# Часть 1: Настройка Apache с HTTPS

## Шаг 1: Установка Apache

Обновление пакетов системы

sudo apt update && sudo apt upgrade -y

Установка Apache

sudo apt install apache2 -y

## Шаг 2: Проверка работы Apache

Проверка статуса службы Apache

sudo systemctl status apache2.service

Запуск Apache (если не запущен)

sudo systemctl start apache2

Включение автозапуска при загрузке системы

sudo systemctl enable apache2

Проверка работы через браузер или curl

curl http://localhost

## Шаг 3: Настройка брандмауэра

Разрешение HTTP и HTTPS трафика

sudo ufw allow 'Apache Full'

Проверка правил брандмауэра

sudo ufw status

## Шаг 4: Создание самоподписанного SSL-сертификата

Создание директории для сертификатов

sudo mkdir /etc/apache2/ssl

Генерация самоподписанного сертификата

sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 \

-keyout /etc/apache2/ssl/apache-selfsigned.key \

-out /etc/apache2/ssl/apache-selfsigned.crt

Во время генерации ответьте на вопросы:

Country Name (2 letter code) [XX]: RU

State or Province Name (full name) []: Moscow

Locality Name (eg, city) [Default City]: Moscow

Organization Name (eg, company) [Default Company Ltd]: MyCompany

Organizational Unit Name (eg, section) []: IT

Common Name (eg, your name or your server's hostname) []: your-server-ip-or-domain

Email Address []: admin@example.com

## Шаг 5: Настройка SSL в Apache

Включение модуля SSL

sudo a2enmod ssl

Включение модуля rewrite для перенаправления HTTP на HTTPS

sudo a2enmod rewrite

Создание виртуального хоста для HTTPS

sudo nano /etc/apache2/sites-available/default-ssl.conf

Добавьте следующую конфигурацию в файл:

<IfModule mod\_ssl.c>

<VirtualHost \_default\_:443>

ServerAdmin admin@example.com

ServerName your-server-ip-or-domain

DocumentRoot /var/www/html

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

SSLEngine on

SSLCertificateFile /etc/apache2/ssl/apache-selfsigned.crt

SSLCertificateKeyFile /etc/apache2/ssl/apache-selfsigned.key

<FilesMatch "\.(cgi|shtml|phtml|php)$">

SSLOptions +StdEnvVars

</FilesMatch>

<Directory /usr/lib/cgi-bin>

SSLOptions +StdEnvVars

</Directory>

</VirtualHost>

</IfModule>

## Шаг 6: Активация SSL сайта и перенаправление HTTP на HTTPS

Активация SSL виртуального хоста

sudo a2ensite default-ssl.conf

Настройка перенаправления HTTP на HTTPS

sudo nano /etc/apache2/sites-available/000-default.conf

Добавьте в файл после директивы DocumentRoot:

RewriteEngine on

RewriteCond %{SERVER\_NAME} =your-server-ip-or-domain

RewriteRule ^ https://%{SERVER\_NAME}%{REQUEST\_URI} [END,NE,R=permanent]

## Шаг 7: Перезапуск Apache

Проверка конфигурации

sudo apache2ctl configtest

Перезапуск Apache

sudo systemctl restart apache2

Проверка работы HTTPS

curl -k https://localhost

# Часть 2: Настройка Nginx с прокси и ModSecurity

## Шаг 1: Установка Nginx

Установка Nginx

sudo apt install nginx -y

Проверка статуса службы

sudo systemctl status nginx

Запуск и включение автозапуска

sudo systemctl start nginx

sudo systemctl enable nginx

## Шаг 2: Настройка брандмауэра для Nginx

Разрешение Nginx трафика

sudo ufw allow 'Nginx Full'

Проверка статуса

sudo ufw status

## Шаг 3: Создание SSL-сертификата для Nginx

Создание директории для сертификатов Nginx

sudo mkdir /etc/nginx/ssl

Генерация самоподписанного сертификата

sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 \

-keyout /etc/nginx/ssl/nginx-selfsigned.key \

-out /etc/nginx/ssl/nginx-selfsigned.crt

## Шаг 4: Настройка прямого прокси в Nginx

Создание конфигурации прямого прокси

sudo nano /etc/nginx/sites-available/forward-proxy

Добавьте конфигурацию:

server {

listen 8080;

location / {

resolver 8.8.8.8;

proxy\_pass http://$http\_host$uri$is\_args$args;

proxy\_set\_header Host $http\_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;

