)
    ip_var_test.cleanup(station_list)
    if ip_var_test.passes():
        print("Full test passed, all connections increased rx bytes")
```

2. ## Test Methods Available With Realm

A. Using `lfcli_base._pass()` and `lfcli_base._fail()`

A. Since our class is inheriting lfcli_base.py, we have access to methods that will help us keep track of passes and fails during our tests. We can access them using `self._pass()` or `self._fail()`. They will take two parameters, a string `message` and an optional boolean `print_pass` and `print_fail` for `_pass()` and `_fail()` respectively. If `print_pass` or `print_fail` are set to True, they will write the message to stdout whenever the functions are called.

B. `lfcli_base` will add a "PASSED: message" or "FAILED: message" to a list when the tests pass or fail. This list can be accessed using the methods

```
get_result_list()
get_failed_result_list()
get_fail_message()
get_all_message()
```

B. Using lfcli_base to check test success

A. `passes()` will return a boolean depending on whether or not there were any fails in the test. If it finds a fail message it will return False, if none are found it will return True.
`get_result_list()` will return all logged pass/fail messages as a list.
`get_failed_result_list()` will return a list of only fail messages.
`get_fail_message()` will return a list of string of fail messages separated by newlines
`get_message()` will return a list of string of all messages separated by newlines

3. ## Building a Station

A. Build Method

A. We will need to do a number of things to setup our build method.

I. To begin we will set the security type of our stations using `station_profile.use_security()`

II. We will then use `station_profile.set_number_template()` to name our stations

III. After this we can set our command flags and parameters using

```
self.station_profile.set_command_flag("add_sta","create_admin_down",1
self.station_profile.set_command_param("set_port","report_timer",1500
self.station_profile.set_command_flag("set_port","rpt_timer", 1)
```

IV. Once our parameters and flags are set, we can pass a list of stations to `station_profile.create()` and `cx_profile.create()`. Our build function could look like this:

```
for station in range(len(self.sta_list)):
    temp_sta_list.append(str(self.resource)+"."+self.sta_list[station])
self.station_profile.create(resource=1, radio="wiphy0", sta_names_=self.sta_list, debug=Fals
self.cx_profile.create(endp_type="lf_udp", side_a=temp_sta_list, side_b="1.eth1", sleep_time
self._pass("PASS: Station build finished")
```

> i The naming convention for the sides will look like **`foo-A`** for side_a and **`foo-B`** for side_b. foo will be set based on the names in the list of stations given.

B. StationProfile

A. The preferred method for creating a station_profile is to use the factory method `new_station_profile()` found in realm

I. We will need to assign some variables for the creation of our stations before we can call `create()`.

i. `self.station_profile.use_security(`*`security_type`*`, `*`ssid`*`, `*`passwd`*`)` is the preferred method to use when setting the security type, ssid, and password variables
Example:
`self.station_profile.use_security("wpa2", "testNet", "testPass")`

ii. `self.station_profile.number_template_` is the numerical prefix for stations. Using a `number_template` of "00" will have stations look like sta01, sta02...sta10
Example:
`self.station_profile.number_template_="00"`

iii. `self.station_profile.mode` determines the wifi mode used by the stations. See here for available modes
Example:
`self.station_profile.mode=0`

4. ## Cross Connects

A. Starting and Stopping Traffic

A. In order for us to be able to start traffic, our stations will need to be admined up, associated, and with an IP. We can bring them up using `station_profile.admin_up()`. We can then use `realm.wait_for_ip(resource, sta_list)` to wait for our stations, as well as eth1, to get an IP address.

B. Once we are sure all of our stations have ip addresses, we can use `cx_profile.start_cx()` to start the traffic for our cross-connects. When we decide to stop the traffic we can just as easily use `cx_profile.stop_cx()` to stop traffic.

B. L3CXProfile

A. `self.local_realm.create_new_l3_cx_profile()` is the preferred method for creating a new Layer 3 CX Profile.

I. We will need to assign some variables for the creation of our stations before we can call `create()`.

i. `self.cx_profile.name_prefix` will be used to specify the name prefix for the cx. Assigning `self.cx_profile.name_prefix` to "test_" would produce cross-connects named test_sta00 with the numbers being dependent on station_profile's `number_template`.
Example:
`self.cx_profile.name_prefix="test_"`

ii. Set the `_min_bps` to the desired amount. `_max_bps` can be set but typically defaults to 0 which sets it to the same as the minimum bps.
Example:

```
self.cx_profile.side_a_min_bps=56000
self.cx_profile.side_b_min_bps=56000
```

5. ## Using TTLS

A. TTLS setup requires a few pieces of information to work correctly. `StationProfile` has a `set_wifi_extra()` method for setting the relevant variables. See here for the available options

B. We will need a key management type (`key_mgmt`), an EAP method (`eap`), an EAP identity string (`identity`), an EAP password string (`passwd`), an 802.11u realm (`realm`), an 802.11u domain (`domain`), and an 802.11u HESSID (`hessid`)
Example:

```
key_mgmt="WPA-EAP"
eap="TTLS"
identity="testuser"
passwd="testpasswd"
realm="localhost.localdomain"
domain="localhost.localdomain"
hessid="00:00:00:00:00:01"
```

We can then use these variables to call the `set_wifi_extra()` method
Example:

```
station_profile.set_wifi_extra(key_mgmt, eap, identity, passwd, realm, domain, hessid
```

6. ## Cleaning Up

A. Cleanup stations and cross connects

A. We have two options for cleaning up everything once we finish:

I. The preferred method to cleanup is to use the individual cleanup methods found in `StationProfile` and `L3CXProfile`. These are `station_profile.cleanup(resource, desired_station_list)` and `cx_profile.cleanup()`. These methods are preferred because they will only delete stations, cross-connects, and endpoints created during the test while leaving others untouched.This is useful if you are running other scripts in the background.

II. The other method for cleanup is to use `Realm`'s `remove_all_stations(), remove_all_endps(), and remove_all_cxs()` methods. These will remove all stations, cxs, and endpoints that exist. These are good for doing a full cleanup, and it is recommended to use them in the order of cx, endpoint, station to prevent potential issues or missed deletions.

7. ## Debugging Stations

A. Debug information for station creation can be output by setting `_debug_on=True` in `StationProfile.create()`

A. There are a few important debug outputs to pay attention to:

I. This is the debug output that appears when using the add_sta command. This is used frequently in `StationProfile.create()`. This debug output will allow you to troubleshoot any **flags** or other information that is being set when creating your stations. It will output the name at the top and the raw JSON data will follow.

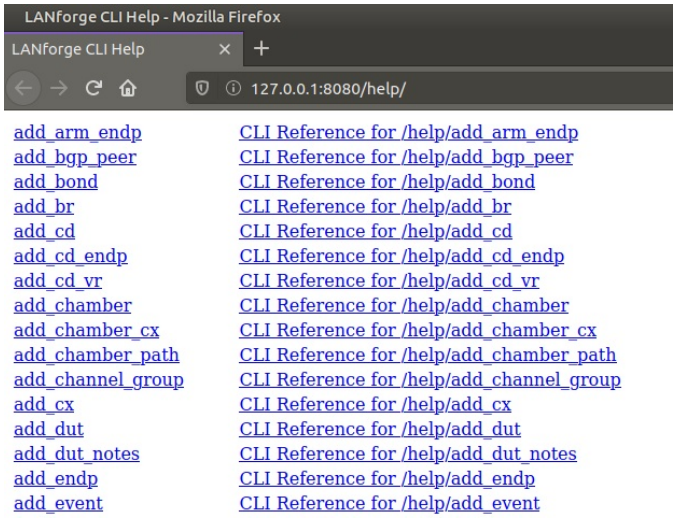
```
- 381 - sta0000- - - - - - - - - - - - - - - - - -
{'flags': 132096,
'flags_mask': 68719608832,
'key': 'testPass',
'mac': 'xx:xx:xx:xx:*:xx',
'mode': 0,'
radio': 'wiphy0',
'resource': 1,
'shelf': 1,
'ssid': 'testNet,
'sta_name': 'sta0000'}
```

II. The next bit of debugging output comes from using the set_port command. We are able to see all of the JSON data that is posted, and can use this to check our flags and other info.

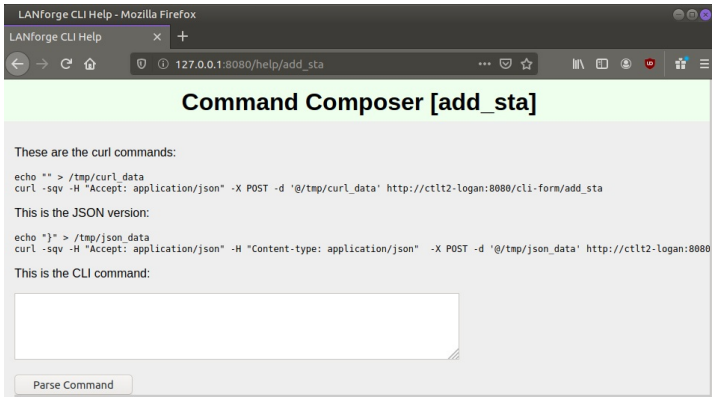
```
{'current_flags': 2147483649,
'interest': 8437762,
'port': 'sta0000',
'report_timer': 1500,
'resource': 1,
'shelf': 1}
<LANforge.LFRequest.LFRequest object at 0x7f13dbc56850>
-~381 - - - - - - - - - - - - - - - - - - - -
```

B. There are a few steps we can take to make validating the information we get through debugging easier.

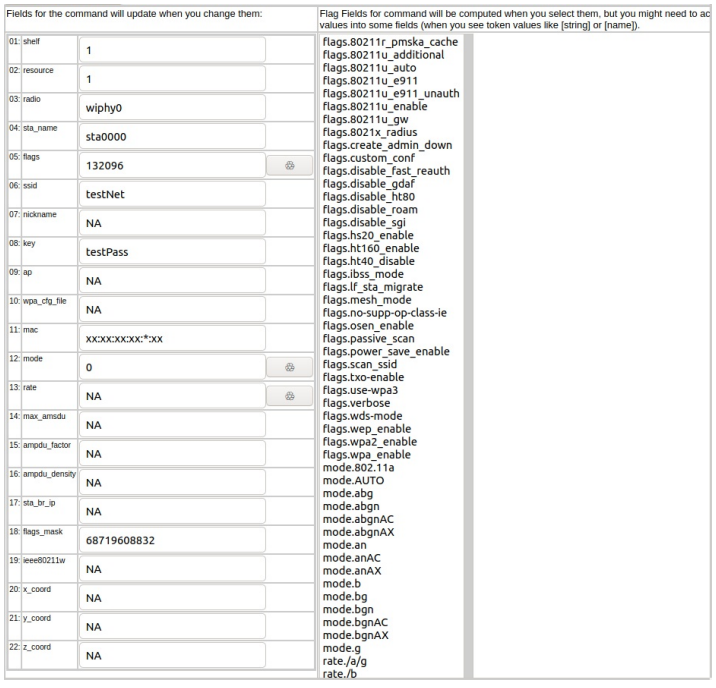
A. We can use the help page available on the address of the machine LANforge is running on. http://127.0.0.1/help/ will take us to a page containing all of the commands we can get help with.



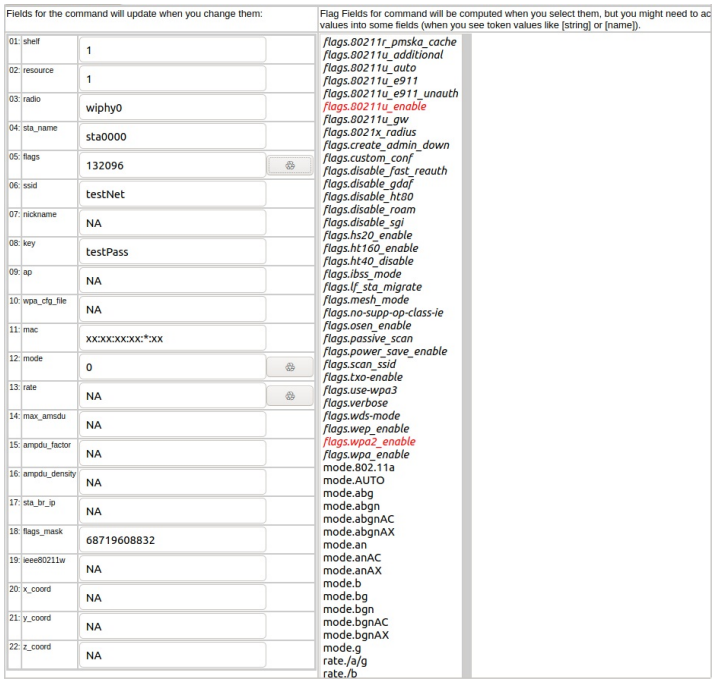
B. Using http://127.0.0.1/help/add_sta will bring us to a page specific to the `add_sta` command.



C. Here we can enter all of the data we got from our debugging output into the correct areas.



D. Flag fields have a button next to them that will calculate and highlight relevant flags in the right hand column of the page. This can be useful for checking that the correct flags are being set.



E. After we have done this, we can click the **parse command** button towards the top of the data inputs. We can then enter this command into LANforge's messages tab in the input box.

Create Python Scripts To Test Layer 4 Traffic

Goal: Create a script to test Layer 4 traffic using Realm

Using the `realm.py` library we will write a script that will allow us to automate the creation of stations and Layer 4 cross connects. We will also be able to start and stop traffic over the cross connects using the script. Station and Cross Connect creation is covered in the Realm Scripting Cookbook. Requires LANforge 5.4.2.

---

1. ### Creating The Profile

A. We will use the factory method `self.local_realm.new_l4_cx_profile()` to create our profile object.

B. After we have done this we can set a few variables for our traffic:

A. `l4_cx_profile.requests_per_ten` will set our rate of requests per ten minutes. Setting `requests_per_ten = 600` will set our URL request rate to 1 per second. There is no limit to what can be used as the rate but common rates are:

- 600 : 1/s
- 1200 : 2/s
- 1800 : 3/s
- 2400 : 4/s

B. `l4_cx_profile.url` is the URL to be used in the requests. We will also need to specify the direction (dl/ul) and a absolute path for the destination. See syntax here.
Example:
```
l4_cx_profile.url = "dl http://10.40.0.1 /dev/null"
```

C. Example Layer 4 profile init:
```
class IPV4L4(LFCliBase):
    def __init__(self, host, port, ssid, security, password, url, requests_pe
        target_requests_per_ten=600, number_template="00000", resource=1, num
        _debug_on=False,
        _exit_on_error=False,
        _exit_on_fail=False):
    super().__init__(host, port, _debug=_debug_on, _halt_on_error=_exit_on_er
    self.host = host
    self.port = port
    self.ssid = ssid
    self.security = security
    self.password = password
    self.url = url
    self.requests_per_ten = requests_per_ten
    self.number_template = number_template
    self.sta_list = station_list
    self.resource = resource
    self.num_tests = num_tests
    self.target_requests_per_ten = target_requests_per_ten

    self.local_realm = realm.Realm(lfclient_host=self.host, lfclient_port=sel
    self.cx_profile = self.local_realm.new_l4_cx_profile()
    self.cx_profile.url = self.url
    self.cx_profile.requests_per_ten = self.requests_per_ten

    # Station Profile init
```

2. ## Starting Traffic

A. When running traffic, if you plan to measure the rate of requests, it is recommended to do so in 10 minute increments. An example of this can be seen here: test_ipv4_l4_urls_per_ten.py. To start the traffic we can use the `l4_cx_profile.start_cx()` method. To stop the traffic we can use the `l4_cx_profile.stop_cx()` method.

B. Example start and build method:
```
def build(self):
    # Build stations
    self.station_profile.use_security(self.security, self.ssid, self.password)
    print("Creating stations")
    self.station_profile.create(resource=1, radio="wiphy0", sta_names_=self.sta_list,
    temp_sta_list = []
    for station in range(len(self.sta_list)):
        temp_sta_list.append(str(self.resource) + "." + self.sta_list[station])

    self.l4_profile.create(ports=temp_sta_list, sleep_time=.5, debug_=self.debug, sup
```
```
def start(self, print_pass=False, print_fail=False):
    temp_stas = self.sta_list.copy()
    temp_stas.append("eth1")
    cur_time = datetime.datetime.now()
    interval_time = cur_time + datetime.timedelta(minutes=1)
    passes = 0
    expected_passes = 0
    self.station_profile.admin_up(1)
    self.local_realm.wait_for_ip(self.resource, temp_stas)
    self.l4_profile.start_cx()
    print("Starting test")
    for test in range(self.num_tests):
        expected_passes += 1
        while cur_time < interval_time:
            time.sleep(1)
            cur_time = datetime.datetime.now()

        if self.l4_profile.check_errors(self.debug):
            if self.__check_request_rate():
                passes += 1
            else:
                self._fail("FAIL: Request rate did not exceed 90% target rate", print
                break
            else:
                self._fail("FAIL: Errors found getting to %s " % self.url, print_fail
                break
            interval_time = cur_time + datetime.timedelta(minutes=1)
    if passes == expected_passes:
        self._pass("PASS: All tests passes", print_pass)
```

3. ## Examining The Results

A. We can use http://localhost:8080/layer4/list to check our Layer 4 endpoints. Adding a ,,?fields to the end of the URL will allow us to specify what we want to look at. We can separate fields by commas to show more than one at a time.
Example: http:/localhost:8080/layer4/list?fields=name,urls/s,total-urls

- Using `total-urls` will show us the total requests made.
- Using `urls/s` will show us the average URL rate per second.
- Using `rx rate` and `tx rate` will show us the rates of received and transeferred traffic.

We can also use the url http://localhost:8080/layer4/all to see all of the available fields.

B. When checking our results for Layer 4 tests we might want to check for common URL related errors:
   - `acc. denied` will show us the number of times we got an access denied error.
   - `bad-url` will show us the number of times a request was made with an invalid URL.
   - `nf (4xx)` will count the number of 400 errors recieved when making requests to our URL.

Create Python Scripts To Test Generic Traffic

Goal: Create a script to test Generic traffic using Realm

Using the `realm.py` library we will write a script that will allow us to automate the creation of stations and generic cross connects. We will also be able to start and stop traffic over the cross connects using the script. Station and Cross Connect creation is covered in the Realm Scripting Cookbook. Requires LANforge 5.4.2.

---

1. ## Creating The Profile

A. We will use the factory method `self.local_realm.new_generic_cx_profile()` to create our profile object.

B. After we have done this we can set a few variables for our traffic:

   A. `gen_cx_profile.type` will determine the type of command to execute.
   Example: `self.cx_profile.type = "lfping"`

   B. `gen_cx_profile.dest` is the destination IP address for the command.
   Example: `self.cx_profile.dest = "127.0.0.1"`

   C. `gen_cx_profile.interval` sets the interval at which the command is run in seconds.
   Example: `self.cx_profile.interval = 1`

   D. Example Generic profile init:

```
class GenTest(LFCliBase):
    def __init__(self, host, port, ssid, security, password, sta_list, name_p
                 number_template="00000", test_duration="5m", type="lfping",
                 interval=1, radio="wiphy0",
                 _debug_on=False,
                 _exit_on_error=False,
                 _exit_on_fail=False):
        super().__init__(host, port, _debug=_debug_on, _halt_on_error=_exit_o
        self.host = host
        self.port = port
        self.ssid = ssid
        self.radio = radio
        self.upstream = upstream
        self.sta_list = sta_list
        self.security = security
        self.password = password

        self.number_template = number_template
        self.name_prefix = name_prefix
        self.test_duration = test_duration

        self.local_realm = realm.Realm(lfclient_host=self.host, lfclient_port
        self.cx_profile = self.local_realm.new_generic_cx_profile()
        self.cx_profile.type = type
        self.cx_profile.dest = dest
        self.cx_profile.interval = interval

    # Station Profile init
```

2. ## Starting Traffic

A. To start the traffic we can use the `gen_cx_profile.start_cx()` method. To stop the traffic we can use the `gen_cx_profile.stop_cx()` method.

B. Example start and build method:

```
def build(self):
    self.station_profile.use_security(self.security, self.ssid, self.password)
    self.station_profile.set_number_template(self.number_template)
    print("Creating stations")
    self.station_profile.set_command_flag("add_sta", "create_admin_down", 1)
    self.station_profile.set_command_param("set_port", "report_timer", 1500)
    self.station_profile.set_command_flag("set_port", "rpt_timer", 1)
    self.station_profile.create(radio=self.radio, sta_names_=self.sta_list, debug=sel
    self.cx_profile.create(ports=self.station_profile.station_names, sleep_time=.5)
    self._pass("PASS: Station build finished")
```

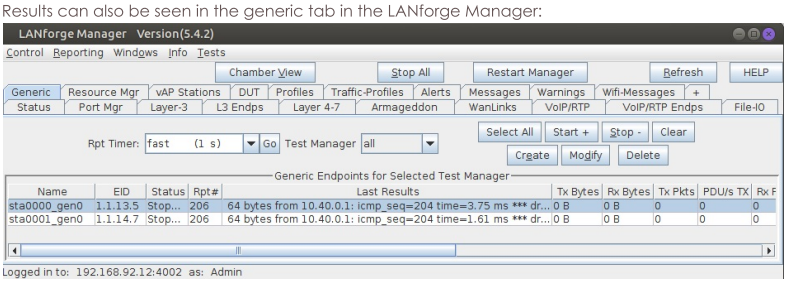
```
def start(self, print_pass=False, print_fail=False):
    self.station_profile.admin_up()
    temp_stas = self.sta_list.copy()
    temp_stas.append(self.upstream)
    if self.local_realm.wait_for_ip(temp_stas):
        self._pass("All stations got IPs", print_pass)
    else:
        self._fail("Stations failed to get IPs", print_fail)
        exit(1)
    cur_time = datetime.datetime.now()
    passes = 0
    expected_passes = 0
    self.cx_profile.start_cx()
    time.sleep(15)
    end_time = self.local_realm.parse_time("30s") + cur_time
    print("Starting Test...")
    while cur_time < end_time:
        cur_time = datetime.datetime.now()
        gen_results = self.json_get("generic/list?fields=name,last+results", debug_=s
        if gen_results['endpoints'] is not None:
            for name in gen_results['endpoints']:
                for k, v in name.items():
                    if v['name'] in self.cx_profile.created_endp and not v['name'].en
                        expected_passes += 1
                        if v['last results'] != "" and "Unreachable" not in v['last r
                            passes += 1
                        else:
                            self._fail("%s Failed to ping %s " % (v['name'], self.cx_
                            break
        time.sleep(1)
    if passes == expected_passes:
        self._pass("PASS: All tests passed", print_pass)
```

3. ## Examining The Results

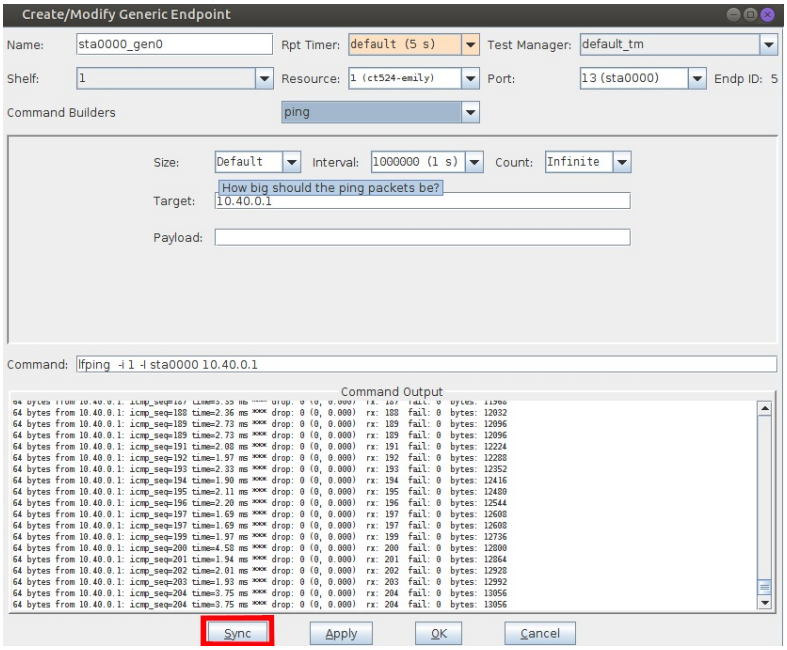
A. For `lfping` we can use the last results of the endpoint to determine if the test was successful. An example of this can be seen in our `start` method. The most common errors for `lfping` will either be a blank last result or `Destination Host Unreachable`. Either of these results indicate a failed ping. Successful pings will look like:

```
64 bytes from 10.40.0.1: icmp_seq=1 time=4.55 ms *** drop: 0 (0, 0.000)  rx: 1  fail:
```

Results can also be seen in the generic tab in the LANforge Manager:



Double-clicking on an endpoint will allow you to see more specific results as well as the command used by the endpoint. Using the **sync** button will allow you to see updated results.



WLAN Theoretical Throughput Calculator for 802.11a/b/g/n/ac standards

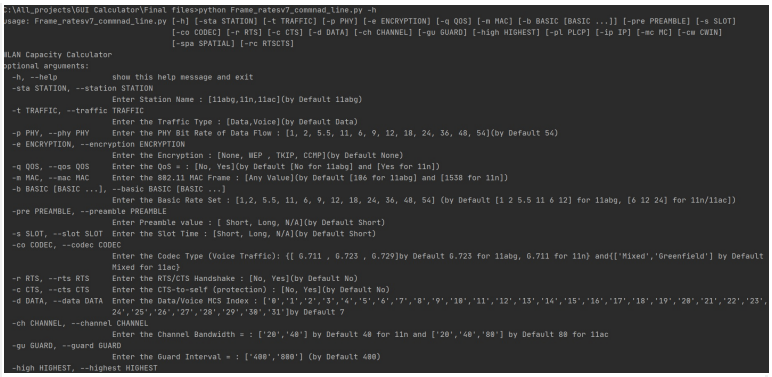
Goal: Use a python script to take various user inputs such as frame sizes, MIMO types, 802.11a/b/g/ac modes etc and calculate the maximum theoretical MAC and IP layer throughput.

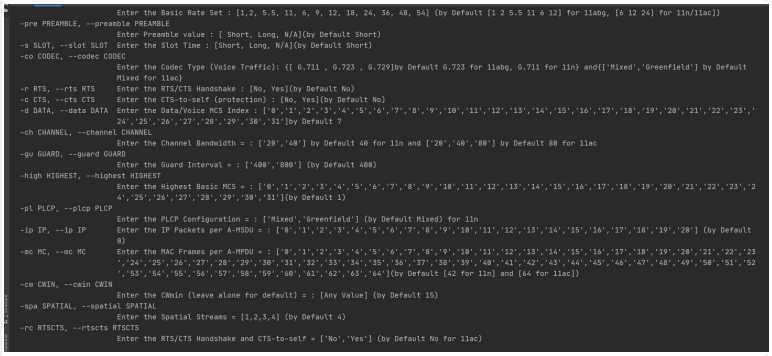
Calculating WLAN theoretical throughput is any important step in being able to benchmark the throughput of Access Points. This calculator will allow users to compare achieved performance to theoretical maximum performance and see how well their APs perform.In this cookbook we show how user can run a simple python script and calculate theoretical throughput for various combinations.Using the script from wlan_capacity_calculator.py.

1. ## Script Input Options

A.

