

Redes en Cloud y Ciberseguridad

IES Leonardo Da Vinci

Curso 2020/2021



Álvaro Gómez Peña

David Riballo Lence

José Luis del Rio Gómez



Índice de contenido

1.	Introducción del proyecto.....	4
2.	Justificación Proyecto.....	4
3.	Tecnologías utilizadas.....	5
4.	Mapa de Red.....	6
5.	Objetivo de cada Reto	8
6.	Azure	10
7.	Reto 1: Crunchyroll.....	12
7.1	VSFTPD.....	12
➤	Configuración	13
➤	Login Virtual	15
7.2	Apache	18
➤	Virtual host	18
➤	Proteger página con contraseña.....	20
7.3	SSH.....	23
➤	Creación de usuario	23
➤	Configuración SSH	23
8.	Reto 2: Rick & Morty	26
8.1	QR INFO.....	26
8.2	Operación C-137	28
8.3	Billete Aeronáutico.	29
8.4	Rusia	31
8.5	Mega	33
8.6	GIF	35
8.7	User Agent.....	37
9.	Reto 3: Man in the middle	40
9.1	Máquina Ubuntu	40
9.2	Windows 10 (Atacante).....	42
9.3	Windows 7 (Cliente FTP)	45
10.	Reto 4: Criptografía	47
10.1	Sopa de letras	47
10.2	Laberinto.....	49



Redes en Cloud y Ciberseguridad

10.3 Crucigrama	50
11. Reto 5: XSS Attack	51
11.1 Configuración de Debian (Servidor Web)	51
11.2 Configuración de Ubuntu (Atacante)	52
12. Página web	53
13. Diagrama de Gantt	56
14. URL de interés	56
15. Conclusiones	58
16. Bibliografía	59



1. Introducción del proyecto

Los CTF o *Capture The Flag* son “**una serie de desafíos informáticos enfocados a la seguridad**” según el Instituto Nacional de Ciberseguridad INCIBE. Tienen por objetivo aprender y mejorar las competencias de hackers o equipos de hackers. El desafío se plantea como el clásico “captura la bandera”, una competición por equipos en la que se debe llegar a esta en el menor tiempo posible y antes que los adversarios, aunque en este caso las banderas o llaves (*keys*) son logros digitales.

Estos eventos, que pueden durar varios días de trabajo casi ininterrumpido a modo de maratón o competición de fondo, están conformados por una serie de retos de complejidad creciente. Actúan como entrenamientos intensivos, puesta a prueba de habilidades previas, nuevos retos para los participantes o un sistema de aprendizaje colectivo.

Para poder entender mejor los CTFs hemos decidido realizar nuestra propia versión de este juego, con diferentes retos, que se centran en temas como la esteganografía o la criptografía. Además, hemos implementado el servicio de Azure y hemos desarrollado una página web contenedora para la introducción de los retos junto con su respectiva explicación.

La cuenta de estudiantes de Azure sólo permite tener 4 cpu de manera que algunos de los retos estarán en máquinas virtuales de Azure y otros, dentro de la página tendrán un link directo para poder descargar las máquinas y poder realizarlos de manera local con Virtualbox.

2. Justificación Proyecto

Dado que una de las asignaturas que más nos han gustado durante este año ha sido ciberseguridad y como hemos tocado el tema de los servicios en la nube, nos ha parecido una buena idea realizar un captura la bandera, ya que, de esta manera podemos poner en práctica bastantes conocimientos de todo lo que hemos adquirido durante estos dos años, tanto a la hora de crear y configurar servidores, como de montar una página web y la utilización de los servicios en la nube que son ofrecidos por las empresas, además de que este sector está en auge y es un buen método para aprender un poco más cómo funcionan, y cuáles podrían ser sus utilidades, incluso como sería en este caso del pequeño laboratorio que vamos a crear.



3. Tecnologías utilizadas

Microsoft Azure

Microsoft Azure es un servicio de computación en la nube creado por Microsoft para construir, probar, desplegar y administrar aplicaciones y servicios mediante el uso de sus centros de datos. Proporciona software como servicio (SaaS), plataforma como servicio (PaaS) e infraestructura como servicio (IaaS) y es compatible con muchos lenguajes, herramientas y marcos de programación diferentes, incluidos software y sistemas específicos de Microsoft y de terceros.

Apache

Apache es un popular servidor web multiplataforma de fuente abierta que, según los números, es el servidor web más popular que existe. Es activamente mantenido por Apache Software Foundation.

FTP (File Transfer Protocol)

El Protocolo de transferencia de archivos (en inglés File Transfer Protocol o FTP) es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la arquitectura cliente-servidor. Desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos, independientemente del sistema operativo utilizado en cada equipo.

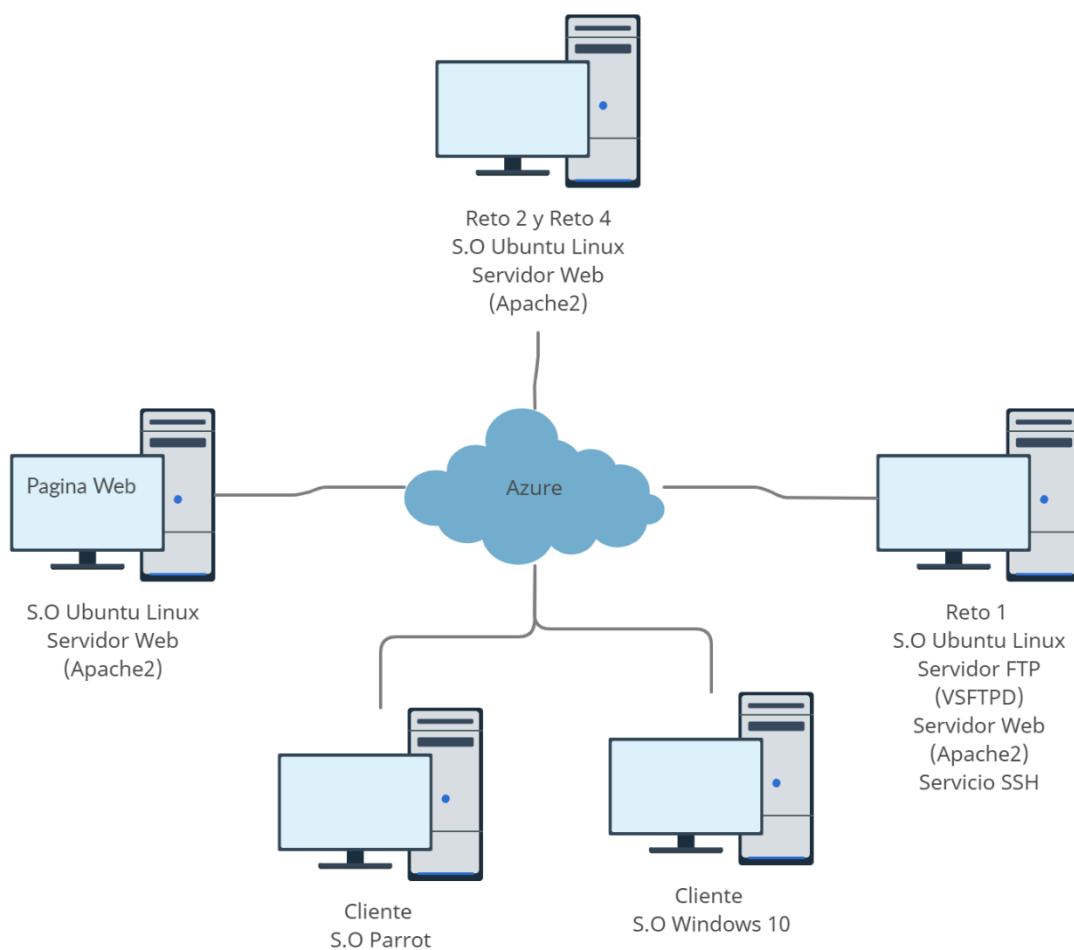
SSH (Secure Shell)

SSH (o Secure SHell) es el nombre de un protocolo y del programa que lo implementa cuya principal función es el acceso remoto a un servidor por medio de un canal seguro en el que toda la información está cifrada. Además de la conexión a otros dispositivos.



4. Mapa de Red

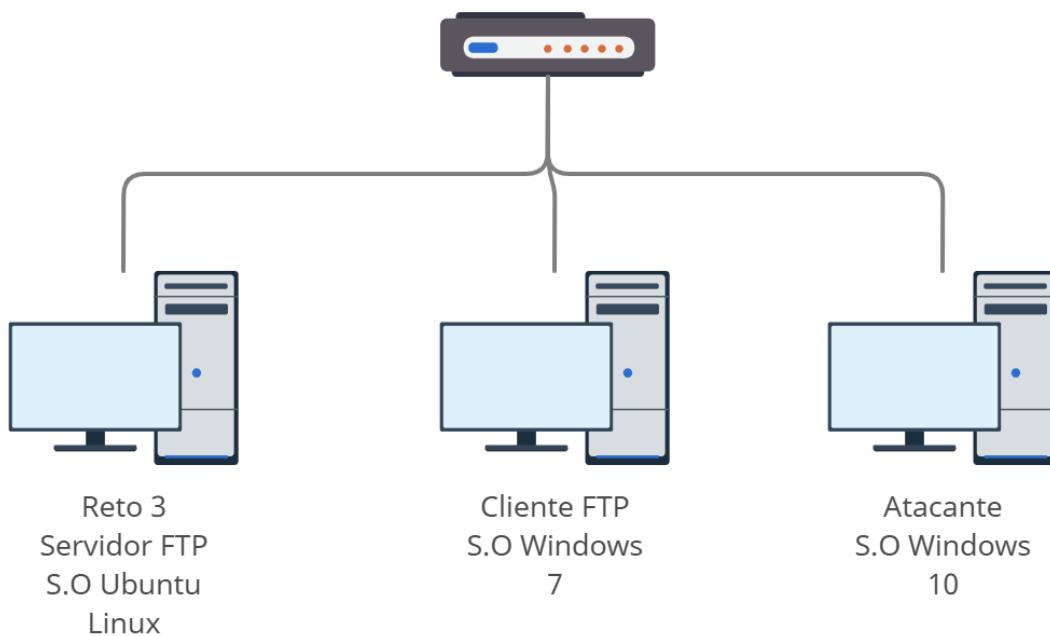
Para desplegar nuestros retos hemos configurado un escenario con múltiples máquinas. Podemos diferenciar tres redes diferentes. El primer diagrama de red representa la estructura montada sobre Azure para la página web, el reto 1, el reto 2 y el reto 4:



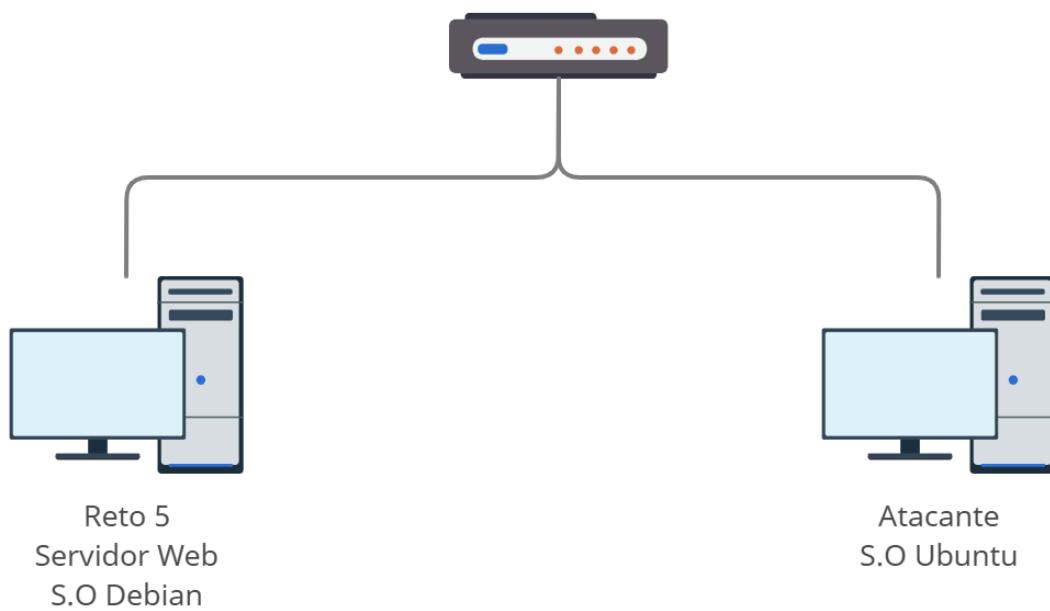


Redes en Cloud y Ciberseguridad

En el siguiente diagrama de red podemos observar la estructura local para el reto 3:



En el último diagrama de red podemos observar la estructura local para el reto 5:





5. Objetivo de cada Reto

Reto 1: Crunchyroll

La principal misión de este reto es conseguir vulnerar la máquina que hay disponible, ya sea por medio de fallos que existen a propósito para realizar el reto, o de programas externos los cuales deberás informarte o conocer sobre ellos para conseguir la última bandera, que se encontrará en el usuario root.

Reto 2: Rick & Morty

Nuestro objetivo en este reto es crear un reto dinámico, con imágenes que puedan llamar la atención, para que las personas que decidan realizarlo, disfruten de alto contenido visual, en este caso, de Rick y Morty. Además, cuenta con una serie de funciones interesantes, como bien puede ser el reenviado automático de correos, los metadatos dentro de un código QR e incluso coordenadas GPS dentro de una imagen. A modo de resumen se podría decir que consiste en realizar una serie de pasos organizados, para alcanzar nuestra preciada bandera. RETO2

Reto 3: Man in the middle

Se te proporcionara una máquina con Windows 10 y algún programa instalado en el escritorio, partiendo de esa base deberá conseguir información para superar el reto. Contarás con escaneadores de redes para ver los equipos de la red y con ayuda de los demás programas instalado tendrás que conseguir cierta contraseña.

Reto 4: Criptografía

El objetivo de este reto es un tanto diferente a los anteriores, puesto que no hemos incluido el uso de tecnologías complejas, ni comandos complicados, simplemente hemos desarrollado una serie de acertijos, con los cuales vamos a hacer un repaso de la historia de la criptografía, puesto que lo consideramos una parte fundamental de la ciberseguridad.



Reto 5: XSS Attack

La principal misión de este reto será crear un pequeño script de manera que se vulnere el formulario que se te proporcionará, gracias a ello podrá crear un php que será alojado en un servidor instantáneo y monitoreado. De manera que al entrar el administrador a su cuenta captaremos la cookie por lo que podremos entrar con sus datos personales.



