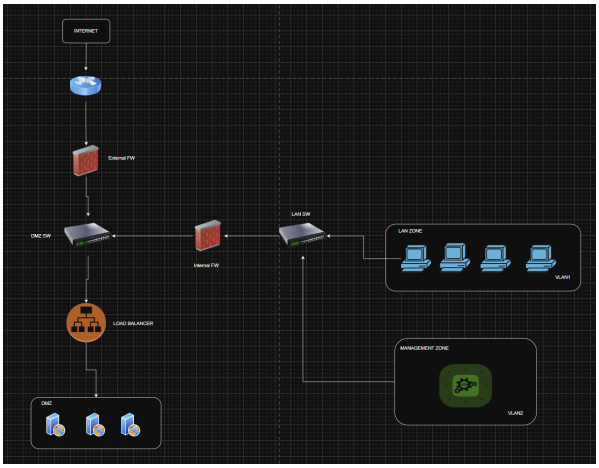
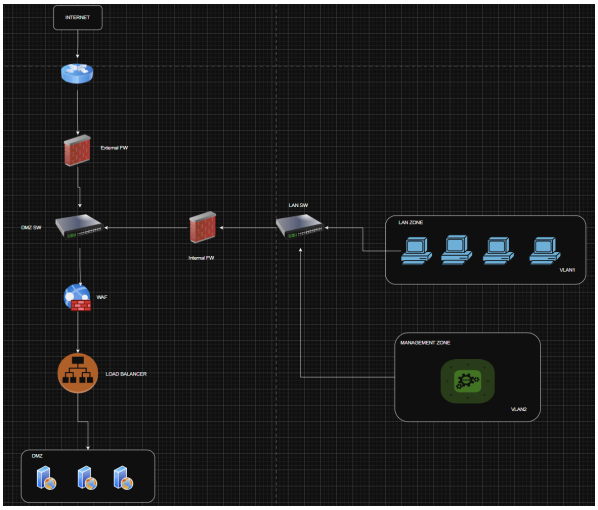
Triển khai hệ thống WAF

1. Mô hình triển khai



Infra ban đầu



Infra cải tiến

---

2. Thực hiện cài đặt

2.1. Web Server

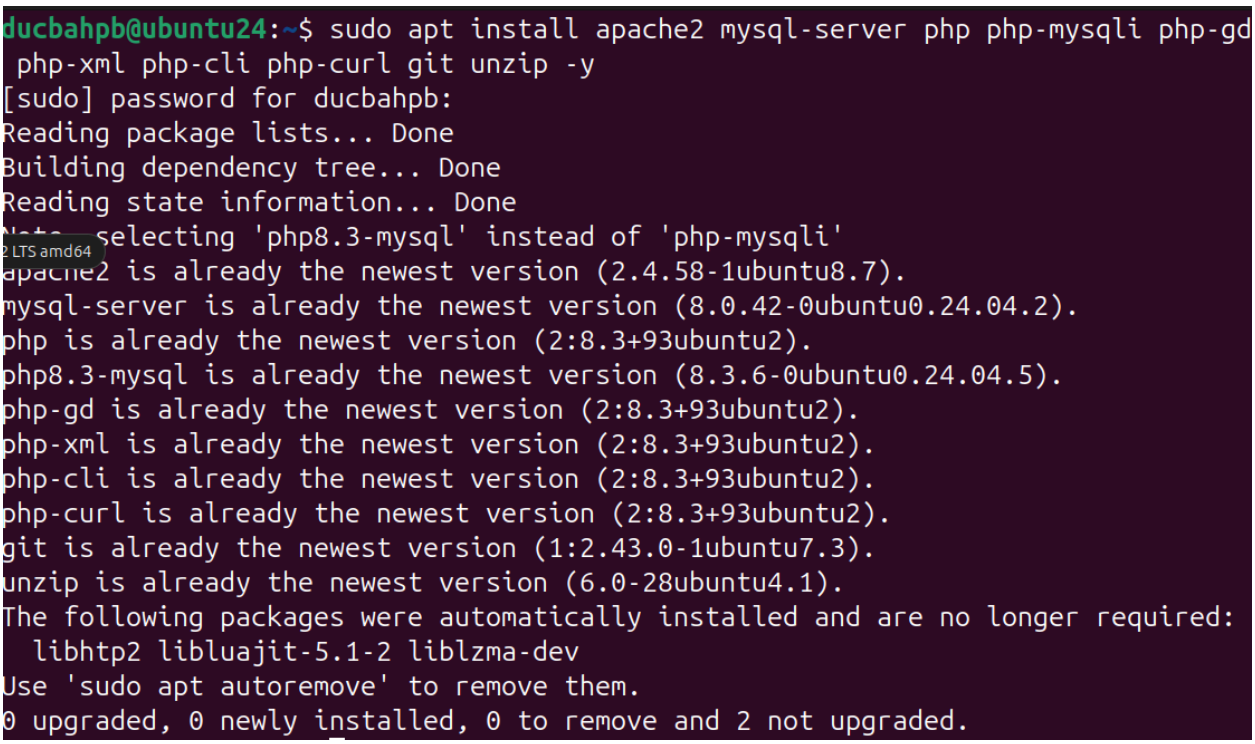
Cài đặt DVWA trên ubuntu 24.04

Bước 1: Cập nhật hệ thống

```
sudo apt update && sudo apt upgrade -y
```

Bước 2: Cài đặt Apache, MySQL và PHP

```
sudo apt install apache2 mysql-server php php-mysqli php-gd php-xml php-cli php-curl git unzip -y
```



