Candela TECHNOLOGIES Network Testing and Emulation Solutions

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

LANforge Scripts Users Guide

Script Table of Contents

Script Information

py-scripts/test_l3.py

```
usage: test_l3.py [-h] [--local_lf_report_dir LOCAL_LF_REPORT_DIR]
                  [--results_dir_name RESULTS_DIR_NAME] [--test_rig TEST_RIG]
                  [--test_tag TEST_TAG] [--dut_hw_version DUT_HW_VERSION]
                  [--dut_sw_version DUT_SW_VERSION]
                  [--dut_model_num DUT_MODEL_NUM]
                  [--dut_serial_num DUT_SERIAL_NUM]
                  [--test_priority TEST_PRIORITY] [--test_id TEST_ID]
                  [-o CSV_OUTFILE] [--tty TTY] [--baud BAUD] [--mgr LFMGR]
                  [--mgr_port LFMGR_PORT] [--test_duration TEST_DURATION]
                  [--tos TOS] [--debug] [--log_level LOG_LEVEL] [-t ENDP_TYPE]
                  [-u UPSTREAM_PORT] [--downstream_port DOWNSTREAM_PORT]
                  [--polling_interval POLLING_INTERVAL] [-r RADIO]
                  [-amr SIDE_A_MIN_BPS] [-amp SIDE_A_MIN_PDU]
                  [-bmr SIDE_B_MIN_BPS] [-bmp SIDE_B_MIN_PDU]
                  [--rates_are_totals] [--multiconn MULTICONN]
                  [--attenuators ATTENUATORS] [--atten_vals ATTEN_VALS]
                  [--wait WAIT] [--sta_start_offset STA_START_OFFSET]
                  [--no_pre_cleanup] [--no_cleanup] [--cleanup_cx]
                  [--csv_data_to_report] [--no_stop_traffic] [--quiesce_cx]
                  [--use_existing_station_list]
                  [--existing_station_list EXISTING_STATION_LIST]
                  [--wait_for_ip_sec WAIT_FOR_IP_SEC]
                  [--lf_logger_config_json LF_LOGGER_CONFIG_JSON] [--ap_read]
                  [--ap_module AP_MODULE] [--ap_test_mode AP_TEST_MODE]
                  [--ap_scheme {serial,telnet,ssh,mux_serial}]
                  [--ap_serial_port AP_SERIAL_PORT]
                  [--ap_serial_baud AP_SERIAL_BAUD] [--ap_ip AP_IP]
                  [--ap_ssh_port AP_SSH_PORT]
                  [--ap_telnet_port AP_TELNET_PORT] [--ap_user AP_USER]
                  [--ap_passwd AP_PASSWD] [--ap_if_2g AP_IF_2G]
                  [--ap_if_5g AP_IF_5G] [--ap_if_6g AP_IF_6G]
                  [--ap_file AP_FILE] [--ap_band_list AP_BAND_LIST]

NAME: test_l3.py

PURPOSE: The Layer 3 Traffic Generation Test is designed to test the performance of the Access Point by running layer-3
         Cross-Connect Traffic.  Layer-3 Cross-Connects represent a stream of data flowing through the system under test.
         A Cross-Connect (CX) is composed of two Endpoints, each of which is associated with a particular Port (physical or virtual ir

         The test will create stations, create cx traffic between upstream port and stations,  run traffic. Verify
         the traffic is being transmitted and received

         * Supports creating user-specified amount stations on multiple radios
         * Supports configuring upload and download requested rates and PDU sizes.
         * Supports generating connections with different ToS values.
         * Supports generating tcp and/or UDP traffic types.
         * Supports iterating over different PDU sizes
         * Supports iterating over different requested tx rates (configurable as total or per-connection value)
         * Supports iterating over attenuation values.
         * Supports testing connection between two ethernet connection - L3 dataplane

         Generic command layout:
         -----------------------
         ./test_l3.py --mgr <ip_address> --test_duration <duration> --endp_type <traffic types> --upstream_port <port>
         --radio "radio==<radio> stations==<number stations> ssid==<ssid> ssid_pw==<ssid password>
         security==<security type: wpa2, open, wpa3>" --debug

EXAMPLE:

#########################################
# Examples
#########################################
Example running traffic with two radios
1. Test duration 30 minutes
2. Traffic IPv4 TCP, UDP
3. Upstream-port eth2
4. Radio #0 wiphy0 has 1 station, ssid = ssid_2g, ssid password = ssid_pw_2g  security = wpa2
5. Radio #1 wiphy1 has 2 stations, ssid = ssid_5g, ssid password = BLANK security = open
6. Create connections with TOS of BK and VI

         # The script now supports multiple radios, each specified with an individual --radio switch.

        # Interopt example Creating stations
            Interopt testing creating stations
            ./test_l3.py --lfmgr 192.168.0.104             --test_duration 60s            --polling_interval 5s            --upstream_

        # Interopt using existing stations
            Interopt testing creating stations
```

```
    ./test_l3.py --lfmgr 192.168.91.50            --test_duration 60s            --polling_interval 5s            --upstream_

   * UDP and TCP bi-directional test, no use of controller.
     ./test_l3.py --mgr 192.168.200.83 --endp_type 'lf_udp,lf_tcp' --upstream_port 1.1.eth1
     --radio "radio==1.1.wiphy0 stations==5 ssid==Netgear2g ssid_pw==lanforge security==wpa2"
     --radio "radio==1.1.wiphy1 stations==1 ssid==Netgear5g ssid_pw==lanforge security==wpa2"
     --test_duration 60s

   * Port resets, chooses random value between min and max
     ./test_l3.py --lfmgr 192.168.200.83 --test_duration 90s --polling_interval 10s --upstream_port eth1
     --radio 'radio==wiphy0,stations==4,ssid==Netgear2g,ssid_pw==lanforge,security==wpa2,reset_port_enable==TRUE,
     reset_port_time_min==10s,reset_port_time_max==20s' --endp_type lf_udp --rates_are_totals --side_a_min_bps=20000
     --side_b_min_bps=300000000

 # Command: (remove carriage returns)
     ./test_l3.py --lfmgr 192.168.200.83 --test_duration 30s --endp_type "lf_tcp,lf_udp" --tos "BK VI" --upstream_port 1.1.eth
     --radio "radio==1.1.wiphy0 stations==1 ssid==Netgear2g ssid_pw==lanforge security==wpa2"

 # Have the stations continue to run after the completion of the script
     ./test_l3.py --lfmgr 192.168.200.83 --endp_type 'lf_udp,lf_tcp' --tos BK --upstream_port 1.1.eth1
     --radio 'radio==wiphy0 stations==2 ssid==Netgear2g ssid_pw==lanforge security==wpa2' --test_duration 30s
     --polling_interval 5s --side_a_min_bps 256000 --side_b_min_bps 102400000 --no_stop_traffic

 #  Have script use existing stations from previous run where traffic was not stopped and also create new stations and leave t
     ./test_l3.py --lfmgr 192.168.200.83 --endp_type 'lf_udp,lf_tcp' --tos BK --upstream_port 1.1.eth1
     --radio 'radio==wiphy0 stations==2 ssid==Netgear2g ssid_pw==lanforge security==wpa2' --sta_start_offset 1000
     --test_duration 30s --polling_interval 5s --side_a_min_bps 256000 --side_b_min_bps 102400000 --use_existing_station_list
     --existing_station_list '1.1.sta0000,1.1.sta0001,1.1.sta0002' --no_stop_traffic

 # Have script use wifi_settings enable flages  ::  wifi_settings==wifi_settings,enable_flags==(ht160_enable&&wpa2_enable&&802
     ./test_l3.py --lfmgr 192.168.200.83 --test_duration 20s --polling_interval 5s --upstream_port 1.1.eth1
     --radio 'radio==1.1.wiphy0,stations==1,ssid==Netgear2g,ssid_pw==lanforge,security==wpa2,wifi_mode==0,wifi_settings==wifi_
     --radio 'radio==1.1.wiphy1,stations==1,ssid==Netgear5g,ssid_pw==lanforge,security==wpa2,wifi_mode==0,wifi_settings==wifi_
     --radio 'radio==1.1.wiphy2,stations==1,ssid==Netgear2g,ssid_pw==lanforge,security==wpa2,wifi_mode==0,wifi_settings==wifi_
     --endp_type lf_udp --rates_are_totals --side_a_min_bps=20000 --side_b_min_bps=300000000 --test_rig ID_003 --test_tag 'l3_
     --dut_hw_version 1.0 --dut_serial_num 12345678 --log_level debug

 # Setting wifi_settings per radio
    ./test_l3.py
    --lfmgr 192.168.100.116
    --local_lf_report_dir /home/lanforge/html-reports/
    --test_duration 15s