```
python wlan_capacity_calculator.py -h

Output display in terminal:
```





2. ## Run Script for 11abg Calculator

Given below are some example of how we can calculate 802.11abg theoretical throughput

A. By Default:

```
> python wlan_capacity_calculator.py -sta 11abg

or
> python wlan_capacity_calculator.py

Output display in terminal (Client 1):
```

```
C:\All_projects\GUI Calculator\Final files>python Frame_ratesv7_commnad_line.py -sta 11abg

******************Station : 11abgCalculator******************************

Theoretical Maximum Offered Load

1 Client:
{
    "Packet Interval(usec)": "393.50",
    "Max Frame Rate(fps)": 2541,
    "Max. Offered Load (802.11)(Mb/s)": "30.861",
    "Offered Load Per 802.11 Client(Mb/s)": "30.861",
    "Offered Load (802.3 Side)(Mb/s)": "30.496",
    "IP Throughput (802.11 -> 802.3)(Mb/s)": "30.130"
}

Theroretical Voice Call Capacity

{
    "Maximum Theoretical R-value": "N/A",
    "Estimated MOS Score": "N/A",
    "Maximum Bidirectional Voice Calls(calls)": "N/A"
}
```

B. With Input Parameters:

```
python wlan_capacity_calculator.py -sta 11abg -t Voice -p 48 -m 106 -e WEP
-q Yes -b 1 2 5.5 11 -pre Long -s N/A -co G.711 -r Yes -c Yes
```

Output display in terminal (Client 1):

```
{
C:\All_projects\modify wlan calculator\lanforge-scripts-master\py-scripts>python wlan_capacity_calculator.py -sta
11abg -t Voice -p 48 -m 106 -e WEP -q Yes -b 1 2 5.5 11 -pre Long -s N/A -co G.711 -r Yes -c Yes

**********Station : 11abgCalculator**********

Theoretical Maximum Offered Load

1 Client:
{
    "Packet Interval(usec)": "890.91",
    "Max Frame Rate(fps)": 1122,
    "Max. Offered Load (802.11)(Mb/s)": "2.209",
    "Offered Load Per 802.11 Client(Mb/s)": "2.209",
    "Offered Load (802.3 Side)(Mb/s)": "0.700",
    "IP Throughput (802.11 -> 802.3)(Mb/s)": "0.539"
}

Theroretical Voice Call Capacity

{
    "Maximum Theoretical R-value": 85.9,
    "Estimated MOS Score": 4.23,
    "Maximum Bidirectional Voice Calls(calls)": 16.77
}

C:\All_projects\modify wlan calculator\lanforge-scripts-master\py-scripts>
```

3. ## Run Script for 11n Calculator

Given below are some example of how we can calculate 802.11ac theoretical throughput

A. By Default:

```
> python wlan_capacity_calculator.py -sta 11n
```

Output display in terminal (Client 1):

```
C:\All_projects\GUI Calculator\Final files>python Frame_ratesv7_commnad_line.py -sta 11n

**********Station : 11nCalculator**********

Theoretical Maximum Offered Load

1 Client:
{
    "MAC PPDU Interval(usec)": "3601.90",
    "Max PPDU Rate(fps)": "277.63",
    "Max MAC MPDU Rate": 11661,
    "Max MAC MSDU Rate": 11661,
    "Max. 802.11 MAC Frame Data Rate(Mb/s)": "141.605",
    "Max. 802.11 MAC Payload Goodput(Mb/s)": "138.807",
    "MAC Goodput Per 802.11 Client(Mb/s)": "138.807",
    "Offered Load (802.3 Side)(Mb/s)": "139.740",
    "IP Goodput (802.11 -> 802.3)(Mb/s)": "138.060"
}

Theroretical Voice Call Capacity

{
    "Maximum Theoretical R-value": "N/A",
    "Estimated MOS Score": "N/A",
    "Maximum Bidirectional Voice Calls(calls)": "N/A"
}
```

B. With Input Parameters:

```
> python wlan_capacity_calculator.py -sta 11n -t Voice -d 17 -ch 40 -gu 800 -high 9
-e WEP -q Yes -ip 5 -mc 42 -b 6 9 12 24 -m 1538 -co G.729 -pl Greenfield
-cw 15 -r Yes -c Yes
```

Output display in terminal (Client 1):

```
C:\All_projects\modify wlan calculator\lanforge-scripts-master\py-scripts>python wlan_capacity_calculator.py -sta
11n -t Voice -d 17 -ch 40 -gu 800 -high 9 -e WEP -q Yes -ip 5 -mc 42 -b 6 9 12 24 -m 1538 -co G.729 -pl Greenfield
 -cw 15 -r Yes -c Yes

*****************Station : 11nCalculator*****************************

Theoretical Maximum Offered Load

1 Client:
{
    "MAC PPDU Interval(usec)": "505.50",
    "Max PPDU Rate(fps)": "1978.24",
    "Max MAC MPDU Rate": 83086,
    "Max MAC MSDU Rate": 415430,
    "Max. 802.11 MAC Frame Data Rate(Mb/s)": "25.258",
    "Max. 802.11 MAC Payload Goodput(Mb/s)": "-59.822",
    "MAC Goodput Per 802.11 Client(Mb/s)": "-59.822",
    "Offered Load (802.3 Side)(Mb/s)": "N/A",
    "IP Goodput (802.11 -> 802.3)(Mb/s)": "N/A"
}

Theroretical Voice Call Capacity

{
    "Maximum Theoretical R-value": 81.7,
    "Estimated MOS Score": "4.09",
    "Maximum Bidirectional Voice Calls(calls)": 4757.86
}
```

4. ## Run Script for 11ac Calculator

Given below are some example of how we can calculate 802.nabg theoretical throughput

A. By Default:

```
> python wlan_capacity_calculator.py -sta 11ac
```

Output display in terminal (Client 1):

```
C:\All_projects\GUI Calculator\Final files>python Frame_ratesv7_commnad_line.py -sta 11ac

*****************Station : 11ac Calculator*****************************

Theoretical Maximum Offered Load

1 Client:
{
    "MAC PPDU Interval(usec)": "655.10",
    "Max PPDU Rate(fps)": "1526.48",
    "Max MAC MPDU Rate": 97695,
    "Max MAC MSDU Rate": 97695,
    "Max. 802.11 MAC Frame Data Rate(Mb/s)": "1186.408",
    "Max. 802.11 MAC Payload Goodput(Mb/s)": "1162.961",
    "MAC Goodput Per 802.11 Client(Mb/s)": "1162.961",
    "Offered Load (802.3 Side)(Mb/s)": "1170.777",
    "IP Goodput (802.11 -> 802.3)(Mb/s)": "1156.709"
}

Theroretical Voice Call Capacity

{
    "Maximum Theoretical R-value": "N/A",
    "Estimated MOS Score": "N/A",
    "Maximum Bidirectional Voice Calls(calls)": "N/A"
}
```

B. With Input Parameters:

```
python wlan_capacity_calculator.py -sta 11ac -t Voice -d 9 -spa 3 -ch 20 -gu 800
-high 1 -e TKIP -q Yes -ip 3 -mc 0 -b 6 12 24 54 -m 1518 -co Greenfield -cw 15 -rc Yes
```

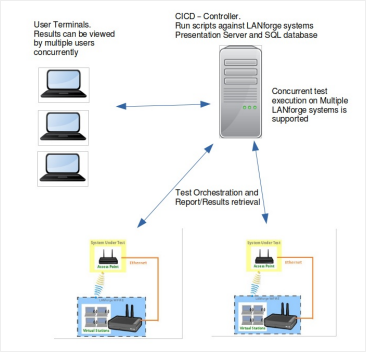
Output display in terminal (Client 1):

```
C:\All_projects\modify wlan calculator\lanforge-scripts-master\py-scripts>python wlan_capacity_calculator.py -sta
11ac -t Voice -d 9 -spa 3 -ch 20 -gu 800 -high 1 -e TKIP -q Yes -ip 3 -mc 0 -b 6 12 24 54 -m 1518 -co Greenfield -
cw 15 -rc Yes

*****************Station : 11ac Calculator*****************************

Theoretical Maximum Offered Load

1 Client:
{
    "MAC PPDU Interval(usec)": "241.50",
    "Max PPDU Rate(fps)": "4140.79",
    "Max MAC MPDU Rate": 4141,
    "Max MAC MSDU Rate": 12422,
    "Max. 802.11 MAC Frame Data Rate(Mb/s)": "3.412",
    "Max. 802.11 MAC Payload Goodput(Mb/s)": "0.000",
    "MAC Goodput Per 802.11 Client(Mb/s)": "0.000",
    "Offered Load (802.3 Side)(Mb/s)": "N/A",
    "IP Goodput (802.11 -> 802.3)(Mb/s)": "N/A"
}

Theroretical Voice Call Capacity

{
    "Maximum Theoretical R-value": 85.9,
    "Estimated MOS Score": "4.23",
    "Maximum Bidirectional Voice Calls(calls)": 171.09
}
```

Basic CICD AP Testing with LANforge

Goal: Set up Basic CICD a LANforge system, Regression Automation and Reporting with data from previous runs.

The LANforge CICD framework provides an ability to execute a suite of tests and report results.



1. **The following steps are discussed**
   A. **Set Up CICD Controller and Environment**
   B. **Set Up The JSON Configuration Files**
   C. **Test Execution**
   D. **Test Results**

2. **Set Up CICD Controller and Environment**
   A. clone lanforge-scripts from `https://github.com/greearb/lanforge-scripts`
   B. run `/lanforge-scripts/py-scripts/update_dependencies.py` to install python packages for generating output
   C. Install web server:
      The web server is to allow for viewing of results from User Terminals
      The CICD - Controller is not dependent on a web server, results may be viewed locally on CICD - Controller

A. **LANforge** LANforge installation using kinstall.pl installs a web server on LANforge
LANforge installation installs an httpd server, LANforge may be used for storing and displaying results.
For the following example a separate LANforge system (Fedora) was used as the CICD - Controller and httpd web server.

B. **Fedora** install httpd and configure server
```
$ sudo dnf install httpd
```

C. **Ubuntu** install apache2 and configure server
```
$ sudo apt install apache2
```

D. Install mail service for email of links to results
For the example below Linux mailx program was used
Installation of mail services is dependent on the environment in which the CICD - Controller is installed.
The CICD - Controller is not dependent on email services

E. Install database sqlite3

A. **Fedora**
```
$ sudo dnf install sqlite3
```

B. **Ubuntu**
```
$ sudo apt-get update
$ sudo apt-get install sqlite3
```

F. Create a html-reports directory. On lanforge /home/lanforge/html-reports

G. Determine sqlite3 database name and location, sqlite3 db will be created. ./tools/qa_sqlite3.db

3. **Set Up The JSON Configuration Files**

A. There are three JSON configuration input files described below. For all the JSON configuration files the CAPITALIZED parameters allow for a value to be entered into one location and used in multiple areas of the CICD framework. For example in ssid_indx=1 the SSID_USED is set to asus11ax-5. For the test suite below the SSID_USED may be entered instead of asusu11ax-5, thus if the SSID changes, the SSID will need to be modified in ct_AX88U_dut, the ct_tests.json will remain untouched. This reduces the need to modify the ct_test.json for SSID changes that would affect multiple tests

A. **--json_rig test_rig.json** this JSON file describes LANforge test rig, Example ct_test_rig.json
The test_rig.json describes the LANforge system and test parameters for the CICD - Controller

B. **--json_dut ct_AX88U_dut.json** this JSON file describes the AP, Example ct_AX88U_dut.json
the ct_AX88U_dut.json describes the device under test parameters, DUT_SET_NAME: DUT_NAME ASUSSRT-AX88U for example is used by Chamberview Tests

C. **--json_test ct_tests.json** this JSON file describes the tests, Example ct_tests.json
The tests may use the CAPITALIZED variables or may be entered with the command line arguments as they would be entered on the command line.
The tests are not limited to only python tests

B. **test_rig.json**
```
{
    "test_rig":{
        "Notes":[
            "This JSON file describes LANforge system and test run configuration"
        ]
    },
    "test_rig_parameters":{
        "TEST_BED": "CT-TEST-001",
        "TEST_RIG": "CT-TEST-001",
        "DATABASE_SQLITE": "./tools/qa_sqlite3.db",
        "LF_MGR_IP": "192.168.100.116",
        "LF_MGR_PORT": "8080",
        "LF_MGR_USER": "lanforge",
        "LF_MGR_PASS": "lanforge",
        "UPSTREAM_PORT":"1.1.eth2",
        "TEST_TIMEOUT": 600,
        "EMAIL_LIST_PRODUCTION": "support@candelatech.com",
        "EMAIL_LIST_TEST": "support@candelatech.com",
        "EMAIL_TITLE_TXT": "Lanforge QA Testing",
        "EMAIL_TXT": "Lanforge QA Testing"
    }
}
```

C. **ct_AX88U_dut.json**
```
{
    "ct_AX88U_dut":{
        "Notes":[
            "The device undertest configuration is contained in this file"
        ]
    },
    "test_dut":{
        "DUT_SET_NAME": "DUT_NAME ASUSRT-AX88U",
        "USE_DUT_NAME": "ASUSRT-AX88U",
        "wireless_network_dict":{
            "ssid_idx=0":{"ssid_idx":"0","SSID_USED":"asus11ax-2","SSID_PW_USED":"hello123","BSSID":"3c:7c:3f:55:4d:60"
            "ssid_idx=1":{"ssid_idx":"1","SSID_USED":"asus11ax-5","SSID_PW_USED":"hello123","BSSID":"3c:7c:3f:55:4d:64"
        }
    }
}
```

D. **ct_tests.json**

```
{
    "ct_tests_001":{
        "Notes":[
            "This JSON file describes tests to be run by LANforge system"
        ]
    },
    "test_suites":{
        "suite_wc":{
            "create_chamberview_dut_wc":{
                "enabled":"TRUE",
                "load_db":"skip",
                "command":"create_chamberview_dut.py",
                "args":"",
                "args_list":[
                    " --lfmgr LF_MGR_IP --port LF_MGR_PORT --dut_name DUT_NAME",
                    " --ssid 'ssid_idx=0 ssid=SSID_USED security=SECURITY_USED password=SSID_PW_USED bssid=BSSID'",
                    " --ssid 'ssid_idx=1 ssid=SSID_USED security=SECURITY_USED password=SSID_PW_USED bssid=BSSID'",
                    " --sw_version DUT_SW --hw_version DUT_HW --serial_num DUT_SERIAL --model_num DUT_NAME"
                ]
            },
            "create_chamberview_wc":{
                "enabled":"TRUE",
                "load_db":"skip",
                "command":"create_chamberview.py",
                "args":"",
                "args_list":[
                    " --lfmgr LF_MGR_IP --port LF_MGR_PORT --delete_scenario",
                    " --create_scenario scenario_wpa2_wc",
                    " --raw_line \"profile_link 1.1 STA-AC 19 'DUT: DUT_NAME Radio-1' NA wiphy7,AUTO -1 NA\" ",
                    " --raw_line \"profile_link 1.1 upstream-dhcp 1 NA NA UPSTREAM_PORT,AUTO -1 NA\""
                ]
            },
            "wifi_capacity":{
                "enabled":"TRUE",
                "timeout":"600",
                "iterations":"1",
                "load_db":"skip",
                "command":"lf_wifi_capacity_test.py",
                "args":"",
                "args_list":[
                    " --mgr LF_MGR_IP --port LF_MGR_PORT --lf_user LF_MGR_USER --lf_password LF_MGR_PASS --instance_nam
                    " --upstream UPSTREAM_PORT --batch_size 1,10,19 --loop_iter 1 --protocol UDP-IPv4 --duration 6000",
                    " --pull_report --local_lf_report_dir REPORT_PATH --test_tag 'wpa2_wc'",
                    " --test_rig TEST_RIG",
                    " --set DUT_SET_NAME"
                ]
            },
            "lf_qa":{
                "enabled":"TRUE",
                "timeout":"600",
                "load_db":"skip",
                "command":"./tools/lf_qa.py",
                "args":"",
                "args_list":[
                    " --path REPORT_PATH --store --png --database DATABASE_SQLITE"
                ]
            }
        }
    }
}
```

E. **sample command with above data:**

```
./lf_check.py  --json_rig ct_test_rig.json \
       --json_dut ct_AX88U_dut.json \
       --json_test ct_tests.json \
       --suite "suite_wc" \
       --path '/home/lanforge/html-reports/ct_results_directory'
```

4. **Set Up The JSON Configuration Files**

   A. The lf_check.py is run form the lanforge-scripts/py-scripts/tools directory

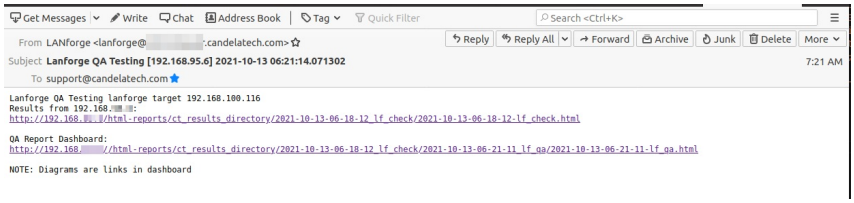
   B. lf_check.py uses three JSON files as input:
   For Example:
   ct_test_rig.json - describes the LANforge test rig configuration
   ct_AX88U_dut.json - describes the device under test
   ct_tests.json -describe the tests to be run.

5. **lf_check.py execution, simple command example**

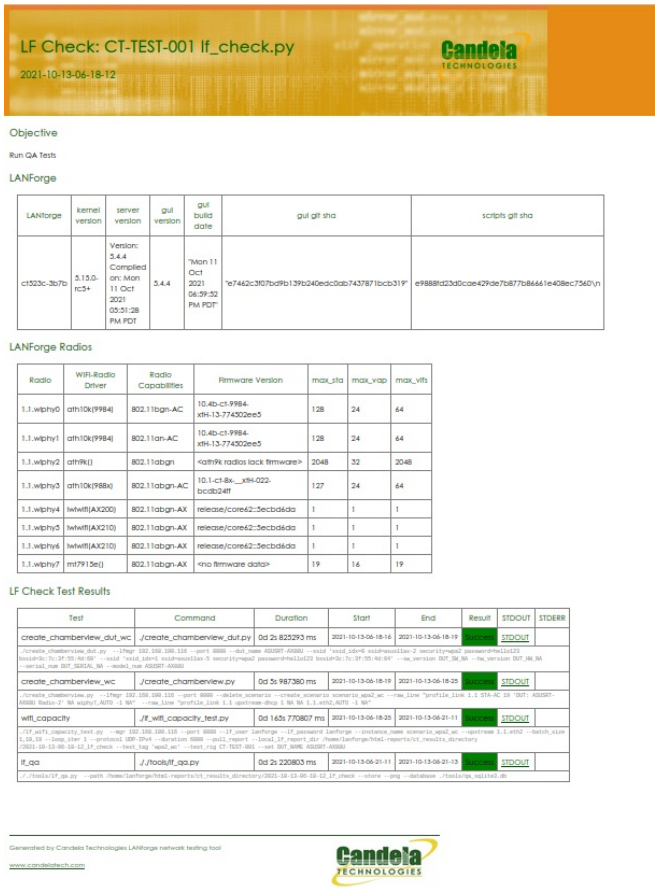
   A.

```
./lf_check.py  --json_rig ct_test_rig.json \
       --json_dut ct_AX88U_dut.json \
       --json_test ct_tests.json \
       --suite "suite_wc" \
       --path '/home/lanforge/html-reports/ct_results_directory'
```

6. **Sample email sent on run**



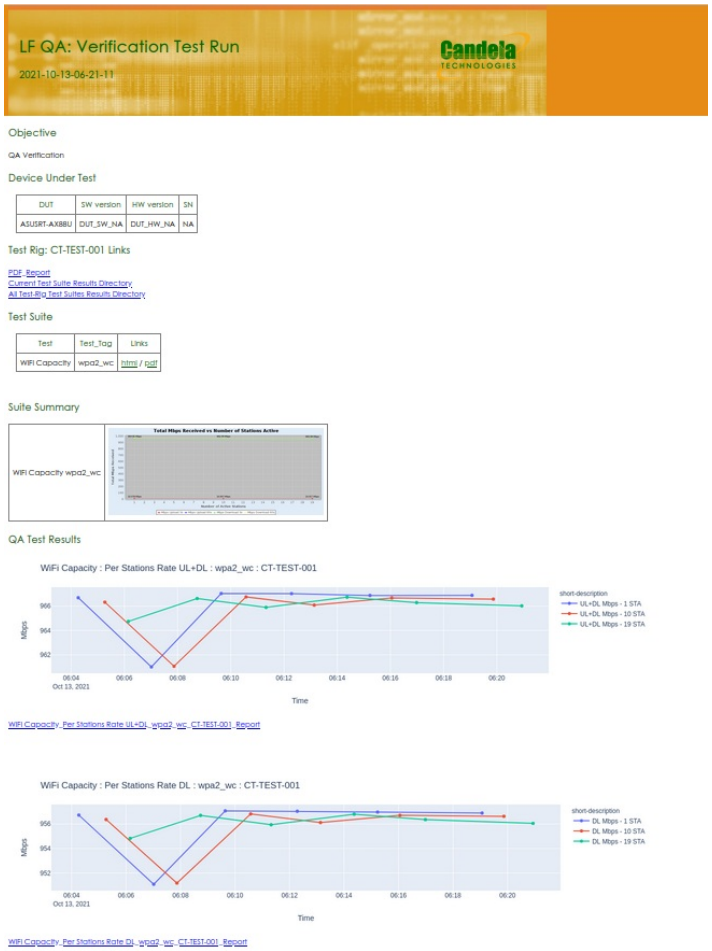
7. **lf_check.py:** sample lf_check.py Report



8. **lf_qa.py**
   lf_qa.py: process kpi.csv, produces html/pdf results, produces plotly png and interactive graphs from test run kpi
   sample command:

```
./lf_qa.py  --path  /home/lanforge/html-reports/ct_results_directory/(results dir of lf_check.py)\
   --store  \
   --png \
   --database ./tools/qa_aqlite3.db
```

9. **lf_qa.py:** sample lf_qa.py Report



10. **Sample lf_heck.py Output** example lf_check Report
11. **Test Control Inputs in Test Suite JSON**
    A. `"enabled":"TRUE"`
    Allows for individual test enable and disable of the test.
    B. `"load_db":"CUSTOM_DATABASE"`
    Allows for loading a LANforge database prior to the test run.
    C. `"timeout":"300"`
    Allows for test to have individual timeout other then default.
    D. `"iterations":"2"`
    Allows for test to run multiple iterations.

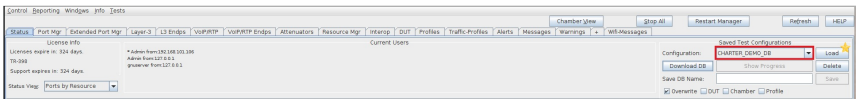
Mobile STA Mesh Test - TPUT/ROAM

Goal: Perform a Mobile Station Mest Test

Demonstrate that a station with the Interop app install can roam in the test environment. Requires LANforge 5.4.6.

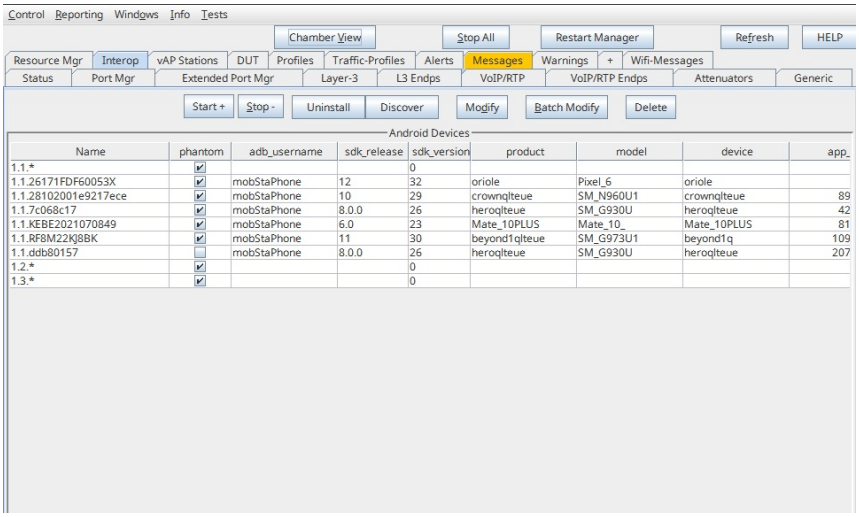
(February 10, 2023) Performed on: https://candelatech.atlassian.net/l/cp/f5CSRysM DUT: Samsung Galaxy S7

Setting Up for the Test



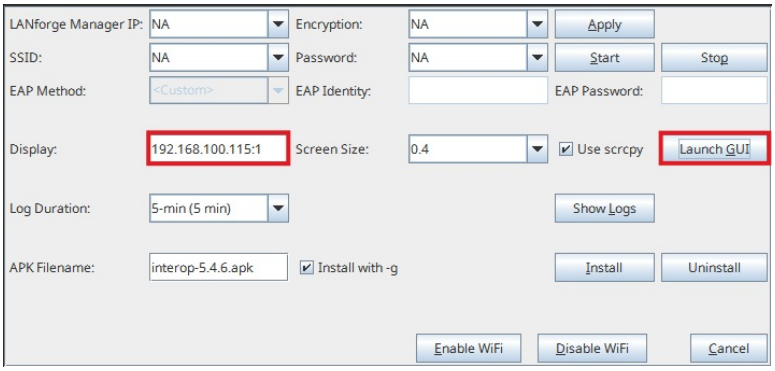
The Charter-Demo database was loaded, since this is the most recently saved stable state for the test.

First, ensure the DUT is on. *NOTE: This DUT has already been configured with the Interop app installed; as well as username/Mgmt IP set, and automatic WiFi connectivity to the test's generated SSIDs.



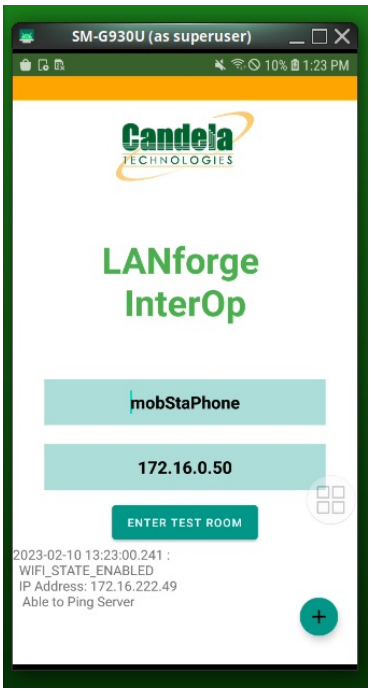
My device, "ddb80157" is on and recognized by forge. If yours isn't recognized, ensure that the device is on and USB debugging is enabled. Checking ADB connectivity can be troubleshooted from the command line on your LANforge machine.

You will want to view the DUT's screen remotely in order to monitor traffic stats. To do this, select 'Batch Modify' from the Interop tab.

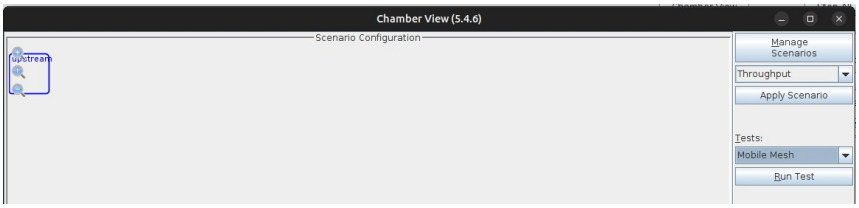


Fill in the IP of the display that you want the screen to appear on. This is likely the IP of your resource #1 LANforge machine.

In a few seconds, a window mirroring your device's screen should appear. Open the Interop app.



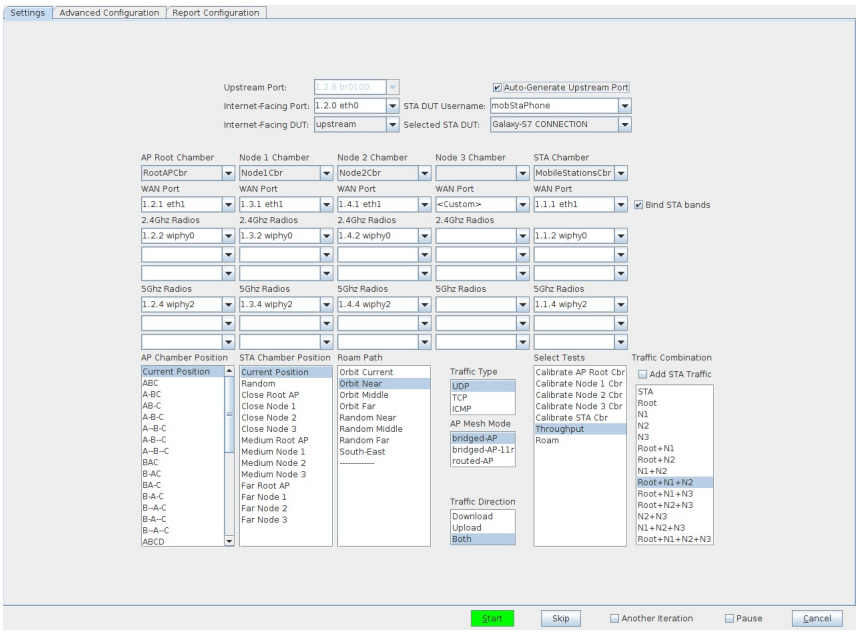
Alright, the DUT is ready for testing. Open Chamber View and select the Mobile Mesh test from the 'Tests' dropdown. Then 'Run Test'.

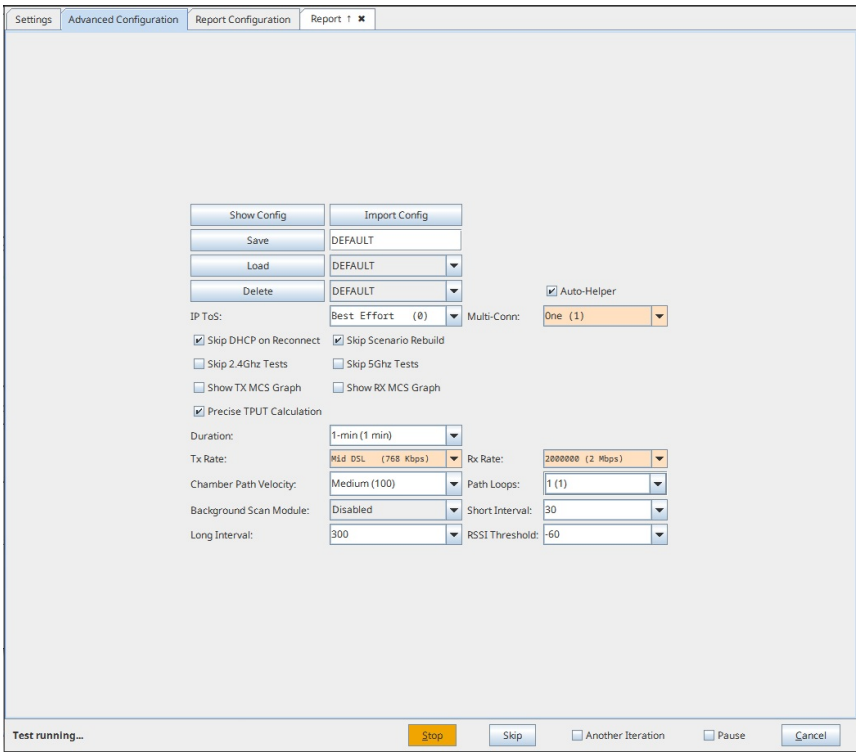


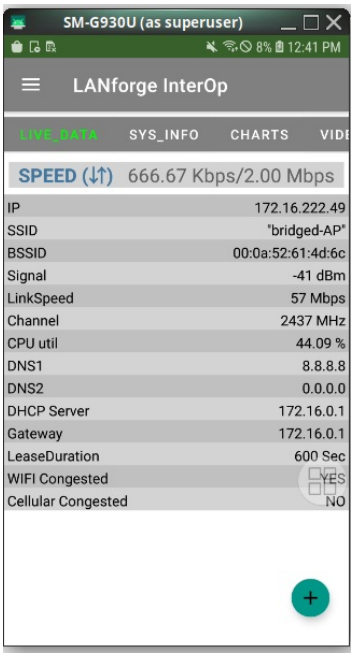
Below are some example test configurations, along with their generated reports.

Throughput Example:

(Bridged Scenario) (1 - 2.4GHz, 1 - 5GHz vAP per chamber) (UDP UL/DL) (Throughput)

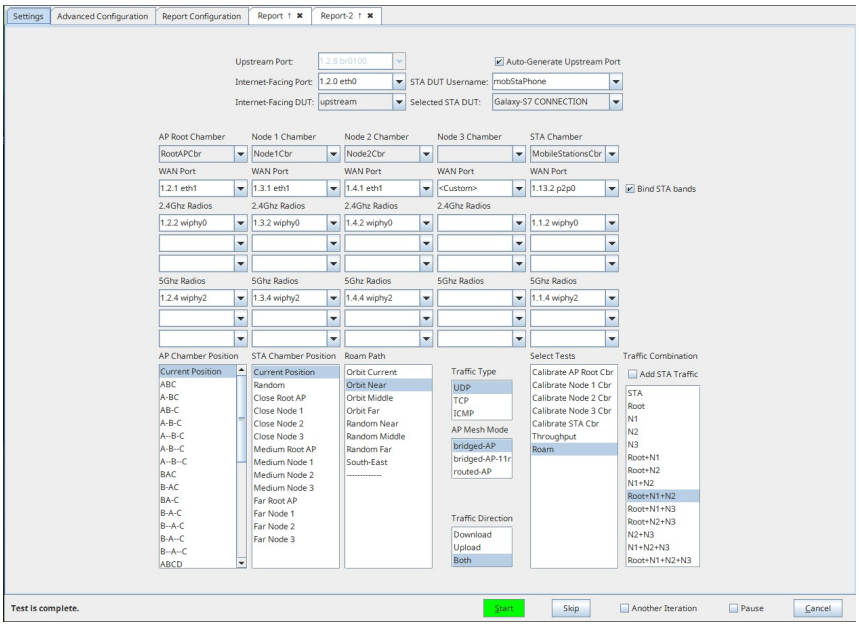


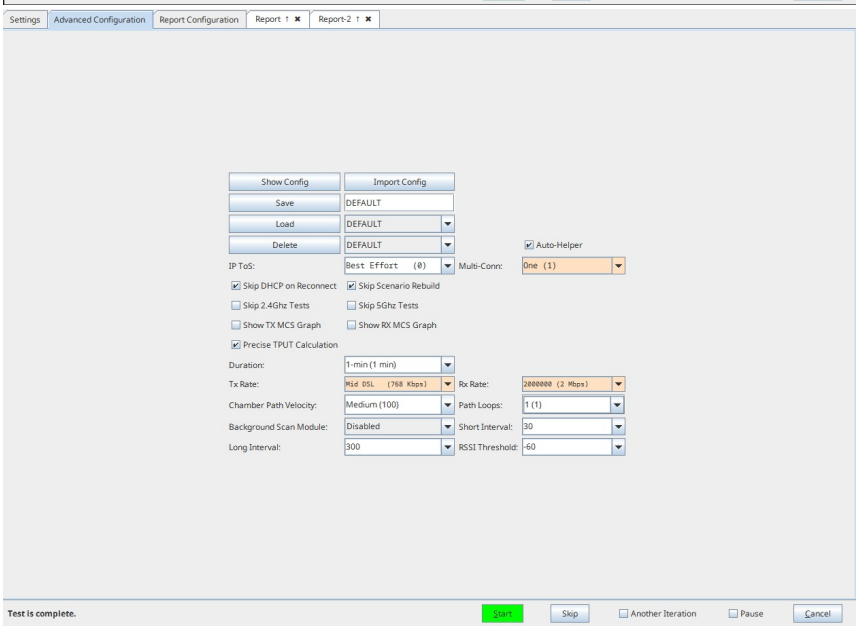


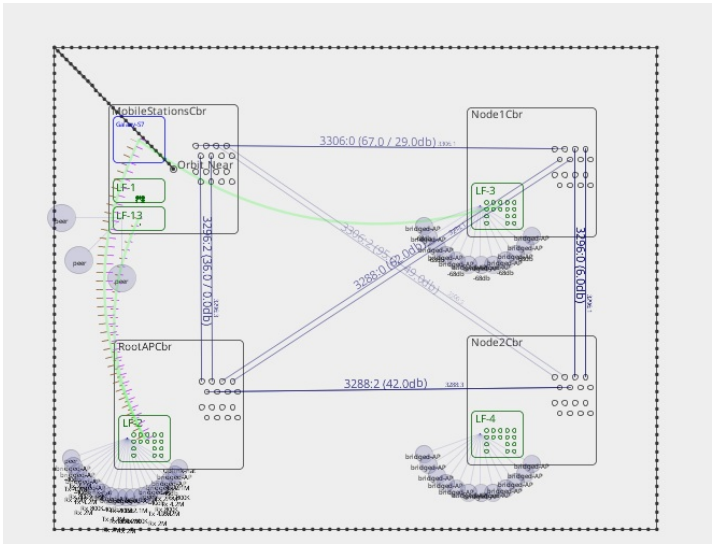


Roam Example:

(Bridged Scenario) (1 - 2.4GHz, 1 - 5GHz vAP per chamber) (UDP UL/DL) (Roam)


Settings
Advanced Configuration
Report Configuration
Report ↑
Report-2 ↑
Upstream Port:
1.2.8 br0100
Auto-Generate Upstream Port
Internet-Facing Port:
1.2.0 eth0
STA DUT Username:
mobStaPhone
Internet-Facing DUT:
upstream
Selected STA DUT:
Galaxy-S7 CONNECTION
AP Root Chamber
Node 1 Chamber
Node 2 Chamber
Node 3 Chamber
STA Chamber
RootAPCbr
Node1Cbr
Node2Cbr
MobileStationsCbr
WAN Port
1.2.1 eth1
1.3.1 eth1
1.4.1 eth1
<Custom>
1.13.2 p2p0
Bind STA bands
2.4Ghz Radios
1.2.2 wiphy0
1.3.2 wiphy0
1.4.2 wiphy0
1.1.2 wiphy0
5Ghz Radios
1.2.4 wiphy2
1.3.4 wiphy2
1.4.4 wiphy2
1.1.4 wiphy2
AP Chamber Position
Current Position
ABC
A-BC
AB-C
A-B-C
A--B-C
A-B--C
A--B--C
BAC
B-AC
BA-C
B-A-C
B--A-C
B-A--C
B--A--C
ABCD
STA Chamber Position
Current Position
Random
Close Root AP
Close Node 1
Close Node 2
Close Node 3
Medium Root AP
Medium Node 1
Medium Node 2
Medium Node 3
Far Root AP
Far Node 1
Far Node 2
Far Node 3
Roam Path
Orbit Current
Orbit Near
Orbit Middle
Orbit Far
Random Near
Random Middle
Random Far
South-East
Traffic Type
UDP
TCP
ICMP
AP Mesh Mode
bridged-AP
bridged-AP-11r
routed-AP
Traffic Direction
Download
Upload
Both
Select Tests
Calibrate AP Root Cbr
Calibrate Node 1 Cbr
Calibrate Node 2 Cbr
Calibrate Node 3 Cbr
Calibrate STA Cbr
Throughput
Roam
Traffic Combination
Add STA Traffic
STA
Root
N1
N2
N3
Root+N1
Root+N2
N1+N2
Root+N1+N2
Root+N1+N3
Root+N2+N3
N2+N3
N1+N2+N3
Root+N1+N2+N3
Test is complete.
Start
Skip
Another Iteration
Pause
Cancel


Settings
Advanced Configuration
Report Configuration
Report ↑
Report-2 ↑
Show Config
Import Config
Save
DEFAULT
Load
DEFAULT
Delete
DEFAULT
Auto-Helper
IP ToS:
Best Effort (0)
Multi-Conn:
One (1)
Skip DHCP on Reconnect
Skip Scenario Rebuild
Skip 2.4Ghz Tests
Skip 5Ghz Tests
Show TX MCS Graph
Show RX MCS Graph
Precise TPUT Calculation
Duration:
1-min (1 min)
Tx Rate:
Mid DSL (768 Kbps)
Rx Rate:
2000000 (2 Mbps)
Chamber Path Velocity:
Medium (100)
Path Loops:
1 (1)
Background Scan Module:
Disabled
Short Interval:
30
Long Interval:
300
RSSI Threshold:
-60
Test is complete.
Start
Skip
Another Iteration
Pause
Cancel


MobileStationsCbr
Node1Cbr
Node2Cbr
RootAPCbr
Orbit Near
LF-1
LF-13
LF-3
LF-2
LF-4
3306:0 (67.0 / 29.0db)
3288:2 (42.0db)
3296:0 (67.0 / 29.0db)
3288:0 (42.0db)
3296:0 (67.0 / 29.0db)
bridged-AP
peer

Test Video Streaming using DASH Server and InterOp APK

Goal: Host DASH server on LANforge and test Video Streaming using InterOp apk.

This cookbook requires LANforge GUI version 5.4.6 and above.

Background

DASH is used for embedding videos in a web page without using third party services such as youtube or vimeo. It is mostly used in OTT platforms.

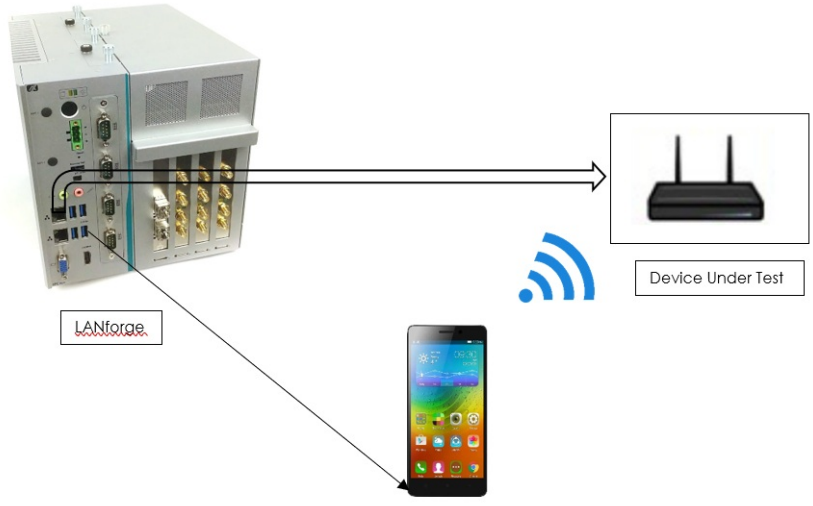
Dynamic Adaptive Streaming over HTTPS (DASH) is a variant of Dynamic Adaptive Streaming (DAS) that uses the HTTPS protocol for delivering video content over the internet. HTTPS is the secure version of HTTP, and it provides encryption and authentication to protect the privacy and integrity of the data being transmitted.

With DASH, the video content is encoded at multiple bitrates and is divided into small segments that can be downloaded and played back in a continuous stream. These segments are delivered over the internet using HTTPS, which provides secure and reliable delivery of the video content.

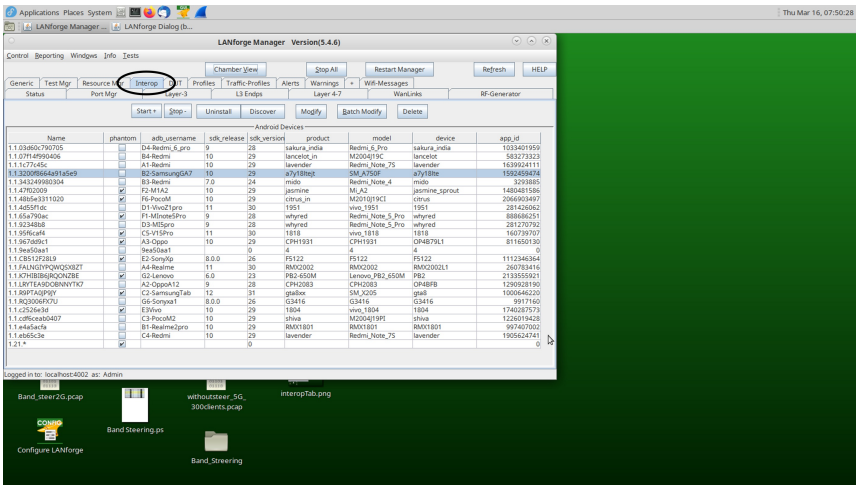
Android App has inbuilt Video Media Player (Exoplayer) which can run various types of Media Types which includes:

- Progressive
- DASH (Dynamic Adaptive Streaming over HTTPS)
- HTTP Live Streaming
- Real Time Streaming Protocol

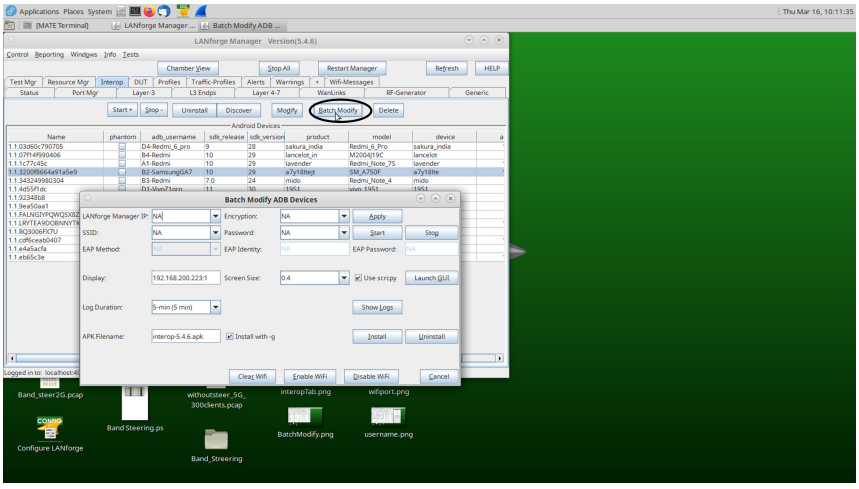
Diagram of Setup



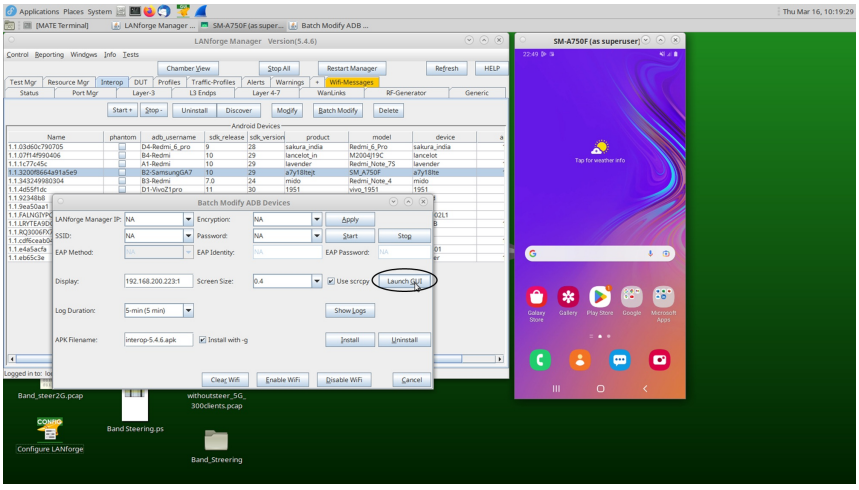
1. Connect Android Phone to LANforge via USB such that the device will appear in Interop Tab.
    - Note: Make sure to enable developer settings in android phones.



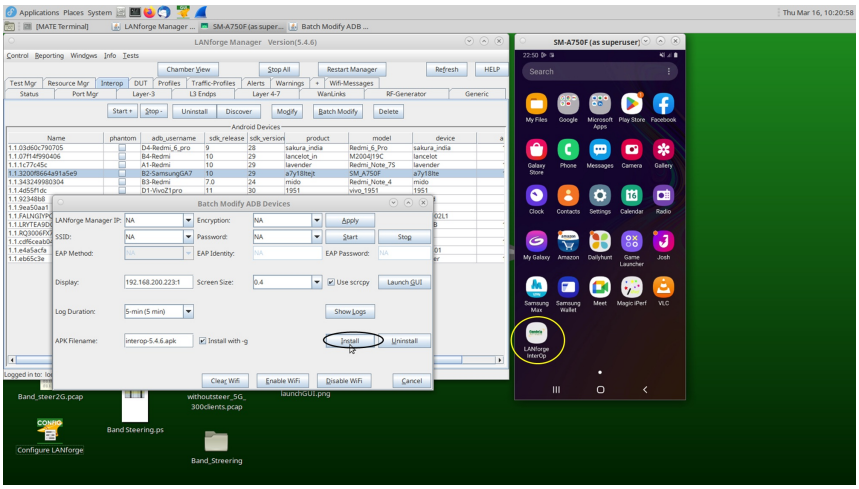
2. Select the device in Interop Tab and click on Batch Modify tab.



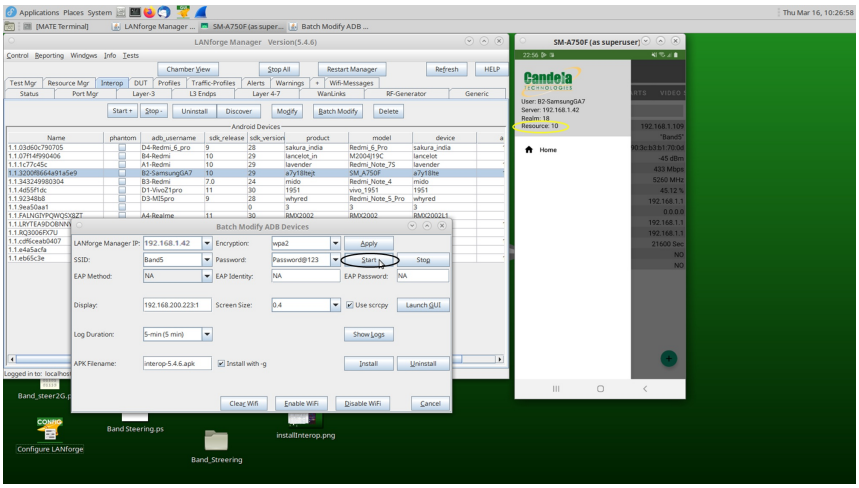
3. Select Launch GUI option to view the mobile screen.



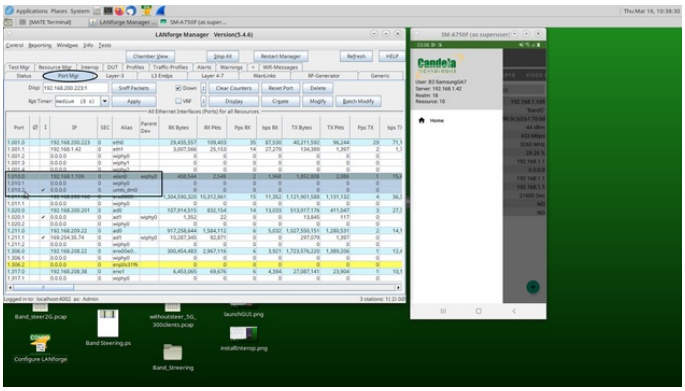
4. Click Install to install Interop apk onto the mobile phone.



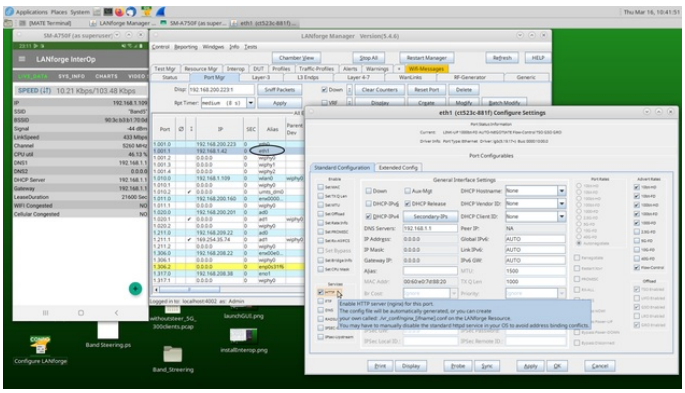
5. Enter LANforge Manager IP, SSID, Encryption and Password. Then, click Apply and start options such that device will be connected to Wi-Fi and enter testroom. Here, mobile got clustered to the LANforge with Resource-Id 10.



6. After entering the Test room, device must appear in Port Manager Tab. That is the Wi-Fi port of Mobile.

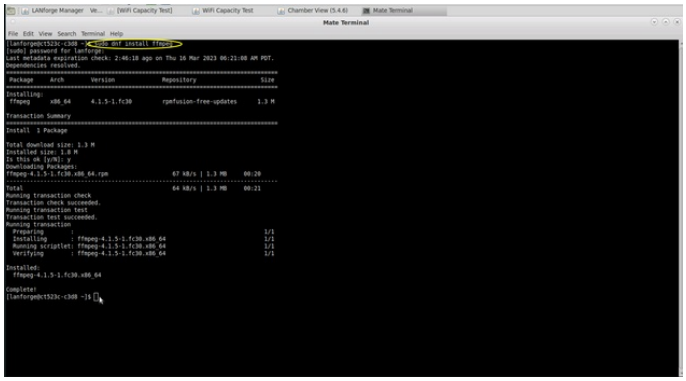


7. To run nginx on eth1 port enable http on eth1 interface by double-clicking on the port and apply the settings.



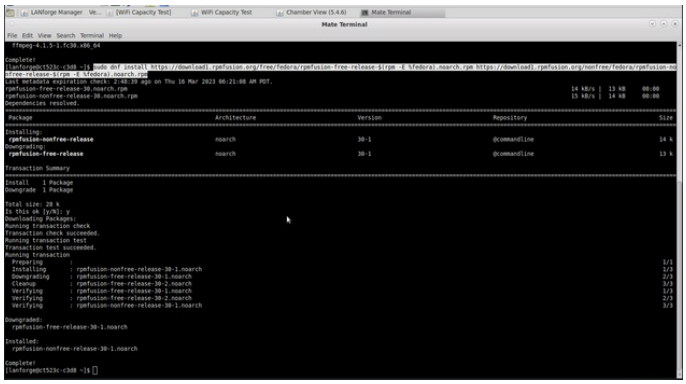
8. To host DASH server on LANforge use the following commands.
    1. `sudo dnf install ffmpeg`

    FFmpeg is a popular multimedia framework that can be used to record, convert, and stream audio and video files.



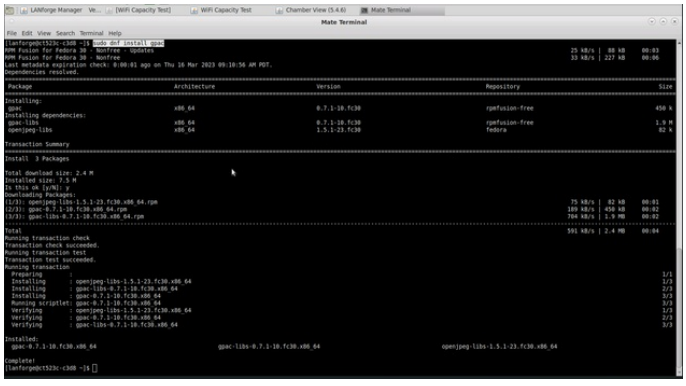
    2. `sudo dnf install https://download1.rpmfusion.org/free/fedora/rpmfusion-free-release-$(rpm -E %fedora).noarch.rpm https://download1.rpmfusion.org/nonfree/fedora/rpmfusion-nonfree-release-$(rpm -E %fedora).noarch.rpm`

    This command installs the RPM Fusion repositories on a Fedora system. RPM Fusion is a repository of software packages that are not included in the default Fedora repositories, such as multimedia codecs and drivers.

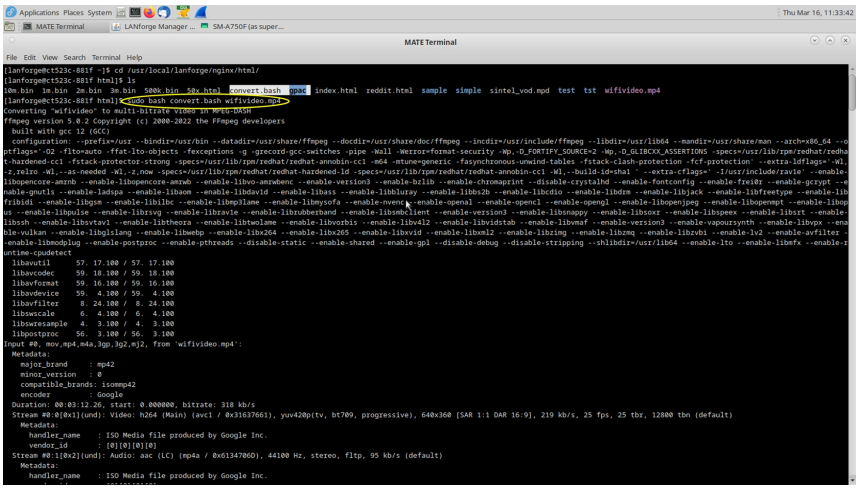


    3. `sudo dnf install gpac`

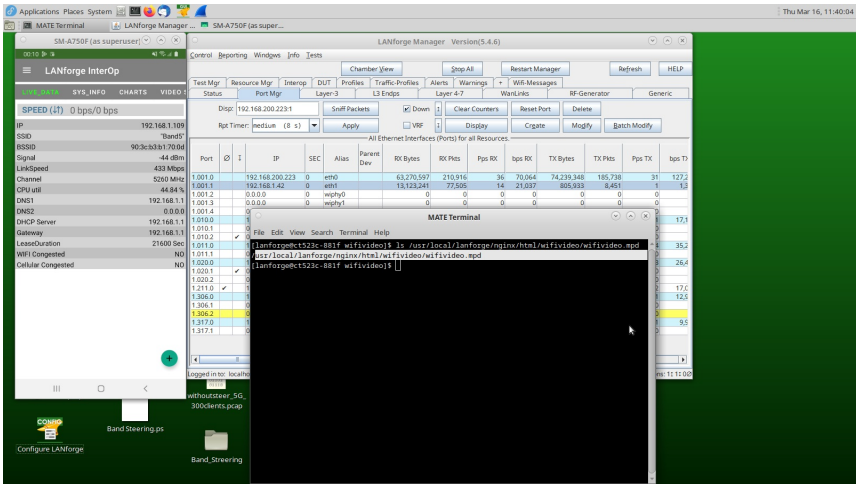
    GPAC is an open-source multimedia framework that can be used to create, package, and stream multimedia content. This command, when run, downloads and installs the GPAC package and its dependencies on your system. After the installation is complete, use the GPAC tools to create and manipulate multimedia content.



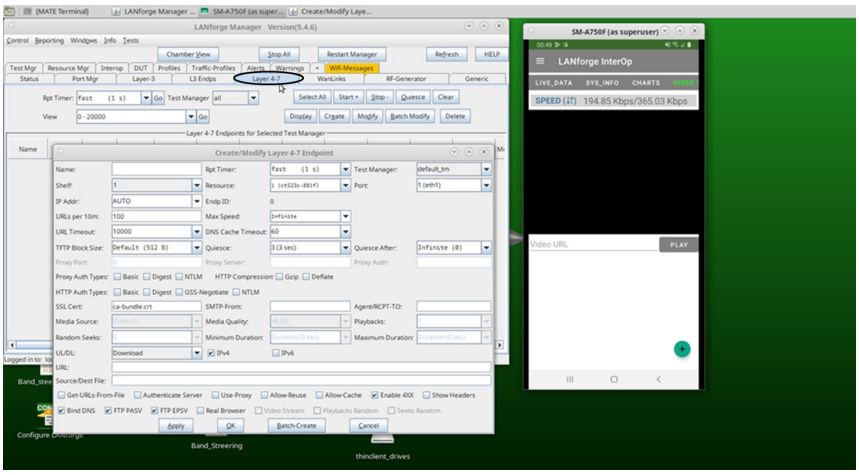
9. Place below mentioned wifivideo.mp4 and convert.bash file in the html directory: **/usr/local/lanforge/nginx/html/**
    1. convert.bash
    2. wifivideo.mp4
10. Run the convert.bash to convert .mp4 to .mpd file using the command:
    `sudo bash convert.bash wifivideo.mp4`



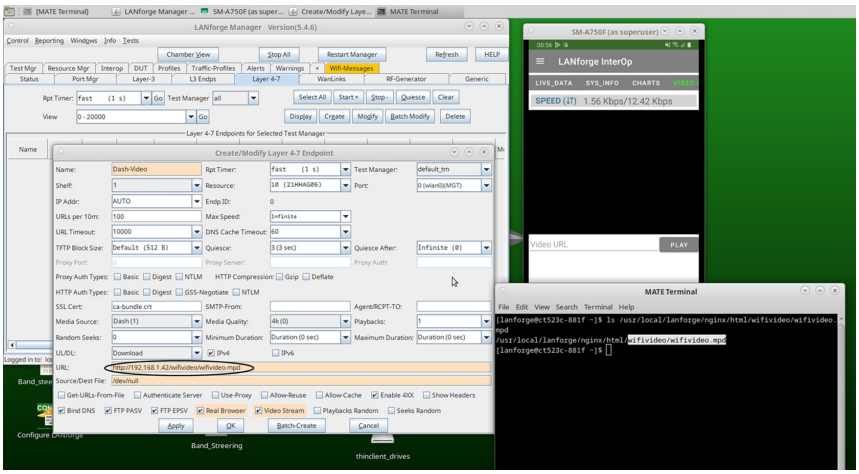
11. A new directory will be created automatically to store the .mpd file. In the below picture, wifivideo.mpd is stored in wifivideo directory.



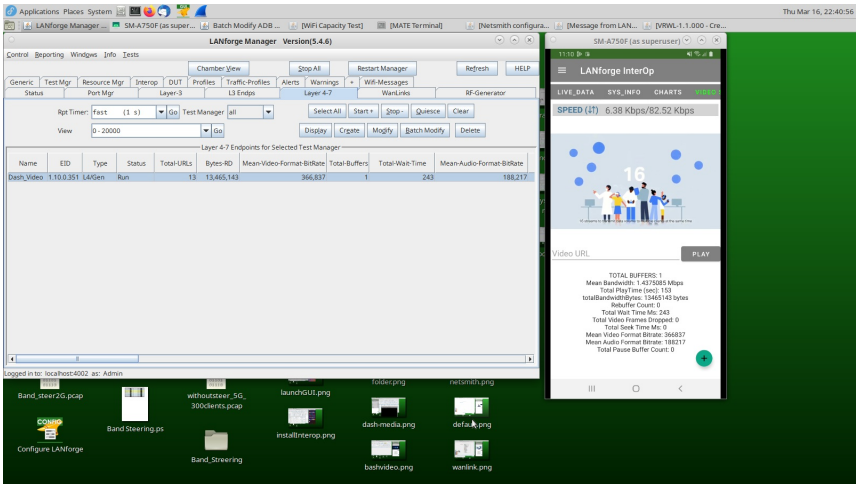
12. Create a L4 Endpoint by clicking on Layer 4-7 tab and opening the Video Streaming tab on app.



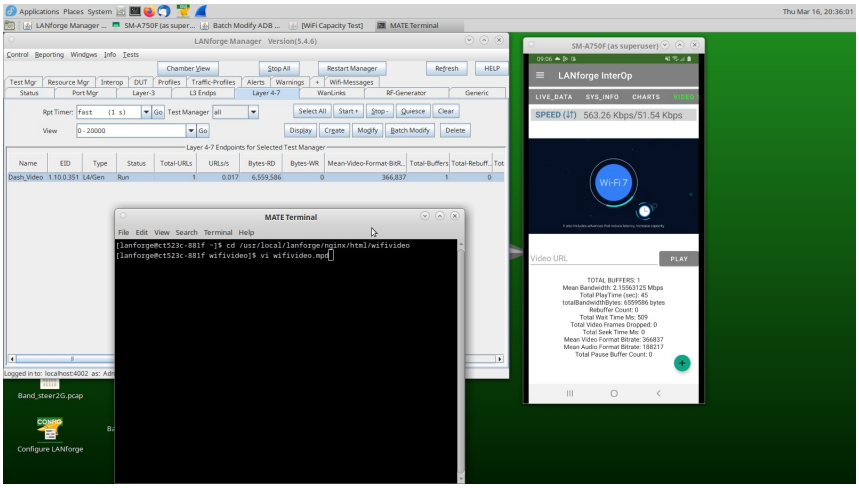
13. Add the URL and Source/Dest File. Select Resource port number of your mobile, enable Real Browser, and Video Stream options. Add the eth1 IP address on which nginx server is enabled and enter the URL.



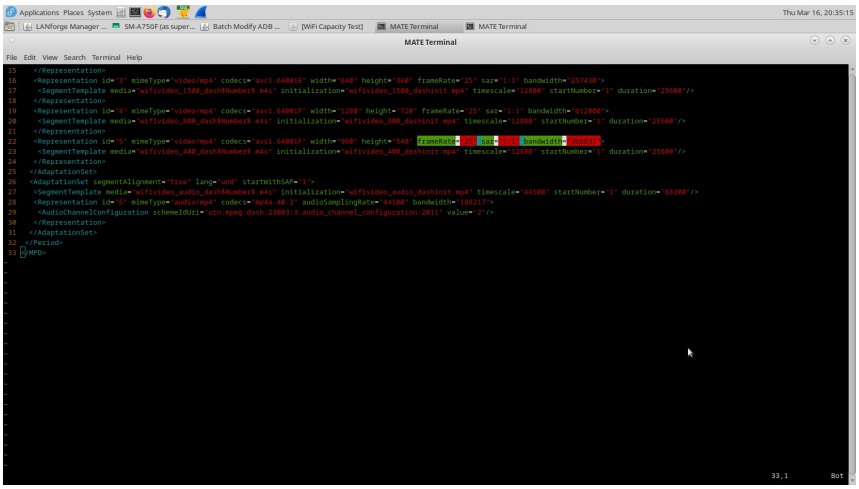
14. Start the endpoint. The video will be play on the phone and the stats will be reported in the Layer-4 tab in the GUI as well as on LANforge Interop app.



The stats can also be viewed in the MPD file by using the command: `vi wifivideo.mpd` in the terminal.

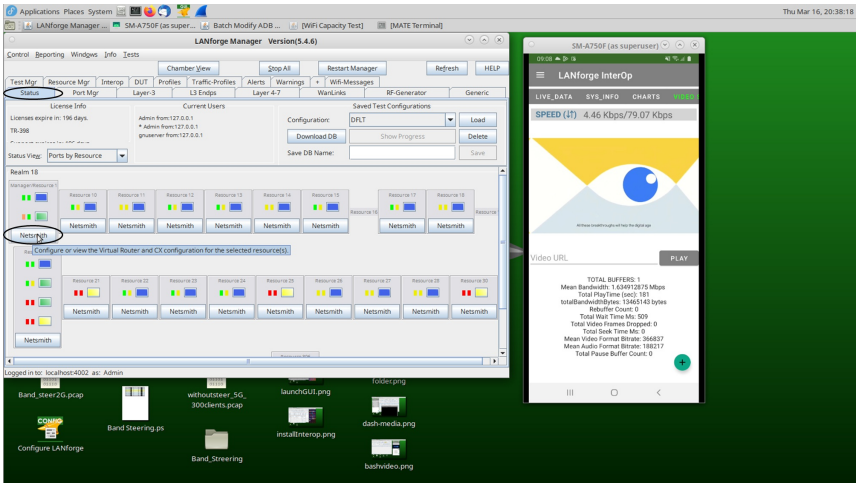


Mpd file maps with bandwidth that is mentioned in LANforge InterOp app. From the Mpd we can observe bandwidth as 366837 which is same as mentioned above in InterOp.

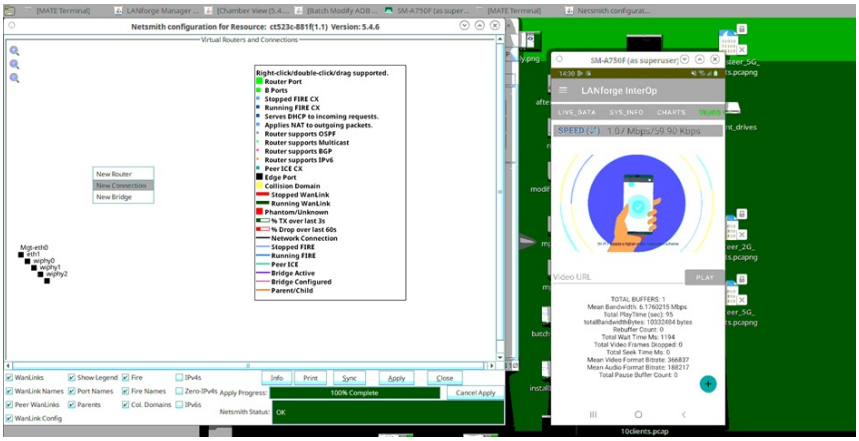


If the network has issues, the bitrate goes down and the same can be reported in LANforge InterOp app. This can be tested using WANlinks in lanforge.

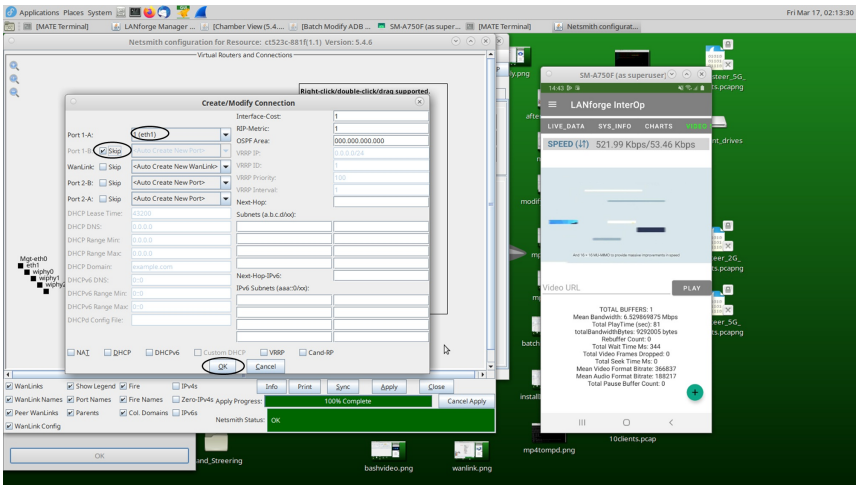
15. To create a WANlink go to the Status tab and select Netsmith.



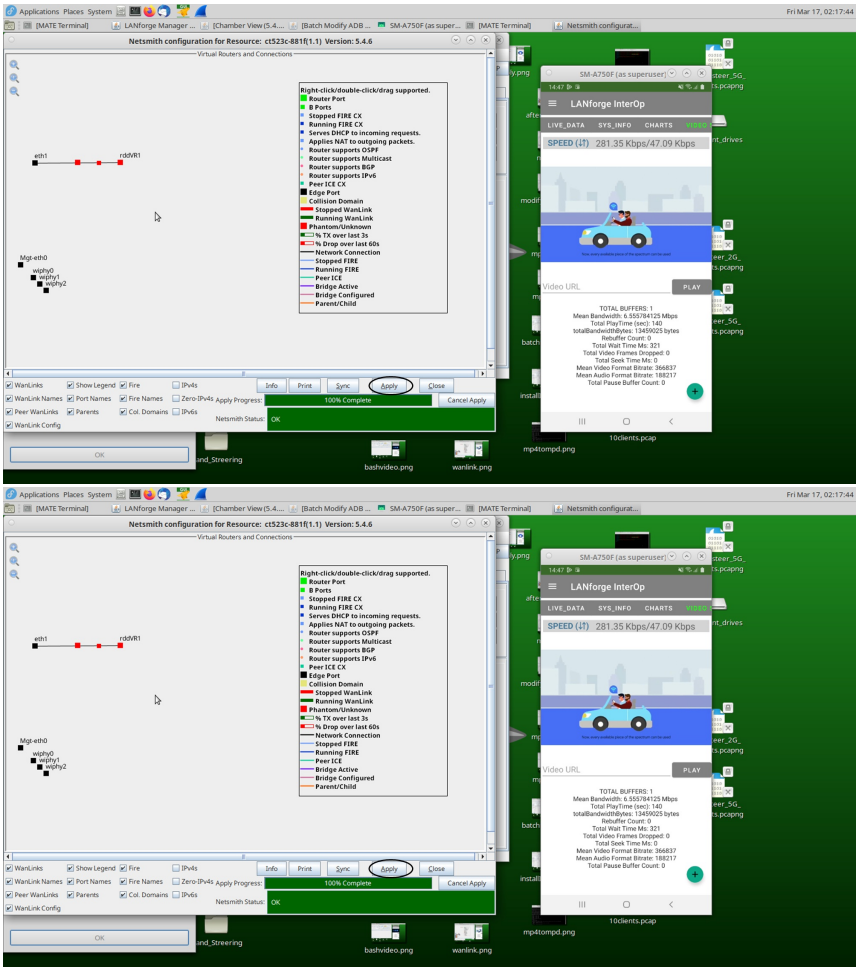
16. Right-click in the Netsmith window and select New Connection.



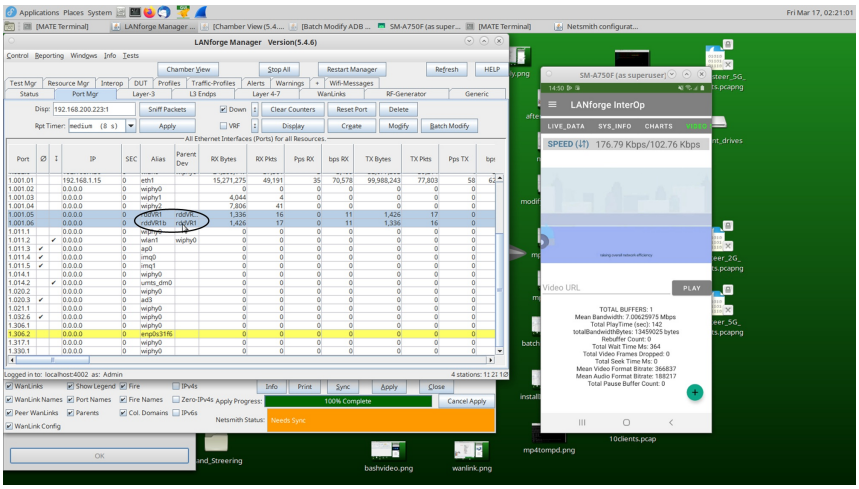
17. Select Port 1-A as the port where the server is running (eth1 in this case) and skip Port 1-B. Click OK.



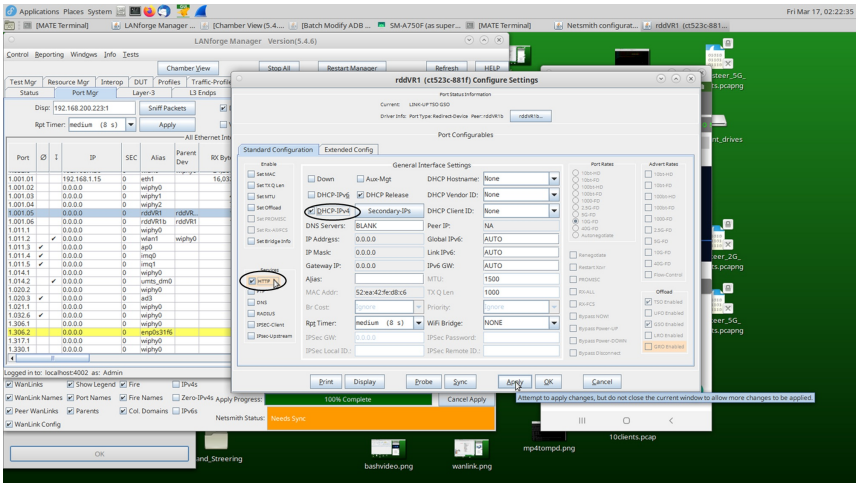
Then click on Apply in the Netsmith window, the WANlink will be created between eth1 and rddvR1.



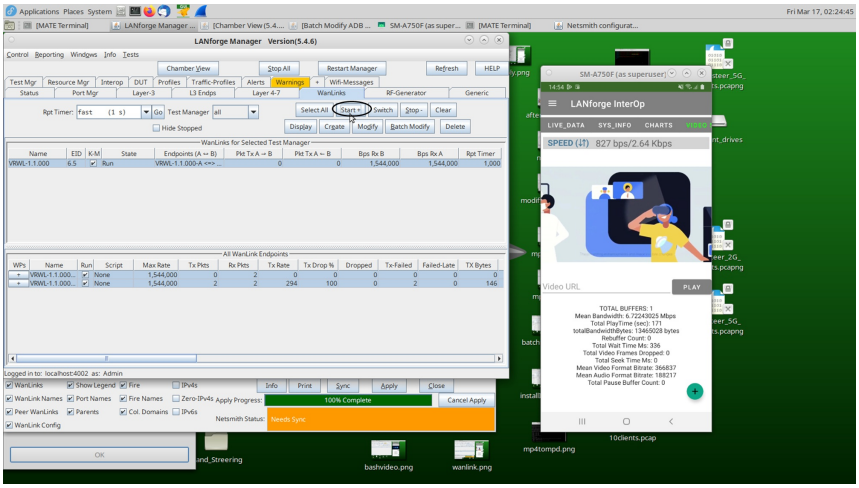
In Port manager this endpoint will also be reflected.



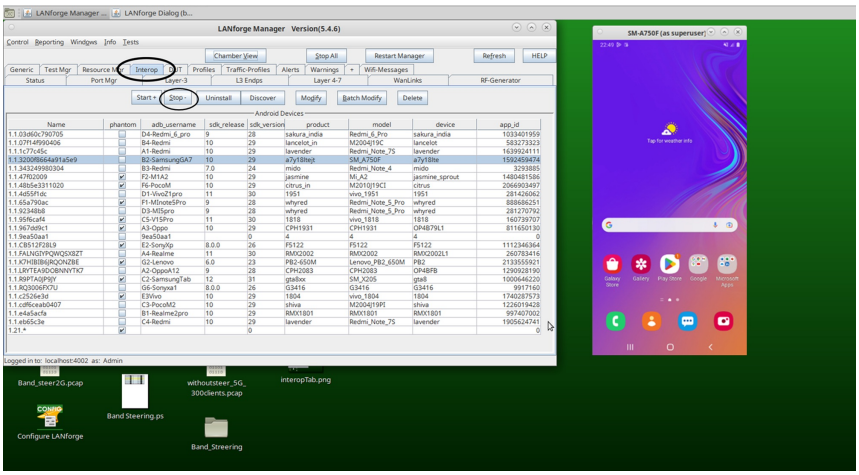
18. Click on the first of the two ports of the RDD pair, which will open a Configuration Window. Click the DHCP-IPv4 and HTTP checkbox.



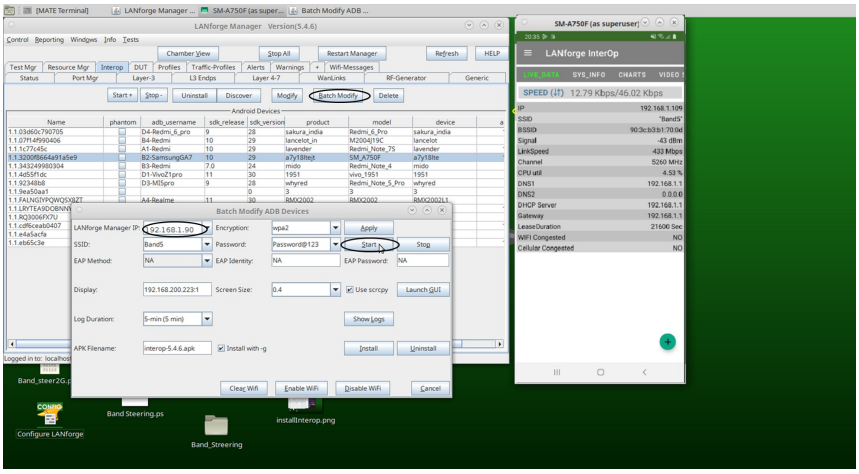
19. Now go to Wanlinks tab and click on start option to run the connection. In the Port Manager the IP address will be allocated to the first (non-b) port of the RDD pair.



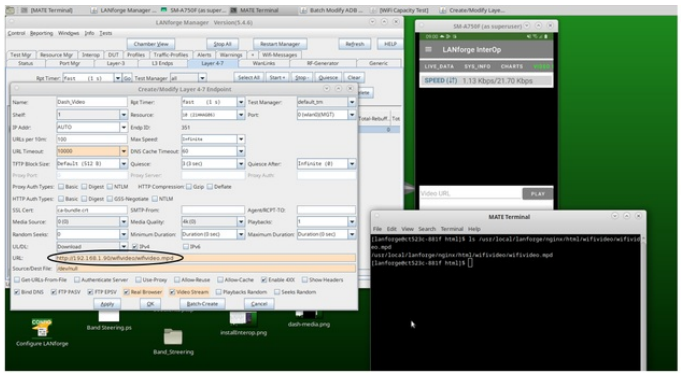
20. Then, Click on Stop to bring the device out of test room.



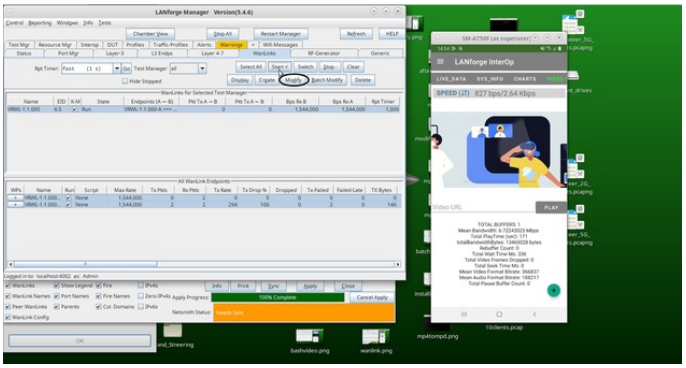
21. Click on the Batch Modify and enter all the details in the screenshot below. LANforge Manager IP must be WAN link IP. Here rddVR1 is allocated with "192.168.1.90" IP address.



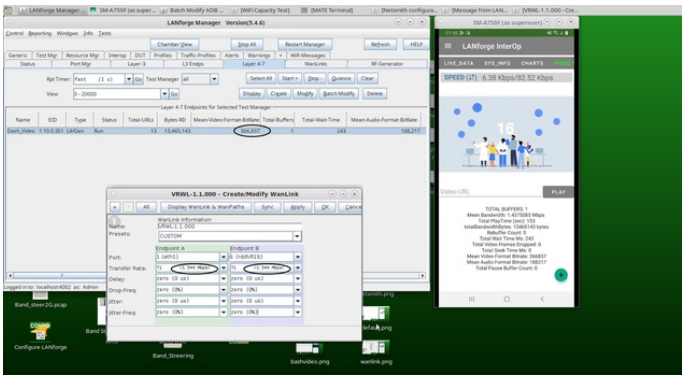
22. Click Layer 4-7 tab, create one cross-connect with the following details, Apply the changes,and play the video. Nginx will run on rddVR1.



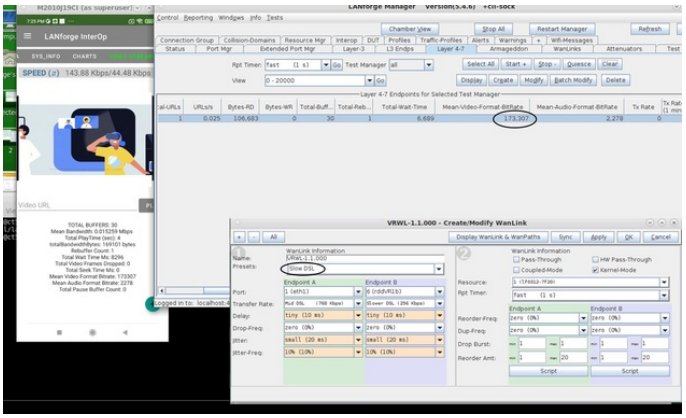
23. Click on WANlinks tab, select the WANlink to modify, and click on Modify button to modify the wanlink.



Here the Wanlink transfer rate is 1.54Mbps and the video bitrate reported is 366837. This can be observed in Layer 4-7 and in InterOp.



24. Modify the WanLink to slow DSL and click on Apply. Video Bit rate is reduced to 173307.



It clearly indicates that the Video Player has adapted the lower quality when network is slow.

We can observe the same in the .mpd file.

LANforge InterOp Setup for Android

Goal: Cable an android mobile phone to a LANforge and have the LANforge recognize the android phone as a resource.

This cookbook requires LANforge GUI version 5.4.6 and above.

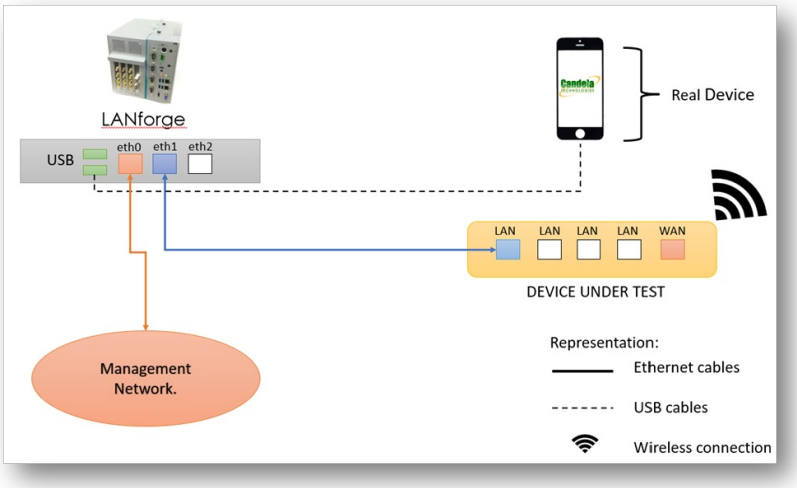
Background

Our LANforge InterOp solution is used to support real clients for testing access points. InterOp gives the ease of handling real clients with complete automation and detailed, customizable test reporting.

We have support for various kinds of real clients:

- Android clients.
- iOS clients.
- Windows Machines.
- Linux Machines.
- MacOS Machines.

Network Topology:



Before getting started:

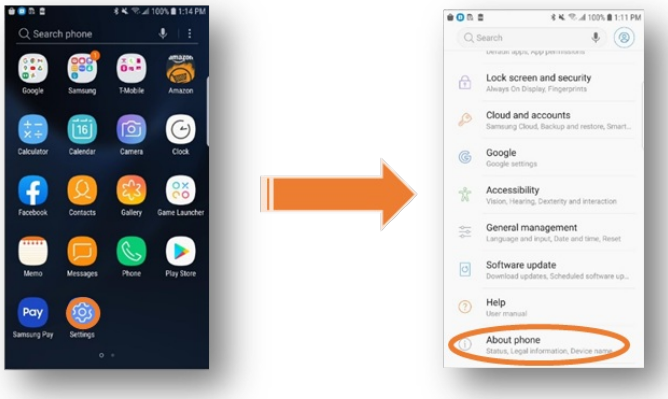
- If your mobile is in developer mode, skip to step-3.
- If you are using XIAOMI mobiles [Redmi, Poco ..etc] for LANforge Interop then please make sure that your mobile has Mi account.

Steps to create Mi account:

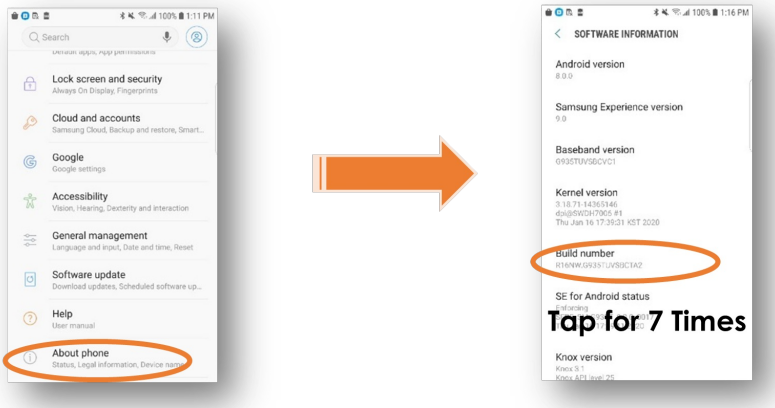
- Insert a sim card in the Mi mobile.
- Create Mi account by providing the credentials.

Steps to setup & connect your android phone to LANforge

1. Open the Settings on the Android phone and click on About Phone.



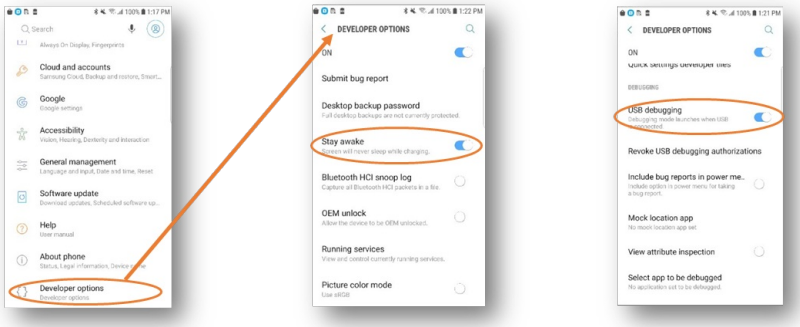
2. In About Phone, click on Build number 7 times to enable developer options.



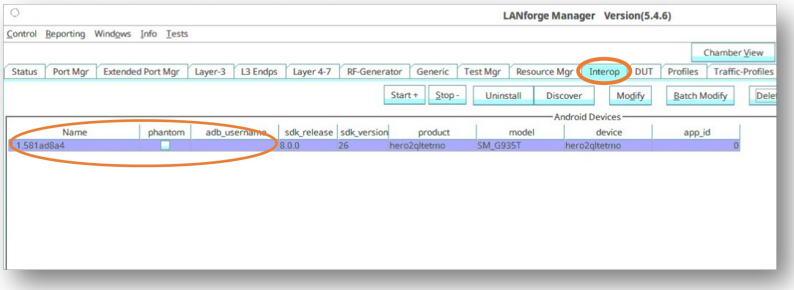
3. Now there will be a new option available in settings tab called "Developer Options". Click on it and enable the following options:

- Stay Awake.

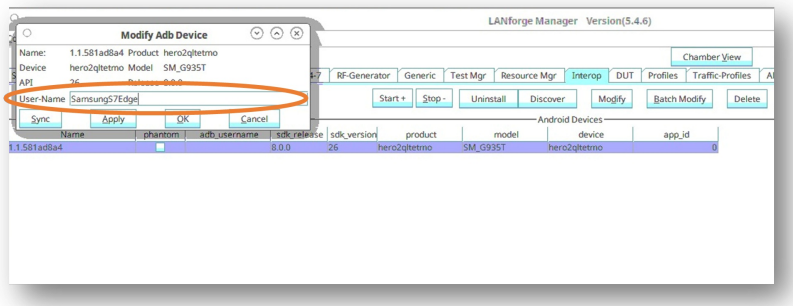
- USB-Debugging.



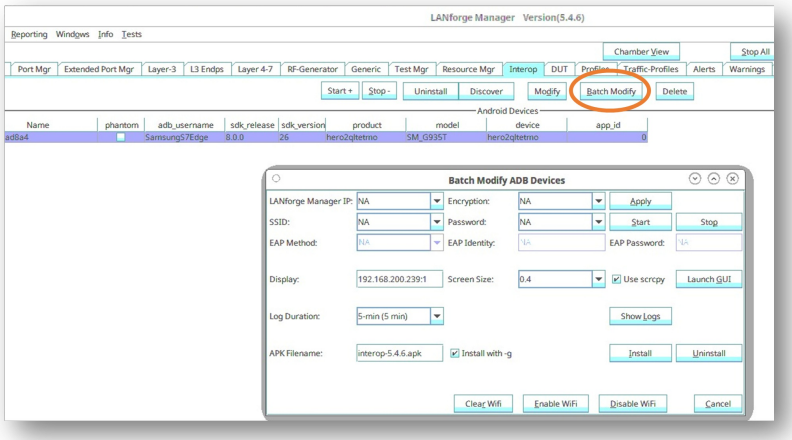
4. Connect the device to the LANforge as per the testbed topology. Plug the phone via the port (usb-c or microusb) at the bottom of the phone to one of the usb-A slots on the LANforge.
5. Open the LANforge GUI window. The Interop tab should appear now. Click on the tab to see your phone detected as a port.



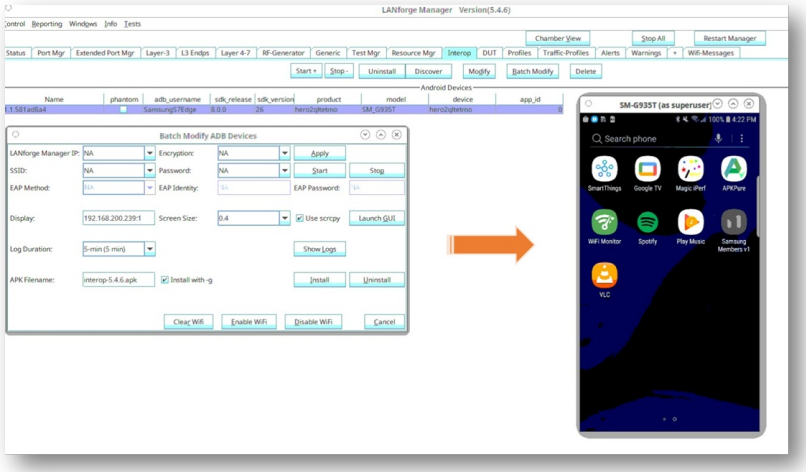
6. in the InterOp tab, double-click on the port to open the Modify ADB Device window. Enter your device user name.



7. Close the Modify ADB Device window. Back in the InterOp tab there is a button called Batch-Modify. Click on Batch-Modify to see options to configure the Wi-Fi profiles of the client.

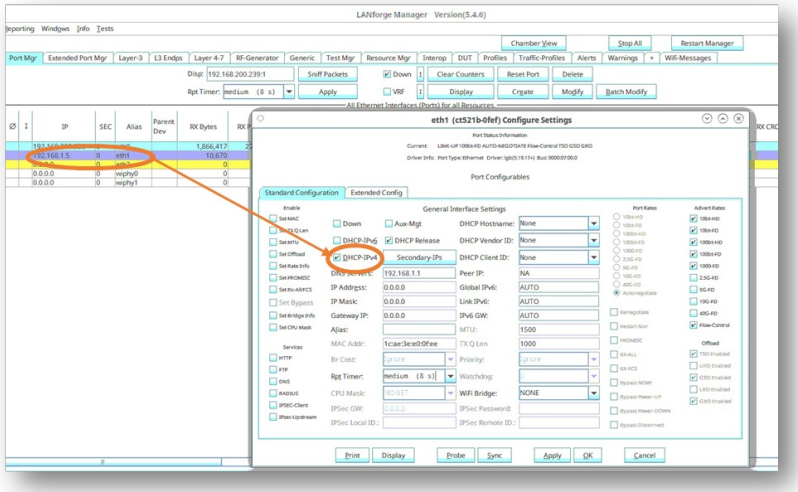


8. In the Batch-Modify window, check the use SCRCPY option and click on Launch GUI. This will open a screen mirror of the mobile in the LANforge GUI and can be used to remotely control the phone if needed.

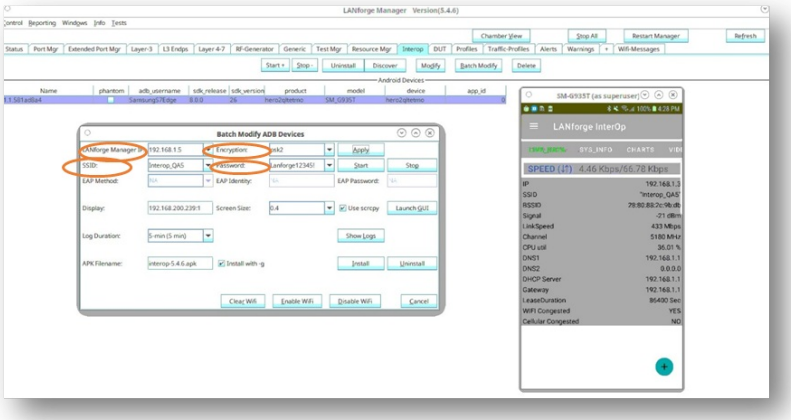


9. Now click on Install in the Batch-Modify window to install the InterOp app (apk).

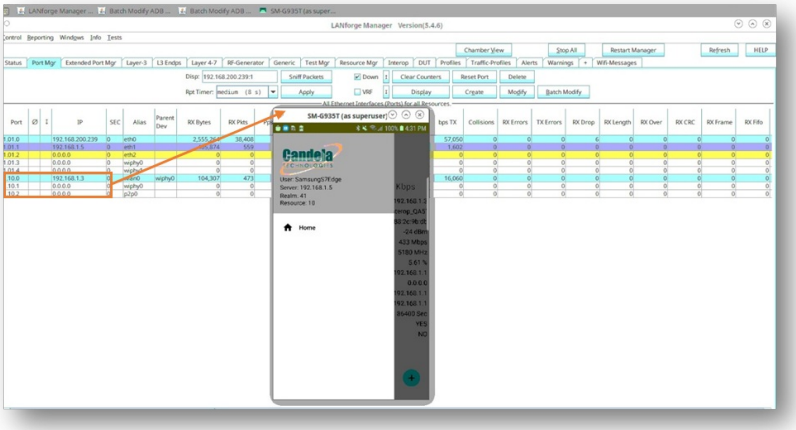
10. Close the Batch-Modify window. Click on the Port Magr tab, double click on eth1, and check the DHCP-IPv4 checkbox or assign a static IP address from the DHCP pool of the AP, which is our Manager IP address.



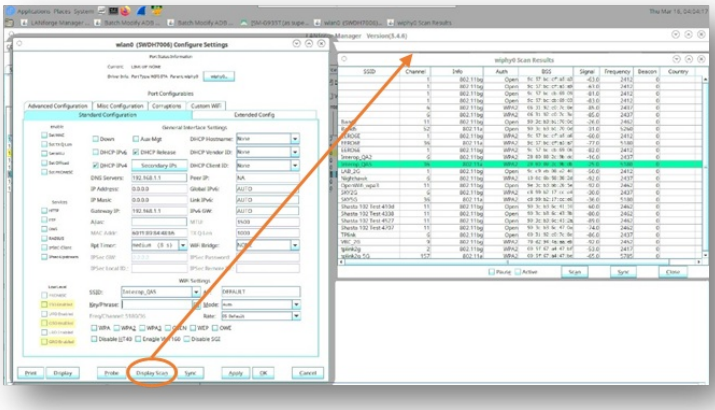
11. Click on the InterOp tab again, double-click on your phone, and fill out the fields of IP address, SSID, encryption modes of the DUT. Click Apply when fields are entered.



12. After clicking Apply, the InterOp app will open in the mobile and the Network stats will be visible. The phone can also be detected in the Resource Mgr and Port Mgr tabs in the GUI now.



13. Below is an example of ports like wiphy0, wlan0, p2p0, which are the ports of the mobile. At this point, if you click on wlan0 port[station] and click on Display Scan, a pop-up in which all the nearby SSID's Broadcasted near the phone are detected.



LANforge Server Installation for Windows Machine

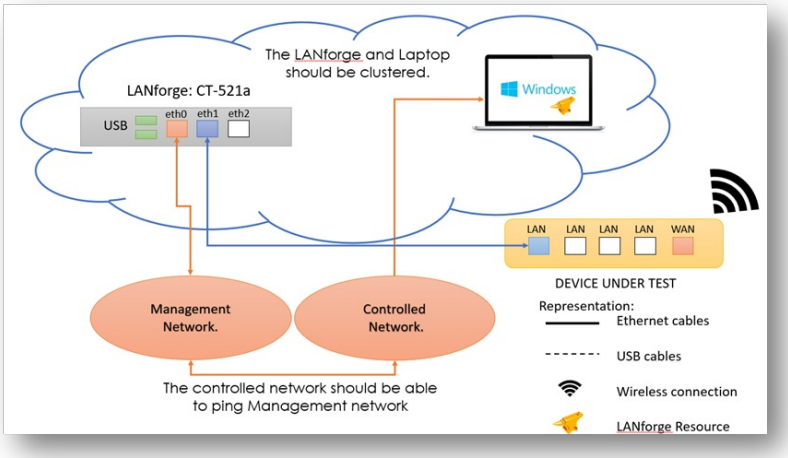
Goal: Install the LANforge Server on the Windows machine, have the LANforge Manager recognize the Windows machine as a resource, and cluster the LANforge Manager with the Windows machine.

The LANforge InterOp solution of Candela technologies is used to support real clients for testing Access Points. InterOp gives the ease of handling real clients with complete automation and populate a detailed reports on the entire duration of the test. This cookbook requires LANforge version 5.4.6 and above.

We have support for various kinds of real clients:

- Android clients.
- iOS clients.
- Windows Machines.
- Linux Machines.
- MacOS Machines.

Network Topology:



Before getting started:

All applications requiring internet for installation should be installed before starting the LANforge server.

Steps to install LANforge Server on MacOS machine

1. Make sure the Windows laptop runs either Windows 10 or 11.

2. Open Windows powershell as administrator and type in the following command, then press enter.

