WPA Too!

Md Sohail Ahmad, AirTight Networks
md.ahmad@airtightnetworks.com

Abstract

WPA2 is considered as the most secure configuration for WiFi networks. It is widely used to secure enterprise and private WiFi networks. Interestingly, it is also being used to secure guest, municipal and public WiFi networks. In this paper, we present a vulnerability of WPA2 protocol which can be exploited by a malicious user to attack and compromise legitimate users. We also present a few attack mitigation techniques which can be used to protect genuine WiFi users.

I. Introduction

The 802.11i [1] specifies security protocols for WiFi networks. RSN is one of the security configurations available in 802.11i and popularly known as WPA2. WPA2 supports two types of authentication- Pres-Shared Key (PSK) and IEEE 802.1x. For data encryption, WPA2 uses AES though it also supports TKIP. TKIP stands for temporal key integrity protocol and used by old devices which are compliant to WEP encryption. AES stands for advanced encryption system. Most of the current generation WiFi devices support AES.

A couple of attacks on WPA/WPA2 authentication and encryption that have been published in the past are mentioned below:

- PSK vulnerability [2]: PSK is vulnerable to eavesdropping and dictionary attack. To solve PSK vulnerability, it is recommended to use the IEEE 802.1x based authentication.

- PEAP vulnerability [3]: A WiFi client's configuration related vulnerability was identified in 2008. It can be avoided by simply following good practices and by not ignoring certificate validation check in client wireless configuration.

- TKIP vulnerability [4]: TKIP vulnerability allows attacker to guess IP address of the subnet and then inject few small size frames to cause disruption in the network. Fast re-keying or AES can be used to fix the vulnerability.

In the next section, we describe attacks based on a vulnerability of WPA2 protocol and discuss its implications. Finally, we discuss a few solutions to mitigate the attacks.

II. GTK Vulnerability

In WPA2, two types of keys are used for data encryption – PTK and GTK. PTK stands for *pairwise transient key* and it is used to encrypt unicast data traffic. GTK stands for *group temporal key* and it is used to encrypt group addressed data traffic. While PTK is derived per association basis, GTK is randomly generated by AP and sent to all associated clients. GTK is a shared key and is known to all associated clients.

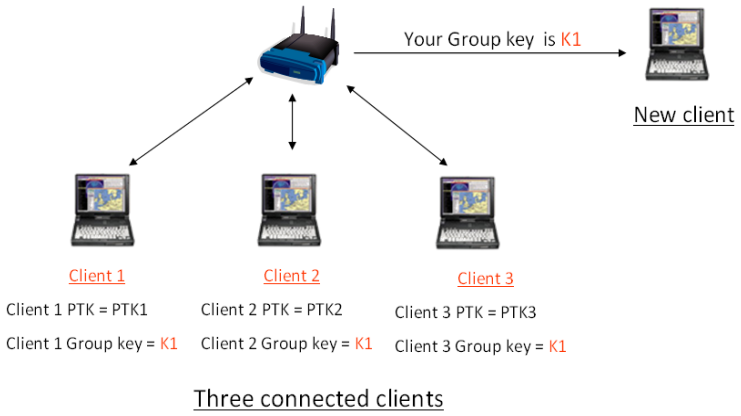


Figure 1: All clients have same copy of GTK "K1"

The purpose of GTK is to be used as an encryption key in AP and as a decryption key in clients. A WiFi client never uses GTK to encrypt data frames as all frames from client to AP are unicast and destined to AP. But a malicious WiFi client can alter its behavior to use GTK to encrypt group addressed data frames of its own and send to all associated clients.

By altering the role of GTK, a malicious client can inject any type of packet to trick attacks in a WLAN e.g. ARP cache poisoning attack. Following attacks are possible on a WPA2 secured WiFi networks:

Attack 1: Stealth ARP cache poisoning/spoofing attack

It allows an attacker to snoop victim's traffic. Attacker may place himself as a Man-in-the-middle (MITM) and steal all his sensitive information. Attacker may launch denial of service (DoS) attack by not serving hosts after poisoning the ARP entry for gateway in all wireless clients.

Step 1: Attacker injects fake ARP packet to poison client's cache for gateway.

Step 2: The ARP cache of victim gets poisoned. For victim, Gateway MAC is now attacker's machine MAC address. Victim's machine sends all traffic to attacker.

Step 3: Attacker can either drop traffic or forward it to actual gateway.

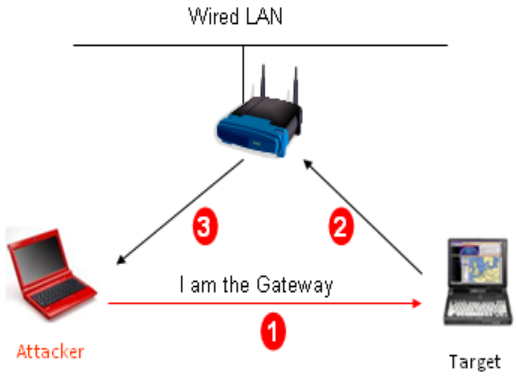


Figure 2: Stealth ARP poisoning

Difference between Normal and Stealth mode ARP poisoning

In normal ARP poisoning [9], injected frames may appear on wire via AP as shown in the figure below. Chance of being detected by wired monitoring tools is very high.

