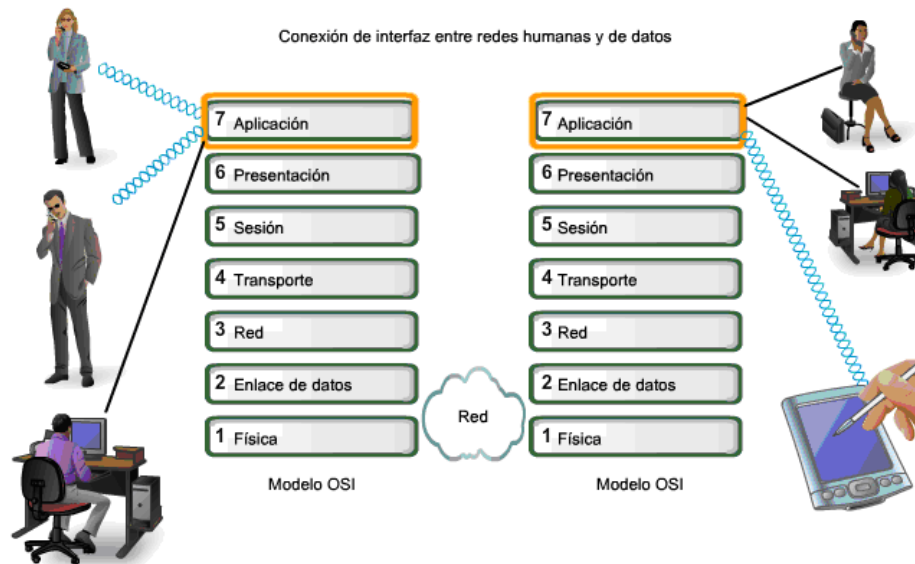


FUNCIONALIDAD y PROTOCOLOS DE LA CAPA DE APLICACIÓN

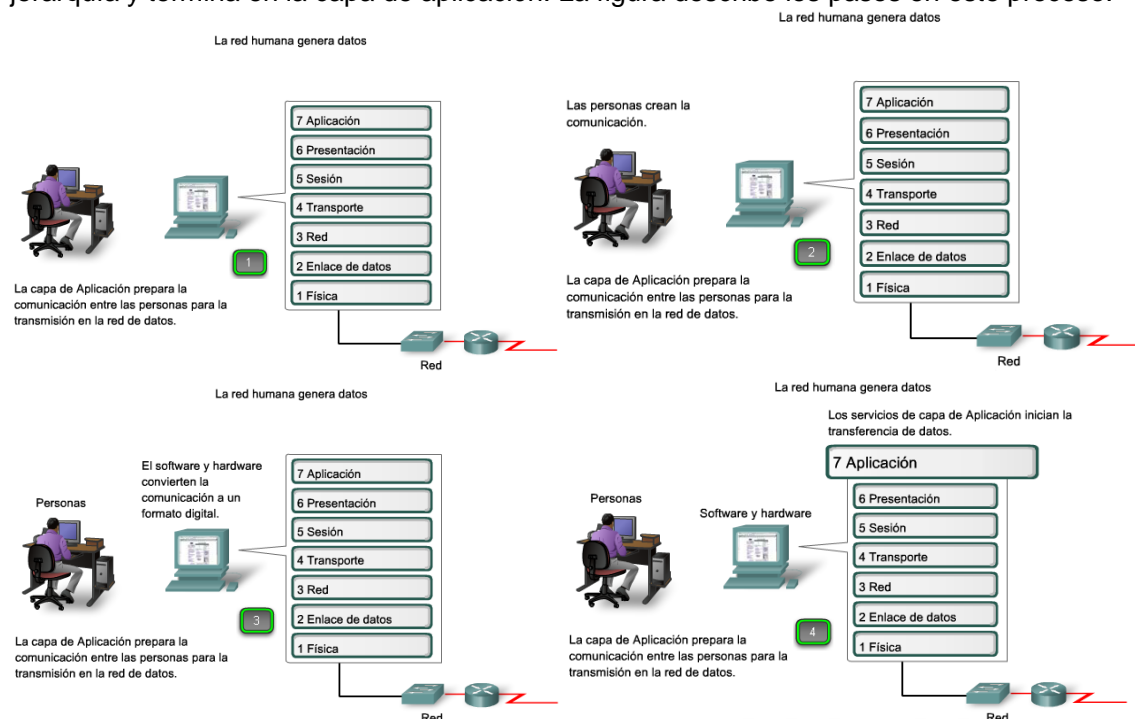


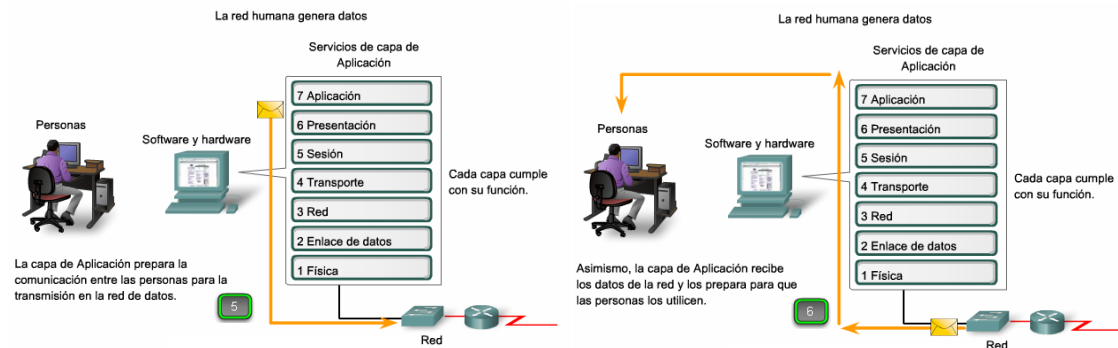
La capa de Aplicación ofrece la interfaz a la red

MODELO OSI y MODELO TCP/IP

El modelo de interconexión de sistemas abiertos es una representación abstracta en capas, creada como guía para el diseño del protocolo de red. El modelo OSI divide el proceso de networking en diferentes capas lógicas, cada una de las cuales tiene una funcionalidad única y a la cual se le asignan protocolos y servicios específicos.

En este modelo, la información se pasa de una capa a otra, comenzando en la capa de aplicación en el host de transmisión, siguiendo por la jerarquía hacia la capa física y pasando por el canal de comunicaciones al host de destino, donde la información vuelve a la jerarquía y termina en la capa de aplicación. La figura describe los pasos en este proceso.





La capa de aplicación, la séptima capa, es la capa superior de los modelos OSI y TCP/IP. Es la capa que proporciona la interfaz entre las aplicaciones que utilizamos para comunicarnos y la red subyacente en la cual se transmiten los mensajes. Los protocolos de capa de aplicación se utilizan para intercambiar los datos entre los programas que se ejecutan en los hosts de origen y destino. Existen muchos protocolos de capa de aplicación y siempre se desarrollan protocolos nuevos.

