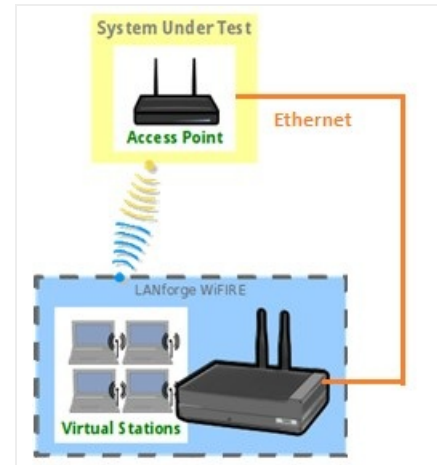


Automate WiFi Capacity and other GUI tests.

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.

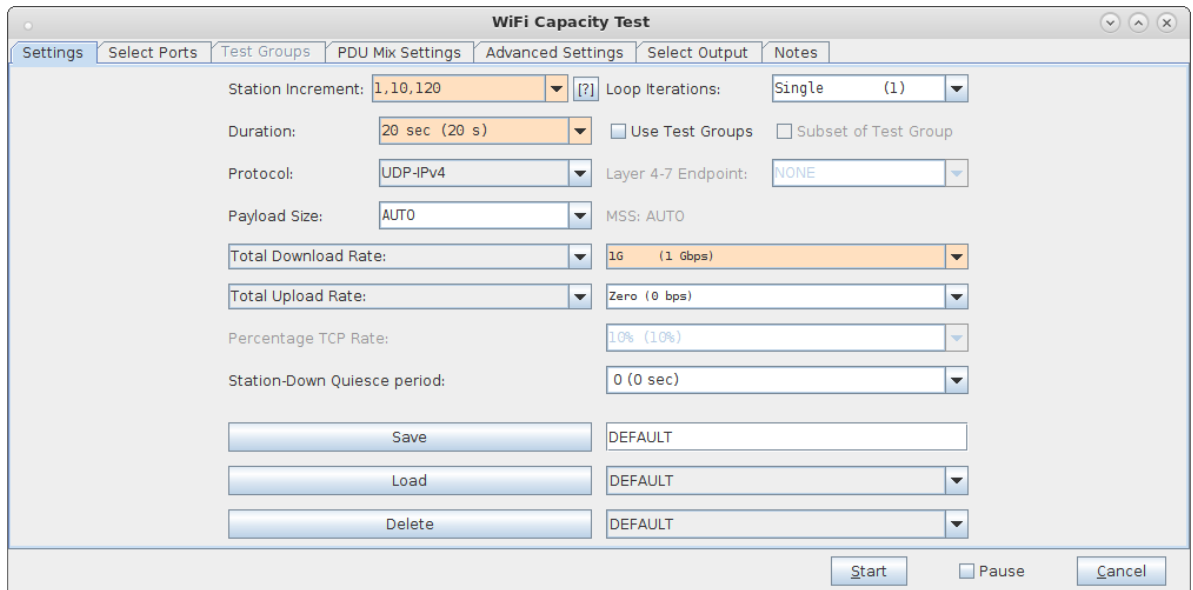


1. Configure WiFi Capacity Test for automated run.

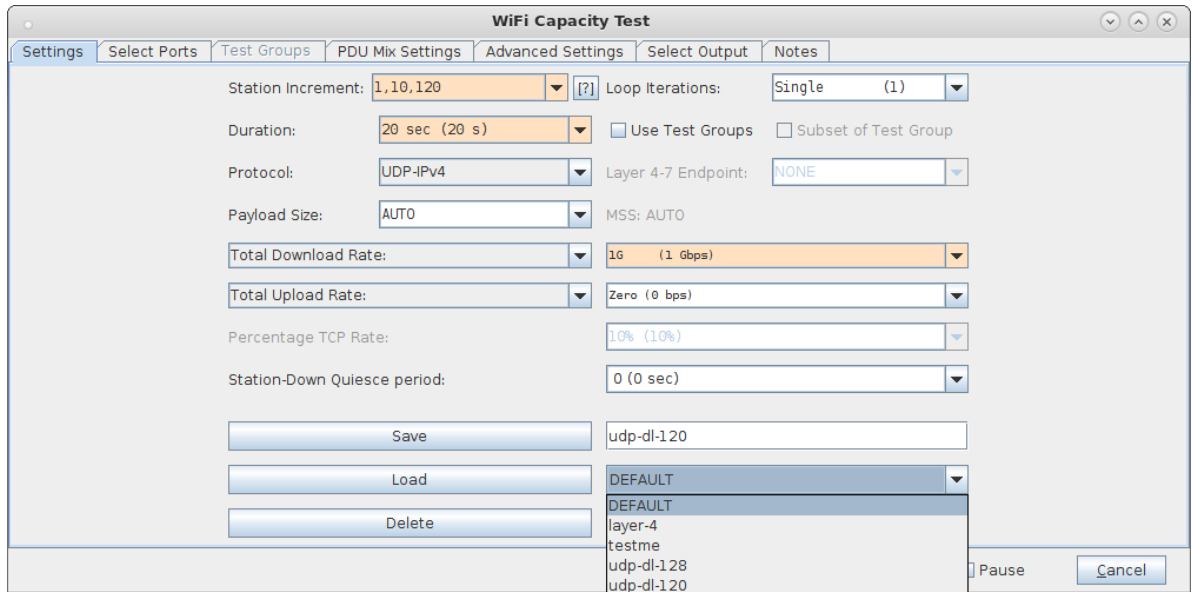
- A. For this to work, the LANforge GUI must be started with the **-cli-socket 3990** argument. This causes it to open a socket to listen for text commands.
- B. Open Chamber View by clicking on the 'Chamber View' button in the LANforge-GUI. Create an appropriate scenario and DUT if you have not already done so. Other cookbook examples have more details of how to do this, please see those if you are unfamiliar with Chamber View.

The screenshot shows the 'Chamber View' window with a 'Scenario Configuration' diagram. The diagram includes a 'comxim-1-4' node connected to 'jw3', which is connected to 'LF-1'. 'LF-1' is connected to 'upstream', which is connected to two 'STA-AC(64) -31 db' nodes, which are connected to two 'tcp-dl-6m-vi' nodes. The interface also features a right-hand sidebar with 'Manage Scenarios', 'If-to-jw3', 'Apply Scenario', 'Tests: WiFi Capacity', 'Run Test', and 'Snap Report' buttons. At the bottom, there are various checkboxes for display options and buttons for 'Info', 'Print', 'Sync', 'Apply', 'Build Scenario', and 'Close'.

