Ex150 Report

Benjamin Ruddy

2022-12-12

Contents

Goal

The goal in this exercise was to exploit a WPA2-EAP wireless network.

Technical Report

Finding: *EAP configuration on wireless network does not validate AP certificates, leading to potential evil twin attack*

Severity Rating: 6.0

CVSS Base Severity Rating: 6.0 AV:A AC:H PR:N UI:R S:C C:H I:L A:N

Vulnerability Description

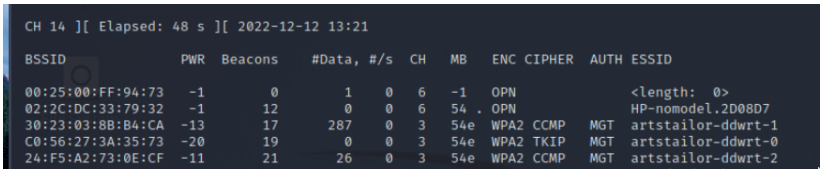
The configuration of Art's Tailor's EAP deployment on their wireless network does not require validation of Access Point (AP) certificates. This creates the possibility for a malicious AP to be configured with the same name as a legitimate AP, that wirless users may subsquently connect to and inadvertently provide their network authentication credentials to.

Confirmation method

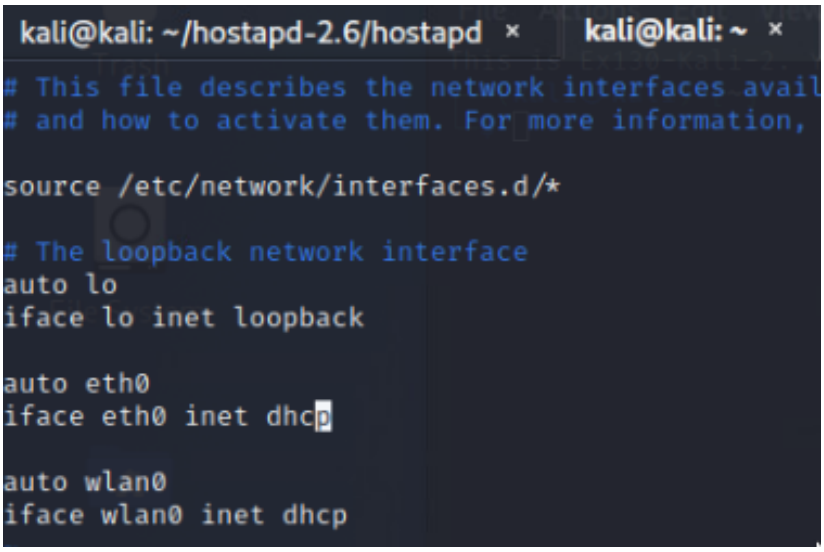
With a monitor-mode capable device, run the following commands to check for the locally available APs from Art's Tailor:

```
sudo airmon-ng check kill
sudo airmon-ng start wlan0
sudo airodump-ng wlan0mon
```

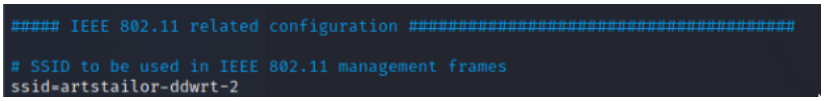
Confirm the channel of the desired AP given by the last command:



