cyberpunk.rs

Wireless Security Protocols: WEP, WPA, WPA2 and WPA3 | CYBERPUNK

19-24 minutes

Introduction: Wireless Security Protocols

A very short overview of Wireless Security Protocols including WEP, WPA, WPA2 and WPA3. For each of them we'll try to point out both their strengths and weaknesses and describe some of the possible attacks.

Jump on specific Wireless Security Protocol:

We'll include cryptography details of each protocol at some other post/time, including execution of individual attacks (step by step). For now, just the basics.

WEP [Wired Equivalent Privacy]

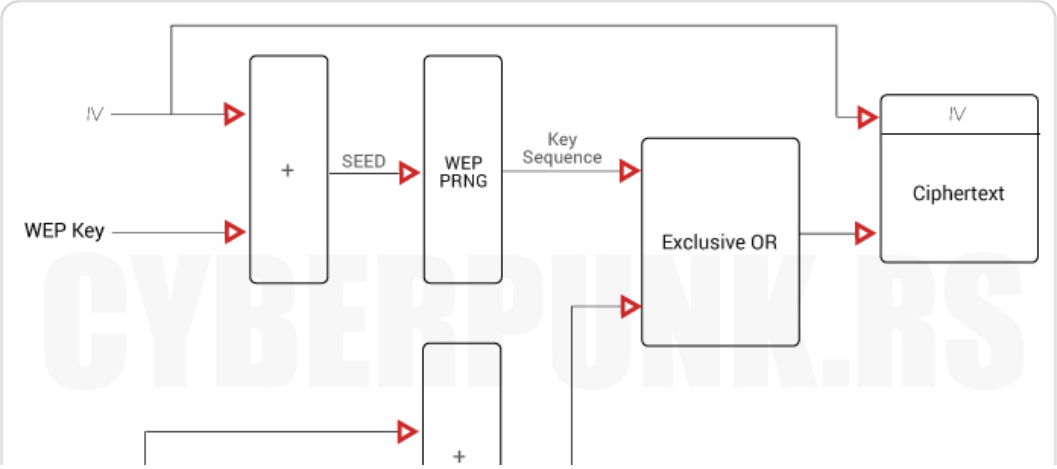
Wired Equivalent Privacy (WEP), introduced as part of the

original 802.11 standard ratified in 1997, it's probably the most used WiFi Security protocol out there. It's pretty recognizable by its key of 10 or 26 hexadecimal digits (40 or 104 bits). In 2004, both `WEP-40` and `WEP-104` were declared deprecated. There were `128-bit` (most common) and `256-bit` WEP variants, but with ever increasing computing power enable attackers to exploit numerous security flaws. All in all, this protocol is "dead".

It uses the RC4 cipher to ensure privacy and a CRC-32 Checksum to ensure integrity of the data transmitted. First, a secret key k is shared between the users of the network (not specified how by the protocol)

To send a message M, one has to compute the integrity checksum c(M) of the message and concatenate it: one has now M.c(M). Then, one encrypts <M.c(M)> by XORing it with a RC4 stream generated by k and a public initialisation vector (IV) of 24 bits, named v. We note it RC4 (v, k).

The result C =<M.c(M)> ⊕ RC4(v, k) is sent to the network, and the user who knows k can get the message by XORing C with RC4(v, k). *"Attacks against the WiFi protocols WEP and WPA"*, M. Caneill & J.L. Gills, October – December 2010



The RC4 stream cipher used by WEP is based upon two algorithms:

- The first one being RC4-Key Scheduled Algorithm (KSA), which transforms a key of length 1 to 256 bits into a initial permutation S of the numbers 0 to N. The internal state of RC4 consists of two numbers i and j used as pointers to elements of S.
- The second algorithm is RC4-Pseudo Random Generation Algorithm (PRGA). It generates a single byte of keystream from the current internal state of RC4 and then updates the internal state. Originally, N=255, but the algorithm can work with different values of N.

  With CRC32 original message is XORed with a constant of 32 bits followed by as many 0 as necessary to reach the length of the message. The result becomes the new “message” and the operation is repeated until the length of the result is under the length of the constant. It is important to note that this hash function is linear and unkeyed.