```
wget http://www.candelatech.com/windows_lf_setup.ps1 -o windows_lf_setup.ps1
```

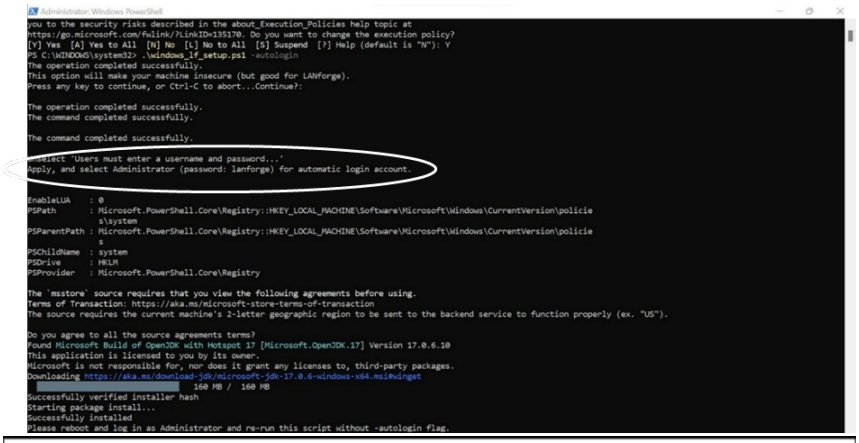
3. Type in the command that allows executing powershell Bypass scripts in powershell, then press enter.

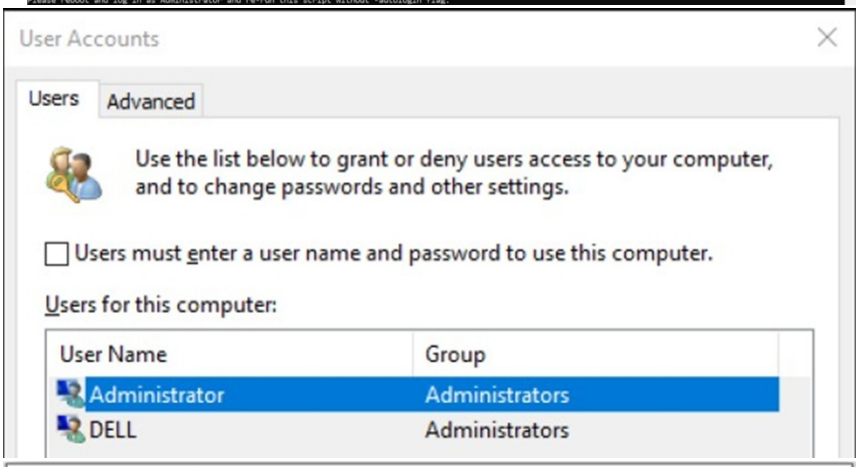
```
Set-ExecutionPolicy Bypass
```

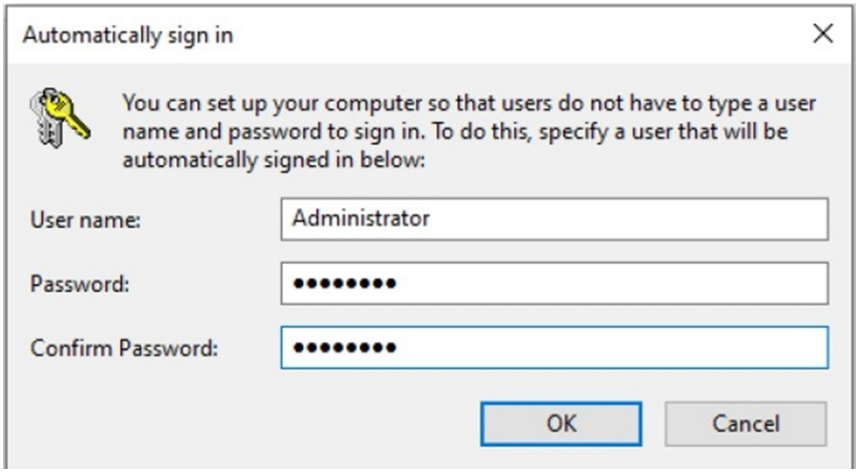
4. Type in the command below to allow auto-login and disable security. Then press enter.

