

Getting Started with Python scripting for LANforge

Goal: After reading this, a user will know what python modules which are available to use LANforge.

There are many python modules which control LANforge. These modules allow the user to automate many tasks. This cookbook will introduce the initial steps to start LANforge automation, the libraries that need to be imported to run Candela's python scripts, and how to create objects in python.

There are two options to run lanforge_scripts. If you are a programmer, you can clone the git repository locally by using `git clone https://github.com/greearb/lanforge-scripts` in your command line into the directory which the folder should be in.

We also offer lanforge_scripts as a pip module. Simply type `pip3 install lanforge-scripts` into your terminal.

The advantage to cloning lanforge_scripts from git is that it is the latest code available, while the pip repository will lag slightly behind the git repo. The advantage to using pip is that there might be occasional errors in the git repository, which will be ironed out by the time we push it to the pip repository. The pip repository is always close to the master branch of the git repo, lagging no more than a couple weeks. Pick the one which best suits your needs.

i Please run Python as the **LANforge** user on your system.

If you run python as root, you might break your LANforge system. The LANforge Linux OS requires some certain python packages and versions to operate and configure networking correctly. Candelatech does not recommend updating python packages as root. Your package manager will automatically update these dependencies when you run updates as well, which can overwrite your changes.

The safe way to run python scripts is as a non-root user. If you have python dependencies installed locally by using `pip --user`, if a dependency breaks you can fix it without potentially harming files your operating system needs to operate. Broken python dependencies on your system Python can break your operating system. When your system updates, it might overwrite the changes you have made.

Users are responsible for making certain the version of Python running on their system is supported by the Python Foundation. An up to date list can be found at [Python Foundation support](#). Candela Technologies does not support versions of Python which have been deprecated by the Python Foundation. Customers with a support contract can contact support@candelatech.com to get support upgrading their LANforge systems.

1. Importing libraries:

Please visit the readme.md page in lanforge-scripts/py-scripts to make sure all the libraries that are needed are imported.

2. Install necessary dependencies

A. In the root of lanforge_scripts, run `pip3 install --user -r requirements.txt --upgrade`

3. How to run a script from a git clone

A. We start in the `/home/lanforge/lanforge-scripts/py-scripts` directory on your lanforge system

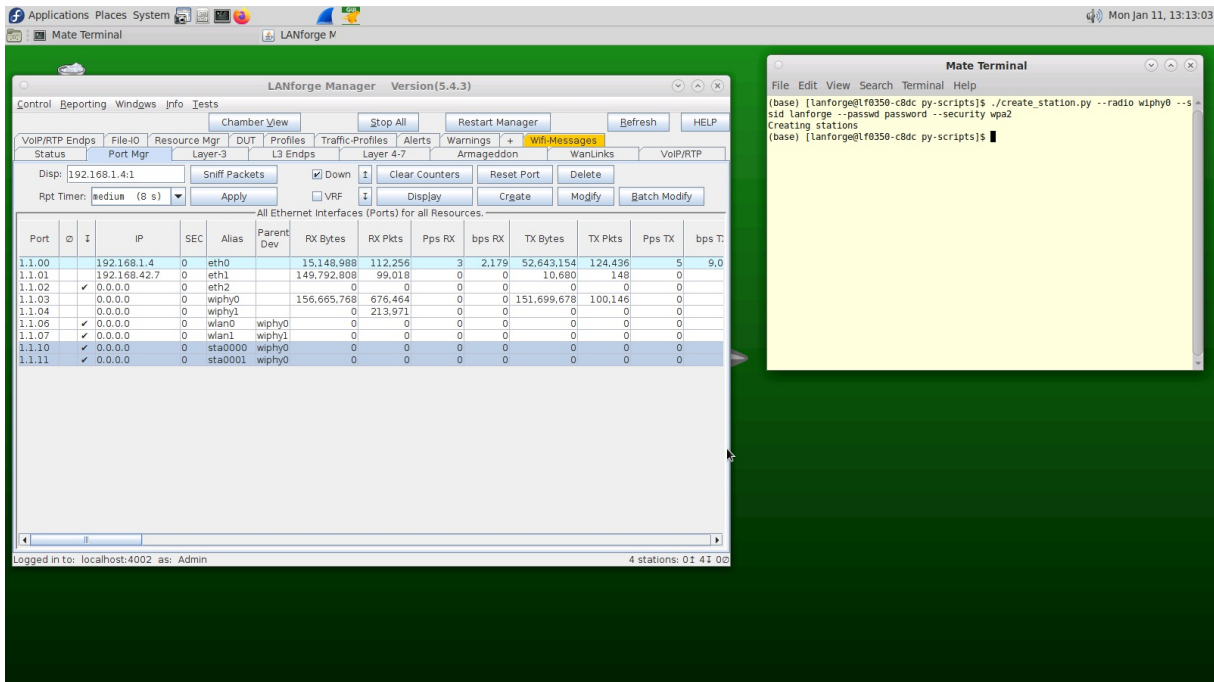
B. On a Linux system, open your terminal and navigate to the py-scripts folder. You can run any python script by typing

```
python3 your_script_here.py --your_flag
```

