

LAPORAN ADVANCED NETWORK SECURITY AND PROTOCOLS

“Implementasi Honeypot Sebagai Pendeteksi Serangan VPS”



DISUSUN OLEH :

ANDINI FEBRIANTI

105841113223

NUR QAMARIAH YUNUS

105841104323

PROGRAM STUDI INFORMATIKA

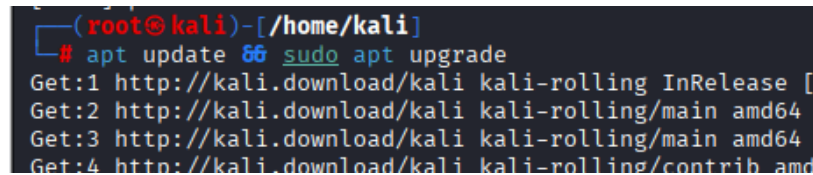
FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH MAKASSAR

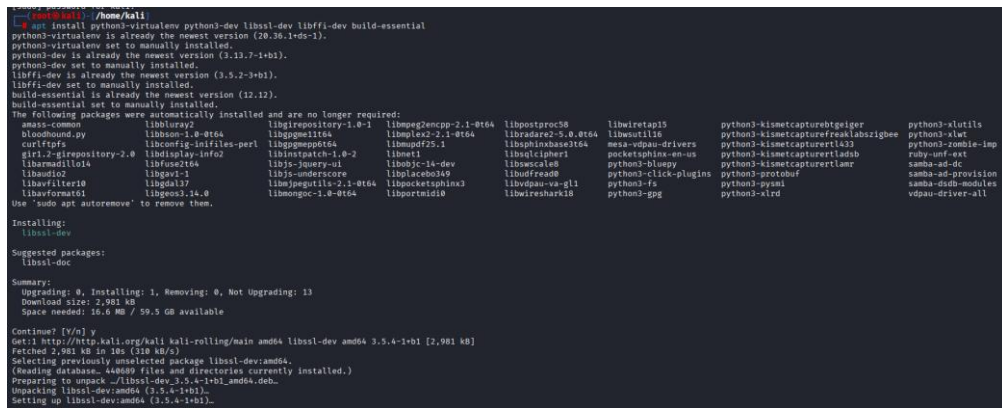
2026

1. Install paket pendukung

Pada tahap awal implementasi, dilakukan pembaruan sistem operasi Kali Linux menggunakan perintah `apt update` dan `apt upgrade`. Proses ini bertujuan untuk memperbarui daftar repository serta menginstal pembaruan paket terbaru agar sistem dalam kondisi stabil dan aman sebelum melakukan instalasi honeypot. Pembaruan sistem penting dilakukan untuk menghindari error dependency serta memastikan seluruh library yang dibutuhkan tersedia dalam versi terbaru.



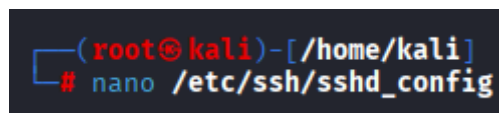
Gambar 1 Proses Pembaruan Repository dan Paket pada Kali Linux



Gambar 2 Proses Instalasi Dependency Pendukung Honeypot

- Ubah port SSH 22 ke port SSH 2222 untuk server, lalu restart.

Pada tahap ini dilakukan konfigurasi layanan SSH dengan mengakses file `sshd_config` menggunakan perintah `nano /etc/ssh/sshd_config`. File konfigurasi ini digunakan untuk mengatur pengaturan utama SSH server, termasuk perubahan port default dari 22 menjadi 2222. Perubahan ini dilakukan untuk mengamankan server asli dan mengalihkan port 22 agar dapat digunakan oleh honeypot sebagai umpan bagi attacker.



Gambar 3 Proses Akses File Konfigurasi SSH Server

```
Include /etc/ssh/sshd_config.d/*.conf
Port 60000
#AddressFamily inet
#ListenAddress 0.0.0.0
#ListenAddress ::
```

Gambar 4 Perubahan Konfigurasi Port pada File sshd_config

Setelah melakukan perubahan konfigurasi pada file sshd_config, langkah selanjutnya adalah me-restart layanan SSH menggunakan perintah systemctl restart ssh. Proses ini dilakukan agar konfigurasi baru, termasuk perubahan port, dapat diterapkan dan berjalan secara aktif pada sistem. Tanpa melakukan restart service, perubahan konfigurasi tidak akan berpengaruh terhadap layanan SSH yang sedang berjalan.

```
(root@kali)-[/home/kali]
# systemctl restart ssh
```

Gambar 5 Proses Restart Layanan SSH

3. Install honeypot cowrie

- Buat user khusus, jangan jalankan sebagai root

Pada tahap ini dilakukan pembuatan user khusus bernama **cowrie** menggunakan perintah `sudo adduser --disabled-password cowrie`. User ini dibuat tanpa password login langsung sebagai langkah keamanan agar honeypot tidak dijalankan menggunakan akun root. Penggunaan user khusus bertujuan untuk membatasi hak akses sistem sehingga jika terjadi eksploitasi pada honeypot, dampaknya tidak langsung mempengaruhi sistem utama.

```
(kali@kali)-[~]
$ sudo adduser --disabled-password cowrie
[sudo] password for kali:
Changing the user information for cowrie
Enter the new value, or press ENTER for the default
  Full Name []: ria
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
```

