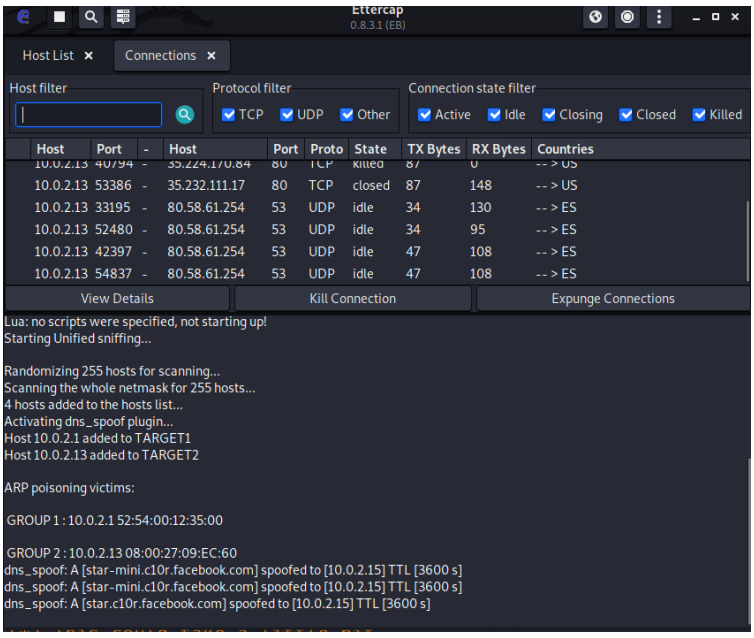
PRACTICA 7

DANIEL KHOMYAKOV TRUBNIKOV

1. Evidenciar el desarrollo de un ataque DNS Spoofing en un entorno controlado, permitiendo redirigir a la víctima a una aplicación web falsa levantanda mediante la herramienta setoolkit (en Kali).

Primero para esta parte necesitaremos dos cosas, la primera es el setoolkit que es lo que usaremos para poner clonar el sitio web deseado que en este caso será la página de Facebook, por ejemplo y luego necesitaremos el Ettercap para redirigir el DNS de Facebook a nuestra página web clonada y con ello haciendo pues el DNS Spoofing. Ahora lo primero que haré será configurar el Ettercap que lo que haré será decir que cuando se llame a la pagina web de Facebook (sin importar de quien es) lo redirigirá a un DNS con la IP 10.0.2.5, la cual es la mía y donde se creará la página copia mía de Facebook.

Ahora pues me pondré a ejecutar el Ettercap y luego cargarle el plugin de DNS Spoofing para poder realizar la redirección de la página (configurando todo el tema de los targets 1 como el 10.0.2.1 (broadcast) y 2 como el objetivo que es la IP de la maquina siendo la 10.0.2.13).



Además de ello, como podemos ver, ya esta haciendo efecto el plugin que le metimos y luego decidí hacer unas pruebas y sí que podía sniffear los paquetes que recibía y pasaba la IP 10.0.2.13 que es de mi otra maquina que he usado como prueba siendo esta un Ubuntu.

```
danekar@danekar:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:09:ec:60 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.13/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 508sec preferred_lft 508sec
    inet6 fe80::7099:8ed7:9ad8:d28d/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
danekar@danekar:~$
```

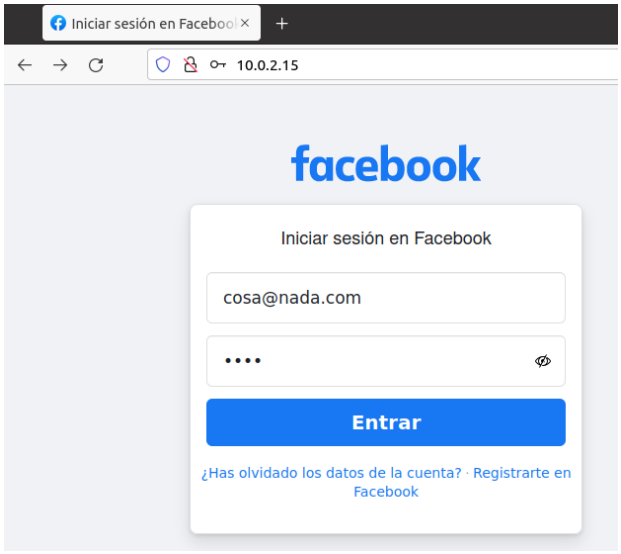
Ahora después de configurar el Ettercap, pues ahora me pondré a configurar el setoolkit

```
[---]        Follow us on Twitter: @TrustedSec         [---]
[---]        Follow me on Twitter: @HackingDave        [---]
[---]       Homepage: https://www.trustedsec.com       [---]
        Welcome to the Social-Engineer Toolkit (SET).
         The one stop shop for all of your SE needs.

   The Social-Engineer Toolkit is a product of TrustedSec.

           Visit: https://www.trustedsec.com

   It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!


 Select from the menu:

   1) Social-Engineering Attacks
   2) Penetration Testing (Fast-Track)
   3) Third Party Modules
   4) Update the Social-Engineer Toolkit
   5) Update SET configuration
   6) Help, Credits, and About

  99) Exit the Social-Engineer Toolkit

set>
```