    --polling_interval 5s
    --upstream_port eth2
    --radio "radio==wiphy1 stations==4 ssid==asus11ax-5 ssid_pw==hello123 security==wpa2  mode==0 wifi_settings==wifi_settings
    --endp_type lf_udp
    --rates_are_totals
    --side_a_min_bps=20000
    --side_b_min_bps=300000000
    --test_rig CT-US-001
    --test_tag 'test_l3'

 # Example : LAN-1927  WPA2-TLS-Configuration
    ./test_l3.py
     --lfmgr 192.168.0.103
     --test_duration 20s
     --polling_interval 5s
     --upstream_port 1.1.eth2
     --radio 'radio==wiphy1,stations==1,ssid==ax88u_5g,ssid_pw==[BLANK],security==wpa2,wifi_settings==wifi_settings,wifi_mode=
     --endp_type lf_udp
     --rates_are_totals
     --side_a_min_bps=256000
     --side_b_min_bps=300000000
     --test_rig ID_003
     --test_tag 'test_l3'
     --dut_model_num GT-AXE11000
     --dut_sw_version 3.0.0.4.386_44266
     --dut_hw_version 1.0
     --dut_serial_num 12345678
     --log_level debug

# Example : LAN-1927  WPA2-TTLS-Configuration
    ./test_l3.py
     --lfmgr 192.168.0.103
     --test_duration 20s
     --polling_interval 5s
     --upstream_port 1.1.eth2
     --radio 'radio==wiphy1,stations==1,ssid==ax88u_5g,ssid_pw==[BLANK],security==wpa2,wifi_settings==wifi_settings,wifi_mode=
     --endp_type lf_udp
     --rates_are_totals
     --side_a_min_bps=256000
     --side_b_min_bps=300000000
     --test_rig ID_003
     --test_tag 'test_l3'
     --dut_model_num GT-AXE11000
     --dut_sw_version 3.0.0.4.386_44266
     --dut_hw_version 1.0
     --dut_serial_num 12345678
     --log_level debug

# Example : LAN-1927  WPA3-TTLS-Configuration
    ./test_l3.py
     --lfmgr 192.168.0.103
     --test_duration 20s
     --polling_interval 5s
     --upstream_port 1.1.eth2
     --radio 'radio==wiphy1,stations==1,ssid==ax88u_5g,ssid_pw==[BLANK],security==wpa3,wifi_settings==wifi_settings,wifi_mode=
     --endp_type lf_ud
     --rates_are_totals
     --side_a_min_bps=256000
     --side_b_min_bps=300000000
     --test_rig ID_003
     --test_tag 'test_l3'
     --dut_model_num GT-AXE11000
     --dut_sw_version 3.0.0.4.386_44266
     --dut_hw_version 1.0
     --dut_serial_num 12345678
     --log_level debug

# Example : LAN-1927  WPA3-TLS-Configuration
    ./test_l3.py
```

```
             --lfmgr 192.168.0.103
             --test_duration 20s
             --polling_interval 5s
             --upstream_port 1.1.eth2
             --radio 'radio==wiphy1,stations==1,ssid==ax88u_5g,ssid_pw==[BLANK],security==wpa3,wifi_settings==wifi_settings,wifi_mode=
             --endp_type lf_udp
             --rates_are_totals
             --side_a_min_bps=256000
             --side_b_min_bps=300000000
             --test_rig ID_003
             --test_tag 'test_l3'
             --dut_model_num GT-AXE11000
             --dut_sw_version 3.0.0.4.386_44266
             --dut_hw_version 1.0
             --dut_serial_num 12345678
             --log_level debug

SCRIPT_CLASSIFICATION:  Creation & Runs Traffic

SCRIPT_CATEGORIES:  Performance, Functional,  KPI Generation,  Report Generation

NOTES:

#################################
# Command switches
#################################

--mgr <hostname for where LANforge GUI is running>',default='localhost'
-d  / --test_duration <how long to run>  example --time 5d (5 days) default: 3m options: number followed by d, h, m or s',default='3m'
--tos:  Support different ToS settings: BK | BE | VI | VO | numeric',default="BE"
--debug:  Enable debugging',default=False
-t  / --endp_type <types of traffic> example --endp_type "lf_udp lf_tcp mc_udp"  Default: lf_udp , options: lf_udp, lf_udp6, lf_tcp, l
                        default='lf_udp', type=valid_endp_types
-u / --upstream_port <cross connect upstream_port> example: --upstream_port eth1',default='eth1')
-o / --outfile <Output file for csv data>", default='longevity_results'

<duration>: number followed by one of the following
d - days
h - hours
m - minutes
s - seconds

<traffic type>:
lf_udp  : IPv4 UDP traffic
lf_tcp  : IPv4 TCP traffic
lf_udp6 : IPv6 UDP traffic
lf_tcp6 : IPv6 TCP traffic
mc_udp  : IPv4 multi cast UDP traffic
mc_udp6 : IPv6 multi cast UDP traffic

<tos>:
BK, BE, VI, VO:  Optional wifi related Tos Settings.  Or, use your preferred numeric values. Cross connects type of service

    * Data 0 (Best Effort, BE): Medium priority queue, medium throughput and delay.
             Most traditional IP data is sent to this queue.
    * Data 1 (Background, BK): Lowest priority queue, high throughput. Bulk data that requires maximum throughput and
             is not time-sensitive is sent to this queue (FTP data, for example).
    * Data 2 (Video, VI): High priority queue, minimum delay. Time-sensitive data such as Video and other streaming
             media are automatically sent to this queue.
    * Data 3 (Voice, VO): Highest priority queue, minimum delay. Time-sensitive data such as Voice over IP (VoIP)
             is automatically sent to this Queue.

<wifi_mode>:
    Input       : Enum Val  : Shown by nc_show_ports

    AUTO        |  0        #  802.11
    802.11a     |  1        #  802.11a
    b           |  2        #  802.11b
    g           |  3        #  802.11g
    abg         |  4        #  802.11abg
    abgn        |  5        #  802.11abgn
    bgn         |  6        #  802.11bgn
    bg          |  7        #  802.11bg
    abgnAC      |  8        #  802.11abgn-AC
    anAC        |  9        #  802.11an-AC
    an          | 10        #  802.11an
    bgnAC       | 11        #  802.11bgn-AC
    abgnAX      | 12        #  802.11abgn-AX
                            #     a/b/g/n/AC/AX (dual-band AX) support
    bgnAX       | 13        #  802.11bgn-AX
    anAX        | 14        #  802.11an-AX
    aAX         | 15        #  802.11a-AX (6E disables /n and /ac)

wifi_settings flags are currently defined as:
    wpa_enable           | 0x10         # Enable WPA
    # Use Custom wpa_supplicant config file.
    custom_conf          | 0x20
    # Use wpa_supplicant configured for WEP encryption.
    wep_enable           | 0x200
    # Use wpa_supplicant configured for WPA2 encryption.
    wpa2_enable          | 0x400
    # Disable HT-40 even if hardware and AP support it.
    ht40_disable         | 0x800
    # Enable SCAN-SSID flag in wpa_supplicant.
    scan_ssid            | 0x1000
    # Use passive scanning (don't send probe requests).
    passive_scan         | 0x2000
    disable_sgi          | 0x4000       # Disable SGI (Short Guard Interval).
    # OK-To-Migrate (Allow station migration between LANforge radios)
    lf_sta_migrate       | 0x8000
    # Verbose-Debug:  Increase debug info in wpa-supplicant and hostapd logs.
    verbose              | 0x10000
    # Enable 802.11u (Interworking) feature.
    80211u_enable        | 0x20000
    # Enable 802.11u (Interworking) Auto-internetworking feature.  Always enabled currently.
    80211u_auto          | 0x40000
    # AP Provides access to internet (802.11u Interworking)
    80211u_gw            | 0x80000
    # AP requires additional step for access (802.11u Interworking)
    80211u_additional    | 0x100000
    # AP claims emergency services reachable (802.11u Interworking)
    80211u_e911          | 0x200000
    # AP provides Unauthenticated emergency services (802.11u Interworking)
    80211u_e911_unauth   | 0x400000
    # Enable Hotspot 2.0 (HS20) feature.  Requires WPA-2.
    hs20_enable          | 0x800000
    # AP:  Disable DGAF (used by HotSpot 2.0).
    disable_gdaf         | 0x1000000
    8021x_radius         | 0x2000000    # Use 802.1x (RADIUS for AP).
    # Enable oportunistic PMSKA caching for WPA2 (Related to 802.11r).
    80211r_pmska_cache   | 0x4000000
    # Disable HT80 (for AC chipset NICs only)
    disable_ht80         | 0x8000000
    ibss_mode            | 0x20000000   # Station should be in IBSS mode.
