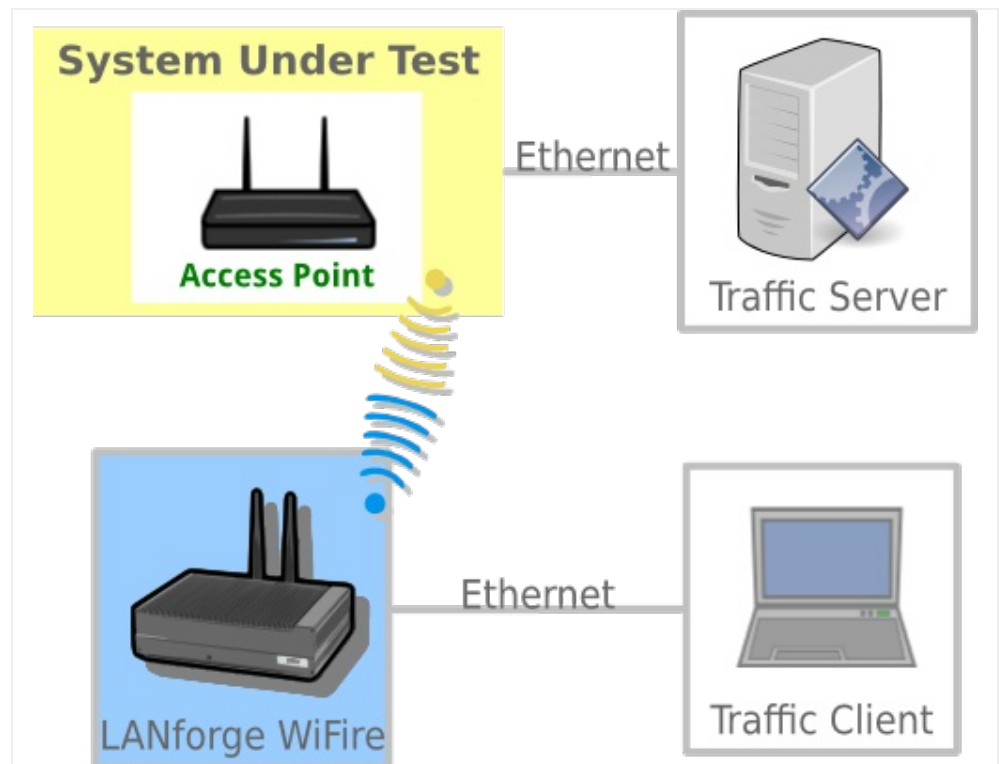


## Generate WiFi Traffic from an External Connection

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

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

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



### 1. Find the MAC Address of the Windows PC.

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

For more information see [Windows IP Addresses](#)

2. Configure your LANforge Wireless Station. We will be configuring the MAC addresses of a station to be the MAC address of the Windows PC we just found.

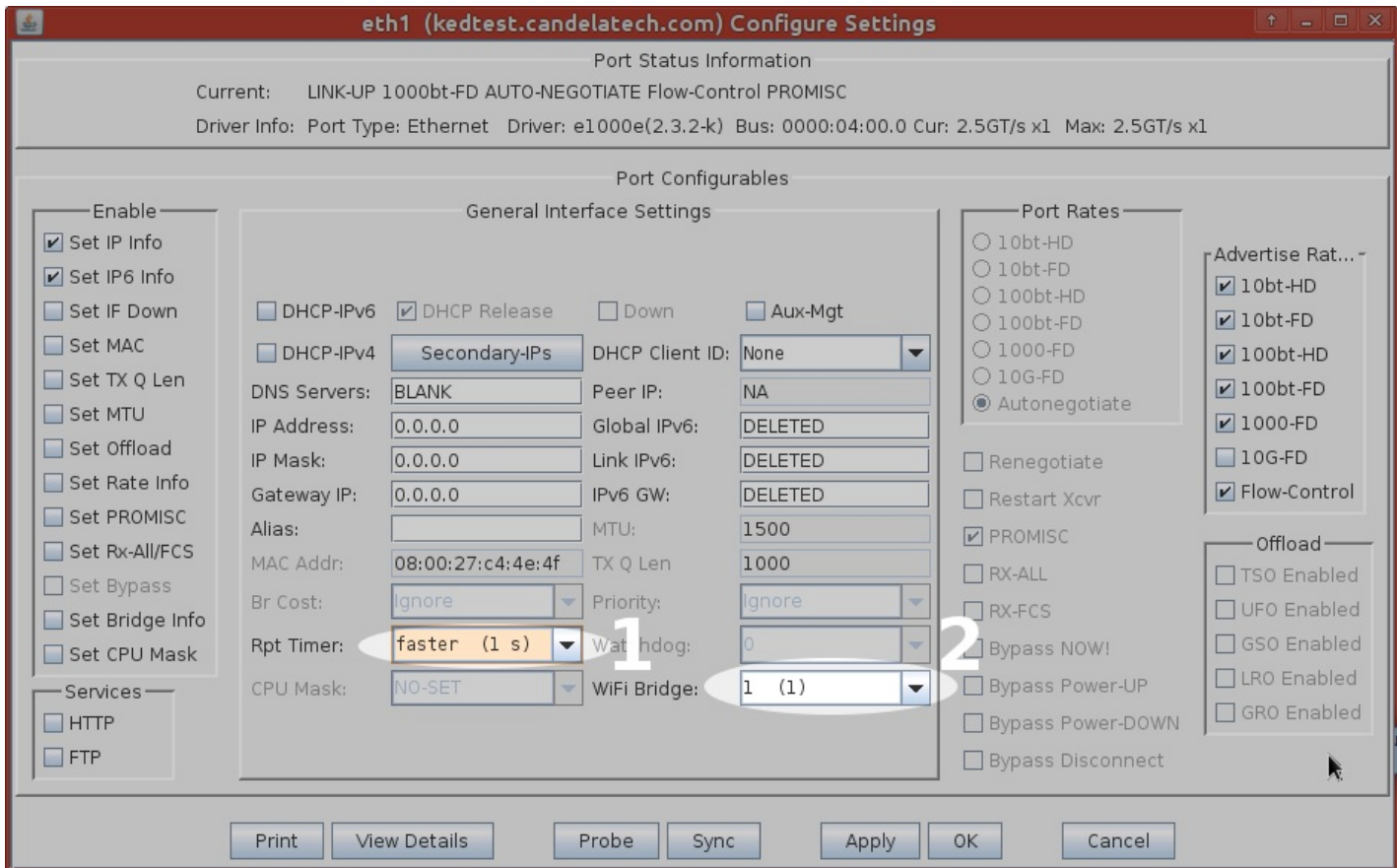
A. In the *Ports* tab, double-click on the ethernet port on the same switch as the Windows PC (*eth1* in this example).