Gambar 6 Proses Pembuatan User Khusus Cowrie

- Pindah ke user cowrie

Setelah user *cowrie* berhasil dibuat, langkah berikutnya adalah berpindah ke user tersebut menggunakan perintah `sudo su - cowrie`. Proses ini dilakukan agar seluruh instalasi dan konfigurasi honeypot dijalankan menggunakan akun non-root. Hal ini bertujuan untuk meningkatkan keamanan sistem dengan menerapkan prinsip *least privilege*, sehingga honeypot tidak memiliki hak akses penuh terhadap sistem utama.

```
(kali@kali)-[~]
$ sudo su - cowrie
(cowrie@kali)-[~]
$
```

Gambar 7 Proses Berpindah ke User Cowrie

- **Clone source code:** git clone <http://github.com/cowrie/cowrie>

Pada tahap ini dilakukan proses pengambilan source code honeypot Cowrie dari repository resmi GitHub menggunakan perintah git clone <http://github.com/cowrie/cowrie>. Proses cloning ini bertujuan untuk mengunduh seluruh file dan struktur program Cowrie ke dalam sistem lokal. Setelah proses selesai, folder cowrie akan terbentuk dan berisi seluruh file konfigurasi serta script yang dibutuhkan untuk instalasi dan menjalankan honeypot

```
(cowrie@kali)-[~]
$ git clone http://github.com/cowrie/cowrie
Cloning into 'cowrie'...
warning: redirecting to https://github.com/cowrie/cowrie/
remote: Enumerating objects: 20802, done.
remote: Counting objects: 100% (65/65), done.
remote: Compressing objects: 100% (49/49), done.
remote: Total 20802 (delta 40), reused 18 (delta 16), pack-reused 20737 (from 2)
Receiving objects: 100% (20802/20802), 11.02 MiB | 4.73 MiB/s, done.
Resolving deltas: 100% (14551/14551), done.
```

Gambar 8 Proses Cloning Repository Cowrie dari GitHub

- **Setup virtual environment**

Pada tahap ini dilakukan pembuatan dan aktivasi virtual environment untuk menjalankan honeypot Cowrie. Virtual environment dibuat menggunakan perintah `virtualenv --python=python3 cowrie-env`, kemudian diaktifkan dengan `source cowrie-env/bin/activate`. Setelah environment aktif, dilakukan pembaruan pip dan instalasi seluruh dependency yang tercantum dalam file `requirements.txt`.

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

(cowrie@kali)-[~/cowrie]
$ virtualenv --python=python3 cowrie-env
created virtual environment CPython3.13.11.final.0-64 in 532ms
creator CPython3Posix(dest=/home/cowrie/cowrie/cowrie-env, clear=False, no_vcs_ignore=False, global=False)
seeder FromAppData(download=False, pip=copy, via=copy, app_data_dir=/home/cowrie/.local/share/virtualenv)
added seed packages: pip=25.3
activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator

(cowrie@kali)-[~/cowrie]
$ source cowrie-env/bin/activate

(cowrie-env)(cowrie@kali)-[~/cowrie]
$ pip install --upgrade pip
Requirement already satisfied: pip in ./cowrie-env/lib/python3.13/site-packages (25.3)

(cowrie-env)(cowrie@kali)-[~/cowrie]
$ pip install -r requirements.txt
Collecting attrs==25.4.0 (from -r requirements.txt (line 1))
  Downloading attrs-25.4.0-py3-none-any.whl.metadata (10 kB)
Collecting bcrypt==5.0.0 (from -r requirements.txt (line 2))
  Downloading bcrypt-5.0.0-cp39-abi3-manylinux_2_34_x86_64.whl.metadata (10 kB)
```

Gambar 9 Pembuatan dan Instalasi Dependency Virtual Environment Cowrie

- Salin file onfigurasi dalam folder cowrie/etc

Pada tahap ini dilakukan konfigurasi awal honeypot dengan masuk ke direktori etc di dalam folder Cowrie. File konfigurasi default cowrie.cfg.dist kemudian disalin menjadi cowrie.cfg menggunakan perintah cp cowrie.cfg.dist cowrie.cfg.

```
(cowrie-env)(cowrie@kali)-[~/cowrie]
$ dir
bin          CONTRIBUTING.rst  docker  etc          INSTALL.rst  Makefile      pyproject.toml  requirements-output.txt  setup.py  var
CHANGELOG.rst  cowrie-env        docs    honeyfs      LICENSE.rst  MANIFEST.in   README.rst      requirements.txt          src

(cowrie-env)(cowrie@kali)-[~/cowrie]
$ ^C

(cowrie-env)(cowrie@kali)-[~/cowrie]
$ cd etc

(cowrie-env)(cowrie@kali)-[~/cowrie/etc]
$ dir
cowrie.cfg.dist  userdb.example

(cowrie-env)(cowrie@kali)-[~/cowrie/etc]
$ cp cowrie.cfg.dist cowrie.cfg
```

Gambar 10 Proses Penyalinan File Konfigurasi Cowrie

- Install cowrie

Pada tahap ini dilakukan proses instalasi Cowrie secara lokal menggunakan perintah pip install -e .. Perintah tersebut menginstal source code Cowrie dalam mode *editable*, sehingga setiap perubahan konfigurasi atau pengembangan dapat langsung diterapkan tanpa perlu instalasi ulang.

```
(cowrie-env)(cowrie@kali)-[~/cowrie]
$ pip install -e .
Obtaining file:///home/cowrie/cowrie
Installing build dependencies ... done
```

Gambar 11 Proses Instalasi Lokal Honeypot Cowrie

- Masuk ke server cowrie dan modifikasi file cowrie.cfg (ubah port nya ke 22)

Pada tahap ini dilakukan pengeditan file konfigurasi utama honeypot menggunakan perintah nano cowrie.cfg. File cowrie.cfg berisi pengaturan penting seperti konfigurasi port, sistem logging, serta parameter operasional honeypot.