into your command line. After the script, put any flags you need in order to run your test.

4. How to run a script from pypi
 - A. In your terminal, run `python3`
 - B. In your python environment, run `import lanforge_scripts`
 - C. You can use `dir(lanforge_scripts)` to see all of the different classes you can use in `lanforge_scripts`. This is useful if you want to make your own driver script which uses `lanforge_scripts` as a dependency.

5. Station Creation



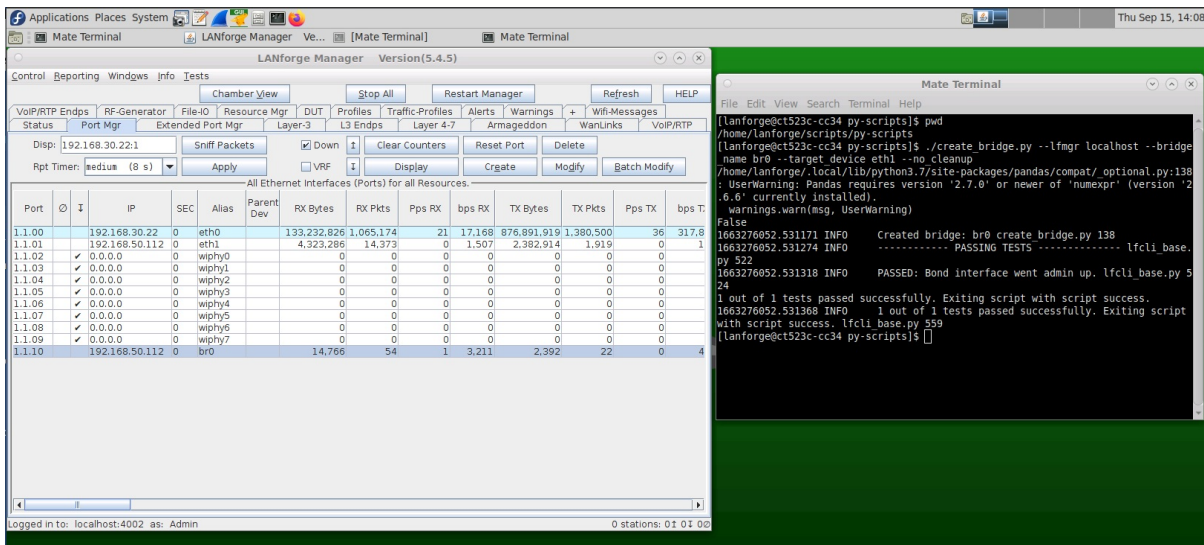
- A. The `create_station.py` script creates wi-fi stations from your terminal.
- B. Type `./create_station.py --radio wiphy0 --ssid lanforge --passwd password --security wpa2` into your terminal. Please remember to change the SSID, Passwd, and Security fields to match your network credentials.
- C. This will create 2 stations on your lanforge device off of your wiphy0 radio. You might need to wait 30 seconds for them to stop being phantom ports.
- D. You can specify the following flags as well:
 - A. `num_stations` - Specify a different number of stations to create off of your antenna. The default is 2.
 - B. `debug` - call this flag if you want detailed diagnostic information
6. Find available networks
 - A. You can find available WiFi networks on any Linux device by typing `sudo iw dev wlan0 scan | grep SSID` into your command line
 - B. You can also find available WiFi networks by clicking on the station you created in the previous step,
7. Associate to a specific BSSID:

If you want to connect to a specific MAC Address for your router (which is called the BSSID) you can specify that on each station, you can do that in both the GUI and in the command line.

- A. Connect via Command line
 - A. Each of your scripts has an optional AP tag where you can define the MAC address of the router you want to connect to. To do so, simply append `--ap` to the end of the command line argument you are running followed by the router's MAC Address.
 - B. Connect via GUI

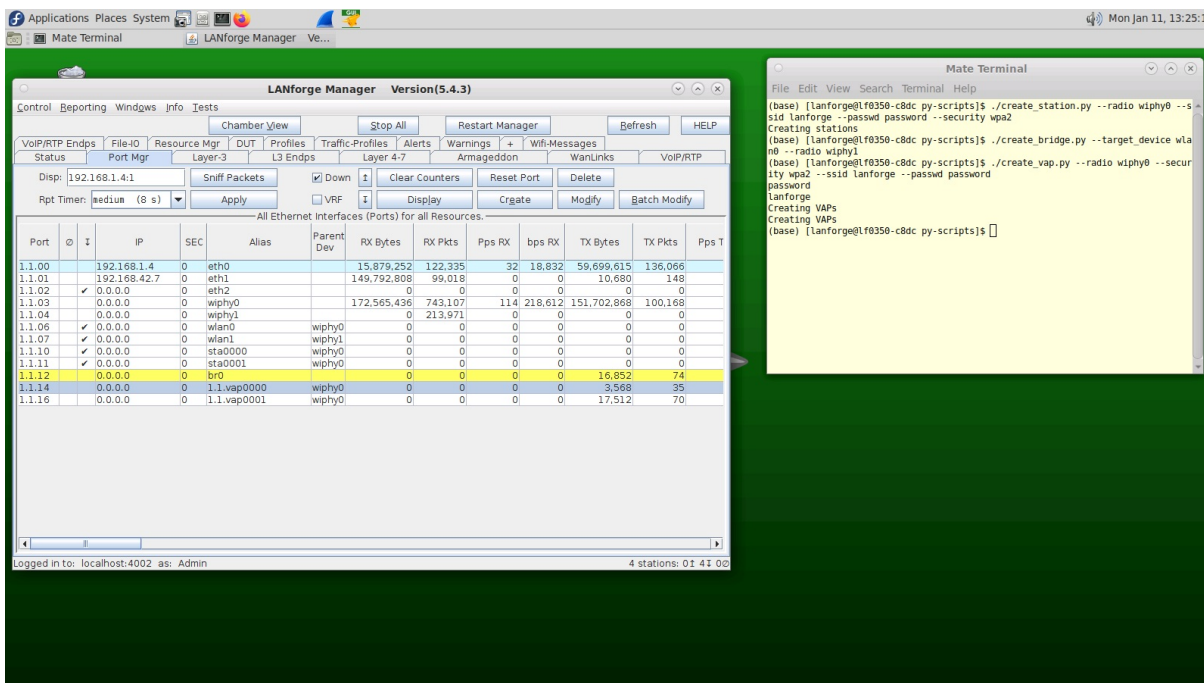
- A. When you double click on a Station there is an AP field inside WiFi Settings. Type the MAC Address of your router in that box and everything will work assuming the MAC Address is correct.

