

1. Datacenters.
2. Servidores. Concepto y uso.
3. Protocolos de comunicación.
4. Tipos de servidores y protocolos soportados (mails, bases de datos, web's, FTP).
5. Que trabajo realiza un servidor
6. Cómo funciona la lógica de un servidor
7. Protocolos Cliente Servidor
8. Sistemas operativos y conectividad.
9. Modelo Cliente Servidor
10. Tipos de servidores.
11. Desde el procesamiento: FTP – Correo – Web – IMAP - Streaming
12. Desde la estructura: Hosting -VPS - Servidor dedicado Cloud - Co-Location.
13. Comparativas: Amazon – Google Cloud – Microsoft Azure.

Arquitectura Cliente-Servidor

Dentro del contexto de desarrollo web, esta arquitectura hace referencia a un modelo de comunicación que vincula a varios dispositivos con un servidor a través de internet.

¿Cómo funciona un servidor?

La puesta a disposición de los servicios del servidor a través de una red informática se basa en el modelo cliente-servidor, concepto que hace posible distribuir las tareas entre los diferentes ordenadores y hacerlas accesibles para más de un usuario final de manera independiente. Cada servicio disponible a través de una red será ofrecido por un servidor (software) que está permanentemente en espera. Este es el único modo de asegurar que los clientes como el navegador web o los clientes de correo electrónico siempre tengan la posibilidad de acceder al servidor activamente y de usar el servicio en función de sus necesidades.

A QUÉ LLAMAMOS CLIENTE

Son los dispositivos que hacen peticiones de servicios o recursos a un servidor.

Pueden ser: una computadora, un teléfono celular, una tablet, una consola de video juegos o cualquier implemento que tenga la capacidad de conectarse a una red.

Dentro de Internet, el cliente suele acceder a estos servicios y recursos a través de un navegador web.

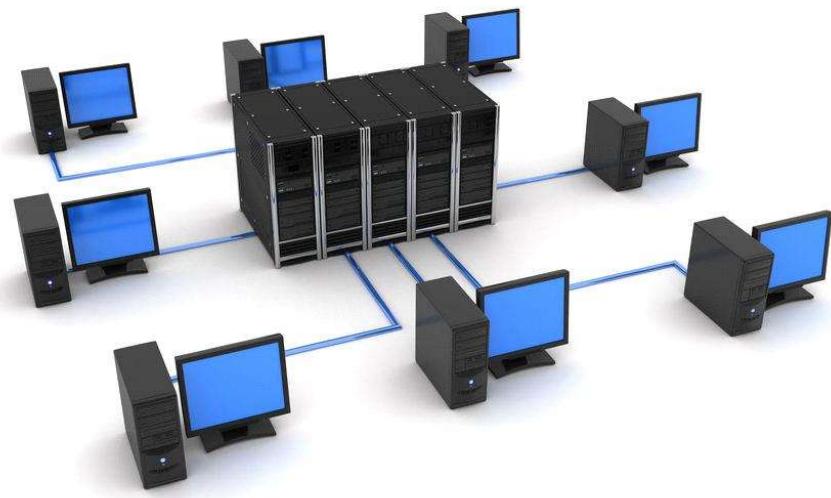


A QUÉ LLAMAMOS SERVIDOR

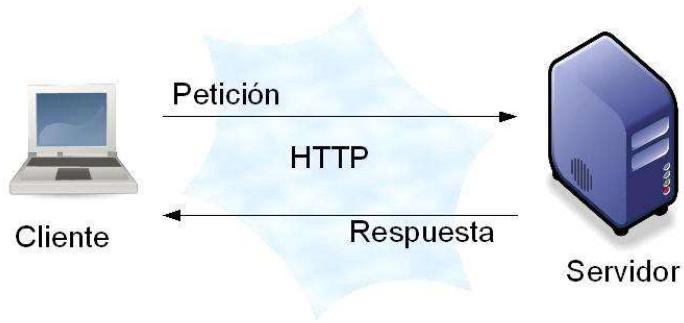
Es el equipo que brinda los servicios y recursos a los que acceden los clientes.

Es importante tener en cuenta que la misma computadora puede ser el cliente y el servidor al mismo tiempo.

De hecho, es lo más normal en el entorno de desarrollo de un sitio o aplicación web.



FLUJO CLIENTE SERVIDOR



REQUEST / SOLICITUD

Es la solicitud que hacemos a través del navegador (el cliente) a un servidor, en este ejemplo, la página de Facebook que está almacenada en sus servidores.

RESPONSE/ RESPUESTA

El servidor recibe nuestra solicitud, la procesa, y envía como resultado una respuesta al cliente (navegador), en este ejemplo devolverá la página principal del sitio.

¿Por qué es importante conocer este flujo Request-Response?

Porque dentro del mundo del desarrollo web, la mayoría de las aplicaciones tienen dos claros frentes: el frontend y el backend.

Front-End

Es todo lo que pasa del lado del cliente (en el navegador).

Aquí se incluyen todos los elementos gráficos que conforman la interfaz del sitio.

Los lenguajes que se manejan son HTML, para la estructura, CSS, para los estilos visuales y Javascript, para la interacción dentro del sitio.

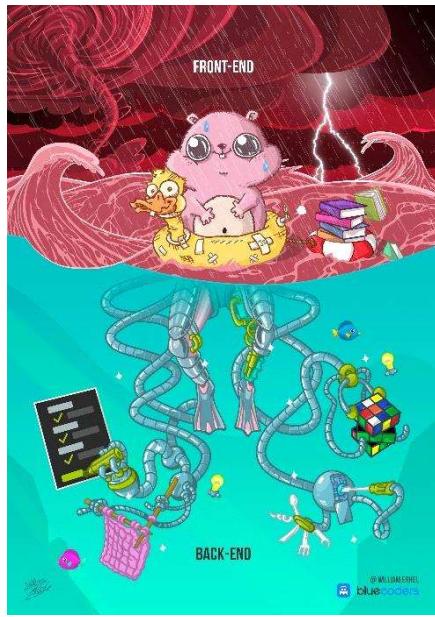


Back-End

Es todo lo que pasa del lado del servidor.

Aquí se incluye todo el funcionamiento interno y lógica del sitio. Es lo que permite que se carguen todas las peticiones solicitadas por el cliente.

Algunos de los lenguajes que se manejan son MySQL, para base de datos, PHP, para sitios webs dinámicos, entre otros.



Tipos de Servidores

Servidor web: la tarea principal de un servidor web es la de guardar y organizar páginas web y entregarlas a clientes como navegadores web o crawlers. La comunicación entre servidor (software) y cliente se basa en HTTP, es decir, en el protocolo de transferencia de hipertexto o en HTTPS, la variante codificada. Por regla general, se transmiten documentos HTML y los elementos integrados en ellos, tales como imágenes, hojas de estilo o scripts. Los servidores web más populares son el servidor HTTP Apache, los servicios de Internet Information Server de Microsoft (ISS) o el servidor Nginx.

Servidor de archivos: un servidor de archivos se encarga de almacenar los datos a los que acceden los diferentes clientes a través de una red. Las empresas apuestan por dicha gestión de archivos para que sea mayor el número de grupos de trabajo que tengan acceso a los mismos datos. Un servidor de archivos contrarresta los conflictos originados por las diferentes versiones de archivos locales y hace posible tanto la creación automática de las diferentes versiones de datos como la realización de una copia de seguridad central de la totalidad de datos de la empresa. En el acceso al servidor de archivos por medio de Internet entran en juego protocolos de transmisión como FTP (File Transfer Protocol), SFTP (Secure File Transfer Protocol), FTPS (FTP over SSL) o SCP (Secure Copy). Los protocolos SMB (Server Message Block) y NFS (Network File System) se encuentran habitualmente en las redes de área locales (LAN).

Servidor de correo electrónico: un servidor de correo electrónico consta de varios módulos de software cuya interacción hace posible la recepción, el envío y el reenvío de correos electrónicos, así como su puesta a punto para que estén disponibles. Por regla general funciona mediante el protocolo de transferencia simple de correo (SMTP). Los usuarios que quieran acceder a un servidor de correo electrónico necesitan un cliente de correo electrónico que recoja los mensajes del servidor y los entregue en la bandeja de

entrada, proceso que tiene lugar a través de los protocolos IMAP (Internet Message Access Protocol) o POP (Post Office Protocol).

Servidor de base de datos: un servidor de base de datos es un programa informático que posibilita que otros programas puedan acceder a uno o varios sistemas de bases de datos a través de una red. Las soluciones de software con una elevada cuota de mercado son Oracle, MySQL, Microsoft SQL Server, PostgreSQL y DB2. Los servidores de bases de datos ayudan a los servidores web, por regla general, a la hora de almacenar y entregar datos.

Servidor de juegos: los servidores de juegos son servidores (software) creados específicamente para juegos multijugador online. Estos servidores gestionan los datos del juego online y permiten la interacción sincrónica con el mundo virtual. La base de hardware de un servidor de juegos se encuentra en el centro de datos de los proveedores especializados o está disponible en una red doméstica local.

Servidor proxy: el servidor proxy sirve como interfaz de comunicación en las redes informáticas. En su papel de intermediario, el servidor proxy recibe las solicitudes de red y las transmite a través de su propia dirección IP. Los servidores proxy se usan para filtrar la comunicación, para controlar el ancho de banda, para aumentar la disponibilidad a través del reparto de cargas, así como para guardar datos temporalmente (caching). Además, los servidores proxy permiten una amplia anonimización, ya que la dirección IP del cliente queda oculta en el proxy.

Servidor DNS: el servidor DNS o servidor de nombres permite la resolución de nombres en una red. Los servidores DNS son de vital importancia para la red informática mundial (WWW), ya que traducen los nombres de host como www.example.com en la correspondiente dirección IP. Si quieres saber más sobre los servidores de nombres y sobre el sistema de nombres de dominio (DNS), visita nuestra guía digital.

En teoría, un único dispositivo físico puede alojar diferentes tipos de servidores. Sin embargo, es habitual alojar cada uno de los servidores en un ordenador independiente o que estos se repartan en más de un ordenador. De esta manera, se evita que la utilización del hardware de un servicio repercuta en el rendimiento de otros servicios.

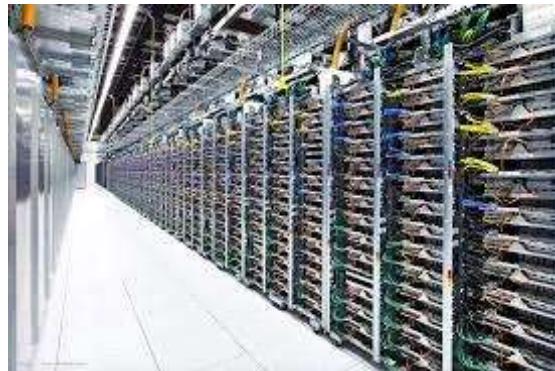
¿En qué consiste el alojamiento de servidores?