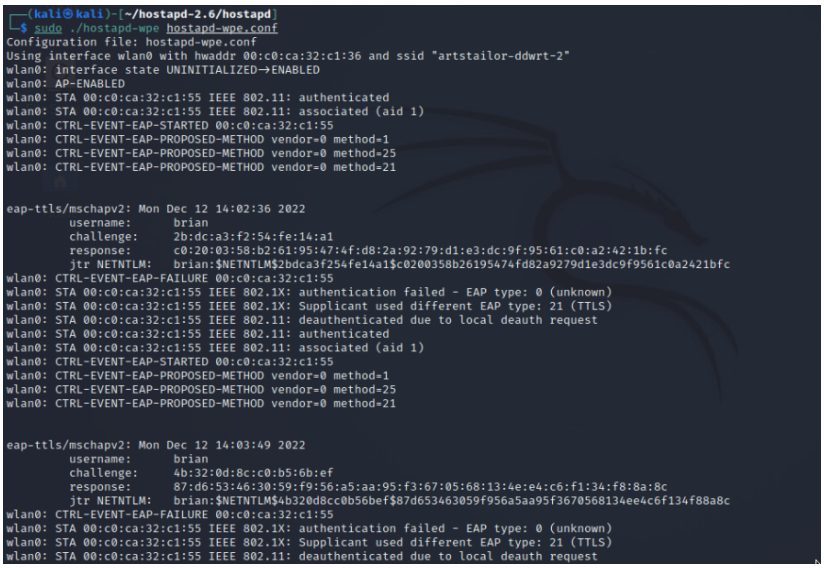
Next, restart the attacker host. run `sudo airmon-ng check kill` once more, and then specify the network interfaces manually in `/etc/network/interfaces`:



Now, c reate a `hostpad-wpe.conf` file with the following:



...and run it as follows:



As seen in the screenshot above, hashed credentials for the legitimate AP were captured. These can be subsequently cracked for an attacker to authenticate with to the real network (see Attack Narrative for details).

Mitigation or Resolution Strategy

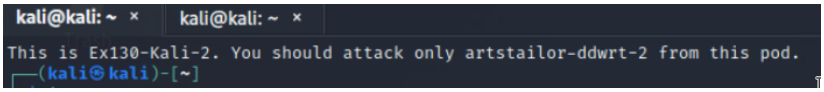
Enable AP certificate validation for the current deployment of EAP, and ensure that clients are required to validate said certificate upon connecting.

Consult section 14.4 of EAP-TTLS RFC (https://www.rfc-editor.org/rfc/rfc5281#section-14.4) for more information.

Attack Narrative

Enabling monitor mode, finding AP channel number

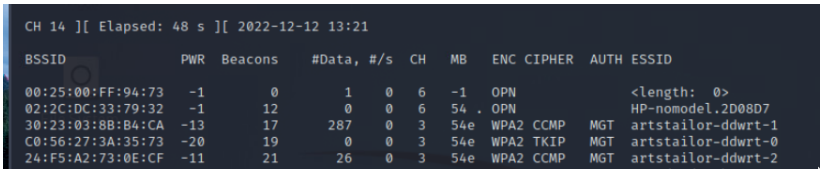
We are tasked with finding one of three access points (APs) along with their channel numbers. In this case, the target access point is `artstailor-ddwrt-2`.



With a dedicated monitor-mode capable wireless adapter, the commands that we execute to scan for access points and their channels are:

```
sudo airmon-ng check kill
sudo airmon-ng start wlan0
sudo airodump-ng wlan0mon
```

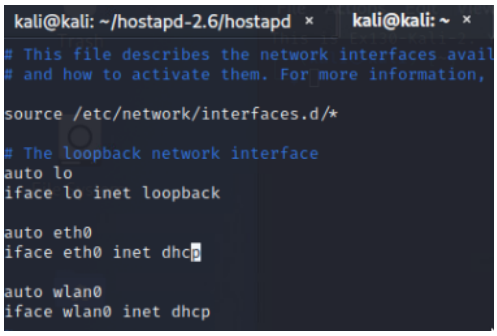
As seen in the below `airodump-ng` output, the channel number for `artstailor-ddwrt-2` is 3.



Deploying a fake RADIUS to capture credentials with hostapd-wpe

Our goal is now to set up an access point with the same name as artstailor-ddwrt-2 to see if we can capture credentials from users that may be attempting authentication to it. We do so with the help of `hostpad-wpe`.