```
(cowrie-env)(cowrie@kali)-[~/cowrie/etc]
$ nano cowrie.cfg
```

Gambar 12 Proses Pengeditan File Konfigurasi Cowrie

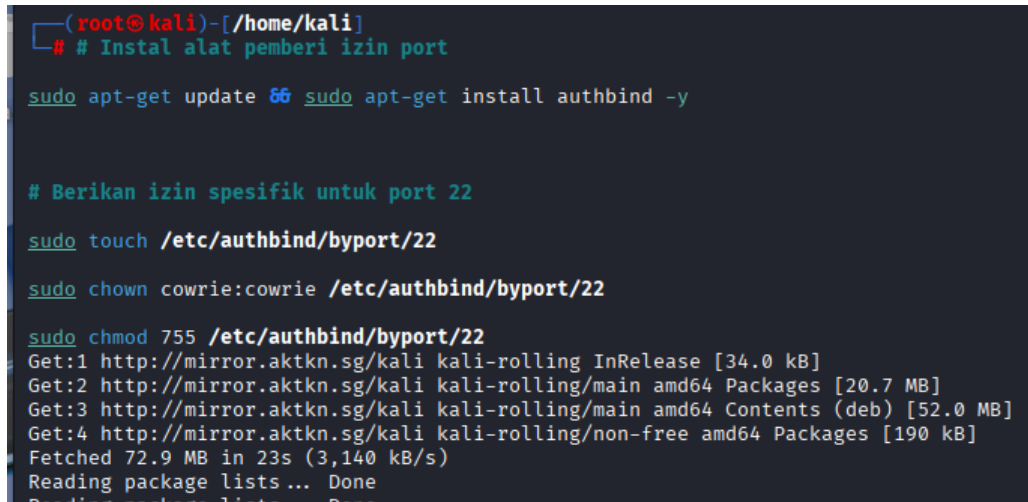
```
listen_endpoints = tcp:22:interface=0.0.0.0

# Enable the SFTP subsystem
# (default: true)
sftp_enabled = true
```

Gambar 13 Konfigurasi Port Honeypot pada File cowrie.cfg

- Izinkan Port 22 untuk User Non-Root (Authbind)

Pada tahap ini dilakukan instalasi dan konfigurasi **authbind** untuk mengizinkan user non-root menjalankan layanan pada port di bawah 1024, khususnya port 22. Secara default, port 22 hanya dapat dijalankan oleh user root. Oleh karena itu, authbind digunakan agar honeypot Cowrie yang berjalan dengan user *cowrie* tetap dapat menggunakan port 22 tanpa harus dijalankan sebagai root.



```
(root@kali)-[/home/kali]
# # Instal alat pemberi izin port

sudo apt-get update && sudo apt-get install authbind -y

# Berikan izin spesifik untuk port 22

sudo touch /etc/authbind/byport/22

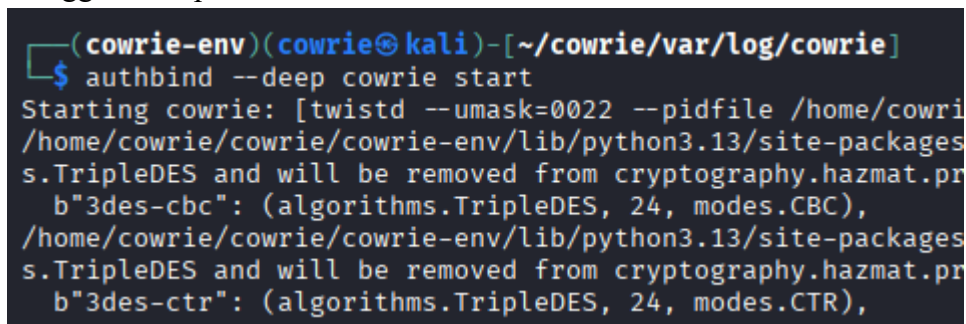
sudo chown cowrie:cowrie /etc/authbind/byport/22

sudo chmod 755 /etc/authbind/byport/22
Get:1 http://mirror.aktkn.sg/kali kali-rolling InRelease [34.0 kB]
Get:2 http://mirror.aktkn.sg/kali kali-rolling/main amd64 Packages [20.7 MB]
Get:3 http://mirror.aktkn.sg/kali kali-rolling/main amd64 Contents (deb) [52.0 MB]
Get:4 http://mirror.aktkn.sg/kali kali-rolling/non-free amd64 Packages [190 kB]
Fetched 72.9 MB in 23s (3,140 kB/s)
Reading package lists... Done
```

Gambar 14 Konfigurasi Authbind untuk Izin Port 22

- Masuk ke cowrie dan jalankan dengan authbind

Pada tahap ini honeypot Cowrie dijalankan menggunakan perintah **authbind --deep bin/cowrie start**. Penggunaan opsi **--deep** memastikan bahwa seluruh proses turunan (child process) juga tetap memiliki izin menggunakan port 22.

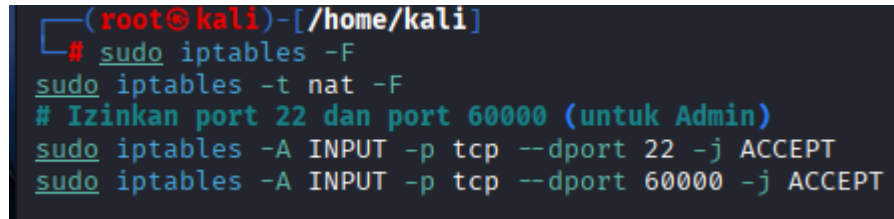