The screenshot shows the LANforge Manager interface. The 'Port Mgr' tab is active, displaying a table of Ethernet interfaces for all resources. The table includes columns for Port, Phase, Down status, IP, SEC, Alias, Parent Dev, RX Bytes, RX Pkts, Pps RX, bps RX, TX Bytes, TX Pkts, and Pps TX. The interface also features various control buttons like 'Stop All', 'Restart Manager', 'Refresh', and 'HELP' at the top, and a 'Control' menu with options like 'Reporting', 'Tear-Off', 'Info', and 'Plugins'. A status bar at the bottom indicates the user is logged in as Admin.

Port	Pha...	Down	IP	SEC	Alias	Parent Dev	RX Bytes	RX Pkts	Pps RX	bps RX	TX Bytes	TX Pkts	Pps TX
1.2.0	<input type="checkbox"/>	<input type="checkbox"/>	192.168.100.42	0	eth0		2,967,835,...	24,653,...	10	8,851	58,977,585...	45,381,...	13
1.2.1	<input type="checkbox"/>	<input type="checkbox"/>	0.0.0.0	0	eth1		5,652,446,...	3,877,467	0	245	18,835,970...	12,408,...	0
1.2.10	<input type="checkbox"/>	<input type="checkbox"/>	0.0.0.0	0	wiphy2		3,269,375,...	2,943,4...	0	0	4,304,428,...	4,184,586	0
1.2.2	<input type="checkbox"/>	<input type="checkbox"/>	0.0.0.0	0	sta0	wiphy0	64,609	311	0	0	647,686	2,672	0
1.2.6	<input type="checkbox"/>	<input type="checkbox"/>	0.0.0.0	0	wiphy0		1,388,554,...	344,068...	4	5,168	7,318,205,...	9,304,864	0
1.2.8	<input type="checkbox"/>	<input type="checkbox"/>	0.0.0.0	0	wiphy1		2,326,554,...	670,035...	0	0	1,269,984,...	1,978,064	0

