TRƯỜNG ĐẠI HỌC ĐÀ LẠT

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO NHIỆM VỤ 4**

**NHÓM: 1**

Giáo viên hướng dẫn: Thầy Nguyễn Văn Huy Dũng

Sinh viên thực hiện: 2111903 – Nguyễn Nhật Trường

2111860 – Nguyễn Hoàng Bảo Long

2111886 – Nguyễn Quốc Thắng

***Đà Lạt, tháng 8 năm 2024***

**MỤC LỤC**

[**Chương 1. Tại sao phải cần tối ưu hoá hệ thống hosting** 3](#_Toc181140129)

[**1.1.** **Hosting là gì?** 3](#_Toc181140130)

[**1.2.** **Cách hoạt động** 3](#_Toc181140131)

[**Chương 2. Cấu hình, thiết lập và tối ưu hệ thống** 4](#_Toc181140132)

[**2.1.**  **Tối ưu hoá Apache** 4](#_Toc181140133)

[**2.1.1. Mod\_deflate** 4](#_Toc181140134)

[**2.1.2. Cấu hình cài đặt KeepAlive** 5](#_Toc181140135)

[**2.1.3. Điều chỉnh thông số MPM** 6](#_Toc181140136)

[**2.1.4. Tối ưu hoá .htaccess** 7](#_Toc181140137)

[**2.1.5. Bật caching** 8](#_Toc181140138)

[**2.2.**  **Tối ưu hoá PHP** 9](#_Toc181140139)

[**2.3.** **Tối ưu hóa Mariadb** 11](#_Toc181140140)

[**2.4.** **Thay đổi port của phpmyadmin** 13](#_Toc181140141)

[**2.5.** **Cài FTP cho LAMP** 14](#_Toc181140142)

# **Chương 1. Tại sao phải cần tối ưu hoá hệ thống hosting**

## **1.1. Hosting là gì?**

- Hosting hay Web hosting là không gian để lưu trữ dữ liệu, được chia nhỏ từ server giúp đẩy toàn bộ dữ liệu hoặc các ứng dụng lên trên internet. Khi sử dụng một số hệ thống hosting đặt lên server của một đơn vị cung cấp các tệp dữ liệu cần thiết để website hay ứng dụng online có thể chạy được.

- Những nhà cung cấp hosting là người sẽ chịu trách nhiệm đảm bảo các máy chủ hoạt động liên tục. Đồng thời cũng sẽ đảm bảo việc cung cấp hosting luôn diễn ra suôn sẻ. Ngoài ra, chúng sẽ chống xâm nhập máy chủ, đồng thời xử lý dữ liệu từ hosting tốt nhất đến trình duyệt cho người dùng.

## **1.2. Cách hoạt động**

- Về phía nhà cung cấp dịch vụ sẽ chuẩn bị server lưu trữ cho người sử dụng hosting. Bằng cách chia sẻ các tài nguyên trên server thành các không gian lưu trữ nhỏ hơn. Những không gian lưu trữ này gọi là các hosting.

- Tuỳ theo yêu cầu của người dùng cần mở rộng hay thu hẹp các gói hosting, thì nhà cung cấp sẽ điều chỉnh những thông số cần thiết.

- Về phía người dùng thì chỉ cần thuê hosting. Người dùng hoàn toàn có thể truy cập hosting từ các thiết bị kết nối internet khác, thông qua việc gửi yêu cầu đến tên miền hoặc địa chỉ IP của hosting.

- Hosting sẽ trả về cho người dùng những tập tin được yêu cầu tương ứng. Trong quá trình sử dụng, nếu người dùng có nhu cầu mở rộng gói hosting, người dùng phải liên hệ đến nhà cung cấp để chỉnh sửa cấu hình.

# **Chương 2. Cấu hình, thiết lập và tối ưu hệ thống**

## **2.1. Tối ưu hoá Apache**

- Apache được tối ưu hoá để cải thiện hiệu suất máy chủ web của mình. Một trong những cách phổ biến là tối ưu hoá hiệu suất là sử dụng caching.