```
(cowrie-env)(cowrie@kali)-[~/cowrie/var/log/cowrie]
$ authbind --deep cowrie start
Starting cowrie: [twistd --umask=0022 --pidfile /home/cowrie/
/home/cowrie/cowrie/cowrie-env/lib/python3.13/site-packages
s.TripleDES and will be removed from cryptography.hazmat.pr
b"3des-cbc": (algorithms.TripleDES, 24, modes.CBC),
/home/cowrie/cowrie/cowrie-env/lib/python3.13/site-packages
s.TripleDES and will be removed from cryptography.hazmat.pr
b"3des-ctr": (algorithms.TripleDES, 24, modes.CTR),
```

Gambar 15 Proses Menjalankan Honeypot Cowrie dengan Authbin

- Bersihkan iptables dan izinkan port 22 dan 60000 untuk admin

Pada tahap ini dilakukan konfigurasi firewall menggunakan iptables untuk memastikan lalu lintas jaringan berjalan sesuai kebutuhan implementasi honeypot. Perintah iptables -F dan iptables -t nat -F digunakan untuk membersihkan aturan firewall sebelumnya agar tidak terjadi konflik konfigurasi.



```
(root@kali)-[/home/kali]
# sudo iptables -F
sudo iptables -t nat -F
# Izinkan port 22 dan port 60000 (untuk Admin)
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 60000 -j ACCEPT
```

Gambar 16 Konfigurasi Firewall untuk Mengizinkan Port Honeypot dan Admin

B. Tools yang digunakan

1. Kali linux

Kali Linux digunakan sebagai sistem operasi utama dalam proses implementasi dan pengujian. Sistem ini digunakan untuk melakukan konfigurasi server, instalasi honeypot, serta menjalankan berbagai tools pengujian keamanan seperti Nmap, Hydra, dan LOIC.

2. Cowrie Honeypot

Cowrie merupakan honeypot berbasis SSH yang digunakan untuk mensimulasikan layanan SSH palsu pada port 22. Tools ini berfungsi untuk:

- Merekam aktivitas koneksi masuk
- Mencatat percobaan login
- Menyimpan informasi IP penyerang
- Mendokumentasikan detail sesi dan protokol

Cowrie dijalankan menggunakan virtual environment dan dikonfigurasi agar berjalan pada port 22 menggunakan authbind.

3. Virtualenv (Python Virtual Environment)

Virtualenv digunakan untuk membuat lingkungan Python terisolasi khusus untuk menjalankan Cowrie. Hal ini bertujuan agar dependency yang digunakan tidak mengganggu sistem utama.

4. Authbind

Authbind digunakan untuk mengizinkan user non-root (user cowrie) menjalankan layanan pada port 22. Secara default, port di bawah 1024 hanya dapat dijalankan

oleh root, sehingga authbind diperlukan untuk menerapkan prinsip keamanan least privilege.

5. IPTables

IPTables digunakan untuk mengatur firewall pada server. Dalam implementasi ini digunakan untuk:

- Membersihkan rule sebelumnya
- Mengizinkan akses ke port 22 (honeypot)
- Mengizinkan akses ke port 60000 (SSH admin)

6. Nmap

Nmap digunakan untuk melakukan pengujian port scanning terhadap server target.

Fungsinya:

- Mendeteksi port terbuka
- Mengidentifikasi layanan aktif
- Mengetahui versi service dengan opsi -sV

Nmap digunakan pada pengujian Individual, Double, dan Multiple Attack.

7. Hydra

Hydra digunakan untuk melakukan pengujian brute force terhadap layanan SSH.

Fungsinya:

- Mencoba berbagai kombinasi password
- Menguji keamanan kredensial login
- Mensimulasikan serangan login berulang

8. Hydra low Orbit Ion Cannon (LOIC)

LOIC digunakan untuk mensimulasikan serangan DDoS dengan metode TCP flooding ke port 22.

Fungsinya:

- Menghasilkan lonjakan koneksi dalam jumlah besar
- Menguji kemampuan honeypot dalam mendeteksi trafik abnormal

9. Nano Text Editor

Nano digunakan untuk mengedit file konfigurasi seperti:

- /etc/ssh/sshd_config
- cowrie.cfg

C. Pengujian

Sebelum melakukan pengujian serangan, langkah awal yang dilakukan adalah mengetahui alamat IP server menggunakan perintah `ip a`. Informasi ini diperlukan agar proses pengujian seperti port scanning, brute force, dan DDoS dapat diarahkan ke target yang benar.

```
(cowrie-env)(cowrie@kali)-[~/cowrie/var/log/cowrie]
$ 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 00:0c:29:c9:03:10 brd ff:ff:ff:ff:ff:ff
   inet 10.3.101.83/23 brd 10.3.101.255 scope global dynamic noprefixroute eth0
       valid_lft 352sec preferred_lft 352sec
   inet6 fe80::83b8:9209:ede2:51a2/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Gambar 17 Identifikasi Alamat IP Server Menggunakan Perintah `ip a`

1. Individual

- Port Scanning

- a. Lakukan nmap pada ip server

Pada tahap pengujian individual, dilakukan serangan **port scanning** menggunakan tools Nmap dengan perintah `nmap -p 22,2222 10.3.101.83`. Pengujian ini bertujuan untuk mengetahui status port yang terbuka pada server target.

```
(root@kali)-[/home/kali]
# nmap -p 22,2222 10.3.101.83
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-30 01:31 EST
Nmap scan report for 10.3.101.83
Host is up (0.00054s latency).

PORT      STATE SERVICE
22/tcp    open  ssh
2222/tcp  closed EtherNetIP-1
MAC Address: 4C:82:A9:8C:78:2B (Cloud Network Technology Singapore PTE.)

Nmap done: 1 IP address (1 host up) scanned in 0.14 seconds
```