}

}

## Шаг 5: Настройка обратного прокси с HTTPS

Создание конфигурации обратного прокси

sudo nano /etc/nginx/sites-available/reverse-proxy

Добавьте конфигурацию:

HTTP перенаправление на HTTPS

server {

listen 80;

server\_name your-server-ip-or-domain;

return 301 https://$server\_name$request\_uri;

}

HTTPS сервер с обратным прокси

server {

listen 443 ssl;

server\_name your-server-ip-or-domain;

ssl\_certificate /etc/nginx/ssl/nginx-selfsigned.crt;

ssl\_certificate\_key /etc/nginx/ssl/nginx-selfsigned.key;

Безопасность SSL

ssl\_protocols TLSv1.2 TLSv1.3;

ssl\_ciphers ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384;

ssl\_prefer\_server\_ciphers off;

location / {

Проксирование на другой сервер (например, Apache на порту 8080)

proxy\_pass http://localhost:8080;

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;

Дополнительные настройки безопасности

proxy\_hide\_header X-Powered-By;

add\_header X-Content-Type-Options nosniff;

add\_header X-Frame-Options DENY;

add\_header X-XSS-Protection "1; mode=block";

}

Статические файлы

location /static/ {

alias /var/www/html/static/;

expires 1y;

add\_header Cache-Control "public, immutable";

}

}

## Шаг 6: Активация конфигураций Nginx

Активация сайтов

sudo ln -s /etc/nginx/sites-available/forward-proxy /etc/nginx/sites-enabled/

sudo ln -s /etc/nginx/sites-available/reverse-proxy /etc/nginx/sites-enabled/

Удаление дефолтной конфигурации

sudo rm /etc/nginx/sites-enabled/default

Проверка конфигурации

sudo nginx -t

Перезагрузка Nginx

sudo systemctl reload nginx

## Шаг 7: Установка и настройка ModSecurity

Установка ModSecurity для Nginx

sudo apt install libnginx-mod-http-modsecurity -y

Проверка установки модуля

nginx -V 2>&1 | grep -o with-http\_modsecurity\_module

Скачивание правил OWASP ModSecurity Core Rule Set (CRS)

sudo apt install git -y

sudo git clone https://github.com/coreruleset/coreruleset /etc/nginx/modsecurity/coreruleset

Создание конфигурации ModSecurity

sudo mkdir -p /etc/nginx/modsecurity

sudo cp /etc/modsecurity/modsecurity.conf-recommended /etc/nginx/modsecurity/modsecurity.conf

## Шаг 8: Настройка ModSecurity

Редактирование основного конфигурационного файла

sudo nano /etc/nginx/modsecurity/modsecurity.conf

Измените следующие параметры:

SecRuleEngine On

SecAuditEngine RelevantOnly

SecAuditLog /var/log/nginx/modsec\_audit.log

SecDebugLog /var/log/nginx/modsec\_debug.log

SecAuditLogType Serial

SecAuditLogStorageDir /var/log/nginx/modsec\_audit/

## Шаг 9: Настройка правил CRS

Копирование необходимых файлов правил

sudo cp /etc/nginx/modsecurity/coreruleset/crs-setup.conf.example /etc/nginx/modsecurity/coreruleset/crs-setup.conf

sudo cp /etc/nginx/modsecurity/coreruleset/rules/REQUEST-942-APPLICATION-ATTACK-SQLI.conf /etc/nginx/modsecurity/coreruleset/

Создание конфигурации для включения ModSecurity в Nginx

sudo nano /etc/nginx/conf.d/modsecurity.conf

Добавьте содержимое:

modsecurity on;

modsecurity\_rules\_file /etc/nginx/modsecurity/modsecurity.conf;

Включение основных правил CRS

Include /etc/nginx/modsecurity/coreruleset/crs-setup.conf

Include /etc/nginx/modsecurity/coreruleset/rules/REQUEST-942-APPLICATION-ATTACK-SQLI.conf