Bước 3: Cấu hình MySQL cho DVWA

```
sudo mysql
```

```
CREATE DATABASE dvwa;
CREATE USER 'dvwauser'@'localhost' IDENTIFIED BY 'dvwasecret';
GRANT ALL PRIVILEGES ON dvwa.* TO 'dvwauser'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

Bước 4: Tải và cài đặt DVWA

1. Clone từ GitHub

```
cd /var/www/html
sudo git clone https://github.com/digininja/DVWA.git
sudo chown -R www-data:www-data DVWA
```

```
ducbahpb@ubuntu24:~$ sudo git clone https://github.com/digininja/DVWA.git
Cloning into 'DVWA'...
remote: Enumerating objects: 5373, done.
remote: Total 5373 (delta 0), reused 0 (delta 0), pack-reused 5373 (from 1)
Receiving objects: 100% (5373/5373), 2.58 MiB | 4.33 MiB/s, done.
Resolving deltas: 100% (2667/2667), done.
```

2. Cấu hình

```
cd DVWA/config
sudo cp config.inc.php.dist config.inc.php
sudo nano config.inc.php
```

3. Sửa phần MySQL

```
$_DVWA[ 'db_user' ] = 'dvwauser';
$_DVWA[ 'db_password' ] = 'dvwasecret';
$_DVWA[ 'db_database' ] = 'dvwa';
```

```
<?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' ]   = getenv('DB_SERVER') ?: '127.0.0.1';
$_DVWA[ 'db_user' ] = 'dvwauser';
$_DVWA[ 'db_password' ] = 'dvwasecret';
$_DVWA[ 'db_database' ] = 'dvwa';

$_DVWA[ 'db_port']      = getenv('DB_PORT') ?: '3306';
```

- Lưu lại và thoát ra

Bước 5: Khởi động lại Apache

```
sudo systemctl restart apache2
```

2.1. Modsecurity+nginx WAF

Bước 1: Cập nhật hệ thống và cài đặt phụ thuộc

```
apt update
apt upgrade -y
```

```
sudo apt install g++ flex bison curl apache2-dev doxygen libyajl-dev ssdeep li
blua5.2-dev libgeoip-dev libtool dh-autoreconf libcurl4-gnutls-dev libxml2 lib
xml2-dev git liblmdb-dev libpkgconf3 lmdb-doc pkgconf zlib1g-dev libssl-dev
libpcre3-dev -y
```

- Cài đặt các gói thư viện cần thiết để biên dịch và xây dựng ModSecurity từ mã nguồn.

Bước 2: Cài đặt ModSecurity v3

```
wget https://github.com/SpiderLabs/ModSecurity/releases/download/v3.0.8/
modsecurity-v3.0.8.tar.gz
tar -xvzf modsecurity-v3.0.8.tar.gz
cd modsecurity-v3.0.8
./build.sh
./configure
make
make install
```

Bước 3: Cài đặt module kết nối Nginx với Modsecurity

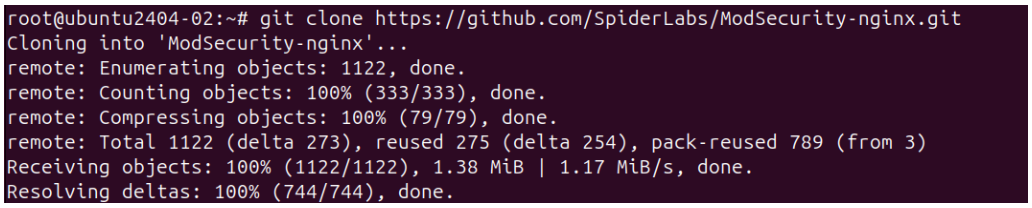
```
git clone https://github.com/SpiderLabs/ModSecurity-nginx.git
```

- Tải module bridge ModSecurity-nginx dùng để tích hợp ModSecurity v3 vào Nginx.

Bước 4: Tải, biên dịch và cài đặt Nginx

```
wget https://nginx.org/download/nginx-1.20.2.tar.gz
tar xzf nginx-1.20.2.tar.gz
useradd -r -M -s /sbin/nologin -d /usr/local/nginx nginx
```

- Tải source code Nginx và tạo user riêng để chạy Nginx an toàn hơn.




