

REPASO

Ejercicio en Grupo.

Imaginaros que sois un equipo de expertos en ciberseguridad vais a utilizar todas las herramientas que hemos visto hasta ahora para realizar vuestra labor, esto incluye pentestig para revisar que puertos tenéis abiertos, análisis de tráfico de red, de paquetes, que ordenadores están conectados a la red. También realizar análisis de logs, probando desde crear usuarios y grupos dar permisos, auditoría web etc.

También debéis demostrar como proteger vuestros equipos, por ejemplo con el uso de firewall.

El sistema operativo que vamos a usar en estos ejemplos es Kali.

Esto que veáis aquí es a modo de ejemplo, hay herramientas que no hemos visto y que por tanto no utilizaremos.

Ejercicio de Grupo — Operación de Seguridad (Lab con Kali)

Objetivo general: en equipo realizar un ejercicio completo de seguridad: reconocimiento y enumeración de la red, análisis de tráfico y paquetes, auditoría de servicios y logs, creación de usuarios y control de accesos, y finalmente aplicar medidas de protección (firewall, endurecimiento). Generar informe con hallazgos y medidas de remediación.

Duración sugerida: 1 sesión de 4–6 h (o varias sesiones según disponibilidad).

Roles recomendados en el grupo (3–5 personas): Líder/Coordinador, Recon/Red, Forense/Logs, Sysadmin/Hardening, Documentador.

ATENCIÓN: LAS IPS MOSTRADAS SON DE REFERENCIA. CADA VEZ QUE ARRANCAMOS LAS MÁQUINAS VIRTUALES LAS IPS SE ASIGNAN DINÁMICAMENTE

FASE 0 —

- Definir ordenadores/segmentos IP (ej. 192.168.56.0/24).
- Documentar alcance y reglas (qué está permitido).
- Crear repositorio con estructura: recon/, pcaps/, logs/, report/.

FASE 1 — Reconocimiento y mapeo de red

Objetivo: identificar hosts activos, puertos y servicios.

Herramientas y comandos (ejemplos):

- Descubrimiento de hosts:
 - arp-scan (si está permitido):
 - sudo arp-scan -l
 - netdiscover:
 - sudo netdiscover -r 192.168.56.0/24

122 Captured ARP Req/Rep packets, from 68 hosts. Total size: 7320

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.0.67	18:60:24:e9:3a:50	8	480	Hewlett Packard
192.168.0.1	00:a0:26:d2:68:9a	1	60	TELDAT, S.A.
192.168.0.54	6c:3b:e5:0e:91:2d	4	240	Hewlett Packard
192.168.0.66	98:ee:cb:46:94:f7	2	120	Wistron Infocomm (Zhongshan) Corporation
192.168.0.5	00:26:73:99:57:8c	1	60	RICOH COMPANY,LTD.
192.168.0.32	08:2e:5f:06:d8:2d	2	120	Hewlett Packard
192.168.0.102	6c:3b:e5:40:2c:2e	3	180	Hewlett Packard
192.168.0.123	34:17:eb:c4:6c:81	7	420	Dell Inc.
192.168.0.118	2c:41:38:ac:7d:1c	2	120	Hewlett Packard
192.168.0.41	08:2e:5f:09:05:21	2	120	Hewlett Packard
192.168.0.21	08:00:27:20:94:3e	1	60	PCS Systemtechnik GmbH
192.168.0.23	08:2e:5f:03:ac:3e	2	120	Hewlett Packard
192.168.0.24	d8:43:ae:44:e2:72	2	120	Micro-Star INTL CO., LTD.
192.168.0.25	ec:71:db:a7:4b:40	1	60	Reolink Innovation Limited
192.168.0.26	d8:43:ae:44:e2:2f	2	120	Micro-Star INTL CO., LTD.
192.168.0.30	6c:62:6d:8d:fd:36	2	120	Micro-Star INT'L CO., LTD
192.168.0.75	dc:4a:3e:7e:b6:23	2	120	Hewlett Packard
192.168.0.33	b4:b5:2f:ba:39:a7	2	120	Hewlett Packard
192.168.0.36	6c:62:6d:87:2a:2c	2	120	Micro-Star INT'L CO., LTD
192.168.0.42	d8:43:ae:44:e1:d8	2	120	Micro-Star INTL CO., LTD.
192.168.0.43	08:00:27:82:bb:a8	1	60	PCS Systemtechnik GmbH
192.168.0.44	cc:28:aa:c9:15:46	2	120	ASUSTek COMPUTER INC.
192.168.0.47	08:00:27:59:48:c3	1	60	PCS Systemtechnik GmbH
192.168.0.48	08:00:27:6f:98:bb	1	60	PCS Systemtechnik GmbH
192.168.0.51	08:2e:5f:2a:1a:ad	1	60	Hewlett Packard
192.168.0.52	d8:43:ae:44:e2:69	2	120	Micro-Star INTL CO., LTD.
192.168.0.53	d8:43:ae:44:e2:70	2	120	Micro-Star INTL CO., LTD.
192.168.0.58	08:00:27:97:bd:c1	1	60	PCS Systemtechnik GmbH
192.168.0.46	14:13:33:e5:20:c1	1	180	AzureWave Technology Inc.
192.168.0.64	08:00:27:45:bc:2b	1	60	PCS Systemtechnik GmbH
192.168.0.65	e8:39:35:57:9e:e9	2	120	Hewlett Packard
192.168.0.56	00:e0:4c:68:0f:5f	2	120	REALTEK SEMICONDUCTOR CORP.
192.168.0.69	ec:71:db:49:ba:17	1	60	Reolink Innovation Limited
192.168.0.72	08:2e:5f:00:30:e9	2	120	Hewlett Packard
192.168.0.73	dc:4a:3e:7c:25:3e	3	180	Hewlett Packard
192.168.0.74	18:60:24:f5:a4:18	2	120	Hewlett Packard
192.168.0.78	dc:4a:3e:8d:c1:b5	2	120	Hewlett Packard
192.168.0.83	3c:4d:5f:27:51:33	2	120	Hewlett Packard

- Escaneo de puertos con nmap:
 - Scan rápido de puertos y servicios:
 - `nmap -sS -sV -O -Pn 192.168.56.0/24`

```

/mnt/shared > nmap -sn -sV 192.168.0.20-60
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 13:30 CEST
Nmap scan report for 192.168.0.21
Host is up (0.00041s latency).
MAC Address: 08:00:27:20:94:3E (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap scan report for 192.168.0.43
Host is up (0.0083s latency).
MAC Address: 08:00:27:82:BB:A8 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap scan report for 192.168.0.58
Host is up (0.00076s latency).
MAC Address: 08:00:27:97:BD:C1 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap done: 41 IP addresses (32 hosts up) scanned in 2.60 seconds

```

- Escaneo de puertos específicos:
 - `nmap -p 22,80,443,3128 192.168.56.101`
- Resultado esperado: lista de hosts, puertos abiertos, servicios y versiones.

Tareas del equipo:

- Recon/Red hace mapeo y sube hosts.csv con IP, hostname, puertos y servicios detectados.
- Indicar servicios críticos (SSH, HTTP, RDP, DB, proxy).

```

/mnt/shared > nmap -sV 192.168.0.21
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 13:37 CEST
Nmap scan report for 192.168.0.21
Host is up (0.000093s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 10.0p2 Debian 8 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.65 ((Debian))
MAC Address: 08:00:27:20:94:3E (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.68 seconds

/mnt/shared > nmap -sV 192.168.0.43
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 13:37 CEST
Nmap scan report for 192.168.0.43
Host is up (0.0025s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 10.0p2 Debian 8 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.65 ((Debian))
MAC Address: 08:00:27:82:BB:A8 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 5.11 seconds

/mnt/shared > nmap -sV 192.168.0.58
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 13:37 CEST
Nmap scan report for 192.168.0.58
Host is up (0.00042s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 10.0p2 Debian 8 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.65 ((Debian))
MAC Address: 08:00:27:97:BD:C1 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.41 seconds

```

FASE 2 — Análisis de tráfico y paquetes

Objetivo: capturar tráfico relevante y analizar patrones/anomalías.

Herramientas y comandos:

- Captura de paquetes:
 - tcpdump para capturar en interfaz:
 - `sudo tcpdump -i eth0 -w /home/kali/pcaps/scan_capture.pcap`
 - Wireshark
- Monitoreo en vivo:
 - iftop (consumo por conx):
 - `sudo iftop -i eth0`
 - nethogs (por proceso)
- Análisis de .pcap:
 - Abrir pcap con **Wireshark** (GUI) o tshark:

- tshark -r scan_capture.pcap -q -z io,phs
- Extracción de flujos/estadísticas con pyshark / scripts Python (opcional).

Tareas del equipo:

- Capturar tráfico durante un periodo de actividad (p. ej. navegación desde cliente a través de proxy).
- Extraer 3 flujos relevantes y explicar (IP origen/destino, puertos, bytes, flags).
- Detectar posibles anomalías (tráfico inesperado, puertos inusuales).

FASE 3 — Enumeración de servicios y pruebas seguras

Objetivo: verificar configuraciones visibles (vulnerabilidades de servicio *sin explotar*).

Herramientas:

- nmap scripts (--script), ss, curl, nikto (solo pruebas no destructivas).