    # Enable OSEN protocol (OSU Server-only Authentication)
    osen_enable          | 0x40000000
    # Disable automatic station roaming based on scan results.
    disable_roam         | 0x80000000
    ht160_enable         | 0x100000000  # Enable HT160 mode.
    # Disable fast_reauth option for virtual stations.
    disable_fast_reauth  | 0x200000000
    mesh_mode            | 0x400000000  # Station should be in MESH mode.
    # Station should enable power-save.  May not work in all drivers/configurations.
    power_save_enable    | 0x800000000
    create_admin_down    | 0x1000000000 # Station should be created admin-down.
    # WDS station (sort of like a lame mesh), not supported on ath10k
    wds-mode             | 0x2000000000
    # Do not include supported-oper-class-IE in assoc requests.  May work around AP bugs.
    no-supp-op-class-ie  | 0x4000000000
    # Enable/disable tx-offloads, typically managed by set_wifi_txo command
    txo-enable           | 0x8000000000
    use-wpa3             | 0x10000000000 # Enable WPA-3 (SAE Personal) mode.
    use-bss-transition   | 0x80000000000 # Enable BSS transition.
    disable-twt          | 0x100000000000 # Disable TWT mode

For wifi_extra_keys syntax :
    telnet <lanforge ip> 4001
    type: help set_wifi_extra
wifi_extra keys:
                key_mgmt  (Key Mangement)
                pairwise  (Pairwise Ciphers)
                group   (Group Ciphers)
                psk     (WPA PSK)
                wep_key
                ca_cert (CA Cert File)
                eap     (EAP Methods) EAP method: MD5, MSCHAPV2, OTP, GTC, TLS, PEAP, TTLS. (note different the GUI no appended EAP-)
                identity    (EAP Identity)
                anonymous_identity  (EAP Anon Identity)
                phase1  (Phase-1)
                phase2  (Phase-2)
                passwd  (EAP Password)
                pin (EAP Pin)
                pac_file    (PAC file)
                private_key (Private Key)
                pk_password (PK Password)
                hessid="00:00:00:00:00:00"
                realm   (Realm)
                client_cert (Client Cert)
                imsi    (IMSI)
                milenage    (Milenage)
                domain  (Domain)
                roaming_consortium  (Consortium)
                venue_group ()
                network_type    (Network Auth)
                ipaddr_type_avail   ()
                network_auth_type ()
                anqp_3gpp_cell_net ()

                ieee80211w :   0,1,2

Multicast traffic :
        Multicast traffic default IGMP Address in the range of 224.0.0.0 to 239.255.255.255,
        so I have provided 224.9.9.9 as IGMP address and IGMP Dest port as 9999 and MIN-IP PORT as 9999.
        these values must be same on the eth1(server side) and client side, then the traffic will run.

===============================================================================
 ** FURTHER INFORMATION **
    Using the layer3_cols flag:

    Currently the output function does not support inputting the columns in layer3_cols the way they are displayed in the GUI. This qu
    certain columns in the GUI in your final report, please match the according GUI column display to it's counterpart to have the col
    your report.

    GUI Column Display       Layer3_cols argument to type in (to print in report)

    Name                |  'name'
    EID                 |  'eid'
    Run                 |  'run'
    Mng                 |  'mng'
    Script              |  'script'
    Tx Rate             |  'tx rate'
    Tx Rate (1 min)     |  'tx rate (1 min)'
    Tx Rate (last)      |  'tx rate (last)'
    Tx Rate LL          |  'tx rate ll'
    Rx Rate             |  'rx rate'
    Rx Rate (1 min)     |  'rx rate (1 min)'
    Rx Rate (last)      |  'rx rate (last)'
    Rx Rate LL          |  'rx rate ll'
    Rx Drop %           |  'rx drop %'
    Tx PDUs             |  'tx pdus'
    Tx Pkts LL          |  'tx pkts ll'
    PDU/s TX            |  'pdu/s tx'
    Pps TX LL           |  'pps tx ll'
    Rx PDUs             |  'rx pdus'
    Rx Pkts LL          |  'pps rx ll'
    PDU/s RX            |  'pdu/s tx'
    Pps RX LL           |  'pps rx ll'
    Delay               |  'delay'
    Dropped             |  'dropped'
    Jitter              |  'jitter'
    Tx Bytes            |  'tx bytes'
    Rx Bytes            |  'rx bytes'
    Replays             |  'replays'
    TCP Rtx             |  'tcp rtx'
    Dup Pkts            |  'dup pkts'
    Rx Dup %            |  'rx dup %'
    OOO Pkts            |  'ooo pkts'
    Rx OOO %            |  'rx ooo %'
    RX Wrong Dev        |  'rx wrong dev'
    CRC Fail            |  'crc fail'
    RX BER              |  'rx ber'
    CX Active           |  'cx active'
    CX Estab/s          |  'cx estab/s'
    1st RX              |  '1st rx'
    CX TO               |  'cx to'
    Pattern             |  'pattern'
    Min PDU             |  'min pdu'
    Max PDU             |  'max pdu'
    Min Rate            |  'min rate'
    Max Rate            |  'max rate'
    Send Buf            |  'send buf'
    Rcv Buf             |  'rcv buf'
    CWND                |  'cwnd'
    TCP MSS             |  'tcp mss'
    Bursty              |  'bursty'
    A/B                 |  'a/b'
    Elapsed             |  'elapsed'
    Destination Addr    |  'destination addr'
    Source Addr         |  'source addr'

    Using the port_mgr_cols flag:
         '4way time (us)'
         'activity'
         'alias'
         'anqp time (us)'
         'ap'
         'beacon'
         'bps rx'
         'bps rx ll'
         'bps tx'
         'bps tx ll'
         'bytes rx ll'
         'bytes tx ll'
         'channel'
         'collisions'
         'connections'
         'crypt'
         'cx ago'
         'cx time (us)'
         'device'
         'dhcp (ms)'
         'down'
         'entity id'
         'gateway ip'
         'ip'
         'ipv6 address'
         'ipv6 gateway'
         'key/phrase'
         'login-fail'
         'login-ok'
         'logout-fail'
         'logout-ok'
         'mac'
         'mask'
         'misc'
         'mode'
         'mtu'
         'no cx (us)'
         'noise'
         'parent dev'
         'phantom'
         'port'
         'port type'
         'pps rx'
         'pps tx'
         'qlen'
         'reset'
         'retry failed'
         'rx bytes'
         'rx crc'
         'rx drop'
         'rx errors'
         'rx fifo'
         'rx frame'
         'rx length'
         'rx miss'
         'rx over'
         'rx pkts'
         'rx-rate'
         'sec'
         'signal'
         'ssid'
         'status'
         'time-stamp'
         'tx abort'
         'tx bytes'
         'tx crr'
         'tx errors'
         'tx fifo'
         'tx hb'
         'tx pkts'
         'tx wind'
         'tx-failed %'
         'tx-rate'
         'wifi retries'

    Can't decide what columns to use? You can just use 'all' to select all available columns from both tables.

STATUS: Functional

VERIFIED_ON:   18-JULY-2023,
             GUI Version:  5.4.6
             Kernel Version: 5.19.17+
LICENSE:
          Free to distribute and modify. LANforge systems must be licensed.