8. Bridge Creation



- A. To create a bridge, you can use the `create_bridge.py` script in the `py_scripts` folder.
- B. Type `./create_bridge.py --lf_mgr localhost --bridge_name br0 --target_device eth1 --no_cleanup`
 Into your terminal, remembering to change the `ssid`, `passwd`, and security fields to match your network credentials.
- C. `create_bridge` requires the following arguments:
- `bridge_name` - Name of the bridge to create
 - `target_device` - which device the bridge is going to connect
- D. It is not valid to add stations to a bridge, they don't work like you would expect. Bridges can have: eth ports, redirects (`rdd`) ports, vaps, and qvlan ports. Ports in a bridge cannot have IP addresses.

9. VAP Creation



- A. You can create a VAP from your terminal with the `create_vap.py` script.
- B. Type `./create_vap.py --radio wiphy0 --security wpa2 --ssid lanforge --passwd password`
 into your terminal, remembering to change the `ssid`, `passwd`, and security fields to match your network credentials.
- C. `create_vap` supports the following flags:

- A. num_vaps - A user defined number of VAPs to create off of your antenna
- B. upstream_port - if your ethernet cable to your router is not eth1, define it using this flag. Eth1 is the default for this flag.
- C. debug - call this flag if you want detailed diagnostic information

10. MAC-VLAN Creation

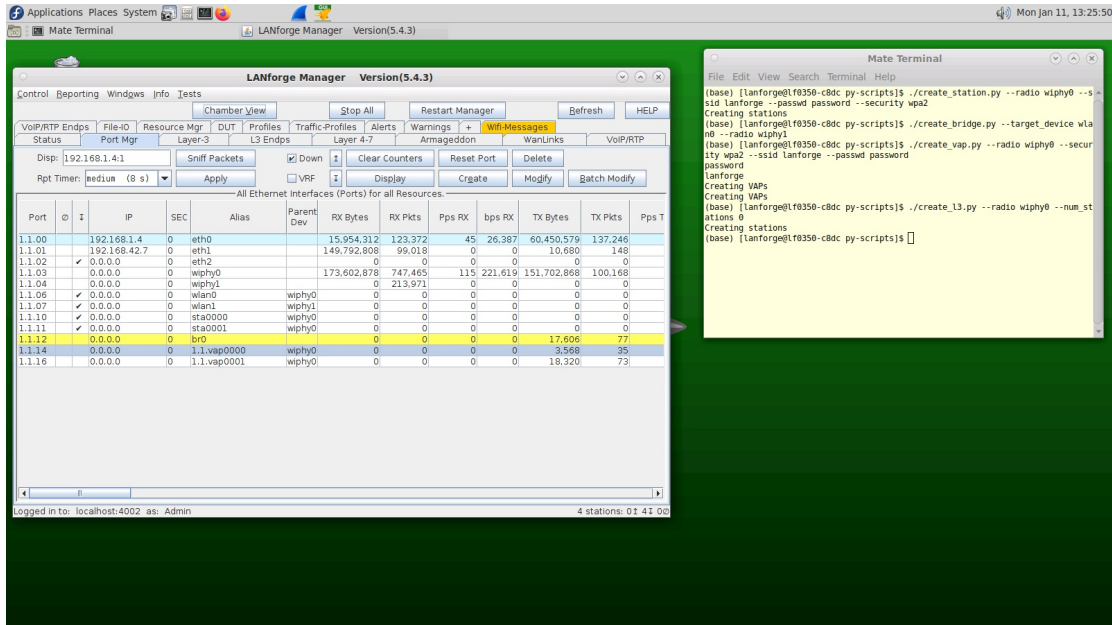
- A. In the py-scripts folder, there is a script named create_macvlan.py, which creates a mac-vlan based on the macvlan_parent , num_ports , first_macvlan_ip, netmask, and gateway input

