Aircrack-ng

Injection test

Important note: This option is only available on aircrack-ng 0.9 and up.

Description

The injection test determines if your card can successfully inject and determine the ping response times to the Access Point (AP). If you have two wireless cards, it can also determine which specific injection tests can be successfully performed.

The basic injection test provides additional valuable information as well. First, it lists access points in the area which respond to broadcast probes. Second, for each, it does a 30 packet test that indicates the connection quality. This connection quality quantifies the ability of your card to successfully send and then receive a response to the test packet. The percentage of responses received gives an excellent indication of the link quality.

You may optionally specify the AP name and MAC address. This can be used to test a specific AP or test a hidden SSID.

So how does it work? The following will briefly describe how the testing is performed.

The program initially sends out broadcast probe requests. These are probe requests which ask any AP listening to respond with a description of itself. Not every AP will respond to this type of request. A list of responding APs is assembled to be used in subsequent steps. If any AP responds, a messages is printed on the screen indicating that the card can successfully inject.

At the same time, any AP identified via a beacon packet is also added to the list of APs to be processed in subsequent steps.

If a specific AP was optionally listed on the command line (BSSID and SSID), this is also added to the list of APs to be processed.

Then for each AP in the list, 30 directed probe requests are sent out. A directed probe request is addressed to a specific AP. The count of probe responses received plus the percentage is then printed on the screen. This indicates if you can communicate with the AP and how well.

If two wireless cards were specified then each attack mode is tried and the results printed on the screen.

An additional feature is the ability to test connectivity to airserv-ng. Once the basic connectivity test is completed then it proceeds with the standard injection tests via the wireless card linked to airserv-ng.

Usage

aireplay-ng -9 -e teddy -a 00:de:ad:ca:fe:00 -i wlan1 wlan0

Where:

- -9 means injection test. Long form is --test.
- -e teddy is the network name (SSID). This is optional.
- -a 00:de:ad:ca:fe:00 ath0 is MAC address of the access point (BSSID). This is optional.
- -i wlan1 is interface name of the second card if you want to determine which attacks your card supports. This interfaces acts as an AP and receives packets. This is optional.
- wlan0 is the interface name or airserv-ng IP Address plus port number. This interface is used to send packets. For example - 127.0.0.1:666. (Mandatory)

IMPORTANT: You must set your card to monitor mode and to the desired channel with airmon-ng prior to running any of the tests.

Usage Examples

Basic Test

This is a basic test to determine if you card successfully supports injection.

```
aireplay-ng -9 wlan0
```

The system responds:

```
16:29:41  wlan0 channel: 9
16:29:41  Trying broadcast probe requests...
16:29:41  Injection is working!
16:29:42  Found 5 APs

16:29:42  Trying directed probe requests...
16:29:42  00:09:5B:5C:CD:2A - channel: 11 - 'NETGEAR'
16:29:48  0/30: 0%
16:29:48  00:14:BF:A8:65:AC - channel: 9 - 'title'
16:29:54  0/30: 0%
16:29:54  00:14:6C:7E:40:80 - channel: 9 - 'teddy'
16:29:55  Ping (min/avg/max): 2.763ms/4.190ms/8.159ms
16:29:55  27/30: 90%
16:29:55  00:C0:49:E2:C4:39 - channel: 11 - 'mossy'
16:30:01  0/30: 0%
16:30:01  00:0F:66:C3:14:4E - channel: 9 - 'tupper'
16:30:07  0/30: 0%
```

Analysis of the response:

- **16:29:41 wlan0 channel: 9**: This tells you which interface was used and the channel it was running on.
- **16:29:41 Injection is working!**: This confirms your card can inject.
- **16:29:42 Found 5 APs**: These access points (APs) were found either through the broadcast probes or received beacons.
- **16:29:42 00:09:5B:5C:CD:2A - channel: 11 - 'NETGEAR'**: Notice that this AP is on channel 11 and not on our card channel of 9. It is common for adjacent channels to spill over.
- **16:29:55 Ping (min/avg/max): 2.763ms/4.190ms/8.159ms**: It an AP responds with one or more packets then the ping times are calculated.
- **16:29:55 27/30: 90%** for teddy: This is the only AP that the card can successfully communicate with. This is another verification that your card can inject. You will also notice that all the other APs have 0%.

Hidden or Specific SSID

You can check a hidden SSID or check a specific SSID with the following command:

```
aireplay-ng --test -e teddy -a 00:14:6C:7E:40:80 ath0
```

The system responds:

```
11:01:06  ath0 channel: 9
11:01:06  Trying broadcast probe requests...
11:01:06  Injection is working!
11:01:07  Found 1 APs

11:01:07  Trying directed probe requests...
11:01:07  00:14:6C:7E:40:80 - channel: 9 - 'teddy'
11:01:07  Ping (min/avg/max): 2.763ms/4.190ms/8.159ms
11:01:07  30/30: 100%
```

Analysis of the response:

- It confirms that the card can inject and successfully communicate with the specified network.

