Wireless Hacking

Keys upon Keys. So many keys….

Just interesting

To discover passphrase stored on machine

- Netsh wlan show profiles
- Netsh wlan show profile bellaliant706 key=clear
- Security key
- Key content

Your passphrase is stored here

So Many ID’s

- SSID – Service Set ID, Alphanumeric name of your wireless network.
- BSSID – Basic Service Set ID, Hexademicimal number of the WAP mac address
- IBSS – Independent Basic Service Set ID, randomly generated 48 bit number used in place of a BSSID (MAC) on adhoc networks without an AP.

To display: netsh wlan show interfaces

4-WAY Handshake

In order to exchange encrypted traffic between an Access Point and a Client, we need two Keys:

- The Pairwise Transit Key (PTK) for unicast traffic.
  - Unique between Client and AP
- The Group Temporal Key (GTK) for multicast traffic
  - Shared by all Clients of the AP

First we need to come up with a Pre-Shared Key (PSK) that both sides know but is never transmitted.

4-Way Handshake

Recipe for Calculating the Pre-shared Key.

Ingredients:

- SSID
- Passphrase
- a few other things related to these two like the **length of the passphrase** and **level of desired encryption** (i.e. 4096)
- Hashing algorithm (i.e. Sha-1)

Both the Wireless Access Point (**The Authenticator**) and the Client (**The Supplicant**) have all this information.

The output of the hash of these values is the Pre-Shared Key. This will be used to calculate other encryption keys based on information that is shared. It also serves as a substitute for the Master Session Key in Non-Enterprise WPA Ap's

4-Way Handshake

To encrypt the traffic between the AP and the Client the ***Pairwise Transit Key (PTK)*** *must be calculated*.

To calculate the **PTK** we need something called the **Pairwise Master Key (PMK)**

In WPA2/PSK the **Pre-Shared Key (PSK)** becomes the **Pairwise Master Key (PMK).** Otherwise, it is generated from the Master Session Key (more later). Never transmitted.

- Both sides know the **MAC of the AP** and the **MAC of the Client.**
- The AP broadcasts an **Anounce** (A one time number generated by the Authenticator)
- The Client broadcasts an **Snounce** (A one time number generated by the Supplicant)

The **Pairwise Transit Key** is:

PTK = PRF(PMK + Anounce + Snounce + MAC of AP + MAC of Client)

or

PTK = PRF(PSK + Anounce + Snounce + MAC of AP + MAC of Client)

So what is **PRF**?

Answer: **A function**

The PTK is the encryption key for unicast traffic between AP <-> Client and is unique for each Client

4-Way Handshake

*To encrypt the traffic between the AP and all Clients the **Group Temporal Key (GTK)** must be calculated. (i.e. this is a Broadcast Key)*

To calculate the **GTK** we need something called the **Group Master Key (GMK).**

The **GMK** is derived from the **Master Session Key (MSK).**

The **Master Session Key** is created from:

- 802.1X/EAP; or
- The Pre-Shared Key (PSK)

So the sequence is: MSK -> GMK -> GTK ***OR*** PSK->GMK->GTK

The GTK is shared on all attached devices.

4-way handshake

PMK (Pairwise Master key) – generated by
Master Session Key. Resides on all attached devices.

PTK (Pairwise Transit Key)

–used to encrypt unicast traffic.
AP generates announce.
Client generates snounce .
Mac addr of c Client.
Mac addr of Access Point.
PTF pseudo randon function.

ptk = PTF(PMK+announce+ snounce+MacA+MacS)