WEP Attacks:

- Packet Injections
- Fake Authentication
- FMS Attack, statistical, 2001
- KoreK Attack, statistical, 2004

- ChopChop Attack, fake ARP, 2004
- Fragmentation Attack, fragmentation, 2005
- PTW Attack (Pychkine, Tews, Weinmann) , statistical, 2007

Packet Injection

This allows an outsider to generate a large amount of traffic on a network without being associated to it in any way. First, he must capture a packet of specific type. Although hidden behind encryption, packet type can be easily guessed based on packet size.

An ARP request packet is always 28 bytes. By re-injecting it into the network, AP will respond to this forged request sending out packets to legitimate clients. Additional traffic is used to gather encrypted packets faster, and with more packets more the chance he'll break the WEP faster.

Fake Authentication

Fake Authentication attack allows an attacker to join a WEP protected network even if he doesn't know the root key. There are two ways a client can authenticate itself in an WEP protected network:

1. The first method is Open System authentication, basically unprotected.
2. The second method is called Shared Key authentication. This one uses the secret root key and a challenge-response authentication. Client asks AP to connect, AP sends a frame containing a challenge (random byte string, cleartext)

and the client answers with a WEP encrypted frame. If ok, AP answers back with success.

Attacker who sniffs out that handshake can join the network itself. Besides AP challenge, all bytes in 3rd frame are constant. Challenge is transmitted in cleartext in 2nd frame, so attacker can recover key stream (and IV) which is used to encrypt 3rd frame. With that he can now initiate authentication handshake and construct a valid frame (num. 3).

FMS Attack

Released in 2001 by Fluhrrer, Mantin and Shamir, it's based on `RC4` weakness combined with the awarness of IV (Initialization Vector or a nonce, 3 bytes of the per packet key).

Attacker can perform a manupulation on RC4, enabling him to guess a byte of the key (5% probability). If key is wrong, attacker retries with a new key. To reach 50% success rate, attacker will need to capture a lot of packets (up to 6 million).

If we know first "l" bytes of the per packet key, we can simulate "l" first steps of `RC4-KSA`. Don't want to go too deep in the math here, basically next byte of key depends (is somewhat related) on the current one and that can be used to check if we're on the right track. With each iteration we're getting one more byte of key, eventually testing it. If it's wrong, byte of the key is being switched with another probable value and process is restarted.

KoreK Attack

This one is based on FMS attack (first appeared on netstumbler forum, 2004), but lets attacker finds the key faster.

ChopChop Attack

Also found by “KoreK”, and opposed to exploiting a weakness in RC4, it attacks WEP protocol itself (CRC32 checksum and the lack of replay protection). It gives an attacker the ability to decryt a packet without knowing the key.

Flipping a bit in the cipher text and then calculating which bit in encrypted CRC32 value must be flipped so that the packet is still valid. Frequently mentioned is approach is to take away last byte and try to guess its value.

By injecting the altered packet back into the network, packet ends up as invalid because of incorrect ICV. The attacker can make it valid by XORing it with the value that depends on the truncated byte (0-255). The attacker can bruteforce that value. When found, AP (Access Point) will return the packet into the network. Knowing this value, the attacker can calculate the byte of plaintext (and the keystream). By repeating this operation, the attacker is able to decrypt a packet, getting both plaintext and keystream without main password.

The ChopChop Theory.

Fragmentation Attack

Great attack to run if there are no clients currently connected to the access point. Similar to ChopChop attack it speeds up cracking process by injection arbitrary packets into AP. It's going to generate enough traffic to capture large number of IVs improving your chance of cracking the key (aircrack-ng). The "aireplay-ng" & "packetforge-ng" are standard toolkit for this attack.

Released by Bittau in 2005. By sniffing the packets, attacker can find/guess first 8 bytes of clear text. By XORing these 8 bytes with 8 corresponding bytes of cipher text, he can obtain 8 bytes of keystream for a specific IV. Now, he can't use that to send whole packet, but WEP allows him to send a single packet in up to 16 fragments. So, the attacker now uses those 8 bytes of keystream to broadcast a packet containing 64 bytes of known text in 16 fragments. AP on the receiving end takes those fragments, deciphers & combines them into a single packet, encrypting it and send it back to the network.