11. Monitor Creation: Under Construction

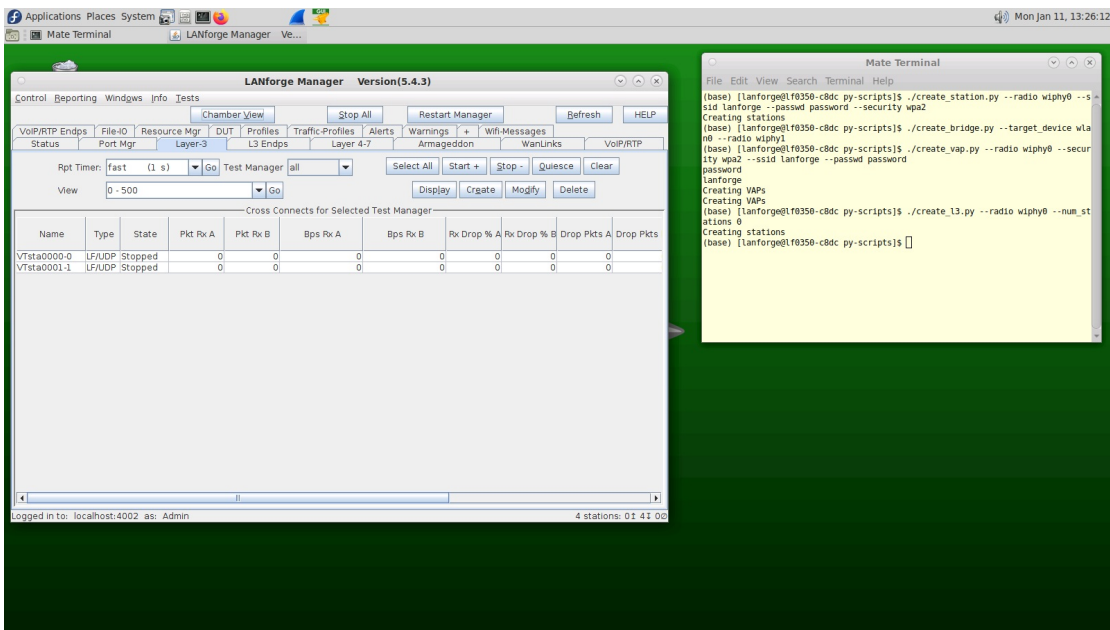
12. Layer-3 Cross Connect

- A. In your py-scripts, there is a create_l3 script which allows you to create stations in your terminal.
- B. Type `./create_l3.py --radio wiphy0 --num_stations 0` into your terminal, remembering to change the ssid, passwd, and security fields to match your network credentials.
- C. This will create a layer3 cross connect between each station you already built on your lanforge device and your wiphy0 radio. You might need to wait 30 seconds for them to stop being phantom ports. If you do not specify num_stations 0 it would have created two stations by default off your specified radio. You can change which port the cross connect will be connected to with the --upstream_port option.
- D. You can specify the following flags as well:
 - A. num_stations - Specify a different number of stations to create off of your antenna.
 - B. upstream_port - if your ethernet cable to your router is not eth1, define it using this flag. Eth1 is the default for this flag.
 - C. debug - call this flag if you want to get error messages in case anything goes wrong.

