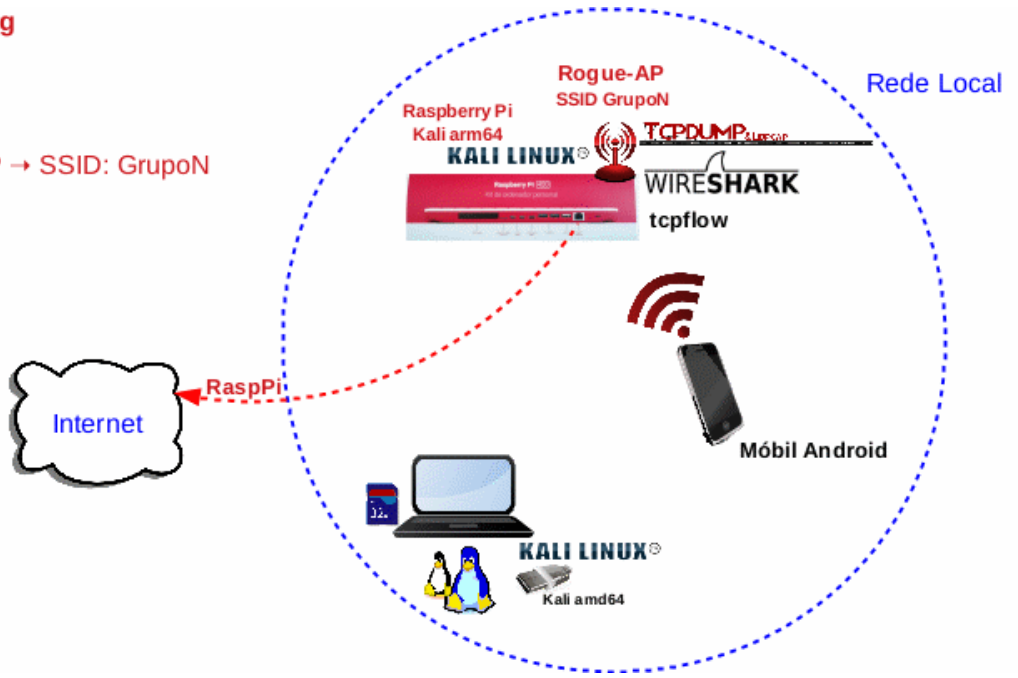
TALLER SI – PRÁCTICA 11		
NÚMERO DE GRUPO	**FUNCIÓNS**	**Apelidos, Nome**
	Coordinador/a:	
	Responsable Limpeza:	
	Responsable Documentación:	

ESCENARIO: Rogue AP → Phishing

Raspberry Pi Grupo:
Rede Local + Internet
EvilTrust-kali-rpi-Automatic-Boot → AP → SSID: GrupoN
tcpdump
Wireshark
tcpflow

Móbil alumnado Android



Material necesario	Práctica: Phishing. “Roubo” de credenciais Sniffers
■ [1] Práctica 9 ■ Raspberry Pi 4 (ou 400) con acceso á rede local e Internet (material que posúe o grupo) ■ [2] Repositorio evilTrust-kali-rpi-Automatic-Boot ■ [3] README.md ■ Móbiles alumnado Android ■ [4] tcpdump ■ [5] wireshark ■ [6] 03-Exercicio3-Wireshark-ICMP-ARP.pdf ■ [7] tcpflow	(1) Prerrequisito: Ter realizada a Práctica 9 [1] (2) Raspberry PI a) Rogue AP lanzado a espera de “*víctimas*” (3) Sniffers: tcpdump, wireshark a) Interceptar comunicacións. b) Capturar comunicacións (4) Tratamento de datos: tcpdump, wireshark, tcpflow

Procedemento:

(1) Realizar a Práctica 9 [1], tal que agora teremos lanzado un **Rogue AP GrupoN** na canle **5.** A este terminal ímolo denominar **terminalA**.

> ***NOTA: N=número de grupo, SSID=GrupoN, channel=N***. Por exemplo, se:
> Número de grupo=5 → SSID=Grupo5, channel=5

(2) Raspberry Pi: **tcpdump** [4]

(a) Antes da conexión de calquera víctima, abrir un terminal *(a partir deste momento denominado **terminalB**)* e sniffar o tráfico POST para ver que como as comunicacións teñen lugar mediante o protocolo HTTP (sen cifrado) e polo tanto tamén se poden visualizar as credenciais unha vez introducidas. Para iso, executar:

```
# tcpdump -i wlan0 -lAn | egrep -i "POST /|user|pass" | tee -a fileRevision.txt #Ver as credenciais introducidas e engadilas ao ficheiro fileRevision.txt
```

(b) Abre un novo terminal(ímolo denominar **terminalC**)

(3) Móbiles alumnado Android: Conectar ao **Rogue AP GrupoN,** onde N=número de grupo.

(a) Introduce unhas credenciais ficticias e valídate na páxina. Que acontece?

i. Raspberry Pi. Captura unha imaxe do **terminalA**.

ii. Raspberry Pi. Captura unha imaxe do **terminalB**.

iii. Raspberry Pi. Captura unha imaxe coa saída do seguinte comando no **terminalC**:

```
# grep -i 'password_aulavirtual=' fileRevision.txt
```

(b) Abre unha nova páxina e accede a www.gmail.com Introduce unhas credenciais ficticias e valídate na páxina. Que acontece?

i. Raspberry Pi. Captura unha imaxe do **terminalA**.

ii. Raspberry Pi. Captura unha imaxe do **terminalB**.

iii. Raspberry Pi. Captura unha imaxe coa saída do seguinte comando no **terminalC**:

```
# grep -Ei 'identifier|password|accounts.google.com|gmail.com' fileRevision.txt
# grep -Ei 'ssl' fileRevision.txt
```

(c) Avisar ao docente para revisión.