          Copyright 2023 Candela Technologies Inc

INCLUDE_IN_README: False



options:
  -h, --help            show this help message and exit

arguments defined in test_l3.py file:
  --local_lf_report_dir LOCAL_LF_REPORT_DIR
                        --local_lf_report_dir override the report path (lanforge/html-reports), primary used when making another direc
  --results_dir_name RESULTS_DIR_NAME
                        the name of the directory that contains the output from the test /lanforge/html-reports/<results_dir_name> def
  --test_rig TEST_RIG   test rig for kpi.csv, testbed that the tests are run on
  --test_tag TEST_TAG   test tag for kpi.csv,  test specific information to differenciate the test
  --dut_hw_version DUT_HW_VERSION
                        dut hw version for kpi.csv, hardware version of the device under test
  --dut_sw_version DUT_SW_VERSION
                        dut sw version for kpi.csv, software version of the device under test
  --dut_model_num DUT_MODEL_NUM
                        dut model for kpi.csv,  model number / name of the device under test
  --dut_serial_num DUT_SERIAL_NUM
                        dut serial for kpi.csv, serial number / serial number of the device under test
  --test_priority TEST_PRIORITY
                        dut model for kpi.csv,  test-priority is arbitrary number
  --test_id TEST_ID     test-id for kpi.csv,  script or test name
  -o CSV_OUTFILE, --csv_outfile CSV_OUTFILE
                        --csv_outfile <Output file for csv data>
  --tty TTY             --tty "/dev/ttyUSB2" the serial interface to the AP
  --baud BAUD           --baud "9600"  AP baud rate for the serial interface
  --mgr LFMGR, --lfmgr LFMGR
                        --lfmgr <hostname for where LANforge GUI is running>
  --mgr_port LFMGR_PORT, --lfmgr_port LFMGR_PORT
                        --lfmgr_port <port LANforge GUI HTTP service is running on>
  --test_duration TEST_DURATION
                        --test_duration <how long to run>  example --time 5d (5 days) default: 3m options: number followed by d, h, m
  --tos TOS             --tos:  Support different ToS settings: BK,BE,VI,VO,numeric
  --debug               --debug this will enable debugging in py-json method
  --log_level LOG_LEVEL
                        Set logging level: debug | info | warning | error | critical
  -t ENDP_TYPE, --endp_type ENDP_TYPE
                        --endp_type <types of traffic> example --endp_type "lf_udp lf_tcp mc_udp"  Default: lf_udp , options: lf_udp,
  -u UPSTREAM_PORT, --upstream_port UPSTREAM_PORT
                        --upstream_port <cross connect upstream_port> example: --upstream_port eth1
  --downstream_port DOWNSTREAM_PORT
                        --downstream_port <cross connect downstream_port>  for use when downstream is ethernet (eth to eth connection)
  --polling_interval POLLING_INTERVAL
                        --polling_interval <seconds>
  -r RADIO, --radio RADIO
                         --radio "radio==<number_of_wiphy stations==<number of stations> ssid==<ssid> ssid_pw==<ssid password> securit
  -amr SIDE_A_MIN_BPS, --side_a_min_bps SIDE_A_MIN_BPS, --upload_min_bps SIDE_A_MIN_BPS
                        --side_a_min_bps, requested downstream min tx rate at stations / client, comma separated list for multiple ite
                                        When running with tcp/udp traffic along with mcast , mcast will ignore the upload value
  -amp SIDE_A_MIN_PDU, --side_a_min_pdu SIDE_A_MIN_PDU
                        --side_a_min_pdu, downstream pdu size, comma separated list for multiple iterations.  Default MTU
  -bmr SIDE_B_MIN_BPS, --download_min_bps SIDE_B_MIN_BPS, --side_b_min_bps SIDE_B_MIN_BPS, --do SIDE_B_MIN_BPS
                        --side_b_min_bps or --download_min_bps, requested upstream min tx rate, comma separated list for multiple iter
                                        When runnign with tcp/udp and mcast will use this value
  -bmp SIDE_B_MIN_PDU, --side_b_min_pdu SIDE_B_MIN_PDU
                        --side_b_min_pdu, upstream pdu size, comma separated list for multiple iterations. Default MTU
  --rates_are_totals    Treat configured rates as totals instead of using the un-modified rate for every connection.
  --multiconn MULTICONN
                        Configure multi-conn setting for endpoints.  Default is 1 (auto-helper is enabled by default as well).
  --attenuators ATTENUATORS
                        --attenuators,  comma separated list of attenuator module eids:  shelf.resource.atten-serno.atten-idx
  --atten_vals ATTEN_VALS
                        --atten_vals,  comma separated list of attenuator settings in ddb units (1/10 of db)
  --wait WAIT           --wait <time> , time to wait at the end of the test
  --sta_start_offset STA_START_OFFSET
                        Station start offset for building stations
  --no_pre_cleanup      Do not pre cleanup stations on start
  --no_cleanup          Do not cleanup before exit
  --cleanup_cx          cleanup cx before exit
  --csv_data_to_report  collected interval data in csv for each cx will be put in report
  --no_stop_traffic     leave traffic running
  --quiesce_cx          --quiesce store true,  allow the cx to drain then stop so as to not have rx drop pkts
  --use_existing_station_list
                        --use_station_list ,full eid must be given,the script will use stations from the list, no configuration on the
  --existing_station_list EXISTING_STATION_LIST
                        --station_list [list of stations] , use the stations in the list , multiple station lists may be entered
  --wait_for_ip_sec WAIT_FOR_IP_SEC
                        --wait_for_ip_sec <seconds>  default : 120s
  --lf_logger_config_json LF_LOGGER_CONFIG_JSON
                        --lf_logger_config_json <json file> , json configuration of logger
  --ap_read             --ap_read  flag present enable reading ap
  --ap_module AP_MODULE
                        series module
  --ap_test_mode AP_TEST_MODE
                        --ap_mode
  --ap_scheme {serial,telnet,ssh,mux_serial}
                        --ap_scheme '/dev/ttyUSB0'
  --ap_serial_port AP_SERIAL_PORT
                        --ap_serial_port '/dev/ttyUSB0'
  --ap_serial_baud AP_SERIAL_BAUD
                        --ap_baud '115200'',  default='115200
  --ap_ip AP_IP         --ap_ip
  --ap_ssh_port AP_SSH_PORT
                        --ap_ssh_port
  --ap_telnet_port AP_TELNET_PORT
                        --ap_telnet_port
  --ap_user AP_USER     --ap_user , the user name for the ap, default = lanforge
  --ap_passwd AP_PASSWD
                        --ap_passwd, the password for the ap default = lanforge
  --ap_if_2g AP_IF_2G   --ap_if_2g eth6
  --ap_if_5g AP_IF_5G   --ap_if_5g eth7
  --ap_if_6g AP_IF_6G   --ap_if_6g eth8
  --ap_file AP_FILE     --ap_file 'ap_file.txt'
  --ap_band_list AP_BAND_LIST
                        --ap_band_list '2g,5g,6g' supported bands

        Useful Information:
            1. Polling interval for checking traffic is fixed at 1 minute
            2. The test will generate csv file
            3. The tx/rx rates are fixed at 256000 bits per second
            4. Maximum stations per radio based on radio
```

py-scripts/test_l3_longevity.py

```
usage: test_l3_longevity.py [-h] [--help_summary]
                            [--local_lf_report_dir LOCAL_LF_REPORT_DIR]
                            [--test_rig TEST_RIG] [--test_tag TEST_TAG]
                            [--dut_hw_version DUT_HW_VERSION]
                            [--dut_sw_version DUT_SW_VERSION]
                            [--dut_model_num DUT_MODEL_NUM]
                            [--dut_serial_num DUT_SERIAL_NUM]
                            [--test_priority TEST_PRIORITY]
                            [--test_id TEST_ID] [-o CSV_OUTFILE] [--tty TTY]
                            [--baud BAUD] [--mgr LFMGR]
                            [--mgr_port LFMGR_PORT]
                            [--test_duration TEST_DURATION] [--tos TOS]
                            [--debug] [-t ENDP_TYPE] [-u UPSTREAM_PORT]
                            [--downstream_port DOWNSTREAM_PORT]
                            [--polling_interval POLLING_INTERVAL] [-r RADIO]
                            [--collect_layer3_data] [--ap_read]
                            [--ap_scheme {serial,telnet,ssh,mux_serial}]
                            [--ap_port AP_PORT] [--ap_baud AP_BAUD]
                            [--ap_ip AP_IP] [--ap_ssh_port AP_SSH_PORT]
                            [--ap_user AP_USER] [--ap_passwd AP_PASSWD]
                            [--ap_if_2g AP_IF_2G] [--ap_if_5g AP_IF_5G]
                            [--ap_if_6g AP_IF_6G] [--ap_cmd_6g AP_CMD_6G]
                            [--ap_cmd_5g AP_CMD_5G] [--ap_cmd_2g AP_CMD_2G]
                            [--ap_cmd_ul_6g AP_CMD_UL_6G]
                            [--ap_cmd_ul_5g AP_CMD_UL_5G]
                            [--ap_cmd_ul_2g AP_CMD_UL_2G]
                            [--ap_chanim_cmd_6g AP_CHANIM_CMD_6G]
                            [--ap_chanim_cmd_5g AP_CHANIM_CMD_5G]
                            [--ap_chanim_cmd_2g AP_CHANIM_CMD_2G]
                            [--ap_scheduler_stats] [--ap_ofdma_stats]
                            [--ap_test_mode] [-amr SIDE_A_MIN_BPS]
                            [-amp SIDE_A_MIN_PDU] [-bmr SIDE_B_MIN_BPS]
                            [-bmp SIDE_B_MIN_PDU] [--rates_are_totals]
                            [--multiconn MULTICONN]
                            [--attenuators ATTENUATORS]
                            [--atten_vals ATTEN_VALS]
                            [--influx_host INFLUX_HOST]
                            [--influx_port INFLUX_PORT]
                            [--influx_org INFLUX_ORG]
                            [--influx_token INFLUX_TOKEN]
                            [--influx_bucket INFLUX_BUCKET]
                            [--influx_tag INFLUX_TAG INFLUX_TAG]
                            [--cap_ctl_out] [--wait WAIT]
                            [--sta_start_offset STA_START_OFFSET]
                            [--no_pre_cleanup] [--no_cleanup]
                            [--no_stop_traffic] [--use_existing_station_list]
                            [--existing_station_list EXISTING_STATION_LIST]
                            [--wait_for_ip_sec WAIT_FOR_IP_SEC]
                            [--log_level LOG_LEVEL]
                            [--lf_logger_config_json LF_LOGGER_CONFIG_JSON]

test_l3_longevity.py:
--------------------

Summary :
----------
The Layer 3 Traffic Generation Test is designed to test the performance of the
Access Point by running layer 3 Cross-Connect Traffic.  Layer-3 Cross-Connects represent a stream
of data flowing through the system under test. A Cross-Connect (CX) is composed of two Endpoints,
each of which is associated with a particular Port (physical or virtual interface).