Mientras que a las grandes empresas les sale rentable la adquisición de hardware de servidores, los autónomos y los particulares que quieren desarrollar proyectos en un servidor propio recurren normalmente al alquiler. Los proveedores especializados ofrecen diferentes modelos de servidores de alquiler en los que los usuarios no tienen que preocuparse por el funcionamiento de la máquina física. La gama de productos abarca desde servidores dedicados cuyos componentes de hardware se ponen a disposición de los usuarios de manera exclusiva, hasta servicios de hosting compartido para alojar a varios clientes virtuales en una base de hardware común. Para obtener más información, visita nuestra guía sobre las ventajas y los inconvenientes de los diferentes modelos de alojamiento.

¿Que es un Datacenter?

El término Data Center es un término habitual para muchos. Sin embargo entrar en la definición del mismo puede ser interesante para otros muchos y, como una imagen vale más que mil palabras, los más interesados pueden ver en este video

<https://youtu.be/XZmGGAbHqa0>



algunas de las instalaciones del Data Center de la empresa Google y como es un el personal que trabaja en esos centros de información.

Un Data Center es, tal y como su nombre indica, un “centro de datos” o “Centro de Proceso de Datos” (CPD). Esta definición engloba las dependencias y los sistemas asociados gracias a los cuales:

- Los datos son almacenados, tratados y distribuidos al personal o procesos autorizados para consultarlos y/o modificarlos.
- Los servidores en los que se albergan estos datos se mantienen en un entorno de funcionamiento óptimo.

Los primeros Data Centers se diseñaron siguiendo las arquitecturas clásicas de informática de red, en las que los equipos eran “apilables” en mesas, armarios o racks como la siguiente imagen:



La necesidad de fácil gestión y de optimización del espacio han hecho que se evolucione hacia sistemas basados en equipos cuyas dimensiones permiten aprovechar al máximo el volumen disponible en los racks (equipos “enrancables”), logrando una alta densidad de equipos por unidad de espacio.

Un Datacenter físico puede alojar Datacenters virtuales, cuyo costo es menor gracias a la virtualización. Cada Centro de Datos virtual es independiente del resto y dispone de máximas garantías de seguridad, disponibilidad y flexibilidad.

Los Data Center iniciales tampoco estaban diseñados para proporcionar facilidades de red avanzadas, ni los requerimientos mínimos de ancho de banda y velocidad de las arquitecturas actuales. La rápida evolución de Internet y la necesidad de estar conectados en todo momento han obligado a las empresas a requerir un alto nivel de fiabilidad y seguridad, de tal forma que se proteja la información corporativa y esté disponible sin interrupciones o degradación del acceso, con el objetivo de no poner en peligro sus negocios, sean del tamaño que sean. El cumplimiento de estos requisitos, cada día más demandados, es posible dentro de un Data Center. Igual que un banco es el mejor sitio para guardar y gestionar el dinero, un centro de datos lo es para albergar los equipos y sistemas de información.

Los datos almacenados, no son datos estáticos, están en constante movimiento, se interrelacionan unos con otros y dan como resultado nuevos datos. Su crecimiento es constante y ello implica no solo que deben estar protegidos mediante las medidas de seguridad adecuadas, sino también dotados de estupendos “motores que les permitan moverse ágilmente por las autopistas de la información”.

El crecimiento exponencial del número de usuarios de los servicios online ha llevado a las empresas a subcontratar la gestión, mantenimiento y administración de sus equipos informáticos y de comunicaciones en los Data Center. Esto les permite centrarse en el desarrollo de su propio negocio y olvidarse de complejidades tecnológicas derivadas de las características anteriormente comentadas, así como prestar el servicio sin la necesidad de realizar una inversión elevada en equipamiento dedicado a este fin.

Que busca una empresa contratando infraestructura externa?

El modelo de Datacenter se ha replicado y mejorado con el tiempo para que las empresas clientes puedan solicitar:

- **Ancho de banda:** Es necesario contar con una gran capacidad de transferencia de datos, de tal forma que no sea apreciable ningún tipo de “retardo” provocado por la utilización de la red.
- **Fiabilidad y Alta disponibilidad:** Los sistemas deben responder a cualquier situación crítica, haciendo posible la prestación del servicio sin pérdida apreciable de calidad incluso cuando es necesario atender gran cantidad de peticiones de forma puntual (“pico”) o continuada. Resulta imprescindible contar con sistemas de alta disponibilidad y redundancia a través de modernas arquitecturas de red y servicios.
- **Seguridad:** Cubriendo fundamentalmente tres aspectos:

1 – Seguridad física: Comprendiendo la seguridad de los sistemas hardware, soportes, dependencias y demás entidades «tangibles» del entorno del Data Center.

2 – Seguridad lógica: Incluyendo los aspectos de protección aportados por aplicaciones, protocolos y procesos que intervienen en el sistema, y complementado por elementos de seguridad de red (cortafuegos), detección de intrusos (IDS), y análisis a nivel de aplicación (antivirus).

3 – Seguridad político-corporativa: Formada por los aspectos de seguridad relativos a política general de la organización, normas, procedimientos y convenciones internas aplicables. En este aspecto se debe tener en cuenta el cumplimiento de la legislación aplicable.

Dichas áreas están interrelacionadas, y la existencia coherente de medidas de seguridad en cada una ellas garantiza el nivel de protección óptimo frente a las posibles amenazas de seguridad.

- **Escalabilidad, flexibilidad y rapidez a la hora de implementar sus proyectos (time-to-market):** Los equipos y los web sites del Data Center tienen que funcionar con la misma agilidad y rapidez que Internet para ser competitivos. La infraestructura existente tiene que permitir poner en marcha un proyecto en el mínimo tiempo posible, así como ampliar el número de elementos o la capacidad de los existentes de forma rápida y sin impacto en el servicio. Cada proyecto requerirá una solución específica cuyos requisitos variarán en el tiempo, si la infraestructura y el Data Center no permiten efectuar estos cambios con la celeridad necesaria nunca se logrará estar a la altura de la demanda del usuario o el servicio prestado por los posibles competidores.

- **Arquitecturas sofisticadas de comunicaciones:** Para cubrir la demanda de los usuarios es necesario contar con los últimos desarrollos que la industria Internet lanza al mercado (firewalls, balanceadores de carga, sistemas de replicación de contenidos, etc).
- **La gestión y administración por personal especializado:** Los sistemas implicados en la prestación del servicio requieren un alto nivel de especialización. Así mismo, la monitorización continua 24×7 y detección precoz de errores se han convertido en requisitos imprescindibles para garantizar la calidad del servicio prestado, y lograr detectar los “cuellos de botella” antes de que representen un problema.

Tipos de Servidores Web

VPS + Cloud + Hosting Compartido + Dedicado + Hosting Wordpress

VPS

Un servidor virtual privado (VPS, por sus siglas en inglés) es una partición virtual dentro de un servidor físico que le asigna recursos exclusivos a cada partición. Éstas otorgan acceso raíz que permite instalar un sistema operativo y trabajar con alto grado de libertad.

El hosting VPS es uno de los servicios de alojamiento más populares que puedes elegir para tu sitio web. Utiliza tecnología de virtualización para proporcionarte recursos dedicados (privados) en un servidor con múltiples usuarios.

Es una solución más segura y estable que el hosting compartido donde no se obtiene espacio de servidor dedicado. Sin embargo, es de menor escala y más barato que alquilar un servidor completo.

El hosting VPS generalmente es elegido por los propietarios de sitios web que tienen un tráfico de nivel medio que excede los límites de los planes de hosting compartido pero que aún no necesitan los recursos de un servidor dedicado.

Las soluciones de VPS generalmente ofrecen varios planes de alojamiento. Por ejemplo, una empresa de alojamiento, tiene seis planes de hosting VPS que se adaptan a las diferentes necesidades y te permiten escalar tu sitio sin problemas cuando necesites más recursos.

- ¿Cuándo usar VPS y cómo decidir si es hora de una mejora?*
- ¿Hay alguna alternativa al VPS? ¿Qué es un VPS administrado?*
- En resumen, ¿qué es un VPS?*
- ¿Cómo funciona el hosting VPS?*

Un servidor es una computadora en la que tu proveedor de alojamiento web almacena los archivos y las bases de datos necesarios para tu sitio web. Cada vez que un visitante en línea quiere acceder a tu sitio web, su navegador le envía una solicitud a tu servidor y transfiere los archivos necesarios a través de Internet. El hosting VPS te proporciona un servidor virtual que simula un servidor físico; sin embargo, en realidad, la máquina se comparte entre varios usuarios.

Al usar la tecnología de virtualización, tu proveedor de alojamiento web instala una capa virtual sobre el sistema operativo del servidor. Esta capa divide el servidor en particiones y le permite a cada usuario instalar su propio sistema operativo y software.

Por lo tanto, un servidor privado virtual (VPS) es tanto virtual como privado porque tienes control absoluto. Está separado de otros usuarios del servidor a nivel del sistema operativo. De hecho, la tecnología VPS es similar a la creación de particiones en tu computadora

cuando quieras ejecutar más de un sistema operativo (por ejemplo, Windows y Linux) sin tener que reiniciar.

Un VPS te permite configurar tu sitio web dentro de un contenedor seguro con recursos garantizados (memoria, espacio en disco, núcleos de CPU, etc.) que no tienes que compartir con otros usuarios. Con el hosting VPS, tienes el mismo acceso de nivel raíz que si alquilaras un servidor dedicado, pero a un costo mucho más bajo.

Comparación de VPS con otros tipos de alojamiento web

Los diferentes tipos de alojamiento web te permiten realizar diferentes niveles de personalización en tu servidor. Varían en precio, rendimiento (por ejemplo, tiempo de carga de la página) y la disponibilidad del servicio (por ejemplo, tiempo de actividad). A continuación, cómo el hosting VPS se compara con otras soluciones de hosting.

Hosting Compartido

El hosting compartido es la solución para los propietarios de sitios web con menos tráfico. Es el punto de partida de la mayoría de las pequeñas empresas y bloggers. Con el hosting compartido, se divide el mismo servidor físico entre varios clientes de la empresa de hosting. No obtienes recursos dedicados para ti, ya que tu sitio se ejecuta en el mismo sistema operativo que el de los demás.

Por lo tanto, la memoria y el poder de computación que tu sitio puede usar se ven afectados por las necesidades de los otros usuarios del servicio. Por ejemplo, si hay un pico de tráfico repentino en un sitio web alojado en el mismo servidor, puede aumentar el tiempo de carga de tu página. Tampoco puedes elegir tu sistema operativo ni otro software de servidor, ya que todos los usuarios usan la misma configuración. En definitiva, es tu proveedor de hosting el que se encarga de cada aspecto de tu entorno de hosting compartido.