E. Your port manager will look similar to this

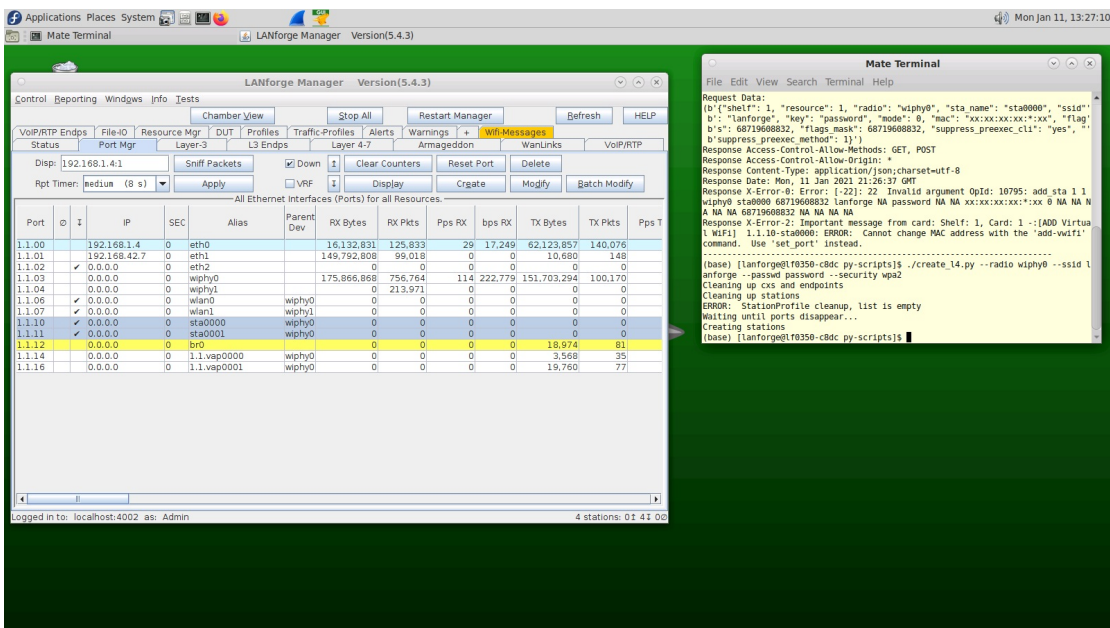


F. Your Layer 3 connections will look similar to this

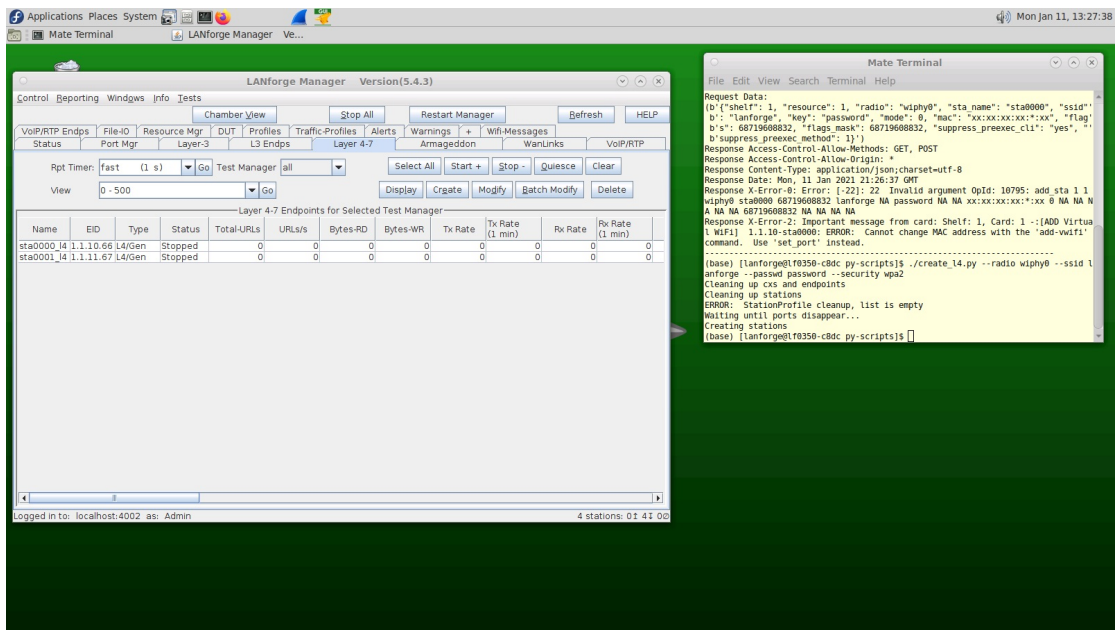


13. Layer-4 Cross Connect

- A. In your py-scripts, there is a create_l4 script which allows you to create stations in your terminal.
- B. Type `./create_l4.py --radio wiphy0 --ssid lanforge --passwd password --security wpa2` into your terminal
- C. This will automatically create 2 stations on your Lanforge device off of your wiphy0 radio and also create a cross connect from each station to your eth1 port. You can change which port the cross connect will be connected to with the `--upstream_port` option.
- D. You can specify the following flags as well:
 - A. `num_stations` - Specify a different number of stations to create off of your antenna.
 - B. `upstream_port` - if your Ethernet cable to your router is not eth1, define it using this flag. Eth1 is the default for this flag.
 - C. `debug` - call this flag if you want detailed diagnostic information
- E. Your port manager will look similar to this



- F. Your Layer 4 connections will look similar to this



G. Your Netsmith display will look similar to this. The connections on this picture have been oriented for legibility.

14. Monitor and record an IPV4 variable time test. The purpose of this test is to detect whether your router is able to keep a steady signal when being barraged by multiple users.

A. In the first part of this tutorial, you are going to connect various numbers of stations, record them for 1 and 10 minutes, and then save as a CSV, excel, or pickle format. A station represents a device which is connected to a network, LANforge creates representations of stations which create real traffic on your network and then records statistics on that traffic. This module determines whether your device is able to sustain a heavy load of traffic for an user defined period of time. By recording the traffic it is then possible to go back and detect where any problems occurred which allows a network manager to fix problems which could be facing your network.