Aunque el grupo de protocolos TCP/IP se desarrolló antes de la definición del modelo OSI, la funcionalidad de los protocolos de la capa de aplicación de TCP/IP se adaptan aproximadamente a la estructura de las tres capas superiores del modelo OSI. Capas de aplicación, presentación y sesión.

La mayoría de los protocolos de la capa de aplicación de TCP/IP se desarrollaron antes de la aparición de computadoras personales, interfaces del usuario gráficas y objetos multimedia. Como resultado, estos protocolos implementan muy poco de la funcionalidad que es específica en las capas de presentación y sesión del modelo OSI.

La capa de presentación

La capa de presentación tiene tres funciones principales:

- **Codificación y conversión de datos** de la capa de aplicación para garantizar que los datos del dispositivo de origen se puedan interpretar por la aplicación adecuada en el dispositivo de destino.
- **Compresión de los datos** de forma que los pueda descomprimir el dispositivo de destino.
- **Encriptación de los datos** para la transmisión y la encriptación de los mismos cuando lleguen a su destino.

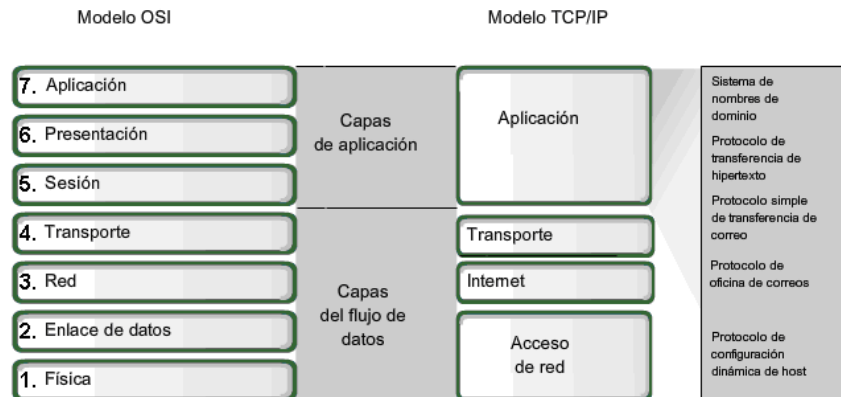
Generalmente, las implementaciones de la capa de presentación no están relacionadas con un stack de protocolos en particular. Los **estándares para videos y gráficos** son algunos ejemplos. Dentro de los estándares más conocidos para video encontramos QuickTime y el Grupo de expertos en películas (MPEG). QuickTime es una especificación de Apple Computer para audio y video, y MPEG es un estándar para la codificación y compresión de videos.