- Caching cho phép máy chủ web lưu trữ các tài nguyên trên máy chủ của nó để truy cập nhanh hơn cho các yêu cầu tiếp theo. Apache cung cấp nhiều module cache khác nhau, cho phép lưu trữ các tài nguyên như HTML, CSS và JavaScript.

### **2.1.1. Mod\_deflate**

- Module mod\_deflate cho phép Apache web server nén dữ liệu trước khi gửi lại cho trình duyệt. Việc nén dữ liệu giúp tăng tốc độ truy câoj vào website và giảm băng thông truy cập.

A diagram of a computer program

Description automatically generated

- Trước khi gửi dữ liệu lại cho trình duyệt thì web server thực hiện nén dữ liệu. Việc nén dữ liệu giảm kích thước của website do đso website của bạn sẽ được tải nhanh hơn đồng thời tiết kiệm băng thông cho bạn. Mặc định mod\_deflate đã được bật trên các server.

Bước 1: Bật mod\_deflate

sudo a2enmod deflate

A black background with white text

Description automatically generated

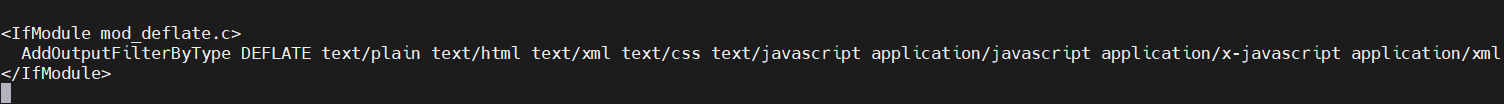
Bước 2: Cấu hình file deflate.conf

Nano /etc/apache2/mode-enabled/deflate.conf

<IfModule mod\_deflate.c>

AddOutputFilterByType DEFLATE text/plain text/html text/xml text/css text/javascript application/javascript application/x-javascript application/xml

</IfModule>



### **2.1.2. Cấu hình cài đặt KeepAlive**

- Ngoài ra, ta có thể tối ưu hoá hiệu suất bằng cách cấu hình các tuỳ chọn như KeepAlive, thời gian chờ và thời gian kết nối tối đa.

Bước 1: Chỉnh sửa file cấu hình apache2.conf

sudo nano /etc/apache2/apache2.conf

KeepAlive On #bật module keepalive.

KeepAliveTimeout 5 #Số giây để chờ yêu cầu tiếp theo từ cùng một client cho cùng một kết nối.

MaxKeepAliveRequests 100 #thời gian mà số lượng tối đa yêu cầu kết nối.

A screenshot of a computer screen

Description automatically generated

### **2.1.3. Điều chỉnh thông số MPM**

- Đây là MPM mặc định khi bạn cài đặt Apache. Các tiến trình con của Apache được chạy thì trong mỗi tiến trình đó chỉ có một luồng chạy để lắng nghe các yêu cầu kết nối gửi đến, cũng như xử lý các kết nối đó.

- Do cách hoạt động đó – mỗi yêu cầu được xử lý trên một tiến trình riêng – độc lập, mà Apache trên server cần nhiều tài nguyên hơn các MPM khác. Tuy nhiên nó vẫn được sử dụng, có thể vì yêu cầu an toàn nào đó hoặc bắt buộc Apache phải sử dụng loại thư viện không đa luồng như mod\_php mà MPM prefork vẫn dùng được.

Bước 1: Chỉnh sửa file cấu hình mpm\_prefork.conf

Nano /etc/apache2/mode-available/mpm\_prefork.conf

StartServers 5 #thiết lập khởi tạo sẵn khi hệ thống khởi động số process chờ request gửi đến.

MinSpareServers 5 #thiết lập số lượng process apache nhỏ nhất có thể chờ, lượng process chờ này do Apache tự tạo ra hay kill nó tuỳ thuộc vào nó đánh giá số lượng request gửi đến.

MaxSpareServers 10 #thiết lập số lượng process apache lớn nhất có thể chờ, lượng process chờ này do Apache tự tạo ra hay kill nó tuỳ thuộc vào nó đánh giá số lượng request gửi đến.

