Hotspot Architecture Tests

Test 1: Run Two CoovaChilli Instances

Goal

Verify that you can run two independent CoovaChilli instances using separate config files, each bound to a different interface (like `eth0.100` and `eth0.200`).

Step-by-Step Process

1. Create VLAN Interfaces

```
sudo ip link add link eth0 name eth0.100 type vlan id 100
sudo ip link add link eth0 name eth0.200 type vlan id 200
sudo ip link set eth0.100 up
sudo ip link set eth0.200 up
```

2. Create Config for `cafe_latte` (eth0.100)

```
sudo tee /etc/chilli/cafe_latte.conf > /dev/null <<EOF
dhcpif=eth0.100
net=192.168.100.0/24
uamlisten=192.168.100.1
uamport=3990
uamserver=http://192.168.100.1:3990
uamsecret=secret123
radiusserver1=127.0.0.1
radiussecret=testing123
radiusnasid=cafe_latte
cmdsocket=/usr/local/var/run/chilli.cafe_latte.sock
EOF
```

3. Create Config for `techfair` (eth0.200)

```
sudo tee /etc/chilli/techfair.conf > /dev/null <<EOF
dhcpif=eth0.200
net=192.168.200.0/24
uamlisten=192.168.200.1
uamport=4990
uamserver=http://192.168.200.1:4990
uamsecret=secret123
radiusserver1=127.0.0.1
radiussecret=testing123
radiusnasid=techfair
cmdsocket=/usr/local/var/run/chilli.techfair.sock
EOF
```

4. Start Both Instances

You can run them manually like this:

```
sudo /usr/local/sbin/chilli -c /etc/chilli/cafe_latte.conf -f -d &
sudo /usr/local/sbin/chilli -c /etc/chilli/techfair.conf -f -d &
```

5. Verify They're Running

Check processes:

```
ps aux | grep chilli
```

Check DHCP leases:

```
ip a show tun0
ip a show tun1
```

Test 1: Run Two CoovaChilli Instances — PASSED

- `eth0.100` → `tun0` → `192.168.100.1`
- `eth0.200` → `tun1` → `192.168.200.1`
- Unique `cmdsocket` per instance
- No port or interface collision

Test 2: FreeRADIUS Authentication

Goal

Confirm that CoovaChilli is:

- Sending authentication requests to FreeRADIUS
- Receiving responses
- Logging any user-level actions (accept/reject)

Test 2: FreeRADIUS Authentication — PASSED

- CoovaChilli → RADIUS request → custom **SQL-based** backend
- RADIUS DB client config recognized CoovaChilli on `192.168.100.1`
- `radtest` returned **Access-Accept**, verifying secrets, NAS, and credentials based on guest service

Test 3: Simulated Captive Portal Traffic Through Squid (Without a Portal Page)

Goal

Validate that **HTTP traffic from clients connected via CoovaChilli interface** (e.g. `tun0`) is **intercepted and logged** by **Squid proxy**.

- Assigning a static IP to a test container, VM, or namespace.
- Setting default gateway = `192.168.100.1` (chilli side).
- Sending HTTP traffic (`curl` or browser) to an external site.
- Checking if Squid logs the access.