Dentro de los **formatos de imagen gráfica** más conocidos encontramos el Formato de intercambio gráfico (GIF), Grupo de expertos en fotografía (JPEG) y Formato de archivo de imagen etiquetada (TIFF). GIF y JPEG son estándares de compresión y codificación para imágenes gráficas, y TIFF es un formato de codificación estándar para imágenes gráficas.

La capa de sesión

Como lo indica el nombre de la capa de sesión, **las funciones en esta capa crean y mantienen diálogos entre las aplicaciones de origen y destino**. La capa de sesión maneja el intercambio de información para iniciar los diálogos y mantenerlos activos, y para reiniciar sesiones que se interrumpieron o desactivaron durante un periodo de tiempo prolongado.

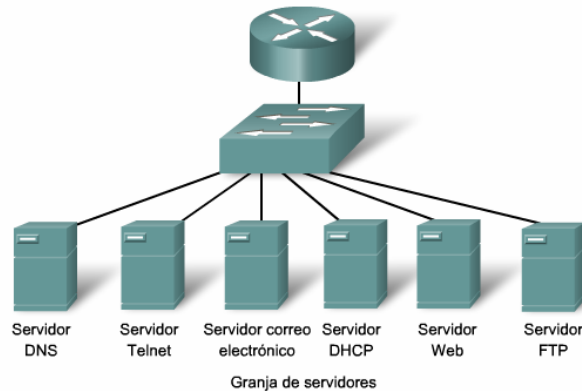
La mayoría de las aplicaciones, como los exploradores Web o los clientes de correo electrónico, incorporan la funcionalidad de las Capas 5, 6 y 7 del modelo OSI.



Los protocolos de capa de aplicación de TCP/IP más conocidos son aquéllos que proporcionan intercambio de la información del usuario. Estos protocolos especifican la información de control y formato necesaria para muchas de las funciones de comunicación de Internet más comunes. Algunos de los protocolos TCP/IP son:

- El **Protocolo servicio de nombres de dominio (DNS, Domain Name Service)** se utiliza para resolver nombres de Internet para direcciones IP.
- El **Protocolo de transferencia de hipertexto (HTTP, Hypertext Transfer Protocol)** se utiliza para transferir archivos que forman las páginas Web de la World Wide Web.
- El **Protocolo simple de transferencia de correo (SMTP)** se utiliza para la transferencia de mensajes de correo y adjuntos.
- **Telnet**, un protocolo de emulación de terminal, se utiliza para proporcionar acceso remoto a servidores y a dispositivos de red.
- El **Protocolo de transferencia de archivos (FTP)** se utiliza para la transferencia de archivos interactiva entre sistemas.

Los protocolos en la suite de TCP/IP los definen generalmente las Solicitudes de comentarios (RFC). El Grupo de trabajo de ingeniería de Internet mantiene las RFC como los estándares para la suite de TCP/IP.



SOFTWARE DE LA CAPA DE APLICACIONES

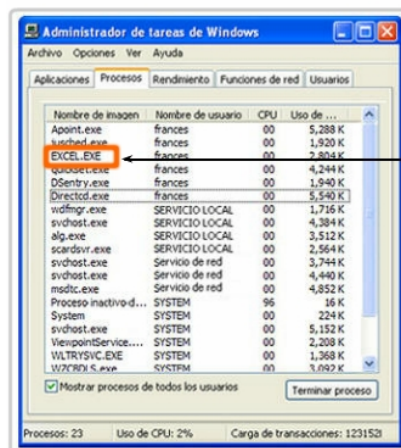
Las funciones asociadas con los protocolos de la capa de aplicación permiten a la red humana comunicarse con la red de datos subyacente. Cuando abrimos un explorador Web o una ventana de mensajería instantánea se inicia una aplicación, y el programa se coloca en la memoria del dispositivo donde se ejecuta. Cada programa ejecutable cargado a un dispositivo se denomina proceso.

