

GUÍA IMPLEMENTACIÓN DE UN HONEYPOT



HONEYPOT

Realizado por:

Henrique Alves

Contenido

Introducción	3
Metodología.....	4
Instalación de Cowrie	4
Realización de los Ataques	5
1. Escaneo de Puertos con Nmap.....	5
2. Ataque de Fuerza Bruta SSH.....	6
3. Ataque fuerza bruta HTTP GET	7
4. Ataque por SSH.....	8
Conclusión de la Prueba de Ataques	11
Referencias	12

Introducción

El objetivo de este informe es presentar la configuración y pruebas realizadas sobre un honeypot **Cowrie** instalado en una máquina virtual con **Docker**, emulando servicios como SSH y HTTP. Se utilizó una infraestructura basada en **Kali Linux** como máquina atacante y **Ubuntu Server** con **Cowrie** como máquina objetivo. A través de herramientas como **Hydra** y **Nmap**, se realizaron ataques simulados para evaluar la capacidad del honeypot para registrar intentos de intrusión y su efectividad en un entorno controlado.

Paso	Descripción	Resultado
Configuración de Red	Se utilizaron máquinas virtuales en Red Interna (Kali Linux: 192.168.5.10, Ubuntu Server: 192.168.5.11).	La red interna permitió la comunicación entre Kali y Ubuntu.
Instalación de Cowrie	Cowrie se instaló en Ubuntu Server mediante Docker : <code>sudo docker run -p 2222:2222 cowrie/cowrie</code> .	Cowrie emuló un servicio SSH en el puerto 2222.
Ataque de Fuerza Bruta SSH	Se utilizó Hydra con la lista rockyou.txt para realizar un ataque de fuerza bruta sobre el puerto 2222.	Cowrie registró los intentos de acceso fallidos.
Escaneo de Puertos con Nmap	Nmap escaneó todos los puertos de Ubuntu Server .	Nmap identificó los puertos abiertos, incluyendo SSH (2222).
Análisis de Logs	Los logs generados por Cowrie registraron los intentos de acceso y comportamiento de los ataques simulados.	Cowrie registró correctamente los intentos de ataque SSH.

Metodología

1. Configuración de la Infraestructura

- **Máquina Atacante: Kali Linux** (IP: 192.168.x.x).
- **Máquina Objetivo: Ubuntu Server con Cowrie** (IP: 192.168.x.x).
- Ambas máquinas fueron configuradas en una **Red Interna en VirtualBox** para permitir la comunicación exclusiva entre ellas.

Instalación de Cowrie

Para la implementación del honeypot **Cowrie**, se utilizó **Docker**, que facilita la creación y ejecución del contenedor sin tener que preocuparse por las configuraciones del sistema operativo subyacente o las dependencias manuales. **Cowrie** es un honeypot que emula servicios como **SSH** y **Telnet**, con el objetivo de simular ataques cibernéticos y registrar los intentos de acceso maliciosos para su posterior análisis. Esta instalación fue realizada en una máquina virtual con **Ubuntu Server** como sistema operativo, mientras que la máquina atacante, **Kali Linux**, fue utilizada para ejecutar las pruebas de intrusión.

El primer paso fue ejecutar el siguiente comando en **Ubuntu Server** para descargar e iniciar el contenedor de **Cowrie**:

Instalación de Docker en Ubuntu server:

```
sudo snap install Docker
```

Ejecutar Honeypot Cowrie:

```
sudo docker run -p 2222:2222 cowrie/cowrie:latest
```

```
henrique@ubuntuserver:~$ sudo docker run -p 2222:2222 cowrie/cowrie:latest_
```

```
/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:105: CryptographyDeprecationWarning: TripleDES has been moved to cryptography.hazmat.decrepit.ciphers.algorithms.TripleDES and will be removed from cryptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
  b"3des-ctr": (algorithms.TripleDES, 24, modes.CTR)
/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:112: CryptographyDeprecationWarning: TripleDES has been moved to cryptography.hazmat.decrepit.ciphers.algorithms.TripleDES and will be removed from cryptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
  b"3des-ctr": (algorithms.TripleDES, 24, modes.CTR).
2025-03-27T14:52:54+0000 [-] Reading configuration from ['/cowrie/cowrie-git/etc/cowrie.cfg.dist']
2025-03-27T14:52:54+0000 [-] Python Version 3.11.2 (main, Sep 14 2024, 03:00:30) [GCC 12.2.0]
2025-03-27T14:52:54+0000 [-] Twisted Version 24.10.0
2025-03-27T14:52:54+0000 [-] Cowrie Version 2.6.1
2025-03-27T14:52:54+0000 [-] Loaded output engine: Jsonlog
2025-03-27T14:52:54+0000 [twisted.scripts._twisted_unix.UnixAppLogger#info] twisted 24.10.0 (/cowrie/cowrie-env/bin/python3 3.11.2) starting up.
2025-03-27T14:52:54+0000 [twisted.scripts._twisted_unix.UnixAppLogger#info] reactor class: twisted.internet.epollreactor.EPollReactor.
2025-03-27T14:52:54+0000 [-] CowrieSSHFactory starting on 2222
2025-03-27T14:52:54+0000 [cowrie.ssh.factory.CowrieSSHFactory#info] Starting factory <cowrie.ssh.factory.CowrieSSHFactory object at 0x7477204de090>
2025-03-27T14:52:54+0000 [-] Generating new RSA keypair...
2025-03-27T14:52:54+0000 [-] Generating new ECDSA keypair...
2025-03-27T14:52:54+0000 [-] Generating new ed25519 keypair...
2025-03-27T14:52:54+0000 [-] Ready to accept SSH connections
```

Realización de los Ataques

Con el **honeypot Cowrie** correctamente instalado y emulando servicios SSH, el siguiente paso fue realizar una serie de ataques simulados para evaluar cómo el sistema reacciona y registra estos intentos. En esta práctica se utilizaron dos tipos de ataques principales: **fuerza bruta SSH** y **escaneo de puertos**. Ambas pruebas permiten ver cómo el honeypot simula una intrusión y cómo registra los intentos maliciosos.

1. Escaneo de Puertos con Nmap

El siguiente paso fue realizar un escaneo de puertos sobre el Ubuntu Server utilizando Nmap, una herramienta de escaneo de redes. Nmap es útil para identificar qué puertos están abiertos en una máquina, lo cual es esencial para cualquier atacante que quiera encontrar servicios vulnerables a explotar.

El comando utilizado fue:

```
nmap -p- 192.168.5.11
```

```
(kali㉿kali)-[~]  
$ nmap -p- 192.168.5.11  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-27 16:01 CET  
Nmap scan report for 192.168.5.11  
Host is up (0.00035s latency).  
Not shown: 65533 filtered tcp ports (no-response)  
PORT      STATE SERVICE  
22/tcp    open  ssh  
80/tcp    closed http  
MAC Address: 08:00:27:4F:11:16 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)  
  
Nmap done: 1 IP address (1 host up) scanned in 118.14 seconds
```

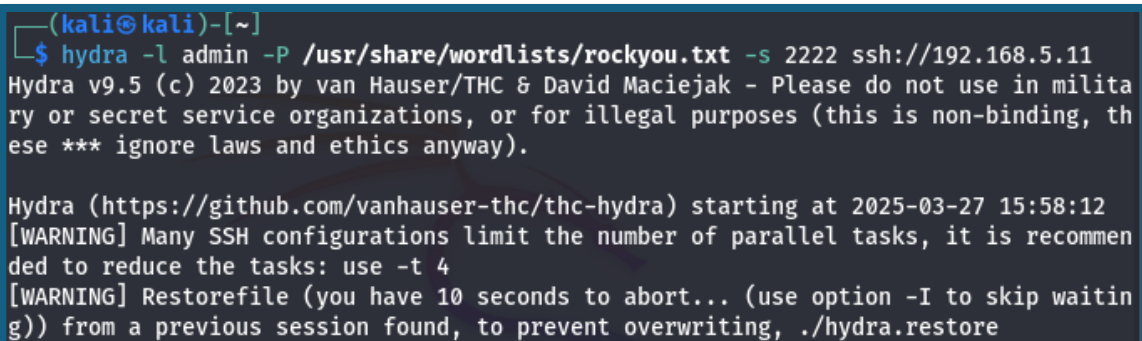
2. Ataque de Fuerza Bruta SSH

El ataque de **fuerza bruta SSH** es uno de los métodos más comunes utilizados por los atacantes para obtener acceso no autorizado a un servidor a través del protocolo SSH. En este tipo de ataque, se intentan múltiples combinaciones de nombre de usuario y contraseña hasta que se encuentra una que permita el acceso.