C. Select WiFi Capacity test, and click **Run Test** to configure it as desired.

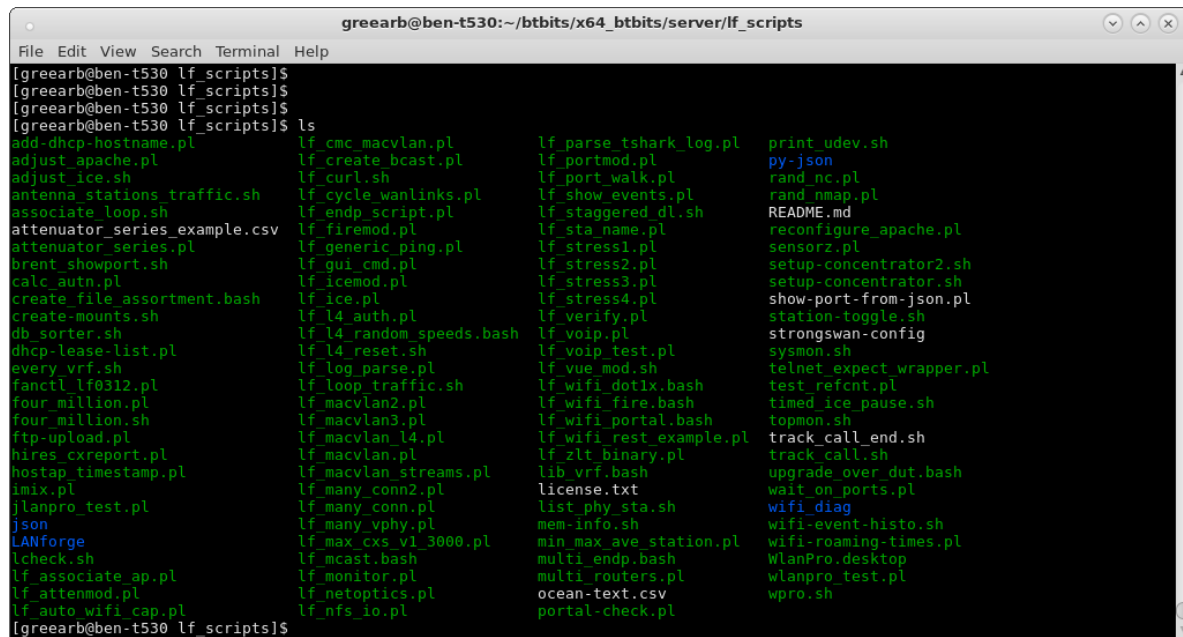


D. Enter a name in the 'Save' field, click save, and make sure it shows up as a loadable configuration. In this case, we are saving the configuration as 'udp-dl-120'



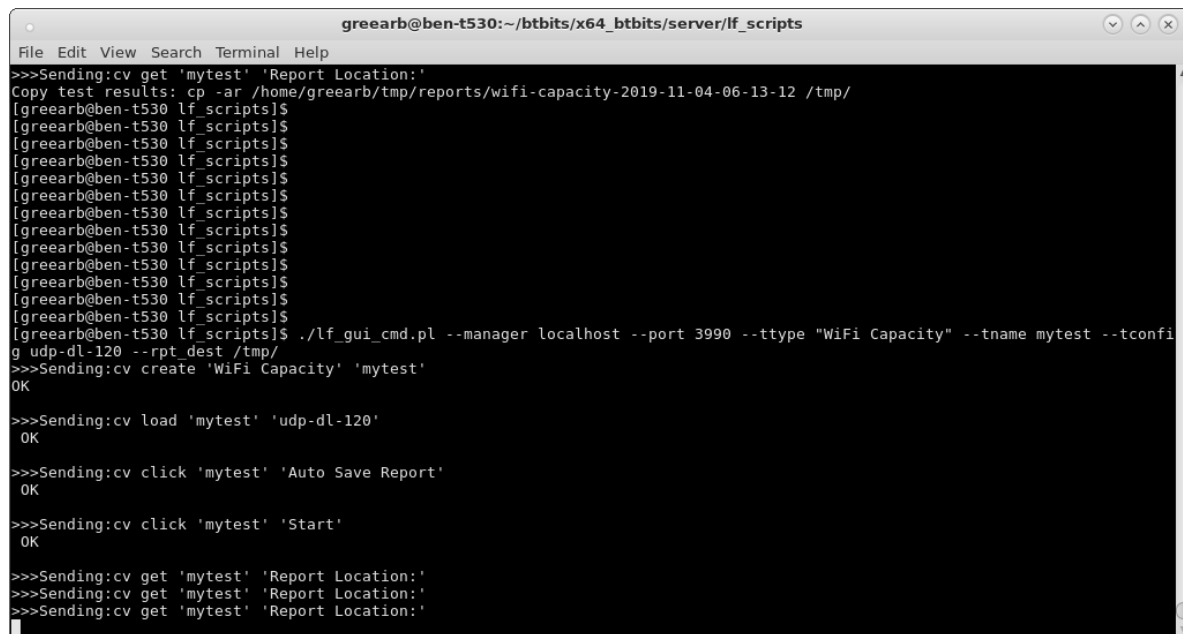
2. Use the `lf_gui_cmd.pl` script to launch the WiFi Capacity Test.