Dentro de la capa de aplicación, existen dos formas de procesos o programas de software que proporcionan acceso a la red: aplicaciones y servicios.

Aplicaciones reconocidas por la red

Las aplicaciones son los programas de software que utiliza la gente para comunicarse a través de la red.. Algunas aplicaciones de usuario final son reconocidas por la red, lo cual significa que implementan los protocolos de la capa de aplicación y pueden comunicarse directamente con las capas inferiores del stack de protocolos. Los clientes de correo electrónico y los exploradores Web son ejemplos de este tipo de aplicaciones.

Procesos de software



Ejemplos de procesos en ejecución en el sistema operativo Windows

Los procesos son programas de software individuales que se ejecutan en forma simultánea.

Los procesos pueden ser

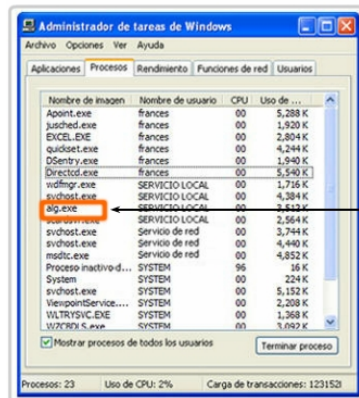
- 1 Aplicaciones
- 2 Servicios
- 3 Operaciones del sistema
- 4 Un programa puede estar en ejecución varias veces, cada vez dentro de su propio proceso.

Servicios de la capa de aplicación

Otros programas pueden necesitar la ayuda de los servicios de la capa de aplicación para utilizar los recursos de la red, como transferencia de archivos o cola de impresión en la red. Aunque son transparentes para el usuario, estos servicios son los programas que se comunican con la red y preparan los datos para la transferencia. Diferentes tipos de datos, ya sea texto, gráfico o video, requieren de diversos servicios de red para asegurarse de que estén bien preparados para procesar las funciones de las capas inferiores del modelo OSI.

Cada servicio de red o aplicación utiliza protocolos que definen los estándares y formatos de datos a utilizarse. Sin protocolos, la red de datos no tendría una manera común de formatear y direccionar los datos. Es necesario familiarizarse con los protocolos subyacentes que rigen la operación de los diferentes servicios de red para entender su función.

Procesos de software



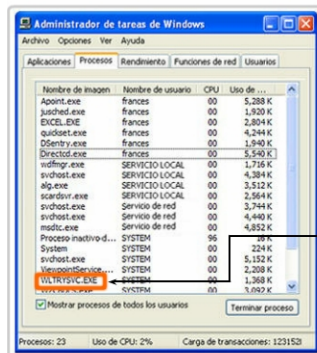
Los procesos son programas de software individuales que se ejecutan en forma simultánea.

Los procesos pueden ser

- 1 Aplicaciones
- 2 Servicios
- 3 Operaciones del sistema
- 4 Un programa puede estar en ejecución varias veces, cada vez dentro de su propio proceso.

Ejemplos de procesos en ejecución en el sistema operativo Windows

Procesos de software



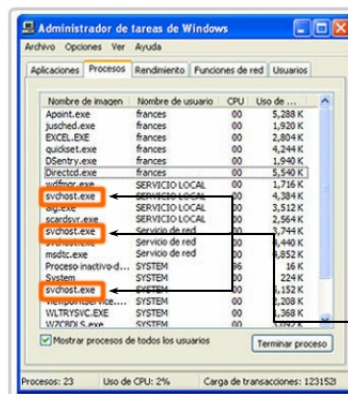
Los procesos son programas de software individuales que se ejecutan en forma simultánea.

Los procesos pueden ser

- 1 Aplicaciones
- 2 Servicios
- 3 Operaciones del sistema
- 4 Un programa puede estar en ejecución varias veces, cada vez dentro de su propio proceso.

Ejemplos de procesos en ejecución en el sistema operativo Windows

Procesos de software



Los procesos son programas de software individuales que se ejecutan en forma simultánea.

Los procesos pueden ser

- 1 Aplicaciones
- 2 Servicios
- 3 Operaciones del sistema
- 4 Un programa puede estar en ejecución varias veces, cada vez dentro de su propio proceso.