Client

Authenticator

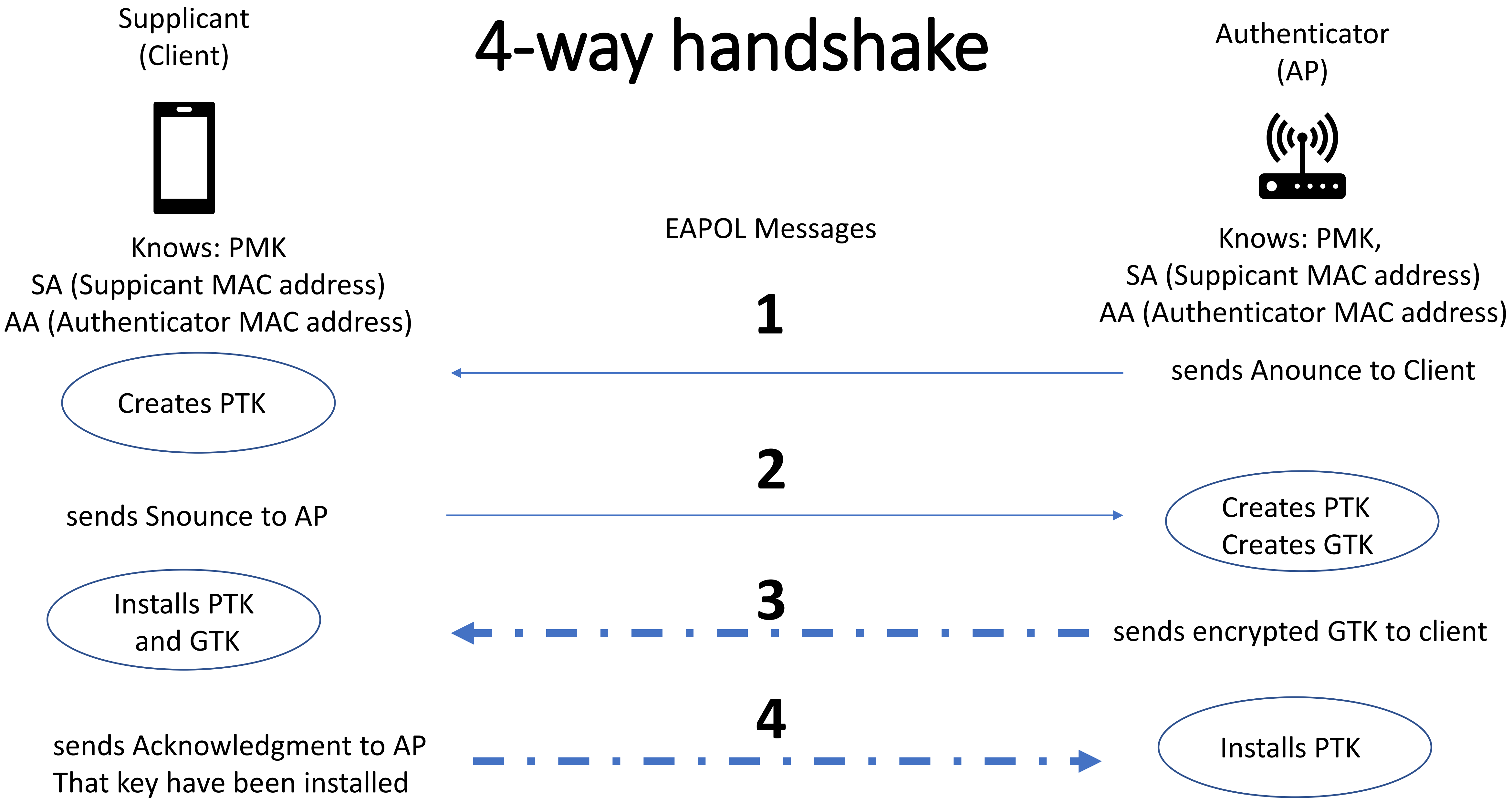
GMK (Group Master Key) – Used in handshake
to create GTK.

GTK (Group temporal Key) –used to encrypt broadcast and multi-cast.

MSK (Master Session Key) The first Key created

https://www.wifi-professionals.com/2019/01/4-way-handshake

4-way handshake



Supplicant
(Client)

Knows: PMK
SA (Suppicant MAC address)
AA (Authenticator MAC address)

Authenticator
(AP)

Knows: PMK,
SA (Suppicant MAC address)
AA (Authenticator MAC address)

EAPOL Messages

1

sends Anounce to Client

Creates PTK

2

sends Snounce to AP

Creates PTK
Creates GTK

3

sends encrypted GTK to client

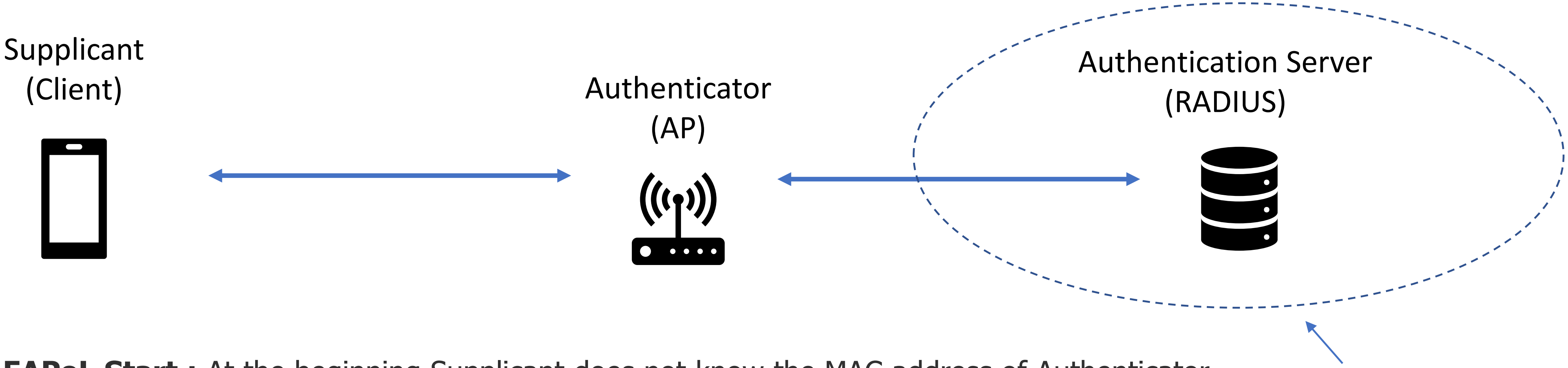
Installs PTK
and GTK

4

sends Acknowledgment to AP
That key have been installed

Installs PTK

Extensible Authentication Protocol Over LAN (EAPOL)
(Ether Type 0x888E)



Not present in WPA/PSK (Home networks) only those that use EAP

EAPoL Start : At the beginning Supplicant does not know the MAC address of Authenticator. So, it sends this message to a multicast group to learn that if are there any Authenticators in the LAN.