Puedes pensar en hosting compartido como un alquiler en el que compartes el mismo apartamento con varios compañeros. El hosting VPS sigue siendo una solución compartida, sin embargo, cada quien tiene su propio cuarto donde pueden personalizar el espacio de acuerdo a sus necesidades. Por ejemplo, pueden elegir la pintura, los muebles, la decoración, etc.

Cloud Hosting

Con el cloud hosting o hosting en la nube no usas un solo servidor, sino un clúster (varias computadoras interconectadas) que se ejecuta en la nube. Cada servidor en el clúster almacena una copia actualizada de tu sitio web. Cuando uno de los servidores está demasiado ocupado, el clúster redirige automáticamente el tráfico a uno que no lo esté tanto. Como resultado, el hosting en la nube no tiene tiempo de inactividad (downtime), ya que siempre hay un servidor en el clúster que puede atender las solicitudes de los visitantes de tu sitio web.

El alojamiento en la nube y el alojamiento VPS no son mutuamente excluyentes. En cambio, muchas empresas de hosting ofrecen alojamiento VPS en una infraestructura en la nube. Esta es también una solución que hemos elegido para comentar, ya que descubrimos

que la combinación de VPS con las tecnologías en la nube conduce al rendimiento y la confiabilidad más altos posibles en el entorno del servidor virtual.

Hosting WordPress

El hosting WordPress es un servicio específicamente ofrecido para los propietarios de sitios de WordPress. Viene con varias características relacionadas con WordPress que solo puedes usar si tienes un sitio de WordPress, como instalación con un solo clic, plugins preinstalados o una interfaz de línea de comandos de WP. Los servidores están configurados para las necesidades de WordPress. Por lo tanto, los proveedores de hosting ofrecen hosting para WordPress como parte de su servicio de hosting compartido.

Aunque también es posible configurar un sitio de WordPress en un servidor privado virtual, no puedes obtener acceso a los servidores personalizados que se han configurado con WordPress en mente. Sin embargo, si aún eliges VPS para tu sitio de WordPress, puedes configurar tu entorno de alojamiento según las necesidades de tu negocio.

Hosting dedicado

Con hosting dedicado, alquilas un servidor físico completo para tu negocio. Si tienes un sitio web de alto tráfico, el hosting dedicado puede ser la mejor solución para ti, ya que los servidores dedicados son rápidos, flexibles y totalmente personalizables. Sin embargo, el servicio también viene con un precio, por eso no es la mejor solución para todos, especialmente si tienes un sitio web pequeño o mediano.

Si bien el hosting VPS te permite elegir y configurar tu sistema operativo y aplicaciones del servidor, el hosting dedicado va un paso más allá. No solo te permite configurar el software sino también el hardware, ya que todo el servidor es tuyo y nadie tiene voz en cuanto a la configuración. También puedes ejecutar un servidor dedicado en el sitio (por ejemplo, en tu oficina), sin embargo, no obtendrás el apoyo de un equipo de hosting profesional en ese caso.

Pros y contras del VPS

El hosting VPS puede ser una solución ideal para ti si es el servicio que realmente necesitas. A continuación, puedes leer los pros y contras de tener un servidor privado virtual.

Pros

- Es más rápido y más confiable que un servidor de hosting compartido.
- Dado que los recursos del servidor, como la memoria o la potencia de procesamiento, están garantizados, existe una fluctuación de cero a mínima en los recursos disponibles.
- Los problemas y aumentos de tráfico de otros usuarios del servidor no afectan a tu sitio.
- Obtienes acceso de superusuario (raíz) a tu servidor.
- Obtienes mayor privacidad, ya que tus archivos y bases de datos están bloqueados para otros usuarios del servidor.
- Es un servicio fácil de escalar. A medida que tu sitio web crece, puedes mejorar fácilmente los recursos de tu servidor (RAM, CPU, espacio en disco, ancho de banda, etc.).

Contras

- Es más caro que el hosting compartido.
- Se requiere más conocimiento técnico para administrar tu servidor.
- Los servidores mal configurados pueden generar vulnerabilidades de seguridad.

¿Cuándo usar VPS y cómo decidir si es hora de una mejora?

El hosting VPS generalmente se ve como el paso siguiente después de que tu sitio web crezca fuera de los límites de recursos del hosting compartido. Si el plan de hosting web compartido más avanzado ya no es suficiente para ejecutar tu sitio web sin problemas, vale la pena pasarse a un plan de VPS. En tales casos, el hosting VPS puede proporcionarte lo mejor de dos mundos: hosting compartido y dedicado.

Sin embargo, hay otros casos de uso en los que puede ser una buena idea comenzar con un plan de VPS desde el principio. Por ejemplo, los sitios web de comercio electrónico donde se necesita garantizar pagos seguros con un entorno rápido y estable pueden beneficiarse mucho de un servidor privado virtual. De hecho, si almacenas cualquier tipo de información confidencial o tienes que procesar pagos en línea, un VPS puede ayudarte a reducir el riesgo de violaciones de seguridad y robo de identidad.

Si esperas recibir picos de tráfico en determinados momentos en tu sitio, por ejemplo, en el caso de un sitio web de planificación de eventos o de venta de boletos, un plan de hosting VPS puede ser un regalo del cielo. Los servidores de juegos y otros sitios con muchos recursos también pueden ejecutarse con un rendimiento mucho mejor en un servidor privado virtual.

¿Hay alguna alternativa al VPS? ¿Qué es un VPS administrado?

Uno de los mayores escollos del hosting VPS es tener que gestionar todo el entorno del servidor por tu cuenta. Si tu servidor virtual no está bien configurado y mantenido, podrías enfrentarte con serios riesgos de seguridad y de pérdida de rendimiento. Los servicios administrados de hosting VPS han aparecido en el mercado como una solución viable a este problema. Puedes considerarlo como una alternativa mejorada al hosting VPS, que viene con soporte técnico completo.

El VPS administrado sigue siendo una novedad en el mercado, sin embargo cada vez más proveedores de hosting lo ofrecen a sus clientes. Por ejemplo, proporcionar un entorno de hosting totalmente administrado con planes de cloud hosting.

En resumen, ¿qué es un VPS?

El hosting VPS te brinda acceso a recursos garantizados y un control total sin tener que manejar tu propio servidor. Es una excelente solución para sitios web de mediano a alto tráfico, de recursos pesados y de comercio electrónico. Sin embargo, las empresas que esperan un crecimiento rápido también pueden beneficiarse de la estabilidad del servicio. Si quieres un entorno de hospedaje confiable y de escala empresarial a un precio amigable, definitivamente vale la pena considerar el hosting VPS para tu sitio web.

Funcionamiento de un servidor Web

Un servidor es un dispositivo virtual que le brinda espacio y estructura a los sitios web para que almacenen sus datos y manejen sus páginas.

¿Alguna vez te has preguntado dónde están localizados todos los contenidos de tu estrategia de marketing digital que subes a tu página web?

Es, pues, en los servidores web. Estos son fundamentales para Internet y si ellos no existieran, navegar por la web sería muy diferente a lo que conocemos hoy en día.

Con la transformación digital y el creciente uso de Internet era más que necesario crear servidores que fueran capaces de almacenar y emitir la biblioteca de información (casi infinita) que encontramos en la web.

¿Qué es un servidor?

Un servidor web (server) es un ordenador de gran potencia que se encarga de “prestar el servicio” de transmitir la información pedida por sus clientes (otros ordenadores, dispositivos móviles, impresoras, personas, etc.)

Los servidores web (web server) son un componente de los servidores que tienen como principal función almacenar, en web hosting, todos los archivos propios de una página web (imágenes, textos, videos, etc.) y transmitirlos a los usuarios a través de los navegadores mediante el protocolo HTTP (Hypertext Transfer Protocol).

¿Para qué sirve un servidor web en Internet?

El rol principal de un servidor web es almacenar y transmitir el contenido solicitado de un sitio web al navegador del usuario.

Este proceso, para los internautas no dura más que un segundo, sin embargo, a nivel del web server es una secuencia más complicada de lo que parece.

Para cumplir con sus funciones el servidor deberá tener la capacidad de estar siempre encendido para evitar interrumpir el servicio que le ofrece a sus clientes. Si dicho servidor falla o se apaga, los internautas tendrán problemas al ingresar al sitio web.

¿Cómo funciona un servidor web?

La comunicación entre un servidor y sus clientes se basa en HTTP, es decir, en el protocolo de transferencia de hipertexto o en su variante codificada HTTPS.

Para saber cómo funciona, primero es necesario conocer que el web server está permanentemente en espera de una solicitud de información.

Además, ten en cuenta que toda computadora, smartphone o tablet tiene una dirección IP única e irrepetible que lo identifica de otro dispositivo en la red, así es como el servidor web envía la información exacta que el internauta está esperando.

Ahora bien, para que el web server pueda cumplir con su función es necesario que reciba la petición por parte de un navegador, en otras palabras, se envía un pedido desde una dirección IP hacia la dirección IP del servidor que aloja los archivos del sitio en cuestión.

A continuación, el servidor web busca en sus archivos la información que se le está solicitando, procede a interpretar las líneas de código y a enviar el resultado al navegador cuya dirección IP fue la solicitante.

Este resultado se le muestra a los internautas y es lo que siempre sucede cuando se navega en sitios de Internet. Cuando este proceso se completa podemos decir que el web server ha cumplido con su función.

¿Cuáles son las características de un servidor web?

Como características necesarias de un servidor web a nivel de software y hardware, podemos encontrar:

A nivel de software

Sistema Operativo

Se encarga de que el hardware funcione y logre interactuar con los servicios que corre el sistema. Algunos ejemplos son:

Unix,
Linux,
o Windows.

Sistemas de archivos

Es una guía lógica que permite que el sistema pueda ubicar, ordenar y filtrar datos en el disco duro, con el fin de que podamos leerlos, modificarlos o eliminarlos.

Software servidor HTTP

Son los diferentes tipos de servidores web especializados en transmitir el contenido vía web (Apache, Nginx, IIS, Caddy, etc.).

Virtual Hosting

Permite que bajo el mismo web server e IP se alojen en varios sitios web distintos.

Despacho de ficheros estáticos y dinámicos

Los ficheros estáticos brindan soporte para alojar y despachar archivos como:

JPG,
GIF,
PNG,
BMP,
CSS,
TXT,
HTML,
Javascript,
MP3

y MP4.

Los ficheros dinámicos funcionan para información y se basan en archivos que serán procesados por el servidor dependiendo generalmente de un lenguaje de desarrollo, por ejemplo: en PHP, ASP, Python, Ruby y GO.

Monitoreo de Red y Límites

Permite monitorear el tránsito de red, paquetes que entran y salen, así como servicios de sistema y uso de hardware como:

- el uso del Almacenamiento;
- consumo de RAM;
- porcentaje de ocupación del CPU;
- velocidad de la red;
- rendimiento de escritura/lectura en disco.

