Institut Puig Castellar
Santa Coloma de Gramenet

Cisco Security Network

Projecte de desenvolupament

CFGM Sistemes Microinformàtics i Xarxes

Alejandro Parrilla y Kevin Muñoz
SMX2A
CFGM Sistemes Microinformàtics i Xarxes

Índice

1. Introducción

El proyecto que queremos llevar a cabo es una combinación de nuestra idea anterior, que es el hacking portátil, pero queremos hacerlo más extenso, por lo que hemos decidido hacer la parte del cortafuegos y la parte del atacante, es decir queremos crear un sistema de pasarelas que actúen como cortafuegos para proteger una red local de clientes y desde un atacante ir atacando con varios scripts e ir rompiendo la seguridad y al mismo tiempo ir mejorando la seguridad de la pasarela en la mayor medida posible. El proyecto se llevará a cabo en varias máquinas virtuales, pero queremos crear una página web donde se encuentre la documentación y una aplicación web que permita realizar los ataques/scripts que queremos realizar desde un entorno web.

1.1 Surgimiento de la idea

Anteriormente teníamos la idea de realizar una hacking portátil con una Raspberry Pi, pero debido a la subida de sus precios por la escasez de Stock, nos hemos tenido que adaptar, y escoger otra idea para realizar. De entre varias opciones que nos vinieron a la cabeza (Una matriz de LEDs, tocar arduino para automatizar algo que los demás no vayan a hacer, etc…), decidimos escoger esta fusión entre la idea del hacking portátil y un sistema de cortafuegos en una red interna. Hemos mantenido la idea del hacking portátil por si lográramos conseguir una Raspberry cuando el stock se restablezca. Tenemos en cuenta la posibilidad de que o no se restablezca el Stock a tiempo, o el precio se infle demasiado, entre muchas posibilidades. Es por ello que tenemos en mente un plan B por si no llegamos a conseguir la Raspberry.
(El plan B consistiría en cambiar el lugar de la Raspberry Pi por un portátil como el que utilizamos en clases, es decir, todo lo que se haría en la Raspberry, como instalar una iso de ParrotOS, hacer y probar los scripts con la potencia que esta podría tener, etc… Se pasará a realizar al portátil.)

1.2 ¿De qué se trata?

(Versión Resumida) - Se trata de un sistema de pasarelas que contendrá una red interna donde se encontraran clientes entre otras cosas. Entre las pasarelas también se encontrarán sistemas de seguridad para proteger la red interna, y cosas como el servidor web en el que se encontrará parte de la documentación. Por otra parte estará el atacante que se tratara de un/a portátil/raspberry pi que dispondrá de varios scripts para realizar los ataques que nos interesan para realizar el proyecto, también se encontrará un servidor web donde buscamos poder realizar los ataques de manera gráfica desde una aplicación web.

1.3 Objetivos

Nuestros objetivos han ido cambiando desde el día que planteamos el proyecto hasta el día de la fecha. Los primeros objetivos que tuvimos fue elaborar una pequeña red y hacerla lo más segura y restringida posible entre los equipos de la misma red, como con la parte que llega a internet. Teníamos pensado que al principio fuera insegura, y lograr ir mejorando más y más.

Los objetivos cambiaron al darnos cuenta que con los routers de cisco, tocar la seguridad a ese nivel era más complejo de lo que parecía, por lo que cambiamos el objetivo a uno un tanto más sencillo, pero interesante.

El objetivo que tenemos actualmente es realizar una pequeña red con los routers Cisco que disponemos y máquinas virtuales, y ver cómo reacciona esta red a determinados ataques, es decir, ver si por ejemplo las máquinas virtuales o el router son capaces de soportar un pequeño DDOS

1.4 Diagrama de gantt

Tareas	Marzo				Abril				Mayo			
	1	2	3	4	1	2	3	4	1	2	3	4
Documentación												
Configuración de los routers												
Diseño pagina web												
Codigo HTML												
Testeo de la red entre routers												
Busqueda y testeo de los ataques												
Testeo ataques desde dentro de la red												

2. Preparación de la red

2.1 Materiales

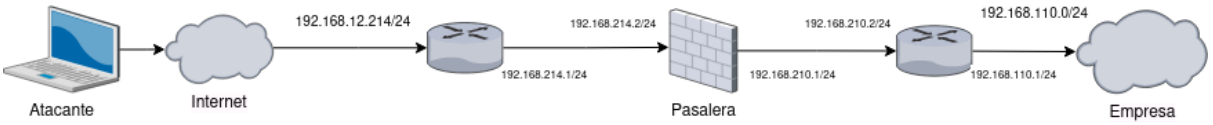
Los materiales tanto físicos como digitales que hemos utilizado son:
Dos routers de Cisco, exactamente el 1841 y el 2801

Un router típico de casa para que haga de switch, exactamente es un switch de vodafone.

Un par de latiguillos de red para conectar todos los dispositivos de red que utilizamos entre ellos, y en la propia red del instituto.
Una máquina virtual cuya función se explicará más adelante.

2.2 Diagramas de red

Diagrama lógico:



2.3 Configuración dispositivos de red

2.3.1 Configuración general routers

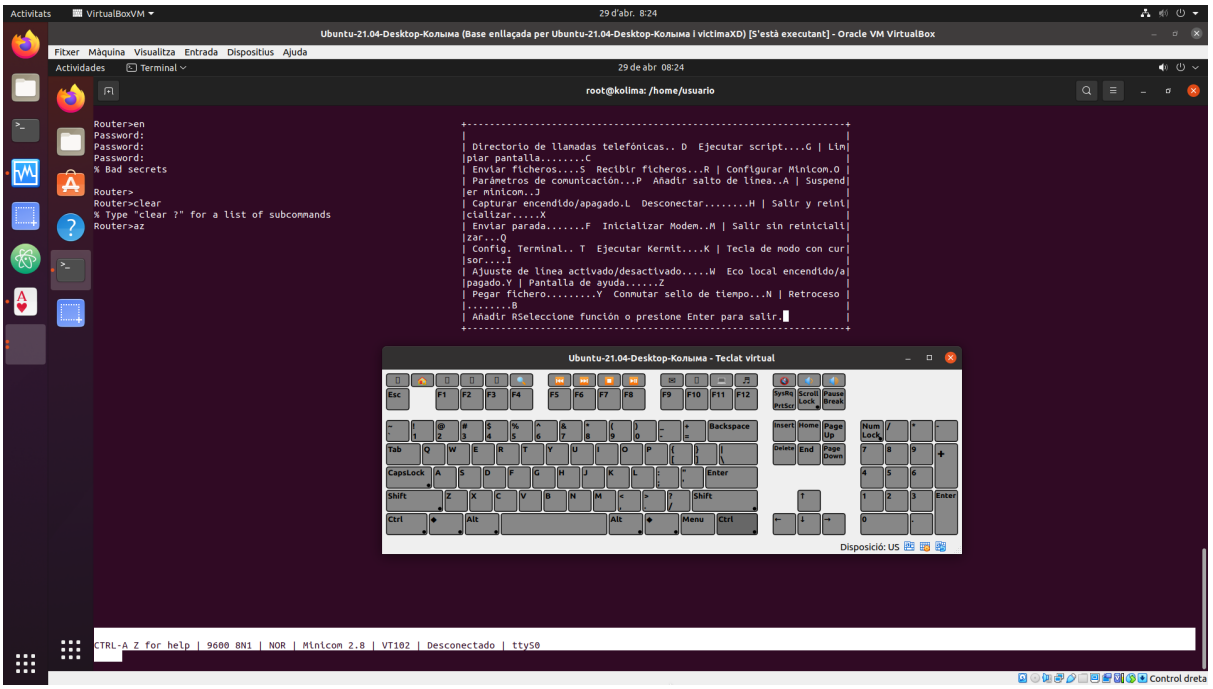
Lo primero que debemos hacer es resetear el router a su configuración de fábrica. Para ello hemos utilizado la información proporcionada en la siguiente página web:

https://www.cisco.com/c/en/us/support/docs/routers/1700-series-modular-access-routers/22187-pswdrec-1700.html