```
.\windows_lf_setup.ps1 -autologin
```

Then, unselect the users for "Users must enter a username and password" to prevent the machine requiring a username or password after reboot.







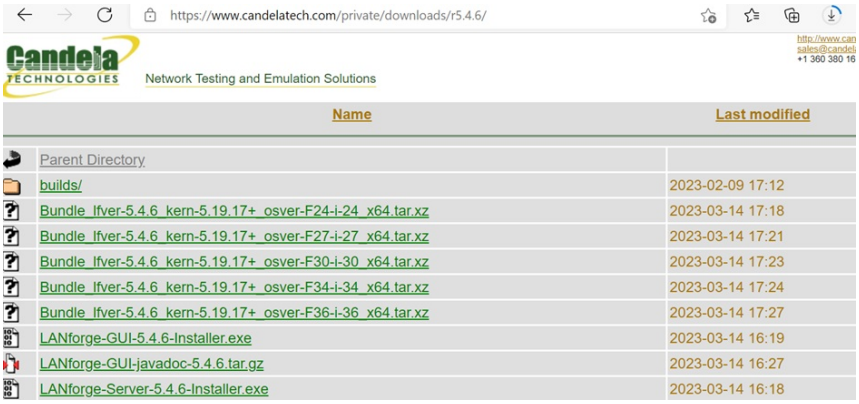
5. Reboot the laptop (with the below command). After the typing in the command, press enter.

```
shutdown/r
```

6. Now connect your laptop as per the Network topology.
7. In the powershell terminal, download the setup script again (as user admin) with the below command. After typing in the command, press enter. `wget http://www.candelatech.com/windows_lf_setup.ps1 -o windows_lf_setup.ps1`
8. Install the following applications from Google:

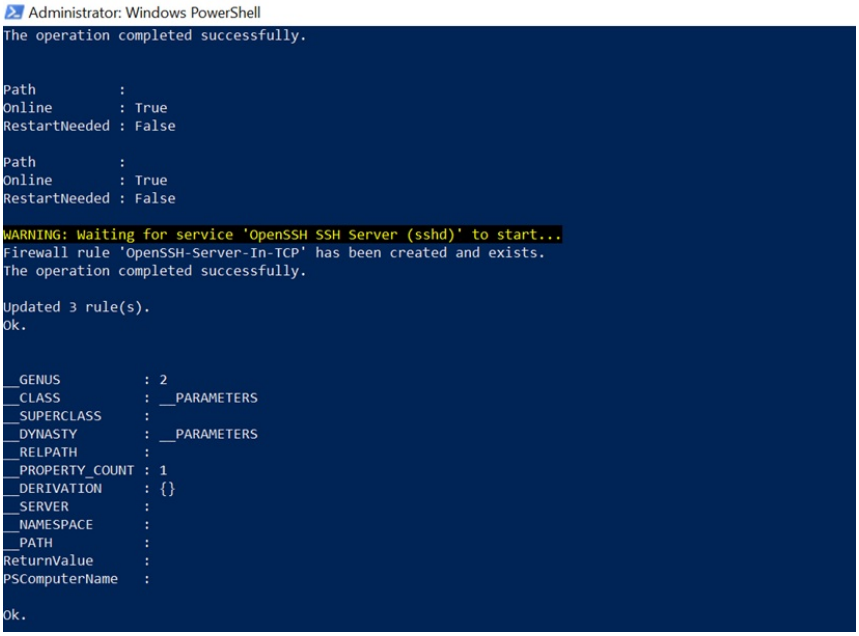
- OpenSSH client and Server
- Wireshark
- Vnc Server

- LANforge-Server-5.4.6-Installer.exe file from: Candela Technologies Website- 5.4.6
  - This link is only for 5.4.6 server. Replace the version number with whatever version number your lanforge has.

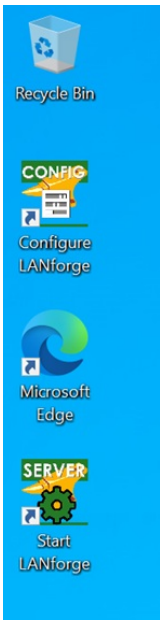


9. Type in the below command in the terminal to start the LANforge server.

```
.\windows_lf_setup.ps1 -lfver 5.4.6
```

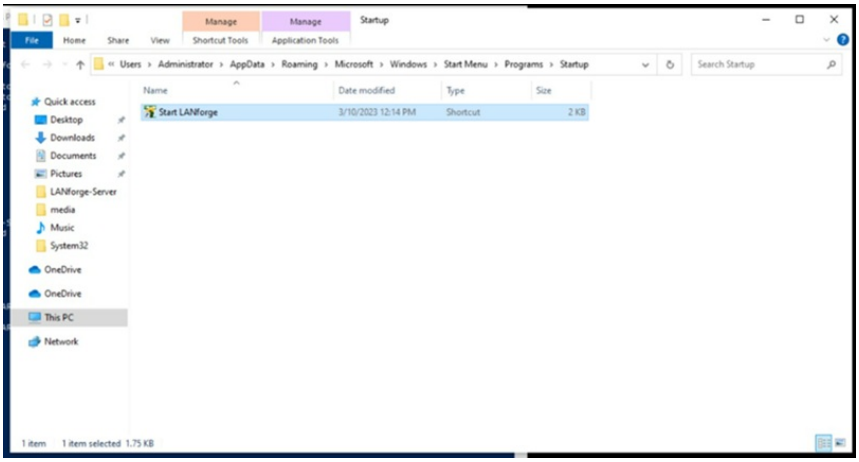


10. After the LANforge Server exe file is downloaded, run the exe file by double clicking on the file. After running the file, two icons should pop up on the Desktop page (see image below).



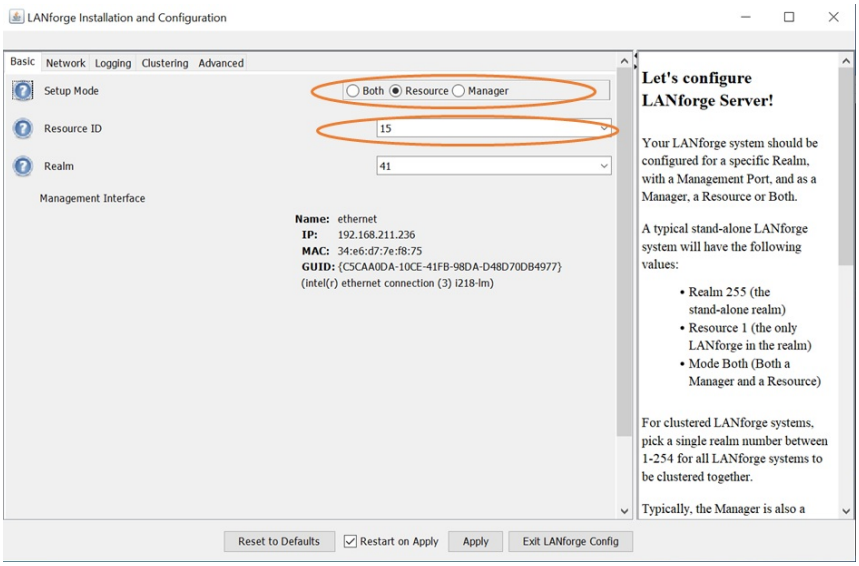
Copy the start LANforge option to the following directory such that the app will directly auto start whenever the windows machine is rebooted.

[Users >> Administrator >> AppData >> Roaming >> Windows >> StartMenu >> Programs >> Startup.]

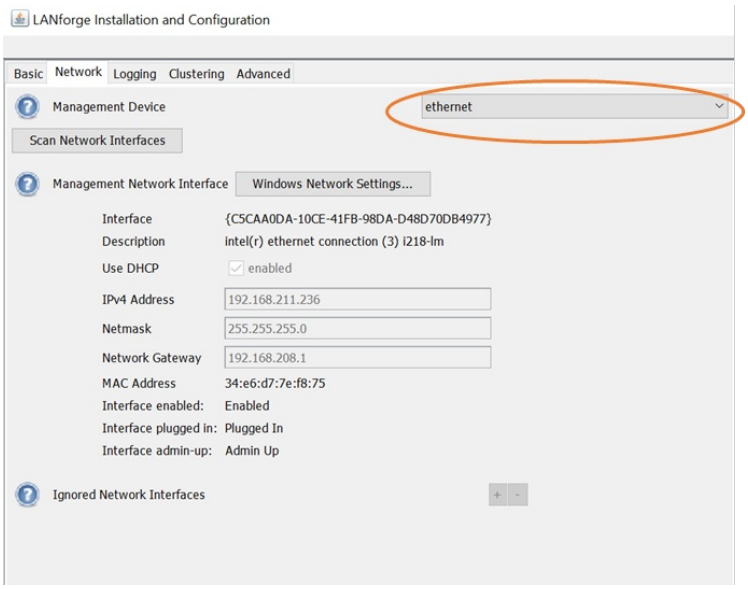


11. Now we can configure the parameters in the Configure LANforge regarding the resource ID, Realm ID and many more.

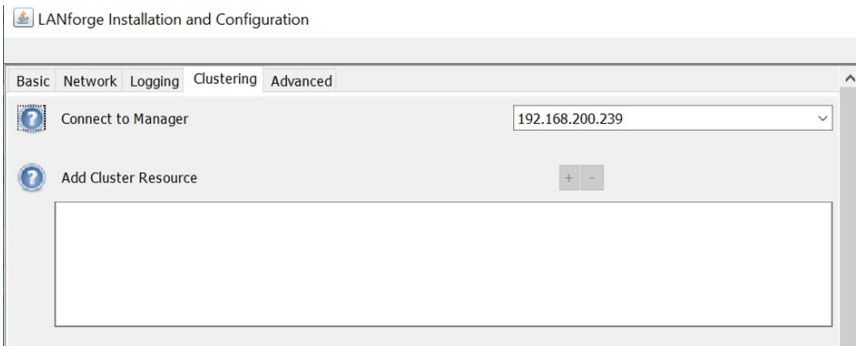
Configure the basic settings like Mode, Resource ID and realm.



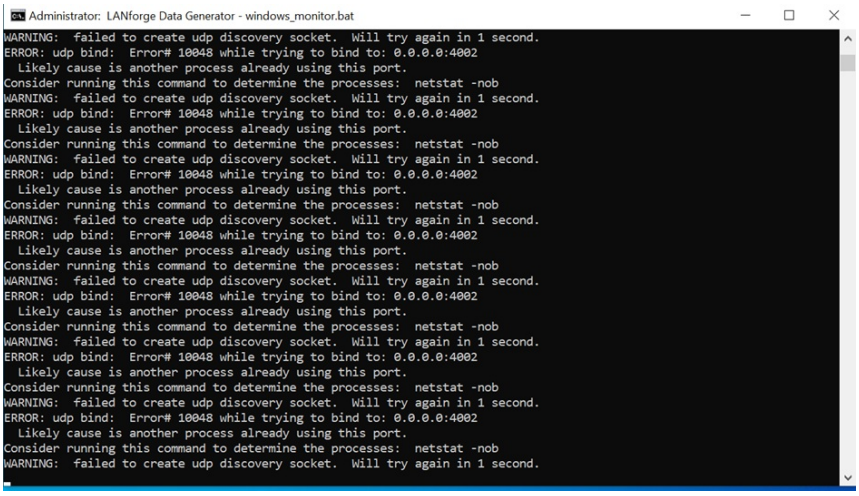
Click on the network tab and select your ethernet interface.



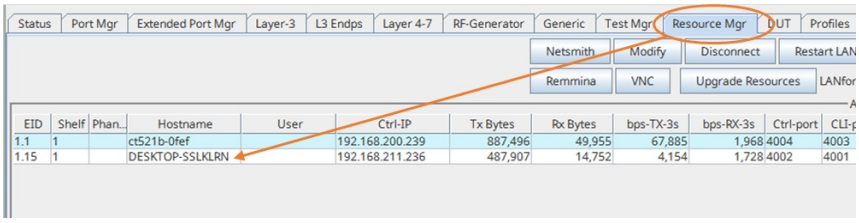
Provide the Lanforge IP for clustering the laptop to that specific IP address.



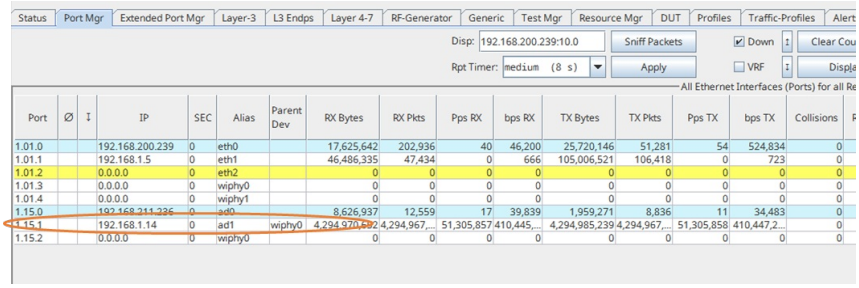
12. Click on start LANforge and you can see a following prompt on the screen.



13. Now you can detect a resource in the Resource Manager of LANforge GUI.



14. Click on Port Mgr to see number of interfaces of the Windows laptop. The "ad1" port is the Wi-Fi interface of the Windows laptop.



LANforge Server Installation for MacOS

Goal: Install the LANforge Server on the MacOS machine, have the LANforge Manager recognize the MacOS machine as a resource, and cluster the LANforge Manager with the MacOS machine.

This cookbook requires LANforge GUI version 5.4.6 and above.

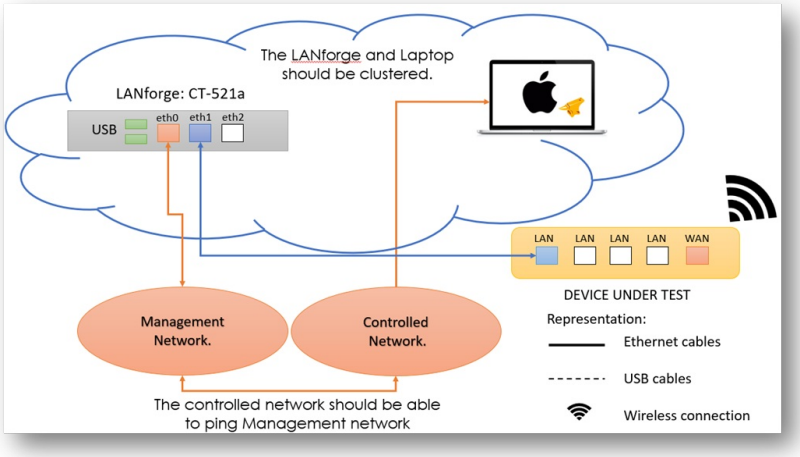
Background

The LANforge InterOp solution of Candela technologies are used to support real clients for testing Access Points. InterOp gives the ease of handling real clients with complete automation and populate a detailed reports on the entire duration of the test.

We have support for various kinds of real clients:

- Android clients.
- iOS clients.
- Windows Machines.
- Linux Machines.
- MacOS Machines.

Network Topology:

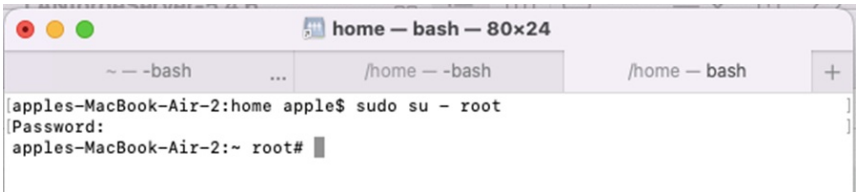


Before getting started:

All applications requiring internet for installation should be installed before starting the LANforge server.

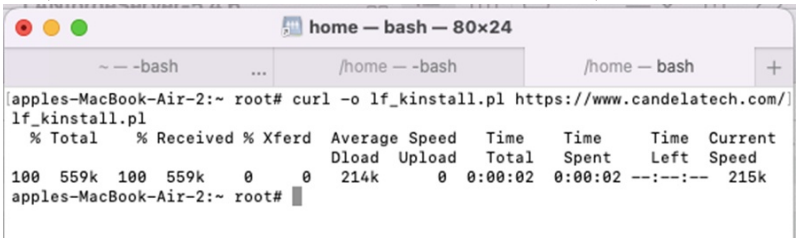
Steps to install LANforge Server on MacOS machine

1. Open a new terminal.
2. Type in the command: `sudo su – root` to become root user.



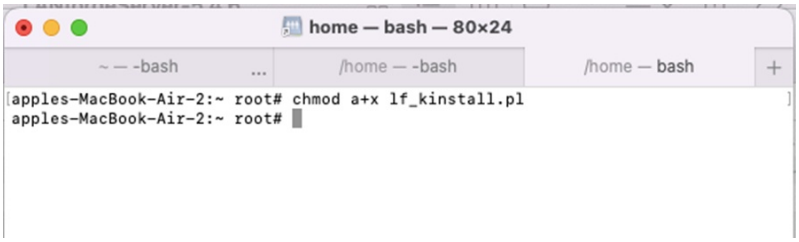
3. Type into the terminal:
    1. `curl -o lf_kinstall.pl https://www.candelatech.com/lf_kinstall.pl`

   Then, press enter. This command downloads Candela's server installation script.



    2. `chmod a+x lf_kinstall.pl`

   Then, press enter. This command give permissions and set proper bits.



3. `./lf_kinstall.pl --do_all_ct --lfver 5.4.6**`

Then, press enter. This command runs the server installation script we downloaded above (with the proper arguments).

```
# [12.6] Command: ./lf_kinstall.pl --do_all_ct --lfver 5.4.6
= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
# Done.
apples-MacBook-Air-2:~ root# ./lf_kinstall.pl --do_all_ct --lfver 5.4.6
grep: /proc/cpuinfo: No such file or directory
Argument "" isn't numeric in int at ./lf_kinstall.pl line 116.

INSTALL_LOG located at /var/root/tmp/lf_kinstall_20230317_121229.log

# Defaulting kernel version to 5.19.17+
#     is-vm 0         is-ubuntu: 0      mate-inst'd: 0    install-mate: 0
#     do-burn: 0      disk-test: 0      is-arm: 0         is-fedora: 0
#     skip-burn: 1    do-gnome: 0       uveri: 14         rpi4: 0
#     do_sys_rcfg: 0  is-deb-based: 0   osveri: 12603        osver: 12.6
#     create-bndl: 0  use-bundle: 0     is-64: 1
#     skip-kern: -1   kver: 5.19.17+   do_upgrade: 0    do_grub: 0
# Determining if: /tmp is readable
# Determining if --tmp_dir /tmp is writable
# no license.txt found
Checking md5sum of ./lf_kinstall.pl...
################################################################################ 100.0%
```

4. Type in the terminal: `cd /Users/lanforge` and press enter. Then, type in: `./lfconfig` and press enter. These 2 commands change terminal to /Users/lanforge and run the command to configure LANforge realms.

```
apples-MacBook-Air-2:/ root# cd Users/lanforge/
apples-MacBook-Air-2:lanforge root# sudo ./lfconfig
Interfaces: USB 10/100 LAN, Wi-Fi
Resource interface assignment:
  Resource 44:  Wi-Fi
Specified Resource Addresses:

Key               Acceptable Values          Value
************************************************
log_level         [0-65535]                  7
log_dir           [directory path]           /Users/lanforge
add_resource_addr [host:port]                SEE LIST ABOVE
rem_resource_addr [host:port]                SEE LIST ABOVE
realm             [1-255]                    41
resource          [1-511]                    44
mgt_dev           [ethernet device]          USB 10/100 LAN
mode              [resource, manager, both]  resource
log_file_len      [0-2G]                     0
bind_mgt          [0-1]                      0
shelf             [1-8]                      1
dev_ignore        [eth0 eth1 ... ethN]
first_cli_port    [1025-4199]                4001
connect_mgr       [host:port]                192.168.200.239:4002
gps_dev           [device file]              NONE
```

5. Specify your lanforge ip manager and press enter. Below is the example of specifying manager 192.168.200.239

`connect_mgr 192.168.200.239:4002`

```
Your command: connect_mgr 192.168.200.239:4002

Interfaces: USB 10/100 LAN, Wi-Fi
Resource interface assignment:
  Resource 44:  Wi-Fi
Specified Resource Addresses:

Key               Acceptable Values          Value
************************************************
log_level         [0-65535]                  7
log_dir           [directory path]           /Users/lanforge
add_resource_addr [host:port]                SEE LIST ABOVE
rem_resource_addr [host:port]                SEE LIST ABOVE
realm             [1-255]                    41
resource          [1-511]                    44
mgt_dev           [ethernet device]          USB 10/100 LAN
mode              [resource, manager, both]  resource
Other Commands:   help, show_all
************************************************

If these values are correct, enter 'config', otherwise change
the values by entering the key followed by the new value, for example:
mode manager
Your command:
```

8. Specify your realm id and press enter. Below is command example of specifying realm 41.