Sistema de seguridad

El sistema de seguridad de un servidor debe:

- imponer límites de acceso por dirección IP;
- denegar o permitirle acceso a ciertos archivos o URLs;
- solicitar usuario y contraseña para autenticación básica HTTP;
- realizar un filtrado de peticiones inseguras;
- dar soporte para despachar información cifrada con certificados de seguridad SSL vía HTTPS.

A nivel de hardware

Rack y gabinete

El rack se refiere al lugar donde se alojan los servidores físicamente y el gabinete es el armazón que sostiene los componentes de hardware de una computadora.

CPU

Es el centro de procesamiento de datos del servidor desde donde se realizan todos los cálculos lógicos y matemáticos para que el usuario pueda manipular y acceder a los datos como necesita.

Memoria RAM

Se utiliza para almacenar información y datos de forma temporal dependiendo de la demanda del usuario a través del sistema operativo.

Unidades de almacenamiento

El almacenamiento de servidores web se hace en discos duros, los cuales permiten almacenar la información del sistema operativo, los servicios de sistema, y en última instancia los datos cargados por el usuario.

Puerto de red

El ancho de banda es el que te permite tener un volumen suficiente para transmitir información de ida y vuelta hacia y desde tu servidor web.

¿Qué tipos de servidores web existen?

Existen muchos tipos de servidores web, conoce cuáles son los servidores web más usados en la actualidad:

Apache

Es el más común y utilizado en el mundo, sin embargo, ha perdido popularidad frente a Microsoft IIS y Nginx.

Entre las ventajas de Apache está que es un código abierto, con software gratuito y multiplataforma, y entre sus desventajas su bajo rendimiento cuando recibe miles de requests (peticiones) simultáneas en procesamiento de contenido dinámico o archivos estáticos.

Nginx

Conocido y popularizado como una de las mejores alternativas de Apache. Nginx es un servidor web de código abierto y gratuito (aunque también existe una versión comercial) que se destaca por su alto rendimiento.

Entre sus beneficios resalta una configuración simple, ligera, rápida y excelente en cuanto a seguridad y rendimiento, además permite ser configurado para integrarse nativamente con casi cualquier tecnología y lenguaje de programación moderno.

Como desventajas podemos encontrar que no soporta los archivos .htaccess (de Apache), aunque incluye su propio lenguaje de rewrites.

LiteSpeed

Es un software de despacho HTTP desarrollado por LiteSpeedTech, existe una versión de código abierto (open source) y una versión comercial que incluye diferentes tipos de licencia.

Algunos de los beneficios de LiteSpeed es que soporta grandes cantidades de conexiones simultáneas con un consumo de recursos realmente bajo (incluso con aplicaciones demandantes como las que utilizan PHP) y a nivel de archivos estáticos está a la altura de Nginx.

Microsoft IIS

Internet Information Services o IIS se ha popularizado para ofrecer servicios en la nube, principalmente en Azure (la plataforma de Cloud Hosting de Microsoft).

Además, su perfecta integración con Windows (claro está), Visual Studio y sus herramientas hicieron que este web server se posicionara como el servidor líder en el mundo empresarial.

Otros servidores web conocidos

Lighttpd;

Caddy;

Cherokee;

NodeJS;

Sun Java System Web Server;

Google Web Server (GWS), es el servidor privado de Google por lo que no puede ser descargado.

Cada servidor web es indicado para ciertas funciones, por lo que escoger cuál vas a usar va a depender de lo que pretendas hacer con ellos.

En resumen:

No olvides que al considerar elegir un web server debes tener en cuenta:

- que tan bien funciona con el sistema operativo y otros servidores;
- su capacidad para manejar la programación del servidor;
- las características de seguridad;
- las herramientas particulares de publicación;
- motor de búsqueda;
- la creación de sitios que vienen con él.

Los servidores Web no solo permiten publicar tu sitio en Internet, sino que también entregan el contenido a tus lectores. Conocer cuál se adapta más a ti y a tus lectores es fundamental para que el web server que elijas cumpla adecuadamente sus funciones.

Arquitectura de servidores: Cliente-Servidor y Multicapa

Ya sabemos lo que es un servidor. Se trata de una computadora, con alta capacidad que se encarga de compartir toda clase de información con un número casi infinito de otras computadoras en todo el mundo, siempre y cuando estén conectados a él.

Esta conexión usualmente es inalámbrica pero puede ser alámbrica también. Sin embargo, muchos aún desconocen cómo funcionan estas arquitecturas. Hablaremos ahora sobre la arquitectura de servidores: Cliente-Servidor y Multicapa.

Qué es la arquitectura de servidores

La arquitectura de servidores, se refiere a la arquitectura de una red informática en la que muchos procesadores remotos, conocidos como “clientes” solicitan y reciben servicio de un sistema centralizado, es decir, de una computadora host a la que se le llama servidor. De allí el término “modelo cliente servidor” del cual hablaremos a continuación.

Modelo Cliente Servidor

Cliente-servidor es una relación en la cual un programa (el cliente) solicita un servicio o recurso de otro programa (el servidor).

A principios del siglo pasado, la etiqueta cliente-servidor se usó para distinguir la computación distribuida por computadoras personales (PC) del modelo de computación monolítico y centralizado utilizado por los mainframes. Hoy en día, las transacciones informáticas en las que el servidor cumple una solicitud realizada por un cliente son muy comunes y el modelo cliente-servidor se ha convertido en una de las ideas centrales de la informática en red.

En este contexto, el cliente establece una conexión con el servidor a través de una red de área local (LAN) o una red de área amplia (WAN), como Internet. Una vez que el servidor ha cumplido la solicitud del cliente, la conexión finaliza. Debido a que varios programas de clientes comparten los servicios del mismo programa de servidor, un servidor especial llamado “daemon” puede activarse solo para esperar las solicitudes de los clientes.

En los primeros días de Internet, la mayoría del tráfico de red era entre clientes remotos que solicitaban contenido web y los servidores del centro de datos que proporcionaban el contenido. Este patrón de tráfico se conoce como tráfico norte-sur. Hoy, con la madurez de la virtualización y la computación en la nube, es más probable que el tráfico de red sea de servidor a servidor, un patrón conocido como tráfico este-oeste.

Esto, a su vez, ha cambiado el enfoque del administrador de un modelo de seguridad centralizado diseñado para proteger el perímetro de la red a un modelo de seguridad descentralizado que se centra más en controlar el acceso de los usuarios individuales a los servicios y datos, y auditar su comportamiento para garantizar el cumplimiento de las políticas y regulaciones.

Ventajas y desventajas del modelo cliente-servidor

Una ventaja importante del modelo cliente-servidor es que su arquitectura centralizada ayuda a facilitar la protección de los datos con controles de acceso impuestos por las políticas de seguridad. Además, no importa si los clientes y el servidor se basan en el mismo sistema operativo porque los datos se transfieren a través de protocolos cliente-servidor que son independientes de la plataforma.

Una desventaja importante del modelo cliente-servidor es que si demasiados clientes solicitan simultáneamente datos del servidor, pueden sobrecargarse. Además de causar congestión en la red, demasiadas solicitudes pueden resultar en una denegación de servicio.

Protocolos cliente-servidor

Los clientes generalmente se comunican con los servidores mediante el conjunto de protocolos TCP / IP. TCP es un protocolo orientado a la conexión, lo que significa que se establece y mantiene una conexión hasta que los programas de aplicación en cada extremo hayan terminado de intercambiar mensajes.

Determina cómo dividir los datos de la aplicación en paquetes que las redes pueden entregar, envía paquetes y acepta paquetes de la capa de red, gestiona el control de flujo y maneja la retransmisión de paquetes descartados o confusos, así como el reconocimiento de todos los paquetes que llegan. En el modelo de comunicación de interconexión de sistemas abiertos (OSI), TCP cubre partes de la capa 4, la capa de transporte y partes de la capa 5, la capa de sesión.

Por el contrario, IP es un protocolo sin conexión, lo que significa que no hay una conexión continua entre los puntos finales que se están comunicando.

Cada paquete que viaja a través de Internet se trata como una unidad de datos independiente sin ninguna relación con ninguna otra unidad de datos. (La razón por la que los paquetes se colocan en el orden correcto es debido a TCP). En el modelo de comunicación de Interconexión de Sistemas Abiertos (OSI), IP está en la capa 3, la Capa de Redes.

Modelo Multicapa

Cuando hablamos del modelo multicapa, hablamos del modelo OSI que significa interconexión de sistemas abiertos. Ha sido desarrollado por ISO (Organización Internacional de Normalización), en el año 1974. Es una arquitectura de 7 capas, cada una tiene una funcionalidad específica para realizar. Todas estas 7 capas trabajan en colaboración para transmitir los datos de una persona a otra en todo el mundo. Te las explicamos a continuación:

Capa 1 o Capa Física

La capa más baja del modelo de referencia OSI es la capa física. Es responsable de la conexión física real entre los dispositivos. La capa física contiene información en forma de

bits. Es responsable de la conexión física real entre los dispositivos. Al recibir datos, esta capa obtendrá la señal recibida y la convertirá en 0 y 1 y los enviará a la capa de enlace de datos, que volverá a unir el marco.

Las funciones de la capa física son:

Sincronización de bits: la capa física proporciona la sincronización de los bits al proporcionar un reloj. Este reloj controla tanto el emisor como el receptor, proporcionando así la sincronización a nivel de bit.

Control de velocidad de bits: la capa física también define la velocidad de transmisión, es decir, el número de bits enviados por segundo.

Topologías físicas: la capa física especifica la forma en que los diferentes dispositivos / nodos se organizan en una red, es decir, topología de bus, estrella o malla.

Modo de transmisión: la capa física también define la forma en que los datos fluyen entre los dos dispositivos conectados. Los diversos modos de transmisión posibles son: Simplex, half-duplex y full-duplex.

Capa 2 o Capa de enlace de datos

La capa de enlace de datos es responsable de la entrega nodo a nodo del mensaje. La función principal de esta capa es asegurarse de que la transferencia de datos esté libre de errores de un nodo a otro, sobre la capa física. Cuando un paquete llega a una red, es responsabilidad de DLL transmitirlo al Host utilizando tu dirección MAC.

La capa de enlace de datos se divide en dos subcapas:

Control de enlace lógico (LLC)

Control de acceso a medios (MAC)

El paquete recibido de la capa de red se divide en tramas dependiendo del tamaño de trama de la NIC (tarjeta de interfaz de red). DLL también encapsula la dirección MAC del remitente y del receptor en el encabezado. La dirección MAC del receptor se obtiene colocando una solicitud ARP (Protocolo de resolución de direcciones) en el cable preguntando "¿Quién tiene esa dirección IP?" Y el host de destino responderá con su dirección MAC.

Las funciones de la capa de enlace de datos son:

Enmarcado: el enmarcado es una función de la capa de enlace de datos. Proporciona una manera para que un remitente transmita un conjunto de bits que son significativos para el receptor. Esto se puede lograr adjuntando patrones de bits especiales al principio y al final del cuadro.