The test will create stations, create cx traffic between upstream port and stations, run traffic
and generate a report.

Generic command layout:
-----------------------
./test_l3_longevity.py --mgr <ip_address> --test_duration <duration> --endp_type <traffic types>
--upstream_port <shelf>.<resource>.<port>
--radio "radio==<shelf>.<resource>.<radio> stations==<number stations> ssid==<ssid> ssid_pw==<ssid password>
security==<security type: wpa2, open, wpa3>" --debug

Multiple radios may be entered with individual --radio switches

# UDP bi-directional test, no use of controller.
./test_l3_longevity.py --mgr localhost --endp_type 'lf_udp lf_tcp' --upstream_port 1.1.eth1
--radio "radio==1.1.wiphy0 stations==10 ssid==ASUS_70 ssid_pw==[BLANK] security==open"
--radio "radio==1.1.wiphy2 stations==1 ssid==ASUS_70 ssid_pw==[BLANK] security==open" --test_duration 30s

# Port resets, chooses random value between min and max
./test_l3_longevity.py --lfmgr LF_MGR_IP --test_duration 90s --polling_interval 10s --upstream_port 1.1.eth2
--radio 'radio==1.1.wiphy1,stations==4,ssid==SSID_USED,ssid_pw==SSID_PW_USED,security==SECURITY_USED,
reset_port_enable==TRUE,reset_port_time_min==10s,reset_port_time_max==20s'
--endp_type lf_udp --rates_are_totals --side_a_min_bps=20000 --side_b_min_bps=300000000"

<duration>: number followed by one of the following
d - days
h - hours
m - minutes
s - seconds

<traffic type>:
lf_udp  : IPv4 UDP traffic
lf_tcp  : IPv4 TCP traffic
lf_udp6 : IPv6 UDP traffic
lf_tcp6 : IPv6 TCP traffic
mc_udp  : IPv4 multi cast UDP traffic
mc_udp6 : IPv6 multi cast UDP traffic

<tos>:
```

```
BK, BE, VI, VO:  Optional wifi related Tos Settings.  Or, use your preferred numeric values.

#################################
# Command switches
#################################

--mgr <hostname for where LANforge GUI is running>',default='localhost'
-d  / --test_duration <how long to run>  example --time 5d (5 days) default: 3m options: number followed by d, h, m or s',default='3m'
--tos:  Support different ToS settings: BK | BE | VI | VO | numeric',default="BE"
--debug:  Enable debugging',default=False
-t  / --endp_type <types of traffic> example --endp_type "lf_udp lf_tcp mc_udp"  Default: lf_udp , options: lf_udp, lf_udp6, lf_tcp, l
                        default='lf_udp', type=valid_endp_types
-u / --upstream_port <cross connect upstream_port> example: --upstream_port eth1',default='eth1')
-o / --outfile <Output file for csv data>", default='longevity_results'

#########################################
# Examples
# #######################################
Example #1  running traffic with two radios
1. Test duration 30 minutes
2. Traffic IPv4 TCP, UDP
3. Upstream-port eth2
4. Radio #0 wiphy0 has 1 station, ssid = ssid_2g, ssid password = ssid_pw_2g  security = wpa2
5. Radio #1 wiphy1 has 2 stations, ssid = ssid_5g, ssid password = BLANK security = open

Command:
python3 ./test_l3_longevity.py --test_duration 30s --endp_type "lf_tcp lf_udp" --tos "BK VI" --upstream_port eth2
--radio "radio==wiphy0 stations==1 ssid==ssid_2g ssid_pw==ssid_pw_2g security==wpa2"
--radio "radio==wiphy1 stations==2 ssid==ssid_5g ssid_pw==BLANK security==open"

Example : Have the stations continue to run after the completion of the script
    ./test_l3_longevity.py --lfmgr 192.168.0.101 --endp_type 'lf_udp,lf_tcp' --tos BK --upstream_port 1.1.eth2
        --radio 'radio==wiphy1 stations==2 ssid==asus_2g ssid_pw==lf_asus_2g security==wpa2'
        --test_duration 30s --polling_interval 5s
        --side_a_min_bps 256000 --side_b_min_bps 102400000
        --no_stop_traffic

Example : Have script use existing stations from previous run where traffic was not stopped and also create new stations and
        leave traffic running
        ./test_l3_longevity.py --lfmgr 192.168.0.101 --endp_type 'lf_udp,lf_tcp' --tos BK --upstream_port 1.1.eth2
        --radio 'radio==wiphy0 stations==2 ssid==asus_5g ssid_pw==lf_asus_5g security==wpa2'
        --sta_start_offset 1000
        --test_duration 30s --polling_interval 5s
        --side_a_min_bps 256000 --side_b_min_bps 102400000
        --use_existing_station_list
        --existing_station_list '1.1.sta0000,1.1.sta0001'
        --no_stop_traffic

Example : Add the following switches to use ssh to access ASUS (both ssh and serial supported), the interfaces need to be provided

                --ap_read
                --ap_scheme ssh
                --ap_ip 192.168.50.1
                --ap_ssh_port 1025
                --ap_user lanforge
                --ap_passwd lanforge
                --ap_if_2g eth6
                --ap_if_5g eth7
                --ap_if_6g eth8

Setting wifi_settings per radio
./test_l3_longevity.py  --lfmgr 192.168.100.116 --local_lf_report_dir /home/lanforge/html-reports/ --test_duration 15s
--polling_interval 5s --upstream_port 1.1.eth2
--radio "radio==1.1.wiphy1 stations==4 ssid==asus11ax-5 ssid_pw==hello123 security==wpa2  mode==0 wifi_settings==wifi_settings
    enable_flags==('ht160_enable'|'wpa2_enable'|'80211u_enable'|'create_admin_down'|'ht160_enable') "
--endp_type lf_udp --rates_are_totals --side_a_min_bps=20000 --side_b_min_bps=300000000 --test_rig CT-US-001 --test_tag 'l3_longevity'

Note: for enable flags can us && as separator in vscode

        wifi_mode
        Input       : Enum Val  : Shown by nc_show_ports

        AUTO        |  0        #  802.11g
        802.11a     |  1        #  802.11a
        b           |  2        #  802.11b
        g           |  3        #  802.11g
        abg         |  4        #  802.11abg
        abgn        |  5        #  802.11abgn
        bgn         |  6        #  802.11bgn
        bg          |  7        #  802.11bg
        abgnAC      |  8        #  802.11abgn-AC
        anAC        |  9        #  802.11an-AC
        an          | 10        #  802.11an
        bgnAC       | 11        #  802.11bgn-AC
        abgnAX      | 12        #  802.11abgn-AX
                                #     a/b/g/n/AC/AX (dual-band AX) support
        bgnAX       | 13        #  802.11bgn-AX
        anAX        | 14        #  802.11an-AX
        aAX         | 15        #  802.11a-AX (6E disables /n and /ac)

        wifi_settings flags are currently defined as:
        wpa_enable           | 0x10         # Enable WPA
        custom_conf          | 0x20         # Use Custom wpa_supplicant config file.
        wep_enable           | 0x200        # Use wpa_supplicant configured for WEP encryption.
        wpa2_enable          | 0x400        # Use wpa_supplicant configured for WPA2 encryption.
        ht40_disable         | 0x800        # Disable HT-40 even if hardware and AP support it.
        scan_ssid            | 0x1000       # Enable SCAN-SSID flag in wpa_supplicant.
        passive_scan         | 0x2000       # Use passive scanning (don't send probe requests).
        disable_sgi          | 0x4000       # Disable SGI (Short Guard Interval).
        lf_sta_migrate       | 0x8000       # OK-To-Migrate (Allow station migration between LANforge radios)
        verbose              | 0x10000      # Verbose-Debug:  Increase debug info in wpa-supplicant and hostapd logs.
        80211u_enable        | 0x20000      # Enable 802.11u (Interworking) feature.
        80211u_auto          | 0x40000      # Enable 802.11u (Interworking) Auto-internetworking feature.  Always enabled currently.