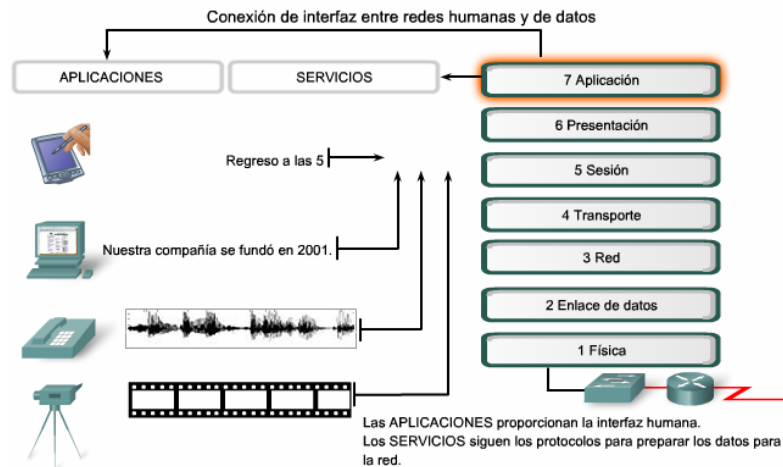
Ejemplos de procesos en ejecución en el sistema operativo Windows

APLICACIONES DEL USUARIO, SERVICIOS Y PROTOCOLOS DE LA DE APLICACIÓN.

Como se mencionó anteriormente, la capa de aplicación utiliza los protocolos implementados dentro de las aplicaciones y servicios. Mientras que las aplicaciones proporcionan a las personas una forma de crear mensajes y los servicios de la capa de aplicación establecen una interfaz con la red, los protocolos proporcionan las reglas y los formatos que regulan el trato de los datos. Un solo programa ejecutable debe utilizar los tres

componentes e inclusive el mismo nombre. Por ejemplo, al hablar de "Telnet" podemos estar refiriéndonos a la aplicación, al servicio o al protocolo.

En el modelo OSI, las aplicaciones que interactúan directamente con la gente se considera que están en la parte superior del stack, como la misma gente. Al igual que todas las personas dentro del modelo OSI, la capa de aplicación se basa en las funciones de las capas inferiores para completar el proceso de comunicación. Dentro de la capa de aplicación, los protocolos especifican qué mensajes se intercambian entre los hosts de origen y de destino, la sintaxis de los comandos de control, el tipo y el formato de los datos que se transmiten y los métodos adecuados para notificación y recuperación de errores.



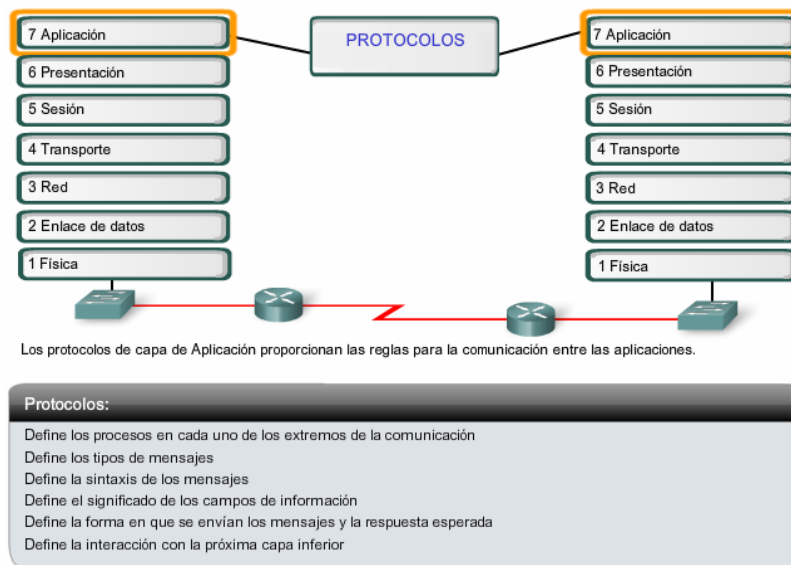
FUNCIONES DEL PROTOCOLO DE LA CAPA DE APLICACIÓN

Los protocolos de la capa de aplicación los utilizan tanto los dispositivos de origen como de destino durante una sesión de comunicación. Los protocolos de la capa de aplicación que se implementaron en los hosts de origen y destino deben coincidir para que las comunicaciones tengan éxito.

Los protocolos establecen reglas consistentes para el intercambio de datos entre aplicaciones y servicios cargados en los dispositivos participantes. Los protocolos especifican cómo se estructuran los datos dentro de los mensajes y los tipos de mensajes que se envían entre origen y destino. Estos mensajes pueden ser solicitudes de servicios, acuses de recibo, mensajes de datos, mensajes de estado o mensajes de error. Los protocolos también definen los diálogos de mensajes, asegurando que un mensaje enviado encuentre la respuesta esperada y se invoquen los servicios correspondientes cuando se realiza la transferencia de datos.