Attack Tests

This test requires two wireless cards in monitor mode. The card specified by "-i" acts as the access point.

Run the following command:

```
aireplay-ng -9 -i wlan1 wlan0
```

Where:

- -9 means injection test.
- -i wlan1 is the interface to mimic the AP and receives packets.
- wlan0 is the injection interface.

The system responds:

```
11:06:05  wlan0 channel: 9, wlan1 channel: 9
11:06:05  Trying broadcast probe requests...
11:06:05  Injection is working!
11:06:05  Found 1 APs

11:06:05  Trying directed probe requests...
11:06:05  00:de:ad:ca:fe:00 - channel: 9 - 'teddy'
11:06:05  Ping (min/avg/max): 2.763ms/4.190ms/8.159ms
11:06:07  26/30: 87%

11:06:07  Trying card-to-card injection...
11:06:07  Attack -0:        OK
11:06:07  Attack -1 (open): OK
11:06:07  Attack -1 (psk):  OK
11:06:07  Attack -2/-3/-4:  OK
11:06:07  Attack -5:        OK
```

Analysis of the response:

- **11:06:05 wlan0 channel: 9, wlan1 channel: 9**: It is import to make sure both your cards are on the same channel otherwise the tests will not work correctly.
- The first part of the output is identical to what has been presented earlier.
- The last part shows that wlan0 card is able to perform all attack types successfully.
- If you get a failure on attack 5, it may still work in the field if the injection MAC address matches the current card MAC address. With some drivers, it will fail if they are not the same.

Airserv-ng Test

Run Airserv-ng:

```
airserv-ng -d wlan0
```

The system responds:

```
Opening card wlan0
Setting chan 1
Opening sock port 666
Serving wlan0 chan 1 on port 666
```

Then run the following command:

```
aireplay-ng -9 127.0.0.1:666
```

Where:

- -9 means injection test.

- 127.0.0.1:666 is the IP address and port number of airserv-ng. It does not have to be the local loopback address as in this example. It can be any IP address.

The system responds:

```
14:57:23  Testing connection to injection device 127.0.0.1:666
14:57:23  TCP connection successful
14:57:23  airserv-ng found
14:57:23  ping 127.0.0.1:666 (min/avg/max): 0.310ms/0.358ms/0.621ms

Connecting to 127.0.0.1 port 666...
Connection successful

14:57:24  127.0.0.1:666 channel: 9
14:57:24  Trying broadcast probe requests...
14:57:24  Injection is working!
14:57:25  Found 1 AP

14:57:25  Trying directed probe requests...
14:57:25  00:14:6C:7E:40:80 - channel: 9 - 'teddy'
14:57:26  Ping (min/avg/max): 1.907ms/38.308ms/39.826ms
14:57:26  30/30: 100%
```

Analysis of the response:

- **Connection successful**: This indicates that there is connectivity It is import to make sure both your cards are on the same channel otherwise the tests will not work correctly.
- The second part of the output is identical to what has been presented earlier.

Usage Tips

Nothing at this point in time.

Usage Troubleshooting

General

- Make sure you use the correct interface name. For mac80211 drivers, it is typically “mon0”. For madwifi-ng, it is typically “ath0”. As well, ensure you don't have multiple monitor interfaces created meaning “mon0”, “mon1”, etc. is bad and the extra interfaces need to be destroyed.
- Make sure the card(s) are on the same channel as your AP and locked on this channel. When putting your card into monitor mode, be sure to specify the channel via airmon-ng. You can use iwconfig to confirm which channel your card is currently on. The injection test will fail if your card and access point are on different channels.
- Make sure your card is not channel hopping. A very common mistake is to have airodump-ng running in channel hopping mode. If you use airodump-ng, be sure to use the “-c <channel>” option. Additionally, ensure all network managers and similar are killed off.

"Network is down" error message

If you get error messages similar to the following for Atheros-based cards and the madwifi-ng driver:

```
aireplay-ng -9 -e teddy -a 00:14:6C:7E:40:80 -B ath0
Interface ath0 -> driver: Madwifi-NG
12:36:30  ath0 channel: 10
12:36:30  Trying broadcast probe requests...
write failed: Network is down
wi_write(): Network is down
```

Remove the “-B” bitrate option from the request. There is an underlying problem with the madwifi-ng driver concerning

changing bitrates on the fly. Simply put, you cannot currently change bitrates on the fly. A request has been made to the madwifi-ng developers to fix this. Until this is done, you cannot use the "-B" option with madwifi-ng drivers.

Airodump-ng shows APs but they don't respond

The injection test uses broadcast probe requests. Not all APs respond to broadcast probe requests. So the injection test may fail because the APs are ignoring the broadcast packets. As well, you quite often can receive packets from APs further away then your card can transmit to. So the injection test may fail because your card cannot transmit far enough for the AP to receive them.

In both cases, try another channel with multiple APs. Or try the specific SSID test described above.

---

injection_test.txt · Last modified: 2013/04/25 11:17 by jano