Direccionamiento físico: después de crear tramas, la capa de enlace de datos agrega direcciones físicas (dirección MAC) del remitente y / o receptor en el encabezado de cada trama.

Control de errores: la capa de enlace de datos proporciona el mecanismo de control de errores en el que detecta y retransmite tramas dañadas o perdidas.

Control de flujo: la velocidad de datos debe ser constante en ambos lados; de lo contrario, los datos pueden corromperse, por lo tanto, el control de flujo coordina esa cantidad de datos que se pueden enviar antes de recibir el acuse de recibo.

Control de acceso: cuando varios dispositivos comparten un único canal de comunicación, la subcapa MAC de la capa de enlace de datos ayuda a determinar qué dispositivo tiene control sobre el canal en un momento dado.

Capa 3 o Capa de Red

La capa de red funciona para la transmisión de datos de un host a otro ubicado en diferentes redes. También se encarga del enrutamiento de paquetes, es decir, la selección de la ruta más corta para transmitir el paquete, a partir del número de rutas disponibles. La dirección IP del remitente y el receptor se colocan en el encabezado por capa de red.

Las funciones de la capa de red son:

Enrutamiento: los protocolos de la capa de red determinan qué ruta es adecuada desde el origen hasta el destino. Esta función de la capa de red se conoce como enrutamiento.

Direccionamiento lógico: para identificar cada dispositivo en la red interna de manera única, la capa de red define un esquema de direccionamiento. La dirección IP del remitente y el receptor se colocan en el encabezado por capa de red. Tal dirección distingue cada dispositivo de manera única y universal.

Capa 4 o Capa Transporte

La capa de transporte proporciona servicios a la capa de aplicación y toma ayuda de la capa de red. Los datos en la capa de transporte se denominan segmentos. Es responsable de la entrega de extremo a extremo del mensaje completo. La capa de transporte también proporciona el reconocimiento de la transmisión exitosa de datos y retransmite los datos si se encuentra un error.

En el lado del remitente: la capa de transporte recibe los datos formateados de las capas superiores, realiza la segmentación y también implementa el control de flujo y error para garantizar una transmisión de datos adecuada. También agrega el número de puerto de origen y destino en su encabezado y reenvía los datos segmentados a la capa de red.

En el lado del receptor: la capa de transporte lee el número de puerto de su encabezado y reenvía los datos que ha recibido a la aplicación correspondiente. También realiza la secuenciación y el reensamblaje de los datos segmentados.

Las funciones de la capa de transporte son:

Segmentación y reensamblaje: esta capa acepta el mensaje de la capa (sesión), divide el mensaje en unidades más pequeñas. Cada uno de los segmentos producidos tiene un encabezado asociado. La capa de transporte en la estación de destino vuelve a ensamblar el mensaje.

Direccionamiento de punto de servicio: para entregar el mensaje al proceso correcto, el encabezado de la capa de transporte incluye un tipo de dirección llamada dirección de punto de servicio o dirección de puerto. Por lo tanto, al especificar esta dirección, la capa de transporte se asegura de que el mensaje se entregue al proceso correcto.

Capa 5 o Capa de sesión

Esta capa es responsable del establecimiento de la conexión, el mantenimiento de las sesiones, la autenticación y también garantiza la seguridad.

Las funciones de la capa de sesión son:

Establecimiento, mantenimiento y finalización de la sesión: la capa permite que los dos procesos establezcan, usen y finalicen una conexión.

Sincronización: esta capa permite que un proceso agregue puntos de verificación que se consideran puntos de sincronización en los datos. Estos puntos de sincronización ayudan a identificar el error para que los datos se vuelvan a sincronizar correctamente y los extremos de los mensajes no se corten prematuramente y se evite la pérdida de datos.

Controlador de diálogo: la capa de sesión permite que dos sistemas inicien la comunicación entre sí en half-duplex o full-duplex.

Capa 6 o Capa de presentación

La capa de presentación también se llama capa de traducción. Los datos de la capa de aplicación se extraen aquí y se manipulan según el formato requerido para transmitir a través de la red.

Las funciones de la capa de presentación son:

Traducción: por ejemplo, ASCII a EBCDIC.

Cifrado / descifrado: el cifrado de datos traduce los datos a otra forma o código. Los datos cifrados se conocen como texto cifrado y los datos descifrados se conocen como texto sin formato. Se utiliza un valor clave para cifrar y descifrar datos.

Compresión: reduce la cantidad de bits que deben transmitirse en la red.

Capa 7 o Capa de Aplicación

En la parte superior de la pila de capas del modelo de referencia OSI, encontramos la capa de aplicación que implementan las aplicaciones de red.

Estas aplicaciones producen los datos, que deben transferirse a través de la red. Esta capa también sirve como una ventana para que los servicios de la aplicación accedan a la red y para mostrar la información recibida al usuario.

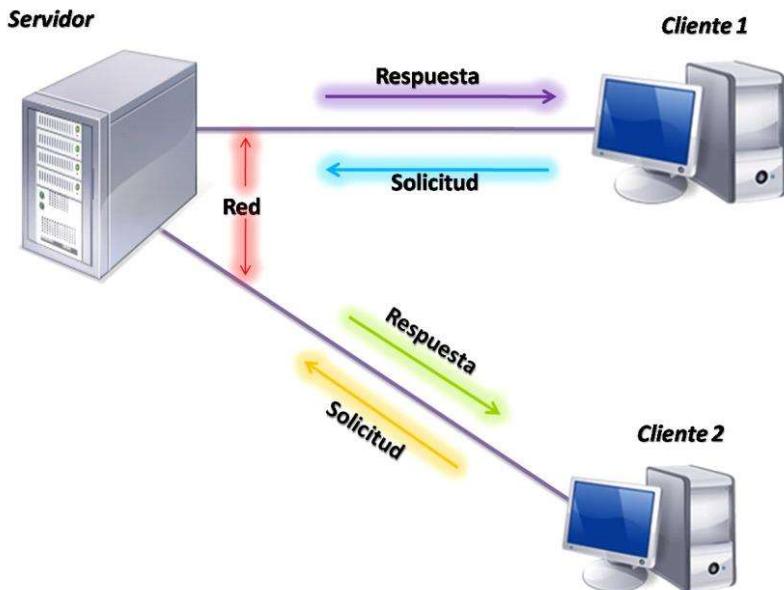
Las funciones de la capa de aplicación son:

Terminal virtual de red
Acceso y gestión de transferencia de archivos FTAM
Servicios de correo
Directorio de Servicios

PROTOCOLO TCP/IP

Arquitectura cliente/servidor

Según TIC, (2016) La estructura cliente - servidor es una arquitectura de computación en la que se consigue un procesamiento cooperativo de la información por medio de un conjunto de procesadores, de tal forma que uno o varios clientes, distribuidos geográficamente o no, solicitan servicios de computación a uno o más servidores.



De esta forma, y gracias a esta arquitectura, la totalidad de los procesadores, clientes y servidores, trabajan de forma cooperativa para realizar un determinado tratamiento de la información.

Atendiendo a esta visión descentralizada, la arquitectura cliente - servidor consiste en una arquitectura distribuida de computación, en la que las tareas de cómputo se reparten entre distintos procesadores, obteniendo los usuarios finales el resultado final de forma transparente, con independencia del número de equipos (servidores) que han intervenido en el tratamiento. Se puede decir por tanto que la arquitectura cliente - servidor es un tipo de arquitectura distribuida, posiblemente la más extendida.

Elementos que forman parte de una arquitectura cliente - servidor

Un sistema Cliente/Servidor es un Sistema de Información distribuido basado en las siguientes características:

Servicio: unidad básica de diseño. El servidor los proporciona y el cliente los utiliza.

Recursos compartidos: Muchos clientes utilizan los mismos servidores y, a través de ellos, comparten tanto recursos lógicos como físicos.

Protocolos asimétricos: Los clientes inician "conversaciones". Los servidores esperan su establecimiento pasivamente.

Transparencia de localización física de los servidores y clientes: El cliente no tiene por qué saber dónde se encuentra situado el recurso que desea utilizar.

Independencia de la plataforma hardware y/o software que se emplee.

Sistemas débilmente acoplados. Interacción basada en envío de mensajes.

Encapsulamiento de servicios. Los detalles de la implementación de un servicio son transparentes al cliente.

Escalabilidad horizontal (añadir clientes) y vertical (ampliar potencia de los servidores).

Integridad: Datos y programas centralizados en servidores facilitan su integridad y mantenimiento.

En el modelo usual Cliente/Servidor, un servidor, (daemon en la terminología sajona basada en sistemas UNIX/LINUX, traducido como "demonio") se activa y espera las solicitudes de los clientes.

Lo normal es que los servicios de un mismo servidor puedan ser utilizados por múltiples clientes distintos. Tanto los programas cliente como los servidores son con frecuencia parte de un programa o aplicación mayores.

ESQUEMA DE FUNCIONAMIENTO DE UN SISTEMA SEGÚN LA ARQUITECTURA CLIENTE - SERVIDOR.

El Esquema de funcionamiento de un Sistema Cliente/Servidor sería:

- El cliente solicita una información al servidor.
- El servidor recibe la petición del cliente.
- El servidor procesa dicha solicitud.
- El servidor envía el resultado obtenido al cliente.
- El cliente recibe el resultado y lo procesa.

COMPONENTES DE LA ARQUITECTURA CLIENTE - SERVIDOR

El modelo Cliente/Servidor es un modelo basado en la idea del servicio, en el que el cliente es un proceso consumidor de servicios y el servidor es un proceso proveedor de servicios. Además esta relación está establecida en función del intercambio de mensajes que es el único elemento de acoplamiento entre ambos.

Esta descomposición principalmente consiste en separar los elementos estructurales de esta tecnología en función de aspectos más funcionales de la misma:

Nivel de Presentación: Agrupa a todos los elementos asociados al componente Cliente.

Nivel de Aplicación: Agrupa a todos los elementos asociados al componente Servidor.

Nivel de comunicación: Agrupa a todos los elementos que hacen posible la comunicación entre los componentes Cliente y servidor.

Nivel de base de datos: Agrupa a todas las actividades asociadas al acceso de los datos.

Elementos principales

CLIENTE

Un cliente es todo proceso que reclama servicios de otro. Una definición un poco más elaborada podría ser la siguiente: cliente es el proceso que permite al usuario formular los requerimientos y pasarlos al servidor. Se lo conoce con el término front-end.

Las funciones que lleva a cabo el proceso cliente se resumen en los siguientes puntos:

- Administrar la interfaz de usuario.
- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.
- Generar requerimientos de bases de datos.
- Recibir resultados del servidor.
- Formatear resultados.

De este modo el cliente se puede clasificar en:

Cliente basado en aplicación de usuario. Si los datos son de baja interacción y están fuertemente relacionados con la actividad de los usuarios de esos clientes.

Cliente basado en lógica de negocio. Toma datos suministrados por el usuario y/o la base de datos y efectúa los cálculos necesarios según los requerimientos del usuario.