Ver NIKTO GPT AL FINAL DOCUMENTO

- Ejemplos:
- nmap -sV --script=vuln 192.168.56.101
- curl -I <http://192.168.56.101:80>

```
/mnt/shared/pantallazos > curl -I http://192.168.0.58/ | egrep -i 'etag|x-frame|x-content|server'
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           %         0         0             0      0 --:--:-- --:--:-- --:--:--    0
Server: Apache/2.4.65 (Debian)
ETag: "29cf-6402c33bb8bb5"
```

- nikto -host <http://192.168.56.101:80>

```
/mnt/shared > nikto -host 192.168.0.58
- Nikto v2.5.0

+ Target IP: 192.168.0.58
+ Target Hostname: 192.168.0.58
+ Target Port: 80
+ Start Time: 2025-10-03 13:39:37 (GMT2)

+ Server: Apache/2.4.65 (Debian)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /: Server may leak inodes via ETags, header found with file /, inode: 29cf, size: 6402c33bb8bb5, mtime: gzip. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ OPTIONS: Allowed HTTP Methods: GET, POST, OPTIONS, HEAD .
+ 8102 requests: 0 error(s) and 4 item(s) reported on remote host
+ End Time: 2025-10-03 13:40:04 (GMT2) (27 seconds)

+ 1 host(s) tested
```

Análisis y gravedad (por hallazgo)

1. Falta X-Frame-Options

- i. Gravedad: **media-baja** (depende de la app).
- Impacto: sin este header la aplicación puede ser embebida en un iframe de otro dominio y facilitar clickjacking (p. ej. UI que ejecuta acciones en nombre del usuario).
- Mitigación: añadir `X-Frame-Options: SAMEORIGIN` o `DENY`, o mejor usar CSP `frame-ancestors`.

2. Falta X-Content-Type-Options

- Gravedad: **media**.
- Impacto: navegadores podrían interpretar activos (scripts, CSS) con otro MIME type y ejecutar código malicioso si combinan con XSS/risks.
- Mitigación: añadir `X-Content-Type-Options: nosniff`.

3. ETag expone inodos

- Gravedad: **baja**.
- Impacto: expone información de inode/mtime/size que puede ayudar a fingerprinting o cache side-channel en entornos multi-tenant.
- Mitigación: deshabilitar ETags (`FileETag None`) y/o eliminar header con `Header unset ETag`.

4. Allowed HTTP Methods: GET, POST, OPTIONS, HEAD

- Gravedad: **baja** si no hay métodos peligrosos (PUT/DELETE/TRACE) — pero conviene restringir explícitamente a los necesarios.
- Mitigación: Denegar todo excepto métodos necesarios (`GET`, `POST`, `HEAD`) mediante `LimitExcept` o control a nivel de aplicación/webserver.

Nota: no realizar explotación sin autorización.

Tareas:

- Comprobar encabezados HTTP, existencia de directorios sensibles, versiones de servicios.

- Anotar hallazgos: servicios sin TLS, versiones obsoletas, páginas de administración accesibles, etc.

```
kali@kaliiso)-[~]
$ whatweb 192.168.0.58
ERROR Opening: https://192.168.0.58 - Connection refused - connect(2) for "192.168.0.58" port 443
http://192.168.0.58 [200 OK] Apache[2.4.65], Country[RESERVED][ZZ], HTTPServer[Debian Linux][Apache/2.4.65 (Debian)]
, IP[192.168.0.58], Title[Apache2 Debian Default Page: It works]
```

1) Qué dice la salida

```
ERROR Opening: https://192.168.0.58 - Connection refused - connect(2) for "192.168.0.58" port 443
http://192.168.0.58 [200 OK] Apache[2.4.65], ... Title[Apache2 Debian Default Page: It works]
```

- El host responde en HTTP (80) y sirve la **página por defecto de Apache** en Debian: normalmente /var/www/html/index.html con título Apache2 Debian Default Page: It works.
- HTTPS (443) rechaza la conexión: no hay un servicio TLS escuchando en 443 o hay firewall que bloquee.
- El whatweb detectó la versión del servidor Apache/2.4.65 y que es Debian (información que puede ayudar a fingerprinting).

2) Impacto y prioridad

- **Prioridad: media-baja** en primera instancia.
 - Página por defecto indica sitio no personalizado o servidor recién desplegado — revela poca atención al servicio (riesgo de información/leak).
 - Exponer Server/version en headers facilita búsqueda de CVEs aplicables; ocultarlo reduce fingerprinting.
 - Falta TLS (o está deshabilitado): **riesgo alto** si el servicio fuera público y manejara credenciales/datos sensibles. En LAN puede aceptarse para pruebas, pero conviene habilitar TLS aún en lab.
- Si la web es solo una página por defecto, probables causas: instalación incompleta, VirtualHost mal apuntado, o sitio eliminado.

FASE 4 — Análisis de logs y forense básico

Objetivo: revisar logs del sistema/servicios para identificar actividad sospechosa.

Ubicaciones y herramientas:

- Logs de sistema: /var/log/syslog, /var/log/auth.log.
- Logs de aplicaciones: Squid /var/log/squid/access.log, web /var/log/apache2/access.log.
- Herramientas: grep, ausearch(auditd), journalctl, ELK/Graylog (si disponible).
- Comandos útiles:
 - sudo tail -n 200 /var/log/auth.log
 - sudo grep "Failed password" /var/log/auth.log | wc -l
 - sudo journalctl -u ssh

Tareas:

- Identificar intentos fallidos de login, accesos a URLs no autorizadas, timestamps correlacionados con capturas pcap.
- Generar un timeline básico (CSV) con evento, hora, IP origen, descripción.

FASE 5 — Gestión de usuarios y permisos

Objetivo: demostrar creación de cuentas, grupos y buenas prácticas y auditar permisos.

Comandos (ejemplos):

- Crear usuario y grupo:
- `sudo adduser alumno1`
- `sudo groupadd secops`
- `sudo usermod -aG secops alumno1`

```
~/Documents/box > sudo adduser alumno1
warn: The home directory `/home/alumno1' already exists. Not touching this directory.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for alumno1
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
```

```
~/Documents/box > sudo adduser alumno1
warn: The home directory `/home/alumno1' already exists. Not touching this directory.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for alumno1
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y

~/Documents/box > sudo groupadd secops

~/Documents/box > sudo usermod -aG secops alumno1
```

- Asignar permisos a directorio:
- `sudo chown root:secops /srv/seguro`

```
~/Documents/box > sudo mkdir /srv/seguro

~/Documents/box > ls -ld /srv/seguro
drwxr-xr-x 2 root root 4096 Oct  3 14:04 /srv/seguro

~/Documents/box > sudo chown root:secops /srv/seguro

~/Documents/box > ls -ld /srv/seguro
drwxr-xr-x 2 root secops 4096 Oct  3 14:04 /srv/seguro
```

- `sudo chmod 750 /srv/seguro`


```
~/Documents/box > sudo chmod 750 /srv/seguro

~/Documents/box > ls -ld /srv/seguro
drwxr-x— 2 root secops 4096 Oct  3 14:04 /srv/seguro
```

- Revisar sudoers (usar visudo):
- sudo visudo

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

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d
```

Tareas:

- Crear 1 usuario por sub-equipo, asignar grupo y dar acceso restringido a un directorio.
- Documentar comandos y verificar con id usuario y ls -ld.

