





















Web Servers

Web servers can be of various types. Each one has a specific function, and hence a specific configuration. Let us read about some of the most common web servers.

**Application Server**- This server executes the main business logic of the application. Whenever the user requests for something, the application server runs the code written by the developer.

**Database Server**- A database server is a system where all the data is stored. Whenever the user requests for some data, it is fetched from the database server. The data is stored here in an efficient and secure manner.

**Backup Server**- This server helps us create backups for files, data, etc. This is done to prevent the loss of data in case of an unexpected failure. A backup server can also act like the secondary server, in case the primary server is down.

**DNS Server**- The Domain Name Server manages the domain names and their IP addresses. The main function of a DNS server is to map a domain name to its respective IP address.

**Mail Server**- A mail server is used for sending and receiving emails. Some of the protocols used for this transfer are SMTP, POP, IMAP, etc. The Microsoft Exchange Server is an example of a mail server.

Depending on the size of the web application, all these servers can be present on one physical server or on separate servers.

Web Server Architecture

Now, let’s look at these 5 servers we had read about in the previous chapter.

So, like we said, it is not necessary to have these as 5 different servers, but a combination of these can be present in one physical server, depending on the requirement of the application.

Now, this server will have some architecture that should be appropriate for the kind of functions that the server will perform.

This architecture is called a web server architecture. It is made up of these 5 basic elements. Let’s look at each one of these.

**Server OS**- Just like every computer has an operating system, similarly the computer that hosts the website also needs to have an OS.

Examples are Linux, Windows, IBM AIX, etc.

**Server Software**- We know that every website needs to address the incoming requests of the users. This request could be for a web page in the website, or for any other functionality that the website provides. For this, the server needs to run the code of the website to generate a response for the user. But, to handle all this function, the server needs a software which is called the server software.

Examples are Apache, nginx, IIS, etc.

**Programming Language**- Every website has a backend part which is basically written as lines of code, using a programming language. So, the web server architecture includes a particular programming language that is used to write this code.

Examples are: PHP, Python, Perl, Ruby, ASP (.NET), JSP, etc.

**Database Software**- Every website has users and it stores the information of these users in the database. So your login credentials, your preferences, cart items in case of an e-commerce, or any other details that you provide while accessing a website is stored in the database in a secure and efficient manner. And to access this data from the database, a software is required. This is known as the database software.

Examples are: MySQL, MS SQL, MongoDB, Casandra DB, Postgre SQL, etc.

**Front End Components**- So, we know that every website has a frontend or a user interface, which is what the user sees on the browser while browsing through the website. So, there needs to be a front end language to write the front ends code.

Examples are: HTML, javascript, Jquery, CSS, Bootstrap, etc.

Now, we know that the architecture will contain one of each of these elements. We will look at some of the most common combinations of web server architecture in the next lesson.

Web Server Architecture Combinations

Some of the most common web server architecture combinations are:

(The front end component is not mentioned in any of these architecture combinations.)

**WAMP**- WAMP stands for Windows, Apache, MySQL, PHP.

**LAMP**- LAMP stands for Linux, Apache, MySQL, PHP. It is one of the most frequently used combinations since all the components are available free of cost.

**MAMP**- MAMP stands for Mac, Apache, MySQL, PHP. It is most commonly used for web development and local testing processes by Mac OS based developers.

**XAMPP**- Unlike other web server architectures, XAMPP can be used across any operating system. So the X in XAMPP stands for cross platform. The rest of it stands for Apache, MariaDB and PHP.

**WIMSA**- It is the most commonly used Windows architecture. WIMSA stands for Windows, IIS, MS SQL, ASP.NET.

Some of the other non abbreviated web server architectures are:

Windows, tomcat, JSP, Postgre SQL

PHP, nginx, mongoDB

Python, nginx, mongoDB

To give you a clear picture, the most commonly used OS is Linux. Apache is the most commonly used server software and PHP is the most commonly used server side programming language.

Summary

Now that we have come to the end of this topic, you should be able to:

1. Understand the working of websites

2. Understand common web architectures and differentiate between them

3. Explain the common security misconceptions and the reasons behind them