Escribimos minicom y encendemos el router para comprobar si se conecta con nuestra máquina.

```
root@kolima:/home/usuario# minicom
```

Una vez que sabemos que si conecta tenemos que usar "ctrl+a+z" (con el teclado en pantalla de “Virtualbox”)



Después de tener las opciones en pantalla, apagamos el router y lo encendemos entonces pulsamos F, de esa forma entraremos en el modo rommon. Al entrar en el modo rommon, entonces se escribe el siguiente comando.

```
monitor: command "boot" aborted due to user interrupt
rommon 1 > confreg 0x2142
```

Al poner el comando anterior entraremos en el rommon 2, ahí tendremos que escribir reset

```
You must reset or power cycle for new config to take effect
rommon 2 > reset
```

Al poner este comando nos preguntaran varias cosas. A todas les debemos decir que no o pulsar “ctrl+c” (que salta todas las preguntas) para poder usar el modo de administración . Después de entrar al router tendremos que poner “en” o “enable”, de esta forma entraremos en el modo root

Cuando ya nos encontramos dentro del modo root lo primero que hemos hecho es ver cómo está la configuración, para ver si se ha reseteado a la perfección o hemos fallado en el intento. La captura resultante es la siguiente:

```
Router#show running-config
Building configuration...

Current configuration : 756 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
```

Por último y no menos importante debemos guardar la configuración que vayamos a realizar una vez el equipo se encuentre de fábrica, para ello se debe realizar la siguiente línea en la shell:

```
Cisco-1800#copy running-config nvram
Destination filename [nvram]? cisco1800
%Warning:There is a file already existing with this name
Do you want to over write? [confirm]

827 bytes copied in 1.104 secs (749 bytes/sec)
```

En esta línea lo que hemos hecho es guardar la configuración del primer router en la nvram, esta, es la tarjeta de información con la que vienen los routers, por así decirlo se trata de su pequeño disco duro.

Una vez volvemos a iniciar la máquina debemos poner el siguiente comando para poder tener acceso al archivo donde hemos guardado la configuración.

```
Router#configure memory
```

```
Router#copy cisco2800 running-config
Destination filename [running-config]?

915 bytes copied in 0.436 secs (2099 bytes/sec)
```

(pulsar enter sin escribir nada)

2.3.2 Configuración Router Cisco 1841

Después de realizar las acciones pertinentes en el punto anterior, entramos en modo “Root” mediante el comando en (enable) y tras eso nos hemos ido a configurarlo con el siguiente comando:

```
Router#configure terminal
```

Al entrar en la terminal de configuración, lo primero y esencial que hemos realizado es cambiar el nombre del router y la contraseña a lo siguiente, además de desactivar el domain-lookup, esto último deshabilita la búsqueda de dominios, lo cual hace de forma automática al cambiar el Hostname y al escribir un comando de forma incorrecta.

```
Router(config)#hostname Cisco-1800
Cisco-1800(config)#enable secret cisco9614
```

```
Cisco-1800#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Cisco-1800(config)#no ip domain-lookup
```

Y para comprobar que esto ha funcionado, hemos realizado el mismo comando que la vez anterior.

```
Cisco-1800#show running-config
Building configuration...

Current configuration : 807 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Cisco-1800
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$zrje$UvfmTOU4eAe/JTxUF/70a0
```

Una vez hechos estos datos generales básicos, hemos procedido a mirar el estado de los puertos Ethernet, y subir el que utilizaremos.

```
Cisco-1800#show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            192.168.2.1     YES TFTP   administratively down down
FastEthernet0/1            1.0.0.1         YES TFTP   administratively down down
Serial0/0/0                unassigned      YES TFTP   administratively down down
Serial0/0/1                unassigned      YES TFTP   administratively down down
Cisco-1800#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Cisco-1800(config)#interface FastEthernet0/0
```

Tras esto lo que hemos hecho es ponerle la ip que tenemos indicada en nuestro diagrama. Para ello primero debemos entrar en la configuración de la propia interfaz, cambiarle la ip, y tras tener los cambios, la subimos.

```
Cisco-1800(config)#interface FastEthernet0/0
Cisco-1800(config-if)#ip address 192.168.12.214 255.255.255.0
Cisco-1800(config-if)#no shutdown
Cisco-1800(config-if)#exit
Cisco-1800(config)#interface FastEthernet0/1
Cisco-1800(config-if)#ip address 192.168.214.1 255.255.255.0
Cisco-1800(config-if)#no shutdown
Cisco-1800(config-if)#exit
```

En la siguiente imagen se puede ver como quedarían las interfaces de red subidas con sus ip's como corresponden:

```
Cisco-1800#show ip interface brief
Interface                  IP-Address      OK? Method Status                Protl
FastEthernet0/0            192.168.12.214  YES manual up                    down
FastEthernet0/1            192.168.214.1   YES manual up                    down
Serial0/0/0                unassigned      YES NVRAM  administratively down down
Serial0/0/1                unassigned      YES NVRAM  administratively down down
```

Con la Interfaz subida,

Aunque ya tengamos las ip's configuradas, no está todo terminado, pues, nos queda lo último y no menos importante de configurar, lo cual se trata de las ip-routes en este router.

```
Cisco-1800(config)#ip route 10.0.104.0 255.255.255.0 192.168.214.2
Cisco-1800(config)#ip route 192.168.210.0 255.255.255.0 192.168.214.2
Cisco-1800(config)#ip route 192.168.110.0 255.255.255.0 192.168.214.2
```

Con esto, ya hemos terminado de configurar este router.