6. Azure

Las máquinas que vamos a utilizar para los retos están almacenadas en el servicio de Azure. Cuenta con un montón de recursos, e imágenes para crear las máquinas, diferentes sistemas operativos para estas, crear grupos entre ellas, y demás características. Microsoft Azure es una nube pública de pago por uso, por lo que, para realizar el proyecto de forma gratuita hemos solicitado una cuenta de estudiantes con el correo proporcionado por el instituto. Gracias a ello nos han ofrecido cien créditos a utilizar durante un año o hasta que se acaben, además de poner a nuestra disposición su soporte técnico veinticuatro horas.

The screenshot shows the Microsoft Azure portal's home page. At the top, there's a navigation bar with links for 'Crear un recurso', 'Máquinas virtuales', 'Monitor', 'Suscripciones', 'Administración de costos +...', 'Todos los recursos', 'Grupos de recursos', 'App Services', 'Cuentas de almacenamiento...', and 'Más servicios'. Below the navigation bar is a section titled 'Recursos recientes' (Recent resources) which lists several resources with their names, types, and last accessed times:

Nombre	Tipo	Última consulta
Azure para estudiantes	Suscripción	hace 24 minutos
pruebasApache-ip	Dirección IP pública	hace 3 semanas
pruebasApache	Máquina virtual	hace 3 semanas
apacheweb	Máquina virtual	hace 3 semanas
pruebasApache_group	Grupo de recursos	hace 3 semanas
retosPruebasApache_group	Grupo de recursos	hace 3 semanas
CTF	Grupo de recursos	hace 4 semanas

Es tan sencillo como seguir los siguientes pasos:

1. Pulsar en máquinas virtuales.
2. Agregar máquina.
3. Rellenamos los datos pertinentes.
4. Agregamos las reglas de los puertos, si tuviésemos que añadir otras diferente lo podemos hacer después, cuando ya esté creada.

This screenshot shows the 'Create Virtual Machine' wizard in the Azure portal, specifically the 'Configure instance details' step. The form includes the following fields:

- Nombre de máquina virtual ***: Reto1-Alvaro
- Región ***: (Europe) Oeste de Europa
- Opciones de disponibilidad**: No se requiere redundancia de la infraestructura
- Imagen ***: Ubuntu Server 18.04 LTS - Gen1
Examinar todas las imágenes públicas y privadas
- Instancia de Azure de acceso puntual**: (unchecked)
- Tamaño ***: Standard_B1ls - 1 vcpu, 0.5 GiB de memoria (3.69 €/mes)
Seleccionar tamaño
- Cuenta de administrador**
 - Tipo de autenticación**: Clave pública SSH (radio button)
 - Contraseña**: (radio button selected)
- Nombre de usuario ***: grupo1azure
- Contraseña ***: (redacted)
- Confirmar contraseña ***: (redacted)
- Reglas de puerto de entrada**: (empty field)



Redes en Cloud y Ciberseguridad

Reglas de puerto de entrada

Seleccione los puertos de red de máquina virtual que son accesibles desde la red Internet pública. Puede especificar acceso de red más limitado o granular en la pestaña Red.

Puertos de entrada públicos * ⓘ

- Ninguno
 Permitir los puertos seleccionados

Seleccionar puertos de entrada *

HTTP (80), SSH (22)

⚠ Esto permitirá que todas las direcciones IP accedan a la máquina virtual.
Esto solo se recomienda para las pruebas. Use los controles avanzados de la pestaña Redes a fin de crear reglas para limitar el tráfico entrante a las direcciones IP conocidas.

Una vez finalicemos esto podemos dar a crear máquina y conectarnos por ejemplo a través del cmd de Windows u otros programas como Putty.

```
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
grupolazure@Retol-Alvaro:~$
```

Ya estaría todo listo para preparar la máquina para el reto. (Todas las maquinas creadas en Azure de la misma manera, a excepción de los puertos abiertos, que dependen de las necesidades de cada una).



7. Reto 1: Crunchyroll

Preparación de archivos

En la primera parte vamos a utilizar un documento txt con un listado de contraseñas que vamos a crear de forma automática y una foto de un personaje de la cual, necesitaremos saber el nombre para poder continuar.

- **Imagen**
- **Listado**

En este paso haremos uso de un programa que nos lo crea de forma automática y poniendo los parámetros que nosotros le indiquemos.

Como en Ubuntu este programa no viene por defecto lo instalamos con el comando.

```
sudo apt-get install crunch
```

Una vez instalado podemos proceder a crear el archivo, la sintaxis será la siguiente.

```
crunch mínimo máximo caracteres -o  
ruta_de_salida_del_archivo/nombre_archivo.txt
```

Todo en la misma línea

```
vegeta@cliente-VirtualBox:~$ crunch 5 5 -o /home/vegeta/Escritorio/listado.txt
Crunch will now generate the following amount of data: 192 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 32
crunch: 100% completed generating output
vegeta@cliente-VirtualBox:~$
```

7.1 VSFTPD

Primero deberemos instalar el servidor ftp con el siguiente comando:

```
sudo apt-get install vsftpd
```

Y comprobamos que el servicio está activo:

```
sudo service vsftpd status
```



➤ Configuración

Antes de todo hacemos una copia de seguridad del fichero de configuración por si nos surgiese algún problema y después de configurarlo no arrancase y no encontrásemos el fallo.

```
sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.bak
```

Las configuraciones que vamos a hacer son las siguientes.

Listen=YES

Permite la conexión de Ipv4.

Listen_ipv6=NO

Permite la conexión de Ipv6.

Anonymous_enabled=YES

Permite las conexiones anónimas.

Local_enable=YES

Permitir conexiones de usuarios locales, aunque sean virtuales debe estar en YES ya que si no, solo se permiten las conexiones anonymous.

Write_enable=NO

No se permite la subida de archivos al servidor.

Anon_root=/srv/ftp/servidorVSFTP

Carpeta donde están los archivos anonymous por defecto, a gusto de cada cual está el cambiarla por otro directorio.

Chroot_local_user=NO

No enjaular a los usuarios locales.

Chroot_list_enable=YES

Activar lista de usuarios encerrados.

Chroot_list_file=/etc/vsftpd.chroot_list

Lista de usuario que estarán encerados en su home. Tendremos que crear el fichero vsftpd.chroot_list con el nombre de los usuarios.



Allow writeable chroot=YES

Permite solucionar un error en el vsftpd ya que si no, no funcionará

Pasv enable=YES

Permite las conexiones en modo pasivo.

Pasv min port=10000

Puerto mínimo para abrir.

Pasv max port=10010

Puerto máximo para abrir.

Seccomp sandbox=NO

Corrige un bug en el modo pasivo para permitir la conexión.

Pasv address=ip pública

Permite pasar al modo pasivo las conexiones.

Una vez tengamos toda la configuración del vsftpd tenemos que permitir la conexión a través de los puertos correspondientes en Azure, siendo estos 20, 21,10000-10010.

The screenshot shows the configuration of an NSG rule named 'FTP' for the subnet 'Reto1-Alvaro-nsg'. The rule is set to 'Allow' traffic from 'Any' source port range '20,21,10000-10010' to 'Any' destination port. The protocol is set to 'TCP'. The action is 'Permitir' (Allow). The priority is set to 310. There are buttons for 'Guardar' (Save), 'Descartar' (Discard), 'Básica' (Basic), and 'Eliminar' (Delete).

Propiedad	Valor
Origen *	Any
Intervalos de puertos de origen *	*
Destino *	Any
Intervalos de puertos de destino *	20,21,10000-10010
Protocolo *	TCP
Acción *	Permitir
Prioridad *	310



➤ Login Virtual

Vamos a pasar a la creación de un login virtual, lo primero será instalar db-util.

```
sudo apt-get install db-util
```

Tenemos que crear una tabla con los logins y password de los usuarios. Para ello vamos a crear un archivo y ponerlo dentro de un directorio que llamaremos vsftpd de esta manera:

```
mkdir /etc/vsftpd  
cd /etc/vsftpd  
touch logins.txt
```

Establecemos los permisos: **chmod 600 logins.txt**

Dentro del archivo pondremos el nombre de cada usuario y su contraseña, uno a uno. En la primera línea el nombre y en la que sigue la contraseña:

```
GNU nano 2.9.3          logins.txt  
todoroki  
11100
```

A continuación vamos a crear la tabla de la base de datos que utilizará PAM para realizar la autenticación. Lo haremos dentro de la carpeta vsftpd creada previamente:

```
cd /etc/vsftpd  
db_load -T -t hash -f logins.txt vsftpd_login.db  
chmod 600 vsftpd_login.db
```

El siguiente paso consiste en modificar el archivo de configuración PAM del servicio vsftpd, este archivo lo podemos encontrar en el directorio **/etc/pam.d/** y debería llevar el mismo nombre especificado en la directiva **"pam_service_name"**, en este caso **"vsftpd"**. Antes de editar el archivo, hacemos una copia de seguridad del mismo y nos aseguramos que el servicio ftp esté parado. Verificado esto, editamos el archivo



comentando todas sus líneas anteponiendo el carácter # y le agregamos las siguientes:

```
sudo cp vsftpd vsftpd.vak
```

Ahora podemos editar el archivo sin miedo.

El archivo debería de quedar de esta manera.

```
auth required pam_userdb.so db=/etc/vsftpd/vsftpd_login  
account required pam_userdb.so db=/etc/vsftpd/vsftpd_login
```

```
vegeta@cliente-virtualbox: /etc/pam.d  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.9.3 vsftpd  
# Standard behaviour for ftpd(8).  
#auth    required      pam_listfile.so item=user sense=deny file=/etc/ftpusers onerr=succeed  
# Note: vsftpd handles anonymous logins on its own. Do not enable pam_ftp.so.  
# Standard pam includes  
#@include common-account  
#@include common-session  
#@include common-auth  
#auth    required      pam_shells.so  
auth required pam_userdb.so db=/etc/vsftpd/vsftpd_login  
account required pam_userdb.so db=/etc/vsftpd/vsftpd_login
```

Estas líneas permiten que los usuarios que se conecten al servicio ftp sean validados contra los registros que contiene la tabla vsftpd_login.db que hemos creado.

Nuevamente editamos el archivo de configuración /etc/vsftpd.conf y agregamos las siguientes líneas al final del archivo:

```
# Activamos el uso de usuarios virtuales.  
dirlist_enable=YES  
guest_enable=YES  
# Esta directiva permite especificar el usuario que se encargará de manejar los  
# por defecto si no se especifica esta línea en Ubuntu es "ftp" que como sabemos  
# /srv/ftp/, pero puede ser cualquier otro usuario si lo deseamos.  
guest_username=ftp  
# Especificamos el directorio de donde el servicio obtendrá la configuración para  
# usuario virtual que agregamos a la tabla "vsftpd_login.db".  
user_config_dir=/etc/vsftpd/user_config
```

Guardamos los cambios del archivo vsftpd.conf y creamos el directorio indicado en la directiva “user_config” dentro de la carpeta /etc/vsftpd/:

```
mkdir user_config
```



En él creamos por cada usuario virtual que agregamos a la tabla “vsftpd_login.db” un archivo de texto plano con el mismo nombre del login del usuario.

```
cd user_config
```

```
touch todoroki
```

Por ejemplo editamos el archivo “todoroki” recientemente creado y le agregamos las siguientes líneas:

```
GNU nano 2.9.3 todoroki
#!/bin/bash
# Indicamos cual será el directorio personal del usuario gabriel
dirlist_enable=YES
local_root=/srv/ftp/todoroki
# Le damos permisos de escritura para su directorio personal.
write_enable=YES
# Con virtual_use_local_privs igualado a YES, indicamos que los usuarios
# tienen los mismos privilegios que los usuarios locales.
virtual_use_local_privs=YES
```

Ahora solo nos resta crear los directorios personales para ambos usuarios virtuales en el directorio /srv/ftp/, cambiarlos de dueño y asignar los permisos correspondientes:

```
cd /srv/ftp
```

```
mkdir todoroki
```

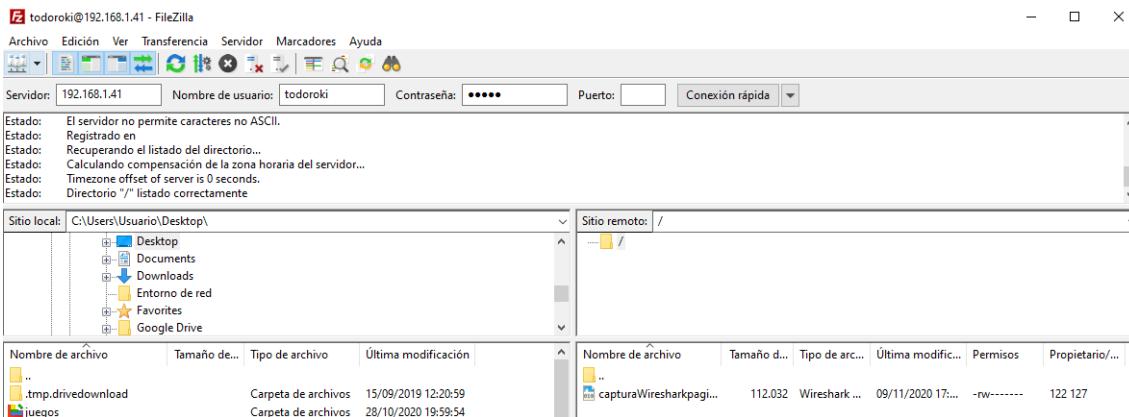
```
chown -R ftp:ftp todoroki
```

```
chmod -R 744 todoroki
```

```
vegeta@cliente-VirtualBox:/srv/ftp$ ll
total 16
drwxr-xr-x 4 root    ftp     4096 nov  7 17:59  .
drwxr-xr-x 3 root    root    4096 nov  7 12:12  ..
drwxrwxr-x 2 vegeta  vegeta  4096 nov  7 12:07 'servidor VSFTP'/
drwxr--r-- 2 ftp     ftp     4096 nov  7 17:59  todoroki/
vegeta@cliente-VirtualBox:/srv/ftp$
```



Una vez hecho esto lo reiniciamos y vemos que funciona de forma correcta.



