



TAREA 1.3 – SERVIDORES DE APLICACIONES

Despliegue de aplicaciones web



10 DE OCTUBRE DE 2023
PABLO VALLADARES GONZALEZ

Tarea 1.3 – Servidores de aplicaciones

Durante este módulo profesional nos vamos a centrar mucho sobre los servidores de aplicaciones web, pero estos no son los únicos que existen. Para trabajarlos y conocerlos, vamos a realizar la siguiente tarea.

1. ¿Qué es una base de datos?

Una base de datos se encarga no solo de almacenar datos, sino también de conectarlos entre sí en una unidad lógica, esto puede llegar a ser un programa o un cuaderno o un fichero con muchos folios/apartados

2. ¿Cuáles son las principales características de una base de datos relacional (una base de datos SQL)?

Una base de datos relacional es un conjunto de tablas que contienen datos que se ajustan a categorías predefinidas. Cada tabla contiene una o varias categorías en columnas. Cada fila contiene una instancia única de datos de las categorías definidas por las columnas.

3. ¿Cuál es la diferencia entre una base de datos relacional (o SQL) y una base de datos NoSQL?

Las BBDD SQL almacenan datos de manera estructurada y las NoSQL lo hacen en su formato original. Las SQL proporcionan una capacidad de escalar baja, en comparación con las NoSQL. Esta es una de las principales ventajas de las NoSQL, ya que están pensadas para grandes volúmenes de información como el Big Data.

4. Pon ejemplos de sistemas gestores de bases de datos SQL y NoSQL.

Los principales SGBD SQL son: MySQL, MariaDB, SQLite (biblioteca escrita en C que implementa un SGBD), PostgreSQL, Microsoft SQL Server y Oracle. Y Algunas de las más populares NoSQL son Riak, Redis, Amazon Dynamo DB, Voldemort, Membase, Dynamite y Tokio Cabinet, Cloudant y Cassandra.

5. ¿Qué es un servidor de correo SMTP?

Un servidor de correo es una aplicación de red de computadoras ubicada en un servidor de Internet, para prestar servicio de correo electrónico. De forma predeterminada, el protocolo estándar para la transferencia de correos entre servidores es el Protocolo Simple de Transferencia de Correo.

6. ¿En qué puerto se suele ejecutar SMTP?

Originalmente, el Simple Mail Transfer Protocol (SMTP) utilizaba el puerto 25. Hoy en día, el SMTP debería utilizar el puerto 587, que es el puerto para las transmisiones de correo electrónico encriptadas mediante SMTP Secure (SMTPS). El puerto 465 también se utiliza a veces para el SMTPS.

7. ¿Qué protocolo a nivel de transporte emplea SMTP?

Apertura de la conexión SMTP: dado que SMTP utiliza el protocolo de control de transmisión (TCP) como protocolo de transporte, este primer paso comienza con una conexión TCP entre el cliente y el servidor.

8. ¿Cómo funciona el envío de un email?

El correo electrónico funciona como un sistema de cliente-servidor. Ya sea desde un webmail o con un programa lo que hacemos es usar un cliente que nos muestra los correos que han llegado a nuestro buzón y nos deja enviar correos a otras personas

9. ¿Qué papel juegan los protocolos POP e IMAP? ¿En qué puerto se suelen ejecutar?

IMAP (Internet Message Access Protocol) o Protocolo de Acceso a Mensajes de Internet. POP3 es la tercera versión que se ha desarrollado del estándar POP (Post Office Protocol) o Protocolo de Oficina Postal.

Los puertos estándar de conexión utilizados por los protocolos POP3 e IMAP son el 110 y 143 respectivamente. Para conexiones seguras mediante SSL/TLS, los puertos son 995 para POP3 y 993 para IMAP.

10. Pon ejemplos de servidores de correo electrónico e indica las características principales de los mismos.

- Gmail. Una de las principales virtudes de Gmail es su diversidad de funciones en lo que a búsqueda, filtrado y clasificación de mensajes
- Yahoo Mail. Protección contra correo no deseado y virus. Soporte POP3 en algunos países (no en Estados Unidos). Sin embargo, el soporte de SMTP requiere actualizarse a una cuenta Plus.
- Outlook.

11. ¿Para qué se emplea el protocolo FTP?

El protocolo FTP = File Transfer Protocol se emplea para la transferencia de archivos y archivos pesados

12. ¿Qué puerto suele emplear FTP?

El servicio FTP es ofrecido por la capa de Aplicación del modelo TCP/IP y utiliza normalmente el puerto 20 y el 21 para conectarse.

13. ¿Qué protocolo a nivel de transporte emplea SMTP?

Apertura de la conexión SMTP: dado que SMTP utiliza el protocolo de control de transmisión (TCP) como protocolo de transporte, este primer paso comienza con una conexión TCP entre el cliente y el servidor.

14. ¿Cuáles son las dos versiones seguras del protocolo FTP?

Los dos protocolos más populares para transferencias seguras en FTP se llaman SFTP (FTP sobre SSH) y FTPS (FTP sobre SSL).

15. Pon ejemplos de servidores FTP.

FileZilla Server

ProFTPD

16. ¿Para qué sirve el protocolo DNS?

El DNS (Domain Name System, Sistema de Nombres de Dominio) es un conjunto de protocolos y servicios que permite a los usuarios utilizar nombres en vez de tener que recordar direcciones IP numéricas. Ésta es ciertamente la función más conocida de los protocolos DNS: la asignación de nombres a direcciones IP.