Gambar 18 Hasil Pengujian Port Scanning Menggunakan Nmap

Pada pengujian ini dilakukan port scanning menggunakan perintah `nmap -sV -p 22 192.168.58.116` untuk mendeteksi versi layanan yang berjalan pada port 22. Opsi `-sV` digunakan untuk mengidentifikasi detail service dan versi sistem yang aktif pada target.

```
(kali@kali)-[~]
$ nmap -sV -p 22 192.168.58.116
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-29 23:19 EST
Nmap scan report for 192.168.58.116
Host is up (0.00079s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.2p1 Debian 2+deb12u3 (protocol 2.0)
MAC Address: 4C:82:A9:8C:78:2B (Cloud Network Technology Singapore PTE.)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org
Nmap done: 1 IP address (1 host up) scanned in 13.68 seconds
```

Gambar 19 Hasil Port Scanning dengan Deteksi Versi Service (`-sV`)

b. Hasil log file sesudah dilakukan nmap

Setelah dilakukan pengujian port scanning menggunakan Nmap, dilakukan pengecekan log honeypot dengan perintah `tail -f cowrie.json`. Hasil log menunjukkan adanya event "cowrie.session.connect" yang mencatat alamat IP sumber (`src_ip`) penyerang serta port tujuan (`dst_port`: 22).

```
(cowrie-env)(cowrie@kali)-[~/cowrie/var/log/cowrie]
$ tail -f cowrie.json
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":52648,"dst_ip":"10.3.101.83","dst_port":22,"session":"b9ae513e57ab","sensor":"kali","uuid":"01-30T01:31:58.129942Z","src_ip":"10.3.101.72","session":"b9ae513e57ab","protocol":"ssh"}
```

Gambar 20 Log Deteksi Serangan Port Scanning pada Honeypot

- Brute force

a. buat file yang berisi daftar password lalu lakukan serangan menggunakan hydra.

Pada tahap ini dibuat file daftar password yang akan digunakan dalam pengujian brute force menggunakan Hydra. Perintah `echo -e "password\n123456\nadmin\nroot\nqwerty" > pass.txt` digunakan untuk membuat file bernama `pass.txt` yang berisi beberapa kombinasi password umum.

```
(root@kali)-[/home/kali]
# echo -e "password\n123456\nadmin\nroot\nqwerty" > pass.txt
```

Gambar 21 Pembuatan File Daftar Password untuk Pengujian Brute Force

Pada tahap ini dilakukan pengujian serangan brute force menggunakan tools Hydra dengan perintah: `hydra -l root -P pass.txt 10.3.101.83 ssh`. Serangan ini bertujuan untuk mencoba berbagai kombinasi password terhadap akun root pada layanan SSH port 22. Hasil pengujian menunjukkan bahwa Hydra berhasil menemukan beberapa password yang dianggap valid, seperti `qwerty`, `password`, dan `admin`.

```
(root@kali)-[/home/kali]
# hydra -l root -P pass.txt 10.3.101.83 ssh
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).

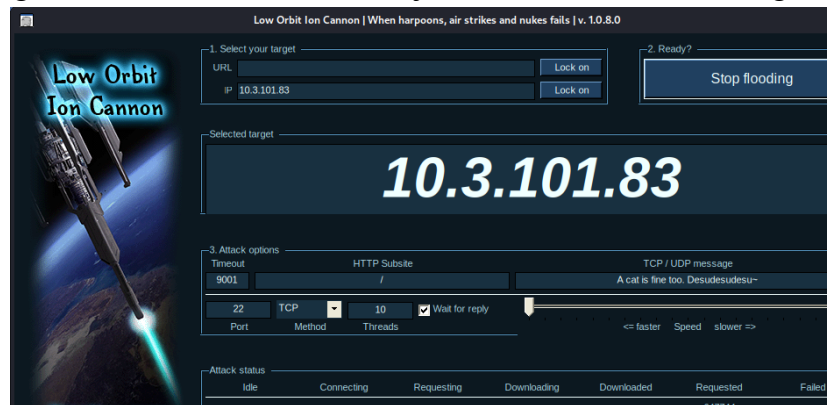
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-01-30 01:33:40
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 5 tasks per 1 server, overall 5 tasks, 5 login tries (l:1/p:5), ~1 try per task
[DATA] attacking ssh://10.3.101.83:22/
[22][ssh] host: 10.3.101.83 login: root password: qwerty
[22][ssh] host: 10.3.101.83 login: root password: password
[22][ssh] host: 10.3.101.83 login: root password: admin
1 of 1 target successfully completed, 3 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2026-01-30 01:33:41
```