In stealth mode ARP poisoning, injected frames are invisible to AP, never go on wire. Hence it can't be detected by network based ARP cache monitoring tool.

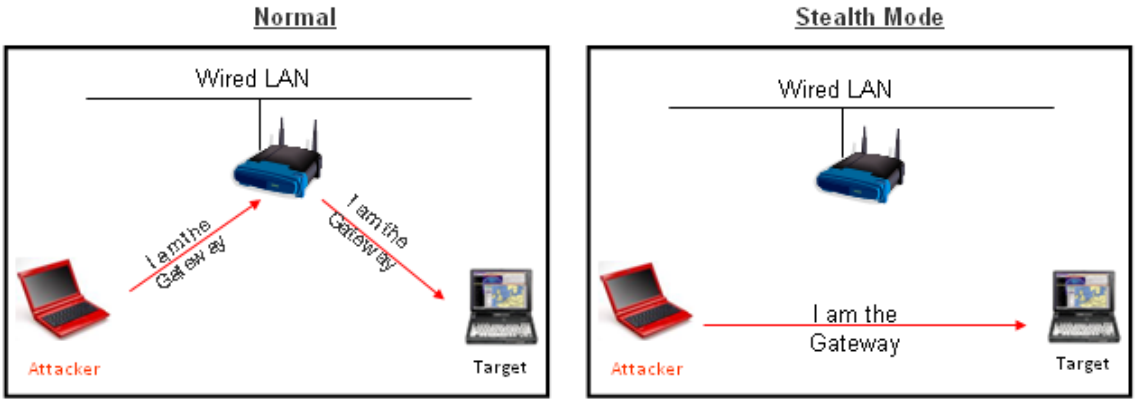


Figure 3: Normal vs Stealth mode ARP poisoning

Attack 2: IP layer targeted attack

To launch targeted attack, IP packet is encapsulated in a group addressed data frames as shown in the figure 4 below. A WiFi client machine whose IP address matches with the destination IP address present in the attack packet accept the packet. All other WiFi clients reject the packet. The technique can be used to trick several TCP and application layer attack in a WPA2 secured WLAN e.g. TCP reset, TCP indirection, DNS manipulation, Port scanning, malware injection, privilege escalation etc.

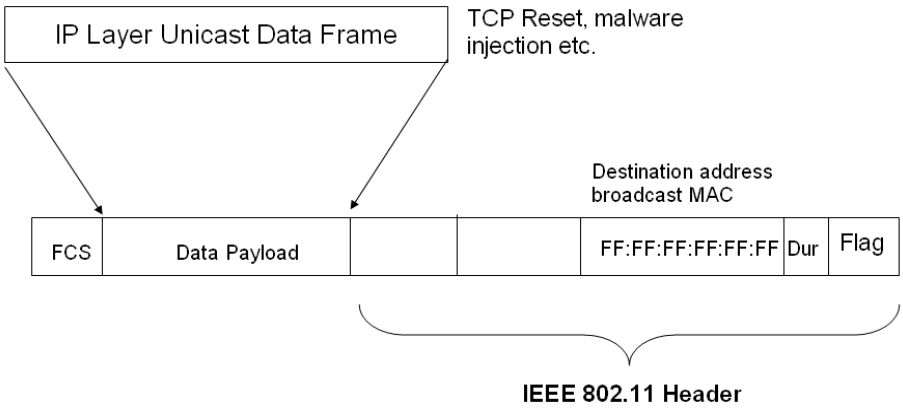
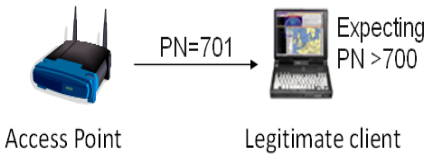


Figure 4: IP packet encapsulated into a group addressed IEEE 802.11 data frame

Replay Attack Detection in WPA2

Replay attack is detected with the help of 48 bit packet number present in all CCMP encrypted data frames. Steps used to detect a replayed frame are given below:

1. *All clients learn the PN associated with a GTK at the time of association*
2. *AP sends a group addressed data frame to all clients with a new PN*
3. *If **new PN > locally cached PN** than packet is decrypted and after successful decryption, old PN is updated with new PN*



Attack 3: Wireless DoS attack

GTK vulnerability can also be exploited to launch DoS attack in a WLAN.

To cause DoS, attacker injects a forged group addressed data frame with a large packet number (PN). All clients present in that network receives forged group addressed data frame and updates locally cached PN with an attacker injected large PN.

