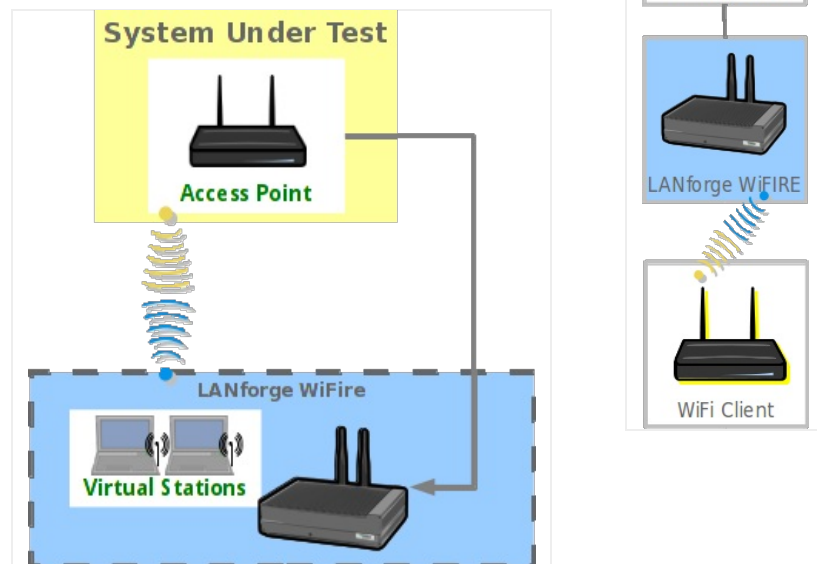


Using LANforge python script to do packet captures of OFDMA and regular WiFi Stations

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..

```

dt
File Edit View Search Terminal Help
[greearb@ben-dt4 lf_scripts]$ ./lf_sniff.py --help
usage: lf_sniff.py [-h] [--sniffer_radios SNIFFER_RADIOS] [--station STATION]
                  [--lfmgr LFMGR] [--duration DURATION]
                  [--moni_flags MONI_FLAGS]

Sniffer control Script

optional arguments:
  -h, --help                show this help message and exit
  --sniffer_radios SNIFFER_RADIOS
                            LANforge sniffer radios to use (2.wiphy0 2.wiphy1 ...)
  --station STATION         LANforge stations to use (1.wlan0 1.wlan1 etc)
  --lfmgr LFMGR             LANforge Manager IP address
  --duration DURATION       Duration to sniff, in minutes
  --moni_flags MONI_FLAGS
                            Monitor port flags, see LANforge CLI help for
                            set wifi_monitor. Default enables 160Mhz
[greearb@ben-dt4 lf_scripts]$

```

- 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 --lmgrr 192.168.100.238 --station "1.sta00000" --sniffer_radios "2.wiphy0" --duration 0.25
```