```
realm 41
```

```
Your command: realm 41

Interfaces: USB 10/100 LAN, Wi-Fi
Resource interface assignment:
  Resource 44:  Wi-Fi
Specified Resource Addresses:

Key               Acceptable Values          Value
************************************************
log_level         [0-65535]                  7
log_dir           [directory path]           /Users/lanforge
add_resource_addr [host:port]                SEE LIST ABOVE
rem_resource_addr [host:port]                SEE LIST ABOVE
realm             [1-255]                    41
resource          [1-511]                    44
mgt_dev           [ethernet device]          USB 10/100 LAN
mode              [resource, manager, both]  resource
Other Commands:   help, show_all
************************************************

If these values are correct, enter 'config', otherwise change
the values by entering the key followed by the new value, for example:
mode manager
Your command:
```

9. Set a resource number for your laptop. This is any resource number that is already used. Below is an example of using resource 44. Press enter after the command is typed.

```
resource 44
```

```
Your command: resource 44

Interfaces: USB 10/100 LAN, Wi-Fi
Resource interface assignment:
  Resource 44:  Wi-Fi
Specified Resource Addresses:

Key               Acceptable Values          Value
************************************************
log_level         [0-65535]                  7
log_dir           [directory path]           /Users/lanforge
add_resource_addr [host:port]                SEE LIST ABOVE
rem_resource_addr [host:port]                SEE LIST ABOVE
realm             [1-255]                    41
resource          [1-511]                    44
mgt_dev           [ethernet device]          USB 10/100 LAN
mode              [resource, manager, both]  resource
Other Commands:   help, show_all
************************************************

If these values are correct, enter 'config', otherwise change
the values by entering the key followed by the new value, for example:
mode manager
Your command:
```

10. Give mode as resource to act the Mac laptop as resource using the command below. Then press enter.

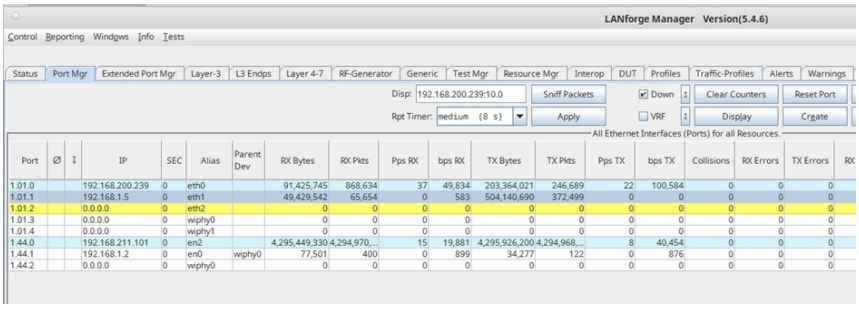
```
mode resource
```

```
Your command: mode resource

Interfaces: USB 10/100 LAN, Wi-Fi
Resource interface assignment:
  Resource 44:  Wi-Fi
Specified Resource Addresses:

Key               Acceptable Values          Value
************************************************
log_level         [0-65535]                  7
log_dir           [directory path]           /Users/lanforge
add_resource_addr [host:port]                SEE LIST ABOVE
rem_resource_addr [host:port]                SEE LIST ABOVE
realm             [1-255]                    41
resource          [1-511]                    44
mgt_dev           [ethernet device]          USB 10/100 LAN
mode              [resource, manager, both]  resource
Other Commands:   help, show_all
************************************************

If these values are correct, enter 'config', otherwise change
the values by entering the key followed by the new value, for example:
mode manager
Your command:
```

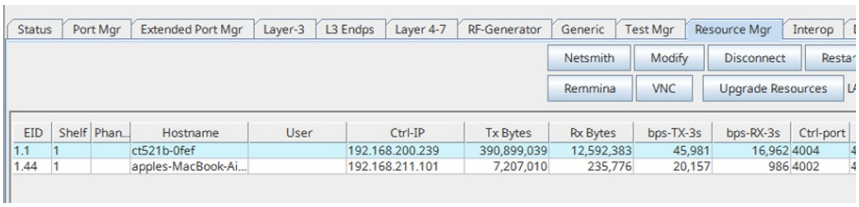
11. Type in `config` and enter to save changes.
12. Start the lanforge server by typing in `./serverctl.bash start` and pressing enter.

```
apples-MacBook-Air-2:lanforge apple$ sudo ./serverctl.bash start
Password:
chown: lanforge: illegal group name
chown: lanforge: illegal group name
Stopping LANforge server processes.
Stopping LANforge with /Users/lanforge/lf_stop

lf_stop:  Killing process: 354 (run_client_44.PID)

lf_stop:  Killing (hard) process: 354 (run_client_44.PID)
kill: 354: No such process

lf_stop:  Killing process: 8273 (btserver_44.PID)

lf_stop:  Killing (hard) process: 8273 (btserver_44.PID)
kill: 8273: No such process
Stopping any virtual_router setup scripts.
Stopping any XORPsh_helper.pl scripts.
Stopping any gen_cfg.pl scripts.
Killing wifi event helpers......
./serverctl.bash: line 202: rfkill: command not found
Killing all Xorp router processes just in case..... DONE
Verifying that DHCP server is installed..... FAILED
  ERROR:  DHCP server is not available on this system.  Configuring
```

After starting the LANforge server, go to the Lanforge GUI Port Mgr tab to find your MacOS machine clustered.



Resource Mgr tab will also show the laptop and the resource id that was given earlier above to the MacOS machine.



Now en0 port can be used as station of laptop wiphy radio and run traffic.

LANforge Server Installation for Linux (Ubuntu) Machines

Goal: Install the LANforge Server on the Linux (specifically Ubuntu) resource machine and configure the new resource to contact the manager machine to join the LANforge realm.

This cookbook requires LANforge GUI version 5.4.6 and above.

Background

The LANforge InterOp solution of Candela technologies are used to support real clients for testing Access Points. InterOp gives the ease of handling real clients with complete automation and populate a detailed reports on the entire duration of the test.

We have support for various kinds of real clients:

- Android clients.
- iOS clients.
- Windows Machines.
- Linux Machines.
- MacOS Machines.

Before getting started:

All applications requiring internet for installation should be installed before starting the LANforge server.

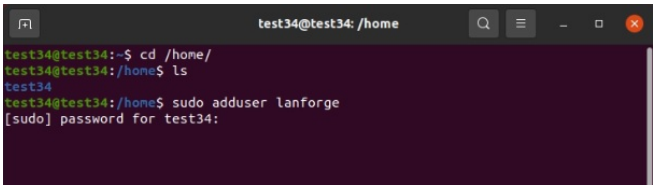
LANforge Server Installation Steps on Ubuntu 18.04.5 and 20.04.5 LTS machine:

1. Open a new terminal.
2. Create LANforge Accounts:

Create a lanforge user and install the LANforge software in the lanforge home directory. Do this by typing in these commands in the terminal:

`sudo adduser lanforge` [ENTER]

`password lanforge` [ENTER]



3. Install the LANforge Server.

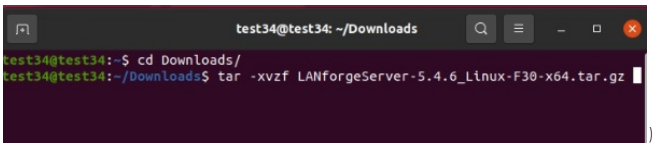
From the candelatech page download the LANforgeServer-5.4.6_Linux-F30-x64.tar.gz file in Ubuntu 18.04.5 or Ubuntu 20.04.5 LTS machine.

)

4. Open a new terminal and navigate to Downloads directory using cd Downloads and untar the downloaded file using the below commands.

`cd Downloads` [ENTER]

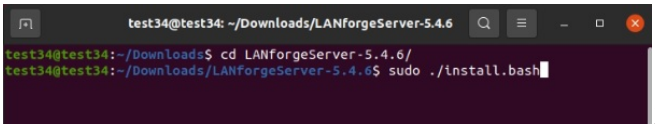
`tar -xvzf LANforgeServer-*_Linux-x86.tar.gz` [ENTER]



5. Navigate to the lanforge server path. Then, as root user, install the lanforge server using the below commands.

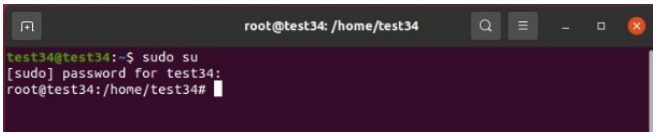
`cd LANforgeServer-5.4.6/`

`sudo ./install.bash`



6. Open a new terminal and type in commands below into that terminal:

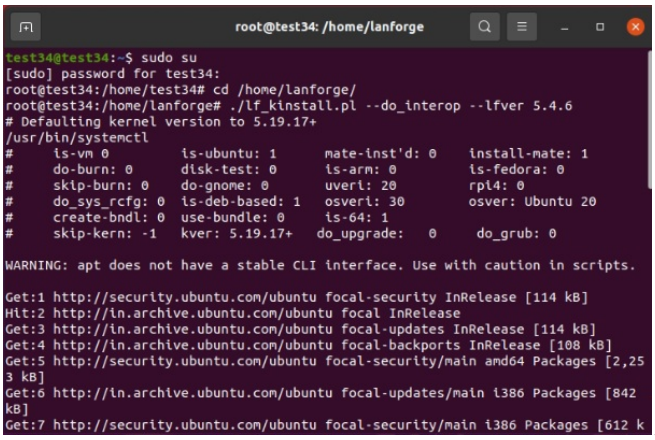
`sudo su` [ENTER] (to be root user)



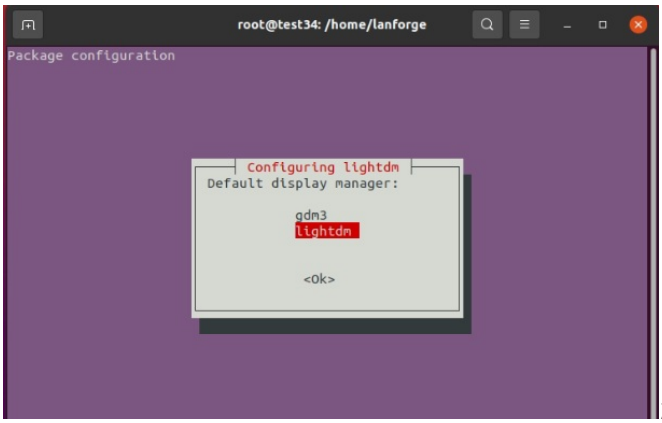
`cd /home/lanforge` [ENTER]

`./lf_kinstall.pl --do_interop --lfver 5.4.6` [ENTER]

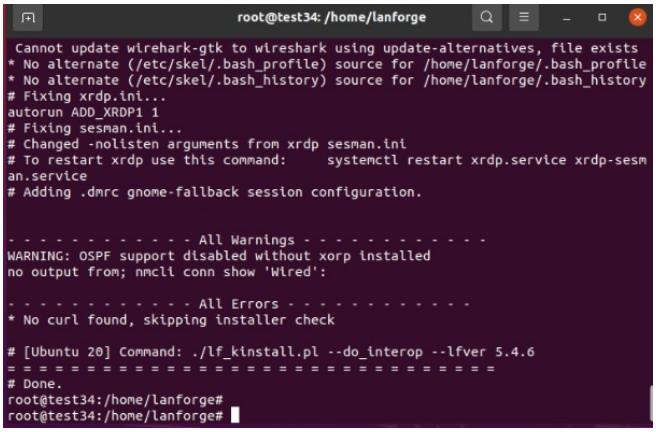
These set of commands will run the server installation script we downloaded above (with the proper arguments).



In the package configuration select “lightdm”.



After the installation is successful a "Done" message will be shown in the mate terminal.

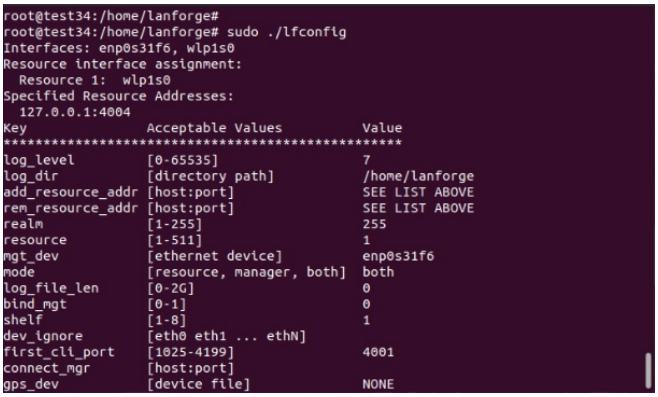


8. Configure Lanforge Server

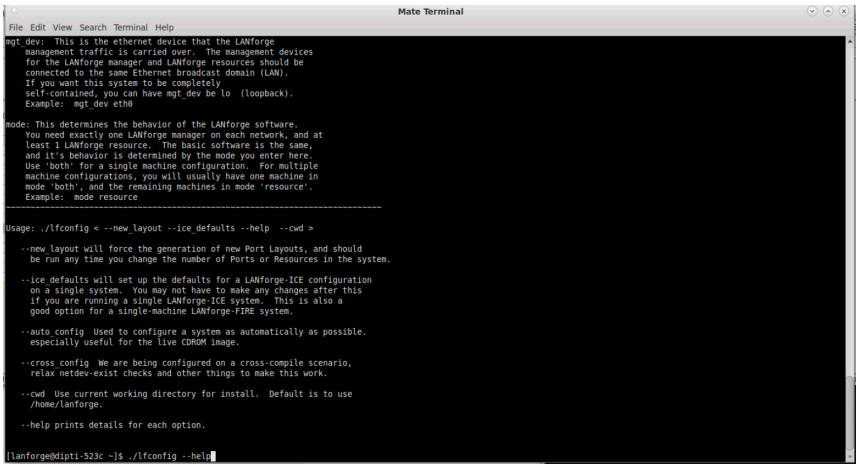
After running the install script, cd to the /home/lanforge directory as root and run the lfconfig script to configure LANforge (as below):

```
cd /home/lanforge
```

```
sudo ./lfconfig
```



Running the command `./lfconfig --help` in the terminal will display all flags for the lfconfig script and what each usage is.

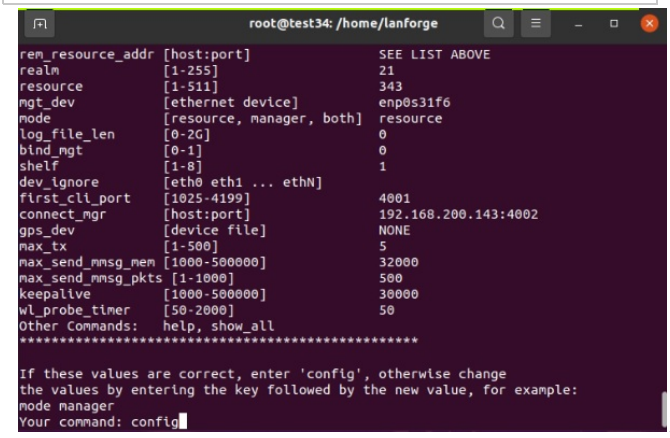


)

9. LANforge Manager with Clustered Resources

Configure LANforge resources in the same realm as the manager as illustrated by the list of lfconfig key value entries in the example below:

Manager (Resource #1)	Resource #2	Resource #3	Resource #4
mgt_dev eth0 resource 1 realm 22 mode both config	mgt_dev eth0 resource 2 realm 22 mode resource config	mgt_dev eth0 resource 3 realm 22 mode resource config	mgt_dev eth0 resource 4 realm 22 mode resource config

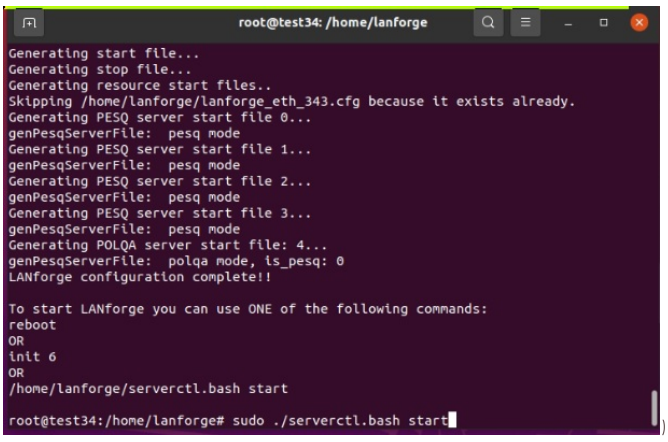


)

10. Start Lanforge server

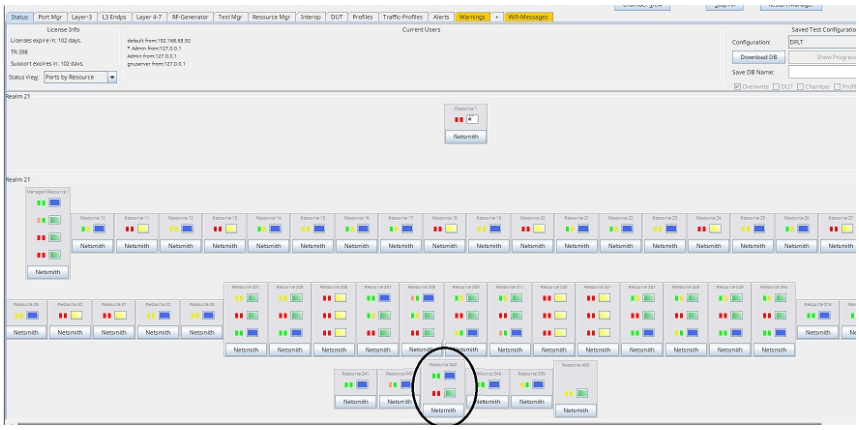
To start the LANforge servers, use the serverctl.bash script as root shown in the commands below:

```
sudo./serverctl.bash start
```



)

11. All clustered resources will now be displayed on the Status page of the LANforge GUI and listed in the Resource Mgr tab.



)

Steps to install LANforge Server on Ubuntu 16.04 LTS machine:

1. Open a new terminal.
2. Create LANforge Accounts

Create a lanforge user and install the LANforge software in the lanforge home directory.