2.3.3 Configuración Router Cisco 2801

En este router realizaremos las mismas configuraciones que en el anterior, cambiando el nombre del dispositivo, y las ips.

```
Cisco-2800#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Cisco-2800(config)#hostname Cisco-2800
Cisco-2800(config)#enable secret cisco9614
Cisco-2800(config)#no ip domain-lookup
Cisco-2800(config)#exit
```

```
Cisco-2800#show running-config
Building configuration...

Current configuration : 806 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Cisco-2800
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$cIY0$TCOvXbhacyMQlcWnOT/fb.
```

Como se aprecia en la siguiente captura, la ip del router es la asignada en el diagrama, y tiene las conexiones configuradas de la forma apropiada.

```
Cisco-2800#show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            192.168.210.2   YES manual up                    down
FastEthernet0/1            192.168.110.1   YES manual up                    down
Serial0/3/0                unassigned      YES TFTP   administratively down down
Serial0/3/1                unassigned      YES TFTP   administratively down down
```

```
Cisco-2800(config)#interface FastEthernet0/0
Cisco-2800(config-if)#ip address 192.168.210.2 255.255.255.0
Cisco-2800(config-if)#no shutdown
Cisco-2800(config-if)#exit
Cisco-2800(config)#interface FastEthernet0/1
Cisco-2800(config-if)#ip address 192.168.110.1 255.255.255.0
Cisco-2800(config-if)#no shutdown
Cisco-2800(config-if)#exit
```

```
Cisco-2800(config)#ip route 10.0.104.0 255.255.255.0 192.168.110.50
Cisco-2800(config)#ip route 192.168.12.0 255.255.255.0 192.168.210.1
Cisco-2800(config)#ip route 192.168.214.0 255.255.255.0 192.168.210.1
```

```
Cisco-2800#copy running-config nvram
Destination filename [nvram]? cisco2800
%Warning:There is a file already existing with this name
Do you want to over write? [confirm]

915 bytes copied in 4.484 secs (204 bytes/sec)
```

2.3.4 Configuración clientes

A todos los clientes de la red que forma parte de la “Empresa” se les tiene que configurar unos pocos aspectos generales.

Uno de los aspectos a configurar es el Netplan, pues, a los clientes les ponemos una ip fija mediante el netplan.

```
# Let NetworkManager manage all devices on this system
network:
  ethernets:
    enp0s3:
      dhcp4: false
      addresses: [192.168.110.50/24]
      gateway4: 192.168.110.1
      nameservers:
        addresses: [192.168.12.10, 1.1.1.1, 8.8.8.8]
  version: 2
```

Estos aspectos que se les debe configurar es una ip que forme parte de una VPN. La configuración de la VPN se hace mediante el Wireguard-tools. Su archivo de configuración se debería de visualizar de la siguiente forma:

```
[Interface]
PrivateKey = iPiG0BoVDTBhEgQSpQN0CVgeqJ51RuTFz3V4o8Q+p1s=
Address = 10.0.104.215/24

[Peer]
Endpoint = 145.239.68.121:40004
PublicKey = u6UY7VpYS5w+Df6uoW5P2OLpy2uiFDWjQXTka4vTBCk=
AllowedIPs = 10.0.104.0/24
PersistentKeepalive = 25
```

Además, utilizado un playbook creado por Victor para que automáticamente descargue y configure el wireshark en los clientes de tal manera que cualquier usuario sin permisos de uso pueda hacer uso del mismo.

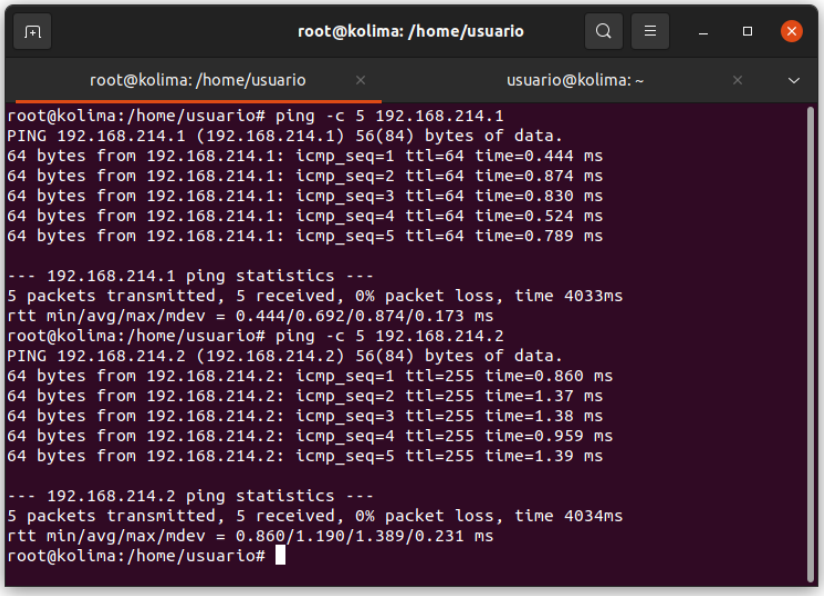
```
- name: Wireshark let non root to capture traffic
  # Before installing any package, this only sets the preference for the question
  debconf:
    name='wireshark-common'
    question='wireshark-common/install-setuid'
    vtype='boolean'
    value='true'
  become: yes
  notify:
    - desinstala wireshark-common
    - instala wireshark-common

- name: Crea el grupo wireshark
  group:
    name: wireshark
    state: present
    system: yes

- name: Instala Wireshark
  apt:
    pkg:
    - wireshark
```

2.4 Testeo de la red

Una vez hemos configurado los routers procedimos a probar si la configuración realizada es funcional con toda la estructura montada. El resultado de esto se puede apreciar en la siguiente imagen:



En la imagen se aprecia que los pings entre los routers, y la pasarela van a la perfección, pero eso no es todo. Hay un ping en el que se pierden paquetes, y es la conexión más importante de todas de las que disponemos, pues, es la conexión que da a la red de internet del centro.

Este es el primer problema con el que nos hemos topado, pues, al principio no sabíamos a qué se debía dicho problema, pero tras analizarlo, y preguntar al profesorado, hemos llegado a la conclusión de que se debe a que los routers no están enmascarando como deben hacerlo, es decir, no les funciona el NAT.
Hemos realizado múltiples configuraciones distintas en busca de solucionarlo, pero la única que ha funcionado en esto es la siguiente (gracias a Juan Morote):

```
Cisco-1800(config)#interface Fa0/0
Cisco-1800(config-if)#ip nat outside
Cisco-1800(config-if)#
*Jan  1 00:04:06.827: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Cisco-1800(config-if)#interface Fa0/1
Cisco-1800(config-if)#ip nat inside
Cisco-1800(config-if)#exit
Cisco-1800(config)#access-list 1 permit any
Cisco-1800(config)#ip nat source list 1 interface Fa0/0 overload
Cisco-1800(config)#
```

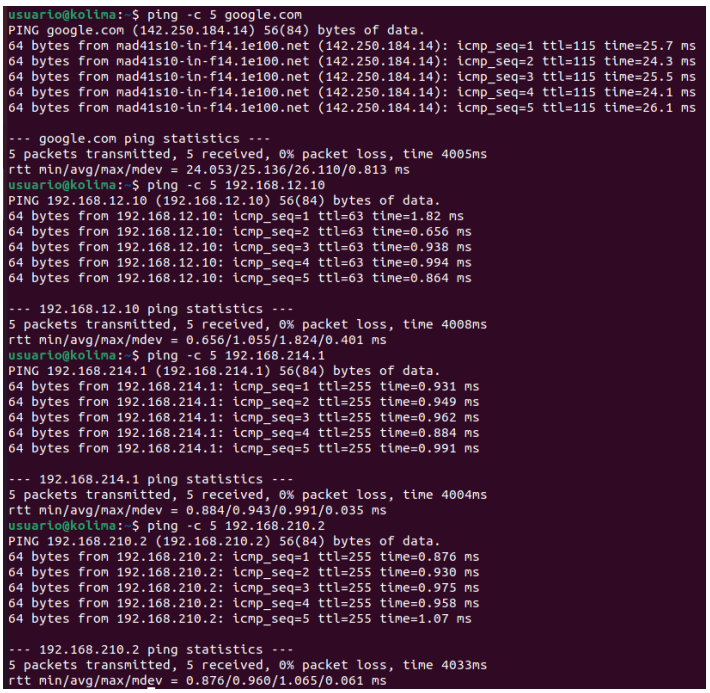
Con esta configuración el problema principal ha sido resuelto, pero tristemente para nosotros, este problema no ha sido el único que se nos ha presentado.