This packet is now 68 bytes long (64 bytes of known text, and 4 bytes ICV). Using XOR, the attacker gets 68 bytes of keystream for a give IV. Repeating this over and over again, attacker can get up to 1500 bytes of keystream for a IV. When he gets that, it's easy to get keystream of other IVs, simply by sending a broadcast packet of 1500 bytes to AP. The AP will relay this encrypted with a new IV.

As $C \oplus M = K$ the attacker can get the keystream for other IVs and build a dictionnary, allowing him to decipher packets on the network and create traffic.

PTW Attack

The Pyshkin Tews Winmann (PTW) attack, released in 2007.

What makes PTW powerful than all the other attacks is the fact it can make use of every packet captured. It implements a key ranking strategy which instead of trying all possible combinations of the key, picks a set number of likely keys and continues the RC4 algorithm based on those. Using different voting strategies the attacker can pick the most likely key byte at each decision in the tree to determine the correct key.

The tests showed that only 35,000 to 40,000 packets were required to get a 50% succes probability. Other sources state that we can get a probability of 95% with 85,000 frames.

The PTW attack is the default method used by Aircrack-ng to crack WEP keys.

WPA [Wi-Fi Protected Access]

Wi-Fi Protected Access (WPA), became available in 2003, and it was the Wi-Fi Alliance's direct response and replacement to the increasingly apparent vulnerabilities of the WEP encryption standard. The most common WPA configuration is `WPA-PSK` (Pre-Shared Key). The keys used by WPA are `256-bit`, a significant increase over the `64-bit` and `128-bit` keys used in the WEP system.

Note: WPA-PSK basicaly means that Wi-Fi network has a password that is shared by every single Wi-Fi network client.

WPA included message integrity checks (to determine if an attacker had captured/altered packets passed between the access point and client) and the Temporal Key Integrity Protocol (TKIP). TKIP employs a per-packet key system that was radically more secure than the fixed key system used by WEP. The TKIP encryption standard was later superseded by Advanced Encryption Standard (AES).

TKIP uses the same underlying mechanism as WEP, and consequently is vulnerable to a number of similar attacks (e.g. Chop-Chop, MIC Key Recovery attack).

Usually people don't attack WPA protocol directly, but supplementary system that was rolled out with WPA – Wi-Fi Protected Setup (WPS).

Note: TKIP (temporal Key Integrity Protocol) – The `RC4` stream cipher is used with a `128-bit` per-packet key, meaning that it dynamically generates a new key for each packet. Although still used, it's considered obsolete after being replaced by CCMP in 2009.

WPA Attacks:

- Back and Tews' Improved Attack on RC4, 2008, Inject
- Ohigashi-Morii Attack (Beck and Tews' + Man in the middle), 2009, inject
- Michael Attacks, 2010, inject

- THe Hole196 Vulnerability, 2010, inject/dos/MITM
- Dictionary Attack against the handshake,key recovery

Back and Tews' Attack

Released in 2008, exploits weakness in TKIP, allowing an attacker to decrypt ARP packets and to inject traffic into a network, enabling a DoS or ARP poisoning.

Attack "requires" Quality of Service (QoS) to be enabled (practical aspect). That allows several channels to be used. Each channel has its own TSC (TKIP Sequence Counter). Channel 0 holds most of the traffic, other channels will have lower TSC. Attack requires Key Renewal Interval to be longer than 15 min (time needed to decrypt an ARP packet).

Attacker de-authenticates a station, then captures ARP packet. Next, he'll perform a modified ChopChop attack to recover ICV (Integrity Check Value) and MIC of the packet. With that, attacker needs to guess the last part of the packet, IP address. Finaly, he reverses MICHAEL algorithm and get MIC key. With that he can now inject custom packet into the network.

Countermeasure**:** Disable QoS.

Ohigashi-Morii Attack

From 2009, an improvement of the Beck-Tews attack on WPA-TKIP, more efficient for all modes of WPA and not just those with QoS features.

Michael Attack

In 2010, Beck found that if the internal state of Michael reaches a certain point, the Michael algorithm resets. With that, an attacker can inject some text in a packet, add a string that resets Michael algorithm. Packet is changed but the Michael's result remains correct. Apparently, requirements of this attack are even tighter compared to "Beck and Tews".