MaxRequestWorkers 150 #thiết lập số lượng request cùng lúc được phục vụ

MaxConnectionsPerChild 0 #thiết lập số lượng máy chủ con sẽ xử lý

A screenshot of a computer program

Description automatically generated

### **2.1.4. Tối ưu hoá .htaccess**

- Tập tin .htaccess (hypertext access) là một file có ở thư mục gốc của các hosting và do apache quản lý và cấp quyền. File này nó thể điều khiển, cấu hình được nhiều thứ với đa dạng các thông số, nó có thể thay đổi được các giá trị được set mặc định của Apache.

Bước 1: nếu có thể hãy tắt .htaccess để có được trải nghiệm tốt hơn

Nano /var/www/html/nhiemvu4.ctk45mmt/.htaccess

<Directory /var/www/html>

AllowOverride None

</Directory>

A screen shot of a computer

Description automatically generated

### **2.1.5. Bật caching**

Bước 1: Bật module cache và cache\_disk

sudo a2enmod cache

sudo a2enmod cache\_disk

A black screen with white text

Description automatically generated  
A screen shot of a computer

Description automatically generated

Bước 2: Chỉnh sửa file cấu hình cache\_disk.conf

sudo nano /etc/apache2/mods-enabled/cache\_disk.conf

<IfModule mod\_cache.c>

<IfModule mod\_cache\_disk.c>

CacheRoot "/var/cache/apache2/mod\_cache\_disk" #phải để thư mục này giống với file /etc/default/apache2

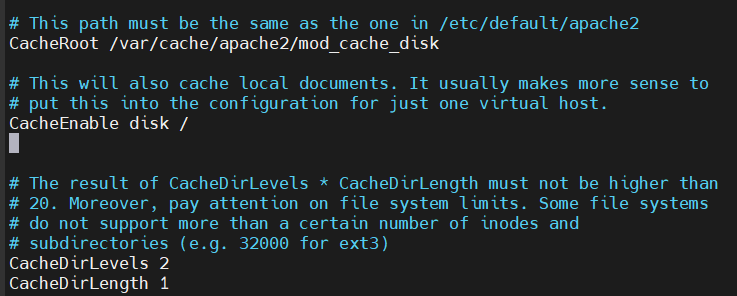
CacheEnable disk / #đây cũng là thư mục để lưu trữ, thông thường sẽ thuận tiện hơn nếu bật chung với máy chủ ảo

CacheDirLevels 2 #số cấp thư mục con trong thư mục

CacheDirLength 1 #số ký tự trong thư mục con

</IfModule>

</IfModule>



## **2.2. Tối ưu hoá PHP**

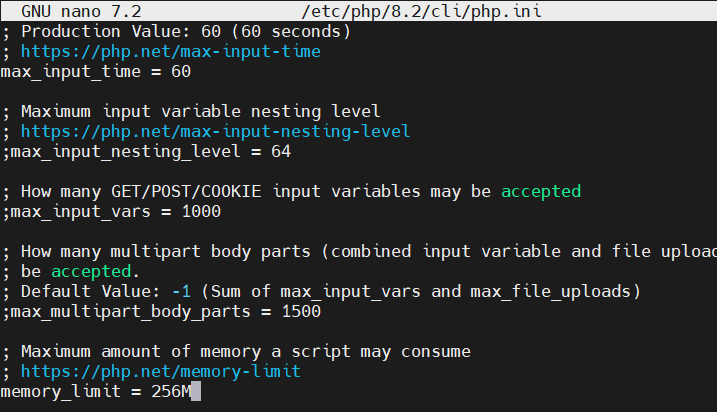
Bước 1. Backup php

sudo cp /etc/php/8.2/cli/php.ini /etc/backup/php.ini.backup



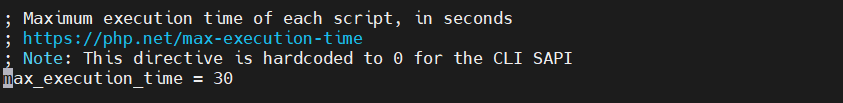
Bước 2. Chỉnh sửa file cấu hình php.ini

sudo nano /etc/php/8.2/cli/php.ini



A screenshot of a computer program

Description automatically generated

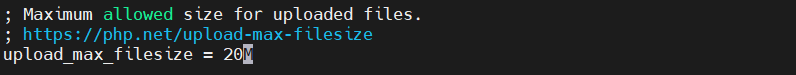


A screen shot of a computer

Description automatically generated

A black screen with white text

Description automatically generated



memory\_limit = 256M // Quy định mức tối đa bộ nhớ mà một script PHP có thể sử dụng.