El siguiente problema que se nos ha presentado se trata de que desde la red, no podemos interactuar con con internet, es decir, si tratamos de conectarnos a internet desde un cliente, no lo logramos, debido a que el router no sabe acceder. Esto no ha sido complicado de solucionar, pues lo único que hemos tenido que hacer es añadir una ruta que hace de Default Gateway. Esto lo hemos hecho de la siguiente manera:

```
ip classless
ip route 0.0.0.0 0.0.0.0 192.168.12.10
```

Debemos añadir esta ruta a todos los dispositivos de nuestra red.

Tras todos estos errores resueltos, hemos vuelto a probar la red haciendo todos los ping posibles. El resultado de esta prueba a sido el siguiente:

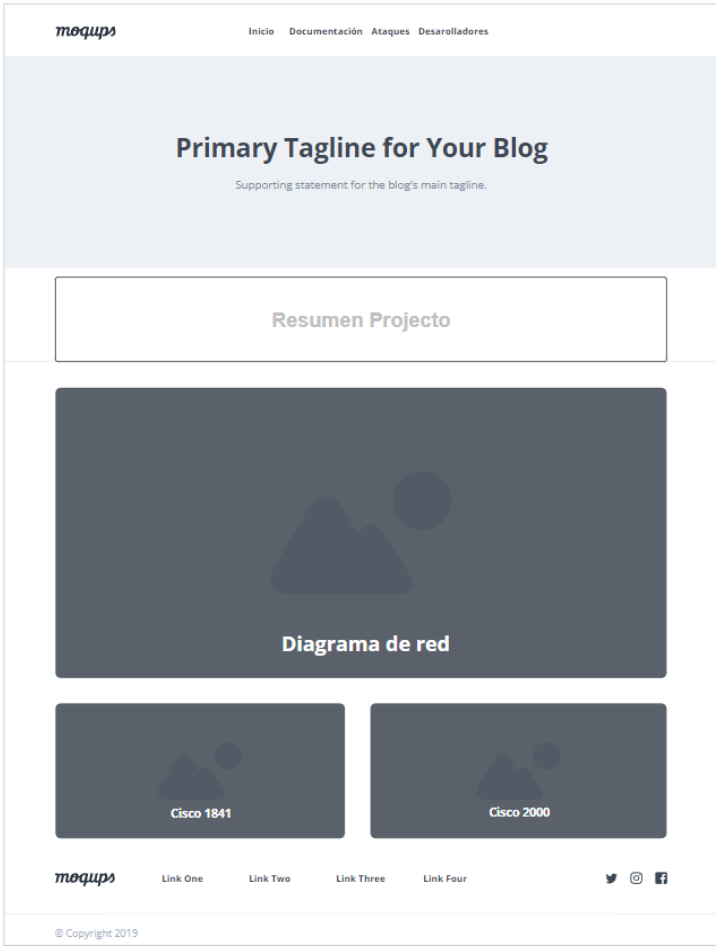


3. Pàgina web

3.1 Diseño página web

Antes de iniciar nuestra página web hicimos uso de una página externa para realizar un diseño primitivo de lo que sería nuestra página web. Realizamos un diseño de todas las páginas, aunque actualmente haya algunas páginas que cambiamos con respecto al diseño

Diseño página inicio:

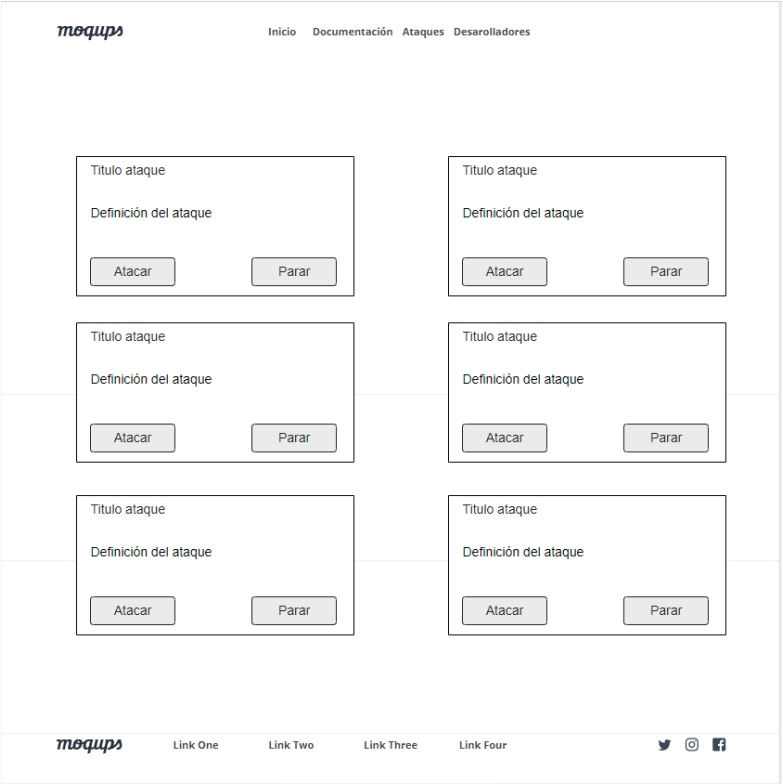


Diseño página memoria/documentación:

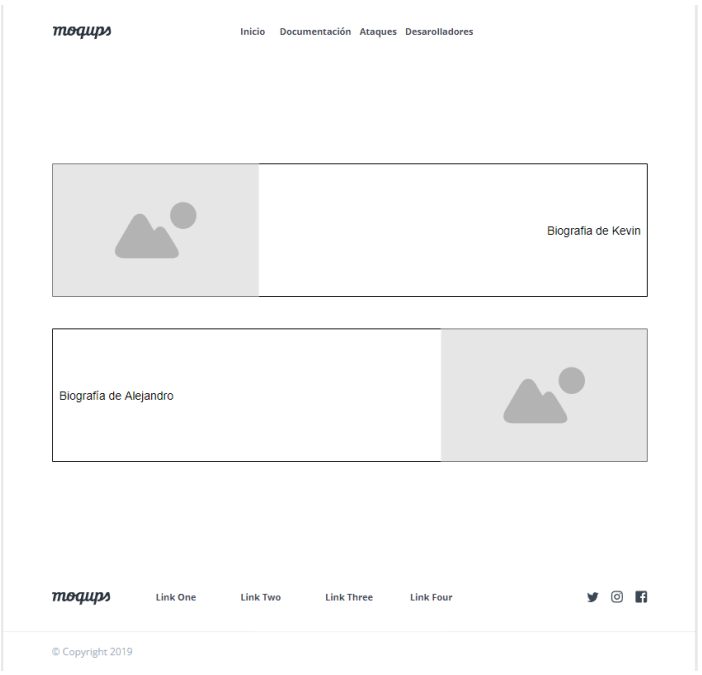
Diseño página ataques:

Página de ataques antigua

Página de ataques actual:



Diseño página desarrolladores:



.