Later on when AP sends group addressed data frames, they are dropped by connected clients as the PNs present in AP's injected data frames are less than locally cached PN in clients. The PN manipulation scenario is shown in the figure 5 below.

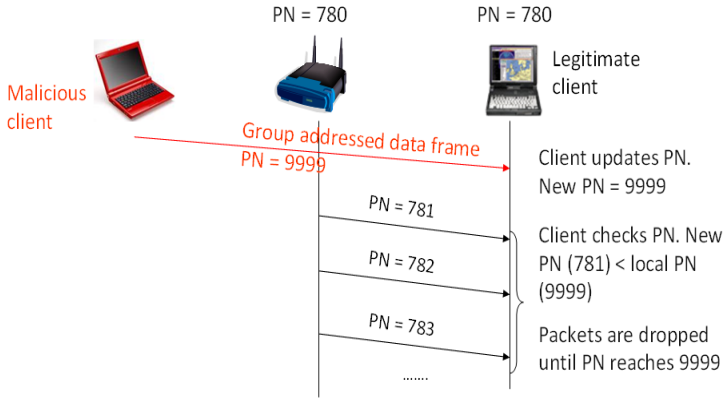


Figure 5: Packet Number (PN) manipulation

As a consequence of PN manipulation, broadcast ARP requests never reaches to wireless client. It fails IP to MAC resolution at the sender. As a results of this IP level communication between sender and receiver never starts.

IV. Attack mitigation

a. Client IDS

Client side IDS such as DecaffeintID [11] or Snort [12] can be used to detect ARP cache poisoning or any inbound connection or malware injection.

Limitations:
Such software is available for either Windows or Linux running laptops while WPA2 networks are accessed by varieties of client devices such as smartphone, notepad, etc.

(a) DecaffeintID detects ARP cache poisoning **(b) PSPF / Client Isolation**

Figure 6: Prevention techniques

b. PSPF or Client isolation [5][6]

The feature restricts peer to peer communication by blocking traffic between two WiFi clients.

Limitations:

Not all controllers or standalone mode APs have PSPF or Client isolation capability.

The feature has known limitation. It does not work across access points for standalone mode APs or across controllers for light weight access points.

c. Software based solution: Deprecate GTK

WPA2 vulnerability can be fixed by deprecating the use of GTK. For *backward Compatibility,* AP should send randomly generated different GTKs to different clients so that all associated clients have different copies of GTK all the time.

Limitations

a. Brings down network throughput
b. Requires AP software upgrade

VI. Conclusion

WPA2 is vulnerable to insider attack. WPA which was introduced as a replacement for WEP is also vulnerable as the group addressed data is handled the same way as it is handled in WPA2. This limitation though known to the designers of WPA or WPA2, is not well understood or appreciated by WiFi users. Our findings presented in this paper show that exploits are possible using off the shelf tools with minor modifications. Legitimate WiFi users who connect to WPA or WPA2 enabled WLAN are vulnerable regardless of the type of authentication or encryption used in the wireless network. In order to provide defense against the insider attack a few solutions have been proposed. Unfortunately, no alternative to GTK vulnerability exists and hence a permanent fix is required at the protocol level. Fixing protocol level problem takes time and hence as an alternative wireless monitoring device e.g. WIPS sensors which are used to detect anomaly in the wireless traffic can be used to detect insider attack.

References

[1] Task Group I, IEEE P802.11i Draft 10.0. Project IEEE 802.11i, 2004.

[2] Aircrack-ng
www.aircrack-ng.org

[3] PEAP: Pwned Extensible Authentication Protocol
http://www.willhackforsushi.com/presentations/PEAP_Shmoocon2008_Wright_Antoniewicz.pdf

[4]. WPA/WPA2 TKIP Exploit: Tip of the Iceberg?
www.cwnp.com/pdf/TKIPExploit08.pdf

[5]. Cisco's PSPF or P2P
http://www.cisco.com/en/US/products/hw/wireless/ps430/products_qanda_item09186a00806a4da3.shtml

[6] Client isolation
http://www.cisecurity.org/tools2/wireless/CIS_Wireless_Addendum_Linksys.pdf

[7]. The Madwifi Project
http://madwifi-project.org/

[8]. Host AP Driver
http://hostap.epitest.fi/

[9]. ARP Cache Poisoning
http://www.grc.com/nat/arp.htm

[10] Detecting Wireless LAN MAC Address Spoofing
http://forskningsnett.uninett.no/wlan/download/wlan-mac-spoof.pdf

[11]. DecaffeinatID
http://www.irongeek.com/i.php?page=security/decaffeinatid-simple-ids-arpwatch-for-windows&mode=print

[12] SNORT
http://www.snort.org/

[13]. Wireless Hotspot Security

http://www.timeatlas.com/Reviews/Reviews/Wireless_Hotspot_Security