Para llevar a cabo este ataque, se utilizó **Hydra**, una herramienta de ataque de fuerza bruta muy popular en Kali Linux. **Hydra** automatiza el proceso de prueba de contraseñas, utilizando diccionarios predefinidos como el **rockyou.txt** para probar una lista de contraseñas contra un nombre de usuario determinado.

El siguiente comando fue utilizado para realizar el ataque:

```
hydra -l admin -P /usr/share/wordlists/rockyou.txt ssh://192.168.5.11:2222
```



```
(kali㉿kali)-[~]  
$ hydra -l admin -P /usr/share/wordlists/rockyou.txt -s 2222 ssh://192.168.5.11  
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in milita  
ry or secret service organizations, or for illegal purposes (this is non-binding, th  
ese *** ignore laws and ethics anyway).  
  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-27 15:58:12  
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommen  
ded to reduce the tasks: use -t 4  
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waitin  
g)) from a previous session found, to prevent overwriting, ./hydra.restore  
█
```

Los registros mostrados reflejan que **Cowrie** detecta y registra los intentos de ataque de **fuerza bruta SSH** realizados con **Hydra**. Cada intento de inicio de sesión fallido con diferentes combinaciones de **usuario (admin)** y **contraseñas** es registrado en los logs. **Cowrie** captura estos intentos, mostrando que el atacante prueba varias contraseñas, como **"bitch"**, **"maganda"**, **"babybug"**, y otras.

Estos intentos fallidos son almacenados como parte de la actividad registrada por el honeypot, evidenciando el ataque y proporcionando datos valiosos sobre las técnicas de los atacantes.

```
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,14,192.168.5.10] Could not read etc/userdb.txt, default database activated
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,14,192.168.5.10] login attempt [b'admin'/b'bitch'] failed
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] b'admin' failed auth b'password'
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] unauthorized login: ()
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] b'admin' trying auth b'password'
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,7,192.168.5.10] Could not read etc/userdb.txt, default database activated
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,7,192.168.5.10] login attempt [b'admin'/b'maganda'] failed
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] b'admin' failed auth b'password'
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] unauthorized login: ()
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] b'admin' trying auth b'password'
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,15,192.168.5.10] Could not read etc/userdb.txt, default database activated
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,15,192.168.5.10] login attempt [b'admin'/b'babyboy'] failed
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] b'admin' failed auth b'password'
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] unauthorized login: ()
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] b'admin' trying auth b'password'
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,6,192.168.5.10] Could not read etc/userdb.txt, default database activated
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,6,192.168.5.10] login attempt [b'admin'/b'casper'] failed
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] b'admin' failed auth b'password'
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] unauthorized login: ()
2025-03-27T14:58:39+0000 [cowrie.ssh.userauth.HoneyPotSSHUserAuthServer#debug] b'admin' trying auth b'password'
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,8,192.168.5.10] Could not read etc/userdb.txt, default database activated
2025-03-27T14:58:39+0000 [HoneyPotSSHTransport,8,192.168.5.10] login attempt [b'admin'/b'brenda'] failed
```

3. Ataque fuerza bruta HTTP GET

Este comando está realizando un **ataque de fuerza bruta HTTP** sobre un formulario de inicio de sesión web. Se está probando el nombre de usuario **admin** junto con una lista de contraseñas utilizando el método **HTTP GET**. **Hydra** enviará solicitudes a la página **/login** de la dirección **192.168.5.11**, intentando encontrar una contraseña correcta para el usuario **admin**.

El comando utilizado fue:

```
(kali@kali)~$ hydra -l admin -P /usr/share/wordlists/rockyou.txt -s 2222 http-get://192.168.5.11/login

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-27 16:14:16
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-get://192.168.5.11:2222/login
^CThe session file ./hydra.restore was written. Type "hydra -R" to resume session.
```



```
2025-03-27T15:14:43+0000 [HoneyPotSSHTransport,1043,192.168.5.10] Connection lost after 0.1 seconds
2025-03-27T15:14:43+0000 [HoneyPotSSHTransport,1044,192.168.5.10] Remote SSH version: GET /login HTTP/1.1
2025-03-27T15:14:43+0000 [HoneyPotSSHTransport,1044,192.168.5.10] Bad protocol version identification: b'GET /login HTTP/1.1'
2025-03-27T15:14:43+0000 [cowrie.ssh.transport.HoneyPotSSHTransport#info] connection lost
2025-03-27T15:14:43+0000 [HoneyPotSSHTransport,1044,192.168.5.10] Connection lost after 0.1 seconds
```