Primero le decimos que queremos realizar un ataque de Social-Engineering, continuado por website attack vector ya que la idea con los DNS SPoofing casi siempre es sacar las contraseñas u otras credenciales de las personas donde continuare por darle a “Credential Harverster Attack Mettod” donde luego nos pedirá por último que tipo de robo credenciales en la web queremos hacer y aquí es donde ya por fin le especificamos que queremos clonar una página web. Ya después de realizar todo ese proceso, en el setoolkit habrá que decirle que función queremos realizar la cual será clonar una página web y hasta llegar hasta hay que indicarle primero que queremos

```
set:webattack> IP address for the POST back in Harvester/Tabnabbing [10
.0.2.15]:10.0.2.15
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:www.facebook.com
```

Lo que hacemos es introducir nuestra IP junto a la página web que queremos que se copie que en estos casos es la IP 10.0.2.15 y la url es www.facebook.com donde ya con estos datos, veremos cómo va todo una vez entrado en mi IP desde la máquina de Ubuntu.



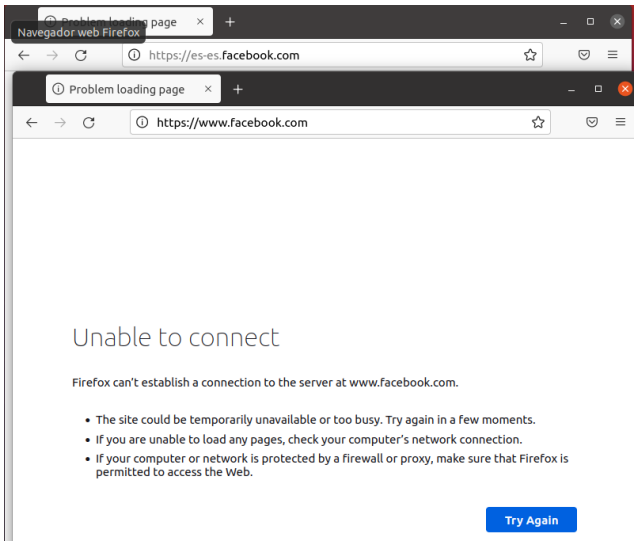
Como podemos observar, una vez introducidos la www.facebook.com, me ha redirigido a “Facebook” cuando realmente es mi página copiada del login.php de facebook donde además la otra información que me dice es que en cuanto me he conectado ya me ha empezado a avisar de posibles datos que podría sacar y cómo podemos ver, el mensaje http que recibimos es de la respuesta que le ha llegado desde mi IP que es el método http, la versión y el código que es 200, que representa que todo ha ido bien.

```
PARAM: lgnrnd=053425_7Myv
PARAM: lgnjs=1640266569
POSSIBLE USERNAME FIELD FOUND: email=cosa@nada.com
POSSIBLE PASSWORD FIELD FOUND: pass=cosa
PARAM: prefill_contact_point=cosa@nada.com
```

Como podemos observar, una vez introducido los datos, me los ha cogido setoolkit y los ha cogido correctamente ya que si miramos la imagen cuando enseño que pasa cuando pongo mi IP, aparecen esos datos.

Pero yo entiendo perfectamente que este no es el tipo de DNS Spoofign que se pedía ya que, en vez de poner la IP, la idea del DNS Spoofing, es el poder redirigir el DNS que al fin y al cabo en groso modo, es el que traduce www.facebook.com en 10.0.2.15, donde tengo mi página de Facebook falsa, pero por alguna razón, el setoolkit no lo pillaba. También miraba que es lo que obtenía el Ettercap para ver si había alguna diferencia, y pues cuando metía el DNS pues me mandaba a la página de Facebook normal y corriente y

recibía con normalidad pues paquetes de ahí, mientras que, si le ponía la IP, pues me mostraba como recibía muchos menos paquetes. Y ya para el final, como tenía que hacer muchas pruebas pues después de un rato Facebook me dejo de aceptar las peticiones:



Aquí muestro dos pestañas porque la de arriba es la que redirige Facebook una vez has intentado meter la contraseña y correo desde la página de copia (poniendo la IP en mi caso), mientras que la de abajo es la que intentas entrar directamente y pues por desgracia no sabía ya como arreglar esto ultimo y donde de mi proceso fallaba.

2. Investigar las diferentes técnicas que permiten rompe r una red Wi-Fi WEP y WPA, comparando las diferentes medidas de seguridad de ambos tipos. Así mismo, el alumno tendrá que investigar y exponer en que consiste el ataque conocido como KRACK, y el tipo de red al que afecta.