7.2 Apache

Primero de todo necesitamos instalar el servidor apache con el comando:

```
sudo apt-get install apache2
```

```
sudo service apache2 status
```

➤ Virtual host

Para la creación de un host virtual primero tenemos que crear la carpeta donde se va a alojar nuestra página al completo, en nuestro caso la crearemos en dentro de los directorio **/var/www**.

```
sudo mkdir bandera.com
```

Si queremos que nuestro usuario regular sea capaz de modificar archivos dentro de nuestros directorios web, debemos cambiar la propiedad haciendo lo siguiente:

```
sudo chown -R $USER:$USER /var/www/bandera.com
```

Debemos además, modificar un poco nuestros permisos para asegurarnos de que el acceso de lectura esté habilitado en el directorio web general y todos los archivos y directorios en él para que todas las páginas puedan ser servidas correctamente:

```
sudo chmod -R 755 /var/www
```



Vamos a ir con una demostración, así que nuestras páginas serán muy simples. Vamos a crear una página.

```
sudo nano index.html
```

De momento con hacer esto es suficiente ya que solo lo estamos probando.

```
vegeta@cliente-VirtualBox: /var/www/bandera.com
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3           index.html           Modificado
<html>
  <head>
    <title>¡Bienvenido a Bandera.com!</title>
  </head>
  <body>
    <h1>¡Lo lograste! El virtual host bandera.com está funcionando</h1>
  </body>
</html>
```

Una vez hecho esto nos dirigimos al siguiente directorio **/etc/apache2/sites-available**

Y copiamos la configuración de **000-default.conf** a un fichero que nombremos nosotros, para nuestro caso será **bandera.conf**

```
cp 000-default.conf bandera.conf
```

Ahora necesitará modificar todas las piezas de información para hacer referencia al segundo dominio. Cuando concluya, se verá algo así:

```
vegeta@cliente-VirtualBox: /etc/apache2/sites-available
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3           bandera.conf
<VirtualHost *:80>

  ServerAdmin vegeta@gmail.com
  DocumentRoot /var/www/bandera.com

  ErrorLog ${APACHE_LOG_DIR}/error.log
  CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>
```

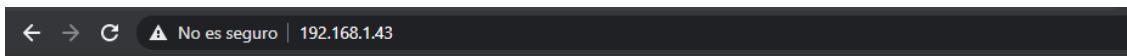


Guardamos y cerramos el archivo, por último nos queda habilitar la página web para que sea accesible.

```
sudo a2ensite bandera.conf
```

Por ultimo nos pedirá que reiniciemos el servicio.

```
sudo systemctl restart apache2
```



¡Lo lograste! El virtual host **bandera.com** está funcionando

➤ Proteger página con contraseña

Para proteger la página web tenemos que crear un archivo **.htaccess** para ello primero tenemos que asegurarnos que en la carpeta de **mods-enabled**, el módulo **auth_basic.load** está activado. Si se diese el caso de que no lo tenemos activado con este par de comandos se debería de activar.

```
sudo ln -s /etc/apache2/mods-available/modulo /etc/apache2/mods-enabled/modulo
```

```
sudo service apache2 restart
```

```
vegeta@cliente-VirtualBox:/etc/apache2/mods-enabled$ ll
total 8
drwxr-xr-x 2 root root 4096 nov  4 16:40 .
drwxr-xr-x 8 root root 4096 nov  9 13:32 ../
lrwxrwxrwx 1 root root   36 nov  4 16:40 access_compat.load -> ../mods-available/access_compat.load
lrwxrwxrwx 1 root root   28 nov  4 16:40 alias.conf -> ../mods-available/alias.conf
lrwxrwxrwx 1 root root   28 nov  4 16:40 alias.load -> ../mods-available/alias.load
lrwxrwxrwx 1 root root   33 nov  4 16:40 auth_basic.load -> ../mods-available/auth_basic.load
lrwxrwxrwx 1 root root   33 nov  4 16:40 authn_core.load -> ../mods-available/authn_core.load
lrwxrwxrwx 1 root root   33 nov  4 16:40 authn_file.load -> ../mods-available/authn_file.load
lrwxrwxrwx 1 root root   33 nov  4 16:40 authz_core.load -> ../mods-available/authz_core.load
lrwxrwxrwx 1 root root   27 nov  4 16:40 authz_host.load -> ../mods-available/authz_host.load
```

Para crear el archivo que contendrá las contraseñas de los usuarios pondremos en un terminal, el siguiente comando y escribimos la contraseña deseada.

```
sudo htpasswd -c .htpasswd usuario
```

Ahora nos desplazamos hasta la configuración de nuestro sitio web. Y lo configuraremos de esta manera para que al entrar en la página nos pida una contraseña.

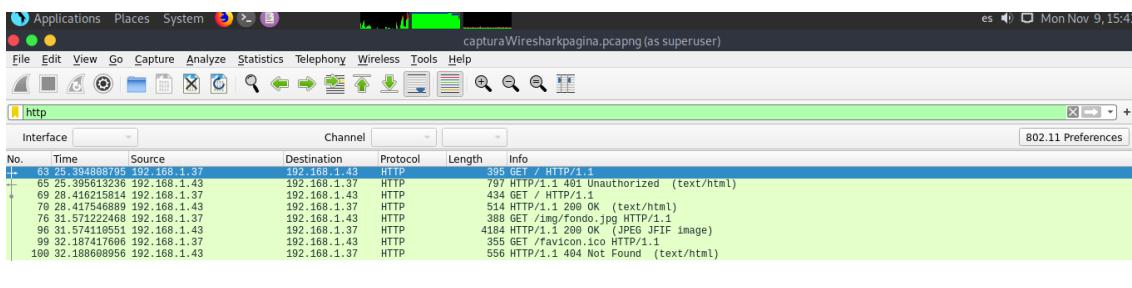
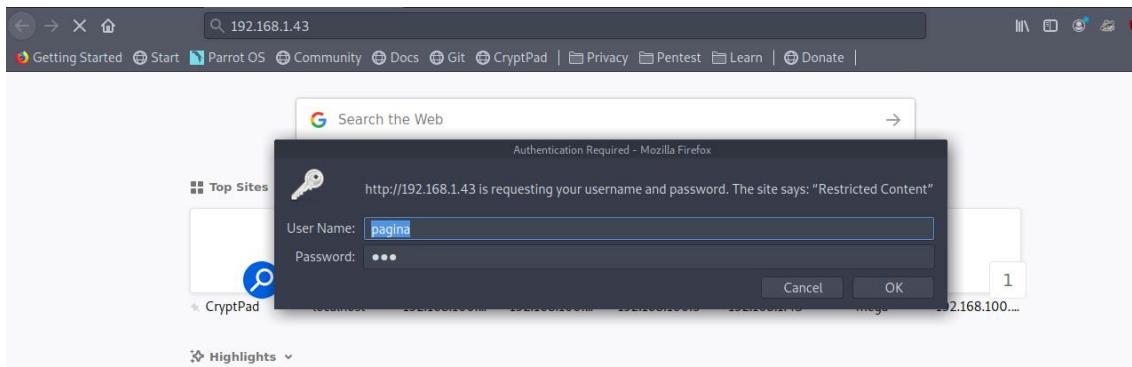


Redes en Cloud y Ciberseguridad

```
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.9.3  
bandera.conf  
  
<VirtualHost *:80>  
  
    ServerAdmin vegeta@gmail.com  
    DocumentRoot /var/www/bandera.com  
  
    ErrorLog ${APACHE_LOG_DIR}/error.log  
    CustomLog ${APACHE_LOG_DIR}/access.log combined  
  
    <Directory "/var/www/bandera.com">  
  
        AuthType Basic  
        AuthName "Restricted Content"  
        AuthUserFile /etc/apache2/.htpasswd  
        Require valid-user  
  
    </Directory>  
</VirtualHost>  
T 20 líneas leídas 1
```

Vamos a crear el archivo para el **wireshark** que debemos de meter en el ftp con el usuario virtual que creamos para poder acceder a la página.

Nos metemos en la página para que nos pida la contraseña y mientras capturamos el tráfico con el programa. El tráfico que nos interesa es http ya que es el que se usa con apache por defecto a menos que lo configuremos para https.



Una vez hecho esto lo subimos al servidor ftp para poder realizar la búsqueda más tarde.

```
curl -u todoroki:11100 -T /home/user/Desktop/capturaWiresharkpagina.pcapng  
ftp://192.168.1.43
```



```
[user@parrot-virtual]~$ ftp 192.168.1.43
Connected to 192.168.1.43.
220 (vsFTPd 3.0.3)
Name (192.168.1.43:user): todoroki
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw----- 1 122 127 112032 Nov 09 16:52 capturaWiresharkpagina.pcapng
226 Directory send OK.
ftp>
```

Por último vamos a montar todo el apache de la siguiente manera, juntando diferentes tipos de archivos .txt, html, php, y sus correspondientes directorios.

directorio principal bandera.com

En estos archivos iremos metiendo diferentes tipos de pistas ya sean listados para usar con algún programa direcciones a otras carpetas, datos codificados en base64, y datos.

El apartado de la imagen que incluye escenografía está realizada mediante el programa stegosuite, tan solo tenemos que abrir la foto con el programa e insertar el mensaje que queramos ocultar, en nuestro caso pondremos los datos del usuario de ssh.

```
vegeta@cliente-VirtualBox:/var/www/bandera.com$ tree
.
├── admin
│   └── admin.php
├── b374k.php
├── find_me
│   └── find_me.html
├── haru
│   └── haru_embed.jpg
│       └── mensaje.txt
├── image
└── img
    └── fondo.jpg
├── index.html
└── listado
    └── gobuster.txt
└── manual
    └── robots.txt
```



7.3 SSH

➤ Creación de usuario

Para la creación del usuario solo será necesario utilizar el comando.

```
sudo adduser tsuna
```

➤ Configuración SSH

Pasamos al último apartado del reto, constara de una conexión de ssh en la cual tendremos que escalar privilegio, lo primero que tenemos que hacer es instalar el servicio de SSH con el comando:

```
sudo apt install openssh-server
```

Una vez instalado tenemos que hacer algunos cambios de configuración importantes sobre todo de cara a la persona que resolverá el reto para que no haga “trampas”. Nos dirigimos al archivo del directorio **/etc/sudoers**. En el si hubiese algún usuario con permisos de root eliminamos esa línea o la comentamos, de esta manera no podrá utilizar ningún comando de root.

```
# Cmnd alias specification  
  
# User privilege specification  
root    ALL=(ALL:ALL) ALL
```

```
tsuna@cliente-VirtualBox:/home/vegeta$ sudo apt-get update  
[sudo] contraseña para tsuna:  
tsuna no está en el archivo sudoers. Se informará de este incidente.  
tsuna@cliente-VirtualBox:/home/vegeta$
```

Por ultimo también debemos cambiar el propietario del archivo de **/etc/passwd** para que sea posible resolver el reto. Esto lo haremos con el siguiente comando:

```
chown tsuna:root /etc/passwd
```

Los permisos estarán bien así.

```
-rw-rw-r--  1 root  root      3 mar  7 2020 papersize  
-rw-r--r--  1 tsuna root    2719 dic 24 19:14 passwd
```



También configuraremos el archivo **sshd_config** para añadir un plus de seguridad:

LoginGraceTime

Estableceremos el tiempo necesario para introducir la contraseña, evitando que el atacante tenga que «pensar mucho».

MaxAuthTries

Número de intentos permitidos al introducir la contraseña antes de desconectarnos.

MaxStartups

Número de logins simultáneos desde una IP, para evitar que se pueda utilizar la fuerza bruta con varias sesiones a la vez.

2: Número de conexiones no autenticadas antes de que empiezamos a eliminar.

20: Porcentaje de probabilidad de caer una vez que llegamos a 10 (aumenta linealmente durante más de 10).

100: Número máximo de conexiones en las que empezamos a descartar todo.

PasswordAuthentication YES

Si queremos habilitar el login en el servicio a través del usuario y contraseña del sistema. Cuando tengamos el fichero configurado lo guardamos y cerramos y reiniciamos el servicio con los comandos:

```
systemctl stop sshd
```

```
systemctl start sshd
```

Por último hacemos un dibujo en Ascii con ayuda de una página web para colocar como bandera en la carpeta root:

<https://manytools.org/hacker-tools/convert-images-to-ascii-art/> (*1)

Una vez lo tengamos creado lo subimos a la máquina con el comando:

```
scp dirección-archivo usuario@ip:maquina:/directorio_donde_queremos_moverlo
```



Redes en Cloud y Ciberseguridad

```
root@Retol-Alvaro:~# cat root.txt
```



8. Reto 2: Rick & Morty

8.1 QR INFO

Para realizar este reto hemos necesitado una máquina virtual en Azure, en la cual hemos instalado un servidor web (apache2).

Para realizar el primer paso, hemos recurrido a una página online, que nos permite generar un código QR y asignarle texto plano.

<https://www.the-qrcode-generator.com/> (*2)



Aquí empieza tu primera prueba, no tenemos mucho más que decir, suerte en tu aventura, Rick.

Lo siguiente que necesitamos es un decoder QR para poder analizar los metadatos correspondientes, por lo que recurrimos a <https://zxing.org/w/decode.jspx> (*3)

Seleccionamos nuestro código QR, enviamos el archivo y podremos ver los metadatos:



ZXing Decoder Online

Decode a 1D or 2D barcode from an image on the web. Supported formats include:

- UPC-A and UPC-E
- EAN-8 and EAN-13
- Code 39
- Code 93
- Code 128
- ITF
- Codabar
- RSS-14 (all variants)
- RSS Expanded (most variants)
- QR Code
- Data Matrix
- Aztec
- PDF 417
- MaxiCode

Enter an image URL:	<input type="text"/>	<input type="button" value="Enviar"/>
Or upload a file (<10MB, <10MP):	<input type="file" value="Elegir archivo qr_rick_1.png"/>	<input type="button" value="Enviar"/>

Decode Succeeded

Raw text	Bienvenido Rick. Hemos detectado que Evil Morty ha estado consultando una web, para llevar a cabo su plan. http://52.137.32.181/OperacionC_137/ El código de acceso a la web es el siguiente: 1a3C1A2b\$ Mucho suerte en tu misión Rick, estamos contigo.
Raw bytes	40 07 24 26 96 56 e7 66 56 e6 96 46 f2 05 26 96 36 b2 e0 a0 a4 86 56 d6 f7 32 06 46 57 46 56 37 46 16 46 f2 07 17 56 52 04 57 66 96 c2 04 d6 f7 27 47 92 06 86 12 06 57 37 46 16 46 f2 06 36 f6 e7 37 56 c7 46 16 e6 46 f2 07 56 e6 12 07 76 56 22 c2 07 06 17 26 12 06 c6 c6 57 66 17 22 06 12 06 36 16 26 f2 07 37 52 07 06 c6 16 e2 e2 00 a0 a6 87 47 47 02 02 5f 9f ca 08 40 60 59 44 4f 63 2d 3e 9d 00 1f 5c 19 5c 98 58 da 5b db 90 d7 cc 4c cd cb c2 82 91 5b 08 18 f0 ec d9 1a 59 db c8 19 19 48 18 58 d8 d9 5c db c8 18 48 1b 18 48 1d d9 58 88 19 5c c8 19 5b 08 1c da 59 dd 5a 59 5b 9d 19 4e 82 82 8c 58 4c d0 cc 50 4c 98 89 02 82 93 5d 58 da 18 48 1c dd 59 5c 9d 19 48 19 5b 88 1d 1d 48 1b 5a 5c da 70 ec db 88 14 9a 58 da cb 08 19 5c dd 18 5b 5b dc c8 18 db db 9d 1a 59 db cb 88 00 ec 11 ec 11 ec 11 ec 11 ec 11 ec 11 ec

A nivel programación, en esta página hemos implementado la imagen con el atributo `` y el texto con `<p>`.