Alerta en Ubuntu Server 1

4. Ataque por SSH

Al observar que el puerto SSH estaba disponible, se decidió realizar un intento manual de acceso utilizando el usuario **root**. En lugar de realizar un ataque de **fuerza bruta**, como se mencionó previamente, el enfoque fue simplemente intentar acceder mediante **SSH** con el siguiente comando:

```
ssh root@192.168.5.11 -p 2222
```



```
(kali@kali)-[~]  
$ ssh -p 2222 root@192.168.5.11  
  
The authenticity of host '[192.168.5.11]:2222 ([192.168.5.11]:2222)' can't be established.  
ED25519 key fingerprint is SHA256:h8form0gi95SBFnd7HaTMYfjJmhunzLqBls0teIgzws.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '[192.168.5.11]:2222' (ED25519) to the list of known hosts.  
root@192.168.5.11's password:  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
root@svr04:~#
```

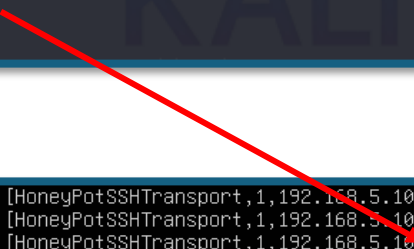
Para sorpresa, el **honeypot Cowrie** permitió el acceso exitoso a **root@192.168.5.11** utilizando la contraseña incorrecta. Este comportamiento simula lo que un atacante haría si se encuentra con un servidor SSH que no tiene configuraciones de seguridad adecuadas. Aunque el **SSH** debería rechazar contraseñas incorrectas, **Cowrie** está diseñado para simular un servidor que permite este tipo de pruebas, lo que lo convierte en un objetivo ideal para estudiar tácticas de ataque.

Registros Generados por Cowrie

Una vez que se obtuvo acceso al servidor, **Cowrie** registró este intento de acceso exitoso en sus logs, junto con el nombre de usuario utilizado, la contraseña, y la dirección IP del atacante.

Este comportamiento de registro es una característica clave de **Cowrie**, ya que proporciona una visión detallada de cómo los atacantes podrían intentar acceder a un sistema.

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@svr04:~# whoami
root
root@svr04:~# ls
root@svr04:~# ifconfig
```



```
2025-03-27T15:30:11+0000 [HoneyPotSSHTransport,1,192.168.5.10] CMD: whoami
2025-03-27T15:30:11+0000 [HoneyPotSSHTransport,1,192.168.5.10] Command found: whoami
2025-03-27T15:30:12+0000 [HoneyPotSSHTransport,1,192.168.5.10] CMD: ls
2025-03-27T15:30:12+0000 [HoneyPotSSHTransport,1,192.168.5.10] Command found: ls
2025-03-27T15:30:16+0000 [HoneyPotSSHTransport,1,192.168.5.10] CMD: ifconfig
2025-03-27T15:30:16+0000 [HoneyPotSSHTransport,1,192.168.5.10] Command found: ifconfig
```

Conclusión de la Prueba de Ataques

Los ataques realizados sobre el honeypot **Cowrie** demostraron cómo el sistema puede simular de manera efectiva la interacción con atacantes reales. Los intentos de **fuerza bruta SSH** y los **escaneos de puertos** fueron correctamente registrados, proporcionando datos valiosos sobre los métodos utilizados por los atacantes para comprometer sistemas. Esto valida la utilidad de **Cowrie** como herramienta de monitoreo de intrusiones y de análisis de amenazas en un entorno controlado.

Los registros generados por **Cowrie** durante los ataques proporcionan una visión profunda sobre las tácticas de ataque, lo que puede ayudar a mejorar la seguridad de sistemas reales mediante la identificación temprana de intentos de intrusión y la optimización de medidas de defensa.

Referencias

Ubuntu Server:

Ubuntu. (n.d.). *Ubuntu Server*. Ubuntu. Recuperado de <https://ubuntu.com/download/server>

Kali Linux:

Offensive Security. (n.d.). *Kali Linux*. Offensive Security. Recuperado de <https://www.kali.org/>

Cowrie:

Cowrie. (n.d.). *Cowrie honeypot*. Cowrie. Recuperado de <https://github.com/cowrie/cowrie>