```
sudo adduser lanforge;password lanforge
```

3. Install LANforge Server

From the candelatech page download the LANforgeServer-5.4.6_Linux-F21-x64.tar.gz file in Ubuntu 16.04 LTS machine.

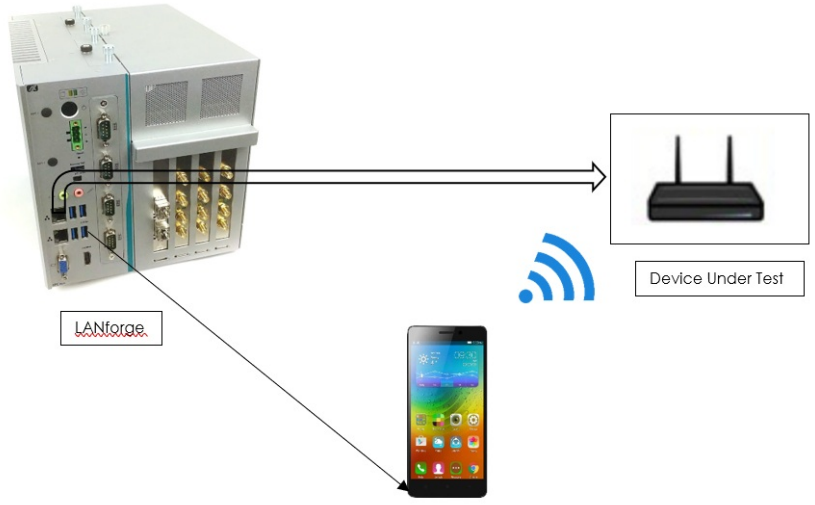
4. After downloading follow all the above steps from 4-11 for installing lanforge server in Ubuntu 16.04 machine.

Real-Browser Test on Android

Goal: In LANforge, Layer 4-7 Traffic is used to emulate Web-Browsing. Web-browsing can be offline or online. This cookbook demonstrates how to emulate web-browsing with the real client Android phone.

This cookbook requires LANforge GUI version 5.4.6 and above.

Test Setup

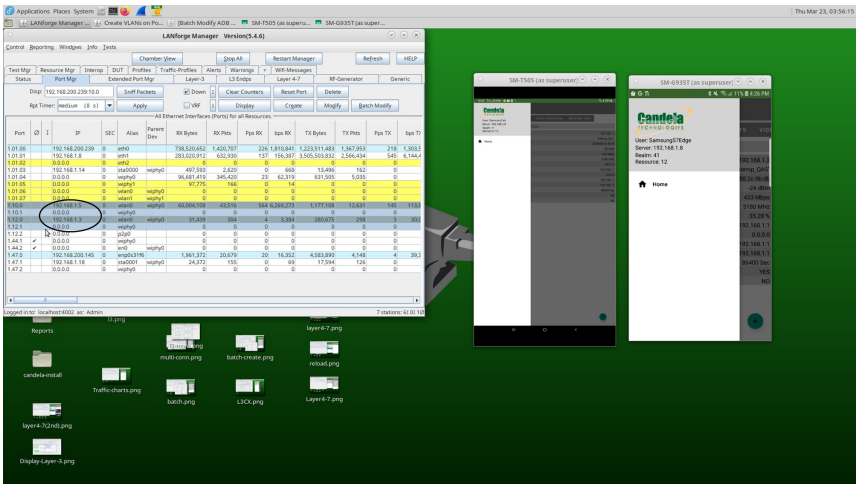


Online Real-Browsing Test

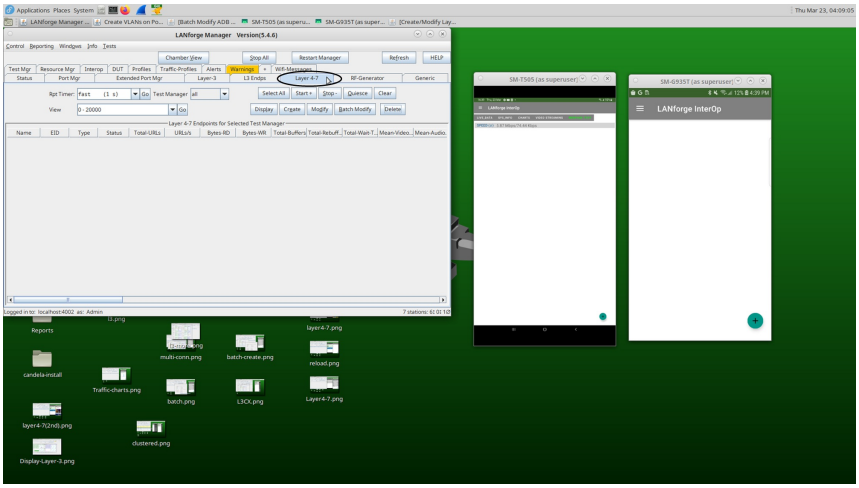
1. Make sure that the DUT is connected to the internet. Initial Android interop setup can be done using the following link.

2. The below picture is an example of two android phones (resource IDs 10 and 12) clustered to the LANforge in the Port Mgr tab. This is both the radios and mgr port of the phones.

Make sure that the mobile phone devices appear in the Port Mgr tab after intial Android InterOp setup.

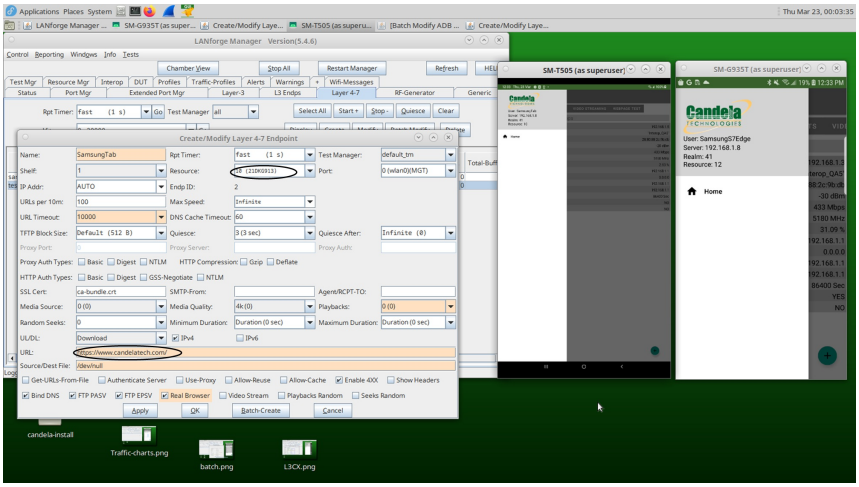


3. Click on the Layer 4-7 tab in LANforge and Webpage Test tab in the InterOp app on the phone.



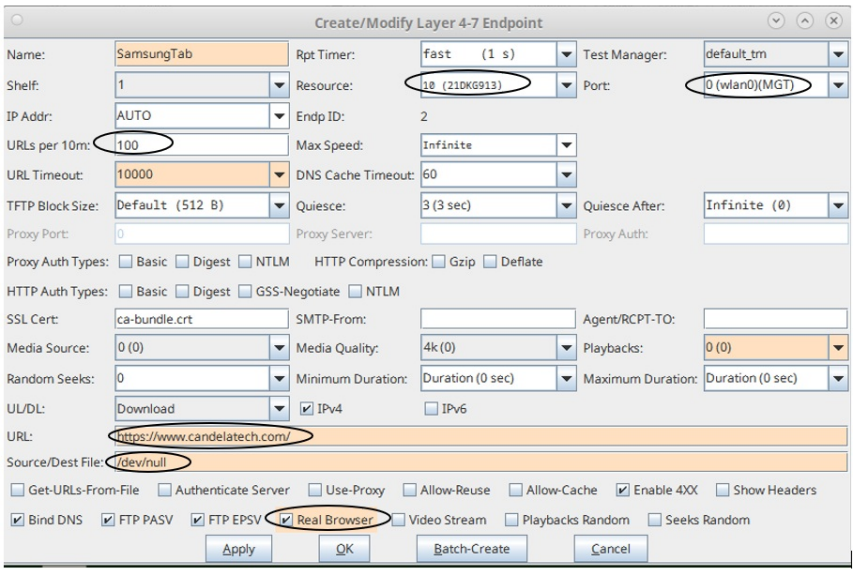
4. Create a Layer4-7 cross-connect in the LANforge Layer 4-7 tab using your phone (and resource ID).

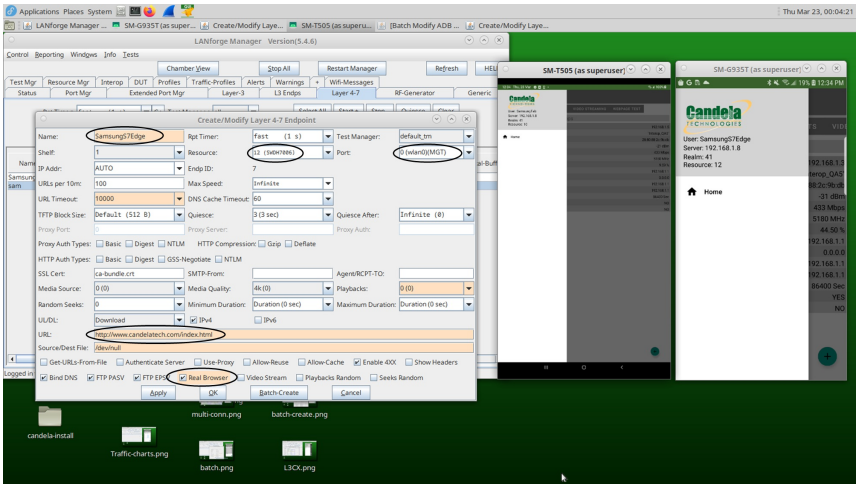
Add the details below.



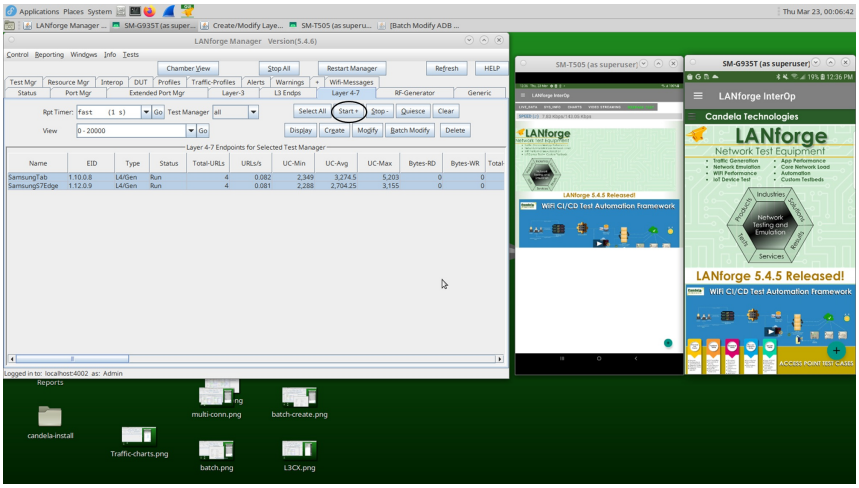
Enter URL to be browsed, source/dest file, select the resource-ID of the phone, and enable the Real Browser option, mention URLs per 10minutes.

Below are examples of creating cross-connects using resource IDs 10 and 12.

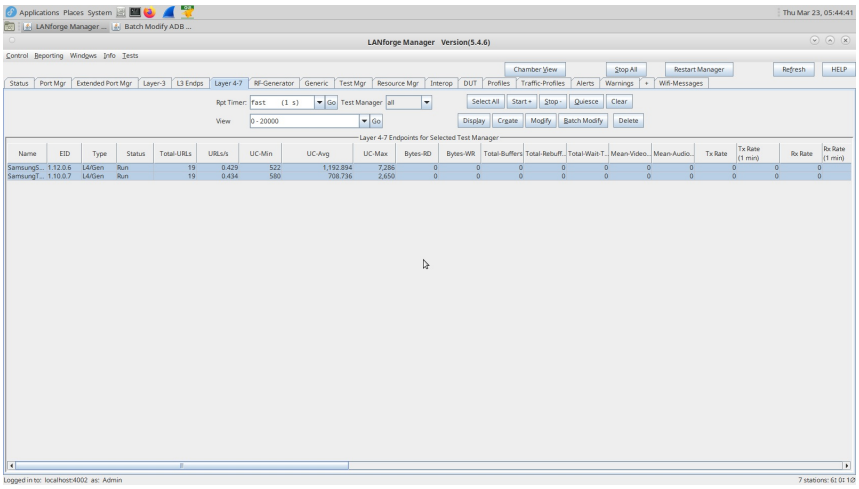




5. Select both the cross-connects and click Start. Then, in the Android Webpage Test tab, the Candela Technologies website should load in both the phone. The statistics should of the Layer 4-7 cross-connect should also start being report in the GUI Layer 4-7 tab.



Stats include Total-URLs, URLs/s, UC-Min, UC-Avg, UC-Max and Total ERR.



Total-URLs: URLs processed and in process. This includes passed and failed URLs.

URLs/s: URLs processed per second over the last minute.

UC-Min: Minimum time in milliseconds to complete processing of the URL for requests made in the last 30 seconds.

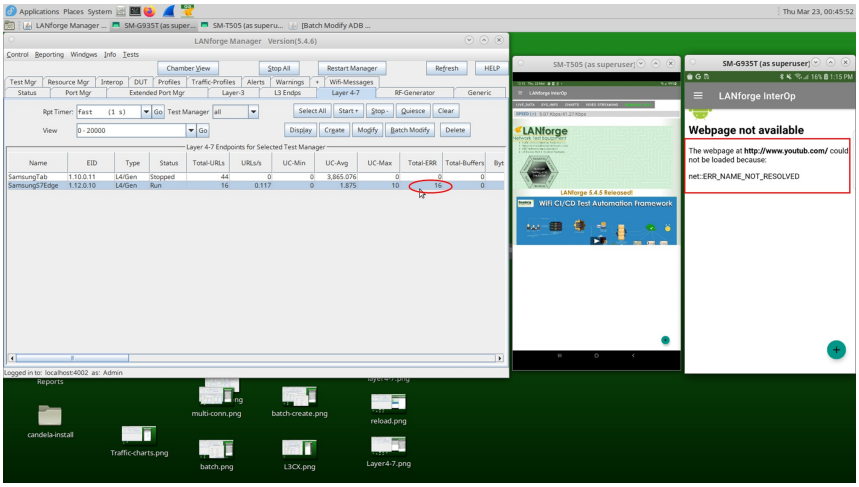
UC-Avg: Average time in milliseconds to complete processing of the URL for the last 100 requests.

UC-Max: Maximum time in milliseconds to complete processing of the URL for requests made in the last 30 seconds.

Total ERR: Total failed URLs.

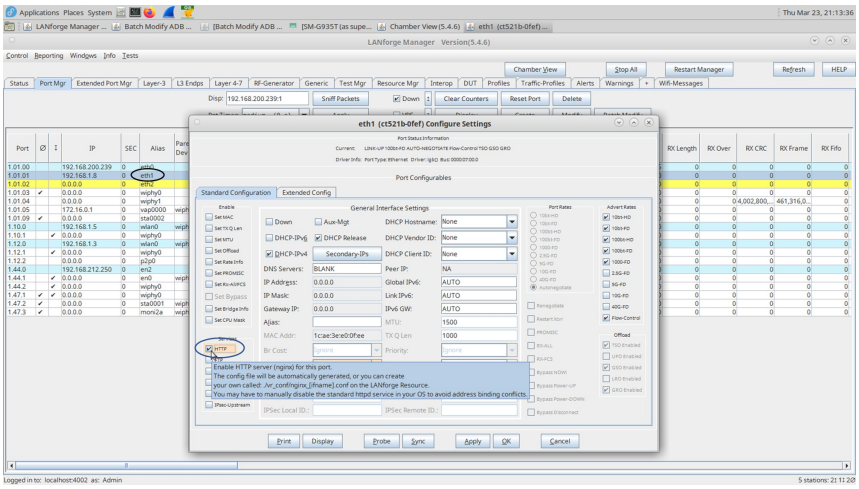
6. Next, enter a wrong URL and start the cross-connect to observe Total-ERR (Total number of failed URLs).

Observed below are 16 failed URLs.



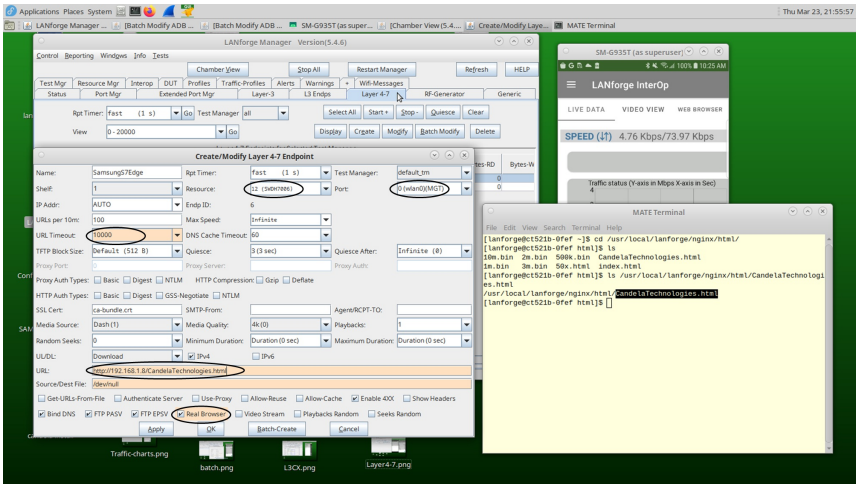
Offline Real-Browser Test

1. In the offline browsing test, there is no need have internet enabled on the DUT. First, run nginx on eth1. Double click on the port, click the HTTP check-box on the left hand side, and Apply the settings.

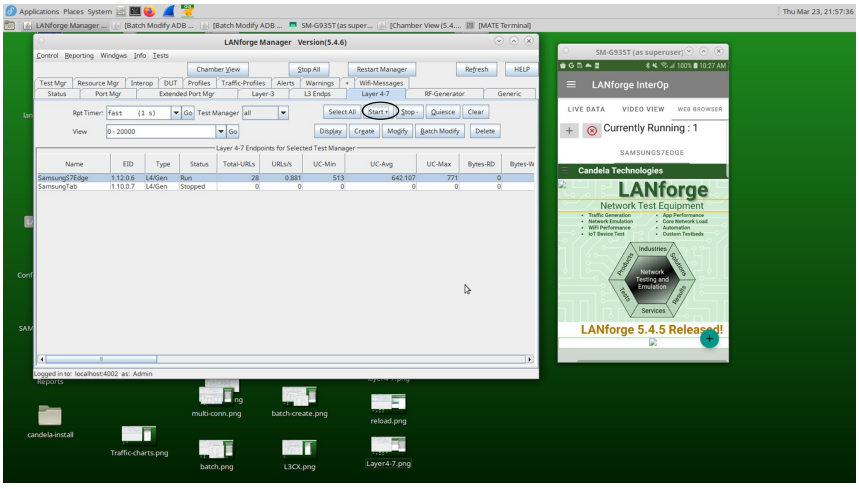


2. Download any HTML page and place the .html page in the html directory: **/usr/local/lanforge/nginx/html/** .
3. As you can see from the image below, CandelaTechnologies.html was placed in the **/usr/local/lanforge/nginx/html/** path.

Next, create a Layer4-7 cross-connect and fill out the boxes with the information given below.



4. Start the cross-connect and view the Web Browser tab in the app see the web page load. Statistics will also start to populate in the Layer 4-7 tab after the cross-connect has started.

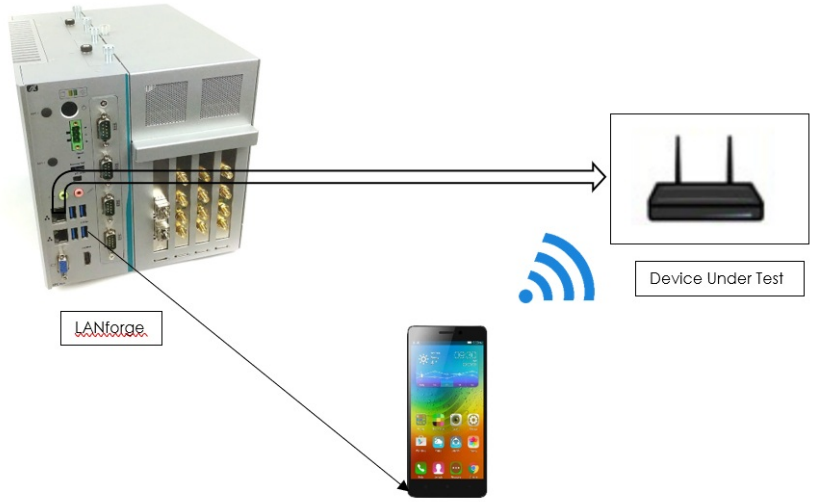


Layer-3 Traffic Test on Android

Goal: Create Layer-3 Traffic between LANforge Manager and Android Resource.

This cookbook requires LANforge GUI version 5.4.6 and above.

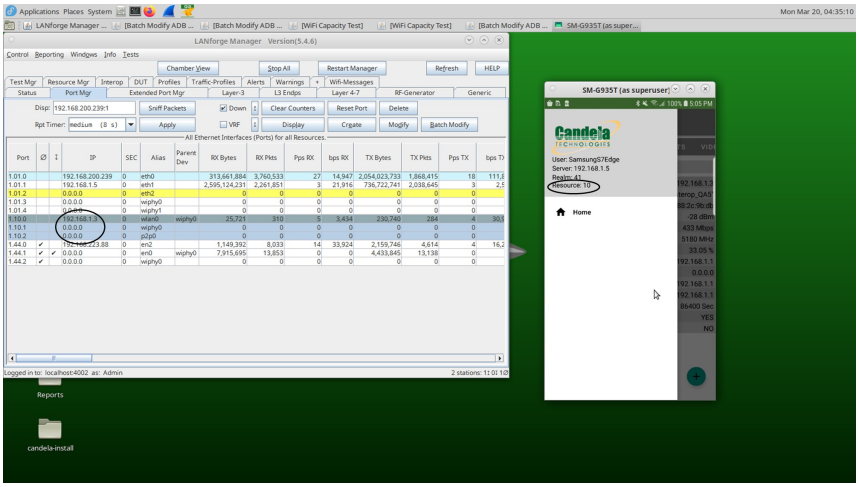
Test Setup



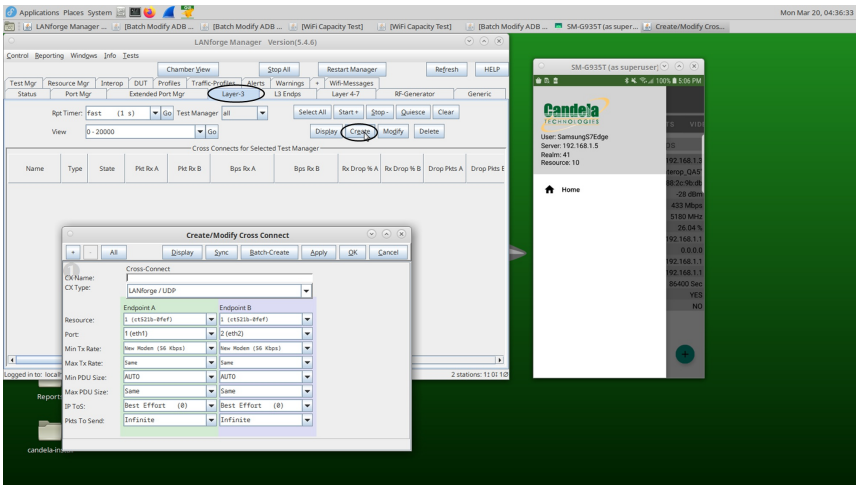
Create Layer-3 TCP/UDP Traffic between LANforge Manager and Android resource

1. Connect Android phone to LANforge via USB such that the device will appear in Interop Tab. Initial Android interop setup can be done using the following link.
2. Below is an example of a phone (resource ID 10) that is now clustered with the LANforge.

After entering the test room, the device should appear in Port Mgr tab.



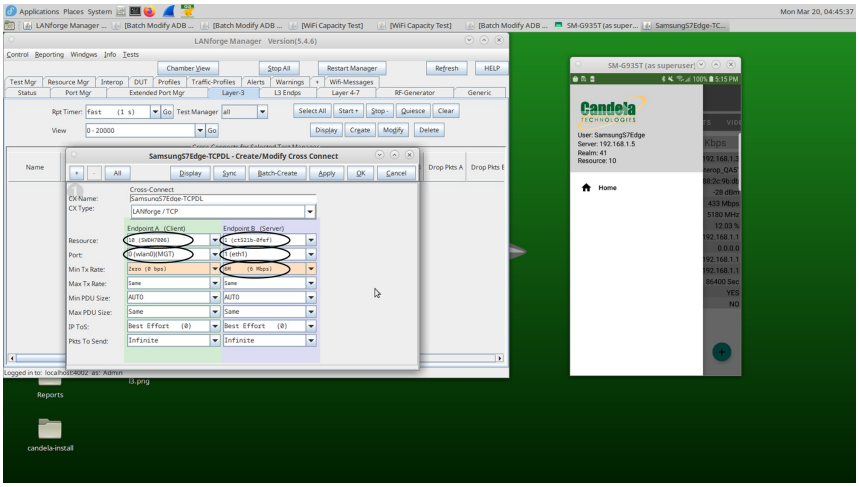
3. Click on the Layer-3 tab, then click on the Create button. This will open a window to create a Layer-3 cross-connect.



4. Add the details as mentioned below.

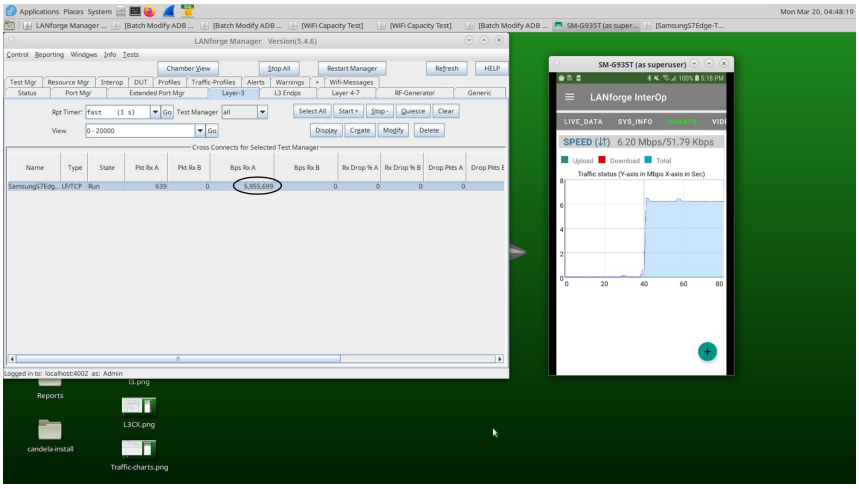
Enter CX name of your choice, CX type (TCP or UDP), endpoint-A value (the resource-ID of your phone),endpoint-B value (the resource-ID of your LANforge in Resource option). Also, select respective ports to run the traffic and set Min Tx Rate as required on server side to run download traffic. Click on Apply and OK to save the cross-connect settings.

The below image is of a **TCP downstream** cross-connect with an intended load of 6 Mbps.

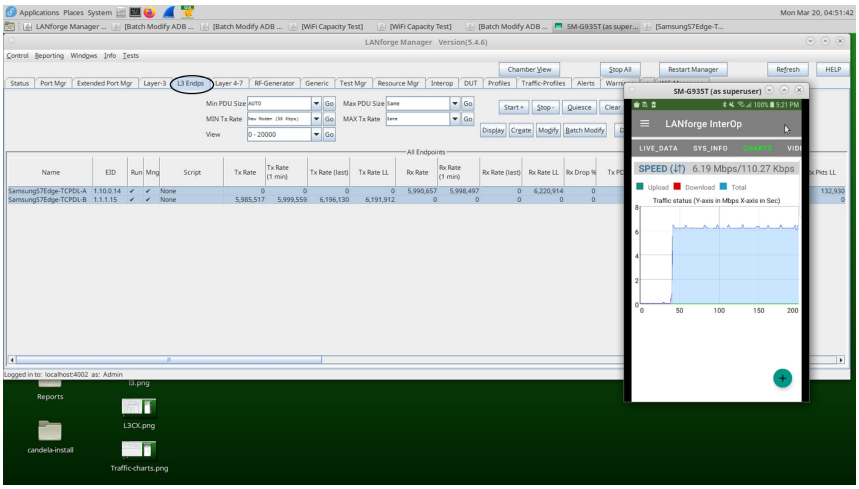


5. Click the Charts tab on the app. Then, in the GUI, start the Layer-3 cross-connect by clicking the cross-connect in the table to highlight it and clicking the Start button in the Layer-3 tab. The graph will start plotting immediately after starting the Layer-3 cross-connect.

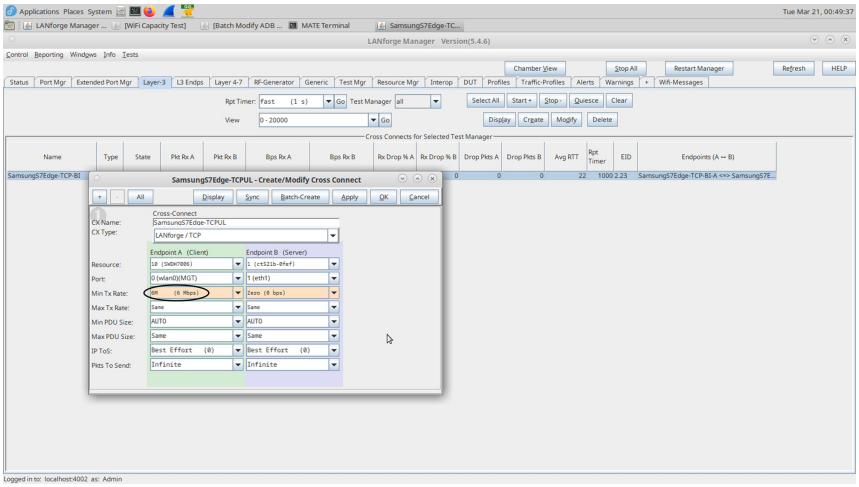
Throughput obtained in Layer-3 will be reflected in the Charts tab in the app.

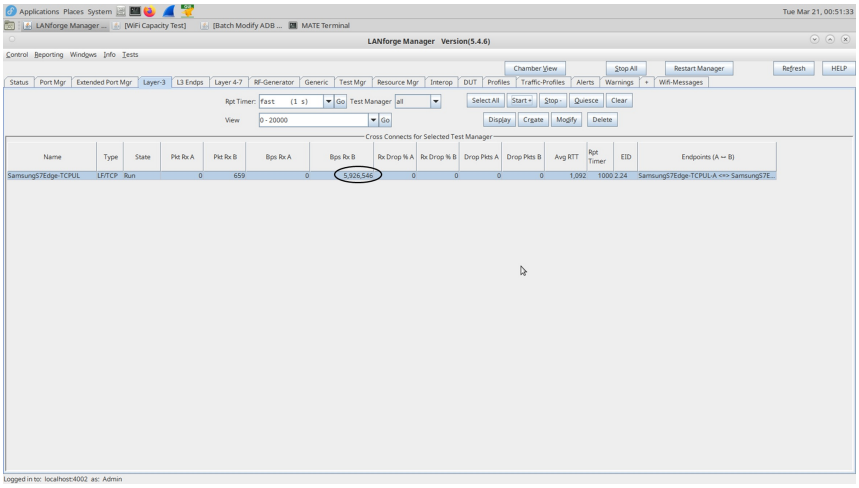


6. Further cross-connect data points are displayed in the L3 Endps tab. Also, if a cx on the Layer-3 tab is highlighted (by single clicking it's row), the endpoints associated with that cx will be selected when switching to the L3 Endps tab.

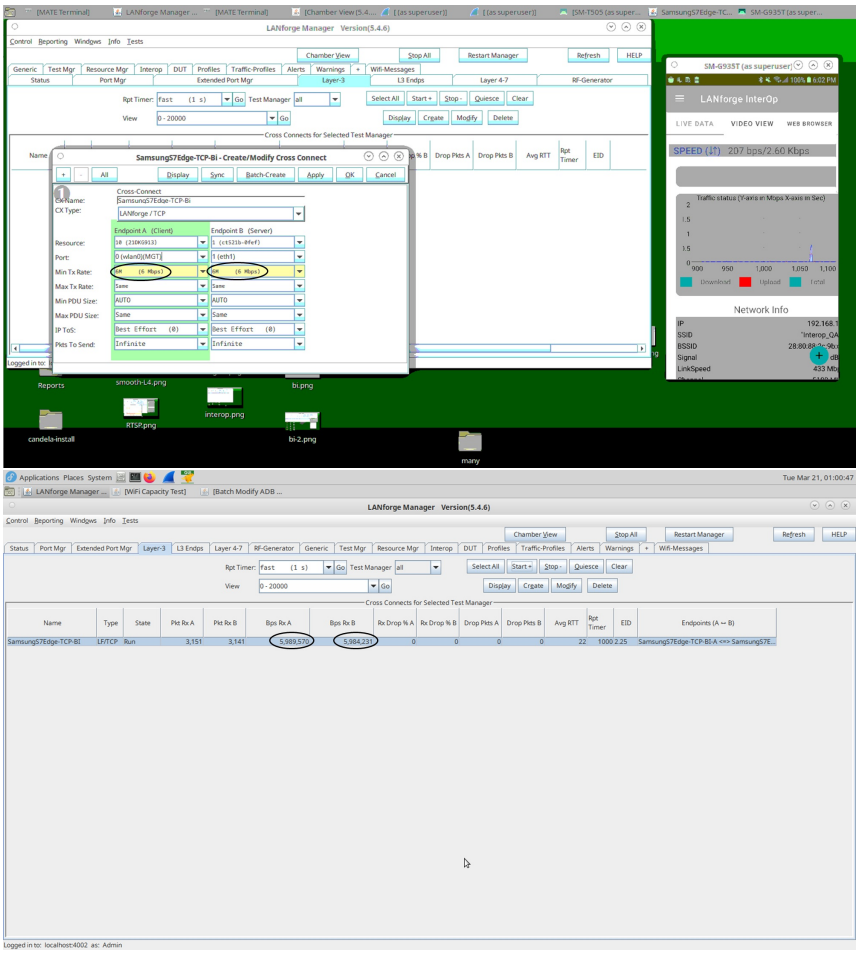


Below is an example of **TCP Upstream** traffic by selecting the Tx rate on client side.



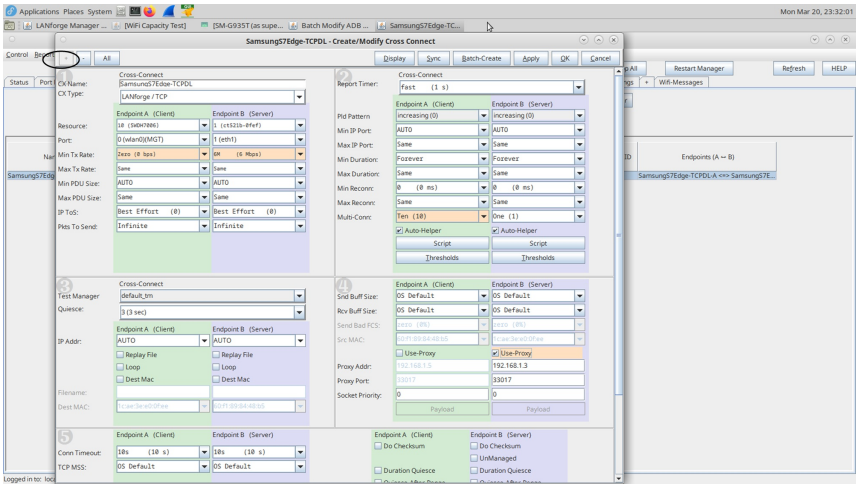


Bi-Directional traffic can be run by setting the Tx-Rate ate on both client and server side.



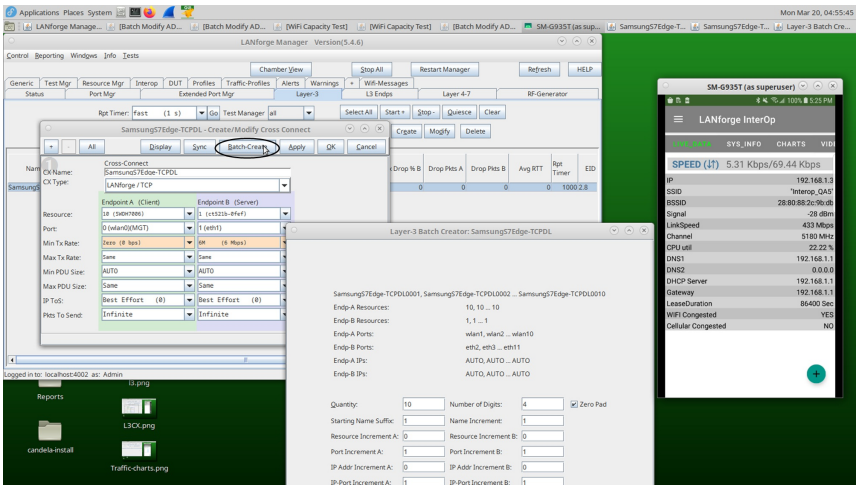
Similarly, a UDP cross-connect can be created by changing the CX Type to LANforge/UDP.

7. The Modify window (found by single-clicking a cross-connect and clicking Modify or double-clicking on a cross-connect) displays the first configuration section which is sufficient for creating a cx. Further cross-connect options are displaying by pressing the + button in the upper right-hand corner.

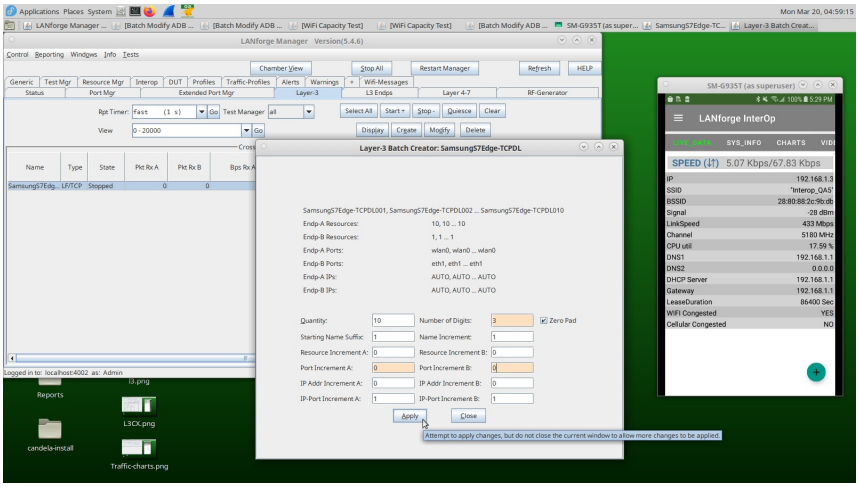


To understand more about each option, click on this link.

8. Multiple cross-connects can be created with a single station by double-clicking on the cross-connect and selecting Batch-Create option.

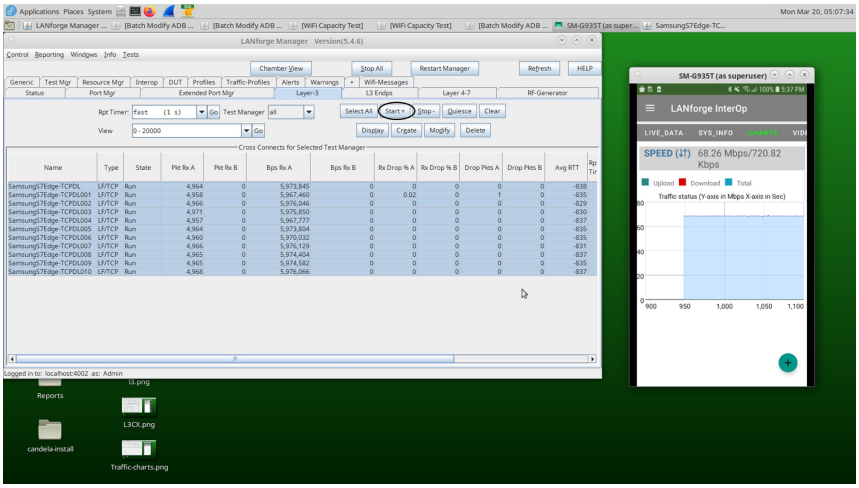


9. Below, Quantity, Port increment, Resource Increment are entered in, then saved by clicking on Apply.



10. Select all the cross-connects (by clicking the Select All button) and press Start when all of them are highlighted. This will start all of them. Then, open the Charts tab in the app to observe the throughput.

Below, the throughput is around 66 Mbps the same thing is reflected in Charts tab.



Video Streaming Test on Android

Goal: Run a video streaming test on an android device to measure access point performance and stability. This test streams videos from the local browser on the LANforge or on the Android phone which is connected to the access point.

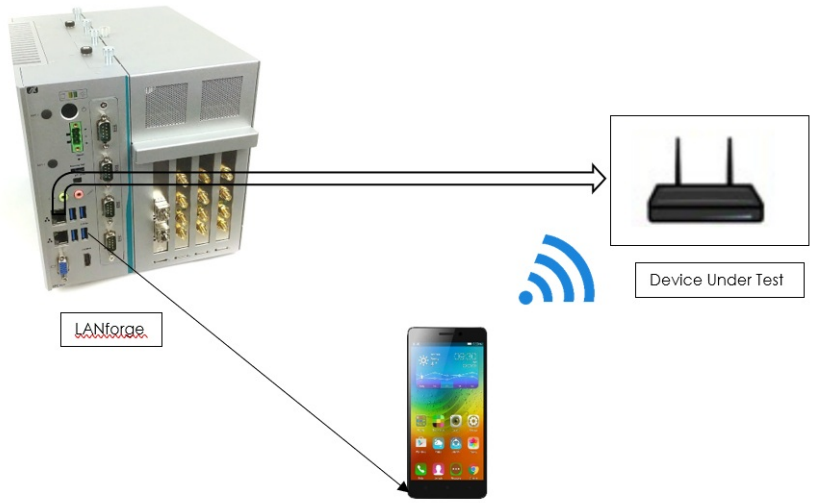
This cookbook requires LANforge GUI version 5.4.6 and above.

Background

The LANforge InterOp App has an inbuilt Video Media Player called Exoplayer which can run various types of media formats. Some examples are:

```
*   MP4
*   DASH (Dynamic Adaptive Streaming over HTTPS)
*   Smooth Streaming Media Source
*   HLS- Media Source (HTTP Live Streaming)
*   Progressive Media Source
*   Real Time Streaming Protocol
```

Test Diagram



Video Streaming can be done offline or online.

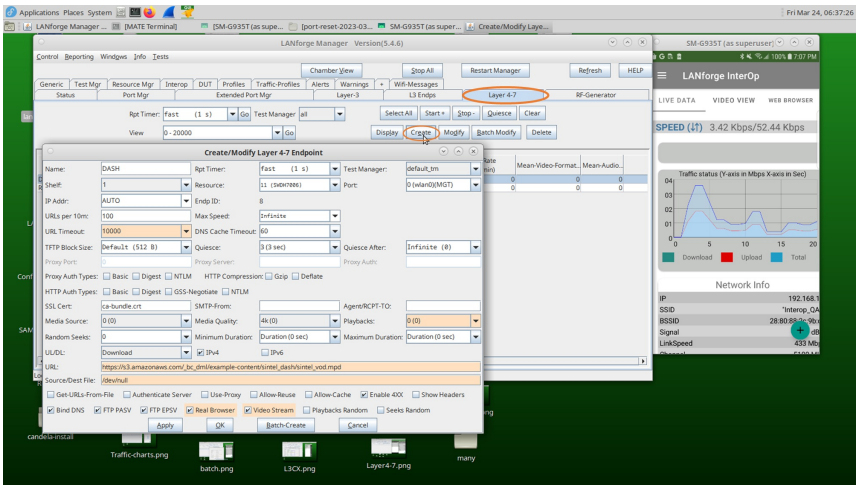
Make sure the DUT is connected to internet & initial Android InterOp setup is complete. Initial Android InterOp setup can be found using the following link: InterOp Android Setup

Online Video Streaming

DASH Video Streaming

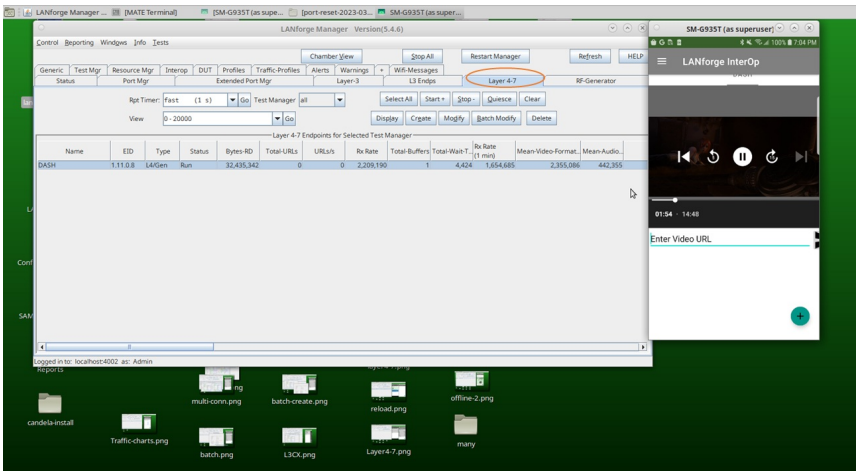
1. Create a Layer 4-7 cross-connect by using the details pictured below.

   Use the following sample URL link to run DASH video : https://s3.amazonaws.com/_bc_dml/example-content/sintel_dash/sintel_vod.mpd



2. Start the Layer 4-7 cross-connect and select the Video-View tab in the app to view the running video.

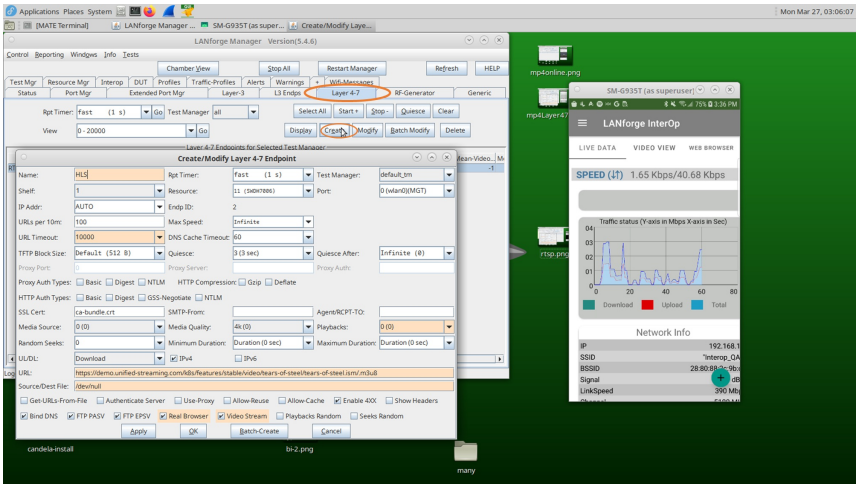
   While the video is running, the Layer 4-7 cross-connect statistics (e.g. Bytes-RD, Total-URLs, URLs/s, Rx Rate, Total-Buffers, Total Wait Time, Mean Video format, Mean Audio format) are able to be viewed in the Layer 4-7 tab.



HLS (HTTP Live Streaming)

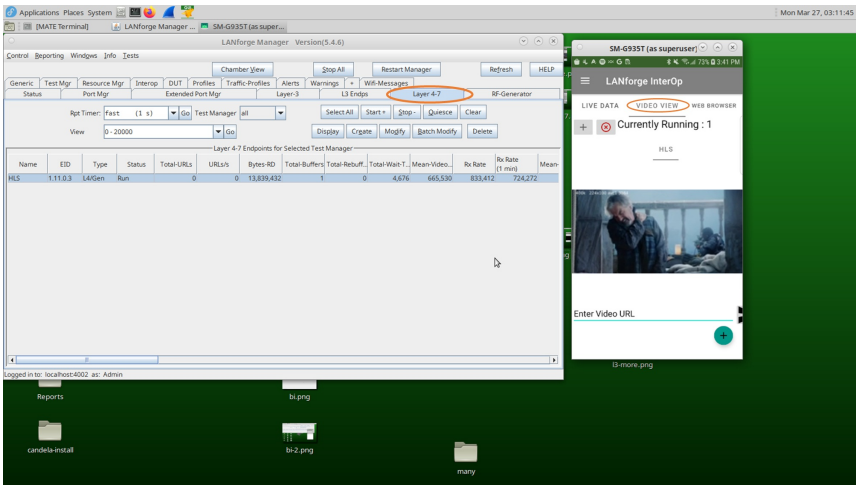
1. Create a Layer 4-7 cross-connect by using the details pictured below.

   Use the following sample URL to run HLS: https://demo.unified-streaming.com/k8s/features/stable/video/tears-of-steel/tears-of-steel.ism/.m3u8



2. Start the Layer 4-7 cross-connect and select the Video-View tab in the app to observe the running video.

While the video is running, the Layer 4-7 cross-connect statistics (such as Bytes-RD, Total-URLs, URLs/s, Rx Rate, Total-Buffers, Total Wait Time, Mean Video format, Mean Audio format) are able to be in the Layer 4-7 tab.



The same method of creating a Layer 4-7 cross connect and observing the media in the app can be applied to other media types such as Smooth, RTSP, and MP4.

Some sample links for those 3 media types:

Smooth: http://playready.directtaps.net/smoothstreaming/SSWSS720H264/SuperSpeedway_720.ism/Manifest

RTSP: rtsp://wowzaec2demo.streamlock.net/vod/mp4:BigBuckBunny_115k.mp4

MP4: https://filesamples.com/samples/video/mp4/sample_640x360.mp4

Offline Video Streaming

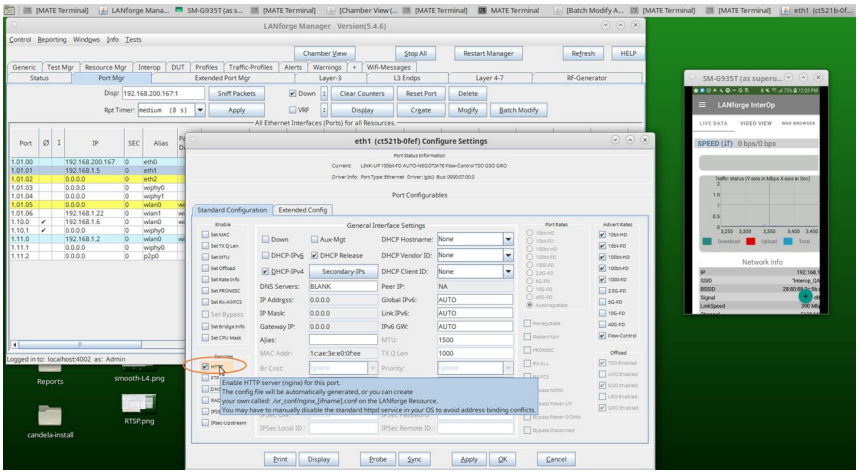
When offline video streaming, the DUT does not need to be connected to the internet.

DASH Video Streaming

To run an offline DASH video, refer to the following cookbook link:
https://www.candelatech.com/cookbook.php?vol=wifire&book=Interop:+Test+Video+Streaming+with+DASH+Server+and+Interop+APK

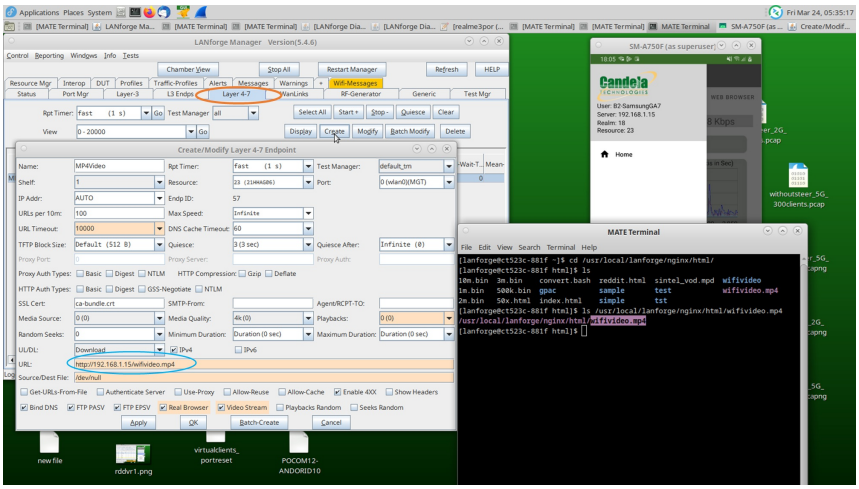
MP4 Streaming

1. Run nginx on eth1 port by double-clicking on the eth1 port in the Port Mgr, checking the HTTP box circled below, and clicking Apply.

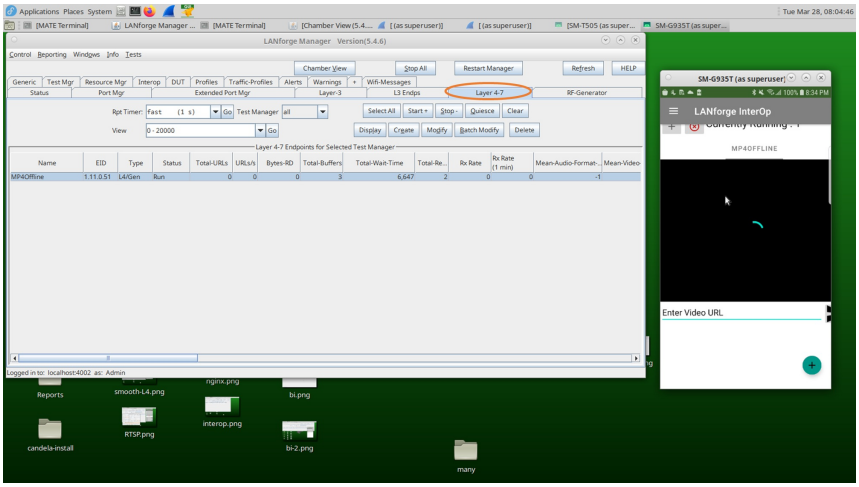


2. Download and place the .mp4 file in the following path: /usr/local/lanforge/nginx/html/

Create a Layer 4-7 cross-connect and add the details pictured below to the Layer 4-7 cross connect. Enter the URL, Source/Dest File, Resource (of the Android mobile), and check the Real Browser and Video Stream boxes.



3. Start Layer4-7 cross-connect and go to Video-View tab in app to observe the running video. Stats can be observed in the Layer 4-7 cross-connect row.



LANforge InterOp Setup for iOS

Goal: Cable an iPhone to a LANforge machine, then have the LANforge machine recognize the iPhone as a resource.

This cookbook requires LANforge GUI version 5.4.7 and above.

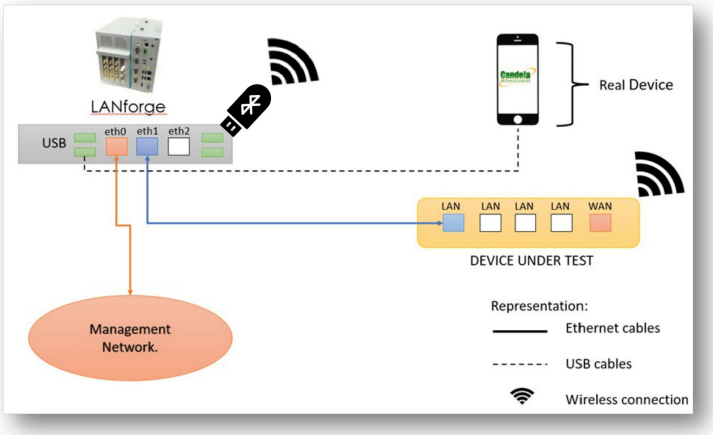
Background

Our LANforge InterOp solution is used to support real clients for testing access points. InterOp gives the ease of handling real clients with complete automation and detailed, customizable test reporting.

We have support for various kinds of real clients:

- Android clients.
- iOS clients.
- Windows Machines.
- Linux Machines.
- MacOS Machines.

Network Topology:



Hands-Free iPhone Control:

In order to provide a hands-free solution for iOS testing, LANforge controls each iPhone via a Bluetooth connection. USB Bluetooth adapters plugged into the LANforge act as Bluetooth keyboard devices. An adapter connects with the iPhone and then sends it keystroke packets. These keystrokes are interpreted by the iPhone as actions, and it's possible to perform system functions, navigate the UI, send text, etc.

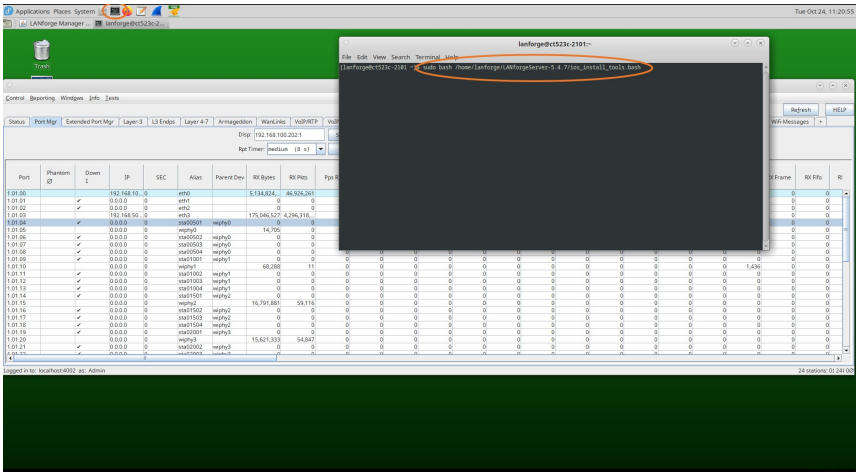
Before getting started:

Hardware Requirements:

- a Bluetooth 5.0 adapter (Edimax BT-8500 is known to work).
- an iOS device running iOS 16 or later.
- a USB cable for the iPhone device -- USB-A to Lightning/USB-C.
- a LANforge system with at least two free USB ports.

LANforge Machine Installs/Config

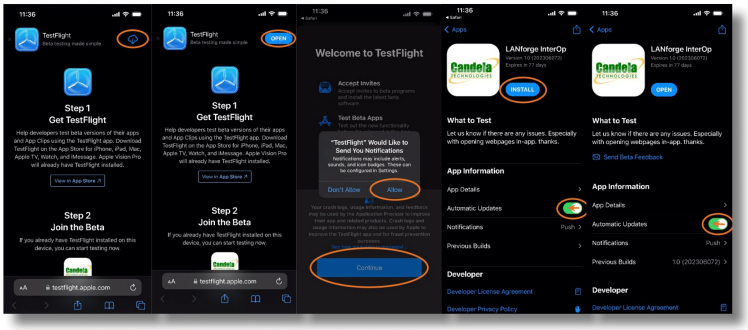
1. Simply open a terminal on your LANforge machine (CTRL + ALT + T) and run:
   `sudo bash /home/lanforge/ios_install_tools.bash`



iOS Device Initial Setup

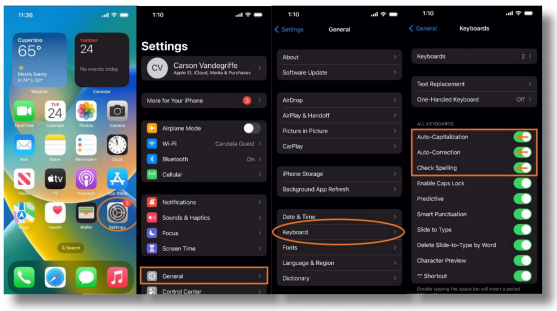
2. If using a new iPhone, ensure that the initial Apple setup is complete and that the device is logged into an Apple account. It's fine to log multiple devices into the same Apple account.
3. Visit the following link: iOS Interop App Install.

4. Download TestFlight, Open TestFlight, and install the LANforge Interop App. Be sure to disable Automatic Updates.

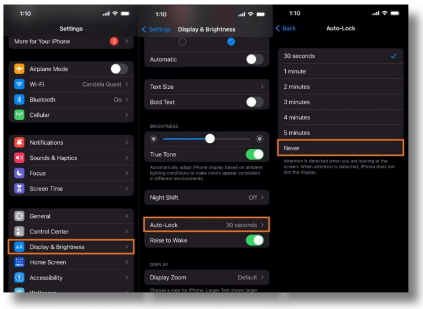


5. Open the LANforge App, select the 'Network' tab, and enable Location access.

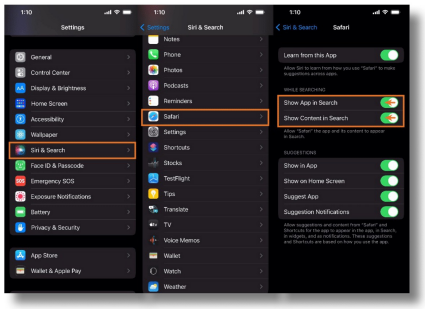
6. Open the Settings App on the iPhone.
    1. Disable Keyboard Autocapitalization and Autocorrection



    2. Set Display Auto-Lock to **Never**



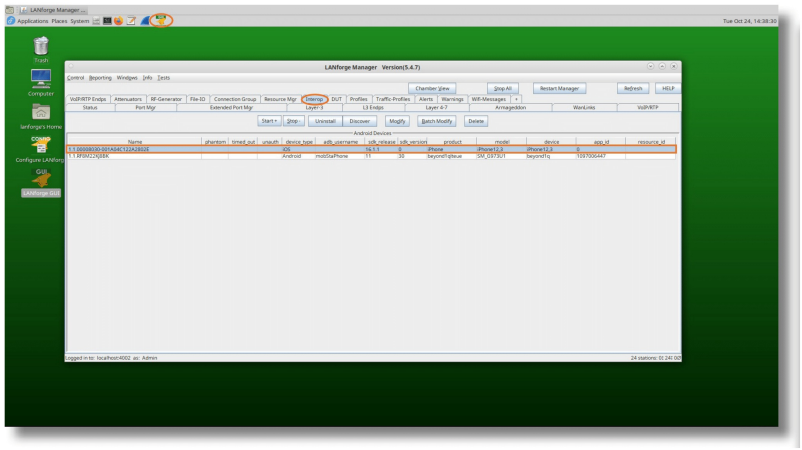
3. Disable Safari Suggestions in Search.



Connecting the iPhone to LANforge

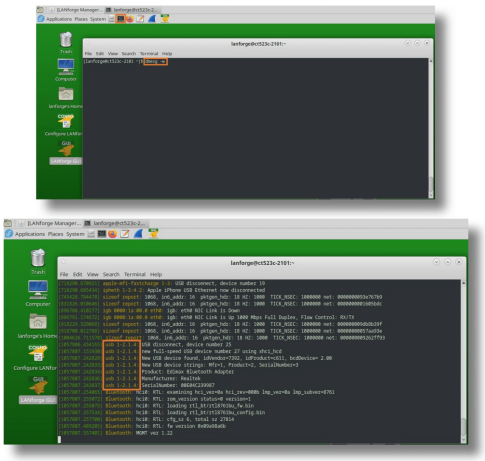
7. Connect the iPhone via its USB port (Lightning or USB-C) to one of the USB-A slots on the LANforge. (as per the testbed topology)
8. On the iPhone screen, when prompted about trusting the computer, select 'Trust'.

9. Open the LANforge GUI window. The Interop tab should appear now. Click on the tab to see your phone detected as an InterOp device.

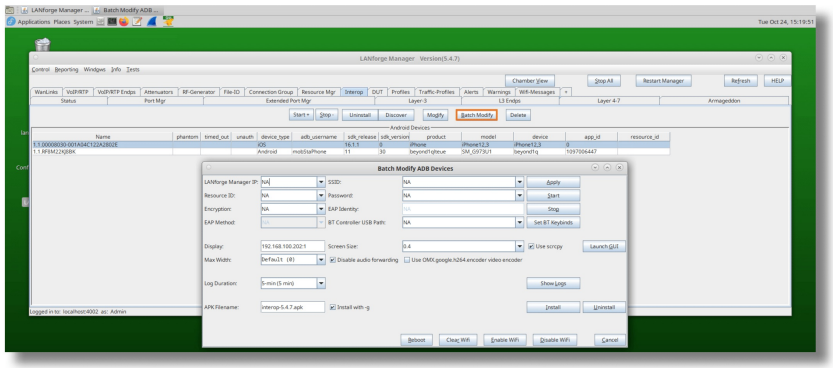


Connecting Bluetooth USB Adapter

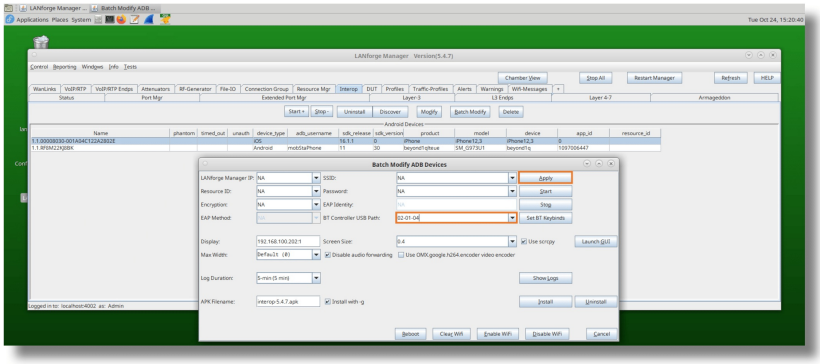
10. Open a terminal on the LANforge machine (CTRL + ALT + T). And run:
    `dmesg -w`
11. Plug your USB Bluetooth dongle into one of the USB ports on the LANforge.
12. The terminal window should show dmesg output acknowledging the new USB device.



13. The string 'usb 1-2.1.4' indicates that '2-1-4' is the USB-path of the adapter. Write this down.
14. On your LANforge machine, select the iPhone in the Interop tab and tap the button labeled Batch-Modify.

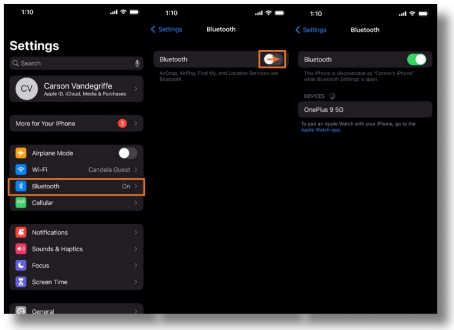


15. In the Batch-Modify window, enter the Bluetooth Controller's USB path (found from dmesg earlier). And click Apply to start the Bluetooth control process.

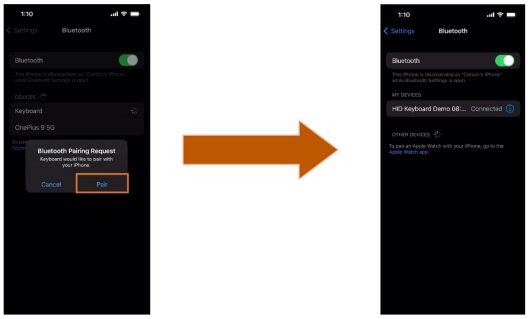


Setting up iOS Device Keyboard Control

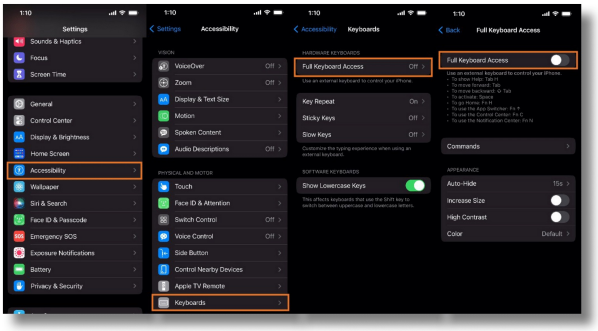
16. On the iPhone, open the Bluetooth settings page and enable Bluetooth.



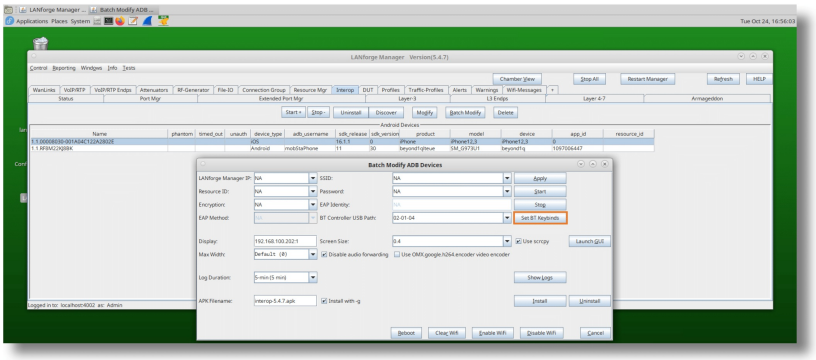
17. Keep watch on the iPhone's Bluetooth settings page and a connection prompt should appear. Select 'Pair' to pair the phone with your Bluetooth adapter.



18. On the iPhone, open the Keyboard Accessibility settings and enable 'Full Keyboard Access'.

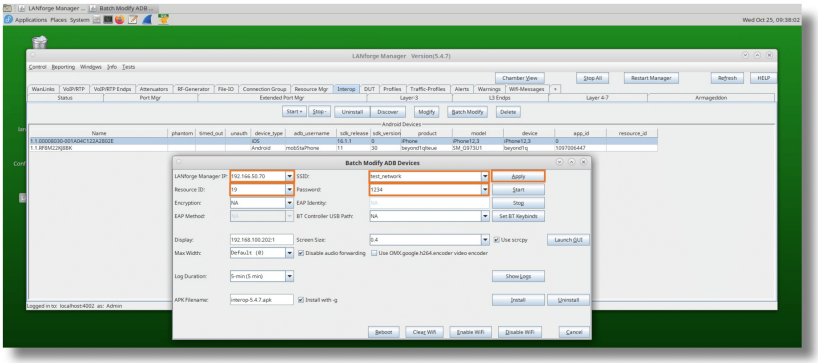


19. Back in the LANForge Batch-Modify window, tap the Set BT Keybinds button. Allow 5-10 minutes for this command to run and do not interfere with the process by touching the screen or power-cycling the phone. Once this command is finished, the phone will return to the home screen and appear idle again.



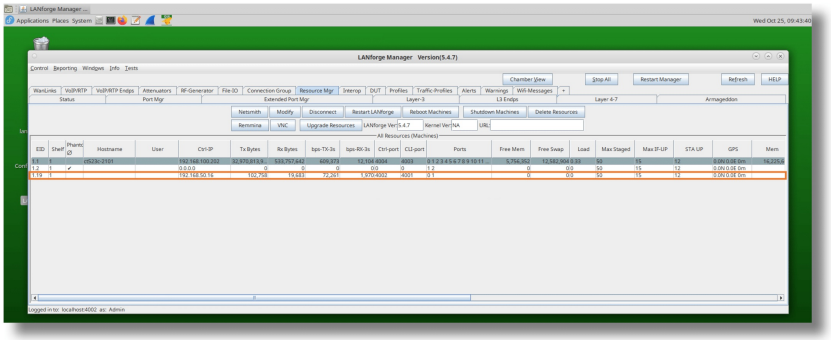
Starting the LANforge App

20. Set the values needed to connect the iPhone to LANforge: LANforge Manager IP and Resource ID (must be a Resource ID that's not in use). And if you wish to automatically connect the phone to a WiFi network, then enter the SSID and password. Click Apply when done populating fields. The app will restart and attempt to connect using the configured values.

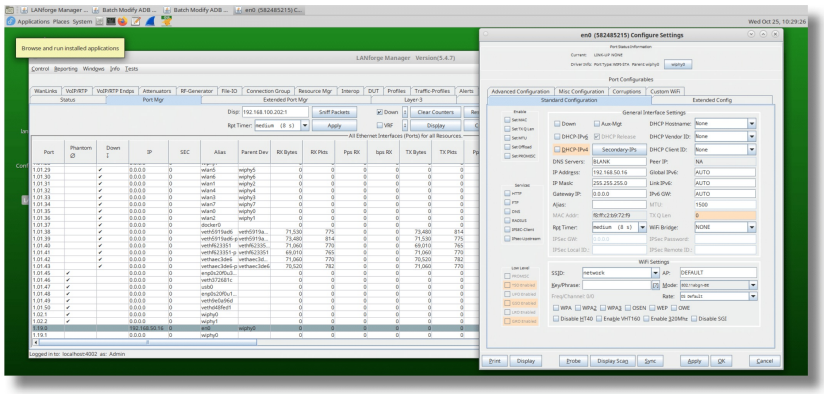


21. Wait for roughly a minute, and if this is your first time connecting the device to LANforge, you will need to accept the following authorization prompt. If you do not see this prompt appear, ensure that your configuration values are correct and that the device is connected to the correct network.

22. Open the Resource Mgr tab and the iPhone should appear under the requested resource ID.



23. Back in the Port Mgr tab, we can see the iPhone's ports, IP address, and MAC address.



24. Your iPhone should now be ready for use with LANforge testing.

Known Limitations:

1. Screen Mirroring iOS devices onto LANforge systems is not yet supported.
2. Automated installation and removal of the iOS Interop app through the Interop tab's Batch-Modify window is not yet supported.