

NAME: OGUNWALE PROGRESS OBALOLUWA

DEPARTMENT: COMPUTER SCIENCE

MATRIC NO: RUN/CMP/24/18052

COURSE: CSC 828 - INTERNET TECHNOLOGY

LECTURER: DR. ADEPOJU S.A.

1. Introduction

Web servers are fundamental to the functioning of the internet, acting as a bridge between the client and the server by processing requests and serving web content. The selection of a web server can significantly impact the performance, scalability, and flexibility of web applications.

Web servers typically use the HTTP or HTTPS protocol to communicate, and they listen on specific network ports (often port 80 for HTTP and port 443 for HTTPS). They are optimized for delivering static content swiftly and efficiently, though they can also generate dynamic content in collaboration with other back-end services or scripting languages, such as PHP, Python, or Node.js

2. Definition of Web Server

A web server is a software or hardware system that hosts websites and serves requested content over the internet using the Hypertext Transfer Protocol (HTTP). A web server handles HTTP requests from clients (typically web browsers), delivering static content such as HTML, CSS, JavaScript, and images. It manages communication between users and the server-side infrastructure, often by serving files directly from the filesystem.

Web servers perform the following functions:

- Processing client requests.
- Serving static and dynamic web pages.
- Handling secure connections (HTTPS).
- Logging and monitoring requests.
- Organize websites
- Media streaming

3. Dedicated vs. Shared

3.1 Dedicated Server

Dedicated server is a good choice for those who have technical background and can manage/configure the server on their own. Yet, the lack of technical skills should not be an obstacle, since we offer different types of Dedicated server management: User-Responsible, Basic, Complete. You may find more information on each type [here](#). Root access, provided with a Dedicated server, is a huge benefit if you need to make some server modifications.

3.2 Shared Server

Shared servers host multiple websites simultaneously; this means you're splitting the server's resources with other websites at the same time. These resources include

- CPU power
- Memory
- Bandwidth

This can pose a problem if your website suddenly gets a lot of traffic. Since the server is already supporting multiple sites, a big traffic spike could slow down or crash your site. Moreover, if one site uses too many resources, it negatively impacts your website's performance.

3.2 Architecture of Modern Web Servers

Modern web servers can adopt various architectures, including but not limited to:

- Client-server architecture.**

Client-server architecture is a fundamental concept in system design where a network involves multiple clients and a server. Clients are devices or programs that request services or resources, while the

server is a powerful machine providing these resources or services. This architecture allows efficient data management and resource sharing, making it popular in web applications, databases, and other network-based systems. By separating roles and distributing tasks, client-server architecture enhances performance, scalability, and security.

- **Microservices architecture.**
- **Load balancing configurations.**

4. Types of Web Servers

4.1 Apache HTTP Server

The Apache HTTP Server, developed by the Apache Software Foundation, is one of the oldest and widely used web servers in the world. It is an open-source, cross-platform solution that supports Linux, Windows, and MacOS. Apache supports virtual hosting, which makes it suitable for hosting multiple websites on a single server. It is commonly used to host dynamic content management systems such as WordPress.

Advantages

- Highly customizable through modules.
- Strong community support.

Disadvantages

- It can be resource-intensive, leading to slower performance compared to newer servers.

4.2 Nginx

Nginx is a high-performance web server and reverse proxy known for its speed, efficiency, and ability to handle a large number of concurrent connections. The high-performance web server Nginx (pronounced "Engine-X") is renowned for its quickness, scalability,

and effective management of several connections at once. It was created by **Igor Sysoev** and when it was first made available in 2004. Unlike Apache, which uses a process-based model, Nginx employs an event-driven, asynchronous architecture. This allows it to serve static content rapidly and act as a load balancer or reverse proxy for application servers. Nginx is cross-platform and has become popular for use in cloud-based environments and micro-services architectures due to its low resource consumption and ability to efficiently distribute traffic.

Advantages

- Excellent for handling concurrent connections and serving static content.

Disadvantages

- Configuration can be complex for newer users.

4.3 Microsoft Internet Information Services (IIS)

Microsoft created the IIS web server, which is intended to function with Windows Server settings. It was created by Microsoft and was made available as a web server for Windows-based platforms in 1995. it is mostly written in C++

It is tightly integrated with other Microsoft technologies such as ASP.NET and the .NET framework, making it a preferred choice in enterprise environments that rely on Microsoft-based software. IIS offers a graphical user interface (GUI) for easy management, along with robust security features including authentication and authorization mechanisms.

IIS's smooth integration with Windows Server settings makes it a great option for businesses, which is one of the main reasons it is so well-liked and credited. Furthermore, IIS provides broad support for

ASP.NET applications that make use of dynamic webpages and web apps.

Advantages

- Seamless integration with other Microsoft technologies.

Disadvantages

- Less flexible than open-source alternatives.

4.4 LiteSpeed

High-performance web servers like LiteSpeed are renowned for their speed and security features. LiteSpeed Technologies created the LiteSpeed Web Server, which was originally made available in 2003 as an Apache substitute with superior performance. It is compatible with Apache configurations, including support for .htaccess files and mod_security, which makes it an easy replacement for Apache in many environments. One of its standout features is its built-in caching engine, which significantly improves the speed of content delivery, especially for dynamic websites like those running WordPress.

Advantages

- It possesses a built-in caching mechanism and supports HTTP/3.

Disadvantages

- Licensing costs for enterprise features can be high.

4.5 Apache Tomcat

Apache Tomcat is an open-source implementation of the Java Servlet, JavaServer Pages (JSP), and WebSocket technologies. It is designed specifically to serve Java-based web applications and is maintained by the Apache Software Foundation. Tomcat is lightweight and easy to deploy, making it ideal for educational purposes and enterprise Java environments. Unlike Apache HTTP Server or Nginx, Tomcat is not a full-featured HTTP web server but is often used in conjunction with them to serve Java applications.

5. Conclusion

Selecting the right web server is crucial for the performance and scalability of web applications. Each server has its unique features, advantages, and suitable use cases, emphasizing the importance of understanding specific requirements before making a decision.

6. References

- Ben-Ari, M. (2020). *Web Server Technologies: An Overview*. Journal of Technology Studies, 26(2), 15-28.
- Fielding, R. T., & Taylor, R. N. (2002). Principled design of the modern web architecture. *ACM Transactions on Internet Technology (TOIT)*, 2(2), 115-150.
- LiteSpeed Technologies. (2023). LiteSpeed Web Server Documentation. *Usage Statistics and Market Share of Web Servers*.
- McCool, M. (2019). *Comparative Analysis of Apache and Nginx Web Servers*. International Journal of Web Server Technologies, 12(1), 5-18.
- Microsoft Corporation. (2022). *Overview of Internet Information Services (IIS)*.
- Schroeder, B., Wierman, A., & Harchol-Balter, M. (2006). Closed versus open system models: a cautionary tale. In *Network System Design and Implementation*, 2006.