```
cd nginx-1.20.2
./configure --user=nginx --group=nginx --with-pcre-jit --with-debug --with-c
ompat --with-http_ssl_module --with-http_realip_module --add-dynamic-mod
ule=/root/ModSecurity-nginx --http-log-path=/var/log/nginx/access.log --err
or-log-path=/var/log/nginx/error.log
make
make modules
make install
```

- Cấu hình Nginx và biên dịch cùng module ModSecurity.
- `make modules`: tạo file `.so` (shared object).
- `make install`: cài Nginx vào `/usr/local/nginx`.

```
ln -s /usr/local/nginx/sbin/nginx /usr/local/sbin/
```

- Tạo symlink để gọi `nginx` từ bất kỳ đâu.

```
root@ubuntu2404-02:~/nginx-1.20.2# nginx -V
nginx version: nginx/1.20.2
built by gcc 13.3.0 (Ubuntu 13.3.0-6ubuntu2~24.04)
built with OpenSSL 3.0.13 30 Jan 2024
TLS SNI support enabled
configure arguments: --user=nginx --group=nginx --with-pcre-jit --with-debug --with-compat -
-with-http_ssl_module --with-http_realip_module --add-dynamic-module=/root/ModSecurity-nginx
 --http-log-path=/var/log/nginx/access.log --error-log-path=/var/log/nginx/error.log
```

Bước 5: Cấu hình ModSecurity

```
cp modsecurity.conf-recommended /usr/local/nginx/conf/modsecurity.conf
cp unicode.mapping /usr/local/nginx/conf/
sed -i 's/SecRuleEngine DetectionOnly/SecRuleEngine On/' modsecurity.conf
```

- Dùng file cấu hình mẫu và bật chế độ **chặn thực sự (On)** thay vì chỉ ghi log (`DetectionOnly`).

Bước 6: Cài đặt OWASP CRS (Core Rule Set)

```
git clone https://github.com/SpiderLabs/owasp-modsecurity-crs.git /usr/local/
nginx/conf/owasp-crs
cp crs-setup.conf{.example,}
```

- Tải và cấu hình bộ quy tắc OWASP để bảo vệ chống lại các kiểu tấn công web phổ biến như XSS, SQLi, RCE...

```
root@ubuntu2404-02:~# git clone https://github.com/SpiderLabs/owasp-modsecurity-crs.git /usr
/local/nginx/conf/owasp-crs
Cloning into '/usr/local/nginx/conf/owasp-crs'...
remote: Enumerating objects: 10486, done.
remote: Total 10486 (delta 0), reused 0 (delta 0), pack-reused 10486 (from 1)
Receiving objects: 100% (10486/10486), 3.33 MiB | 5.66 MiB/s, done.
Resolving deltas: 100% (7687/7687), done.
```

```
echo -e "Include owasp-crs/crs-setup.conf \nInclude owasp-crs/rules/*.conf"
>> /usr/local/nginx/conf/modsecurity.conf
```

- Bao gồm các file rule vào ModSecurity.

```
root@ubuntu2404-02:~# echo -e "Include owasp-crs/crs-setup.conf
Include owasp-crs/rules/*.conf">> /usr/local/nginx/conf/modsecurity.conf
root@ubuntu2404-02:~# nginx -t
nginx: the configuration file /usr/local/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/nginx/conf/nginx.conf test is successful
```

Bước 7: Cấu hình file nginx.conf

```
nano /usr/local/nginx/conf/nginx.conf
```

```
load_module modules/ngx_http_modsecurity_module.so;

user nginx;
worker_processes 1;
pid /run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include       mime.types;
    default_type  application/octet-stream;
    sendfile        on;
    keepalive_timeout 65;

    server {
        listen 80;
        server_name waf.dvwa.local;

        modsecurity on;
        modsecurity_rules_file /usr/local/nginx/conf/modsecurity.conf;

        access_log /var/log/nginx/access.log;
        error_log  /var/log/nginx/error.log;

        location / {
```

```
        proxy_pass http://10.0.0.100;
        proxy_http_version 1.1;

        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
        root /usr/share/nginx/html;
    }
  }
}
```

- xóa các dòng mặc định và thay thế bằng những dòng này

Bước 8: Thiết lập Nginx như 1 dịch vụ systemd

```
nano /etc/systemd/system/nginx.service
```

```
[Unit]
Description=A high performance web server and a reverse proxy server
Documentation=man:nginx(8)
After=network.target nss-lookup.target

[Service]
Type=forking
PIDFile=/run/nginx.pid
ExecStartPre=/usr/local/nginx/sbin/nginx -t -q -g 'daemon on; master_proces
s on;'
ExecStart=/usr/local/nginx/sbin/nginx -g 'daemon on; master_process on;'
ExecReload=/usr/local/nginx/sbin/nginx -g 'daemon on; master_process on;' -
s reload
```

```
ExecStop=-/sbin/start-stop-daemon --quiet --stop --retry QUIT/5 --pidfile /ru
n/nginx.pid
TimeoutStopSec=5
KillMode=mixed

[Install]
WantedBy=multi-user.target
```

- Thêm các dòng này vào file.

```
systemctl daemon-reload
systemctl start nginx
systemctl enable nginx
```

- kích hoạt và cho Nginx tự chạy khi khởi động hệ thống