But first, we have to once again kill any processes that may be interfering with wireless communications with `sudo airmon-ng check kill`, after which we have to specify our network interfaces manually in the `/etc/network/interfaces` file:



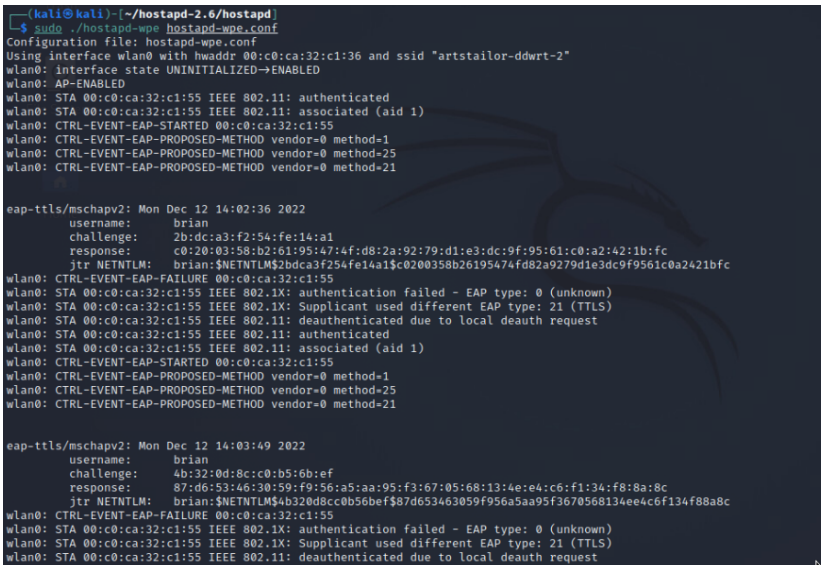
Now, we specify the SSID that we want to create a malicious RADIUS server for in the (`hostpad-wpe.conf` file) and run hostapd-wpe as follows:

```
##### IEEE 802.11 related configuration #####################################

# SSID to be used in IEEE 802.11 management frames
ssid=artstailor-ddwrt-2
```

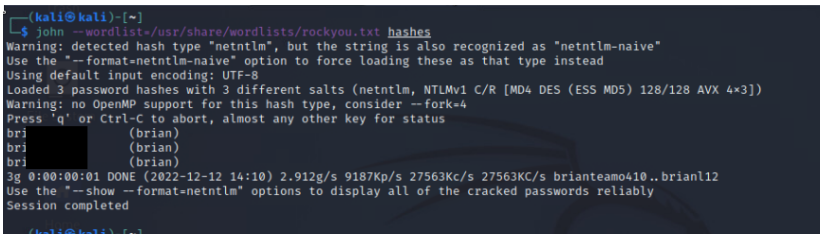
```
./hostpad-wpe hostpad-wpe.conf
```

Cracking the captured hashes

We were able to get multiple authentication attempts coming from the user `brian`, for which numerous hashes were provided:



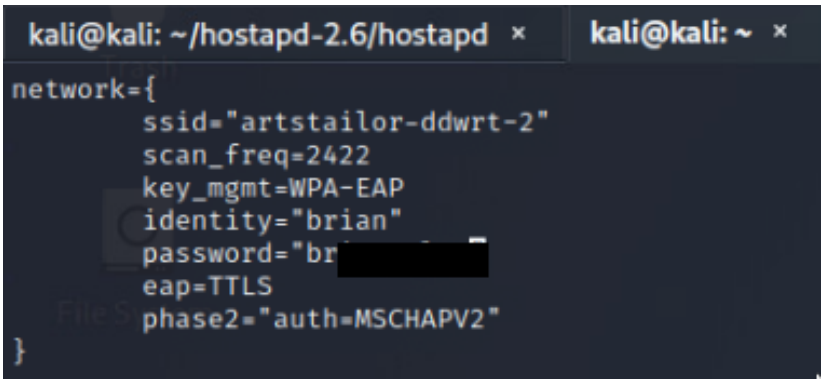
Putting these in a text file and performing a dictionary attack with a popular wordlist via John the Ripper resulted in a successful password crack for Brian's NTLM hash:



Connecting to the network with the captured credentials

To make a connection with our cracked credentials, we need to make use of `wpa_supplicant` in our attacking Linux host since we are connecting to an 802.11 network managed by a centralized RADIUS server.