17. ¿Qué puerto suele emplear DNS?

El puerto numero 53

18. ¿Qué protocolo a nivel de transporte emplea DNS?

DNS usa TCP para la transferencia de zona y UDP para el nombre, y consulta regular (principal) o inversa. UDP se puede usar para intercambiar información pequeña, mientras que TCP debe usarse para intercambiar información de más de 512 bytes.

19. Pon ejemplos de servidores DNS.

- Google Public DNS
- Quad9
- DNS.Watch
- OpenDNS

20. ¿Para qué sirve el protocolo DHCP?

El Protocolo de configuración dinámica de host (DHCP) es un protocolo cliente-servidor que proporciona automáticamente un host de protocolo de Internet (IP) con su dirección IP y otra información de configuración relacionada, como la máscara de subred y la puerta de enlace de predeterminada.

21. ¿Qué puerto suele emplear DHCP?

Este servicio usa el protocolo de transporte UDP, puerto 68 en la parte del cliente y puerto 67 en el servidor.

22. ¿Qué protocolo a nivel de transporte emplea DHCP?

Este servicio usa el protocolo de transporte UDP, puerto 68 en la parte del cliente y puerto 67 en el servidor.

23. Pon ejemplos de servidores DHCP.

Microsoft Windows DHCP Server

Cisco IOS DHCP Server:

24. ¿Qué es un servidor proxy?

El anglicismo proxy o servidor proxy, adaptado al español como proxi, en una red informática, es un servidor —programa o dispositivo—, que hace de intermediario en las peticiones de recursos que realiza un cliente a otro servidor.

25. Define qué es un servidor web, y responde a las preguntas:

Definición:

Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente y generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente.

1. ¿qué características tienen?

- Necesitan un soporte físico. ...
- Requieren un sistema operativo. ...
- Cuentan con un Protocolo de Transferencia de Hipertexto. ...
- Sirven para la búsqueda de información. ...
- Alta capacidad.

2. ¿y funciones?

Un servidor web almacena los archivos que forman parte de un sitio electrónico o de una base de datos. El servidor funge como transmisor de esta información de acuerdo con las demandas de sus usuarios. De este modo, clientes y consumidores pueden acceder a la información solicitada vía remota.

3. ¿qué tipo de configuraciones se suelen incluir en los mismos?

Configuración de Virtual Hosts: Los servidores web pueden configurarse para alojar múltiples sitios web en una única máquina mediante la definición de virtual hosts. Cada virtual host puede tener su propio conjunto de configuraciones y archivos de contenido.

Configuración SSL/TLS: Para proporcionar conexiones seguras, los servidores web se pueden configurar para utilizar SSL/TLS, lo que permite el uso de HTTPS y cifra la comunicación entre el servidor y el cliente.

26. Hay dos principales servidores web empleados para los sistemas web: apache y nginx.

Busca por internet información y realiza una comparación entre ambos, con las ventajas e inconvenientes de cada uno de ellos.

Comparativa:

a nivel de arquitectura de software Apache necesitaría ser completamente reescrito para poder competir con servidores web como Nginx o LiteSpeed.

Apache necesita abrir múltiples hilos para atender múltiples peticiones de clientes, mientras que Nginx con un hilo puede atender múltiples peticiones de clientes. Como resultado, consume menos recursos y es capaz de responder más rápidamente.

Mientras que en alta carga Apache no es capaz de atender múltiples peticiones de forma concurrente y asíncrona, Nginx es todo lo contrario.

Mientras que Apache permite ejecutar PHP dentro del propio servidor web, en Nginx debemos externalizarlo, lo que es mucho más eficiente

Ventajas Nginx:

- Se trata de un software multiplataforma, por lo tanto, podremos instalarlo en la mayoría de nuestros servidores.
- Consume menos recursos que la mayoría de servicios que hacen su misma función.
- Nos proporciona un alto rendimiento soportando mayor carga y respondiendo mejor que sus competidores.

Desventajas Nginx:

- No es tan compatible con aplicaciones web que usen contenido dinámico
- Nginx es muy eficiente con el contenido estático, pero no lo es tanto con el contenido dinámico.

Ventajas Apache:

- De código abierto y gratuito, con una gran comunidad de usuarios.
- Parches de seguridad regulares y actualizados con frecuencia.
- Estructura basada en módulos.
- Multiplataforma.
- Personalización mediante.
- Compatible con los principales CMS y tiendas online y plataformas e-learning.

Desventajas Apache:

- La única desventaja notable es el pobre rendimiento frente a otras alternativas existentes en el mercado que son más veloces y con la capacidad de manejar mucho más tráfico con los mismos recursos.

27. Si, por ejemplo, empleamos Java, y usamos Apache Tomcat como servidor de aplicaciones,

¿necesitamos un servidor web? ¿En qué se diferencian ambos?

No necesitaremos un servidor web ya que Apache Tomcat, es un contenedor de servlet Java de código abierto que funciona como un servidor web

Tomcat se creó específicamente para aplicaciones Java, mientras que Apache es un servidor HTTP de propósito general. Puedes utilizar Apache junto con diferentes lenguajes de programación (PHP, Python, Perl, etc.) con la ayuda del módulo de Apache apropiado