- 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



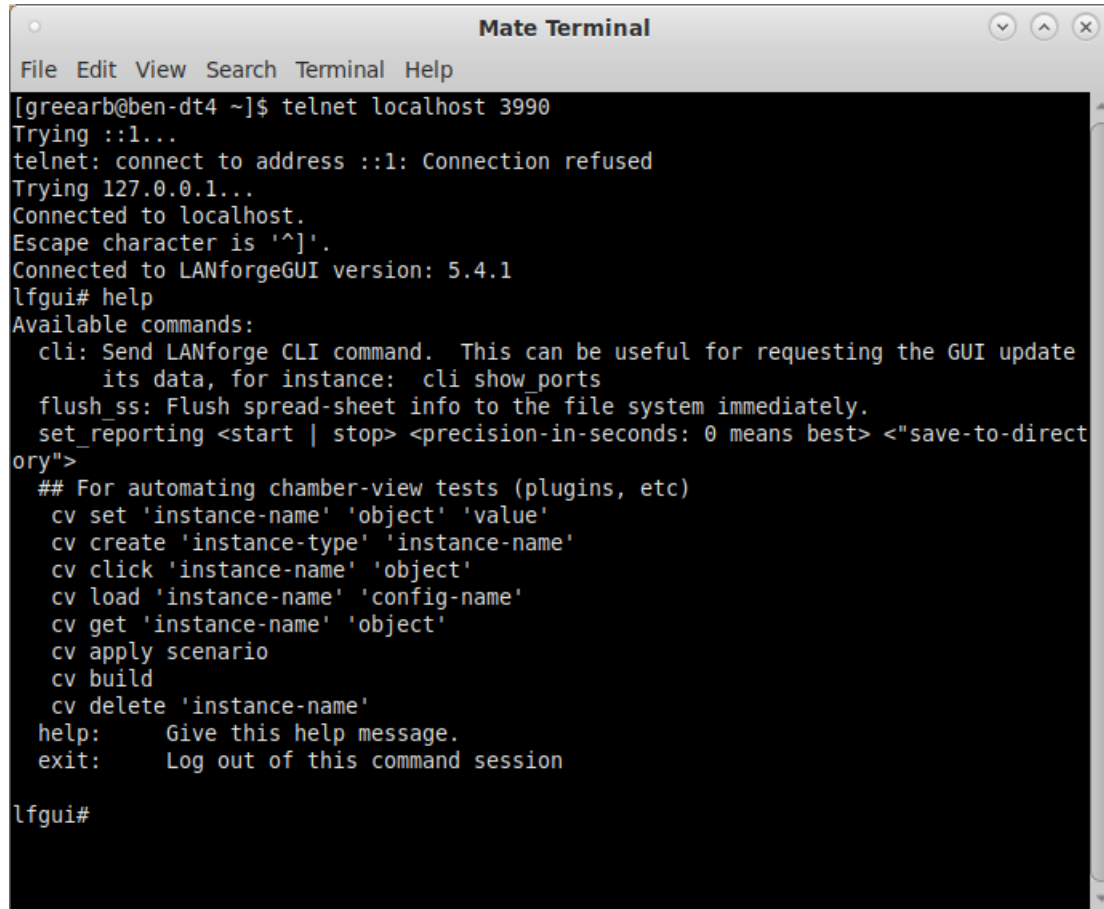
```
greearb@ben-t530:~/btbits/x64_btbits/server/lf_scripts
File Edit View Search Terminal Help
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$ ls
add-dhcp-hostname.pl      lf_cmc_macvlan.pl          lf_parse_tshark_log.pl    print_udev.sh
adjust_apache.pl          lf_create_bcast.pl        lf_portmod.pl             py-json
adjust_ice.sh              lf_curl.sh                 lf_port_walk.pl          rand_nc.pl
antenna_stations_traffic.sh lf_cycle_wanlinks.pl      lf_show_events.pl        rand_nmap.pl
associate_loop.sh         lf_endp_script.pl         lf_staggered_dl.sh       README.md
attenuator_series_example.csv lf_firemod.pl             lf_sta_name.pl           reconfigure_apache.pl
attenuator_series.pl      lf_generic_ping.pl        lf_stress1.pl            sensorz.pl
brent_showport.sh        lf_gui_cmd.pl             lf_stress2.pl            setup-concentrator2.sh
calc_autn.pl              lf_icemod.pl              lf_stress3.pl            setup-concentrator.sh
create_file_assortment.bash lf_ice.pl                  lf_stress4.pl            show-port-from-json.pl
create-mounts.sh          lf_l4_auth.pl             lf_verify.pl             station-toggle.sh
db_sorter.sh              lf_l4_random_speeds.bash lf_voip.pl                strongswan-config
dhcp-lease-list.pl       lf_l4_reset.sh            lf_voip_test.pl          sysmon.sh
every_vrf.sh              lf_log_parse.pl           lf_vue_mod.sh            telnet_expect_wrapper.pl
fanctl_lf0312.pl          lf_loop_traffic.sh        lf_wifi_dot1x.bash       test_refcnt.pl
four_million.pl           lf_macvlan2.pl            lf_wifi_fire.bash        timed_ice_pause.sh
four_million.sh           lf_macvlan3.pl            lf_wifi_portal.bash      topmon.sh
ftp-upload.pl             lf_macvlan_l4.pl          lf_wifi_rest_example.pl  track_call_end.sh
hires_cxreport.pl         lf_macvlan.pl             lf_zlt_binary.pl         track_call.sh
hostap_timestamp.pl       lf_macvlan_streams.pl    lib_vrf.bash              upgrade_over_dut.bash
imix.pl                   lf_many_conn2.pl          license.txt                wait_on_ports.pl
jlanpro_test.pl           lf_many_conn.pl           list_phy_sta.sh           wifi_diag
json                       lf_many_vphy.pl           mem-info.sh                wifi-event-histo.sh
LANforge                  lf_max_cxs_v1_3000.pl     min_max_ave_station.pl   wifi-roaming-times.pl
lcheck.sh                 lf_mcast.bash             multi_endp.bash           wlanPro.desktop
lf_associate_ap.pl        lf_monitor.pl             multi_routers.pl          wlanpro_test.pl
lf_attenmod.pl            lf_netoptics.pl           ocean-text.csv             wpro.sh
lf_auto_wifi_cap.pl       lf_nfs_io.pl              portal-check.pl
[greearb@ben-t530 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.