Gambar 22 Hasil Serangan Brute Force Menggunakan Hydra

- b. Hasil brute force menggunakan hydra

Gambar 23 Log Deteksi Serangan Brute Force pada Honeypot

- DDoS attack
 - a. Lakukan serangan DDoS menggunakan loic



Gambar 24 Simulasi Serangan DDoS Menggunakan LOIC

b. Hasil serangan di file log

```
[cowrie-env](cowrie@kali) [~/cowrie/var/log/cowrie]
tail -f cowrie.json
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46942,"dst_ip":"10.
on: 10.3.101.72:46942 (10.3.101.83:22) [session: 6c3c310dc884]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46948,"dst_ip":"10.
on: 10.3.101.72:46948 (10.3.101.83:22) [session: 5f07745b090e]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46958,"dst_ip":"10.
on: 10.3.101.72:46958 (10.3.101.83:22) [session: df77be0b580c]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46970,"dst_ip":"10.
on: 10.3.101.72:46970 (10.3.101.83:22) [session: a74f911ae061]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46986,"dst_ip":"10.
on: 10.3.101.72:46986 (10.3.101.83:22) [session: 7c8d2c850127]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":47002,"dst_ip":"10.
on: 10.3.101.72:47002 (10.3.101.83:22) [session: 1d12fb13ef0f]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":47008,"dst_ip":"10.
on: 10.3.101.72:47008 (10.3.101.83:22) [session: dac179196522]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":47018,"dst_ip":"10.
on: 10.3.101.72:47018 (10.3.101.83:22) [session: 5c7e5c77f7b4]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":47022,"dst_ip":"10.
on: 10.3.101.72:47022 (10.3.101.83:22) [session: 0ea085ed1ab1]", "sensor": "kali", "uid": "0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":47038,"dst_ip":"10.
on: 10.3.101.72:47038 (10.3.101.83:22) [session: 4cefff4a03b7]", "sensor": "kali", "uid": "0
```

Gambar 25 Log Deteksi Serangan DDoS pada Honeypot

2. Double

- Port Scanning & Brute Force Attack

Pertama, dilakukan scanning menggunakan Nmap untuk memastikan port 22 dalam kondisi terbuka dan layanan SSH aktif. Hasil menunjukkan port 22 berstatus open (ssh), sehingga target dapat diakses untuk tahap berikutnya. Selanjutnya, dilakukan serangan brute force menggunakan Hydra dengan file `pass.txt` sebagai daftar password. Hasil menunjukkan beberapa password terdeteksi sebagai valid, seperti `password`, `admin`, dan `qwerty`. Namun, layanan yang diakses merupakan honeypot Cowrie, sehingga login tersebut merupakan simulasi untuk menjebak penyerang. Kombinasi kedua serangan ini berhasil diarahkan ke honeypot dan seluruh aktivitasnya tercatat pada file log. Dengan demikian, pengujian Double Attack (Port Scanning & Brute Force) dinyatakan berhasil terdeteksi 100%.

```
(root@kali)~/home/kali
# hydra -l root -P pass.txt 10.3.101.83 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do no
service organizations, or for illegal purposes (this is non-binding,
hics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-0
[WARNING] Many SSH configurations limit the number of parallel tasks,
the tasks: use -t 4
[DATA] max 5 tasks per 1 server, overall 5 tasks, 5 login tries (l:1/
[DATA] attacking ssh://10.3.101.83:22/
[22][ssh] host: 10.3.101.83 login: root password: password
[22][ssh] host: 10.3.101.83 login: root password: admin
[22][ssh] host: 10.3.101.83 login: root password: qwerty
1 of 1 target successfully completed, 3 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2026-0
```

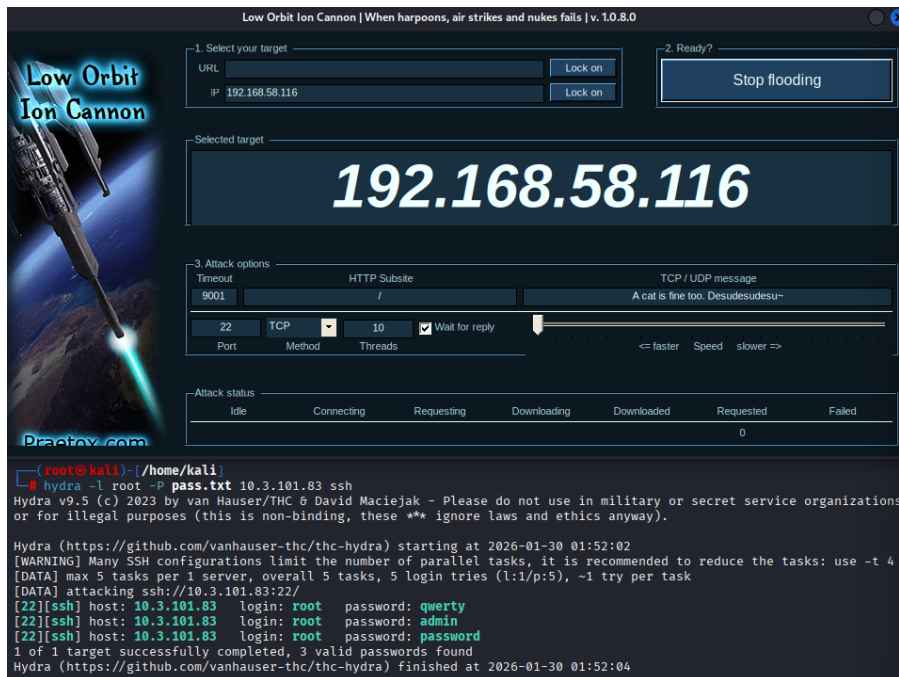
Gambar 26 Pengujian Double Attack: Port Scanning dan Brute Force

```
(cowrie-env)(cowrie@kali)-[~/cowrie/var/log/cowrie]
$ tail -f cowrie.json | grep -a -E "session.connect|login.failed"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46676,"dst_ip":"10
on: 10.3.101.72:46676 (10.3.101.83:22) [session: d3f54061e051],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46688,"dst_ip":"10
on: 10.3.101.72:46688 (10.3.101.83:22) [session: 6212082fd837],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46700,"dst_ip":"10
on: 10.3.101.72:46700 (10.3.101.83:22) [session: 627596eb1235],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46708,"dst_ip":"10
on: 10.3.101.72:46708 (10.3.101.83:22) [session: 94a677c3e1ee],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":46714,"dst_ip":"10
on: 10.3.101.72:46714 (10.3.101.83:22) [session: cdec2d75f4fe],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":35752,"dst_ip":"10
on: 10.3.101.72:35752 (10.3.101.83:22) [session: a3556fac7c3b],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":35768,"dst_ip":"10
on: 10.3.101.72:35768 (10.3.101.83:22) [session: ffc0d342a88],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":35774,"dst_ip":"10
on: 10.3.101.72:35774 (10.3.101.83:22) [session: 65e0bf6c4c13],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":35782,"dst_ip":"10
on: 10.3.101.72:35782 (10.3.101.83:22) [session: 5b351177d7fb],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":35784,"dst_ip":"10
on: 10.3.101.72:35784 (10.3.101.83:22) [session: b7e6215a0f84],"sensor":"kali","uuid":"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":35796,"dst_ip":"10
on: 10.3.101.72:35796 (10.3.101.83:22) [session: 757f98c2e700],"sensor":"kali","uuid":"
{"eventid":"cowrie.login.failed","username":"root","password":"root","message":"login at
imestamp":"2026-01-30T01:48:47.406645Z","src_ip":"10.3.101.72","session":"b7e6215a0f84",
{"eventid":"cowrie.login.failed","username":"root","password":"123456","message":"login
","timestamp":"2026-01-30T01:48:47.457869Z","src_ip":"10.3.101.72","session":"757f98c2e7
```