Countermeasure**:** Disable QoS

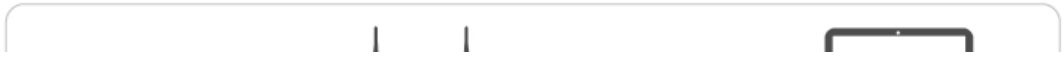
The Hole196

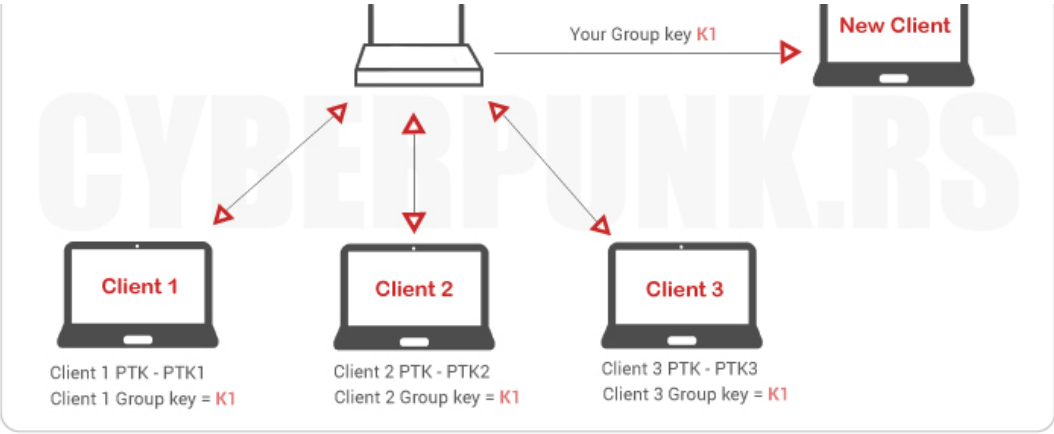
In 2010, Sohail Ahmad found a hole in 802.11. This is a MITM attack, not a key-recovering attack. The attacker has to be an authorized user of the network.

Two types of keys for data encryption:

1. Pairwise Transient Key (PTK) – used to protect unicast data frames
2. Group Temporal Key (GTK) – used to protect group addressed data frames (e.g. broadcast ARP frames)

   Attacker sends an ARP request (with his MAC and IP address of the AP), so the other clients update their ARP tables. With that, all clients will send their packets to the attacker. Attacker will receive packets decrypted by the AP, re-encrypting them with his own key. Everyone can build and broadcast fake packets with GTK. Messages sent with group keys, don't have protection against spoofing.





The point of the attack is to send a message with a GTK key but directed to a target MAC instead of a broadcasting MAC address (Detectable). By doing this in a kinda “stealthy” way, only the victim will process that broadcast packet (unless the ARP table has static resolution for the MAC of the gateway), ending up with IP poisoning, replacing the router.

Possibilities:

- ARP Poisoning/MITM
- Buffer overflow
- Malware Injection
- WDoS

Countermeasure**:** For enterprises client Isolation (PSPF) + endpoint security (ARP poisoning detector like Snort or similar).

WPA2 [Wi-Fi Protected Access II]

Of course, WPA2 replaced WPA. Certification began in September, 2004 and from March 13, 2006 it was

mandatory for all new devices to bear the Wi-Fi trademark. Most important upgrade is mandatory use of AES algorithms (instead of previous RC4) and the introduction of CCMP (AES CCMP, Counter Cipher Mode with Block Chaining Message Authentication Code Protocol, 128 Bit) as a replacement for TKIP (which is still present in WPA2, as a fallback system and WPA interoperability).

As in previous version, attack on WPS is the most frequent one.

Note: WPA/WPA2 MGT (Management) means that the password is not a pre-shaed key, instead authentication service is used, usually a RADIUS service which verifies username/password of the Wi-Fi network client. MGT is most often tied to corporate/professional environments.

WPA2 Attacks:

- KRACK Attack
- PMKID Attack (PSK)
- WPS Attack
- Brute-force/Dictionary attack
- Hole 196

KRACK Attack

Discovered by Mathy Vanhoef and Frank Piessens in 2016. It’s a severe replay attack.

The attack targets the four-way handshake used to

establish a nonce (a kind of “shared secret”) in the WPA2 protocol. The standard for WPA2 anticipates occasional WiFi disconnections, and allows reconnection using the same value for the third handshake (for quick reconnection and continuity). Because the standard does not require a different key to be used in this type of re-connection, which could be needed at any time, a replay attack is possible.