8.2 Operación C-137

En este paso, simplemente tenemos un login, con la clave que nos facilitan en el paso 1.

Clave de Acceso: Entrar

A nivel de programación web, en este paso tenemos lo siguiente:

La función principal de nuestro código, en este paso, es el login, como bien hemos comentado anteriormente:

```
<input type="text" id="texto">
<FORM ACTION="/OperacionC_137/billete.html" METHOD=GET>
<button onclick="clave()">Validar</button>

<script>
    function clave(){
        if(document.getElementById("texto").value == "1a3C1A2b$")
            {
                document.form.submit();
            }
        else
            {
                document.write('Clave Incorrecta, por favor, actualice la página para continuar.');
            }
    }
</script>
```

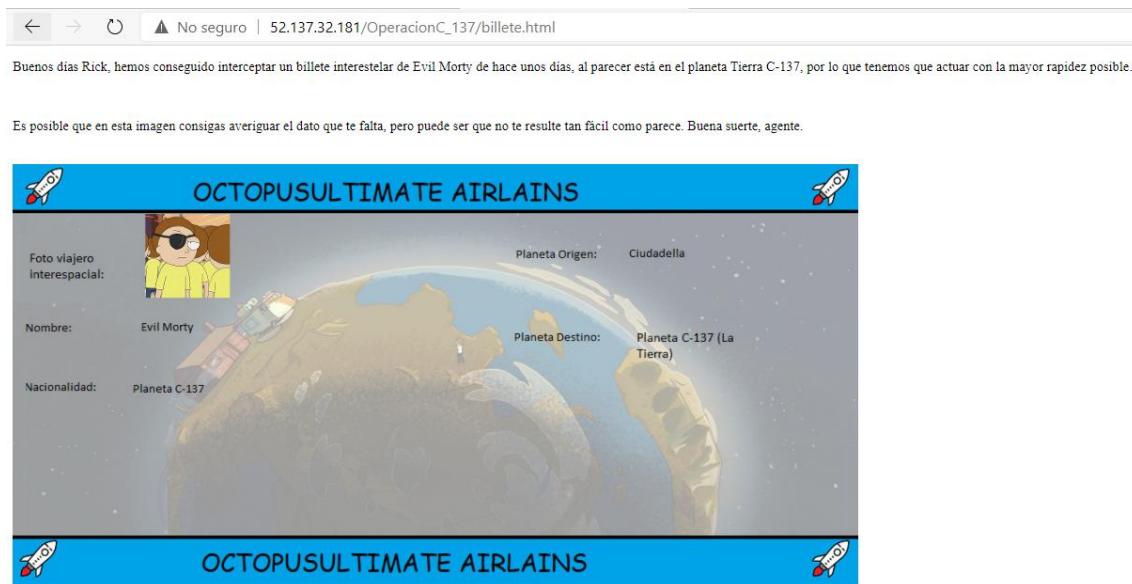
Con este script, deshabilitamos la posibilidad de ver el código fuente (click derecho), para evitar que nuestras contraseñas puedan ser vulneradas, con total facilidad.

```
<script type="text/javascript">
document.oncontextmenu = function(){return false;}
</script>
```



8.3 Billete Aeronáutico.

En este paso tenemos un billete aeroespacial, de Evil Rick, en el cual, podemos ver Planeta Origen, Nacionalidad etc...



El billete ha sido realizado con Paint, puesto que nos proporcionaba las herramientas necesarias para desarrollar el proyecto de forma adecuada; adjunto las imágenes originales:





Redes en Cloud y Ciberseguridad

La información que nos facilita el billete, no nos sirve de nada por lo que vamos a mirar en sus metadatos, en este caso, para analizar la imagen, hemos utilizado este recurso <https://metashieldclean-up.elevenpaths.com/> (4*)

The screenshot shows the Metashield Clean-up Online interface. On the left, there are three navigation links: 'Análisis' (Analysis), 'Limpieza' (Cleaning), and 'Contacto' (Contact). The main area has a title 'Analiza tus ficheros con Metashield Clean-up Online.' and a sub-instruction: 'Para analizar los metadatos de un fichero, selecciona abajo el fichero y pulsa el botón "Analizar". Tras aceptar los Términos y Condiciones de Uso para el servicio Metashield Clean-up Online se mostrará en pantalla un resumen de los metadatos encontrados.' Below this is a file selection input field containing 'billete_rick.jpg', a 'Seleccionar' button, and an 'Analizar' button. A large tree view displays the metadata found in the file, including Country, GeodeticSurveyData, GpsTimestamp, GpsLatitudeReference, GpsLongitudeReference, HorizontalResolution, Latitude, Longitude, Orientation, and Province. Under Latitude and Longitude, specific values are listed: 61°31'26.46001 N, 105°19'7.50000 E.

Además, para incluir la Latitud y la Longitud que nosotros hemos querido, hemos utilizado este software <https://geosetter.de/en/download-en/> (*5)

The screenshot shows the GeoSetter software interface. At the top, there's a menu bar with Archivo, Editor, Búsqueda & Filtro, Mapa, Ver, Ayuda. The main window shows a preview of three images: 'rick_gif', 'ultimo_paso', and 'billete_rick.jpg'. The 'billete_rick.jpg' preview shows coordinates N61°31'26.46001 E105°19'7.50000 and location information: Ayan, Russia, Krasnojarsk. Below the preview is a status bar: '2 Imágenes (1 con Geodatos) - 1 seleccionada(s) (1 con Geodatos)'. To the right is an 'Info Imagen (ExifTool)' panel with a table of metadata. The table includes columns 'Nombre' and 'Valor' and lists various fields like Ratio Submuestreo de..., Tamaño de la Imagen, Megapixels, Location (with sub-fields like Versión Etiqueta GPS, Latitud Norte o Sur, Longitud Este u Oeste, Longitud, Dato Medición Geodésica, Código País, Localización, País, Estado, GPS Position, Sub-location, Province-State, Country-Primary Locat..., Country-Secondary Locat...), and Other (with sub-fields like Latitud Norte o Sur, Longitud Este u Oeste, Longitud, Dato Medición Geodésica, Código País, Localización, País, Estado, GPS Position, Sub-location, Province-State, Country-Primary Locat..., Country-Secondary Locat...). The table shows values such as YCbCr4:4:4 (1:1), 1000x500, 0.500, 2.2.0.0, 61 deg 31' 26.46", 105 deg 19' 7.50", WGS-84, RUS, Ayan, 61 deg 31' 26.46" N, 105 deg 19' 7.50" E, Krasnojarsk, etc. At the bottom of the panel are buttons for Mapa, Info Imagen (ExifTool), Process Queue, and Map Log.



Redes en Cloud y Ciberseguridad

Con la Latitud y Longitud, nos dirigimos a Google Maps para introducir las coordenadas, por lo que nuestro siguiente paso será /Rusia.

Latitud, Longitud y dirección de cualquier punto en el mapa

Haz clic en el mapa para obtener su dirección y coordenadas GPS directamente. La latitud y longitud se muestran en la columna de la izquierda y en el mapa.

Dirección

Obtener Coordenadas GPS

GD (grados decimales)*

Latitud

Longitud

Obtener Dirección

GMS (grados, minutos, segundos)*

Latitud N S ° ' "

Longitud E O ° ' "

Obtener Dirección

* Sistema Geodésico Mundial 1984 (WGS 84)

8.4 Rusia

En Rusia, nos pide una clave de acceso, la cual, realmente sólo podremos obtener, enviando un correo a la dirección que representa al administrador del Área Restringida, es decir, el administrador decidirá si eres apto o no, para acceder.

;Bienvenido a Rusia, Rick!
AREA RESTRINGIDA
Para acceder al contenido de Rusia, debes conseguir la clave de acceso
Quizás te sirva de ayuda contactar con el administrador del portal: grupo1azure@gmail.com
Clave de Acceso:

Quizás te sirva de ayuda contactar con el administrador del portal: grupo1azure@gmail.com



Para este paso, hemos configurado el reenviador de correos automáticos de Google, dado que una vez enviado el correo, te responderá de forma automática con la clave de acceso.

Respuesta automática:

(Envía una respuesta automática a los mensajes entrantes. Si un contacto te envía varios mensajes, esta respuesta se enviará una vez cada 4 días como máximo.)

[Más información](#)

Respuesta automática desactivada
 Respuesta automática activada

Primer día: Último día: (opcional)

Asunto:

Mensaje:

Sans Serif

« Texto

Hola de nuevo Rick.
Lo primero de todo, ¡enhorabuena por tus progresos en la búsqueda de Evil Morty!
Hemos conseguido interceptar una serie de datos de Evil Morty a su paso por Rusia:
Clave de acceso: 134@42x3@#
¡Buena Suerte!

Enviar respuesta solo a mis contactos

Además, lo más destacable a nivel programación web, es el uso de código CSS para implementar un espacio de texto, en un lugar determinado:

```
<style>
.cuadrado{
padding:15px;
opacity: 0.85;
margin:10px;
background-color: #F8F8FF;
border: solid 1px purple;
color: black;
text-align:center;display:block;
}
</style>
```



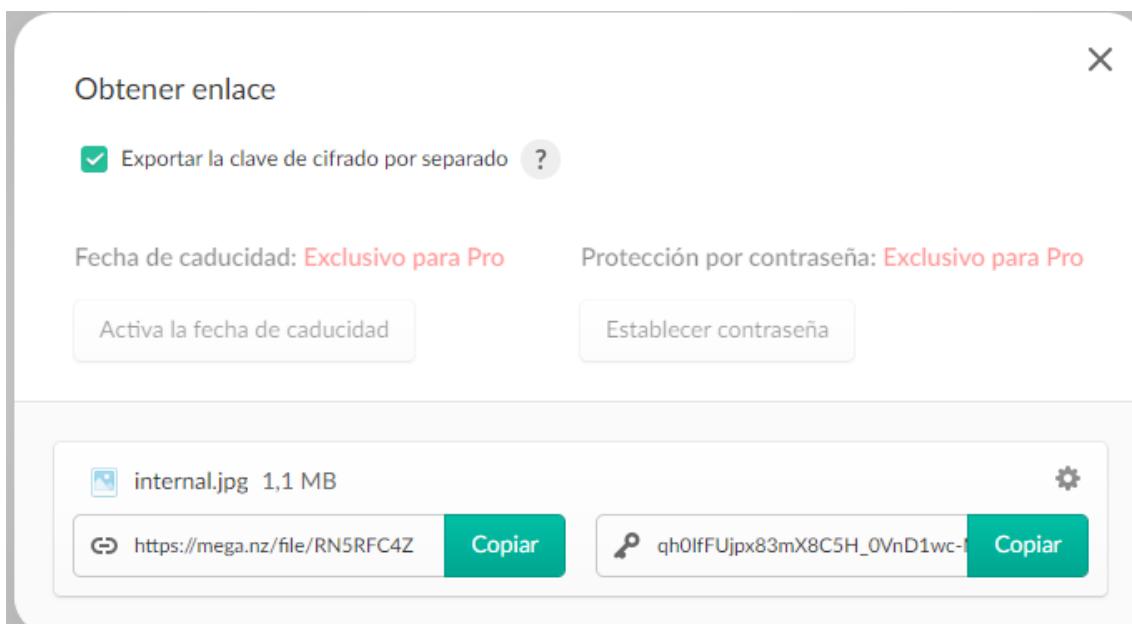
8.5 Mega

zOlbtEnbMZnYY2spQ6Kius3ZU_V-WwUuAbkL_Q22WAW

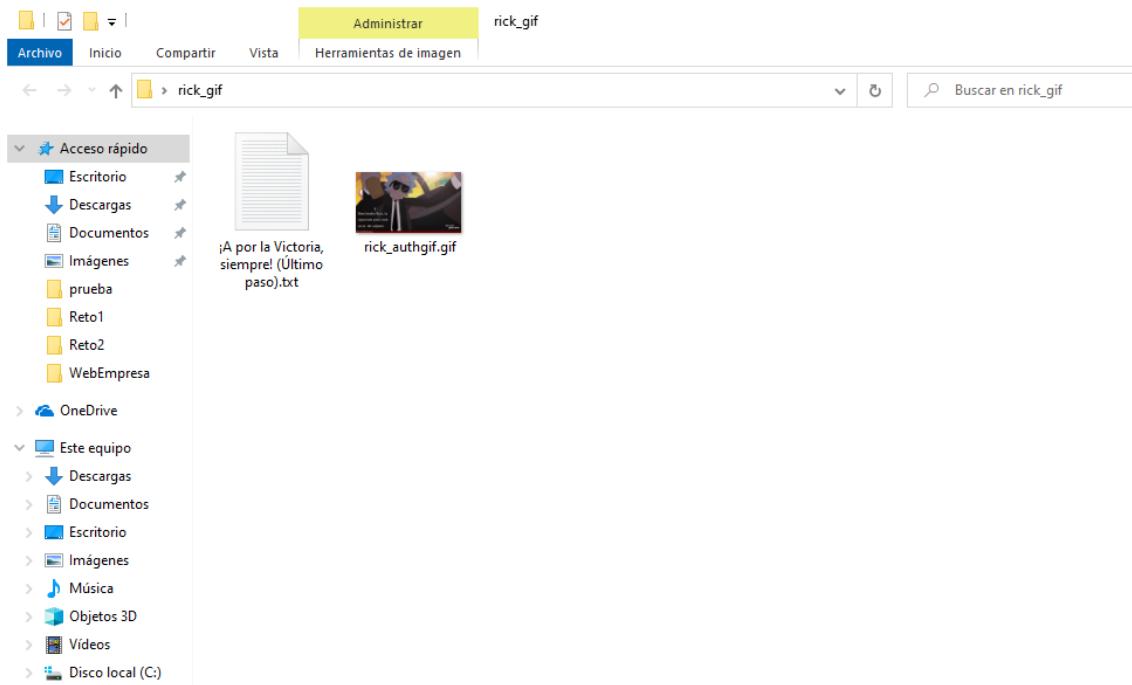
Para completar este paso, hemos creado una cuenta en Mega, en la cual hemos subido nuestro archivo. Además, hemos cifrado mediante una clave el documento, por lo que, para descargarlo, necesitas añadir la clave de cifrado que se muestra en la anterior imagen.