After some experimentation, and by referncing https://w1.fi/cgit/hostap/plain/wpa_supplicant/wpa_supplicant.conf, we arrived at the following configuration file (stored in `/etc/wpa_supplicant.conf`):



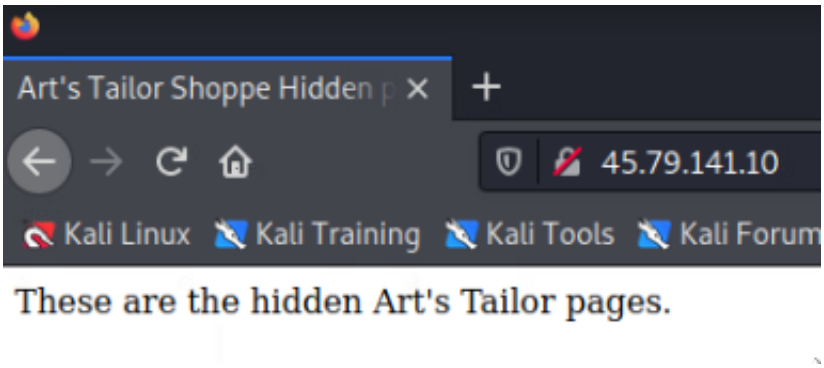
Subsequently, we were able to authenticate to it with the following `wpa_supplicant` command-line options:

Then, we were able to use `dhclient` to request an IP address on the wireless interface, which finally completes our connection to the artstailor-ddwrt-2 AP:

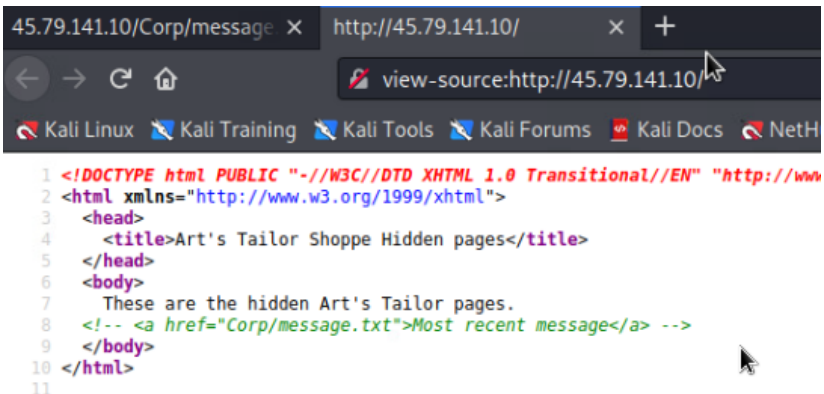
```
┌──(kali㉿kali)-[~]
└─$ sudo dhclient wlan0
[sudo] password for kali:
Sorry, try again.
[sudo] password for kali:

┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:50:56:94:2f:c5 brd ff:ff:ff:ff:ff:ff
    inet 172.24.0.10/24 brd 172.24.0.255 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:fe94:2fc5/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:c0:ca:32:c1:36 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.145/24 brd 192.168.1.255 scope global dynamic wlan0
       valid_lft 86398sec preferred_lft 86398sec
    inet6 fe80::2c0:caff:fe32:c136/64 scope link
       valid_lft forever preferred_lft forever
```

Accessing a hidden page for Art's Tailor

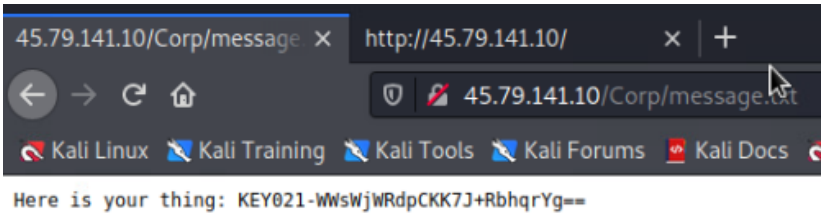
Upon establishing this connection, we are now able to access the page at 45.79.141.10, as mentioned in the briefing, which appears to be a hidden page from Art's Tailor.



Inspecting the source, we see a reference to a file in the `/Corp` directory:



Which upon visiting, grants us `KEY021`:



MITRE ATT&CK Framework TTPs

TA0006: Credential Access
- **T1040:** Network Sniffing
  - **NA:** NA

TA0006: Credential Access
- **T1110:** Brute Force
  - **.002:** Password Cracking