        80211u_gw            | 0x80000      # AP Provides access to internet (802.11u Interworking)
        80211u_additional    | 0x100000     # AP requires additional step for access (802.11u Interworking)
        80211u_e911          | 0x200000     # AP claims emergency services reachable (802.11u Interworking)
        80211u_e911_unauth   | 0x400000     # AP provides Unauthenticated emergency services (802.11u Interworking)
        hs20_enable          | 0x800000     # Enable Hotspot 2.0 (HS20) feature.  Requires WPA-2.
        disable_gdaf         | 0x1000000    # AP:  Disable DGAF (used by HotSpot 2.0).
        8021x_radius         | 0x2000000    # Use 802.1x (RADIUS for AP).
        80211r_pmska_cache   | 0x4000000    # Enable oportunistic PMSKA caching for WPA2 (Related to 802.11r).
        disable_ht80         | 0x8000000    # Disable HT80 (for AC chipset NICs only)
        ibss_mode            | 0x20000000   # Station should be in IBSS mode.
        osen_enable          | 0x40000000   # Enable OSEN protocol (OSU Server-only Authentication)
        disable_roam         | 0x80000000   # Disable automatic station roaming based on scan results.
        ht160_enable         | 0x100000000  # Enable HT160 mode.
        disable_fast_reauth  | 0x200000000  # Disable fast_reauth option for virtual stations.
        mesh_mode            | 0x400000000  # Station should be in MESH mode.
        power_save_enable    | 0x800000000  # Station should enable power-save.  May not work in all drivers/configurations.
        create_admin_down    | 0x1000000000 # Station should be created admin-down.
        wds-mode             | 0x2000000000 # WDS station (sort of like a lame mesh), not supported on ath10k
        no-supp-op-class-ie  | 0x4000000000 # Do not include supported-oper-class-IE in assoc requests.  May work around AP bugs.
        txo-enable           | 0x8000000000 # Enable/disable tx-offloads, typically managed by set_wifi_txo command
        use-wpa3             | 0x10000000000 # Enable WPA-3 (SAE Personal) mode.
        use-bss-transition   | 0x80000000000 # Enable BSS transition.
        disable-twt          | 0x100000000000 # Disable TWT mode

===============================================================================
 ** FURTHER INFORMATION **
    Using the layer3_cols flag:

    Currently the output function does not support inputting the columns in layer3_cols the way they are displayed in the GUI. This qu
    certain columns in the GUI in your final report, please match the according GUI column display to it's counterpart to have the col
    your report.

    GUI Column Display       Layer3_cols argument to type in (to print in report)

    Name                |  'name'
    EID                 |  'eid'
    Run                 |  'run'
    Mng                 |  'mng'
    Script              |  'script'
    Tx Rate             |  'tx rate'
    Tx Rate (1 min)     |  'tx rate (1 min)'
    Tx Rate (last)      |  'tx rate (last)'
    Tx Rate LL          |  'tx rate ll'
    Rx Rate             |  'rx rate'
    Rx Rate (1 min)     |  'rx rate (1 min)'
    Rx Rate (last)      |  'rx rate (last)'
    Rx Rate LL          |  'rx rate ll'
    Rx Drop %           |  'rx drop %'
    Tx PDUs             |  'tx pdus'
    Tx Pkts LL          |  'tx pkts ll'
    PDU/s TX            |  'pdu/s tx'
    Pps TX LL           |  'pps tx ll'
    Rx PDUs             |  'rx pdus'
    Rx Pkts LL          |  'pps rx ll'
    PDU/s RX            |  'pdu/s tx'
    Pps RX LL           |  'pps rx ll'
    Delay               |  'delay'
    Dropped             |  'dropped'
    Jitter              |  'jitter'
    Tx Bytes            |  'tx bytes'
    Rx Bytes            |  'rx bytes'
    Replays             |  'replays'
    TCP Rtx             |  'tcp rtx'
    Dup Pkts            |  'dup pkts'
    Rx Dup %            |  'rx dup %'
    OOO Pkts            |  'ooo pkts'
    Rx OOO %            |  'rx ooo %'
    RX Wrong Dev        |  'rx wrong dev'
    CRC Fail            |  'crc fail'
    RX BER              |  'rx ber'
    CX Active           |  'cx active'
    CX Estab/s          |  'cx estab/s'
    1st RX              |  '1st rx'
    CX TO               |  'cx to'
    Pattern             |  'pattern'
    Min PDU             |  'min pdu'
    Max PDU             |  'max pdu'
    Min Rate            |  'min rate'
    Max Rate            |  'max rate'
    Send Buf            |  'send buf'
    Rcv Buf             |  'rcv buf'
    CWND                |  'cwnd'
    TCP MSS             |  'tcp mss'
    Bursty              |  'bursty'
    A/B                 |  'a/b'
    Elapsed             |  'elapsed'
    Destination Addr    |  'destination addr'
    Source Addr         |  'source addr'

    Using the port_mgr_cols flag:
         '4way time (us)'
         'activity'
         'alias'
         'anqp time (us)'
         'ap'
         'beacon'
         'bps rx'
         'bps rx ll'
         'bps tx'
         'bps tx ll'
         'bytes rx ll'
         'bytes tx ll'
         'channel'
         'collisions'
         'connections'
         'crypt'
         'cx ago'
         'cx time (us)'
         'device'
         'dhcp (ms)'
         'down'
         'entity id'
         'gateway ip'
         'ip'
         'ipv6 address'
         'ipv6 gateway'
         'key/phrase'
         'login-fail'
         'login-ok'
         'logout-fail'
         'logout-ok'
         'mac'
         'mask'
         'misc'
         'mode'
         'mtu'
         'no cx (us)'
         'noise'
         'parent dev'
         'phantom'
         'port'
         'port type'
         'pps rx'
         'pps tx'
         'qlen'
         'reset'
         'retry failed'
         'rx bytes'
         'rx crc'
         'rx drop'
         'rx errors'
         'rx fifo'
         'rx frame'
         'rx length'
         'rx miss'
         'rx over'
         'rx pkts'
         'rx-rate'
         'sec'
         'signal'
         'ssid'
         'status'
         'time-stamp'
         'tx abort'
         'tx bytes'
         'tx crr'
         'tx errors'
         'tx fifo'
         'tx hb'
         'tx pkts'
         'tx wind'
         'tx-failed %'
         'tx-rate'
         'wifi retries'

    Can't decide what columns to use? You can just use 'all' to select all available columns from both tables.



options:
  -h, --help            show this help message and exit
  --help_summary        Show summary of what this script does
  --local_lf_report_dir LOCAL_LF_REPORT_DIR
                        --local_lf_report_dir override the report path, primary use when running test in test suite
  --test_rig TEST_RIG   test rig for kpi.csv, testbed that the tests are run on
  --test_tag TEST_TAG   test tag for kpi.csv,  test specific information to differenciate the test
  --dut_hw_version DUT_HW_VERSION
                        dut hw version for kpi.csv, hardware version of the device under test
  --dut_sw_version DUT_SW_VERSION
                        dut sw version for kpi.csv, software version of the device under test
  --dut_model_num DUT_MODEL_NUM
                        dut model for kpi.csv,  model number / name of the device under test
  --dut_serial_num DUT_SERIAL_NUM
                        dut serial for kpi.csv, serial number / serial number of the device under test
  --test_priority TEST_PRIORITY
                        dut model for kpi.csv,  test-priority is arbitrary number
  --test_id TEST_ID     test-id for kpi.csv,  script or test name
  -o CSV_OUTFILE, --csv_outfile CSV_OUTFILE
                        --csv_outfile <Output file for csv data>
  --tty TTY             --tty "/dev/ttyUSB2" the serial interface to the AP
  --baud BAUD           --baud "9600"  AP baud rate for the serial interface
  --mgr LFMGR, --lfmgr LFMGR
                        --lfmgr <hostname for where LANforge GUI is running>
  --mgr_port LFMGR_PORT, --lfmgr_port LFMGR_PORT
                        --lfmgr_port <port LANforge GUI HTTP service is running on>
  --test_duration TEST_DURATION
                        --test_duration <how long to run>  example --time 5d (5 days) default: 3m options: number followed by d, h, m
  --tos TOS             --tos:  Support different ToS settings: BK | BE | VI | VO | numeric
  --debug               --debug flag present debug on  enable debugging
  -t ENDP_TYPE, --endp_type ENDP_TYPE
                        --endp_type <types of traffic> example --endp_type "lf_udp lf_tcp mc_udp"  Default: lf_udp , options: lf_udp,
  -u UPSTREAM_PORT, --upstream_port UPSTREAM_PORT
                        --upstream_port <cross connect upstream_port> example: --upstream_port eth1
  --downstream_port DOWNSTREAM_PORT
                        --downstream_port <cross connect downstream_port> example: --downstream_port eth2
  --polling_interval POLLING_INTERVAL
                        --polling_interval <seconds>
  -r RADIO, --radio RADIO
                         --radio "radio==<number_of_wiphy stations=<=number of stations> ssid==<ssid> ssid_pw==<ssid password> securit
  --collect_layer3_data
                        --collect_layer3_data flag present creates csv files recording layer3 columns of cxs.
