**Readme**

**Dependencies**

* create\_ap
* Wireshark
* hostapd

**Installation**

### **create\_ap**

**git clone https://github.com/oblique/create\_ap**

**cd create\_ap**

**make install**

### **Wireshark and hostapd**

**sudo apt install hostapd wireshark**

**Create an Access Point (AP)**

A standard AP can be created using the command:

**sudo create\_ap wlan1 wlan0 testnetwork abcdefgh -g 207.241.237.3**

This will create a Wi-Fi network called testnetwork with password **abcdefgh**. The arguments **wlan1** and **wlan0** depend on your machine:

* The argument **wlan0** refers to your built-in network card and may be different on your machine. Find out this name by executing `**ip addr**` and picking the interface that is assigned an IP address.
* The argument **wlan1** refers to the Wi-Fi dongle you plugged in. Find out its name on your machine by executing `**ip addr**` before and after plugging in the Wi-Fi dongle and seeing which interface was added.

**Preparation**

1. Download and install target VPN clients on either mobile or desktop machine.
2. Connect the devices, e.g., cell phones or laptops to the created AP.
3. Enable the VPN connection on the mobile or desktop machine.

**Open Wireshark**

**sudo wireshark**

To inspect the traffic of any client connected to the AP, start Wireshark on the test platform and listen for packets on the virtual interface (or on the interface of the Wi-Fi dongle in case it doesn’t support virtual interfaces).

**Errors and warnings:**

* If you get the error “ERROR: Failed to initialize lock” then execute:   
  **sudo rm /tmp/create\_ap.all.lock**
* If you get the warning “Your adapter does not fully support AP virtual interface” this means your Wi-Fi dongle cannot simultaneously act as a client and AP. This shouldn’t be a problem in our experiments.

**Test LocalNet Attacks**

1. Open Wireshark to monitor the traffic.
2. Then try to visit ‘http://207.241.237.3’ in a browser.
3. Use the Wireshark filter ‘tcp.flags.syn == 1’ to filter for plaintext TCP SYN packets.
4. If TCP SYN packets can be seen as ‘207.241.237.3’, this means that the VPN app is vulnerable.

**Test ServerIP Attacks**

1. Open Wireshark to monitor the packets and start the ‘create\_ap’ script on the test platform. Then connect the device being tested with the created AP.
2. After starting to capture frames, connect the device under test to the VPN server and then use Wireshark to identify the VPN server’s IP address ($VPN\_SERVERIP). Then visit ‘[http://$VPN\_SERVERIP](about:blank)’.
   * Alternatively, one may use the configuration file for the third-party VPN client to identify the VPN server’s IP address by pinging the domain name provided in the configuration file (ex. for **remote str-dxb101.strongconnectivity.com 1194 tcp** → **ping str-dxb101.strongconnectivity.com**) .
3. Use the Wireshark filter ‘tcp.flags.syn == 1’ to filter for plaintext TCP SYN packets. If there are no plaintext TCP SYNs in Wireshark, then the VPN client is not vulnerable.   
   If the VPN protocol is using TCP or UDP, then also try to visit `http://$VPN\_SERVERIP:$PORT` incl. the port used by the server (typically $PORT = 80 for http traffic).
4. In order to test for plaintext DNS, one may further run the following test: In case there are no plaintext TCP SYN packets, the next step is to test whether the VPN client used plaintext DNS to find the VPN server’s IP address. To determine this, use the Wireshark filter ‘dns.a == $VPN\_SERVERIP’. If there are any results, then the VPN client is highly likely vulnerable.