B. Navigate to the py-scripts folder and type the following command into your command line
`./test_ipv4_variable_time.py --radio wiphy0 --security wpa2 --ssid lanforge --password password --output_format csv`

Replace the security, ssid, and password variables with the settings for the network you are testing. This will create 2 wiphy stations by default, connect them to the network you are testing, and report the results to a CSV file. You will find a file with a timestamp within the last 5 minutes in the report_data folder in your home directory. If you are running this script from another machine using the --mgr function, you will need to define the report_file variable.

- C. test_ipv4_variable_time accepts the following flags:
- A. output_format - The format you want to save your results to
 - B. col_names - Which columns should be saved in the output file
 - C. test_duration - how long you want the test to last.
 - D. report_file - where you want the results to be stored

D. Your port manager will look similar to this

Applications Places System Fri Jan 8, 12:00:54

Mate Terminal LANforge Manager... [report-data] Take Screenshot

LANforge Manager Versio...

Control Reporting Windows Info Tests

Chamber View Stop All Restart Manager Refresh HELP

Status Port Mgr Layer-3 L3 Endps Layer 4-7 Armageddon WanLinks VoIP/RTP VoIP/RTP Endps File-ID Resource Mgr DUT Profiles Traffic-Profiles Alerts Warnings + Wifi-Messages

Disp: 192.168.1.41 Sniff Packets VRF Clear Counters Reset Port Delete

Rpt Timer: medium (8 s) Apply VRF Display Create Modify Batch M

All Ethernet Interfaces (Ports) for all Resources

Port	IP	SEC	Alias	Parent Dev	RX Bytes	RX Pkts	Pps RX	bps RX	TX
1.1.00	192.168.1.4	0	eth0		13,650,235	175,885	22	13,263	128.1
1.1.01	192.168.42.7	0	eth1		16,012,862	10,649	0	0	0
1.1.02	0.0.0.0	0	eth2		0	0	0	0	0
1.1.03	0.0.0.0	0	wiphy0		284,765,280	1,433,248	0	1,057	16.1
1.1.04	0.0.0.0	0	wiphy1		1,579,923...	6,981,005	0	0	0
1.1.05	172.16.42.142	0	sta0000	wiphy0	2,090	8	0	1,119	
1.1.07	0.0.0.0	0	wlan0	wiphy0	0	0	0	0	0
1.1.09	0.0.0.0	0	wlan1	wiphy1	0	0	0	0	0
1.1.11	172.16.42.52	0	sta0001	wiphy0	2,266	10	0	1,228	