  --ap_read             --ap_read  flag present enable reading ap
  --ap_scheme {serial,telnet,ssh,mux_serial}
                        --ap_scheme '/dev/ttyUSB0'
  --ap_port AP_PORT     --ap_port '/dev/ttyUSB0'
  --ap_baud AP_BAUD     --ap_baud '115200'',  default='115200
  --ap_ip AP_IP         --ap_ip
  --ap_ssh_port AP_SSH_PORT
                        --ap_ssh_port
  --ap_user AP_USER     --ap_user , the user name for the ap, default = lanforge
  --ap_passwd AP_PASSWD
                        --ap_passwd, the password for the ap default = lanforge
  --ap_if_2g AP_IF_2G   --ap_if_2g eth6
  --ap_if_5g AP_IF_5G   --ap_if_5g eth7
  --ap_if_6g AP_IF_6G   --ap_if_6g eth8
  --ap_cmd_6g AP_CMD_6G
                        ap_cmd_6g 'wl -i wl2 bs_data'
  --ap_cmd_5g AP_CMD_5G
                        ap_cmd_5g 'wl -i wl1 bs_data'
  --ap_cmd_2g AP_CMD_2G
```

```
                        ap_cmd_2g 'wl -i wl0 bs_data'
  --ap_cmd_ul_6g AP_CMD_UL_6G
                        ap_cmd_ul_6g 'wl -i wl2 rx_report'
  --ap_cmd_ul_5g AP_CMD_UL_5G
                        ap_cmd_ul_5g 'wl -i wl1 rx_report'
  --ap_cmd_ul_2g AP_CMD_UL_2G
                        ap_cmd_ul_2g 'wl -i wl0 rx_report'
  --ap_chanim_cmd_6g AP_CHANIM_CMD_6G
                        ap_chanim_cmd_6g 'wl -i wl2 chanim_stats'
  --ap_chanim_cmd_5g AP_CHANIM_CMD_5G
                        ap_chanim_cmd_5g 'wl -i wl1 chanim_stats'
  --ap_chanim_cmd_2g AP_CHANIM_CMD_2G
                        ap_chanim_cmd_2g 'w1 -i wl0 chanim_stats'
  --ap_scheduler_stats  --ap_scheduler_stats flag to clear stats run test then dump ul and dl stats to file
  --ap_ofdma_stats      --ap_ofdma_stats flag to clear stats run test then dumps wl -i wl1 muinfo -v and wl 0i wl0 muinof -v to file
  --ap_test_mode        ap_test_mode flag present use ap canned data
  -amr SIDE_A_MIN_BPS, --side_a_min_bps SIDE_A_MIN_BPS
                        --side_a_min_bps, requested downstream min tx rate, comma separated list for multiple iterations.  Default 256
  -amp SIDE_A_MIN_PDU, --side_a_min_pdu SIDE_A_MIN_PDU
                        --side_a_min_pdu, downstream pdu size, comma separated list for multiple iterations.  Default MTU
  -bmr SIDE_B_MIN_BPS, --side_b_min_bps SIDE_B_MIN_BPS
                        --side_b_min_bps, requested upstream min tx rate, comma separated list for multiple iterations.  Default 25600
  -bmp SIDE_B_MIN_PDU, --side_b_min_pdu SIDE_B_MIN_PDU
                        --side_b_min_pdu, upstream pdu size, comma separated list for multiple iterations. Default MTU
  --rates_are_totals    Treat configured rates as totals instead of using the un-modified rate for every connection.
  --multiconn MULTICONN
                        Configure multi-conn setting for endpoints.  Default is 1 (auto-helper is enabled by default as well).
  --attenuators ATTENUATORS
                        --attenuators,  comma separated list of attenuator module eids:  shelf.resource.atten-serno.atten-idx
  --atten_vals ATTEN_VALS
                        --atten_vals,  comma separated list of attenuator settings in ddb units (1/10 of db)
  --influx_host INFLUX_HOST
                        Hostname for the Influx database
  --influx_port INFLUX_PORT
                        IP Port for the Influx database
  --influx_org INFLUX_ORG
                        Organization for the Influx database
  --influx_token INFLUX_TOKEN
                        Token for the Influx database
  --influx_bucket INFLUX_BUCKET
                        Name of the Influx bucket
  --influx_tag INFLUX_TAG INFLUX_TAG
                        --influx_tag <key> <val>   Can add more than one of these.
  --cap_ctl_out         --cap_ctl_out, switch the controller output will be captured
  --wait WAIT           --wait <time> , time to wait at the end of the test
  --sta_start_offset STA_START_OFFSET
                        Station start offset for building stations
  --no_pre_cleanup      Do not pre cleanup stations on start
  --no_cleanup          Do not cleanup before exit
  --no_stop_traffic     leave traffic running
  --use_existing_station_list
                        --use_station_list ,full eid must be given,the script will use stations from the list, no configuration on the
  --existing_station_list EXISTING_STATION_LIST
                        --station_list [list of stations] , use the stations in the list , multiple station lists may be entered
  --wait_for_ip_sec WAIT_FOR_IP_SEC
                        --wait_for_ip_sec <seconds>  default : 120s
  --log_level LOG_LEVEL
                        Set logging level: debug | info | warning | error | critical
  --lf_logger_config_json LF_LOGGER_CONFIG_JSON
                        --lf_logger_config_json <json file> , json configuration of logger

        Useful Information:
            1. Polling interval for checking traffic is fixed at 1 minute
            2. The test will generate csv file
            3. The tx/rx rates are fixed at 256000 bits per second
            4. Maximum stations per radio based on radio
```

py-scripts/sta_connect2.py

```
usage: sta_connect2.py [-h] [-m MGR] [-o PORT] [-u USER] [-p PASSWD]
                       [--resource RESOURCE]
                       [--upstream_resource UPSTREAM_RESOURCE]
                       [--upstream_port UPSTREAM_PORT] [--radio RADIO]
                       [--sta_mode STA_MODE] [--dut_ssid DUT_SSID]
                       [--dut_security DUT_SECURITY] [--dut_passwd DUT_PASSWD]
                       [--dut_bssid DUT_BSSID] [--download_bps DOWNLOAD_BPS]
                       [--upload_bps UPLOAD_BPS] [--side_a_pdu SIDE_A_PDU]
                       [--side_b_pdu SIDE_B_PDU] [--runtime_sec RUNTIME_SEC]
                       [--debug] [--prefix PREFIX]
                       [--bringup_time BRINGUP_TIME]
                       [--influx_user INFLUX_USER]
                       [--influx_passwd INFLUX_PASSWD] [--influx_db INFLUX_DB]
                       [--influx_host INFLUX_HOST]
                       [--monitor_interval MONITOR_INTERVAL]
                       [--debug_log DEBUG_LOG] [--no_cleanup]
                       [--local_lf_report_dir LOCAL_LF_REPORT_DIR]
                       [--test_rig TEST_RIG] [--test_tag TEST_TAG]
                       [--dut_hw_version DUT_HW_VERSION]
                       [--dut_sw_version DUT_SW_VERSION]
                       [--dut_model_num DUT_MODEL_NUM]
                       [--dut_serial_num DUT_SERIAL_NUM]
                       [--test_priority TEST_PRIORITY] [--test_id TEST_ID]
                       [--csv_outfile CSV_OUTFILE] [--log_level LOG_LEVEL]
                       [--lf_logger_config_json LF_LOGGER_CONFIG_JSON]

---------------------------
LANforge Unit Test:  Connect Station to AP - sta_connect2.py
---------------------------
Summary:
This will create a station, create TCP and UDP traffic, run it a short amount of time, and verify whether traffic
was sent and received.  It also verifies the station connected to the requested BSSID if bssid is specified
as an argument. The script will clean up the station and connections at the end of the test.