opcache.enable=1 // OPcache là một bộ nhớ đệm mã bytecode cho PHP, giúp tăng tốc độ thực thi bằng cách lưu trữ mã bytecode đã biên dịch, giảm thiểu việc biên dịch lại.

opcache.memory\_consumption=128 //Lượng bộ nhớ để lưu trữ bytecode đã biên dịch.

opcache.interned\_strings\_buffer=8 //Lượng bộ nhớ cho các chuỗi thường xuyên sử dụng và không thay đổi.

opcache.max\_accelerated\_files=4000 // Quy định số lượng tệp mã bytecode PHP được OPcache lưu trữ

opcache.revalidate\_freq=60 // Quy định sau bao nhiêu giây OPcache sẽ kiểm tra xem các tệp PHP có thay đổi không. Cập nhật nội dung mã khi có thay đổi nhưng không giảm hiệu suất.

opcache.fast\_shutdown=1 // Cho phép OPcache giải phóng bộ nhớ nhanh chóng khi kết thúc một script.

max\_execution\_time = 30 // Đặt giới hạn thời gian thực thi cho mỗi script PHP.

realpath\_cache\_size = 4096k // Quy định kích thước bộ nhớ cache để lưu trữ đường dẫn thực của tệp.

realpath\_cache\_ttl = 600 // Đặt TTL cho các mục trong bộ nhớ cache đường dẫn thực. giảm số lần PHP phải xử lý lại đường dẫn thực.

post\_max\_size = 20M // Quy định kích thước tối đa cho dữ liệu nhận qua phương thức POST.

upload\_max\_filesize = 20M // Quy định kích thước tối đa của một tệp tải lên

## **2.3. Tối ưu hóa Mariadb**

Bước 1: backup lại file cấu hình

sudo cp /etc/mysql/mariadb.conf.d/50-server.cnf /etc/mysql/mariadb.conf.d/

Bước 2: Chỉnh sửa file cấu hình

sudo nano /etc/mysql/mariadb.conf.d/50-server.cnf

A screenshot of a computer

Description automatically generated

A black background with blue and white text

Description automatically generated

A black background with blue text

Description automatically generated

key\_buffer\_size: tham số này thiết lập dung lượng bộ nhớ để cache các chỉ mục

max\_connections: quy định số lượng kết nối đồng thời tối đa

slow\_query\_log\_file: đường dẫn tới file log lưu lại các truy vấn chậm

slow\_query\_time: thiết lập thời gian tối đa cho một truy vấn trước khi được xem

innodb\_buffer\_pool\_size: Tham số này thiết lập dung lượng bộ nhớ đệm cho bộ lưu trữ InnoDB.

Bước 3: Khởi động lại MariaDB

Systemctl restart mariadb

## **2.4. Thay đổi port của phpmyadmin**

Bước 1: chỉnh sửa file cấu hình apache

A computer screen with text

Description automatically generated

A screen shot of a computer

Description automatically generated

Bước 2: cấu hình ufw cho phép port 8080

Sudo ufw allow 8080/tcp

## **2.5. Cài FTP cho LAMP**

Bước 1: cài đặt vsftpd



sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.backup

Bước 2: chỉnh sửa file cấu hình vsftpd

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A black screen with blue text

Description automatically generated

A screenshot of a computer program

Description automatically generated

Bước 3: tạo người dùng ftp

A screen shot of a computer program

Description automatically generated

Bước 4: cấp quyền cho người dùng



Bước 5: cho phép ufw thông qua các port 20, 21, 1024, 1048

A black background with white text

Description automatically generated

A computer screen with white text

Description automatically generated

Bước 6: kiểm tra kết quả

A screenshot of a computer

Description automatically generated