Muchos tipos de aplicaciones diferentes se comunican a través de las redes de datos. Por lo tanto, los servicios de la capa de aplicación deben implementar protocolos múltiples para proporcionar la variedad deseada de experiencias de comunicación. Cada protocolo tiene un fin específico y contiene las características requeridas para cumplir con dicho propósito. Deben seguirse los detalles del protocolo correspondiente a cada capa, así las funciones en una capa se comunican correctamente con los servicios en la capa inferior.

Las aplicaciones y los servicios también pueden utilizar protocolos múltiples durante el curso de una comunicación simple. Un protocolo puede especificar cómo se establece la conexión de redes y otro describir el proceso para la transferencia de datos cuando el mensaje se pasa a la siguiente capa inferior.



TOMA DE MEDIDAS PARA LAS APLICACIONES Y SERVICIOS

EL MODELO CLIENTE - SERVIDOR

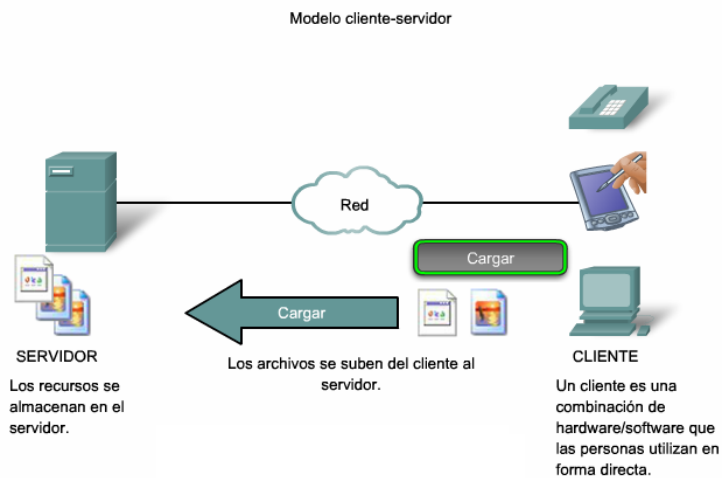
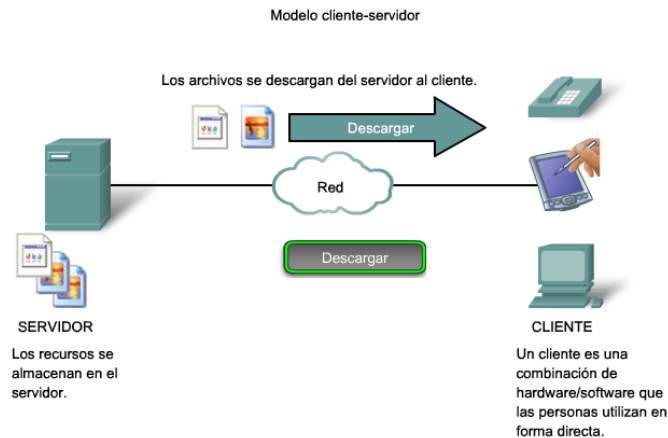
Cuando la gente intenta acceder a información en sus dispositivos, ya sean éstos una computadora personal o portátil, un PDA, un teléfono celular o cualquier otro dispositivo conectado a la red, los datos pueden no estar físicamente almacenados en sus dispositivos. Si así fuera, se debe solicitar permiso al dispositivo que contiene los datos para acceder a esa información.

El modelo cliente-servidor

En el modelo cliente/servidor, el dispositivo que solicita información se denomina cliente y el dispositivo que responde a la solicitud se denomina servidor. Los procesos de cliente y servidor se consideran una parte de la capa de aplicación. El cliente comienza el intercambio solicitando los datos al servidor, quien responde enviando uno o más streams de datos al cliente. Los protocolos de la capa de aplicación describen el formato de las solicitudes y respuestas entre clientes y servidores. Además de la transferencia real de datos, este intercambio puede requerir de información adicional, como la autenticación del usuario y la identificación de un archivo de datos a transferir.

Un ejemplo de una red cliente-servidor es un entorno corporativo donde los empleados utilizan un servidor de correo electrónico de la empresa para enviar, recibir y almacenar correos electrónicos. El cliente de correo electrónico en la computadora de un empleado emite una solicitud al servidor de correo electrónico para un mensaje no leído. El servidor responde enviando al cliente el correo electrónico solicitado.