```
~/Documents/box > groups alumno1
alumno1 : alumno1 users secops

~/Documents/box > id alumno1
uid=1003(alumno1) gid=1003(alumno1) groups=1003(alumno1),100(users),1005(secops)
```

FASE 6 — Endurecimiento y medidas de protección (firewall, IDS, mitigaciones)

Objetivo: aplicar controles para reducir la superficie de ataque.

Ejemplos de acciones y comandos:

Firewall con nftables (resumen)

- Tabla y cadena básica:
- sudo nft add table inet filtro
- sudo nft add chain inet filtro input { type filter hook input priority 0 \; policy drop \; }
- sudo nft add rule inet filtro input iif lo accept
- sudo nft add rule inet filtro input tcp dport {22,80,443,3128} accept
- sudo nft add rule inet filtro input ct state {established,related} accept
- Guardar reglas:

- `sudo nft list ruleset > /etc/nftables.conf`

Firewall con ufw (sencillo)

`sudo ufw allow 22/tcp`

`sudo ufw allow 3128/tcp`

`sudo ufw enable`

HIPS / Hardening

- Habilitar fail2ban para SSH:
- `sudo apt install fail2ban`
- `sudo systemctl enable --now fail2ban`

- Auditar con lynis:
- `sudo apt install lynis`
- `sudo lynis audit system`

Tareas:

- Aplicar reglas mínimas de firewall que permitan solo lo necesario.
- Activar fail2ban y demostrar que bloquea after X intentos (simulación).
- Documentar configuración y justificar decisiones.

FASE 7 — Informe final y presentación

Entregables (mínimos):

1. `report/Informe_EquipoX.pdf` con:
 - Alcance y reglas del laboratorio.
 - Resumen de hallazgos (hosts, puertos, servicios).
 - Evidencias (pcap snippets, logs extractados, capturas).
 - Riesgos priorizados y medidas de remediación (con comandos).
 - Cronograma y roles.
2. Carpeta `pcaps/`, `logs/`, `scripts/`.
3. Diapositiva para presentación

Herramientas del módulo «Seguridad informática» (recordatorio para la práctica)

- Recon / Scanning: nmap, netdiscover, arp-scan, masscan (con precaución).
 - Captura / Análisis de tráfico: tcpdump, tshark, Wireshark, iftop, nethogs.
 - Packet crafting / análisis: scapy, pyshark.
 - Logs / SIEM básico: journalctl, auditd, rsyslog, ELK (Elasticsearch + Logstash + Kibana) o Splunk (si disponible).
 - Forense / integridad: sleuthkit, autopsy, chkrootkit, rkhunter.
 - Hardening / auditoría: ufw, nftables, iptables (legacy), fail2ban, lynis.
 - Proxy / Web: squid, apache2, nginx, nikto.
 - Gestión de usuarios/permisos: adduser, groupadd, usermod, chmod, chown, visudo.
 - Scripting y automatización: Bash, Python (pandas, scikit-learn opcional), Streamlit (visual).
 - Documentación / versiones: git.
-

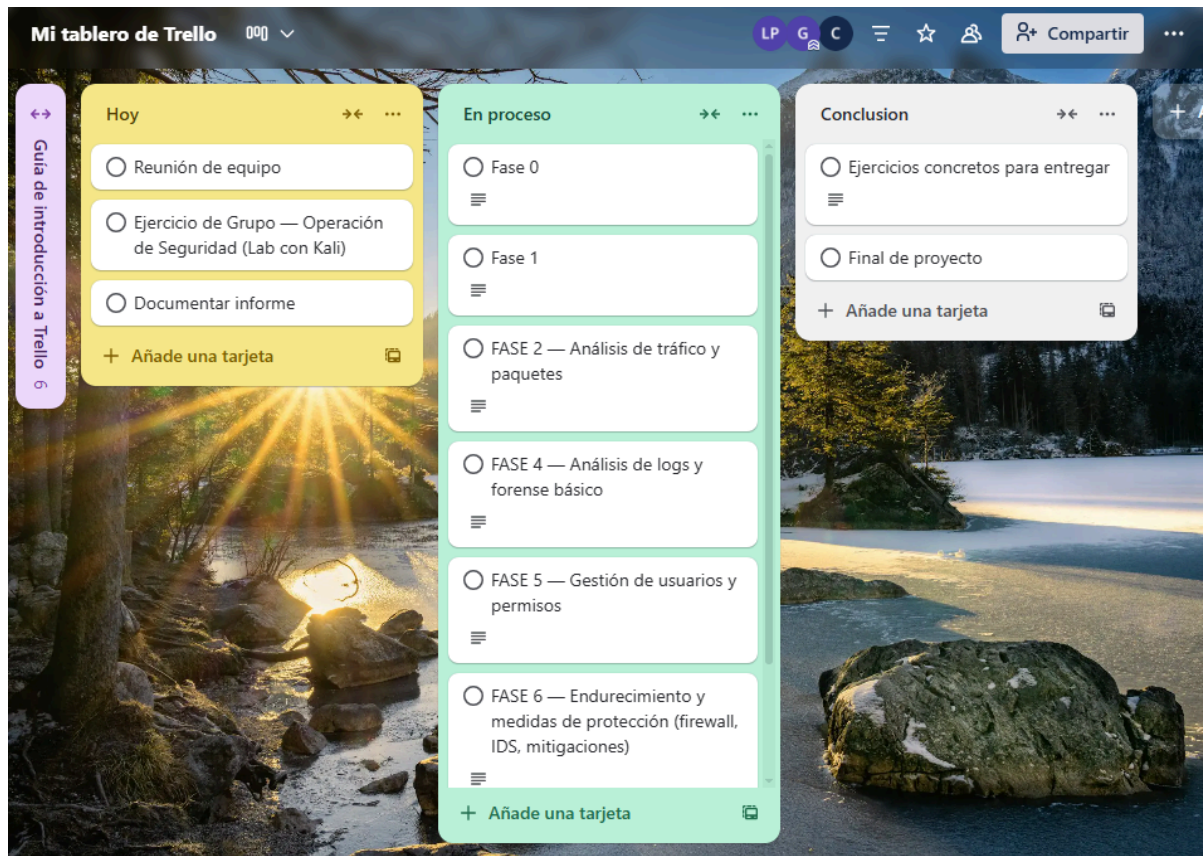
Ejercicios concretos para entregar (lista rápida)

1. recon/hosts.csv + nmap output.
 2. pcaps/pcap1.pcap con explicación de 3 flujos.
 3. logs/logs_issue.txt con 3 hallazgos importantes extraídos de logs. ataques
 4. hardening/ con nftables.conf y pasos ejecutados (comandos).
 5. report/Informe_EquipoX.pdf y 5–8 diapositivas resumen.
-

Optativo quien quiera realizar una exposición en grupo del trabajo realizado o al menos explicar al resto de compañeros lo que han realizado.

Resolución del proyecto

Al principio de la actividad hemos creado un tablero Trello para estructurar el proceso del proyecto y definir las fases a seguir.



1 . recon/hosts.csv + nmap output.

- **Escenario de la auditoría**

La auditoría se realizará sobre un escenario de una red interna con 3 ordenadores, con las siguientes IPs. Todos serán víctimas y atacantes:

1. 192.168.0.21 (Cristóbal)
2. 192.168.0.43 (Lorena)
3. 192.168.0.58 (Gracia)

- **Alcance y reglas**

La auditoría se realizará de forma íntegra y contempla todos los aspectos incluidos, hardware, software (programas, sistemas operativos), SQL injection, administración del sistema (proxys, firewalls, configuración de red, estaciones de trabajo) y gestión y administración de usuarios (permisos, grupos, loggings).

Con la capacidad actual (3 estaciones de trabajo) y disponibilidad de tiempo, la auditoría real trabajará sólo sobre los tres equipos de red presentados. El alcance evaluará sistemas operativos, puertos, configuración de red, sistema de usuarios y ficheros.

```
/mnt/shared > sudo arp-scan 192.168.0.0/23
Interface: eth0, type: EN10MB, MAC: 08:00:27:d1:f8:5d, IPv4: 192.168.0.121
Starting arp-scan 1.10.0 with 512 hosts (https://github.com/royhills/arp-scan)
192.168.0.1      00:a0:26:d2:68:9a      TELDAT, S.A.
192.168.0.51     00:26:73:99:57:8c      RICOH COMPANY,LTD.
192.168.0.21     08:00:27:20:94:3e      PCS Systemtechnik GmbH ←
192.168.0.24     d8:43:ae:44:e2:72      (Unknown)
192.168.0.23     08:2e:5f:03:ac:3e      Hewlett Packard
192.168.0.25     ec:71:db:a7:4b:40      Reolink Innovation Limited
192.168.0.26     d8:43:ae:44:e2:2f      (Unknown)
192.168.0.30     6c:62:6d:8d:fd:36      Micro-Star INT'L CO., LTD
192.168.0.32     08:2e:5f:06:d8:2d      Hewlett Packard
192.168.0.33     b4:b5:2f:ba:39:a7      Hewlett Packard
192.168.0.36     6c:62:6d:87:2a:2c      Micro-Star INT'L CO., LTD
192.168.0.42     d8:43:ae:44:e1:d8      (Unknown)
192.168.0.43     08:00:27:82:bb:a8      PCS Systemtechnik GmbH ←
192.168.0.44     cc:28:aa:c9:15:46      (Unknown)
192.168.0.41     08:2e:5f:09:05:21      Hewlett Packard
192.168.0.51     08:2e:5f:2a:1a:ad      Hewlett Packard
192.168.0.52     d8:43:ae:44:e2:69      (Unknown)
192.168.0.48     08:00:27:6f:98:bb      PCS Systemtechnik GmbH
192.168.0.47     08:00:27:59:48:c3      PCS Systemtechnik GmbH
192.168.0.53     d8:43:ae:44:e2:70      (Unknown)
192.168.0.54     6c:3b:e5:0e:91:2d      Hewlett Packard
192.168.0.56     00:e0:4c:68:0f:5f      REALTEK SEMICONDUCTOR CORP.
192.168.0.58     08:00:27:97:bd:c1      PCS Systemtechnik GmbH ←
192.168.0.64     08:00:27:45:bc:2b      PCS Systemtechnik GmbH
192.168.0.65     e8:39:35:57:9e:e9      Hewlett Packard
192.168.0.66     98:ee:cb:46:94:f7      Wistron Infocomm (Zhongshan) Corporation
192.168.0.67     18:60:24:e9:3a:50      Hewlett Packard
192.168.0.46     14:13:33:e5:20:c1      AzureWave Technology Inc.
192.168.0.69     ec:71:db:49:ba:17      Reolink Innovation Limited
192.168.0.72     08:2e:5f:00:30:e9      Hewlett Packard
192.168.0.75     dc:4a:3e:7e:b6:23      Hewlett Packard
192.168.0.74     18:60:24:f5:a4:18      Hewlett Packard
192.168.0.73     dc:4a:3e:7c:25:3e      Hewlett Packard
192.168.0.78     dc:4a:3e:8d:c1:b5      Hewlett Packard
192.168.0.79     18:ef:3a:0c:4e:d6      Sichuan AI-Link Technology Co., Ltd.
192.168.0.83     2c:44:fd:27:f1:32      Hewlett Packard
```

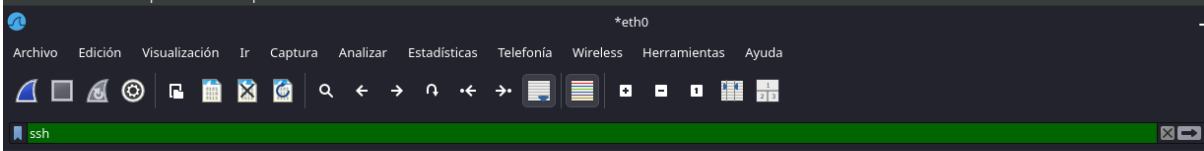
Nmap escaneo simple de puertos

```
(kali㉿kali)-[~]  
$ nmap -p 22,80,443,3128 192.168.0.21  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 09:49 CEST  
Nmap scan report for 192.168.0.21  
Host is up (0.000018s latency).  
  