La idea de cambiar por completo la página de ataques surgió de una reflexión que tuvimos mientras pensábamos en cómo realizar los ataques desde la página web, ya que vimos que sería una gran carga de trabajo las reformas necesarias que

necesitábamos para realizar y parar los ataques, por ello con el cambio de diseño tomamos eso en cuenta y se añadió una descripción de cada ataque para que sea más fácil de entender.

3.2 Página de inicio

En la página inicial tenemos un poco de resumen del punto 1 de este documento. Disponemos de la introducción, imágenes del diagrama,los routers, y tenemos un resumen del objetivo actual.

En cuanto al código HTML y CSS es muy simple, pues lo más complejo que tenemos serían las imágenes. No obstante hemos hecho uso de un grid para poner las imágenes de los routers uno al lado del otro.

```
<div id="fotos-grid">
  <div class="relativo">
    <img id="router1" src="imagenes/router-1841.jpeg" alt="">
    <h3 class="routers-nombre titulo-h3">Router Cisco 1841</h3>
  </div>
  <div class="relativo">
    <img id="router2" src="imagenes/router-2801.jpeg" alt="">
    <h3 class="routers-nombre titulo-h3">Router Cisco 2801</h3>
  </div>
</div>
```

```
#fotos-grid {
  display: grid;
  grid-template-columns: 1fr 1fr
}
#fotos-grid div {

}
#router1 {
  padding-right: 0.5em;
}
#router2 {
  padding-left: 0.5em;
}
```

3.3 Creación del Menú y el Footer

A la hora de crear nuestro menú en la página web hicimos una lista con los enlaces hacia las demás páginas. También hicimos mediante Javascript que insertando solo el parámetro **<menu>** dentro del HTML, se añade todo el código del propio menú. Esto lo hemos hecho tanto con el menú como con el footer.

Tenemos dos menús y dos footers distintos. Uno para todas las páginas, y otro exclusivo para únicamente la página de ataques. Esto lo hemos hecho de esta manera ya que la página de ataques se encuentra en una máquina distinta a lo demás de la página web. Lo mismo ocurre con el footer.

Para crear el footer lo que hicimos fue un grid de 2 columnas, 1 columna para el contacto, cuyo contenido son 2 enlaces hacia nuestros correos, y otra columna para un menú en miniatura y simplificado.

```
<div id="menu">
  <img class="logo" src="imagenes/logo.png" alt="portada">
  <ul>
    <li><a href="index.html">INICIO</a></li>
    <li><a href="resumen.html">DOCUMENTACIÓN</a></li>
    <li><a href="manual.html">MANUAL USUARIO</a></li>
    <li><a href="ataques/ataques.php">ATAQUES</a></li>
    <li><a href="desarrolladores.html">DESARROLLADORES</a></li>
  </ul>
</div>
```

```
<div id="footer">
  <img class="logo" src="imagenes/logo.png" alt="logo">
    <div id="correos">
      <h4>Nuestros Correos</h4>

      <a href="https://mail.google.com/mail/u/0/#inbox?compose=new">aparrilla@elpuig.xeill.net</a> <br>
      <a href="https://mail.google.com/mail/u/0/#inbox?compose=new">amunozgil@elpuig.xeill.net</a>
    </div>
    <div id="menu-footer">
      <ul>
        <li><h4>Paginas</h4></li>
        <li><a href="index.html">Inicio</a></li>
        <li><a href="resumen.html">Documentación</a></li>
        <li><a href="manual.html">Manual usuario</a></li>
        <li><a href="ataques/ataques.php">Ataques</a></li>
        <li><a href="desarrolladores.html">Desarrolladores</a></li>
      </ul>
    </div>
  <img id="licencias" src="imagenes/licencias.png" alt="licencia">
</div>
```

3.4 Página de documentación, manual de usuario y desarrolladores

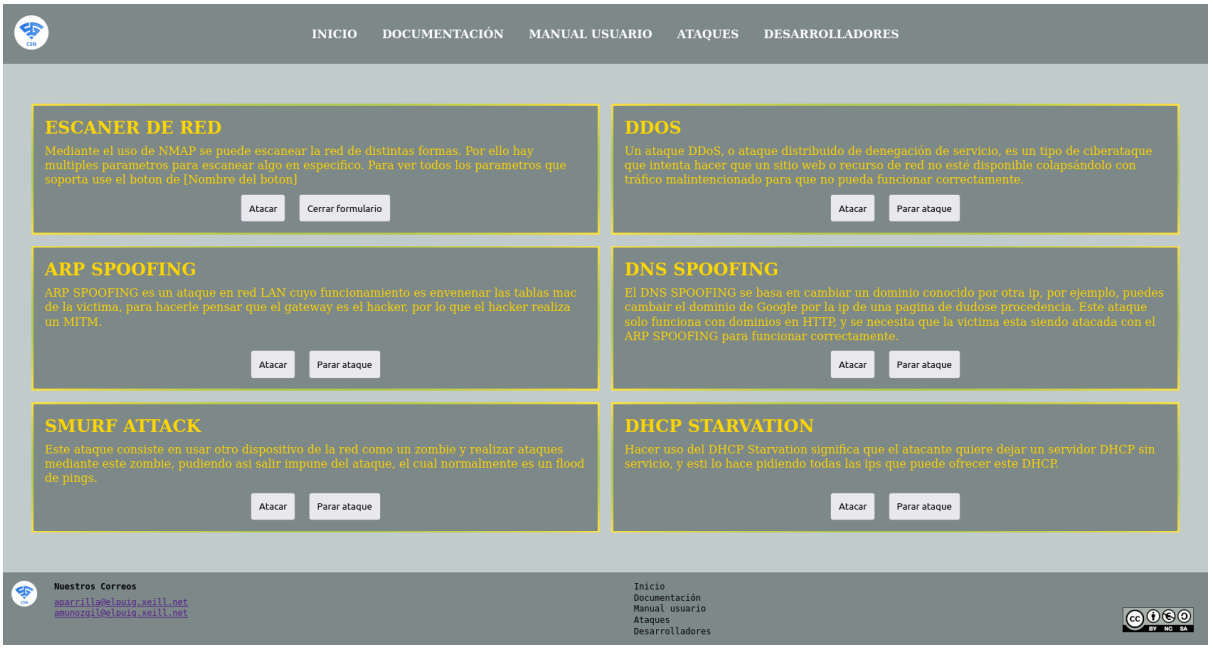
Tanto la página de documentación como el pequeño manual de usuario tienen el mismo formato. Este formato es sencillo, se basa en poner pequeños subtítulos seguidos con el párrafo, y una imagen, solo que en vez de hacer uso de una imagen cuando se trata de código, utilizamos un parámetro llamado **<code>** en el cual ponemos todas las líneas de comandos, quedando de tal forma que parece una terminal dentro de la página web.