Aunque los datos se describen generalmente como el flujo del servidor al cliente, algunos datos fluyen siempre del cliente al servidor. El flujo de datos puede ser el mismo en ambas direcciones, o inclusive puede ser mayor en la dirección que va del cliente al servidor. Por ejemplo, un cliente puede transferir un archivo al servidor con fines de almacenamiento. **La transferencia de datos de un cliente a un servidor se denomina cargar y de datos de un servidor a un cliente se conoce como descarga.**



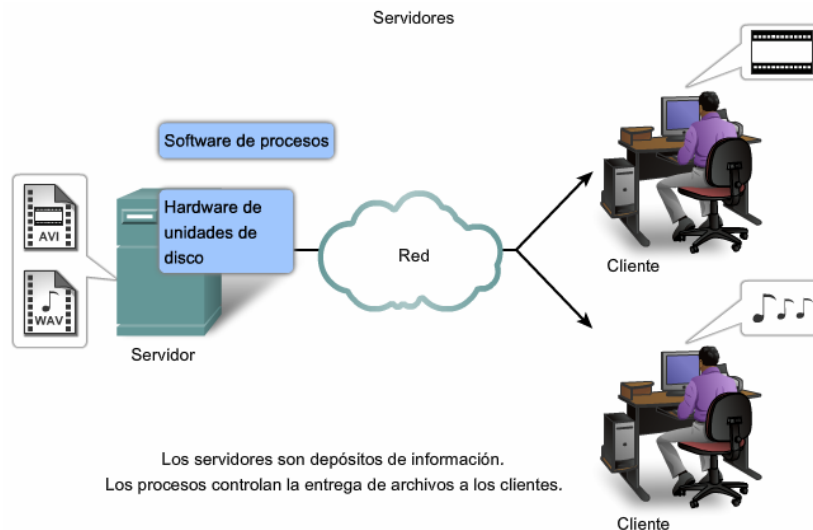
SERVIDORES

En un contexto general de redes, cualquier dispositivo que responde a una solicitud de aplicaciones de cliente funciona como un servidor. Un servidor generalmente es una computadora que contiene información para ser compartida con muchos sistemas de cliente. Por ejemplo, páginas Web, documentos, bases de datos, imágenes, archivos de audio y video pueden almacenarse en un servidor y enviarse a los clientes que lo solicitan. En otros casos, como una impresora de red, el servidor de impresión envía al cliente solicitudes para la impresora que se especifica.

Los diferentes tipos de aplicaciones de servidor pueden tener diferentes requisitos para el acceso del cliente. Algunos servidores pueden requerir de autenticación de la información de cuenta del usuario para verificar si el usuario tiene permiso para acceder a los datos solicitados o para utilizar una operación en particular. Dichos servidores deben contar con una lista central de cuentas de usuarios y autorizaciones, o permisos (para operaciones y acceso a datos) otorgados a cada usuario. Cuando se utiliza un cliente FTP, por ejemplo, si usted pide cargar datos al servidor FTP, se le puede dar permiso para escribir en su carpeta personal, pero no para leer otros archivos del sitio.

En una red cliente-servidor, el servidor ejecuta un servicio o proceso, a veces denominado daemon. Al igual que la mayoría de los servicios, los demonios generalmente se ejecutan en segundo plano y no se encuentran bajo control directo del usuario. Los demonios se describen como servidores que "escuchan" una solicitud del cliente porque están programados para responder cada vez que el servidor recibe una solicitud para el servicio

proporcionado por el demonio. Cuando un demonio "escucha" la solicitud de un cliente, intercambia los mensajes adecuados con el cliente, según lo requerido por su protocolo, y procede a enviar los datos solicitados en el formato correspondiente.