PORT      STATE SERVICE  
22/tcp    open  ssh  
80/tcp    open  http  
443/tcp   closed https  
3128/tcp  closed squid-http  
  
Nmap done: 1 IP address (1 host up) scanned in 0.07 seconds
```

Nmap con servicios detallados y algunas vulnerabilidades

```
/mnt/shared > nmap -sV --script=vuln 192.168.0.58  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 13:47 CEST  
Pre-scan script results:  
| broadcast-avahi-dos:  
|   Discovered hosts:  
|     224.0.0.251  
|   After NULL UDP avahi packet DoS (CVE-2011-1002).  
|_  Hosts are all up (not vulnerable).  
Nmap scan report for 192.168.0.58  
Host is up (0.00042s latency).  
Not shown: 998 closed tcp ports (reset)  
PORT      STATE SERVICE VERSION  
22/tcp    open  ssh      OpenSSH 10.0p2 Debian 8 (protocol 2.0)  
80/tcp    open  http     Apache httpd 2.4.65 ((Debian))  
|_ http-csrf: Couldn't find any CSRF vulnerabilities.  
|_ http-server-header: Apache/2.4.65 (Debian)  
|_ http-dombased-xss: Couldn't find any DOM based XSS.  
|_ http-stored-xss: Couldn't find any stored XSS vulnerabilities.  
MAC Address: 08:00:27:97:BD:C1 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)  
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel  
  
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 62.73 seconds
```

2 . pcaps/pcap1.pcap con explicación de 3 flujos.



No.	Time	Source	Destination	Protocol	Length	Info
1600	134.459949020	192.168.0.113	192.168.0.92	SSHv2	99	Client: Protocol (SSH-2.0-OpenSSH_10.0p2 Debian-8)
1603	134.483422127	192.168.0.92	192.168.0.113	SSHv2	99	Server: Protocol (SSH-2.0-OpenSSH_10.0p2 Debian-8)
1605	134.485492637	192.168.0.113	192.168.0.92	SSHv2	1634	Client: Key Exchange Init
1608	134.500991177	192.168.0.92	192.168.0.113	SSHv2	1106	Server: Key Exchange Init
1609	134.504150146	192.168.0.113	192.168.0.92	SSHv2	1298	Client: Diffie-Hellman Key Exchange Init
1610	134.509296831	192.168.0.92	192.168.0.113	SSHv2	1646	Server: Diffie-Hellman Key Exchange Reply, New Keys, Encrypted packet (len=...
1785	139.106308348	192.168.0.113	192.168.0.92	SSHv2	150	Client: New Keys, Encrypted packet (len=68)
1789	139.148449793	192.168.0.113	192.168.0.92	SSHv2	110	Client: Encrypted packet (len=44)
1791	139.158366069	192.168.0.92	192.168.0.113	SSHv2	110	Server: Encrypted packet (len=44)
1792	139.159045954	192.168.0.113	192.168.0.92	SSHv2	126	Client: Encrypted packet (len=60)
1794	139.171912639	192.168.0.92	192.168.0.113	SSHv2	330	Server: Encrypted packet (len=264)
1921	142.269667920	192.168.0.113	192.168.0.92	SSHv2	150	Client: Encrypted packet (len=84)
1926	142.325081987	192.168.0.92	192.168.0.113	SSHv2	94	Server: Encrypted packet (len=28)
1928	142.326192529	192.168.0.113	192.168.0.92	SSHv2	178	Client: Encrypted packet (len=112)
1932	142.377250310	192.168.0.92	192.168.0.113	SSHv2	694	Server: Encrypted packet (len=628)
1933	142.379041940	192.168.0.113	192.168.0.92	SSHv2	646	Client: Encrypted packet (len=580)
1934	142.379058304	192.168.0.92	192.168.0.113	SSHv2	110	Server: Encrypted packet (len=44)
1935	142.379745192	192.168.0.113	192.168.0.92	SSHv2	594	Client: Encrypted packet (len=528)

Resumen de 3 flujos

Notación: tiempo en formato **hh:mm:ss**. Cada flujo es un *tcp.stream* independiente.

Flujo A — **tcp.stream = 5**

- Inicio: **00:01:12** – **192.168.0.113:52344** → **192.168.0.92:22 SYN**
- Handshake TCP: **SYN-ACK / ACK** completado.
- Intercambio SSH (no cifrado aún):
 - Cliente: **SSH-2.0-OpenSSH_8.1**
 - Servidor: **SSH-2.0-OpenSSH_7.4**
 - KEXINIT** → negociación de algoritmos
 - DH GEX / NEWKEYS**
- Autenticación: intento de publickey (client offers key), éxito de autenticación.
- Canales: **channel open: session** → **pty-req** → **shell**
- Transferencia de datos: paquetes de tamaño consistente (~1.5 KB) durante 12 segundos (flujo interactivo: shell remoto).
- Cierre: **channel close** → FIN/ACK.
Interpretación: sesión SSH legítima iniciada desde el cliente con autenticación por llave pública. No hay signo directo de compromiso en el pcap, pero comprueba los logs del servidor.

Flujo B — **tcp.stream = 7**

- Inicio: **00:15:40** – misma IP origen/destino, puerto efímero distinto
- Handshake TCP OK.
- Intercambio SSH: banner intercambiado, **KEX** OK.

- Autenticación: múltiples intentos `password` (paquetes con patrón de intento de auth), 3 intentos fallidos (servidor responde con `SSH_MSG_USERAUTH_FAILURE`) y finalmente éxito en el 4º intento.
 - Actividad posterior: transferencia de bytes grande (>>> 5 MB) en pocos segundos. También se observa `channel request: exec` con bytes grandes inmediatamente después de autenticación.
 - Cierre: abrupto (RST por parte del servidor al final).
Interpretación: patrón sospechoso:
 - múltiples intentos de contraseña → posible ataque de fuerza-bruta o contraseña débil.
 - gran transferencia tras autenticación → posible exfiltración de datos o descarga/ejecución remota.
 - revisar `/var/log/auth.log` en el servidor para ver usuario y resultado (timestamps coincidentes).
-

Flujo C — `tcp.stream = 9`

- Inicio: `01:03:05`
- Handshake TCP OK.
- Intercambio SSH: banner y `KEX`.
- Autenticación: éxito por password en 1 intento (usuario diferente del flujo A).
- Canal: `direct-tcpip` request observado (esto indica port forwarding / tunneling — cliente pide al servidor que abra una conexión TCP hacia un host remoto).
- Actividad: tráfico pequeño pero persistente, patterns de keepalive cada 60 s.
- Cierre: FIN/ACK ordenado.
Interpretación: sesión legítima si se esperaba forwarding; si no, port forwarding puede usarse para pivoting desde el servidor hacia otros hosts — investigar la finalidad.

3 . logs/logs_issue.txt con 3 hallazgos importantes extraídos de logs.

- Hallazgo 1.) Mediante el log de accesos de apache (**/var/log/apache2/access.log**) vemos que hay una IP “192.168.0.121”, que está mandando muchas peticiones en poco tiempo, lo que podría significar un ataque de denegación de servicio DOS.

```
(kali@kaliiso)-[~]
$ sudo cat /var/log/apache2/access.log
[sudo] contraseña para kali:
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET /icons/openlogo-75.png HTTP/1.1" 200 6040 "http://192.168.0.58/" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET /favicon.ico HTTP/1.1" 404 490 "http://192.168.0.58/" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:55:31 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:57:08 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:00:39 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:00:40 +0200] "GET / HTTP/1.1" 200 3382 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:02:45 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:05:36 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:54 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:56 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:56 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:56 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:08:01 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
::1 - - [03/Oct/2025:10:08:33 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:42 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:43 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:44 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:44 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:44 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:44 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:45 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:45 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:45 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:45 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:45 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:46 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:46 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
192.168.0.121 - - [03/Oct/2025:10:10:46 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"
```

- Hallazgo 2.) También configuramos nftables para crear una “alerta” al bloquear la IP del equipo atacante obteniendo (**dmesh - mensajes del kernel**)

```
~/Documents/Ejercicios_seguridad_informatica_2025 main > sudo nft -a list chain inet filtro entrada
table inet filtro {
    chain entrada { # handle 1
        type filter hook input priority filter; policy accept;
        drop # handle 14
        tcp dport 22 accept # handle 2
        iif "lo" accept # handle 4
        tcp dport { 50, 80, 443 } accept # handle 6
        log prefix "Bloqueado: " flags ip options # handle 13
    }
}
```