SERVIDOR

Un servidor es todo proceso que proporciona un servicio a otros. Es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Al proceso servidor se lo conoce con el término back-end. El servidor normalmente maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos. Las principales funciones que lleva a cabo el proceso servidor se enumeran a continuación:

- Aceptar los requerimientos de bases de datos que hacen los clientes.
- Procesar requerimientos de bases de datos.
- Formatear datos para trasmitirlos a los clientes.
- Procesar la lógica de la aplicación y realizar validaciones a nivel de bases de datos.

MIDDLEWARE

El middleware es un módulo intermedio que actúa como conductor entre sistemas permitiendo a cualquier usuario de sistemas de información comunicarse con varias fuentes de información que se encuentran conectadas por una red. En el caso que nos concierne, es el intermediario entre el cliente y el servidor y se ejecuta en ambas partes.

El middleware se estructura en tres niveles:

- Protocolo de transporte.
- Network Operating System (NOS).
- Protocolo específico del servicio.

Las principales características de un middleware son:

- Simplifica el proceso de desarrollo de aplicaciones al independizar los entornos propietarios.
- Permite la interconectividad de los Sistemas de Información del Organismo.
- Proporciona mayor control del negocio al poder contar con información procedente de distintas plataformas sobre el mismo soporte.
- Facilita el desarrollo de sistemas complejos con diferentes tecnologías y arquitecturas.

COMUNICACIÓN ENTRE LOS ELEMENTOS (NOS)

Como se ha comentado en el apartado anterior, el middleware es un conjunto de aplicaciones encargadas de enlazar al cliente con el servidor. Para ello se estructura en tres capas diferentes:

Protocolo de transporte: comunes a otras aplicaciones.

Network Operating System (NOS).

Protocolo específico del servicio: especiales para distintos tipos de sistemas Cliente/Servidor.

El NOS es el encargado de proporcionar una apariencia de sistema único a un sistema Cliente/Servidor. Se trata pues, de una extensión del Sistema Operativo:

El cliente realiza una llamada a un servicio como si fuera local.

El NOS:

- Intercepta la llamada.
- Redirige la llamada al servidor apropiado.
- Devuelve la contestación.

El NOS debe proporcionar transparencia a los procesos Cliente/Servidor con respecto a:

- **Localización:** Los recursos sólo se conocen por su nombre. El sistema en el que se ejecutan es irrelevante.
- **Espacio de nombres:** Las convenciones de los nombres de los recursos deben ser iguales, independientemente del sistema que los soporte.
- **Conexión:** Un único usuario y contraseña para todo el sistema.
- **Replicación:** No se debe diferenciar entre copias de un mismo recurso.
- **Acceso local / remoto:** El acceso a un recurso se debe realizar como si estuviera localizado en el mismo sistema que el programa cliente.
- **Tiempo:** Los relojes de todos los elementos del sistema deben estar sincronizados.
- **Fallos:** El sistema debe proporcionar servicios de detección de fallos, redundancia y reconexión tras un fallo.
- **Administración:** Un único sistema de gestión de todos los recursos.
- **Protocolos:** Idéntica interfaz de programación para todos los protocolos de transporte.

Tipos de arquitectura cliente / servidor

Uno de los aspectos claves para entender la tecnología Cliente/Servidor, y por tanto contar con la capacidad de proponer y llevar a cabo soluciones de este tipo, es llegar a conocer la arquitectura de este modelo y los conceptos o ideas asociados al mismo.

Un esquema de clasificación basado en los conceptos de Fat Client/Thin Client, Fat Server/Thin Server, es decir, basado en el tamaño de los componentes. En segundo lugar tenemos una clasificación según la naturaleza del servicio que nos ofrecen.

TIPOS DE ARQUITECTURA CLIENTE - SERVIDOR POR TAMAÑO DE COMPONENTES.

Este tipo de clasificación se basa en los grados de libertad que brinda el modelo Cliente/Servidor para balancear la carga de proceso entre los niveles de presentación, aplicación y base de datos.

Dependiendo de qué segmento de las capas de software tenga que soportar la mayor o menor carga de procesamiento, se habla de Fat Client (Thin Server) o Fat server (Thin Client).

Consideraciones de este tipo son importantes en el momento de decidir una plataforma de desarrollo, al mismo tiempo que pueden definir la viabilidad o no de las mismas para enfrentar un cierto número de restricciones impuestas por una problemática a resolver.

FAT CLIENT (THIN SERVER)

En este esquema de arquitectura el peso de la aplicación es ejecutada en el cliente, es decir, el nivel de presentación y el nivel de aplicación corren en un único proceso cliente, y el servidor es relegado a realizar las funciones que provee un administrador de base de datos.

En general este tipo de arquitectura tiene mejor aplicación en sistemas de apoyo de decisiones (DSS: Decision Support System) y sistemas de información ejecutiva (EIS: Executive Information System), y como se concluirá más adelante, tiene pocas posibilidades de aplicarse en sistemas de misión crítica.

FAT SERVER (THIN CLIENT)

Este es el caso opuesto al anterior, el proceso cliente es restringido a la presentación de la interfaz de usuario, mientras que el peso de la aplicación corre por el lado del servidor de aplicación.

En general este tipo de arquitectura presenta una flexibilidad mayor para desarrollar una gran variedad de aplicaciones, incluyendo los sistemas de misión crítica a través de servidores de transacciones.

TIPOS DE ARQUITECTURA CLIENTE - SERVIDOR SEGÚN

LA NATURALEZA DE SERVICIO PROPORCIONADO.

SERVIDORES DE FICHEROS

Con un servidor de archivos, un cliente lo que hace es requerimientos de los mismos sobre una red. Esta es una forma muy primitiva de servicios de datos, la cual necesita intercambio de muchos mensajes sobre una red para hallar el dato requerido.

Los servidores de archivos usan recursos compartidos sobre la red y son necesarios para crear repositorios de documentos, imágenes y archivos grandes sobre la red.

SERVIDORES DE BASES DE DATOS

Este análisis está elaborado desde el punto de vista del modelo Cliente/Servidor, y está directamente relacionado con la arquitectura en dos planos, que se describirá en el apartado siguiente.

Obviamente la creación de aplicaciones Cliente/Servidor está asociada a la utilización de servidores de bases de datos relacionales SQL, y dependiendo de los requerimientos y restricciones se debe elegir entre una arquitectura dos o tres planos.

SERVIDORES DE TRANSACCIONES

Estos tipos de sistemas se pueden implementar con cualquiera de las modalidades Cliente/Servidor en dos o tres planos, pero incorporan un elemento principal sobre el cual se elabora y basa toda la fortaleza de este modelo, el concepto de transacción.

Con un servidor de transacciones el proceso cliente llama a funciones, procedimientos o métodos que residen en el servidor, ya sea que se trate de un servidor de bases de datos o un servidor de aplicaciones.

SERVIDORES DE OBJETOS

Con un servidor de objetos, las aplicaciones Cliente/Servidor son escritas como un conjunto de objetos que se comunican. Los objetos cliente se comunican con los objetos servidores usando un Object Request Broker (ORB). El cliente invoca un método de un objeto remoto. El ORB localiza el método del objeto en el servidor, y lo ejecuta para devolver el resultado al objeto cliente.

Los servidores de objetos deben soportar concurrencia. La parte central de la comunicación en los servidores de objetos es el ORB:

- Elemento central y principal de esta arquitectura.
- Bus de objetos. Permite la comunicación entre ellos.
- Middleware avanzado: Permite llamadas estáticas y dinámicas a objetos.
- Lenguaje de descripción de interfaces independiente del lenguaje de programación.

SERVIDORES WEB

La primera aplicación cliente servidor que cubre todo el planeta es el World Wide Web. Este nuevo modelo consiste en clientes simples que hablan con servidores Web. Un servidor Web devuelve documentos cuando el cliente pregunta por el nombre de los mismos. Los clientes y los servidores se comunican usando un protocolo basado en RPC, llamado HTTP. Este protocolo define un conjunto simple de comandos, los parámetros son

pasados como cadenas y no provee tipos de datos. La Web y los objetos distribuidos están comenzando a crear un conjunto muy interactivo de computación Cliente/Servidor.

Modelos cliente/servidor

Una de las clasificaciones mejor conocidas de las arquitecturas Cliente/Servidor se basa en la idea de planos (tier), la cual es una variación sobre la división o clasificación por tamaño de componentes.

A nivel de software

Este enfoque o clasificación es el más generalizado y el que más se ajusta a los enfoques modernos, dado que se fundamenta en los componentes lógicos de la estructura Cliente/Servidor y en la madurez y popularidad de la computación distribuida

MODELO CLIENTE/SERVIDOR 2 CAPAS

Esta estructura se caracteriza por la conexión directa entre el proceso cliente y un administrador de bases de datos. Dependiendo de donde se localice el grupo de tareas correspondientes a la lógica de negocios se pueden tener a su vez dos tipos distintos dentro de esta misma categoría:

IMPLEMENTADO CON SQL REMOTO

En este esquema el cliente envía mensajes con solicitudes SQL al servidor de bases de datos y el resultado de cada instrucción SQL es devuelto por la red, no importando si son uno, diez, cien o mil registros. Es el mismo cliente quien debe procesar todos los registros que le fueron devueltos por el servidor de base de datos, según el requerimiento que él mismo hizo.

Ventajas:

Presenta una estructura de desarrollo bastante simple ya que el programador maneja un único ambiente de desarrollo (es más simple respecto al Cliente/Servidor en tres planos, puesto que reduce una capa de programación, como se verá más adelante).

Inconvenientes:

La gran cantidad de información que viaja al cliente congestionada demasiado el tráfico de red, lo que se traduce en bajo rendimiento.

Por su bajo rendimiento esta estructura tiene un bajo espectro de aplicación, limitándose a la construcción de sistemas no críticos.

IMPLEMENTADO CON PROCEDIMIENTOS ALMACENADOS

En este esquema el cliente envía llamadas a funciones que residen en la base de datos, y es ésta quien resuelve y procesa la totalidad de las instrucciones SQL agrupadas en la mencionada función.

Ventajas: Presenta las mismas ventajas de una arquitectura dos planos con procedimientos almacenados, pero mejora considerablemente el rendimiento sobre ésta, dado que reduce el tráfico por la red al procesar los datos en la misma base de datos, haciendo viajar sólo el resultado final de un conjunto de instrucciones SQL.

Inconvenientes: Si bien la complejidad de desarrollo se ve disminuida, se pierde flexibilidad y escalabilidad en las soluciones implantadas. Obliga a basar el peso de la aplicación en SQL extendido, propios del proveedor de la base de datos que se elija. Debiera considerarse que si bien los procedimientos almacenados (stored procedures), los desencadenantes (triggers) y las reglas (constraint) son útiles, en rigor son ajenos al estándar de SQL.

