Understanding Technology Stacks: LAMP, LEMP, MERN, and More

What is a Stack?

A technology stack, often referred to as a "stack," is a combination of software tools and technologies used together to build and run applications. It typically includes:

- **Operating System** (e.g., Linux, Windows, macOS)
- **Web Server** (e.g., Apache, Nginx)
- **Database** (e.g., MySQL, PostgreSQL, MongoDB)
- **Programming Language/Runtime** (e.g., PHP, Python, JavaScript, Node.js)

Why Do We Use Stacks?

Stacks provide a structured approach to application development and deployment. Benefits include:

- **Compatibility**: Ensuring that different software components work seamlessly together.
- **Scalability**: Making it easier to scale applications as demand increases.
- **Standardization**: Following best practices and industry standards for development.
- **Optimization**: Preconfigured solutions help optimize performance and security.

Common Stacks

Several well-known technology stacks exist, each tailored to specific use cases:

1. LAMP (Linux, Apache, MySQL, PHP)

LAMP is a widely used stack for building dynamic web applications. It consists of:

- **Linux**: The operating system.
- **Apache**: The web server that handles HTTP requests.
- **MySQL**: The database management system.
- **PHP**: The programming language for server-side scripting.

Key Features of LAMP:

- **Widely Used**: A popular choice for web hosting and application deployment.
- **Apache Web Server**: Well-documented, highly configurable, and widely supported.
- **Supports Dynamic Content**: PHP processes server-side scripting efficiently.

Installing LAMP on Ubuntu

```
sudo apt update
sudo apt install apache2 -y
sudo apt install mysql-server -y
sudo mysql_secure_installation
sudo apt install php libapache2-mod-php php-mysql -y
sudo systemctl restart apache2
```

Installing LAMP on AWS Amazon Linux

```
sudo yum update -y
sudo yum install httpd -y
sudo systemctl start httpd
sudo systemctl enable httpd
sudo yum install mariadb-server -y
```

```
sudo systemctl start mariadb
sudo systemctl enable mariadb
sudo mysql_secure_installation
sudo yum install php php-mysqlnd -y
sudo systemctl restart httpd
```

2. LEMP (Linux, Nginx, MySQL, PHP/Python/Perl)

LEMP is similar to LAMP but replaces Apache with **Nginx**, which is known for its high performance and efficient resource utilization. It consists of:

- **Linux**: The operating system.
- **Nginx**: A high-performance web server.
- **MySQL**: The database management system.
- **PHP/Python/Perl**: The backend programming language.

Key Features of LEMP:

- **High Performance**: Nginx handles high concurrent connections better than Apache.
- **Efficient Resource Utilization**: Uses event-driven architecture for better scalability.
- **Optimized for Static Content**: Nginx serves static files faster than Apache.

Installing LEMP on Ubuntu

```
sudo apt update
sudo apt install nginx -y
sudo apt install mysql-server -y
sudo mysql_secure_installation
sudo apt install php-fpm php-mysql -y
sudo systemctl restart nginx
```

Installing LEMP on AWS Amazon Linux

```
sudo yum update -y
sudo amazon-linux-extras enable nginx1
sudo yum install nginx -y
sudo systemctl start nginx
sudo systemctl enable nginx
sudo yum install mariadb-server -y
sudo systemctl start mariadb
sudo systemctl enable mariadb
sudo mysql_secure_installation
sudo yum install php-fpm php-mysqlnd –y
sudo systemctl restart nginx
```

Why Does index.php Get Downloaded Instead of Executed?

By default, browsers only support HTML, CSS, and JavaScript. When a file like index.php is requested, the web server needs to interpret and execute it. If PHP is not properly configured, the server treats it as a regular file and forces a download instead of executing the script. So to solve this issue, we use **PHP-FPM**, which converts PHP code into HTML

3. MERN (MongoDB, Express.js, React, Node.js)

A JavaScript-based stack used for modern web applications:

- **MongoDB**: A NoSQL database.
- **Express.js**: A backend framework for Node.js.
- **React**: A frontend JavaScript library.
- **Node.js**: A JavaScript runtime for server-side applications.

.

LAMP vs. LEMP: Key Differences

Feature	LAMP (Apache)	LEMP (Nginx)
Web Server	Apache	Nginx
Performance	Good for moderate traffic	High performance, handles concurrent users better
Resource Usage	Higher CPU & RAM usage	Lower resource consumption
Configuration	Easier for beginners	More complex setup but optimized for speed
Static File Handling	Slower compared to Nginx	Faster and more efficient
Flexibility	Supports .htaccess for easy configuration	No .htaccess; uses central config files

Conclusion

Understanding technology stacks is crucial for developers and DevOps engineers. Whether using LAMP, LEMP, MERN, or other stacks, knowing how each component interacts helps in building scalable and efficient applications. Proper configuration of tools like PHP-FPM ensures seamless execution of PHP files, avoiding common issues like forced downloads.

Deploying these stacks on AWS provides additional scalability and flexibility, making cloud-based applications more efficient and reliable.