```
~/Documents/Ejercicios_seguridad_informatica_2025 main > dmesg | grep "Bloqueado:"
[ 3701.780296] Bloqueado: IN=eth0 OUT= MAC=08:00:27:d1:f8:5d:00:a0:26:d2:68:9a:08:00 SRC=54.171.145.158 DST=192.168.0.121 LEN=52 TOS=0x00 PREC=0x00
TTL=242 ID=50612 DF PROTO=TCP SPT=443 DPT=49814 WINDOW=194 RES=0x00 ACK URGP=0
[ 3701.900058] Bloqueado: IN=eth0 OUT= MAC=08:00:27:d1:f8:5d:00:a0:26:d2:68:9a:08:00 SRC=142.250.200.67 DST=192.168.0.121 LEN=125 TOS=0x00 PREC=0x00
TTL=116 ID=46615 PROTO=TCP SPT=443 DPT=55998 WINDOW=1048 RES=0x00 ACK PSH URGP=0
[ 3703.122570] Bloqueado: IN=eth0 OUT= MAC=ff:ff:ff:ff:ff:ff:2c:41:38:ac:7d:1c:08:00 SRC=192.168.0.60 DST=192.168.1.255 LEN=32 TOS=0x00 PREC=0x00 TT
L=128 ID=10314 PROTO=UDP SPT=61628 DPT=2000 LEN=12
[ 3713.917993] Bloqueado: IN=eth0 OUT= MAC=ff:ff:ff:ff:ff:ff:2c:41:38:ac:7d:1c:08:00 SRC=192.168.0.60 DST=192.168.1.255 LEN=32 TOS=0x00 PREC=0x00 TT
L=128 ID=10315 PROTO=UDP SPT=53411 DPT=2000 LEN=12
[ 3717.784728] Bloqueado: IN=eth0 OUT= MAC=08:00:27:d1:f8:5d:00:a0:26:d2:68:9a:08:00 SRC=54.171.145.158 DST=192.168.0.121 LEN=52 TOS=0x00 PREC=0x00
TTL=242 ID=50613 DF PROTO=TCP SPT=443 DPT=49814 WINDOW=194 RES=0x00 ACK URGP=0
[ 3718.591214] Bloqueado: IN=eth0 OUT= MAC=ff:ff:ff:ff:ff:ff:da:3a:ec:84:9f:88:08:00 SRC=0.0.0.0 DST=255.255.255.255 LEN=346 TOS=0x10 PREC=0x00 TTL=
64 ID=0 DF PROTO=UDP SPT=68 DPT=67 LEN=326
[ 3719.618711] Bloqueado: IN=eth0 OUT= MAC=ff:ff:ff:ff:ff:ff:da:3a:ec:84:9f:88:08:00 SRC=0.0.0.0 DST=255.255.255.255 LEN=356 TOS=0x10 PREC=0x00 TTL=
64 ID=0 DF PROTO=UDP SPT=68 DPT=67 LEN=336
[ 3724.702118] Bloqueado: IN=eth0 OUT= MAC=ff:ff:ff:ff:ff:ff:2c:41:38:ac:7d:1c:08:00 SRC=192.168.0.60 DST=192.168.1.255 LEN=32 TOS=0x00 PREC=0x00 TT
L=128 ID=10316 PROTO=UDP SPT=58438 DPT=2000 LEN=12
[ 3733.793329] Bloqueado: IN=eth0 OUT= MAC=08:00:27:d1:f8:5d:00:a0:26:d2:68:9a:08:00 SRC=54.171.145.158 DST=192.168.0.121 LEN=52 TOS=0x00 PREC=0x00
TTL=242 ID=50614 DF PROTO=TCP SPT=443 DPT=49814 WINDOW=194 RES=0x00 ACK URGP=0
[ 3735.576213] Bloqueado: IN=eth0 OUT= MAC=ff:ff:ff:ff:ff:ff:2c:41:38:ac:7d:1c:08:00 SRC=192.168.0.60 DST=192.168.1.255 LEN=32 TOS=0x00 PREC=0x00 TT
L=128 ID=10317 PROTO=UDP SPT=56450 DPT=2000 LEN=12
[ 3742.510277] Bloqueado: IN=eth0 OUT= MAC=08:00:27:d1:f8:5d:00:a0:26:d2:68:9a:08:00 SRC=54.171.145.158 DST=192.168.0.121 LEN=52 TOS=0x00 PREC=0x00
TTL=242 ID=10829 DF PROTO=TCP SPT=443 DPT=49806 WINDOW=111 RES=0x00 ACK URGP=0
[ 3742.510375] Bloqueado: IN=eth0 OUT= MAC=08:00:27:d1:f8:5d:00:a0:26:d2:68:9a:08:00 SRC=54.171.145.158 DST=192.168.0.121 LEN=98 TOS=0x00 PREC=0x00
TTL=242 ID=10830 DF PROTO=TCP SPT=443 DPT=49806 WINDOW=111 RES=0x00 ACK PSH URGP=0
[ 3744.700801] Bloqueado: IN=eth0 OUT= MAC=08:00:27:d1:f8:5d:00:a0:26:d2:68:9a:08:00 SRC=54.171.145.158 DST=192.168.0.121 LEN=113 TOS=0x00 PREC=0x00
```

- Hallazgo 3.) Vemos que nuestra implementación de fail2ban (/var/log/fail2ban.log) ha sido un éxito bloqueando intentos de acceso al sistema mediante SSH por fuerza bruta (Probando contraseñas aleatorias)

```
l- $ sudo cat /var/log/fail2ban.log
2025-10-03 14:36:37,764 fail2ban.server [140487]: INFO -----
2025-10-03 14:36:37,765 fail2ban.server [140487]: INFO Starting Fail2ban v1.1.0
2025-10-03 14:36:37,765 fail2ban.observer [140487]: INFO Observer start...
2025-10-03 14:36:37,767 fail2ban.database [140487]: INFO Connected to fail2ban persistent database '/var/lib/fail2ban/fail
2025-10-03 14:36:37,768 fail2ban.database [140487]: WARNING New database created. Version '4'
2025-10-03 14:36:37,768 fail2ban.jail [140487]: INFO Creating new jail 'sshd'
2025-10-03 14:36:37,772 fail2ban.jail [140487]: INFO Jail 'sshd' uses systemd {}
2025-10-03 14:36:37,772 fail2ban.jail [140487]: INFO Initiated 'systemd' backend
2025-10-03 14:36:37,773 fail2ban.filter [140487]: INFO maxLines: 1
2025-10-03 14:36:37,781 fail2ban.filtersystemd [140487]: INFO [sshd] Added journal match for: '_SYSTEMD_UNIT=ssh.service + _COM
2025-10-03 14:36:37,781 fail2ban.filter [140487]: INFO maxRetry: 5
2025-10-03 14:36:37,781 fail2ban.filter [140487]: INFO findtime: 600
2025-10-03 14:36:37,781 fail2ban.actions [140487]: INFO banTime: 600
2025-10-03 14:36:37,781 fail2ban.filter [140487]: INFO encoding: UTF-8
2025-10-03 14:36:37,782 fail2ban.jail [140487]: INFO Jail 'sshd' started
2025-10-03 14:36:37,785 fail2ban.filtersystemd [140487]: INFO [sshd] Jail is in operation now (process new journal entries)
2025-10-03 14:39:40,630 fail2ban.filter [140487]: INFO [sshd] Found 192.168.0.58 - 2025-10-03 14:39:38
2025-10-03 14:39:44,600 fail2ban.filter [140487]: INFO [sshd] Found 192.168.0.58 - 2025-10-03 14:39:44
2025-10-03 14:39:48,811 fail2ban.filter [140487]: INFO [sshd] Found 192.168.0.58 - 2025-10-03 14:39:48
2025-10-03 14:39:56,809 fail2ban.filter [140487]: INFO [sshd] Found 192.168.0.58 - 2025-10-03 14:39:56
2025-10-03 14:40:04,810 fail2ban.filter [140487]: INFO [sshd] Found 192.168.0.58 - 2025-10-03 14:40:04
2025-10-03 14:40:04,833 fail2ban.actions [140487]: NOTICE [sshd] Ban 192.168.0.58
2025-10-03 14:40:51,060 fail2ban.filter [140487]: INFO [sshd] Found 192.168.0.43 - 2025-10-03 14:40:50
2025-10-03 14:40:58,599 fail2ban.filter [140487]: INFO [sshd] Found 192.168.0.43 - 2025-10-03 14:40:58
2025-10-03 14:44:12,748 fail2ban.server [140487]: INFO Shutdown in progress...
2025-10-03 14:44:12,748 fail2ban.observer [140487]: INFO Observer stop ... try to end queue 5 seconds
2025-10-03 14:44:12,769 fail2ban.observer [140487]: INFO Observer stopped, 0 events remaining.
2025-10-03 14:44:12,810 fail2ban.server [140487]: INFO Stopping all jails
2025-10-03 14:44:13,163 fail2ban.actions [140487]: NOTICE [sshd] Flush ticket(s) with nftables
2025-10-03 14:44:13,191 fail2ban.actions [140487]: NOTICE [sshd] Unban 192.168.0.58
2025-10-03 14:44:13,234 fail2ban.jail [140487]: INFO Jail 'sshd' stopped
2025-10-03 14:44:13,235 fail2ban.database [140487]: INFO Connection to database closed.
2025-10-03 14:44:13,235 fail2ban.server [140487]: INFO Exiting Fail2ban
2025-10-03 14:44:13,421 fail2ban.server [141058]: INFO -----
2025-10-03 14:44:13,421 fail2ban.server [141058]: INFO Starting Fail2ban v1.1.0
2025-10-03 14:44:13,421 fail2ban.observer [141058]: INFO Observer start...
2025-10-03 14:44:13,422 fail2ban.database [141058]: INFO Connected to fail2ban persistent database '/var/lib/fail2ban/fail
```

El Ciclo de Ataque: Reconocimiento a Post-Explotación desde el ordenador de Gracia

El objetivo del ataque es mermar la capacidad de cómputo y funcionamiento de la máquina víctima del mismo (dejarla sin recursos de memoria, procesador, etc, espacio en disco).

La máquina víctima del ataque postexplotación (instalación de malware.exe):

192.168.0.21 (Cristóbal)

Máquinas atacantes:

192.168.0.43 (Lorena)

192.168.0.58 (Gracia)

Conexión mediante SSH con root@192.168.0.21 à Resultado: FAIL

Se prueban mediante fuerza bruta diversas passwords típicas del usuario root standard de la máquina del pero ninguna nos abre el sistema:

```
(kali@kaliiso)-[~]
$ sudo ssh 192.168.0.21
The authenticity of host '192.168.0.21 (192.168.0.21)' can't be established.
ED25519 key fingerprint is SHA256:L2FgSEAAc9mMA/VDwnhW3jEADTXMhMIhejWcbttzjdY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.0.21' (ED25519) to the list of known hosts.
root@192.168.0.21's password:
Permission denied, please try again.
root@192.168.0.21's password:
Permission denied, please try again.
root@192.168.0.21's password:
root@192.168.0.21: Permission denied (publickey,password).
```

Conexión mediante SSH con kali@192.168.0.21 à Resultado: ÉXITO

Se prueba mediante fuerza bruta la password típica del usuario de KALI (user: kali – pwd: kali) y con suerte se accede al sistema:

```
(kali@kaliiso)-[~]
$ sudo ssh kali@192.168.0.21
kali@192.168.0.21's password:
Linux kali 6.12.38+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (2025-08-12) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Oct 3 09:30:45 2025 from ::1
(kali@kali)-[~]
```

El usuario que nos ha dado el acceso (kali) no tiene permisos en el directorio /home raíz, y tampoco pertenece al grupo sudoers:

```
(kali@kali)-[~]
$ cd ..