MODELO CLIENTE/SERVIDOR 3 CAPAS

Esta estructura se caracteriza por elaborar la aplicación en base a dos capas principales de software, más la capa correspondiente al servidor de base de datos. Al igual que en la arquitectura dos capas, y según las decisiones de diseño que se tomen, se puede balancear la carga de trabajo entre el proceso cliente y el nuevo proceso correspondiente al servidor de aplicación.

Ventajas:

Reduce el tráfico de información en la red por lo que mejora el rendimiento de los sistemas (especialmente respecto a la estructura en dos planos).

Brinda una mayor flexibilidad de desarrollo y de elección de plataformas sobre la cual montar las aplicaciones. Provee escalabilidad horizontal y vertical.

Se mantiene la independencia entre el código de la aplicación (reglas y conocimiento del negocio) y los datos, mejorando la portabilidad de las aplicaciones.

Inconvenientes:

Dependiendo de la elección de los lenguajes de desarrollo, puede presentar mayor complejidad en comparación con Cliente/Servidor dos planos.

Existen pocos proveedores de herramientas integradas de desarrollo con relación al modelo Cliente/Servidor dos planos, y normalmente son de alto costo.

A nivel de hardware

Esta clasificación del modelo Cliente/Servidor se basa igualmente en la distribución de los procesos y elementos entre sus componentes, pero centrándose en la parte física del mismo, en el que la administración de la interfaz gráfica se asocia a los clientes PC y la seguridad e integridad de los datos quedan asociados a ambientes mainframe o por lo menos a servidores locales y/o centrales.

MODELO CLIENTE / SERVIDOR 2 CAPAS

Los clientes son conectados vía LAN a un servidor de aplicaciones local, el cual, dependiendo de la aplicación puede dar acceso a los datos administrados por él.

MODELO CLIENTE / SERVIDOR 3 CAPAS

Los clientes son conectados vía LAN a un servidor de aplicaciones local, el cual a su vez se comunica con un servidor central de bases de datos. El servidor local tiene un comportamiento dual, dado que actúa como cliente o servidor en función de la dirección de la comunicación.

Tipos de Servidores según el servicio realizado

FTP – Correo – Web – IMAP - Streaming

Un servidor o server, en el lenguaje informático, es un ordenador y sus programas, que están al servicio de otros ordenadores.

El servidor atiende y responde a las peticiones que le hacen los otros ordenadores. Los otros ordenadores, que le hacen peticiones, serán los "clientes" del servidor.

Precisamente se llaman servidores porque sirven cosas y están al servicio de otros ordenadores.

Por ejemplo si tienes un correo electrónico, lo recibes de un servidor de correo electrónico, si deseas ver una página web, la recibes de un servidor web, si trabajas en una red de ordenadores todos los servicios compartidos de la red estarán en un servidor de red y así otros muchos servicios y tipos de servidores que veremos.

El modelo o arquitectura que siguen los servidores es el de cliente-servidor, es decir el cliente/s pide y el servidor proporciona los recursos o servicios.

Los servidores se utilizan para gestionar los recursos de una red.

Un servidor deberá estar siempre encendido, ya que si se apaga dejará de dar servicio a los demás. Cuando un servidor falla (se apaga o tiene errores) hace que los demás usuarios de la red tengan problemas, porque no disponen de los servicios que proporciona ese servidor.

Por ejemplo, un usuario puede configurar un servidor para controlar el acceso a una red, enviar/recibir correo electrónico, gestionar los trabajos de impresión, o alojar un sitio web.

La red más conocida y más grande es Internet, y está llena de servidores. Pero ojo hay servidores dentro de redes pequeñas y particulares, incluso tu puedes hacer que tu propio ordenador sea un servidor.

Dependiendo del servicio que de el servidor, tiene que disponer de software (programas) específicos capaces de ofrecer esos servicios. El hardware es simplemente un ordenador, aunque es recomendable que sea de gama alta, para dar respuesta a las peticiones lo más rápido posible.

En la siguiente imagen vemos el apilamiento de los servidores de una empresa que se dedica a proporcionar almacenamiento de información, también llamado Cloud Computing.

servidores de internet

Normalmente, la mayoría de los servidores están diseñadas para operar sin ninguna intervención manual durante su funcionamiento. Eso sí, antes se deberán configurar correctamente.

Tipos de Servidores

Vamos a ver los principales tipos de servidores y explicar para qué sirve cada uno:

- **Servidor de Correo Electrónico o Mail Server:** Es un ordenador dentro de una red que funciona como una oficina de correo virtual. Transfiere y almacena los mensajes de correo electrónico a través de una red.

Estos servidores tienen programas capaces de almacenar correos para los usuarios locales y con un conjunto de reglas definidas por el usuario que determinan cómo el servidor de correo debe reaccionar ante el destino de un mensaje específico.

Normalmente estos servidores se dividen en otros 2 diferentes, una para el correo entrante (llamados POP3) y otro para el correo saliente (llamados SMTP).

Los servidores POP3 retienen los mensajes de correo electrónico entrantes hasta que el usuario compruebe su correo y entonces los transfieren al equipo cuando el usuario lo pide.

Los servidores SMTP administran el envío de los mensajes de correo electrónico a Internet. El servidor SMTP administra el correo electrónico saliente y se utiliza en combinación con un servidor POP3 o IMAP de correo electrónico entrante. Cuando el usuario da la orden de enviar, el servidor lo envía.

Otro tipo de servidores de correo son los IMAP que permiten trabajar con los mensajes de correo electrónico sin necesidad de descargarlos antes al equipo. Puedes obtener una vista previa, eliminar y organizar los mensajes directamente en el servidor de correo sin descargarlos en tu equipo. Ejemplos son los correos de yahoo, Hotmail, etc.

También están los servidores Fax que hacen lo mismo que los de correo, pero para la recepción y transmisión de faxes.

- **Servidor FTP:** Se trata de uno de los más antiguos en Internet, "file transfer protocol" o en Español Protocolo Para la Transferencia de Archivos. Se utilizan para realizar una transferencia segura de archivos entre ordenadores (envío de archivos de un sitio a otro). Los FTP garantiza la seguridad de los archivos y control de su transferencia.

En este caso el cliente 1 envía una petición al servidor FTP para que le envíe un archivo al cliente 2. El servidor se lo envía y el cliente 2 lo recibe. Todo este proceso se realiza mediante un programa llamado FTP instalado en el cliente 1 y en el 2. El servidor dispondrá de otro programa (software) que se encargará de la recepción y el envío.

Este tipo de servidores se utilizan para subir archivos de páginas web a los servidores web, archivos de imágenes, videos, para hacer backup (copias de seguridad), etc.

- **Web Server o Servidor Web:** Todas las páginas web que puedes ver por internet están almacenadas en servidores, llamados servidores web.

Un servidor web almacena los archivos de una web y los proporciona a los clientes que los solicitan haciendo la transferencia de los archivos a través de la red mediante los navegadores. El cliente lo pide a través de su navegador y el servidor web lo envía al mismo navegador del cliente para que este lo pueda visualizar.

Los archivos web incluyen texto, imágenes, videos, etc.. y que solo los navegadores pueden visualizar.

El servidor "sirve" (envía) el archivo web (por ejemplo una web en formato html) al navegador del cliente para que lo pueda visualizar. El servidor, el navegador y la comunicación a través de la red seguirán unas normas llamadas "protocolo HTTP".

El espacio que te dejan estos servidores para alojar tu web se llama Hosting. Hay dos tipos principales de hosting:

Hosting Compartido: en el servidor web hay varias páginas alojadas de distintos clientes.

Hosting Dedicado: tienes un servidor para ti solito donde puedes alojar tus webs. Lógicamente son más caros.

Muchas veces se dice servidor web compartido o dedicado para hacer referencia a este tipo de hosting.

Los servidores web utilizan programas específicos para administrar sus servicios. En función del programa que utiliza el servidor web para administrar y servir las páginas web pueden ser de varios tipo. Todos los tipos que vamos a ver a continuación son realmente programas de gestión del servidor web (software).

Tipos de Servidores Web

- Servidor Apache HTTP: Este es el servidor web más popular del mundo desarrollado por la Apache Software Foundation. El servidor web Apache es un software de código abierto y se puede instalar en casi todos los sistemas operativos incluyendo Linux, Unix, Windows, FreeBSD, Mac OS X y más. Alrededor del 60% de los ordenadores usados como servidor web ejecuta el servidor Web Apache.
- Microsoft IIS es un Servidor Web de alto rendimiento de Microsoft. Este servidor Web se ejecuta en plataformas Windows NT / 2000 y 2003 (y en las próximas nuevas versiones de Windows también). IIS viene incluido con Windows NT / 2000 y 2003; Dado que IIS está estrechamente integrado con el sistema operativo, es relativamente fácil administrarlo.
- El Lighttpd , pronunciado lighty es también un servidor web gratuito que se distribuye con el sistema operativo FreeBSD. Este servidor web de código abierto es rápido, seguro y consume mucha menos energía de la CPU. Lighttpd también se puede ejecutar en los sistemas operativos Windows, Mac OS X, Linux y Solaris.

- Sun Java System Web Server es un servidor web adecuado para grandes sitios web de medianas y grandes empresas. Aunque el servidor es libre no es de código abierto. Sin embargo, se ejecuta en plataformas Windows, Linux y Unix. El servidor web de Sun Java System soporta varios idiomas, guiones y tecnologías necesarias para la Web 2.0, tales como JSP, servlets Java, PHP, Perl, Python, Ruby on Rails, ASP y ColdFusion, etc.

- Jigsaw (Servidor de W3C) proviene del World Wide Web Consortium. Es de código abierto y libre y puede ejecutarse en varias plataformas como Linux, Unix, Windows, Mac OS X Free BSD, etc. Jigsaw ha sido escrito en Java y se puede ejecutar scripts CGI y programas PHP.

- El servidor Ngnix es un servidor Web muy ligero y trabaja sobre sistemas Unix y Windows. Se ha convertido en el 4º servidor HTTP más popular de la red y también se distribuye bajo licencia BSD. Se utiliza en el 19% de los servidores web.

Dentro de la red de internet hay unos servidores que se llaman DNS que son los que se encargan de gestionar los nombres de los dominios de las páginas web (las direcciones de las webs). Estos servidores se llaman Servidores DNS. Para saber más sobre esto visita el siguiente enlace: ¿Qué es el DNS y servidores DNS?.

- Servidores Proxy o Servidores de Red: Se utilizan para administrar una red de ordenadores, permitiendo el acceso o no a la red de los clientes. Suelen incluir protección de la red como por ejemplo un firewall (cortafuegos).

- Servidores de Bases de Datos: Son ordenadores preparados para alojar bases de datos para ser utilizadas por uno o más clientes. Además estos servidores realizan tareas como el análisis de los datos, el almacenamiento, la manipulación de datos, y otras tareas específicas.

- Servidores de Audio/Vídeo: Permiten transmitir contenido multimedia en streaming. El streaming es una técnica de envío continuo de información, que permite por ejemplo, ir viendo una película según se va descargando, sin necesidad de descargarla por completo para visualizarla.