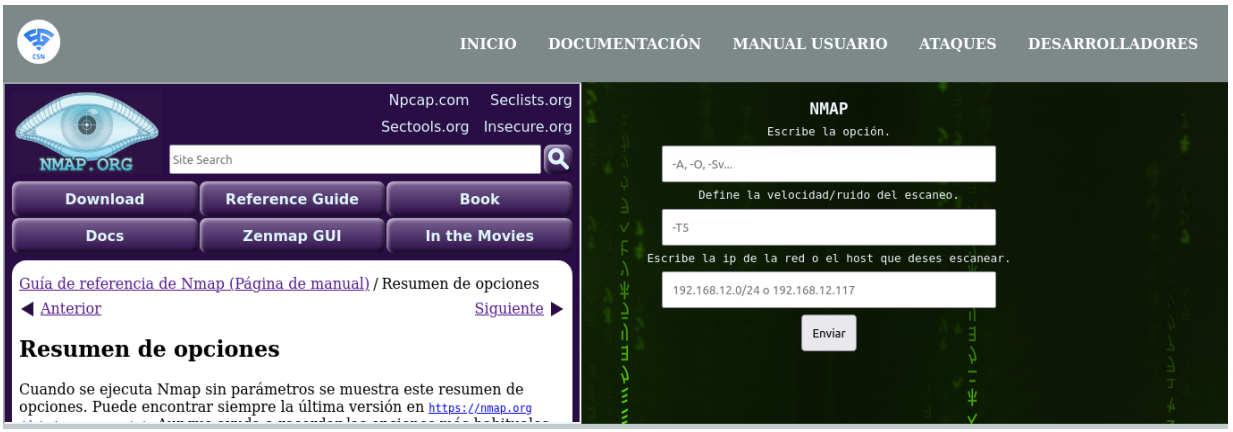
```
<h2><li>CONFIGURACION CISCO 1841</li></h2>
  <p>La configuración especial que tuvo el cisco 1841 en comparación al 2801 es la siguiente</p><br>
  <code>
    <p>
      Cisco-1800(config)#interface FastEthernet0/0 <br>
      Cisco-1800(config-if)#ip address 192.168.12.214 255.255.255.0 <br>
      Cisco-1800(config-if)#no shutdown <br>
      Cisco-1800(config-if)#exit <br>
      Cisco-1800(config)#interface FastEthernet0/1 <br>
      Cisco-1800(config-if)#ip address 192.168.214.1 255.255.255.0 <br>
      Cisco-1800(config-if)#no shutdown <br>
      Cisco-1800(config-if)#exit <br>
    </p>
  </code>
```

3.5 Página de ataques

La página de ataques está conformada de HTML, CSS, JavaScript y PHP. En ella se encuentran 6 ataques diferentes: Escaneo de red, DDOS, ARP SPOOFING, DNS SPOOFING, SMURF ATTACK y DHCP STARVATION. Cada ataque cuenta con su propio formulario a rellenar, desde el cual se pueden introducir los datos necesarios para realizar el ataque solicitado.

En el apartado visual la página es bastante simple, lo más llamativo en término visual serían los cuestionarios que sirven para atacar entre ellos. El que más destacaría sería el del escaneo de red, ya que cuenta con IFRAME que apunta a la página donde se pueden ver las opciones del ataque.





El código de la página de ataques es lo que ha presentado sus dificultades, pues era necesario tener todo bien identificado y ordenado, para que el JavaScript y el PHP no diesen errores.

Ejemplo de formulario:

```
<div class="formularios" id="F-NMAP" style="display: none;">
  <div id="INFO-NMAP">
    <iframe src="https://nmap.org/man/es/man-briefoptions.html" width="100%" height="100%"></iframe>
  </div>
  <form id="NMAP-form" action="scripts.php" method="post" onsubmit="return fetchNMAP();">
    <input type="hidden" name="NMAP" value="1">
    <h2>NMAP</h2>
    <p>Escribe la opción.</p>
    <input type="text" name="OPCION" placeholder="-A, -O, -Sv...">
    <br>
    <p>Define la velocidad/ruido del escaneo.</p>
    <input type="text" name="VELOCIDAD" placeholder="-T5">
    <br>
    <p>Escribe la ip de la red o el host que deses escanear.</p>
    <input type="text" name="HOST_NMAP" placeholder="192.168.12.0/24 o 192.168.12.117">
    <br>
    <button id="carga" type="submit" name="button">Enviar</button>
    <div class="loader" id="loader" style="display: none;"></div>
  </form>
</div>
```

Ejemplo de caja de ataques:

```
  <div class="ataques">
    <h2>ESCANER DE RED</h2>
    <p>Mediante el uso de NMAP se puede escanear la red de distintas formas. Por ello hay multiples
    <form class="boton" id="ataque1" action="scripts.php" method="post" onsubmit="return kill1();">
      <button id="NMAP" type="button" name="button">Atacar</button>
      <button type="submit" name="stop">Cerrar formulario</button>
      <input type="hidden" name="NMAP-stop" value="1">
    </form>
  </div>
```

Estas cajas se encontraban todas dentro de un DIV que tenía la función de grid para ordenar los ataques de 2 en 2.

3.6 JavaScript y Php para ejecutar los ataques

La tarea de hacer que los ataques funcionaran, ha sido la tarea más difícil e importante que hemos realizado durante este proyecto. Para conseguir que los ataques se pudieran realizar desde la página web hemos necesitado un total de 3 ficheros diferentes, 2 de ellos en lenguaje JavaScript y el último en PHP.

En el fichero de PHP se encuentran las líneas de código que hacen posible la ejecución de los ataques y su detención, cada uno de los ataques cuenta con su propio apartado en el fichero scripts.php para ser ejecutado y detenido. El código de este fichero sería el siguiente (ejemplo de un ataque):

```php
<?php
  $DDOS = 'python2 scripts/Torshammer/torshammer.py -t'." ".$_POST['IP']." ".'-p'." ".$_POST['PORT']." ".'-r'." ".$_POST['RE'];

  if (isset($_POST['DDOS']))
  {
    print_r($DDOS);
    shell_exec($DDOS);
  }

   if(isset($_POST['DDOS-stop']))
    {
      shell_exec("kill -9 `pidof python2`");
    }
?>
```

Los siguiente ficheros necesarios para el funcionamiento de los ataques se han realizado con JavaScript, empezaré explicando el fichero de main.js.
En el fichero main.js a parte de encontrarse el código que permite ver el menú y el footer, se encuentra el código necesario para hacer aparecer los formularios a rellenar para realizar los ataques, para hacerlos desaparecer (para que no se superpongan entre ellos) y el principio de la animación de carga para el escaneo de red.