EAPoL Key : This message is used by Authenticator to send encrypted keys.

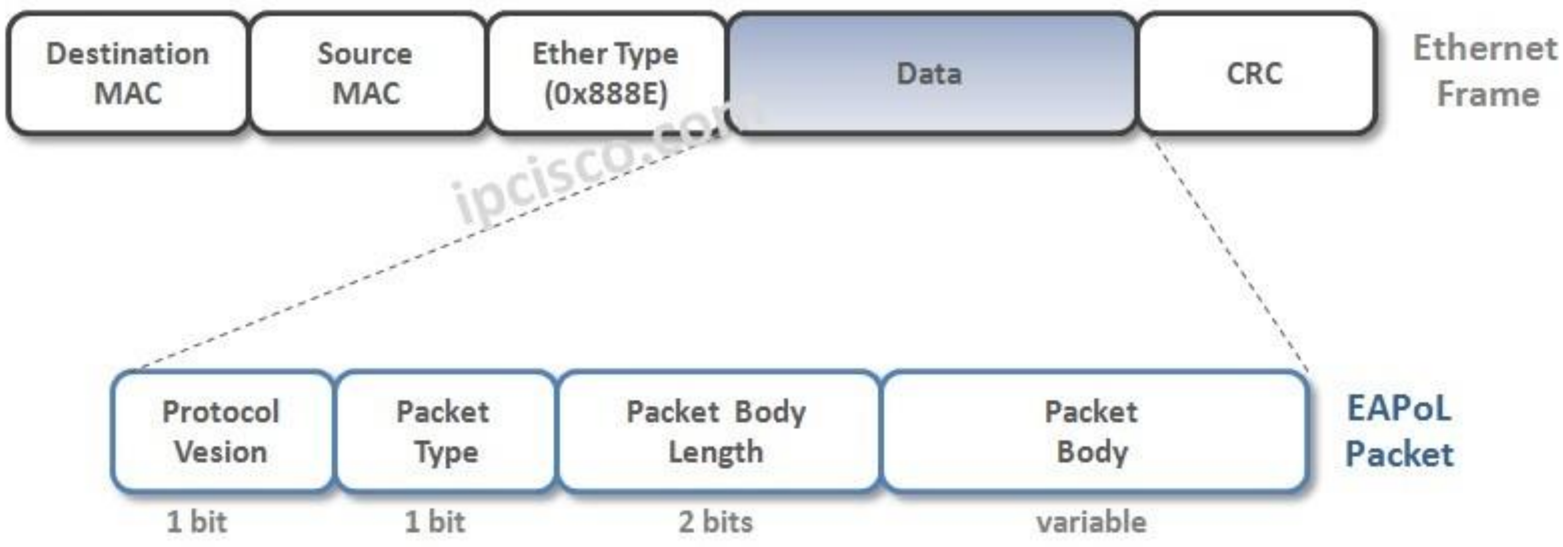
EAPoL Packet : The message that is sent for Normal EAP frames.

EAPoL Logoff : The message that shows that the Supplicant wants to terminate the connection.

EAPoL Encapsulated ASF Alert : It is sent for alerts about unauthorized ports.

Extensible Authentication Protocol Over LAN (EAPOL)
(Ether Type 0x888E)

EAPOL is an Encapsulation Protocol



IPCisco.com (N.D.) *EAPoL (Extensible Authentication Protocol over LAN)*
https://ipcisco.com/lesson/eapol-extensible-authentication-protocol-over-lan/

Decrypting

You will need: The Passphrase & a Capture of the 4-way handshake

We can get most of the elements that we need to calculate the PTK from capturing the 4-way handshake. BUT we don’t get the PTK. Instead we get a hash of the PSK.

The only piece we are missing to get the PTK is the value of the PSK (or passphrase). If this is weak…WE CAN GUESS IT! We get as many guesses as we want.

Recall:

PTK = PRF(PSK + Anounce + Snounce + MAC of AP + MAC of Client)

Decrypting

First we have to get the Client and the AP to drop the connection and reconnect several times.

We need to obtain:

- A wireless adapter with the **RTL8812AU** chipset, that is capable of both Monitor Mode and Packet Injecton.
- A method to Deauth both the client and the AP to get several samples of the handshake.
- A password list. (rockyou, goodguess)
- A method of trying different hash combinations really fast.

Enter the **airgedden** tool.

airgeddon

airgedden, v1s1t0r1sh3r3 (January 19, 2021) *github*, https://github.com/v1s1t0r1sh3r3/airgeddon