- Chat Server o Servidor Chat: Es un equipo dedicado a manejar y mantener un chat y sus usuarios. Los más famosos son los IRC. Ahora también se les conoce como servidores en tiempo real, porque permiten intercambiar información de forma instantánea.

- Servidores Groupware: Son servidores que facilitan el trabajo en grupo de varios ordenadores, con un objetivo común (por ejemplo un proyecto).

Estos servidores disponen de software que permite colaborar a los usuarios del servidor independientemente de donde están ubicados, permitiéndoles así hacer un trabajo colaborativo.

Los archivos y datos almacenados en un servidor groupware pueden ser alterados, acceder y recuperados por los miembros del grupo de trabajo. Groupware también se conoce como software de colaboración.

- **Servidor Telnet:** Son servidores que nos permiten iniciar sesión en cualquier ordenador y realizar tareas en otro ordenador. Podemos trabajar con nuestro ordenador de forma remota, es decir desde otro ordenador.

- **Servidor SIP:** Se encargan de gestionar el establecimiento de las llamadas telefónicas por internet. Los SIP almacenan la dirección IP donde deben acceder para realizar la comunicación con un usuario. No transmite ni audio ni video, solo establece la comunicación.

- **List Server o Servidores Lista:** Permiten gestionar listas de correos.

- **Servidores Cloud:** Realmente estos servidores lo único que hacen es dejarte o alquilarte un espacio del servidor. La mayoría se utilizan para almacenar grandes cantidades de información en el servidor y tenerla protegida fuera de nuestro ordenador. Muchas empresas alquilan servidores cloud (en la nube) para tener en ellos toda la valiosa información de la empresa, utilizándola cuando quieran y realizando el propio servidor copias de seguridad.

- **Cluster de Servidores:** Un clúster de servidores es la agrupación de varios servidores dedicados a la misma tarea. Hay veces que un solo servidor se queda pequeño para toda la demanda de los clientes y es necesario más. En estos casos se agrupan en lo que se conoce como Cluster de Servidores.

Plataformas en la Nube

Al momento de elegir qué plataforma usar para desarrollar nuestros proyectos digitales es necesario estar bien educado sobre todos los esfuerzos necesarios que representan desplegar proyectos que requieren poder de cómputo.

Normalmente nos encontramos con diferentes preguntas como: ¿es mejor una infraestructura física o en la nube? ¿Cuál es la mejor opción en la nube? ¿Cuál opción es más barata? ¿que beneficios tienen cada una?

Comencemos por lo tradicional, en la mayoría de empresas prefieren los centros de datos físicos, los cuales requieren una inversión significativa en equipos de hardware, software, instalaciones y un ejército de personal capacitado para la instalación y mantenimiento de todo el Data Center.

Esto representa un impedimento para los proyectos que no cuentan con grandes presupuestos, sin embargo, existen grandes compañías que ya están apostando por las tecnologías de virtualización para el procesamiento de datos a gran escala.

El auge de las tecnologías cloud ha hecho accesible disponer de servicios en poder de cómputo de clase mundial, desde multinacionales hasta dos estudiantes, que desde sus casas desarrollan una nueva app.

En la actualidad existen 3 empresas que reinan en el mundo del Cloud Computing como son: Microsoft Azure, Google Cloud Platform y Amazon Web Service, cada uno con una amplia gama de productos que cumplen diversas funciones.

En esta oportunidad abordaremos los tres al mismo tiempo para explorar cual elegir al momento de desarrollar proyectos propios o cuando vendemos servicios.

Productos y Soluciones

Vamos a utilizar los términos “productos” y “servicios” indistintamente; una solución, sin embargo, es un concepto más específico que se escucha mucho cuando se trata de servicios en la nube.

En pocas palabras, una solución es un conjunto de productos preconfigurados orientadas a una necesidad muy específica. Con abundante documentación, casos de uso y los testimonios que te guiarán a través del proceso de adopción de la infraestructura en la nube. Amazon Aws vs Microsoft Azure vs Google Cloud.

Uno de los grandes beneficios de los proveedores de servicios Cloud como Amazon Web services, GCP y otros, son sus estrategias de precios competitivos y flexibles.

Estas empresas están en una constante lucha para brindar la mejor oferta al consumidor, en su mayoría apuestan al método de Pago por Uso (PAYG por sus siglas en inglés).

Anteriormente se pagaba por uso mensual, sin embargo las cosas están cambiando y girando hacia la tendencia de cobrar por segundo de potencia en cómputo.

Precios en AWS.

La novedosa tendencia del PAYG hace que los precios se ajusten a las necesidades del proyecto, no es necesario comprar costosas soluciones de nube privada que se vuelven obsoleta en poco tiempo. Con Amazon Web Services solo necesita ajustar su máquina virtual y la nube de AWS se adaptará según los requerimientos, para solo pagar por el poder de cómputo usado.

Desafortunadamente cuando se trata de almacenar grande volúmenes de datos (entre 50 TB y 500 TB) AWS pierde rendimiento porcentual.

Amazon puede ser ideal para colocar grandes bases de datos en la nube pero cuando se trata de subir aplicaciones la nube de Microsoft Azure está mejor equipada.

Estos detalles de precios de AWS son útiles porque, como Azure y Google, los niveles de niveles de nube varían mucho y es bueno compararlos con los requisitos que necesita nuestro proyecto.

También tenemos el Costo Total de la Propiedad (TCO por sus siglas en inglés) que es importante cuando se construye un caso de negocios y se obtiene una mejor estimación de lo que se necesita para satisfacer las necesidades de la organización.

Precios de Microsoft Azure

Al igual que AWS, esta herramienta de Microsoft tiene muchos elementos convincentes, como el desglose de precios dependiendo de las necesidades del proyecto, sin embargo sus costos son mas agresivos que GCP y AWS.

Esta forma agresiva de marcar costos es con la finalidad de liderar los monopolios Cloud, como es el segmento de mover aplicaciones a la nube donde compite de manera agresiva por esta cuota de mercado.

Para Azure, la calculadora de costo total de la propiedad hace las siguientes preguntas:

¿Desea reducir el costo total de propiedad de su infraestructura local?

¿Cuáles son los ahorros de costos estimados de migrar cargas de trabajo de aplicaciones a Microsoft Azure?

Precios de Google Cloud Platform

Por último la división de negocios del gigante de los buscadores, la cual busca posicionarse cara a cara con sus competidores más cercanos, al tiempo que enfatiza su facturación solo por el costo exacto de uso en potencia de cómputo.

Más allá de comparar el precio de estos tres pesos pesados de la nube, sus características también son un factor muy interesante a la hora de comparar.

En general, estas comparaciones son muy útiles cuando se considera que partner de la nube es el más adecuado para el resultado deseado.

Por ejemplo, aunque todos pueden cubrir análisis de datos y visualización, se puede pensar que AWS es el más progresivo en esta área.

Los tres AWS, Azure y Google tienen su propia forma de categorizar los diferentes elementos, por lo que sugerimos comenzar a evaluar según las necesidades del proyecto y como cada herramienta se ajusta según sus características.

Desafortunadamente, a menudo se ven organizaciones que están tan comprometidas con Azure, por ejemplo, que no reconocen alternativas posiblemente más económicas y eficientes como AWS.

AWS: características

Al igual que los otros dos proveedores de servicios en la nube, AWS tiene diferentes algoritmos con nombres para desglosar sus productos y dividirlos en las siguientes categorías:

- Compute
- Storage
- Database
- Migration
- Networking & Content Delivery
- Developer Tools
- Management Tools
- Security, Identity & Compliance
- Analytics
- Artificial Intelligence
- Mobile Services
- Applications Services
- Messaging
- Business Productivity
- Desktop & App Streaming
- Software
- Internet of Things
- Game Development

Además de esta amplia gama de opciones, en AWS tienen productos específicos con un alto grado de categorización.

Estas soluciones cubren:

- Sitios web
- Copia de seguridad y recuperación
- Archivo
- Recuperación de desastres
- DevOps
- Big Data

Microsoft Azure: características

En la nube de Microsoft tenemos una amplia gama de opciones, muy parecidas a las de AWS con el diferencial de proporcionar ciertas capacidades basadas en usuarios.

Estos beneficios también incluyen la facturación flexible y precios competitivos, cabe destacar que Azure afirma tener un grado de certificación en estándares internacionales mayor a la de sus competidores.

Asegurar esta superioridad es una jugada bastante audaz del gigante de la computación, a pesar de tener un rango de características muy similar a AWS, sin embargo esto lo hacen con la finalidad de buscar que sus clientes depositen la confianza en su nube.

Google Cloud Platform: características

Aunque no es necesariamente el proveedor de computación en la nube más histórico, está lanzándose al mundo del Cloud Computing.

Google tiene tres puntos clave detrás de sus soluciones, destacando:

- Infraestructura a prueba de futuro
- Datos y análisis serios y potentes
- Sin servidor, solo código

Con todas estas características de cada una de las herramientas, no es fácil saber cual es la más conveniente, sin embargo con este análisis podrás tener una visión más amplia y ajustar todo según tus requerimientos y recursos.

Aws Vs Azure Vs Google Cloud Platform: Implementación

En lo que respecta al despliegue de proyectos en estas nubes privadas, todas poseen opciones de implementación relativamente sencillas, cabe destacar que esto también depende de la complejidad de lo que necesitemos hacer.

Implementación de AWS <https://aws.amazon.com/>

La nube de Amazon proporciona una página de implementación muy fácil e intuitiva de usar para comenzar a usar sus herramientas y soluciones.

Proporcionan un IDE donde puedes elegir qué plataforma usar, por lo tanto si estás escribiendo código PHP o implementando una App en iOS, te proporcionan códigos de inicio para hacerlo de forma más sencilla.

Implementación de Microsoft Azure <https://azure.microsoft.com/>

Microsoft proporciona beneficios en la implementación con guías muy detalladas en profundidad bien sea para los departamentos más relevantes.

- Azure Developer Guide
- Azure Operations Guide

Con estas dos guías y otros recursos como videos instructivos es posible hacer la implementación según nuestros intereses.

Implementación de Google Cloud Platform <https://cloud.google.com/>

Google siempre se ha caracterizado por tener productos enfocados en el usuario, en este caso llaman “Cloud Launcher” a su proceso de implementación, el cual proporciona documentos de inicio en su nube y diversos beneficios de sus servicios.

Si quieras documentarte mejor con GCP puedes visitar el sitio de <https://cloud.google.com/why-google-cloud/> aquí.

¿Cuál elegir?

Ya vimos en detalle los precios, características e implementación, todo con la finalidad de informarte cuando necesites tomar una decisión tan importante.

Si bien no es nada fácil llegar a una decisión concreta por el número de variables involucradas, al menos tendrás el conocimiento necesario para discernir por la opción correcta de soluciones cloud para tu empresa.