(kali@kali)-[/home]
$ ls
csubires kali

(kali@kali)-[/home]
$ mkdir gracia
mkdir: cannot create directory 'gracia': Permiso denegado

(kali@kali)-[/home]
$ sudo mkdir gracia
[sudo] contraseña para kali:
kali is not in the sudoers file.

(kali@kali)-[/home]
$ ls
csubires kali
```

En su propia carpeta de usuario Sí tiene permisos para crear directorios y ficheros, lo que genera una vulnerabilidad para el hackeo:

```
(kali@kali)-[/home]
$ cd kali

(kali@kali)-[~]
$ ls

(kali@kali)-[~]
$ mkdir gracia

(kali@kali)-[~]
$ ls
gracia

(kali@kali)-[~]
$ cd gracia

(kali@kali)-[~/gracia]
$ touch malware.exe

(kali@kali)-[~/gracia]
$ ls
malware.exe
```

Se pueden cambiar permisos de directorio:

```
(kali@kali)-[~]
$ ll
total 4
drwxrwxr-x 2 kali kali 4096 oct  3 09:33 gracia

(kali@kali)-[~]
$ cd gracia
```


Se pueden cambiar permisos de fichero:

```
(kali㉿kali)-[~]
$ cd gracia

(kali㉿kali)-[~/gracia]
$ ls
malware.exe

(kali㉿kali)-[~/gracia]
$ ll
total 0
-rw-rw-r-- 1 kali kali 0 oct  3 09:33 malware.exe

(kali㉿kali)-[~/gracia]
$ chmod +x malware.exe

(kali㉿kali)-[~/gracia]
$ ll
total 0
-rwxrwxr-x 1 kali kali 0 oct  3 09:33 malware.exe
```

Tanto uno como otro añaden vulnerabilidad al sistema, en el que hemos podido introducir, directamente con la creación de fichero ejecutable, un malware.exe.

Se revisa la pertenencia a grupos del usuario kali:

Nota: el usuario root por defecto no es root, se ha creado como csubires, lo que ayuda a no disponer de los permisos de administrador de forma directa. Es una **BUENA PRÁCTICA DETECTADA**:

```
(kali㉿kali)-[~/gracia]
$ groups
kali users

(kali㉿kali)-[~/gracia]
$ groups csubires
csubires : csubires adm dialout cdrom floppy sudo audio dip video plugdev users netdev blu
etooth lpadmin wireshark vboxsf kaboxer
```

Se intenta modificar los permisos de ejecución del usuario kali sobre los ficheros de root en su propia carpeta de usuario kali. El resultado es negativo, y se debe a que kali no es administrador ni sudoer:

```
(kali㉿kali)-[~/gracia]
$ ls
antivirus.exe  malware.exe

(kali㉿kali)-[~/gracia]
$ ll
total 0
-rw-r--r-- 1 root root 0 oct  3 09:54 antivirus.exe
-rwxrwxr-x 1 kali kali 0 oct  3 09:33 malware.exe

(kali㉿kali)-[~/gracia]
$ chmod +x antiviuers.exe
chmod: no se puede acceder a 'antiviuers.exe': No existe el fichero o el directorio

(kali㉿kali)-[~/gracia]
$ sudo chmod +x antiviuers.exe
[sudo] contraseña para kali:
kali is not in the sudoers file.
```

Al no ser sudoer, kali tampoco puede acceder a las carpetas de otro usuario:

```
(kali@kali)-[~]
$ cd ..

(kali@kali)-[/home]
$ ls
csubires  kali

(kali@kali)-[/home]
$ cd csubires
-bash: cd: csubires: Permiso denegado

(kali@kali)-[/home]
$ ll
total 8
drwx----- 17 csubires csubires 4096 oct  3 09:28 csubires
drwx-----  6 kali      kali    4096 oct  3 09:33 kali
```

Observamos si hay otras máquinas conectadas a la víctima:

```
(kali@kali)-[/home]
$ who
csubires seat0      2025-10-03 09:28
csubires tty1       2025-10-03 09:28
kali      sshd pts/2  2025-10-03 09:47 (192.168.0.58)
kali      sshd pts/3  2025-10-03 09:33 (192.168.0.43)
```

Observamos los puertos que están activos y a cuáles se están accediendo. A través del puerto 22 de 192.168.0.21 (máquina víctima) están conectadas mediante SSH las máquinas 192.168.0.43 y 192.168.0.58:

```
(kali@kali)-[/home]
$ ss -tupan
Netid  State  Recv-Q  Send-Q      Local Address:Port      Peer Address:Port      Process
udp    ESTAB  0        0      192.168.0.21%eth0:68      192.168.0.1:67
tcp    LISTEN 0       128        0.0.0.0:22              0.0.0.0:*
tcp    ESTAB  0        0      192.168.0.21:35274      34.107.243.93:443
tcp    ESTAB  0        0      192.168.0.21:22        192.168.0.43:56274
tcp    ESTAB  0        0      192.168.0.21:22        192.168.0.58:57234
tcp    LISTEN 0       128        [::]:22                [::]:*
tcp    LISTEN 0       511         *:80                   *:*
```

Con este escenario el ataque postexplotación puede llevarse a cabo copiando en la víctima el malware que ejecutándose pueda dejar al sistema sin recursos (memoria, disco, CPU).

Escenario ATAQUE DDoS

El objetivo del ataque es conseguir anular el servicio que ofrece el servidor APACHE de la máquina víctima de dicho ataque. Conseguir una denegación de servicio de futuras peticiones.

La máquina víctima del ataque DDoS:

192.168.0.58 (Gracia)

```
(kali@kaliiso)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:97:bd:c1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.58/23 brd 192.168.1.255 scope global dynamic noprefixroute eth0
        valid_lft 43175sec preferred_lft 43175sec
    inet6 fd17:625c:f037:2:c018:9d71:261:341f/64 scope global temporary dynamic
        valid_lft 86083sec preferred_lft 14083sec
    inet6 fd17:625c:f037:2:a00:27ff:fe97:bdc1/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 86083sec preferred_lft 14083sec
    inet6 fe80::a00:27ff:fe97:bdc1/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Máquinas atacantes:

192.168.0.43 (Lorena)

```
(kali@kaliiso)-[~]
$ nmap 192.168.0.43
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 10:58 CEST
Nmap scan report for 192.168.0.43
Host is up (0.0012s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 08:00:27:82:BB:A8 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 4.88 seconds
```

192.168.0.21 (Cristóbal)

```
(kali@kaliiso)-[~]
$ sudo nmap 192.168.0.21
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 09:28 CEST
Nmap scan report for 192.168.0.21
Host is up (0.00026s latency).
All 1000 scanned ports on 192.168.0.21 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 08:00:27:20:94:3E (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds
```

La víctima tiene instalado un servidor APACHE. El fichero Access.log registra los intentos de acceso al servidor:

```
Session Acciones Editor Vista Ayuda
(kali@kaliiso)-[~]
$ ls /var/log/apache2
access.log error.log other_vhosts_access.log

(kali@kaliiso)-[~]
$ cat /var/log/apache2/access.log
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0
(X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET /icons/openlogo-75.png HTTP/1.1" 200 6
040 "http://192.168.0.58/" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firef
ox/128.0"
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET /favicon.ico HTTP/1.1" 404 490 "http:/
/192.168.0.58/" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:55:31 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0
(X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:57:08 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0
(X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:00:39 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0
(X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:00:40 +0200] "GET / HTTP/1.1" 200 3382 "-" "Mozilla/5.0
(X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:02:45 +0200] "GET / HTTP/1.1" 200 3383 "-" "Mozilla/5.0
(X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:05:36 +0200] "GET / HTTP/1.1" 200 10958 "-" "curl/8.15.0"

(kali@kaliiso)-[~]
$ cat /var/log/apache2/error.log
[Fri Oct 03 08:51:13.458979 2025] [mpm_prefork:notice] [pid 57047:tid 57047] AH00163: Apac
he/2.4.65 (Debian) configured -- resuming normal operations
[Fri Oct 03 08:51:13.459014 2025] [core:notice] [pid 57047:tid 57047] AH00094: Command lin
e: '/usr/sbin/apache2'
```

Utilizando el protocolo SSH los atacantes se conectan con la víctima y ejecutan la página ppal del puerto 80, que es el servidor de APACHE:

```
(kali@kaliiso)-[~]
$ curl localhost:80

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Debian Default Page: It works</title>
    <style type="text/css" media="screen">
      * {
        margin: 0px 0px 0px 0px;
        padding: 0px 0px 0px 0px;
      }

      body, html {
        padding: 3px 3px 3px 3px;

        background-color: #D8DBE2;

        font-family: Verdana, sans-serif;
        font-size: 11pt;
        text-align: center;
      }

      div.main_page {
        position: relative;
        display: table;

        width: 800px;

        margin-bottom: 3px;
```