(4) Raspberry Pi.

(a) Aborta a execución do comando **tcpdump** premendo **Ctrl+C** no **terminal** ou **terminais** onde se está a executar.

(b) Executar no **terminalB**:

```
# DIRIES='/root/evilTrust-kali-rpi-Automatic-Boot/ies-ald-login'
# IP=$(grep ip "${DIRIES}/datos-privados.txt" | head -1 | awk '{print $NF}') #A variable IP garda a IP da primeira "víctima"
# echo ${IP} #Amosar o valor da variable IP, que identifica a IP da primeira "víctima"
```

(c) Aborta[5] a execución do **Rogue AP GrupoN** premendo **Ctrl+C** no **terminalA**

(d) Executar no **terminalB**:

```
# tcpdump -i wlan0 -lAn dst host accounts.google.com  | tee -a fileRevision.txt #Ver os paquetes capturados e engadilos ao ficheiro fileRevision.txt
```

(e) Abrir un novo terminal(**terminalD**) e executar:

```
# tcpdump -i wlan0 -lAn host ${IP} #Executar tcpdump na NIC wlan0 capturando todos os paquetes que coincidan coa IP da primeira "víctima"
```

(f) Abrir un novo terminal(**terminalE**) e executar:

```
# tcpdump -i wlan0 -w fileCaptura.pcap #Executar tcpdump na NIC wlan0 capturando todos os paquetes e gardándoos no ficheiro fileCaptura.pcap para un posterior tratamento.
```

(g) Como no **terminalA** estás situado no cartafol `/root/evilTrust-kali-rpi-Automatic-Boot` lanza de novo o **Rogue AP GrupoN** co seguinte seguinte comando:

```
# bash exec.sh
```

(h) Realizar de novo os apartados (3) e o apartado (4a)

(i) Avisar ao docente para revisión.

(5) Raspberry Pi: **wireshark** [5]

(a) Antes da conexión de calquera víctima, no ***terminalB*** imos sniffar o tráfico POST para ver que como as comunicacións teñen lugar mediante o protocolo HTTP (sen cifrado) e polo tanto tamén se poden visualizar as credenciais unha vez introducidas. Para iso, executar:

```
# wireshark -i wlan0 & #Sniffer wireshark lanzado esperando comunicacións coa NIC wlan0
```

(b) Filtro a executar: `filter → http.request.method == "POST"`

(c) **Menú → Start capturing packets (Premer na aleta do "tiburón")**

(6) Móbiles alumnado Android: Conectar ao **Rogue AP GrupoN,** onde N=número de grupo.

(a) Introduce unhas credenciais ficticias e valídate na páxina. Que acontece?

i. Raspberry Pi. Captura unha imaxe do **terminalA**.

ii. Raspberry Pi. Captura unha imaxe da interface gráfica do **wireshark**.

iii. Raspberry Pi. Unha vez capturado o paquete POST esperado con destino a 172.16.31.1 detén o **Wireshark: Menú → Stop capturing packets (Premer no cadrado de fondo vermello)**

iv. Raspberry Pi. **Selecciona ese paquete → Clic botón dereito → Follow → HTTP Stream → Captura unha imaxe**

> **NOTA:** Podes atopar ese paquete mediante a búsqueda dunha cadea. Así:
> **Menú → Icono lupa -> Display filter → String:** *application/x-www-form-urlencoded*

v. Raspberry Pi. Volta a arrancar a captura de paquetes mediante Wireshark (apartado 5c). Non gardes os paquetes capturados anteriormente (Continue whithout saving)

(b) Abre unha nova páxina e accede a www.gmail.com Introduce unhas credenciais ficticias e valídate na páxina. Que acontece?

i. Raspberry Pi. Captura unha imaxe do **terminalA**.

ii. Raspberry Pi. Captura unha imaxe da interface gráfica do **wireshark**.

iii. Raspberry Pi. Detén o **Wireshark: Menú → Stop capturing packets (Premer no cadrado de fondo vermello)**

iv. RaspberryPi. Wireshark → Filtro a executar: `filter → http.request.method == "POST"` Indica que acontece? Captura unha imaxe da interface gráfica do **wireshark**

v. RaspberryPi. Wireshark → Filtro a executar: `filter → ssl.handshake` Indica que acontece? Captura unha imaxe da interface gráfica do **wireshark**

vi. Que é o protocolo TLS? Para que serve? Que versión se está a empregrar no apartado anterior?

(c) Avisar ao docente para revisión.

(7) Raspberry Pi. Tratamento de datos: **tcpflow** [7]

(a) Executar no **terminalE**

```
# command -v tcpflow ; [ $? -ne 0 ] && apt update && apt -y install tcpflow #Instalar o paquete tcpflow no caso de non estar instalado
# mkdir /tmp/temporal  #Crear directorio /tmp/temporal
# tcpflow -o /tmp/temporal -r fileCaptura.pcap #Atopar dentro do ficheiro fileCaptura.pcap conversacións e almacenalas para o seu estudo dentro do cartafol /tmp/temporal
# cd /tmp/temporal #Acceder ao directorio /tmp/temporal
# file * | grep -i ascii #Atopar os ficheiros ascii
```

(b) Executar no **terminalE**:

```
# grep -i 'password_aulavirtual=' *
```

Que acontece? Indica dirección da comunicación e sockets empregados.

(c) Abrir o ficheiro **fileCaptura.pcap** con **Wireshark** e realizar de novo os apartados (5b) e (6b.v)

Que acontece? Captura imaxes da interface gráfica do **wireshark**

(d) Avisar ao docente para revisión.