Hacer aparecer los formularios (es muy similar a el código necesario para la animación de carga del escaneo de red.):

```javascript
var NMAP = document.getElementById('NMAP');

NMAP.addEventListener("click", ataque_NMAP);

function ataque_NMAP() {
  ocultarFormularios();
  var Form_NMAP = document.getElementById('F-NMAP');
  console.log(Form_NMAP.style.display);
 if (Form_NMAP.style.display == 'none') {
   Form_NMAP.style.display = 'block';
 }
}
```

Ocultar los formularios:

```javascript
function ocultarFormularios() {

  var Form_NMAP = document.getElementById('F-NMAP');
  Form_NMAP.style.display = 'none';
  var Form_DDOS = document.getElementById('F-DDOS');
  Form_DDOS.style.display = 'none';
  var Form_ARPSPOOFING = document.getElementById('F-ARPSPOOFING');
  Form_ARPSPOOFING.style.display = 'none';
  var Form_DNSSPOOFING = document.getElementById('F-DNSSPOOFING');
  Form_DNSSPOOFING.style.display = 'none';
  var Form_SMURF = document.getElementById('F-SMURF');
  Form_SMURF.style.display = 'none';
  var Form_DHCP = document.getElementById('F-DHCP');
  Form_DHCP.style.display = 'none';
}
```

El 2n fichero de JavaScript que usamos en nuestra web es attacks.js este fichero es muy necesario, ya que contiene el código para evitar que al enviar los formularios de los ataques o al detenerlos, nos envíe a la página de script.php. También contiene el código necesario para que el escaneo de red se muestre en pantalla en una alerta y contiene la última parte del código para la animación de carga del escaneo de red.

Evitar el reenvío a scripts.php

```
function fetchcall1 () {
  // (B1) GET FORM DATA
  var data = new FormData(document.getElementById("ARPSPOOFING-form"));

  // (B2) FETCH
  fetch("scripts.php", { method: "POST", body: data })
  .then(res => res.text())
  .then((txt) => {
    console.log(txt);
  })
  .catch((err) => { console.error(err); });
  return false;
}
```

También es necesaria una pequeña opción que se encuentra en la página principal escrita en los forms, la cual es la siguiente: `onsubmit="return fetchcall();"`

Escaneo de red (en lugar de enviar la información a la consola la envía a un alert):

```
function fetchNMAP () {
  // (B1) GET FORM DATA
  var data = new FormData(document.getElementById("NMAP-form"));
  // (B2) FETCH
  fetch("scripts.php", { method: "POST", body: data })
  .then(res => res.text())
  .then((txt) => {
    ocultarCarga();
    alert(txt);
    envioNMAP();

  })
  .catch((err) => { console.error(err); });
  return false;
}
```

3.7 Testeo de los ataques

NMAP

Para probar el escaneo de red hemos utilizado solamente tres parámetros, que son los mismos que están de placeholder en el campo donde se deben meter, además de que todas las pruebas lo hemos hecho con el T5, ya que es el escaneo más rápido que se puede realizar.

```
localhost

 Starting Nmap 7.80 ( https://nmap.org ) at 2022-05-27 08:46 CEST
Nmap scan report for 192.168.12.106
Host is up (0.00021s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
2049/tcp open  nfs
9100/tcp open  jetdirect
MAC Address: D0:50:99:2B:63:7D (ASRock Incorporation)

Nmap done: 1 IP address (1 host up) scanned in 1.25 seconds
```

```
localhost

 Starting Nmap 7.80 ( https://nmap.org ) at 2022-05-27 08:47 CEST
Nmap scan report for 192.168.12.106
Host is up (0.00017s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE    VERSION
22/tcp   open  ssh        OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu
Linux; protocol 2.0)
80/tcp   open  http       Apache httpd 2.4.41 ((Ubuntu))
111/tcp  open  rpcbind    2-4 (RPC #100000)
2049/tcp open  nfs_acl    3 (RPC #100227)
9100/tcp open  jetdirect?
MAC Address: D0:50:99:2B:63:7D (ASRock Incorporation)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.85 seconds
```

DDOS

La mejor forma de demostrar que el ataque DDOS funciona es mostrando la siguiente imagen de wireshark:

8324	55.787130881	192.168.12.118	192.168.12.164	TCP	66 80 → 60172 [ACK] Seq=1 Ack=263 Win=65024 Len=0 TSval=24204786…
8325	55.787161534	192.168.12.164	192.168.12.118	TCP	67 60186 → 80 [PSH, ACK] Seq=286 Ack=1 Win=64256 Len=1 TSval=246…
8326	55.787170816	192.168.12.118	192.168.12.164	TCP	66 80 → 60186 [ACK] Seq=1 Ack=287 Win=64896 Len=0 TSval=24204786…
8327	55.787364449	192.168.12.164	192.168.12.118	TCP	67 60726 → 80 [PSH, ACK] Seq=220 Ack=1 Win=64256 Len=1 TSval=246…
8328	55.787384932	192.168.12.118	192.168.12.164	TCP	66 80 → 60726 [ACK] Seq=1 Ack=221 Win=65024 Len=0 TSval=24204786…
8329	55.788374784	192.168.12.164	192.168.12.118	TCP	67 59996 → 80 [PSH, ACK] Seq=285 Ack=1 Win=64256 Len=1 TSval=246…
8330	55.788413350	192.168.12.118	192.168.12.164	TCP	66 80 → 59996 [ACK] Seq=1 Ack=286 Win=64896 Len=0 TSval=24204786…
8331	55.793558464	192.168.12.164	192.168.12.118	TCP	67 59900 → 80 [PSH, ACK] Seq=275 Ack=1 Win=64256 Len=1 TSval=246…
8332	55.793602990	192.168.12.118	192.168.12.164	TCP	66 80 → 59900 [ACK] Seq=1 Ack=276 Win=64896 Len=0 TSval=24204786…
8333	55.793658117	192.168.12.164	192.168.12.118	TCP	67 60238 → 80 [PSH, ACK] Seq=224 Ack=1 Win=64256 Len=1 TSval=246…
8334	55.793669739	192.168.12.118	192.168.12.164	TCP	66 80 → 60238 [ACK] Seq=1 Ack=225 Win=65024 Len=0 TSval=24204786…
8335	55.796555978	192.168.12.164	192.168.12.118	TCP	67 60480 → 80 [PSH, ACK] Seq=274 Ack=1 Win=64256 Len=1 TSval=246…
8336	55.796603760	192.168.12.118	192.168.12.164	TCP	66 80 → 60480 [ACK] Seq=1 Ack=275 Win=64896 Len=0 TSval=24204786…
8337	55.796643664	192.168.12.164	192.168.12.118	TCP	67 59988 → 80 [PSH, ACK] Seq=279 Ack=1 Win=64256 Len=1 TSval=246…
8338	55.796654151	192.168.12.118	192.168.12.164	TCP	66 80 → 59988 [ACK] Seq=1 Ack=280 Win=64896 Len=0 TSval=24204786…

Como se puede ver en la captura, el wireshark ha recibido múltiples paquetes TCP, los cuales han sido enviados por el DDOS de nuestra página. No obstante, por si hay alguna duda, dejamos aquí un video donde se puede ver como el ataque funciona.