Gambar 27 Log Deteksi Double Attack (Port Scanning dan Brute Force)

Pola ini membuktikan bahwa setelah port berhasil terdeteksi terbuka, attacker langsung melakukan eksploitasi melalui brute force. Honeypot Cowrie berhasil merekam kedua aktivitas tersebut secara bersamaan dalam satu rangkaian log, sehingga pengujian Double Attack (Port Scanning & Brute Force) dinyatakan terdeteksi 100%.

- Brute Force Attack & DDoS Attack



Gambar 29 Pengujian Double Attack: Brute Force dan DDoS

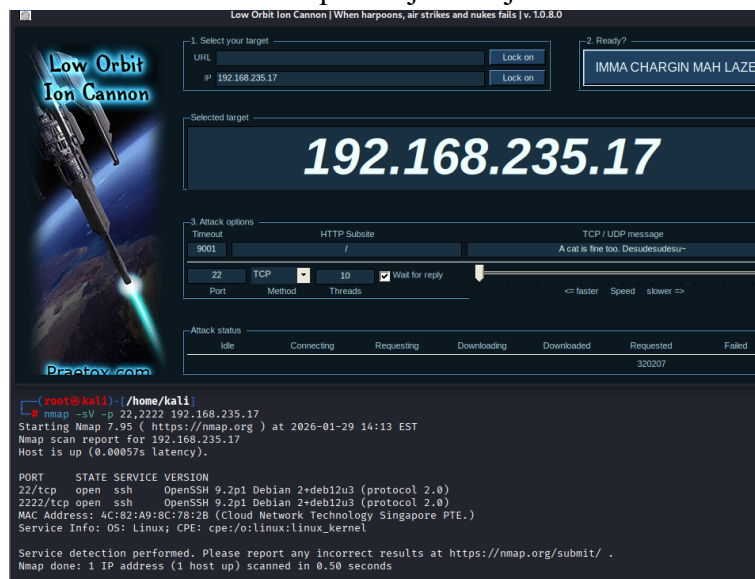
Kombinasi kedua serangan ini mensimulasikan kondisi serangan nyata di mana attacker tidak hanya mencoba menebak kredensial, tetapi juga melakukan flooding untuk melemahkan layanan. Dalam implementasi ini, seluruh aktivitas diarahkan ke honeypot Cowrie dan tidak ke server asli. Honeypot berhasil mencatat percobaan login serta lonjakan koneksi secara bersamaan, sehingga pengujian dinyatakan terdeteksi 100%.

```
(cowrie-env)(cowrie@kali)~/.cowrie/var/log/cowrie
$ tail -f cowrie.json | grep -a -E "cowrie.session.connect|cowrie.login.failed"
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":34694,"dst_ip":"10.3.101.72","dst_port":34694,"session":"b91bc65bbb9f","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":34700,"dst_ip":"10.3.101.72","dst_port":34700,"session":"87411b5e921a","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":34712,"dst_ip":"10.3.101.72","dst_port":34712,"session":"cbd21f58e2c2","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":34726,"dst_ip":"10.3.101.72","dst_port":34726,"session":"ee12b749cc97","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":34738,"dst_ip":"10.3.101.72","dst_port":34738,"session":"2a4dd1db3744","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":33764,"dst_ip":"10.3.101.72","dst_port":33764,"session":"46efd3da01a","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":33770,"dst_ip":"10.3.101.72","dst_port":33770,"session":"c21ffffb5365c","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":33786,"dst_ip":"10.3.101.72","dst_port":33786,"session":"9e3dfe3561c1","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":33800,"dst_ip":"10.3.101.72","dst_port":33800,"session":"a3eba69d69e7","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":33808,"dst_ip":"10.3.101.72","dst_port":33808,"session":"cce5fddd88e9","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":33812,"dst_ip":"10.3.101.72","dst_port":33812,"session":"033bbe6bc66e","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.login.failed","username":"root","password":"root","message":"login attempt failed","src_ip":"10.3.101.72","src_port":33812,"session":"cce5fddd88e9","sensor":"kali","uuid":"033bbe6bc6e"}
{"eventid":"cowrie.login.failed","username":"root","password":"123456","message":"login attempt failed","src_ip":"10.3.101.72","src_port":33812,"session":"cce5fddd88e9","sensor":"kali","uuid":"033bbe6bc6e"}
{"timestamp":"2026-01-30T01:54:01.702232Z","src_ip":"10.3.101.72","session":"033bbe6bc6e"}
```

Gambar 30 Log Deteksi Multiple Attack pada Honeypot

- DDoS Attack & Port Scanning

Kombinasi ini mensimulasikan kondisi di mana attacker melakukan pemetaan layanan (reconnaissance) sekaligus mencoba membanjiri server dengan trafik berlebihan. Dalam implementasi ini, seluruh trafik diarahkan ke honeypot, sehingga server asli tetap aman. Honeypot tetap aktif dan mampu mencatat aktivitas koneksi meskipun terjadi lonjakan trafik.