```
greearb@ben-t530:~/btbits/x64_btbits/server/lf_scripts
File Edit View Search Terminal Help
>>>Sending:cv get 'mytest' 'Report Location:'
Copy test results: cp -ar /home/greearb/tmp/reports/wifi-capacity-2019-11-04-06-13-12 /tmp/
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$
[greearb@ben-t530 lf_scripts]$ ./lf_gui_cli_cmd.pl --manager localhost --port 3990 --ttype "WiFi Capacity" --tname mytest --tconfi
g udp-dl-120 --rpt_dest /tmp/
>>>Sending:cv create 'WiFi Capacity' 'mytest'
OK
>>>Sending:cv load 'mytest' 'udp-dl-120'
OK
>>>Sending:cv click 'mytest' 'Auto Save Report'
OK
>>>Sending:cv click 'mytest' 'Start'
OK
>>>Sending:cv get 'mytest' 'Report Location:'
>>>Sending:cv get 'mytest' 'Report Location:'
>>>Sending:cv get 'mytest' 'Report Location:'
```

- 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.



```
Mate Terminal
File Edit View Search Terminal Help
[greearb@ben-dt4 ~]$ telnet localhost 3990
Trying ::1...
telnet: connect to address ::1: Connection refused
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Connected to LANforgeGUI version: 5.4.1
lfgui# help
Available commands:
  cli: Send LANforge CLI command. This can be useful for requesting the GUI update
      its data, for instance: cli show ports
  flush_ss: Flush spread-sheet info to the file system immediately.
  set_reporting <start | stop> <precision-in-seconds: 0 means best> <"save-to-direct
ory">
## For automating chamber-view tests (plugins, etc)
  cv set 'instance-name' 'object' 'value'
  cv create 'instance-type' 'instance-name'
  cv click 'instance-name' 'object'
  cv load 'instance-name' 'config-name'
  cv get 'instance-name' 'object'
  cv apply scenario
  cv build
  cv delete 'instance-name'
help:      Give this help message.
exit:     Log out of this command session

lfgui#
```