Countermeasure: access points have configuration options that can disable EAPOL-Key frame re-transmission during key installation.

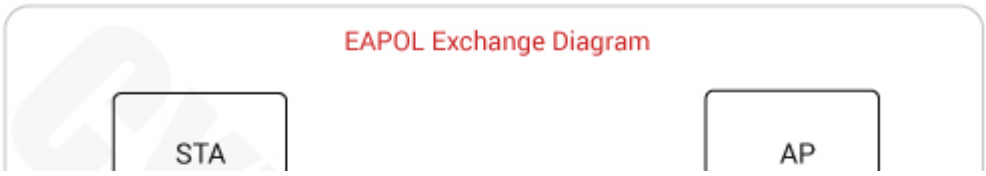
Useful: Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2

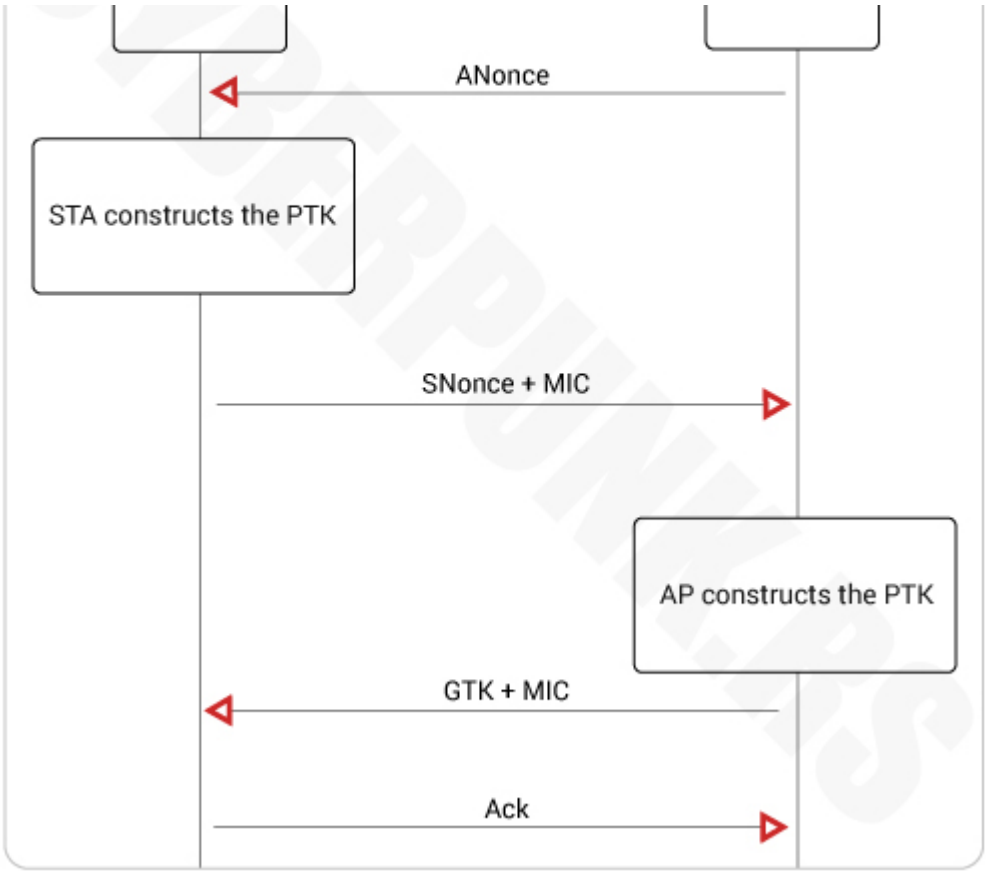
PMKID attack (PSK)

New exploit was anounced on August 4th, 2018, for Wi-Fi networks that use WPA/WPA2-PSK (pre-shared key). The vulnerability allows attackers to obtain the PSK being used for the particular SSID.

The attack was discovered accidentally while looking for new ways to attack the new WPA3 security standard.

The main difference compared to other attacks is that in this attack, capture of a full EAPOL 4-way handshake is not required. The new attack is performed on the RSN IE (Robust Security Network Information Element) of a single EAPOL frame.