---------------------------
CLI Example:
./sta_connect2.py --mgr localhost --dut_ssid <ssid> --dut_passwd <passwd> --dut_security wpa2
--upstream_port eth1 --radio wiphy1

CLI Example for kpi.csv report output:
./sta_connect2.py --mgr localhost --dut_ssid <ssid> --dut_passwd <passwd> --dut_security wpa2
--upstream_port eth2  --csv_outfile sta_connect2.csv --test_rig LF-Lab --test_tag L3 --dut_hw_version Linux
```

```
--dut_model_num 1 --dut_sw_version 5.4.5 --dut_serial_num 1234

CLI Example for kpi.csv, variable tx/rx rates, and pdu size:
./sta_connect2.py --mgr localhost --dut_ssid <ssid> --dut_passwd <passwd> --dut_security wpa2
--upstream_port eth2  --download_bps 768000 --upload_bps 256000 --side_a_pdu 300 --side_b_pdu 750
--csv_outfile sta_connect2.csv --test_rig LF-Lab --test_tag L3 --dut_hw_version Linux --dut_model_num 1
--dut_sw_version 5.4.5 --dut_serial_num 1234

Note: --sta_mode use values in second column

        AUTO        |  0        #  802.11g
        802.11a     |  1        #  802.11a
        b           |  2        #  802.11b
        g           |  3        #  802.11g
        abg         |  4        #  802.11abg
        abgn        |  5        #  802.11abgn
        bgn         |  6        #  802.11bgn
        bg          |  7        #  802.11bg
        abgnAC      |  8        #  802.11abgn-AC
        anAC        |  9        #  802.11an-AC
        an          | 10        #  802.11an
        bgnAC       | 11        #  802.11bgn-AC
        abgnAX      | 12        #  802.11abgn-AX
                                #     a/b/g/n/AC/AX (dual-band AX) support
        bgnAX       | 13        #  802.11bgn-AX
        anAX        | 14        #  802.11an-AX
        aAX         | 15        #  802.11a-AX (6E disables /n and /ac)

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

options:
  -h, --help            show this help message and exit
  -m MGR, --mgr MGR     address of the LANforge GUI machine (localhost is default)
  -o PORT, --port PORT  IP Port the LANforge GUI is listening on (8080 is default)
  -u USER, --user USER  TBD: credential login/username
  -p PASSWD, --passwd PASSWD
                        TBD: credential password
  --resource RESOURCE   LANforge Station resource ID to use, default is 1
  --upstream_resource UPSTREAM_RESOURCE
                        LANforge Ethernet port resource ID to use, default is 1
  --upstream_port UPSTREAM_PORT
                        LANforge Ethernet port name, default is eth2
  --radio RADIO         LANforge radio to use, default is wiphy0
  --sta_mode STA_MODE   LANforge station-mode setting (see add_sta LANforge CLI documentation, default is 0 (auto))
  --dut_ssid DUT_SSID   DUT SSID
  --dut_security DUT_SECURITY
                        DUT security: open, wpa, wpa2, wpa3, owe
  --dut_passwd DUT_PASSWD
                        DUT PSK password.  Do not set for OPEN auth
  --dut_bssid DUT_BSSID
                        DUT BSSID to which we expect to connect.
  --download_bps DOWNLOAD_BPS
                        Set the minimum bps value on test endpoint A. Default: 25g000
  --upload_bps UPLOAD_BPS
                        Set the minimum bps value on test endpoint B. Default: 256000
  --side_a_pdu SIDE_A_PDU
                        Set the minimum pdu value on test endpoint A. Default: 1200
  --side_b_pdu SIDE_B_PDU
                        Set the minimum pdu value on test endpoint B. Default: 1500
  --runtime_sec RUNTIME_SEC
                        Set test duration time. Default: 60 seconds
  --debug               enable debugging
  --prefix PREFIX       Station prefix. Default: 'sta'
  --bringup_time BRINGUP_TIME
                        Seconds to wait for stations to associate and aquire IP. Default: 300
  --influx_user INFLUX_USER
                        Username for your Influx database
  --influx_passwd INFLUX_PASSWD
                        Password for your Influx database
  --influx_db INFLUX_DB
                        Name of your Influx database
  --influx_host INFLUX_HOST
                        Host of your influx database if different from the system you are running on
  --monitor_interval MONITOR_INTERVAL
                        How frequently you want to append to your database
  --debug_log DEBUG_LOG
                        Specify a file to send debug output to
  --no_cleanup          Do not cleanup before exit
  --local_lf_report_dir LOCAL_LF_REPORT_DIR
                        --local_lf_report_dir override the report path, primary use when running test in test suite
  --test_rig TEST_RIG   test rig for kpi.csv, testbed that the tests are run on
  --test_tag TEST_TAG   test tag for kpi.csv,  test specific information to differenciate the test
  --dut_hw_version DUT_HW_VERSION
                        dut hw version for kpi.csv, hardware version of the device under test
  --dut_sw_version DUT_SW_VERSION
                        dut sw version for kpi.csv, software version of the device under test
  --dut_model_num DUT_MODEL_NUM
                        dut model for kpi.csv,  model number / name of the device under test
  --dut_serial_num DUT_SERIAL_NUM
                        dut serial for kpi.csv, serial number / serial number of the device under test
  --test_priority TEST_PRIORITY
                        dut model for kpi.csv,  test-priority is arbitrary number
  --test_id TEST_ID     test-id for kpi.csv,  script or test name
  --csv_outfile CSV_OUTFILE
                        --csv_outfile <prepend input to generated file for csv data>
  --log_level LOG_LEVEL
                        Set logging level: debug | info | warning | error | critical
  --lf_logger_config_json LF_LOGGER_CONFIG_JSON
                        --lf_logger_config_json <json file> , json configuration of logger

---------------------------
CLI Example:
./sta_connect2.py --mgr localhost --dut_ssid <ssid> --dut_passwd <passwd> --dut_security wpa2
--upstream_port eth2 --radio wiphy1
```

py-scripts/lf_interop_ping.py

Example report: interop_ping.pdf

```
usage: interop_ping.py [-h] --mgr MGR --target TARGET --ping_interval
                       PING_INTERVAL --ping_duration PING_DURATION
                       [--ssid SSID] [--mgr_port MGR_PORT]
                       [--mgr_passwd MGR_PASSWD] [--security SECURITY]
                       [--passwd PASSWD] [--virtual] [--num_sta NUM_STA]
                       [--radio RADIO] [--real] [--debug]

        NAME: lf_interop_ping.py

        PURPOSE: lf_interop_ping.py will let the user select real devices, virtual devices or both and then allows them to run
        ping test for user given duration and packet interval on the given target IP or domain name.

        EXAMPLE-1:
        Command Line Interface to run ping test with only virtual clients
        python3 lf_interop_ping.py --mgr 192.168.200.103  --target 192.168.1.3 --virtual --num_sta 1 --radio 1.1.wiphy2 --ssid RDT_wpa
        --passwd OpenWifi --ping_interval 1 --ping_duration 1 --debug

        EXAMPLE-2:
        Command Line Interface to run ping test with only real clients
        python3 lf_interop_ping.py --mgr 192.168.200.103 --real --target 192.168.1.3 --ping_interval 1 --ping_duration 1

        EXAMPLE-3:
        Command Line Interface to run ping test with both real and virtual clients
        python3 lf_interop_ping.py --mgr 192.168.200.103 --target 192.168.1.3 --real --virtual --num_sta 1 --radio 1.1.wiphy2 --ssid R
        --passwd OpenWifi --ping_interval 1 --ping_duration 1

        SCRIPT_CLASSIFICATION : Test

        SCRIPT_CATEGORIES: Performance, Functional, Report Generation

        NOTES:
        1.Use './lf_interop_ping.py --help' to see command line usage and options
        2.Please pass ping_duration in minutes
        3.Please pass ping_interval in seconds
        4.After passing the cli, if --real flag is selected, then a list of available real devices will be displayed on the terminal.
        5.Enter the real device resource numbers seperated by commas (,)

        STATUS: BETA RELEASE

        VERIFIED_ON:
        Working date    - 20/09/2023
        Build version   - 5.4.7
        kernel version  - 6.2.16+

        License: Free to distribute and modify. LANforge systems must be licensed.
        Copyright 2023 Candela Technologies Inc.


options:
  -h, --help            show this help message and exit

Required arguments:
  --mgr MGR             hostname where LANforge GUI is running
  --target TARGET       Target URL for ping test
  --ping_interval PING_INTERVAL
                        Interval (in seconds) between the echo requests
  --ping_duration PING_DURATION
                        Duration (in minutes) to run the ping test

Optional arguments:
  --ssid SSID           SSID for connecting the stations
  --mgr_port MGR_PORT   port on which LANforge HTTP service is running
  --mgr_passwd MGR_PASSWD
                        Password to connect to LANforge GUI
  --security SECURITY   Security protocol for the specified SSID: <open | wep | wpa | wpa2 | wpa3>
  --passwd PASSWD       passphrase for the specified SSID
  --virtual             specify this flag if the test should run on virtual clients
  --num_sta NUM_STA     specify the number of virtual stations to be created.
  --radio RADIO         specify the radio to create the virtual stations
  --real                specify this flag if the test should run on real clients
  --debug               Enable debugging

            Allows user to run the ping test on a target IP for the given duration and packet interval
            with either selected number of virtual stations or provides the list of available real devices
            and allows the user to select the real devices and run ping test on them.
```