ARP SPOOFING

Al igual que con el ataque DDOS, para demostrar su funcionamiento solo debemos realizar una captura desde wireshark, y ver si ha recibido muchos paquetes ARP. Esta imagen demuestra que es funcional:

937	56.503896353	PcsCompu_8a:f3:b2	Broadcast	ARP	60 Who has 192.168.12.10? Tell 192.168.12.164
938	56.503901744	PcsCompu_8a:f3:b2	Broadcast	ARP	60 Who has 192.168.12.10? Tell 192.168.12.164

DNS SPOOFING

Aquí se puede ver una captura del DNS Spoofing siendo funcional, no obstante, por si la captura no es suficiente prueba de que es funcional, hemos realizado un video del ataque en funcionamiento.

SMURF ATTACK

Para comprobar el funcionamiento del ataque debemos repetir el mismo proceso que para el DDOS y para el ARP SPOOFING, solo que en este caso los paquetes serán ICMP. Además, debemos fijarnos en que la ip de destino es la ip que hemos

registrado como Zombie al realizar el ataque.

```
6925… 170.55817316… 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=24121/14686, ttl=64 (requ…
6925… 170.558195631 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=24377/14687, ttl=64 (repl…
6925… 170.558200186 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=24377/14687, ttl=64 (requ…
6925… 170.558205590 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=24633/14688, ttl=64 (repl…
6925… 170.558209819 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=24633/14688, ttl=64 (requ…
6925… 170.558226459 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=24889/14689, ttl=64 (repl…
6925… 170.558231345 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=24889/14689, ttl=64 (requ…
6925… 170.558320587 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=25145/14690, ttl=64 (repl…
6925… 170.558325270 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=25145/14690, ttl=64 (requ…
6925… 170.558345924 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=25401/14691, ttl=64 (repl…
6925… 170.558351368 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=25401/14691, ttl=64 (requ…
6925… 170.558356308 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=25657/14692, ttl=64 (repl…
6925… 170.558360266 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=25657/14692, ttl=64 (requ…
6925… 170.558434847 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=25913/14693, ttl=64 (repl…
6925… 170.558440303 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=25913/14693, ttl=64 (requ…
6925… 170.558470945 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=26169/14694, ttl=64 (repl…
6925… 170.558476595 192.168.12.118   192.168.12.164   ICMP   42 Echo (ping) reply    id=0x911b, seq=26169/14694, ttl=64 (requ…
6925… 170.558483724 192.168.12.164   192.168.12.118   ICMP   60 Echo (ping) request  id=0x911b, seq=26425/14695, ttl=64 (repl…
```

DHCP STARVATION

A la hora de comprobar este ataque, tenemos que sniffar la paqueteria entrante mediante el wireshark como en los anteriores, en busca de paquetes de DHCPDiscover, o DHCP offer. Si logramos localizar ambos tipos de paquetes enviados de forma continua, significa que el ataque ha ido a la perfección.

```
1558 126.375468464 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1566 126.583351234 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1567 126.583358192 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1570 126.792169995 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1571 126.792177434 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1572 127.000395948 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1573 127.000402526 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1574 127.211927065 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1575 127.211933720 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1576 127.421358533 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1577 127.421363754 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1579 127.632264339 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1580 127.632269970 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1582 127.841282095 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
1583 127.841289572 0.0.0.0       255.255.255.255   DHCP    298 DHCP Request  - Transaction ID 0x0
```

4. Conclusiones

Como conclusión nos gustaría añadir que aunque la idea original era completamente distinta a la actual, ya que en ella hacíamos uso de una raspberry y otros sistemas, de igual manera creemos que la forma actual del proyecto es más interesante a la antigua y por ello estamos bastante satisfechos con los resultados finales.
También nos hemos dado cuenta que gracias a él, hemos aprendido muchas cosas nuevas como: PHP, JavaScript, SO Cisco, entre otras cosas. También gracias al proyecto hemos aprendido a ser autosuficientes, ya que hemos realizado un variado repertorio de acciones, de las cuales éramos completamente ignorantes hasta que nos pusimos a investigar para el proyecto.

5. Bibliografía

Página para diagrama: https://www.diagrams.net/
Isard: https://pilotfp.gencat.isardvdi.com/login

Toda la documentación de cisco utilizada

Documentación general:
https://www.cisco.com/c/en/us/td/docs/routers/access/1800/1801/software/configuration/guide/scg.pdf
https://www.cisco.com/c/en/us/support/routers/1800-series-integrated-services-routers-isr/series.html?dtid=osscdc000283
https://www.hardreset99.com/routers/cisco/cisco-1841-factory-reset/
https://learningnetwork.cisco.com/s/question/0D53i00000Kt35KCAR/default-gateway-command
https://community.cisco.com/t5/other-network-architecture/ip-masquerading-on-cisco-router/td-p/6882

Páginas para realizar nuestra pagina web

Diseño páginas: https://moqups.com/

Uso de java para ejecutar los ataques:
https://www.quora.com/How-can-I-make-a-JavaScript-code-to-run-an-installed-program-and-do-something
https://www.php.net/manual/es/function.shell-exec.php

Php: https://www.geeksforgeeks.org/php-shell_exec-vs-exec-function/
https://www.geeksforgeeks.org/php-string-functions-complete-reference/
https://stackoverflow.com/questions/20738329/how-to-call-a-php-function-on-the-click-of-a-button

Pagina para que apache disponga de permisos ROOT:
https://serverfault.com/questions/404105/apache-sudoers-access

Páginas origen de los ataques utilizados

Documentación de Nmap: https://nmap.org/man/es/man-briefoptions.html

Torshammer:
https://elpuig.xeill.net/Members/jordifarrero/2014-15-seguretat-en-xarxes-sm2ab-diurn/uf2-scripts-demo

ARP-Spoofing:
https://www.thepythoncode.com/article/building-arp-spoofer-using-scapy

DNS-Spoofing:
https://www.thepythoncode.com/article/make-dns-spoof-python

Script DHCP-Starvation :
https://github.com/Kurlee/DHCP-Starvation/blob/master/DHCP_Starvation.py

6. Anexos

Repositorio de GitHub: https://github.com/AlejandroP02/Sintesis
Pagina en GitHub: https://alejandrop02.github.io/Sintesis/
Descarga zip:
https://github.com/AlejandroP02/Sintesis/archive/refs/heads/main.zip

Videos de los ataques funcionando:
DDOS: https://youtu.be/AgZXq60dvCM
ARP SPOOFING: https://youtu.be/-wM5PpSHB-g
DNS SPOOFING: https://youtu.be/9VUbd7IpPgE
SMURF ATTACK: https://youtu.be/bbd9li-c26U
DHCP STARVATION: https://youtu.be/MucmwbZGWCo