Empezaré por la primera red “segura” que se creó en su momento siendo WEP, donde luego pasaré a lo que resultó fue ser su sustituto WPA que no fue más que una evolución en la seguridad en la red y los motivos de esto los comentaré cuando este repasando WPA en sí.

Entonces WEP se conoce como Wired Equivalent Privacy donde lo que hacía era cifrar los datos que se pasaban por la red de manera que las personas que intentasen hacer uso de Sniffers para poder interceptar los datos, estos, les llegaban cifrados. El problema de WEP apareció en 2001 ya que usaba un protocolo de cifrado bastante fácil de descifrar (siendo este el RC4) y un estudio en dicha fecha demostró como se hacía, haciendo que dicho sistema de seguridad fuese bastante débil y casi cualquiera persona con un conocimiento básico de cifrado podría sacarlas usando descifradores. La técnica más fácil que se puede emplear aquí es inyección de paquetes los cuales son paquetes maliciosos que permites realizar acciones maliciosas y pasan desapercibidas ya que se ven como paquetes normales y de esta manera estos paquetes pueden realizar acciones como mandar paquetes de cierre de sesión o de solicitud de contraseña, o los que se puedan.

Como podemos haber visto, no es muy seguro este método hoy en día, quizás los años cuando salió si, pero hoy en días se puede romper su cifrado en cuestión de minutos, pero ya para eso apareció su remplazo, WPA. WPA, aunque si que es cierto que mejoró bastante la seguridad en la red en comparación con WEP con la adición de TKIP hacía que las claves de ser implementaban por paquetes, es decir, generaba una clave distinta por cada paquete de 128 bits cifrada en AES. Si que es cierto que fue un gran avance en ese ámbito también llego a dar fallas por otro lado, que son las contraseñas y eso se debía a que las personas cuando obtenían los Reuters, les podían cambiar las contraseñas de manera que hace que fuesen más fáciles de recordar pero que hacía que eso fuera su mayor fallo. Usando una técnica parecida en WEP, se inyectaba paquetes a la red con código malicioso, entre este, hacer que el usuario se desconecte y a su vez estar cogiendo los paquetes que se envían y poder descifrarlos. Ahora usando estas dos técnicas juntas hacía que fuese posible hacer que el usuario se desconecte, y cuando hiciese el 4 WayHandShake pues recopilar esos paquetes que se envíen con el sniffer, descifrarlos y ver la contraseña que se envía, ahora para poder descifrarlo, se solía hacer ataque de fuerza bruta por diccionario, y con contraseñas como: “Password”, “12345”, “[Nombre de la persona]” y etc. Hacía que todo fuese más fácil de craquear comparando las palabras que estén en el diccionario, si la palabra del diccionario hasheada era la misma que aparece en la contraseña. Claro hay que entender que dichos diccionarios cuentan con millones de palabras y que requieren un coste muy grande de la GPU.

Y luego de las dos técnicas que he mencionado antes para ambas partes, esta KRACK la cual se aprovecha de los protocolos que usan 4 WayHandShake como lo que utiliza WPA aunque le afecte a WPA2 en su sistema de seguridad y la técnica es mucho más efectiva. KRACK aprovecha más que ver que contraseña se pone y luego intentar descifrarla, se fija en la parte que se establece la clave y KRACK interviene creando/manipulando la clave con una la cual el atacante tiene, y como el atacante tiene la clave con la cual se está cifrando todos los paquetes que mande el usuario y reciba, este puede descifrarlos todos sin problema al usar su clave para descifrarlos. Lo importante aquí es que la clave que se establece es al tercer paso del 4 WayHandShake haciendo que a partir del tercer paso ya se estén enviando los paquetes de manera que este puede ver perfectamente la contraseña del usuario, los datos de este mismo que se estén mandando, como por ejemplo, cuando te conectas a una red, la cual luego tienes una BDD con información tuya privada, el atacante puede sacar esa información sin ningún problema ya que la información que se esté mandando por la red, aunque este cifrada, está cifrada con la clave del atacante, asique este puede ver toda la información sin problema.

Bibliografía:

- V. Mathy, (2017), Breaking WPA2 by forcing nonce reuse, url: https://www.krackattacks.com/
- NetSpot, (Sin Fecha), Protocolos de seguridad inalámbrica: WEP, WPA, WPA2, y WPA3, url: https://www.netspotapp.com/es/blog/wifi-security/wifi-encryption-and-security.html
- P. Javier, (Octubre 2017), Caos en la seguridad WiFi: un repaso a las vulnerabilidades de WEP, WPA, y WPA2, url: https://www.xataka.com/seguridad/caos-en-la-seguridad-wifi-un-repaso-a-las-vulnerabilidades-de-wep-wap-y-wap2
- Wifi Profesionals, (Enero 2019), 4-WAY HANDSHAKE, url: https://www.wifi-professionals.com/2019/01/4-way-handshake