## Шаг 10: Настройка виртуального хоста с ModSecurity

Обновите конфигурацию обратного прокси:

sudo nano /etc/nginx/sites-available/reverse-proxy

Добавьте в server блок:

Включение ModSecurity

modsecurity on;

modsecurity\_rules\_file /etc/nginx/conf.d/modsecurity.conf;

Логи ModSecurity

location /modsec\_logs {

alias /var/log/nginx/modsec\_audit;

autoindex on;

}

## Шаг 11: Создание тестового приложения

Создание тестовой PHP страницы для демонстрации SQLi

sudo apt install php-fpm -y

sudo mkdir -p /var/www/html/test

sudo nano /var/www/html/test/index.php

Добавьте простой PHP код:

<?php

// Простая форма для тестирования

if ($\_POST) {

$input = $\_POST['search'];

echo "Вы ввели: " . htmlspecialchars($input);

}

?>

<form method="post">

<input type="text" name="search" placeholder="Введите поисковый запрос">

<input type="submit" value="Поиск">

</form>

## Шаг 12: Демонстрация работы ModSecurity

Тест 1: Проверка блокировки SQL-инъекции

Отправка SQL-инъекции - должен вернуть 403 Forbidden

curl -X POST http://your-server-ip/test/index.php \

-d "search=' OR 1=1--" \

-H "Content-Type: application/x-www-form-urlencoded"

Или с использованием HTTPS

curl -k -X POST https://your-server-ip/test/index.php \

-d "search=' UNION SELECT username, password FROM users--" \

-H "Content-Type: application/x-www-form-urlencoded"

Тест 2: Проверка нормального запроса

Нормальный запрос - должен вернуть 200 OK

curl -X POST http://your-server-ip/test/index.php \

-d "search=normal query" \

-H "Content-Type: application/x-www-form-urlencoded"

## Шаг 13: Отключение правила и проверка

Найти ID правила SQL-инъекции:

Просмотр логов ModSecurity

sudo tail -f /var/log/nginx/modsec\_audit.log

Или поиск в файлах правил

sudo grep -r "SQL" /etc/nginx/modsecurity/coreruleset/rules/

Закомментировать правило:

Редактирование файла правил SQL-инъекций

sudo nano /etc/nginx/modsecurity/coreruleset/rules/REQUEST-942-APPLICATION-ATTACK-SQLI.conf

Найдите правило, которое блокирует запрос (обычно по ID, например 942100)

и закомментируйте его, добавив в начале строки:

SecRule REQUEST\_COOKIES|!REQUEST\_COOKIES:/\_\_utm/|REQUEST\_COOKIES\_NAMES|REQUEST\_FILENAME|REQUEST\_HEADERS:User-Agent|REQUEST\_HEADERS:Referer|ARGS\_NAMES|ARGS|XML:/\* "@detectSQLi" \

"id:942100,...

Перезагрузка Nginx и повторная проверка:

Перезагрузка Nginx

sudo systemctl reload nginx

Повторная отправка SQL-инъекции - теперь должен вернуть 200 OK

curl -X POST http://your-server-ip/test/index.php \

-d "search=' OR 1=1--" \

-H "Content-Type: application/x-www-form-urlencoded"

## Шаг 14: Просмотр логов

Просмотр логов Nginx

sudo tail -f /var/log/nginx/access.log

sudo tail -f /var/log/nginx/error.log

Просмотр логов ModSecurity

sudo tail -f /var/log/nginx/modsec\_audit.log

Финальная проверка

Проверка всех служб:

Проверка статуса всех служб

sudo systemctl status apache2

sudo systemctl status nginx

sudo systemctl status php8.1-fpm или ваша версия PHP

Проверка открытых портов

sudo netstat -tulpn | grep -E '(80|443|8080)'

Проверка SSL сертификатов

openssl x509 -in /etc/apache2/ssl/apache-selfsigned.crt -text -noout

openssl x509 -in /etc/nginx/ssl/nginx-selfsigned.crt -text -noout

Этот план предоставляет полное пошаговое руководство по настройке Apache и Nginx с HTTPS, прокси-серверами и ModSecurity для защиты от SQL-инъекций на Ubuntu Server 24.04 LTS.