Logged in to: 192.168.100.26:4002 as: Admin

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



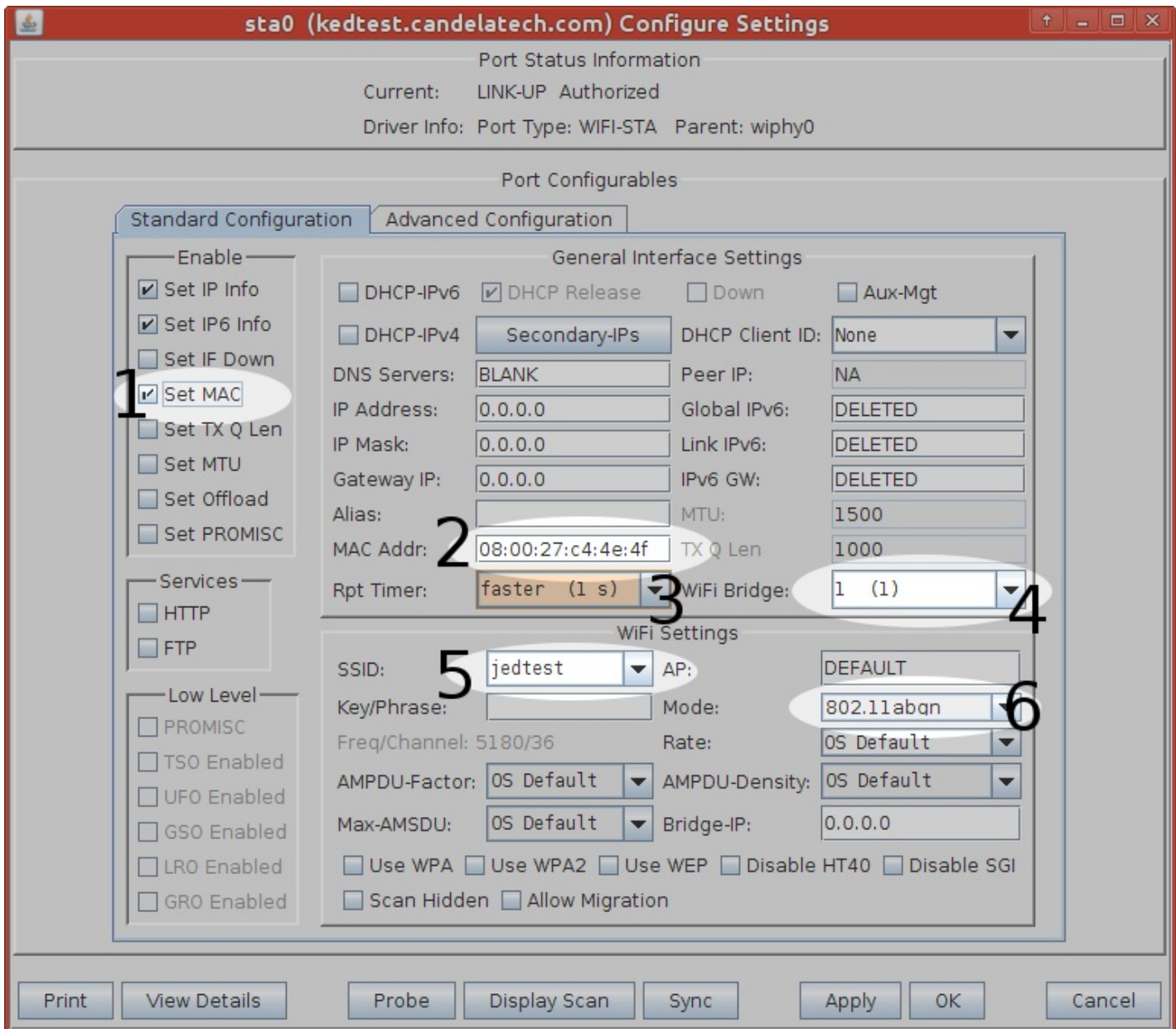
A. Do not set the mac address or the ip address of the port

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

C. Select **1** for *WiFi Bridge*

D. Click the **OK** button

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



A. Enable **Set MAC**

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

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

D. Select **1** for *WiFi Bridge*

E. Enter **jedtest** for the *SSID*

F. Select **802.11abgn** for the *Mode*

G. Click the **OK** button

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

### 3. Sending and Validating Traffic

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

```
C:\Windows\system32\cmd.exe

C:\Users\IEUser>ping -S 10.26.0.10 10.26.0.2_
```

```
C:\Windows\system32\cmd.exe

C:\Users\IEUser>ping -S 10.26.0.10 10.26.0.2

Pinging 10.26.0.2 from 10.26.0.10 with 32 bytes of data:
Reply from 10.26.0.2: bytes=32 time=29ms TTL=64
Reply from 10.26.0.2: bytes=32 time=6ms TTL=64
Reply from 10.26.0.2: bytes=32 time=4ms TTL=64
Reply from 10.26.0.2: bytes=32 time=6ms TTL=64

Ping statistics for 10.26.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 29ms, Average = 11ms

C:\Users\IEUser>
```

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

```
lanforge@kedtest:~

lanforge@kedtest ~
> sudo tcpdump -ni sta0 icmp
tcpdump: WARNING: sta0: no IPv4 address assigned
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on sta0, link-type EN10MB (Ethernet), capture size 65535 bytes
15:22:53.692453 IP 10.26.0.10 > 10.26.0.2: ICMP echo request, id 1, seq 85, length 40
15:22:53.701790 IP 10.26.0.2 > 10.26.0.10: ICMP echo reply, id 1, seq 85, length 40
15:22:54.698267 IP 10.26.0.10 > 10.26.0.2: ICMP echo request, id 1, seq 86, length 40
15:22:54.701143 IP 10.26.0.2 > 10.26.0.10: ICMP echo reply, id 1, seq 86, length 40
15:22:55.728712 IP 10.26.0.10 > 10.26.0.2: ICMP echo request, id 1, seq 87, length 40
15:22:55.732838 IP 10.26.0.2 > 10.26.0.10: ICMP echo reply, id 1, seq 87, length 40
15:22:56.744295 IP 10.26.0.10 > 10.26.0.2: ICMP echo request, id 1, seq 88, length 40
15:22:56.748242 IP 10.26.0.2 > 10.26.0.10: ICMP echo reply, id 1, seq 88, length 40
^C
8 packets captured
8 packets received by filter
0 packets dropped by kernel

lanforge@kedtest ~
>
```