The main advantages of this attack are as follow:

- No more regular users required – because the attacker directly communicates with the AP (aka “client-less” attack).
- You wont need to wait for a complete 4-way handshake between the regular user and the AP.
- No more eventual retransmissions of EAPOL frames (which can lead to uncrackable results) and invalid passwords sent by the regular user.
- There wont be lost EAPOL frames when the regular user or the AP is too far away from the attacker.
- No more fixing of nonce and replay counter values required (resulting in slightly higher speeds).
- No more special output format (`pcap`, `hccapx`, etc.) – final

data will appear as regular hex encoded string.

Countermeasure: It is recommended to disable 802.11r on WPA/WPA2-PSK networks.

WPS Attack

WPS was introduced in 2006, and the goal of the protocol is to allow home users who know little of wireless security to set up Wi-Fi Protected Access, as well as making it easy to add new devices to an existing network without entering long passphrases.

In December 2011 a flaw was revealed that affects wireless routers with the WPS feature. That flaw allows a remote attacker to recover the WPS PIN in a few hours with a brute-force attack and, with the WPS PIN, the network's `WPA/WPA2` pre-shared key.

WPS enables client to send 8 digit pins to the access point, which verifies it and then allows the client to connect. Pin contains only numbers, with WPS there's a delay because attacker needs to wait for AP response. So, attacker can try a few keys per second (or one key per few seconds).

We have here 8 digits with 10 numbers, 10^8 (100.000.000). That's too much. The 8th digit is checksum of first 7 digits, so we have 10^7. Furthermore, the pin number for verification goes in two halves, so we can independently verify the first 4 and the last 4 digits. It's far easier to guess 4 digits 2x than 8 digits 1 at once. Finaly, math ends up with: $10^4 + 10^3 = 11{,}000$ guesses.

While this tactic used to take a number of hours, the newer WPS Pixie-Dust attack can crack networks in seconds. Since 2011, many routers now have protections to detect and slow down (rate-limiting) or shut down a Reaver-type attack (lock with too many failed PIN attempts).

Routers updated some settings to prevent WPS, but flaws still existed in the way they implement encryption. It's relatively difficult to create truly random numbers, which is required to produce strong encryption. To achieve this, there's usually a function that takes "seed" and produces a pseudo-random number.

If there's a use of long or varying "seed" number, you can get same result as a number that's actually random, but if you use an easily guessed "seed", or even worse, the same one again and again, you end up with weak encryption that's easy to break. This is what happened with those updated routers, the thing that WPS Pixie-Dust attack exploits.

Countermeasure: Turn off the WPS feature.

Brute-Force/Dictionary attack

This relies on capturing a WPA handshake, and then using a wordlist or brute-force to try and crack the password. Depending on the password strength (length, charset), it can be difficult or impossible to break it in a "reasonable" amount of time.

Countermeasure**:** Use long passwords (12+) and different charsets (alphanum, special chars, upper/lower case).

WPA3 [Wi-Fi Protected Access III]

In January 2018, the Wi-Fi Alliance announced WPA3 as a replacement to WPA2. The new standard uses `128-bit` encryption in WPA3- Personal mode (`WPA-PSK`, pre-shared key) or `192-bit` in WPA3 – Enterprise (RADIUS authentication server).

WPA3 will be much harder to attack because of its modern key establishment protocol called “Simultaneous Authentication of Equals” (SAE) or the Dragonfly Key Exchange. SAE improves security of the initial key exchange and offers better protection against offline dictionary-based attacks.

It is just as susceptible to man-in-the-middle attacks and offers no protection against evil twin attacks.

Conclusion

There are many useful tools out there to play around with Wi-Fi, e.g Aircrack-ng and Wireshark.

When it comes to protocols, best to worst:

- WPA3
- WPA2 MGT or WPA MGT
- WPA2 + CCMP/AES
- WPA + CCMP/AES
- WPA + TKIP | WPA + TKIP/AES (TKIP present as a fallback method) WEP

- Open Network (no security at all)

  Hope this helps a bit on how to configure your router/wifi and set your defense. Use the highest version possible, long passwords and disable WPS. We'll cover cryptography segments and individual attacks (steps on how to execute them) later on.