File Edit View Search Terminal Help

```
(base) [lanforge@l0350-c8dc py-scripts]$ ./test_ipv4_variable_time.py --radio wiphy0
--security wpa2 --ssid lanforge --password password --output_format csv
Creating stations
Waiting for ips, timeout: 60...
Starting Cx...
```

Logged in to: localhost:4002 as: Admin 4 stations: 21 21 00

E. Your Layer 3 connections will look similar to this

Applications Places System Fri Jan 8, 12:08:01

Mate Terminal LANforge Manager... [report-data] Take Screenshot

LANforge Manager Version(5.4.3)

Control Reporting Windows Info Tests

Chamber View Stop All Restart Manager Refresh HELP

Status Port Mgr Layer-3 L3 Endps Layer 4-7 Armageddon WanLinks VoIP/RTP VoIP/RTP Endps File-ID Resource Mgr DUT Profiles Traffic-Profiles Alerts Warnings + Wifi-Messages

Rpt Timer: fast (1 s) Go Test Manager all Select All Start + Stop - Quiesce Clear

View: 0 - 500 Go Display Create Modify Delete

Cross Connects for Selected Test Manager

Name	Type	State	Pkt Rx A	Pkt Rx B	Bps Rx A	Bps Rx B	Rx Drop % A	Rx Drop % B	Drop Pkts A	Drop Pkts B	Avg RTT	Rpt Timer	EID	Endpoints (A ↔ B)
Vfsta0000-0	LF/UDP	Run	0	3,834	0	255,985	0	0	0	0	0	5000.144		Vfsta0000-0-A ↔ Vfsta0000-0-B
Vfsta0001-1	LF/UDP	Run	0	3,831	0	255,857	0	0	0	0	0	5000.145		Vfsta0001-1-A ↔ Vfsta0001-1-B

Logged in to: localhost:4002 as: Admin 4 stations: 21 21 00

F. Your Layer 3 endpoints will look similar to this

Applications Places System Fri Jan 8, 12:08:12

Mate Terminal LANforge Manager Versi... [report-data]

LANforge Manager Version(5.4.3)

Control Reporting Windows Info Tests

Chamber View Stop All Restart Manager Refresh HELP

Status Port Mgr Layer-3 L3 Endps Layer 4-7 Armageddon WanLinks VoIP/RTP VoIP/RTP Endps File-ID Resource Mgr DUT Profiles Traffic-Profiles Alerts Warnings + Wifi-Messages

Min PDU Size: AUTO Max PDU Size: Same Start + Stop - Quiesce Clear

MIN Tx Rate: Max Rate: 150 Mbps MAX Tx Rate: Same Display Create Modify Batch Modify Delete

View: 0 - 1000 Go

All Endpoints

Name	EID	Run	Mng	Script	Tx Rate	Rx Rate (1 min)	Tx Rate (last)	Tx Rate LL	Rx Rate	Rx Rate (1 min)	Rx Rate (last)	Rx Rate LL	Rx Drop %	Tx PDUs	Tx Pkts LL	PDU/s TX	Pps TX LL	Rx
Vfsta0000-0-A	1.1.5.62	✓	None		255,989	256,051	256,460	262,952	0	0	0	0	0	4,091	4,091	21	21	
Vfsta0000-0-B	1.1.11.63	✓	None		0	0	0	0	255,930	255,922	254,603	263,224	0	0	0	0	0	
Vfsta0001-1-A	1.1.11.64	✓	None		255,872	255,922	254,752	263,128	0	0	0	0	0	4,060	4,060	21	21	
Vfsta0001-1-B	1.1.1.65	✓	None		0	0	0	0	255,872	255,922	254,752	263,224	0	0	0	0	0	

Logged in to: localhost:4002 as: Admin 4 stations: 21 21 00

G. You can view a dynamic report of the connection creating traffic. It will look similar to this