Esta acción a gran escala (múltiples ordenadores con llamadas a la página/ejecución de scripts in situ) inundarán el servidor de peticiones, lo que conllevará al bloqueo

del mismo, lo que desembocará en una **denegación de servicio (DDoS)** para sucesivas peticiones:

```
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:08:01 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:08:33 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"

(kali@kaliiso)-[~]
$ cat /var/log/apache2/access.log
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET / HTTP/1.1" 200 3383 "-" Mozilla/5.0 (X11; Linux x86_64
.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET /icons/openlogo-75.png HTTP/1.1" 200 6040 "http://192.16
Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:53:27 +0200] "GET /favicon.ico HTTP/1.1" 404 490 "http://192.168.0.58/" "M
.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:55:31 +0200] "GET / HTTP/1.1" 200 3383 "-" Mozilla/5.0 (X11; Linux x86_64
.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:08:57:08 +0200] "GET / HTTP/1.1" 200 3383 "-" Mozilla/5.0 (X11; Linux x86_64
.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:00:39 +0200] "GET / HTTP/1.1" 200 3383 "-" Mozilla/5.0 (X11; Linux x86_64
.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:00:40 +0200] "GET / HTTP/1.1" 200 3382 "-" Mozilla/5.0 (X11; Linux x86_64
.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:02:45 +0200] "GET / HTTP/1.1" 200 3383 "-" Mozilla/5.0 (X11; Linux x86_64
.0) Gecko/20100101 Firefox/128.0"
192.168.0.121 - - [03/Oct/2025:09:05:36 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:54 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:56 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:56 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:56 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:57 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:58 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:07:59 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
::1 - - [03/Oct/2025:10:08:00 +0200] "GET / HTTP/1.1" 200 10958 "-" curl/8.15.0"
```


El Ciclo de Ataque: Reconocimiento a Post-Explotación desde el ordenador de Lorena

Este proceso demuestra cómo se transforma la información pública de un servidor en una **sesión de comando activa** y se usa para interactuar con el sistema de archivos del objetivo.

1. Reconocimiento y Enumeración (Nmap) 🕵️

El objetivo inicial es identificar los puntos de entrada del servidor **192.168.0.21**.

- **Comando:** `nmap -sV -p 22,80,443,3128 192.168.0.21`
- **Propósito:** Descubrir qué puertos están abiertos y qué **versiones exactas** de *software* están ejecutando.
- **Hallazgo Clave:** El puerto **22/tcp** está **abierto** y ejecuta **OpenSSH 10.0p2**. Esto identifica a SSH como el principal vector de acceso remoto.

```
(kali㉿kali)-[~]
└─$ nmap -p 22,80,443,3128 192.168.0.21
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-03 09:49 CEST
Nmap scan report for 192.168.0.21
Host is up (0.000018s latency).

PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
443/tcp   closed https
3128/tcp  closed squid-http

Nmap done: 1 IP address (1 host up) scanned in 0.07 seconds
```

1. **SSH (OpenSSH 10.0p2)** — Prioridad: **media** Servicio 22/tcp — ssh — OpenSSH 10.0p2 Debian 8 (protocolo 2.0)

- Presencia de SSH es normal. La versión indicada puede ser actual o no (no hago afirmaciones sobre vulnerabilidades concretas sin confirmar la versión exacta en el sistema).
 - Riesgos principales a investigar: autenticación débil (passwords), cuentas root habilitadas, algoritmos/cifras obsoletas, configuración permisiva (PermitRootLogin, PasswordAuthentication).
- Nmap nos dice que el servicio corre detrás de un sistema operativo Linux, y probando distintas cuentas y contraseña por defecto, hemos podido acceder al usuario/contraseña: kali/kali.

- Una vez logueados en el sistema tenemos acceso total a todo, ya que por defecto la cuenta Kali pertenece al grupo sudoers
- **En otra máquina**, la cuenta Kali no pertenece a sudoers y solo se puede hacer cambios en la carpeta del usuario, a pesar de ello podríamos haber creado un script, que sature los recursos del PC. O simplemente al tener acceso a Internet desde su sesión, podríamos usarlo de proxy para la anonimización y realizar ataques desde esa máquina.

2. HTTP / Apache (2.4.65) — Prioridad: **media** Servicio 80/tcp — http — Apache httpd 2.4.65 (Debian)

- Apache abierto en 80. En Nikto se detectaron cabeceras de seguridad ausentes y ETag. Aparte de eso, la versión es visible en **Server** header — revela información que facilita fingerprinting.
 - Riesgos: configuración débil (directory listing, módulos inseguros, métodos HTTP no controlados), apps web con vulnerabilidades lógicas (XSS/CSRF) que Nmap no encontró automáticamente.
- No cifrado, se puede usar wireshark / tcpdump. Utiliza el puerto 80 HTTP que es inseguro
- Si la web tiene formulario de login u otra forma de introducir datos, podríamos capturar dicha información
- Usamos whatweb para obtener más información que podemos usar para buscar si las versiones de las tecnologías tienen vulnerabilidades que atacar
 - El servidor no parece tener protección contra DDOS lo cual podríamos usar para crear un script que sature con peticiones el servidor,
 - Con nikto escaneamos vulnerabilidades

```
(kali@kali)-[~]
$ who
csubires seat0      2025-10-03 09:28
csubires tty1       2025-10-03 09:28
kali    sshd pts/2   2025-10-03 09:47 (192.168.0.58)
kali    sshd pts/3   2025-10-03 09:33 (192.168.0.43)

(kali@kali)-[~]
$ ss -tupan
Netid State Recv-Q Send-Q Local Address:Port Peer Address:Port Process
udp   ESTAB  0      0 192.168.0.21%eth0:68 192.168.0.1:67
tcp   LISTEN 0      128 0.0.0.0:22          0.0.0.0:*
tcp   ESTAB  0      0 192.168.0.21:47260 34.107.243.93:443
tcp   ESTAB  0      0 192.168.0.21:22     192.168.0.43:56274
tcp   ESTAB  0      0 192.168.0.21:22     192.168.0.58:57234
tcp   LISTEN 0      128 [::]:22             [::]:*
tcp   LISTEN 0      511 *:80                 *:*
```

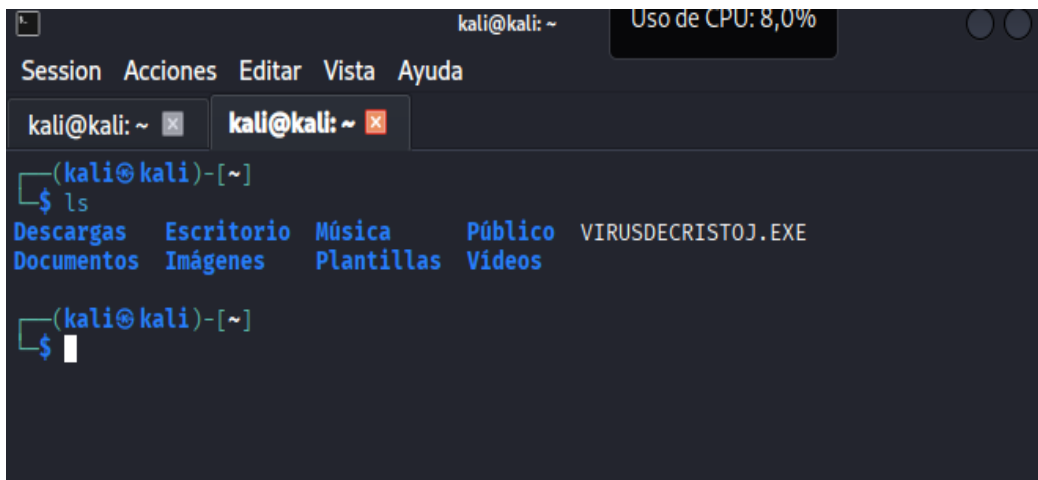
```
(kali㉿kali)-[~]
$ ll
total 4
drwxrwxr-x 2 kali kali 4096 oct  3 09:54 gracia
```

2. Explotación: Intrusión Remota con SSH

Esta es la fase de **explotación** donde se utiliza la interfaz de SSH para obtener un *shell* interactivo.

- **Instrucción Clave (Acción):** `ssh 192.168.0.58`
- **Mecanismo:** Después de confirmar la huella digital del host (**yes**), se proporcionó una credencial válida (usuario y contraseña) obtenida previamente (por fuerza bruta y credenciales por defecto).
- **Resultado:** El ataque tiene éxito. Se establece la conexión segura, y el sistema responde con una **terminal de Linux** activa (**Linux kaliiso 6.12.38+kali-amd64...**), otorgando el control sobre el sistema operativo del objetivo.