Redes en Cloud y Ciberseguridad



Dentro del .zip que nos hemos descargado, tendremos un documento .txt y un archivo .gif.





8.6 GIF

A priori, crear un gif puede resultar una tarea sencilla, pero en nuestro caso además hemos querido personalizarlo, puesto que hemos insertado la clave de este paso, dentro del gif.

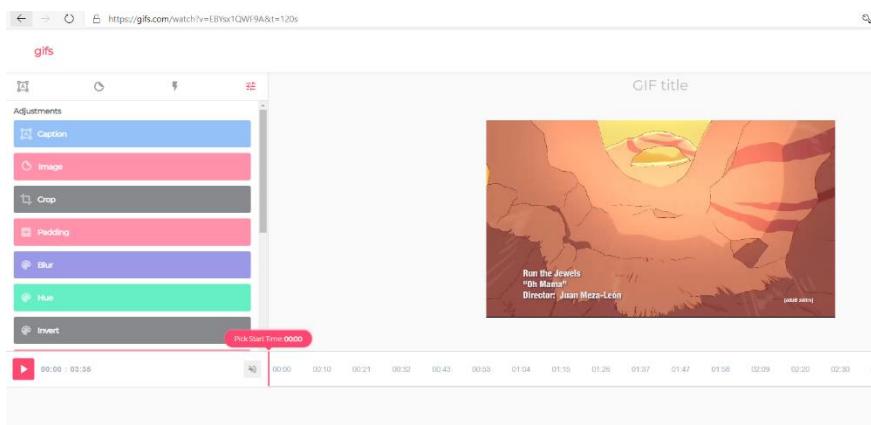
Recurrimos a YouTube para buscar el video original:

The screenshot shows a YouTube search results page with the URL https://www.youtube.com/watch?v=8yuKSvKiAEI. The search bar contains the query 'Rick and Morty'. The main image is a still from the show 'Rick and Morty' titled 'X RUN (LSD)'. It features a purple-skinned character with spiky hair and sunglasses driving a car. Below the image, the video title 'RICK AND MORTY TRIP/ X RUN (LSD)' is displayed, along with '12.390 visualizaciones' and 'Fecha de estreno: 14 abr 2019'. At the bottom right, there are like and dislike buttons with the numbers '133' and '4' respectively.

Una vez elegido el video, utilizamos la herramienta conjunta de Youtube y GIF, que nos permite generar un .gif de cualquier video de Youtube:



Redes en Cloud y Ciberseguridad



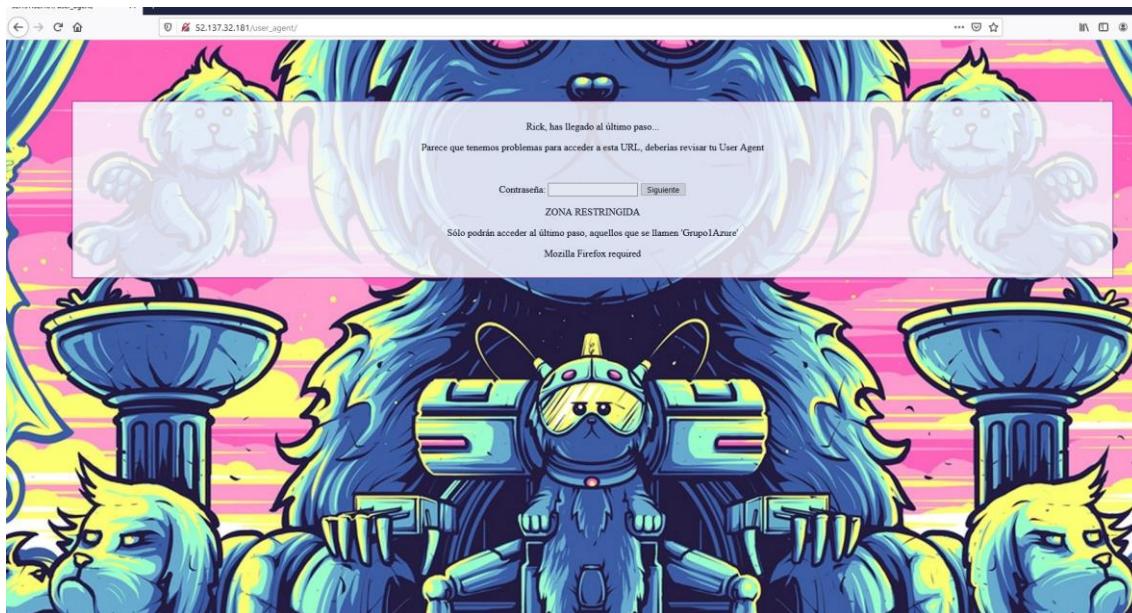
Una vez generado nuestro .gif, sólo nos falta agregar la clave de acceso, por lo que recurrimos a Giffing Tool:





8.7 User Agent

Para el último paso necesitamos cambiar de navegador, puesto que Firefox nos da más flexibilidad con el tema de peticiones HTTP.



Lo más importante de este paso, es nuestra función, que nos permite comprobar si el User Agent es válido:

```
<script>
var x = navigator.userAgent;
function prueba1(){
    if (document.form.password.value=='181220' && x ='Grupo1Azure')
        {
            document.form.submit();
        }
    else
        {
            alert("Por favor, utilice un user_agent válido para este directorio.");
        }
}
</script>
```

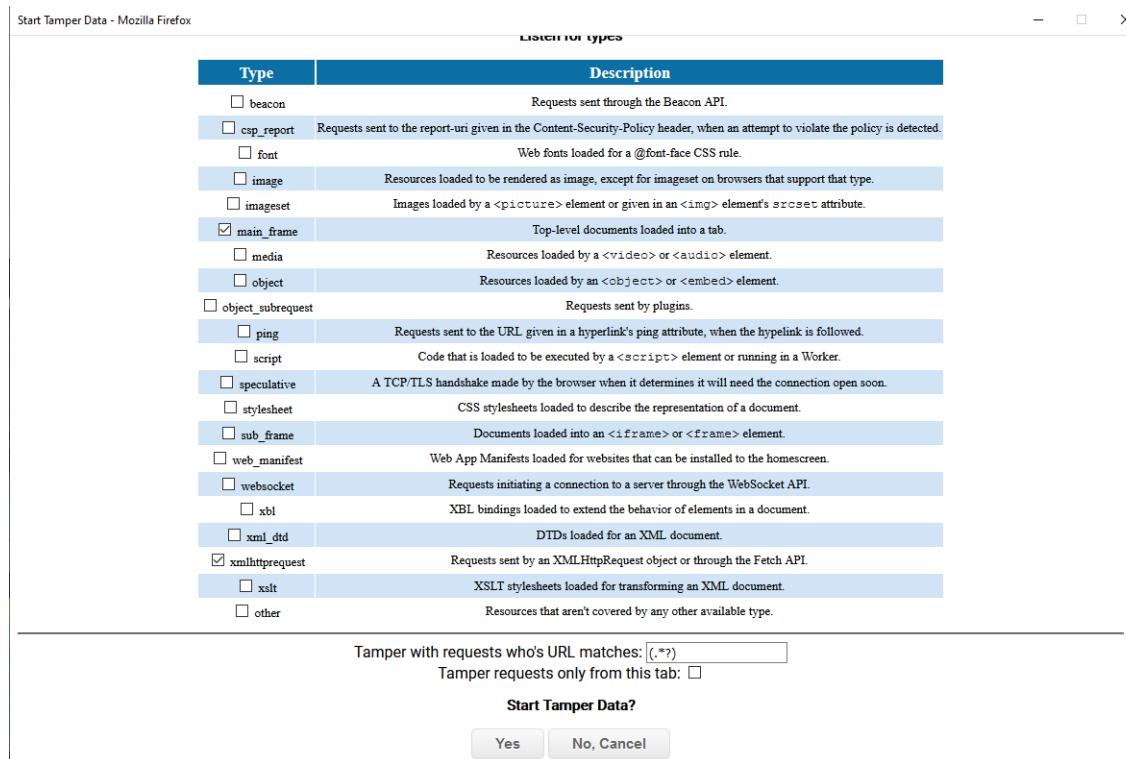
```
<FORM name=form action="/user_agent/ultimopaso.html">
<P>Contraseña: <INPUT type=password name=password>
<INPUT onclick=prueba1() type=button value=Siguiente>
</FORM>
```



Redes en Cloud y Ciberseguridad

Para el desarrollo de este paso, necesitamos instalar una extensión (Tamper Data) en nuestro navegador Mozilla Firefox, para poder editar las peticiones HTTP que le realizamos a nuestro servidor, puesto que sólo es válido el user_agent: "Grupo1Azure".

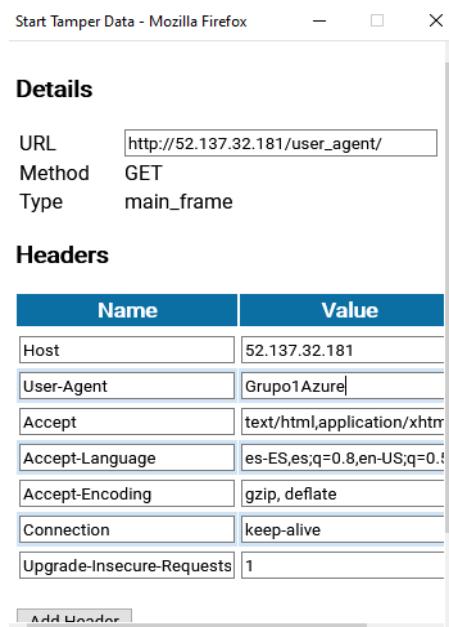
Una vez instalada, iniciamos Tamper Data sobre la URL:



Type	Description
<input type="checkbox"/> beacon	Requests sent through the Beacon API.
<input type="checkbox"/> csp_report	Requests sent to the report-uri given in the Content-Security-Policy header, when an attempt to violate the policy is detected.
<input type="checkbox"/> font	Web fonts loaded for a @font-face CSS rule.
<input type="checkbox"/> image	Resources loaded to be rendered as image, except for imageset on browsers that support that type.
<input type="checkbox"/> imageset	Images loaded by a <picture> element or given in an element's srcset attribute.
<input checked="" type="checkbox"/> main_frame	Top-level documents loaded into a tab.
<input type="checkbox"/> media	Resources loaded by a <video> or <audio> element.
<input type="checkbox"/> object	Resources loaded by an <object> or <embed> element.
<input type="checkbox"/> object_subrequest	Requests sent by plugins.
<input type="checkbox"/> ping	Requests sent to the URL given in a hyperlink's ping attribute, when the hyperlink is followed.
<input type="checkbox"/> script	Code that is loaded to be executed by a <script> element or running in a Worker.
<input type="checkbox"/> speculative	A TCP/TLS handshake made by the browser when it determines it will need the connection open soon.
<input type="checkbox"/> stylesheet	CSS stylesheets loaded to describe the representation of a document.
<input type="checkbox"/> sub_frame	Documents loaded into an <iframe> or <frame> element.
<input type="checkbox"/> web_manifest	Web App Manifests loaded for websites that can be installed to the homescreen.
<input type="checkbox"/> websocket	Requests initiating a connection to a server through the WebSocket API.
<input type="checkbox"/> xbl	XBL bindings loaded to extend the behavior of elements in a document.
<input type="checkbox"/> xml_dtd	DTDs loaded for an XML document.
<input checked="" type="checkbox"/> xmlhttprequest	Requests sent by an XMLHttpRequest object or through the Fetch API.
<input type="checkbox"/> xslt	XSLT stylesheets loaded for transforming an XML document.
<input type="checkbox"/> other	Resources that aren't covered by any other available type.

Tamper with requests who's URL matches: (.*)
Tamper requests only from this tab:

Start Tamper Data?



Details

URL: http://52.137.32.181/user_agent/
Method: GET
Type: main_frame

Headers

Name	Value
Host	52.137.32.181
User-Agent	Grupo1Azure
Accept	text/html,application/xhtml+xml
Accept-Language	es-ES,es;q=0.8,en-US;q=0.5
Accept-Encoding	gzip, deflate
Connection	keep-alive
Upgrade-Insecure-Requests	1



Redes en Cloud y Ciberseguridad

Una vez completado, obtendremos la última página, la cual nos indica, que el reto ha finalizado con éxito. ¡Felicitaciones, Rick, has conseguido atrapar a Evil Morty!