Gambar 31 Pengujian Double Attack: DDoS dan Port Scanning

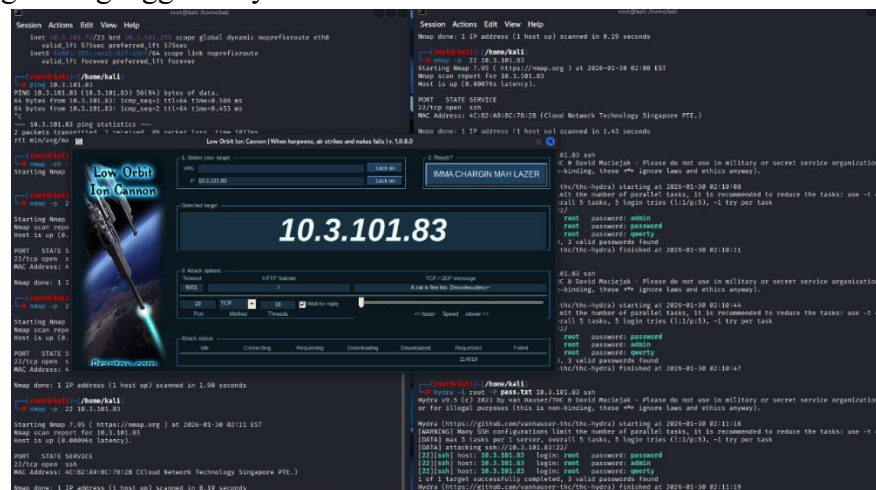
Honeypot Cowrie berhasil mencatat seluruh aktivitas koneksi tersebut secara real-time, termasuk IP penyerang, port sumber, dan session ID. Dengan demikian, kombinasi serangan DDoS dan Port Scanning dinyatakan berhasil terdeteksi 100%.

```
(cowrie-env)(cowrie@kali)-[~/cowrie/var/log/cowrie]
$ tail -f cowrie.json
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44570,"dst_ip":"10.
on: 10.3.101.72:44570 (10.3.101.83:22) [session: fdad38fca96d]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44574,"dst_ip":"10.
on: 10.3.101.72:44574 (10.3.101.83:22) [session: 7ec3dbec8c2f]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44580,"dst_ip":"10.
on: 10.3.101.72:44580 (10.3.101.83:22) [session: ae360acc20af]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44594,"dst_ip":"10.
on: 10.3.101.72:44594 (10.3.101.83:22) [session: d66c6820b789]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44604,"dst_ip":"10.
on: 10.3.101.72:44604 (10.3.101.83:22) [session: 4c8ba8417b78]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44612,"dst_ip":"10.
on: 10.3.101.72:44612 (10.3.101.83:22) [session: ed480e7dc2d5]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44624,"dst_ip":"10.
on: 10.3.101.72:44624 (10.3.101.83:22) [session: 8fb0e3fec3df]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44640,"dst_ip":"10.
on: 10.3.101.72:44640 (10.3.101.83:22) [session: da7c53ef01a7]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44654,"dst_ip":"10.
on: 10.3.101.72:44654 (10.3.101.83:22) [session: 63a01fa99c4f]","sensor":"kali","uuid":"0
{"eventid":"cowrie.session.connect","src_ip":"10.3.101.72","src_port":44656,"dst_ip":"10.
on: 10.3.101.72:44656 (10.3.101.83:22) [session: c849cda40a92]","sensor":"kali","uuid":"0
```

Gambar 32 Log Deteksi Double Attack (DDoS dan Port Scanning)

3. Multiple

Kombinasi ketiga serangan ini mensimulasikan kondisi serangan nyata yang kompleks, di mana attacker melakukan reconnaissance, eksploitasi kredensial, dan flooding trafik dalam satu waktu. Dalam implementasi ini, seluruh aktivitas diarahkan ke honeypot sehingga server asli tetap aman. Honeypot berhasil mencatat seluruh koneksi, percobaan login, serta lonjakan trafik tanpa mengalami gangguan layanan.



Gambar33 Pengujian Multiple Attack (Port Scanning, Brute Force, dan DDoS)

Gambar 34 Log Deteksi Multiple Attack pada Honeypot Cowrie

D. Hasil pengujian

No	Pola Serangan	Nama Pengujian	Kondisi Sebelum (%)	Kondisi Sesudah (%)	Hasil (Terdeteksi / Tidak)
1	Individual	Port Scanning	0%	100%	Terdeteksi
2	Individual	Bruteforce Attack	0%	100%	Terdeteksi
3	Individual	DDoS Attack	0%	100%	Terdeteksi
4	Double	Port Scanning & Bruteforce Attack	0%	100%	Terdeteksi
5	Double	Bruteforce Attack & DDoS Attack	0%	100%	Terdeteksi

6	Double	DDoS Attack & Port Scanning	0%	100%	Terdeteksi
7	Multiple	Port Scanning, Bruteforce Attack, & DDoS Attack	0%	100%	Terdeteksi