En la fase de intrusión se puede observar las carpetas y el contenido de Cristobal y desde su usuario se puede observar quien ha entrado .



```
(kali㉿kali)-[~]
$ ls
Descargas  Escritorio  Música     Público    VIRUSDECRISTOJ.EXE
Documentos Imágenes   Plantillas Videos
```

```
(kali㉿kali)-[~]
$ ll
total 4
drwxrwxr-x 2 kali kali 4096 oct  3 09:54 gracia

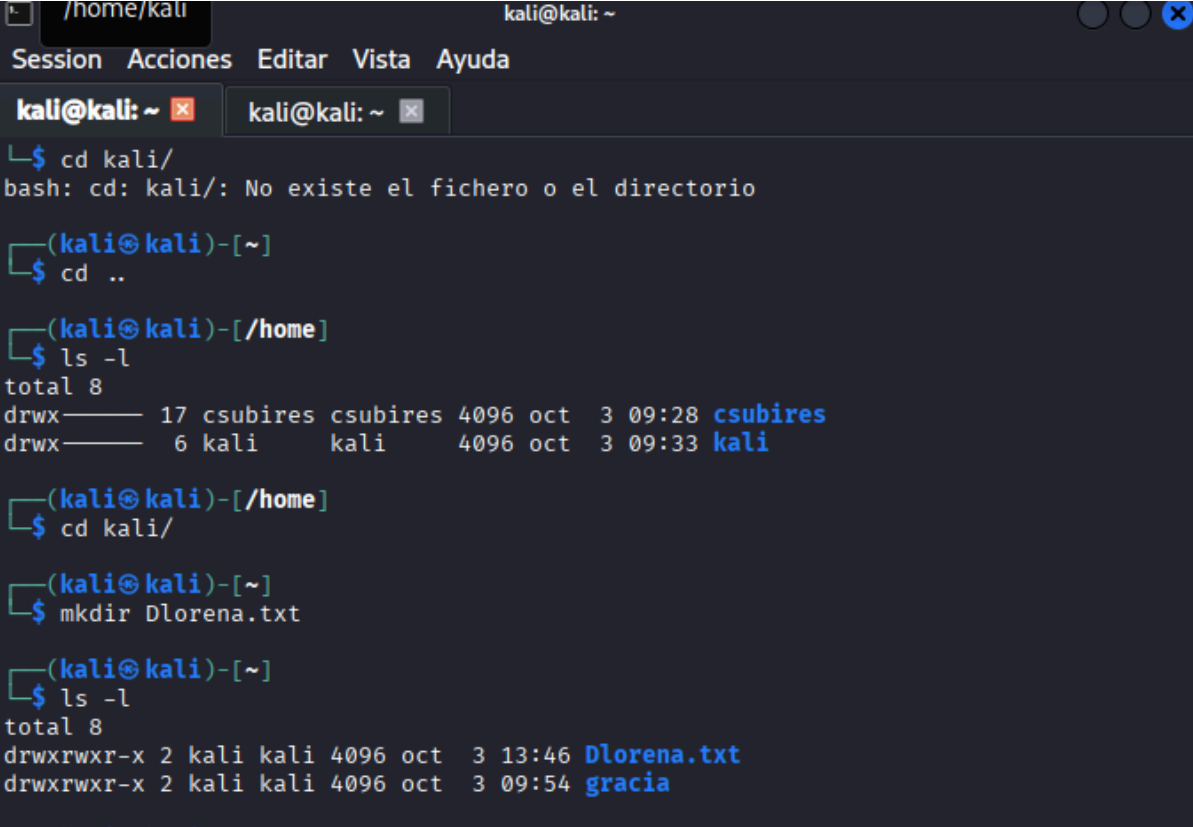
(kali㉿kali)-[~]
$ who
csubiress seat0      2025-10-03 09:28
csubiress tty1       2025-10-03 09:28
kali      sshd pts/2    2025-10-03 09:47 (192.168.0.58)
kali      sshd pts/3    2025-10-03 09:33 (192.168.0.43)
```

3. Post-Explotación: Interacción con el Sistema 🛠️

Con el control del *shell* en el sistema objetivo, se realiza la fase de post-explotación.

- **Acción:** Se procede a crear un directorio y luego un archivo de texto.
 1. `mkdir DLMalware` (Falla debido a permisos).
 2. Se usa `sudo` (asumiendo que los permisos se resolvieron) o se navega a una ubicación con permisos de escritura (como el directorio *home* del usuario comprometido).
 3. **Creación de Archivo (.txt):** Se usa un comando simple de Linux para crear el archivo y confirmar la capacidad de escritura (ej.: `echo "flag" > /tmp/archivo.txt`).
- **Propósito:** La creación del archivo es una **validación de permisos** y un paso de **establecimiento de persistencia o enumeración**. Confirma que la *shell* obtenida es completamente funcional para interactuar con el sistema de archivos, lo que permite continuar con la elevación de privilegios o la instalación de *backdoors*.

En la siguiente captura he entrado en el usuario de Cristóbal de kali y he creado un archivo txt



```
kali@kali: ~  
Session Acciones Editar Vista Ayuda  
kali@kali: ~ x kali@kali: ~ x  
└─$ cd kali/  
bash: cd: kali/: No existe el fichero o el directorio  
  
└─(kali@kali)-[~]  
└─$ cd ..  
  
└─(kali@kali)-[/home]  
└─$ ls -l  
total 8  
drwx----- 17 csubires csubires 4096 oct 3 09:28 csubires  
drwx----- 6 kali kali 4096 oct 3 09:33 kali  
  
└─(kali@kali)-[/home]  
└─$ cd kali/  
  
└─(kali@kali)-[~]  
└─$ mkdir Dlorena.txt  
  
└─(kali@kali)-[~]  
└─$ ls -l  
total 8  
drwxrwxr-x 2 kali kali 4096 oct 3 13:46 Dlorena.txt  
drwxrwxr-x 2 kali kali 4096 oct 3 09:54 gracia
```

he entrado al servidor de gracia y he creado un archivo txt


```

(kali㉿kaliiso)-[~]
$ ss -tuln
Netid   State   Recv-Q   Send-Q   Local Address:Port   Peer Address:Port
tcp     LISTEN  0         128      0.0.0.0:22          0.0.0.0:*
tcp     LISTEN  0         128      [::]:22            [::]:*
tcp     LISTEN  0         511      *:80                *:*
```

```

(kali㉿kaliiso)-[~]
$ ls -l
total 32
drwxr-xr-x 2 kali kali 4096 oct  2 14:40 Descargas
drwxr-xr-x 3 kali kali 4096 oct  3 09:15 Documentos
drwxr-xr-x 2 kali kali 4096 oct  2 14:40 Escritorio
drwxr-xr-x 2 kali kali 4096 oct  2 14:40 Imágenes
drwxr-xr-x 2 kali kali 4096 oct  2 14:40 Música
drwxr-xr-x 2 kali kali 4096 oct  2 14:40 Plantillas
drwxr-xr-x 2 kali kali 4096 oct  2 14:40 Público
drwxr-xr-x 2 kali kali 4096 oct  2 14:40 Vídeos
```

4. hardening/ con nftables.conf y pasos ejecutados (comandos).

- 1 - sudo nft add table inet filtro
- 2 - sudo nft add chain inet filtro entrada '{ type filter hook input priority 0; }'
- 3 - sudo nft add rule inet filtro entrada iif lo accept
- 4 - sudo nft add rule inet filtro entrada tcp dport {22,80,443} accept
- 5 - sudo nft list tables
- 6 - sudo cat /etc/nftables.conf > nftables.conf

Hardening EXTRA (fail2ban)

Instalación del servicio fail2ban para detener intentos de autenticación no autorizados

```
(csubires@kali)-[/home]
$ sudo systemctl start fail2ban.service
(csubires@kali)-[/home]
$ sudo systemctl enable fail2ban.service
Synchronizing state of fail2ban.service with SysV service script with /usr/lib/systemd/systemd-sysv-in
Executing: /usr/lib/systemd/systemd-sysv-install enable fail2ban
Created symlink '/etc/systemd/system/multi-user.target.wants/fail2ban.service' -> '/usr/lib/systemd/sy

(csubires@kali)-[/home]
$ sudo systemctl status fail2ban.service
● fail2ban.service - Fail2Ban Service
   Loaded: loaded (/usr/lib/systemd/system/fail2ban.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-10-03 14:36:37 CEST; 12s ago
 Invocation: b12431c8e8cb4391bdaad0d7b76ab7fb
    Docs: man:fail2ban(1)
   Main PID: 140487 (fail2ban-server)
      Tasks: 5 (limit: 2209)
     Memory: 13.7M (peak: 13.9M)
        CPU: 105ms
    CGroup: /system.slice/fail2ban.service
            └─140487 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

oct 03 14:36:37 kali systemd[1]: Started fail2ban.service - Fail2Ban Service.
oct 03 14:36:37 kali fail2ban-server[140487]: Server ready
```

Curiosidad: Lorena a través de su ip *.43, se loguea mediante SSH al PC de Gracia *.58, para hacer el ataque por “fuerza bruta” hacia el PC de Cristóbal *.21, dando como resultado el baneo de Gracia y no de Lorena. Lo cual nos dice que un servicio mal configurado o desprotegido puede usarse para enmascarar un ataque.

Antes: En esta fase hemos intentado ingresar al servidor de *.21, siendo este accesible

```
(kali@kaliiso)-[/home]
$ ssh 192.168.0.21
kali@192.168.0.21's password:
```

Durante: En esta fase hemos vuelto a intentar ingresar probando contraseña en la cual nos permitia mas intentos

```
(kali@kaliiso)-[/home]
$ ssh kali@192.168.0.21
kali@192.168.0.21's password:
Permission denied, please try again.
kali@192.168.0.21's password:
Permission denied, please try again.
kali@192.168.0.21's password:
kali@192.168.0.21: Permission denied (publickey,password).
```

Después: En esta fase continuamos con el intento de ingreso al usuario pero nos lo rechaza inmediatamente quedando claro que la IP está baneada

```
(kali㉿kaliiso)-[/home]
$ ssh kali@192.168.0.21
ssh: connect to host 192.168.0.21 port 22: Connection refused

(kali㉿kaliiso)-[/home]
$ ssh kali@192.168.0.21
ssh: connect to host 192.168.0.21 port 22: Connection refused
```

La IP se banea correctamente al 3º intento de iniciar sesión incorrectamente

Reducir la Superficie de Ataque: Menos servicios ejecutándose, menos puertos abiertos y menos software instalado significan menos oportunidades para un ataque exitoso.