De manera que al finalizar nos debería de quedar una organización del apache de la siguiente manera.

```
└── qr
    ├── index.html
    └── qr_rick_1.png
├── qr_rick_1.png
├── rick_profile.png
├── rick_user_agent.jpg
└── robots.txt

└── pass.html
└── Rusia
    ├── in
    ├── index.html
    ├── index.html.save
    ├── index1.html
    └── internal
        └── index.html
            └── internal.jpg
        └── rusia.jpg
```

```
└── OperacionC_137
    ├── billete.html
    ├── billete_rick.jpg
    └── index.html
```

```
└── user_agent
    ├── Rick_Sanchez.png
    ├── fin.jpg
    ├── fin1.jpg
    ├── index.html
    ├── rick_user_agent.jpg
    └── ultimopaso.html

    └── user_agent_img.png
    └── user_agent_ric.jpg
```



9. Reto 3: Man in the middle

9.1 Máquina Ubuntu

En la máquina virtual de Ubuntu hacemos un update.

```
apt-get update
```

Procedemos a instalar el servidor ftp.

```
apt-get install vsftpd
```

Creamos un usuario de Linux que nos servirá también para el ftp.

```
adduser pepe
```

Creamos la carpeta de archivos ftp de nuestro usuario.

```
mkdir /home/pepe/ftp
```

Abrimos para proceder a editar el archivo de configuración del ftp.

```
nano /etc/vsftpd.conf
```

No podemos permitir las conexiones anónimas.

```
anonymous_enable=NO
```

Descomentaremos la siguiente opción para que cada usuario no pueda salir de su propio directorio. Guardamos los cambios y salimos.

```
chroot_local_user=YES
```

Arrancamos el servidor ftp.

```
service vsftpd start
```



Comprobamos su estado.

```
service vsftpd status
```

```
● vsftpd.service - vsftpd FTP server
   Loaded: loaded (/lib/systemd/system/vsftpd.service; enabled; ven
   Active: active (running) since mié 2020-11-25 16:04:52 CET; 18mi
     Main PID: 2845 (vsftpd)
       CGrou... └─2845 /usr/sbin/vsftpd /etc/vsftpd.conf
```

Ahora que parece que todo ha ido bien vamos a pasar a esconder la flag en algún punto del ftp, para ello vamos a crear primero varias carpetas para dificultar la búsqueda.

```
administrador@administrador-VirtualBox:/home/pepe$ tree
└── ftp
    ├── files
    │   ├── bbdd
    │   ├── data
    │   ├── redes
    │   │   └── mapa.img
    │   ├── seguridad
    │   │   ├── encrypt
    │   │   ├── firewall
    │   │   ├── logs
    │   │   ├── proxy
    │   │   ├── ssl
    │   │   │   ├── protocol
    │   │   │   └── signatures
    │   │   │       └── flag.txt.asc
    │   │   └── vpn
    │   └── web
    │       ├── css
    │       ├── html
    │       │   ├── index.html
    │       │   └── pagina.html
    │       ├── js
    │       │   ├── funciones.js
    │       │   └── pass.js
    │       ├── php
    │       │   └── login.php
    │       └── python
```

Y ahora después de hacer unas cuantas carpetas introducimos la bandera en la siguiente ruta:

```
administrador@administrador-VirtualBox:/home/pepe/ftp/files/seguridad/ssl/signatures
GNU nano 2.5.3                                     Archivo: flag.txt
FLAG=0415782408475098473594387509847
```

Encriptamos el archivo con gpg y contraseña “iesleonardo”.

```
gpg --ca flag.txt
```



Al desencriptar vemos que todo sale correctamente.

```
administrador@administrador-VirtualBox:/home/pepe/ftp/files/seguirdad/ssl/signatures$ sudo gpg -d flag.txt.asc
gpg: anillo «/home/administrador/.gnupg/secring.gpg» creado
gpg: datos cifrados AES
gpg: cifrado con 1 contraseña
FLAG=0415782408475098473594387509847
```

Pues ésta máquina estaría totalmente preparada y podríamos pasar con la siguiente.

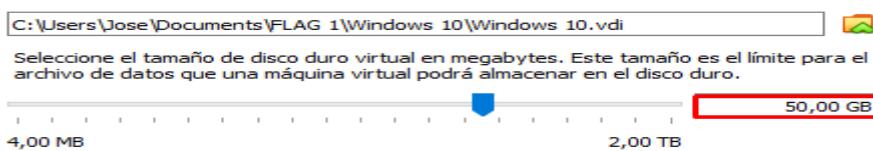
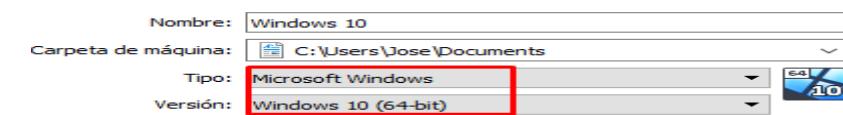
9.2 Windows 10 (Atacante)

Para descargar Windows 10 vamos a ir a la siguiente URL :

[https://mega.nz/file/uc9EhYSA#yS8M7-q5hIrls2J1Qa55Q1bqWloJ3VjNIFRrNNiToKE \(*6\)](https://mega.nz/file/uc9EhYSA#yS8M7-q5hIrls2J1Qa55Q1bqWloJ3VjNIFRrNNiToKE)

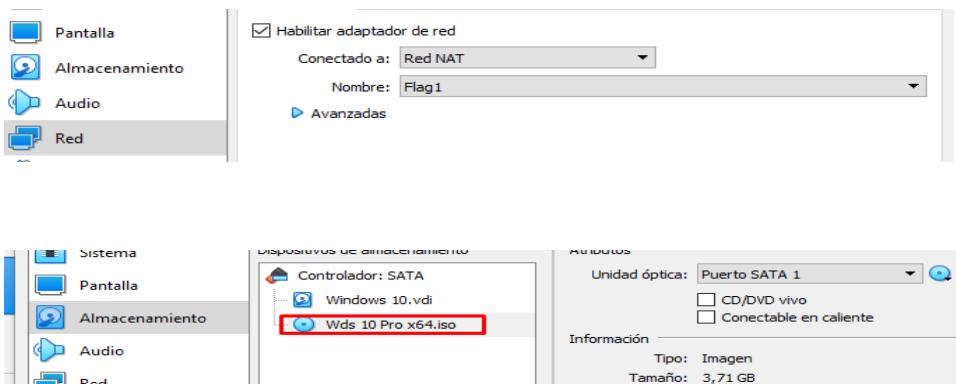
Vamos a virtualbox y para crear la nueva máquina virtual de Windows 10.

Los pasos son los mismos que realizamos con la máquina anterior, dejaré unas capturas de lo más importante.





Redes en Cloud y Ciberseguridad



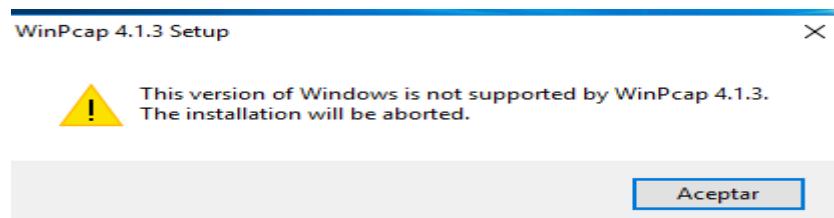
Dado que la instalación de un Windows es bastante sencilla no la documentaremos para no saturar el documento, ya que son solo ventanas pinchando en siguiente.

Ahora una vez tenemos Windows arrancado comprobamos que la configuración de red es la correcta.

Ahora vamos a ir a la siguiente URL para descargar Cain y Abel :

<https://mega.nz/file/Y6R21YBY#syZOkjVhiPQT8zUEgOqtVCMcRcDq4Y-EHxXRcKuY1ps> (7*)

La instalación de este programa no tiene ningún tipo de complicación, se acepta todo según viene predefinido. Al terminar de finalizar nos saltará el siguiente error.



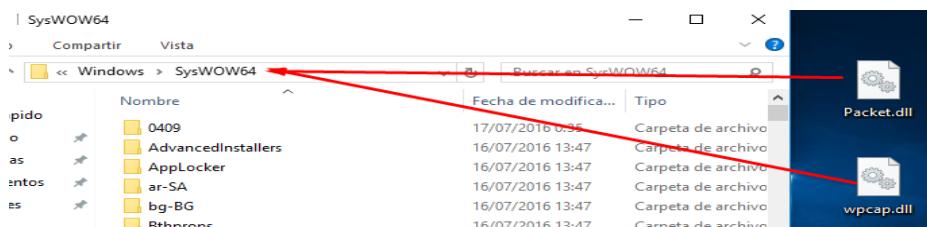


Redes en Cloud y Ciberseguridad

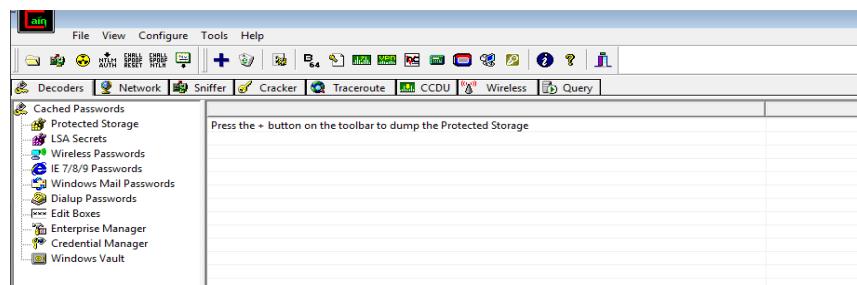
Este error es debido a que le fatan un par de librerías. Para descargarlas vamos a ir a la siguiente URL:

<https://mega.nz/file/5aZCTKDY#yf7R8Hsm5PgGzhPpxerErJAomDGmm2IU-WDQ4R2W3Nw> (*8)

Una vez que lo hayamos descargado, se descomprime y se llevan los dos archivos a la siguiente carpeta.



Como podemos observar ya no hay ningún problema.



Antes de salir de esta máquina vamos a comprobar si el ftp de la anterior es accesible.

```
C:\Users\admin>ftp 10.0.2.9
Conectado a 10.0.2.9.
220 (vsFTPd 3.0.3)
200 Always in UTF8 mode.
Usuario (10.0.2.9:(none)): pepe
331 Please specify the password.
Contraseña:
230 Login successful.
ftp>
```



Dejaremos instalado también el nmap a modo de despiste, lo descargaremos desde la web oficial.

Una vez terminada ésta máquina vamos a terminar con la última instalación.

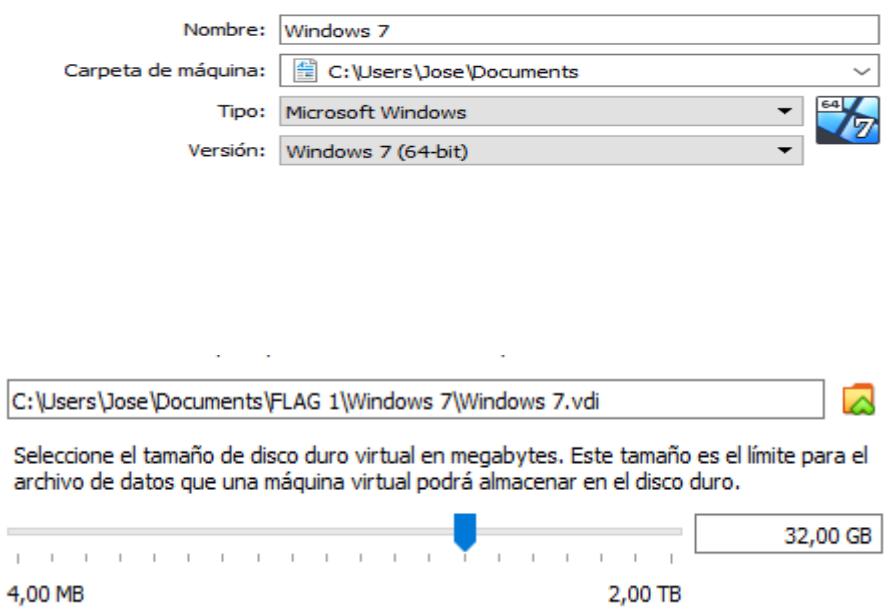
9.3 Windows 7 (Cliente FTP)

Para ponernos en contexto, ésta máquina va a ser la que se conecte al servidor FTP y de la que tiene que obtener Windows 10 la contraseña.

Para descargar este sistema vamos a ir a la siguiente URL :

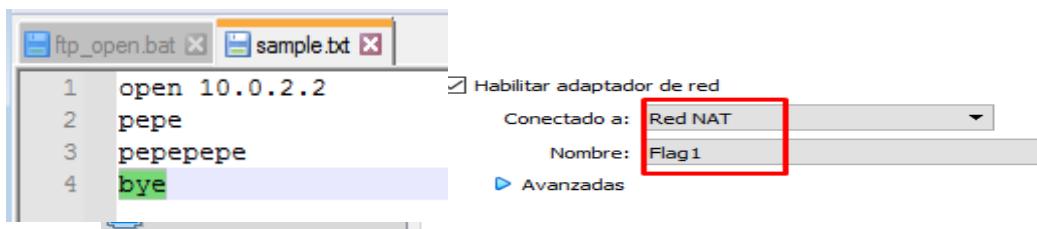
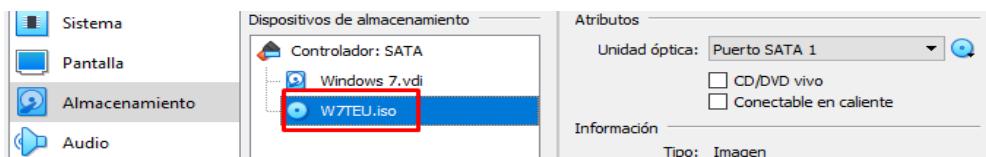
<http://www.mediafire.com/file/dnn60d85sk73nxw/W7TEUISO.zip/file> (*9)

Una vez tengamos la iso, vamos a crear una nueva máquina en virtualbox, dejo las capturas más importantes de nuevo.

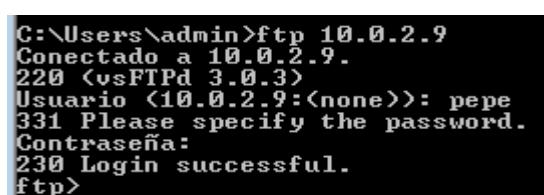
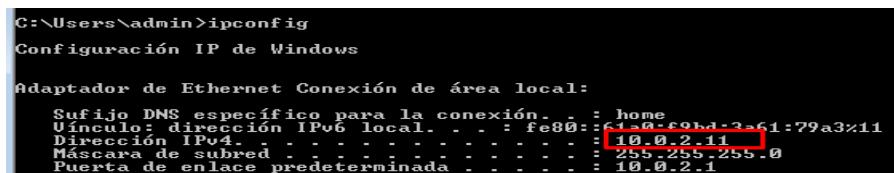




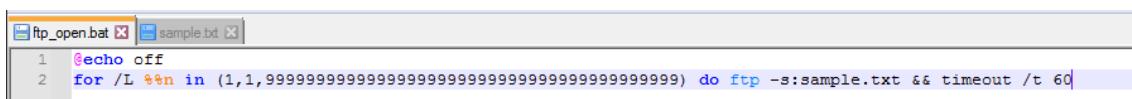
Redes en Cloud y Ciberseguridad



Vamos a obviar los pasos de la instalación ya que como hemos dicho anteriormente, la instalación de Windows es muy simple. Cuando la tengamos instalado habremos terminado, ya que no hay que instalar ningún tipo de programa en ella, simplemente comprobar la red y conexión al ftp.



Dejaremos un script de conexión automática que he hecho, para que en el momento que haya que loguearse todo sea rápido.





10. Reto 4: Criptografía

10.1 Sopa de letras

The screenshot shows a web page titled "SOPA DE LETRAS DEL IMPERIO ROMANO". It displays a 9x9 grid of letters:

D	S	Z	W	E	E	D	I
S	I	J	H	A	R	F	G
V	I	V	X	T	M	Q	S
B	D	N	I	M	B	U	O
U	A	E	P	D	O	M	P
C	E	I	M	P	E	R	A
P	R	Q	D	W	L	I	E
H	G	F	S	A	B	T	E

Below the grid, a message reads: "Tu objetivo es encontrar la siguiente cita de Julio César: 'Divide y obtén el poder'". There is a text input field and a "Validar" button.

Para este reto, hemos utilizado de nuevo nuestra máquina de Azure con APACHE2WEB, creando un nuevo directorio y aplicando las configuraciones pertinentes.

```
└── Reto2
    ├── Laberinto
    │   ├── Crucigrama
    │   │   └── crucigrama.pdf
    │   ├── index.html
    │   ├── crucigrama.pdf
    │   ├── descarga.png
    │   ├── index.html
    │   ├── laberinto.gif
    │   ├── reloj1.jpg
    │   ├── reloj2.jpg
    │   ├── reloj_arena.jpg
    │   └── vaporwave.jpg
    ├── index.html
    ├── julio_cesar.jpg
    └── pass.html
```



La sopa de letras está hecha directamente en código HTML, mediante tablas, para facilitar la inserción del resto del código:

```
<table border="1">
<tr>
    <td></td>
</tr>
</table>
```

Además, contamos con dos scripts fundamentales para el perfecto funcionamiento del reto:

Función VALIDAR:

```
<script>
    var y=0;
    function clave(){
        if(document.getElementById("texto").value == "QVIVQR RG VZCREN")
            {
                document.form.submit();
            }
        else
            {
                document.write('Clave Incorrecta, por favor, actualice la página para continuar. PISTA: IDIOMA ANTIGUA ROMA');
            }
    }
</script>
```

Script para que no se pueda visualizar el código fuente:

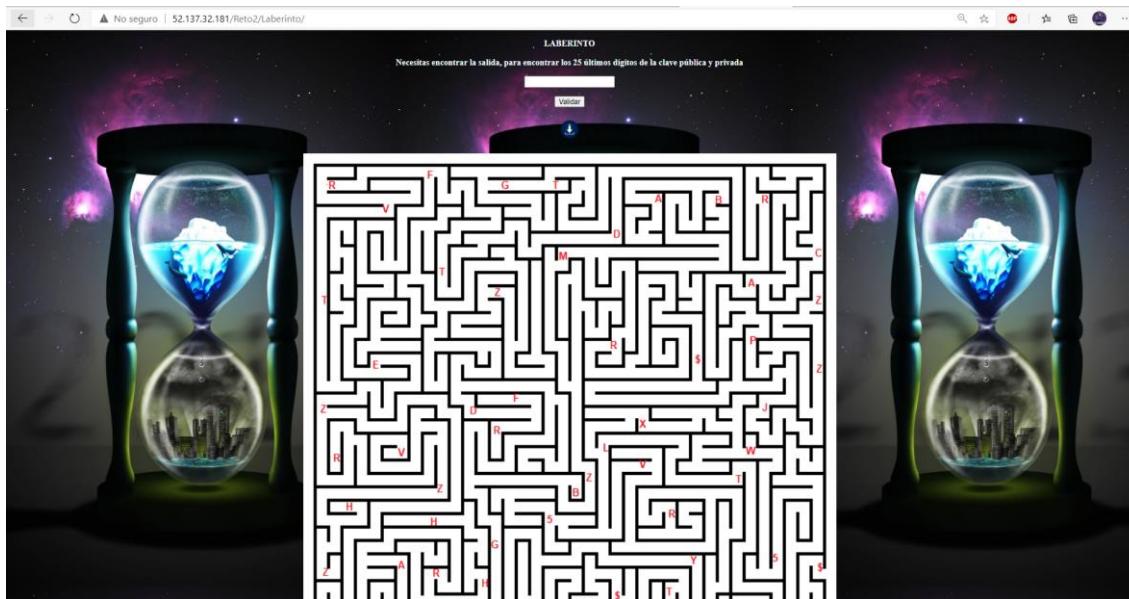
```
<script type="text/javascript">
document.oncontextmenu = function(){return false;}
</script>
```

El estilo de nuestra página está definido con código CSS:

```
<style>
    .prueba { font-size: 200% }
    .cuadrado{
        padding:50px;
        opacity: 0.85;
        margin:100px;
        background-color: #F8F8FF;
        border: solid 1px blue;
        color: black;
        text-align:center;display:block;
        font-size: 200%;
    }
</style>
```



10.2 Laberinto



Para realizar este paso, hemos recurrido a Paint, herramienta estrella de Windows para editar imágenes. Con un poco de paciencia, hemos ido colocando carácter a carácter los 25 dígitos finales de nuestra clave pública.

Contamos con nuestras funciones genéricas VALIDAR y OCULTAR CÓDIGO FUENTE.

Además, hemos implementado un botón de descarga para facilitar la resolución de este reto, puesto que es más sencillo, descargar la imagen y con Paint, dibujar una línea hasta llegar a la solución final:

```
<p align="center">
<input type="text" id="texto">
<FORM ACTION="/Rusia" METHOD=GET>
<button onclick="clave()">Validar</button></p>
<p><a href="https://imgur.com/a/0UgaHTC" download="cute.jpg">

</a></p>
```



10.3 Crucigrama

The screenshot shows a web browser window with a crossword puzzle titled "CryptoCTF". The puzzle grid is partially filled with letters. Below the grid, there are two lists: "HORIZONTALES" and "VERTICALES", each containing numbered clues. The "HORIZONTALES" list includes clues 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, and 12. The "VERTICALES" list includes clues 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, and 12. At the bottom of the page, there is some explanatory text in Spanish about certificate creation and private keys.

Hemos decidido implementar una serie de preguntas rápidas sobre la criptografía, que a su vez, forma parte del reto.

En este caso, hemos generado un crucigrama en la página olesur.com, dándonos así posibilidad de implementar directamente un PDF en nuestra página web, lo que nos ha permitido poder editar el documento, sin tener que implementar otra solución.

El objetivo de este reto es aprender de una manera entretenida, ciertos fundamentos de la criptografía.

Tenemos nuestros dos scripts genéricos aplicados a esta web:

```
<script>
    document.oncontextmenu = function(){return false;}
</script>
<script>
    function clave(){
        if(document.getElementById("texto").value == ".")
        {
            document.form.submit();
        }
        else
        {
            document.write('Clave Privada incorrecta, actualice la página');
        }
    }
</script>
```

Además, hemos implementado el archivo .PDF en nuestra web con iframe:

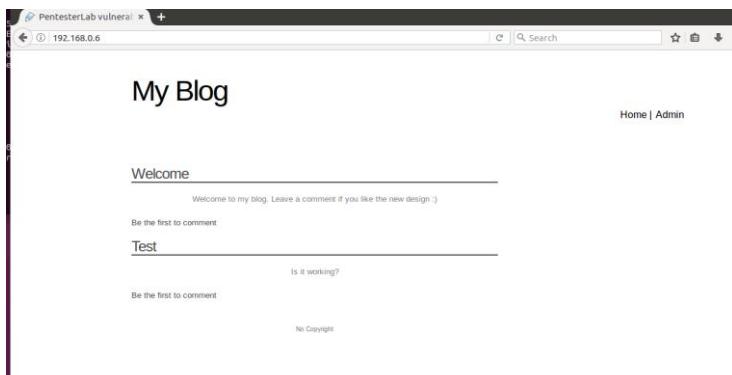
```
<iframe src="crucigrama.pdf" width="100%" height="1500px">
```



11. Reto 5: XSS Attack

11.1 Configuración de Debian (Servidor Web)

Ésta máquina es descargada de Vulnhub. En un principio tiene esta apariencia.



La hemos cambiado algunos colores, traducido la mayoría del texto y ha quedado de esta forma.



Entramos en mysql para comprobar que existe.

```
cd /var/www/clases
```

```
sudo mysql
```



Una vez nos deja entrar, comprobamos que existe la bbdd vinculada a la web.

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| blog |
| mysql |
+-----+
3 rows in set (0.00 sec)
```

Aquí es donde se almacenará nuestro script (“comments”).

```
mysql> show tables;
+-----+
| Tables_in_blog |
+-----+
| comments |
| posts |
| users |
+-----+
3 rows in set (0.00 sec)
```

Escondemos la bandera entre el código de la página del administrador.

```
</table>
<h2 style="color:red;">FLAG=348975698756</h2>
<a href="new.php" style="color:yellow;">Escribe un nuevo post</a>
</div>
```

11.2 Configuración de Ubuntu (Atacante)

En ésta máquina no tendríamos que tocar nada, simplemente le haremos un update para que tenga listos los repositorios por si le hace falta al jugador descargar algún programa.

```
sudo apt-update
```



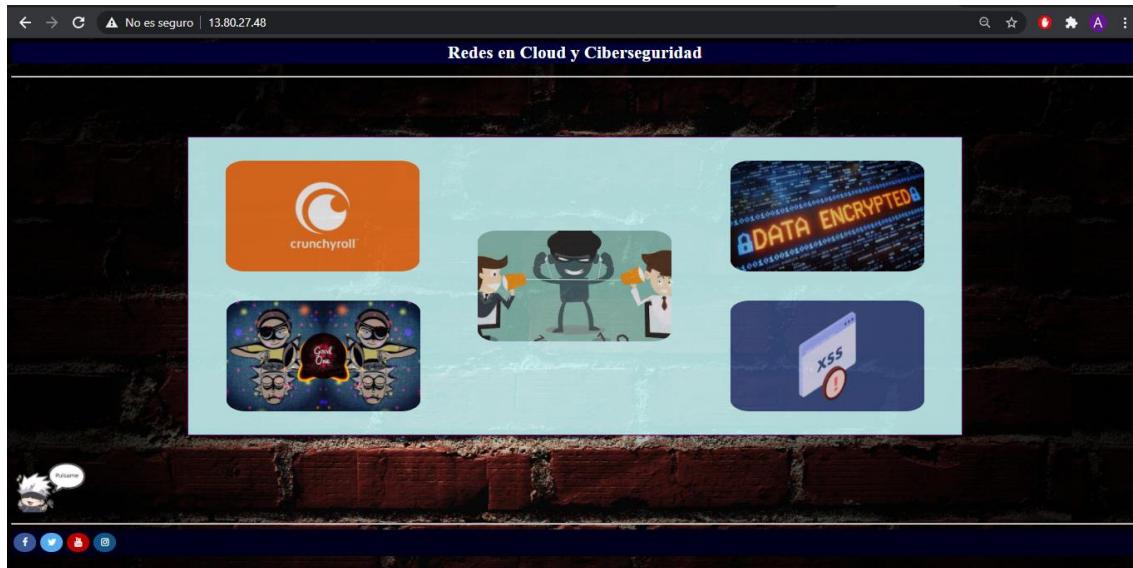
12. Página web

La página web está montada sobre Azure, al igual que las anteriores maquinas usadas en Azure, tiene la mínima configuración, medio giga de disco duro y un CPU, aunque más que suficiente para esta función, además de ello, dejaremos el puerto 80 abierto para las conexiones con ella, ya que si no se hiciera esta regla sobre Azure, este denegaría cualquier conexión hacia ese puerto.

Prioridad	Nombre	Puerto	Protocolo	Origen	Destino	Acción	
300	▲ SSH	22	TCP	Cualquiera	Cualquiera	<input checked="" type="checkbox"/> Permitir	...
320	HTTP	80	TCP	Cualquiera	Cualquiera	<input checked="" type="checkbox"/> Permitir	...

La configuración de apache es exactamente igual a las realizadas en los retos de manera que no es necesaria su explicación nuevamente para el caso de la página web.

La página contendrá una imagen con cada uno de los retos, de manera que si pinchamos sobre ellos nos enviará a la explicación del mismo, junto con su objetivo y correspondiente IP y reglas para superarlo. Las siguientes imágenes hacer referencia a lo explicado anteriormente.





Redes en Cloud y Ciberseguridad

Reto 1: Crunchyroll

Introducción:

En el mundo actual en el que nos movemos las habilidades en la informática y sobre todo las habilidades en cuanto a la ciberseguridad son indispensables, especialmente para las grandes empresas que disponen de cientos de servidores alojados en diferentes partes del mundo y que son atacados casi de manera diaria. Por eso en este pequeño reto os explicamos de una manera muy básica como es posible tomar el control de una máquina cuando esta no está bien configurada o los permisos de ciertos archivos no están bien configurados.

Escenario:

Te encuentras en las oficinas de Crunchyroll, empresa dedicada a la distribución legal de anime. Has decidido a ir allí para una entrevista de trabajo. Como reto para superar la entrevista y ver que tienes conocimientos suficientes para cubrir el puesto ofrecido, te proponen conseguir vulnerar una máquina virtual, en caso de conseguirla el puesto será tuyo.



Objetivo:

El chico de relaciones públicas te explica lo siguiente:

1. Es sencillo solo tienes que conseguir una bandera que se encuentra en el usuario root.

Consejos:

1. Sería bueno que useses un sistema operativo como Kali o Parrot.
2. El servidor está alojado en internet no en una red local.

Pistas: Como le has caído bien al entrevistador te da un par de pistas para que no estés muy perdido.

1. Seguramente los nombres de usuarios tengan que ver con personajes de animes.

2. Creo recordar que el que hizo la máquina mencionó algo de escenografía pero no me acuerdo con que programa lo hizo.

Último apunte:

Espero que disfrutes del reto y sobre todo que puedas finalizarlo para conseguir el puesto de trabajo. Una última cosita... la ip de la máquina es 51.144.174.75

Reto2: Rick & Morty

Introducción:

Dentro de la ciberseguridad, el juego CTF (Capture The Flag) es muy común, puesto que nos permite aumentar nuestras capacidades informáticas tanto de defensa, como de ataque. Así mismo, hay diferentes tipos de CTF y en este reto, vamos a desarrollar la *Esteganografía*, que consiste en encontrar, mediante la utilización de ciertas herramientas, información oculta, dentro de una URL o incluso en una imagen.



Escenario:

Tu nombre es Rick Sánchez, pertenes a la Tierra C-137 y tu misión es encontrar a Evil Morty, que en este caso, está escondido en algún punto del universo. Mediante el uso de diferentes técnicas informáticas, podrás conseguir descubrir el paradero de Evil Morty, antes de que éste decida destruir la Tierra C-137.

Objetivo:

1. Debes llegar al final del reto, siguiendo poco a poco el rastro de Evil Morty.

Consejos:

1. ¡Cuidado! Evil Morty tratará de despistar, intenta centrarte en la misión y no irte por las ramas.
2. Deberás utilizar para la mayoría de retos Google Chrome o en su defecto, Microsoft Edge.

Pistas:

1. ¡Ninguna! Sólo ante el peligro Rick, como siempre.

Último apunte:

Esperamos que disfrutes del reto, igual que disfrutamos nosotros realizándolo. Mucha suerte Rick Sánchez, no será fácil atrapar a Evil Morty, pero con esfuerzo, lograrás tu misión.

[Pulsa aquí para comenzar el reto](#)

Man in the middle

Introducción:

Hoy en día casi todo el mundo poseemos un móvil o un ordenador. Cuando nos encontramos en nuestro hogar, tenemos la tranquilidad de nuestro acceso wifi seguro, pero cuando andamos de aquí para allá y queremos borrar datos, a veces nos surge la tentación de conectarnos a redes abiertas en la calle o centros comerciales. Este es un craso error, porque no sabemos quién ha creado ese acceso o quién está siguiendo su tráfico de red. A continuación, vas a comprobar porque solo debes conectarte a redes seguras. Hay una bandera escondida en algún lugar y debes encontrarla. Tienes que pensar que en vez de una bandera, podrían ser datos personales o económicos, e incluso datos por los que puedan llegar a extorsionarte.



Escenario:

En casa nos aburrimos, acudimos a un centro comercial a dar un paseillo, después de varias vueltas, observamos que el lugar está lleno, y como somos gente curiosa, cogemos nuestro dispositivo y buscamos redes, cómo no, observamos una red abierta sin contraseña y accedemos a ella.

Objetivo:

Comprueba tu mismo, si es verdad que se pueden obtener datos de una red desconocida:

1. Escanea la red y verifica que equipos tienes alrededor.
2. Envenena la red.
3. Realiza un sniffing de contraseñas.
4. Busca la bandera.

Consejos:

1. Busca en la máquina que se te proporciona aplicaciones que analicen la red y los protocolos en el pc.
2. Si necesitas descargar algún programa extra, hazlo.

Pistas:

- Si escoges el programa correcto no tendrás problema en resolver todo ésto, todo depende de tus conocimientos.

Último apunte:

Disfruta del reto y recuerda que es educativo. Sólo se intenta dejar ver los peligros a los que estamos sometidos hoy en día con nuestros dispositivos conectados a internet.

Descarga la máquina desde el siguiente enlace :

[Mega](#)



Redes en Cloud y Ciberseguridad

Reto 4: Criptografía

Introducción:

Como bien hemos visto con los anteriores retos, hay diferentes tipos de retos CTF, como bien puede ser la Esteganografía o, en este caso, la criptografía. Los retos suelen consistir en utilizar diferentes técnicas para desencriptar/encryptar cierta información. En nuestro caso, vamos a hacer uso de diferentes acertijos, para resolver el reto.

Escenario:

Eres parte del equipo de ciberseguridad de una pequeña empresa, por lo que tu jefe, quiere que aprendas más acerca de la criptografía, por lo que te plantea una serie de retos.

Objetivo:

1. Tu objetivo principal es resolver los retos en el menor tiempo posible, siguiendo las propias reglas, además de aprender más sobre la criptografía.

Consejos:

1. Analiza detenidamente cada reto, pensando cuál podría ser la solución final.
2. Recomendamos la utilización de Paint para la resolución de los primeros pasos.

Pistas:

1. En el segundo desafío, deberás empezar el recorrido, desde abajo a la izquierda... ¿O no?

Último apunte:

Google siempre te va a ayudar a encontrar la información pertinente ¡Mucho suerte!

[Pulsa aquí para comenzar el reto](#)



XSS Attack

Introducción:

Existen millones de páginas web, las hay de muchos tipos y para muchos propósitos diferentes, hay algo que muchas tienen en común, que ofrecen un acceso tanto para usuarios como para el administrador de la misma. Para evitar este tipo de acceso existe un tipo de ataque llamado "Cross Site Scripting". Existen dos tipos de este mismo ataque pero nos vamos a centrar en el almacenado.

Escenario:

Nos encontramos frente a una página web de tipo blog, en la que observamos varios post, un acceso para el administrador y un apartado con un formulario para escribir comentarios.

Objetivo:

Ejecuta un ataque XSS a partir de los siguientes pasos:

1. Comprueba la vulnerabilidad de la página ante ataques XSS.
2. Crea una página escrita en "php" para capturar el valor de una cookie.
3. Crea un servidor "php" en tu equipo y mántórelo.
4. Inyecta código javascript en el formulario de los comentarios para desviar el tráfico a tu servidor, con la página "php" que has creado.
5. Captura la cookie con el monitoreo de tu servidor.
6. Busca un programa para agregar la cookie a la página real, en tu propio navegador.

Consejos:

1. Ten clara la mecánica del XSS almacenado antes de realizar éste reto.
2. Comprueba bien el código en busca de errores de sintaxis o direcciones IP que te puedan ocasionar fallos a la hora de realizar el ataque.
3. Si necesitas descargar algún programa extra, hazlo.
4. Si sigues los objetivos uno a uno, con un poco de programación puedes resolver este reto.

Pistas:

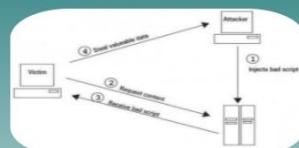
- Si dedicas 5 min a buscar en internet este tipo de ataque puede que obtengas bastante información.

Último apunte:

Disfruta del reto y recuerda que es educativo. Esto solo demuestra lo fácil que sería robar nuestros datos en una página web que no esté protegida contra este tipo de ataques.

Descarga la máquina desde el siguiente enlace:

[Mega](#)



El código de la página se encontrará en un archivo en Typora ya que mantiene la forma de este correctamente junto con sus saltos de línea y tabulaciones. También puedes encontrarlo en la siguiente dirección de nuestro repositorio de Github.

- URL Github: <https://github.com/grupo1azure/webprincipal> (*10)
- URL de la página: <http://leonardoctf.ml/> (*11)
- URL IP de la página: <http://13.80.27.48/> (*12)



13. Diagrama de Gantt

Actividades	Tiempo de Duración											
	Octubre			Noviembre			Diciembre			Enero		
	Semana 1	Semana 2	Semana 3	Semana 4	Semana 1	Semana 2	Semana 3	Semana 4	Semana 1	Semana 2	Semana 3	Semana 4
Discusión sobre el tema del que podría tratar el proyecto.												
Elección sobre la plataforma ha utilizar para el proyecto.												
Reparto de tareas y distribución de las tareas entre los integrantes del equipo.												
Reto 1												
Reto 2												
Reto 3												
Reto 4												
Reto 5												
Página Web												

Para un seguimiento más preciso visite el siguiente link:

<https://trello.com/invite/b/iBSDBBdo/72c0b97a4f778ba5600a0ea2eba3dc2b/tfg-ctf> (*13)

14. URL de interés

(*1) **Manytools**: Herramienta utilizada para crear dibujos en ASCII.

(Consultado: 30 de enero de 2021)

(*2) **Qrcode**: Herramienta utilizada para generar un código qr.

(Consultado: 30 de enero de 2021)

(*3) **Zxing**: Herramienta utilizada para sacar la información del código qr.

(Consultado: 30 de enero de 2021)

(*4) **Metashieldclean**: Herramienta utilizada para extraer las coordenadas de una imagen.

(Consultado: 30 de enero de 2021)

(*5) **Geosetter**: Herramienta para asignar unas coordenadas a una imagen.

(Consultado: 30 de enero de 2021)

(*6) **Mega**: Descargar ISO Windows 10.

(Consultado: 30 de enero de 2021)

(*7) **Mega**: Descargar Cain & Abel.

(Consultado: 30 de enero de 2021)



(*8) Mega: Descargar librerías WinPCap.

(Consultado: 30 de enero de 2021)

(*9) Mediafire: Descargar ISO Windows 7.

(Consultado: 30 de enero de 2021)

(*10) Github: Código fuente de todo el proyecto.

(Consultado: 30 de enero de 2021)

(*11) Leonardoctf: Página web del proyecto.

(Consultado: 30 de enero de 2021)

(*12) <http://13.80.27.48/>: Ip de la página web del proyecto.

(Consultado: 30 de enero de 2021)

(*13) Trello: Organización del proyecto.

(Consultado: 30 de enero de 2021)



15. Conclusiones

Desde el primer momento teníamos una idea más o menos clara de lo que queríamos hacer como proyecto. Como hemos explicado al inicio de esta memoria queríamos realizar un atrapa la bandera con diferentes retos y que todos ellos estuviesen juntos y explicados de una manera lo más clara posible para su realización en una página en internet. Para realizarlo hemos tenido que elegir entre diferentes tecnologías, decantándonos al final por el servicio de Azure, en el cual, contactando con el servicio técnico nos ofrecían un plan de estudiantes sin coste alguno con cien créditos a gastar durante un año completo o hasta agotarlo.

Hemos tocado nuevas tecnologías como la ya mencionada Azure, investigado maneras con las que poder ocultar mensajes mediante la criptografía, hacer ataques de fuerza bruta contra los directorios de páginas web para conocer su estructura como Dirb o Gobuster, creación de una página web sin necesidad de gestores de contenido, tan solo buscando código en internet que podríamos usar e implementarlo en nuestra página, vulnerar una base de datos gracias a un pequeño script, etc.

Este proyecto nos ha permitido adquirir más conocimientos, tanto en programación como en seguridad informática, de tal manera hemos conseguido superar dificultades que durante el curso nos resultaban complicadas e incluso imposibles. Por lo que creemos que este proyecto ha tenido importancia en nuestra educación.

En líneas generales el proyecto en un futuro se podría seguir mejorando tanto en la presentación de la página web como en posibles retos que se podrían añadir cuando adquiramos muchos más conocimientos, puesto que este proyecto es una introducción al mundo de la ciberseguridad, especialmente en los CTFs.

Por otra parte pensábamos que debido a las circunstancias por el virus y que algunos terminaban las prácticas antes que otros no íbamos a poder organizarnos en condiciones pero con un poco de esmero lo hemos conseguido.



16. Bibliografía

Reto 1 y Página web:

- <https://serverfault.com/questions/421161/how-to-configure-vsftpd-to-work-with-passive-mode/421169>
- https://www.youtube.com/watch?v=FOMz7YNXYis&ab_channel=SuGE3K
- <https://www.hacknos.com/vegeta-vulnhub-walkthrough/>
- <https://blogsaverroes.juntadeandalucia.es/plataformaeiv/files/2016/05/P-vsftpd-autenticacion.pdf>
- Servicios de Red e Internet 2 Editorial Garceta (Álvaro García Sánchez, Álvaro González Sotillo, Luis Enamorado Sarmiento)
- https://www.w3schools.com/howto/tryit.asp?filename=tryhow_css_social_media_buttons
- <https://www.josegonzalezbueno.com/crear-popups-con-codigo/>

Reto 2:

- [Metashield Clean-up Online \(elevenpaths.com\)](http://Metashield Clean-up Online (elevenpaths.com))
- Cómo geolocalizar dónde se hizo una fotografía | Tecnología - ComputerHoy.com
- <https://geosetter.de/en/download-en/>
- https://www.bing.com/search?q=russia+latitude+and+longitude+decimal&qs=n&form=QBRE&msbsrank=0_0&sp=-1&pq=russia+latitude+and+longitude+deci&sc=0_34&sk=&cvid=1BC647BD8FBB4C5AB3F3F7C39B15CEB7
- <https://www.peko-step.com/es/tool/alphachannel.html>
- https://www.w3schools.com/cssref/css_colors.asp
- <https://reactgo.com/html-background-image/#:~:text=%20How%20to%20set%20a%20Background%20image%20in,to%20fit%20the%20entire%20html%20element...%20More%20>
- <https://support.google.com/mail/answer/25922?co=GENIE.Platform%3DDesktop&hl=es#:~:text=Abre%20Gmail%20en%20el%20ordenador.%20Arriba%20a%20la,el%20apartado%20%22Respuesta%20autom%C3%A1tica%22.%20Selecciona%20Respuesta%20autom%C3%A1tica%20activada.>
- https://www.w3schools.com/jsref/prop_nav_useragent.asp
- <https://www.tuexperto.com/2016/11/29/como-anadir-texto-a-un-gif-animado/>



- <https://www.alvarolara.com/2013/11/08/centrar-una-imagen-en-html-a-la-mitad-de-la-pantalla/>
- <https://zxing.org/w/decode.jspx>

Reto 3:

- https://dis.um.es/~lopezquesada/documentos/IES_1314/SAD/curso/UT4/ActividadesAlumnos/grupo2/documentos/Tutorial_Cain.pdf
- <https://geekflare.com/es/nmap-on-windows/>
- <https://es.wikipedia.org/wiki/WinPcap#:~:text=Permite%20a%20las%20aplicaciones%20capturar,WinPcap%20consiste%20en%20un%20controlador%2C>

Reto 4:

- https://es.wikipedia.org/wiki/Cifrado_C%C3%A9sar
- http://servicios.educarm.es/templates/portal/images/ficheros/etapasEducativas/secundaria/13/secciones/176/contenidos/6020/codigos_secretosl.pdf
- <http://www.cert.fnmt.es/curso-de-criptografia/criptografia-de-clave-simetrica>
- https://www.w3schools.com/cssref/css_colors.asp
- https://www.w3schools.com/html/html_css.asp
- <https://www.welivesecurity.com/la-es/2020/05/13/test-poner-prueba-conocimientos-sobre-criptografia/>
- http://www.juntadeandalucia.es/averroes/centros-tic/18700232/helvia/sitio/upload/preguntas_sobre_Criptografia_y_cifrado.pdf
- <http://minubeinformatica.com/cuestionarios/2/criptografia/>
- <https://support.microsoft.com/es-es/windows/obtener-microsoft-paint-a6b9578c-ed1c-5b09-0699-4ed8115f9aa9>
- <https://www.olesur.com/>

Reto 5:

- <https://www.vulnhub.com/>
- <https://www.welivesecurity.com/la-es/2015/04/29/vulnerabilidad-xss-cross-site-scripting-sitios-web/>
- <https://diego.com.es/ataques-xss-cross-site-scripting-en-php>
- <https://www.youtube.com/watch?v=HEtPhy-fjIA>