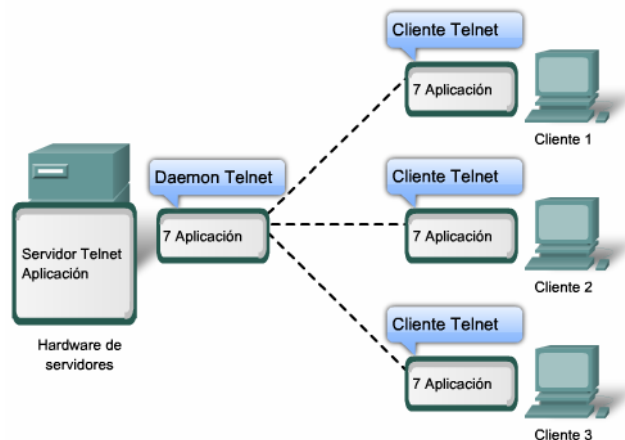


SERVICIOS Y PROTOCOLOS DE LA CAPA DE APLICACIÓN

Una sola aplicación puede emplear diferentes servicios de la capa de aplicación, así lo que aparece para el usuario como una solicitud para una página Web puede, de hecho, equivaler a docenas de solicitudes individuales. Y, para cada solicitud, pueden ejecutarse múltiples procesos. Por ejemplo, un cliente puede necesitar de diversos procesos individuales para formular sólo una solicitud al servidor.

Además, los servidores generalmente tienen múltiples clientes que solicitan información al mismo tiempo. Por ejemplo, un servidor Telnet puede tener varios clientes que requieren conectarse a él. Estas solicitudes individuales del cliente pueden manejarse en forma simultánea y separada para que la red sea exitosa. Los servicios y procesos de la capa de aplicación dependen del soporte de las funciones de la capa inferior para administrar en forma exitosa las múltiples conversaciones.

Los procesos de servidores pueden admitir múltiples clientes.



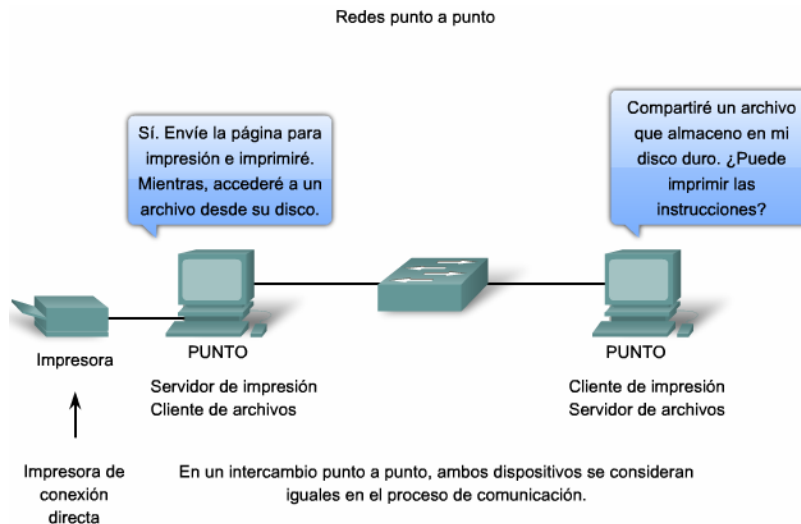
REDES Y APLICACIONES PUNTO A PUNTO (P2P)

El modelo punto a punto: Además del modelo cliente-servidor para networking, existe también un modelo punto a punto. Las redes punto a punto tienen dos formas distintivas: **diseño de redes punto a punto** y **aplicaciones punto a punto (P2P)**. Ambas formas tienen características similares, pero en la práctica son muy diferentes.

Redes punto a punto: En una red punto a punto, dos o más computadoras están conectadas por medio de una red y pueden compartir recursos (como impresoras y archivos) sin tener un servidor dedicado. Cada dispositivo final conectado (conocido como punto) puede funcionar como un servidor o como un cliente. Una computadora puede asumir la función de servidor para una transacción mientras funciona en forma simultánea como cliente para otra transacción. Las funciones de cliente y servidor se establecen por solicitud.

Una red doméstica sencilla con dos computadoras conectadas compartiendo una impresora es un ejemplo de una red punto a punto. Cada persona puede configurar su computadora para compartir archivos, habilitar juegos en red o compartir una conexión de Internet. Otro ejemplo sobre la funcionalidad de la red punto a punto son dos computadoras conectadas a una gran red que utilizan aplicaciones de software para compartir recursos entre ellas a través de la red.

A diferencia del modelo cliente-servidor, que utiliza servidores dedicados, las redes punto a punto descentralizan los recursos en una red. En lugar de ubicar información para compartir en los servidores dedicados, la información puede colocarse en cualquier parte de un dispositivo conectado. La mayoría de los sistemas operativos actuales admiten compartir archivos e impresoras sin requerir software del servidor adicional. Debido a que las redes punto a punto generalmente no utilizan cuentas de usuarios centralizadas, permisos ni monitores, es difícil implementar las políticas de acceso y seguridad en las redes que contienen mayor cantidad de computadoras. Se deben establecer cuentas de usuario y derechos de acceso en forma individual para cada dispositivo.

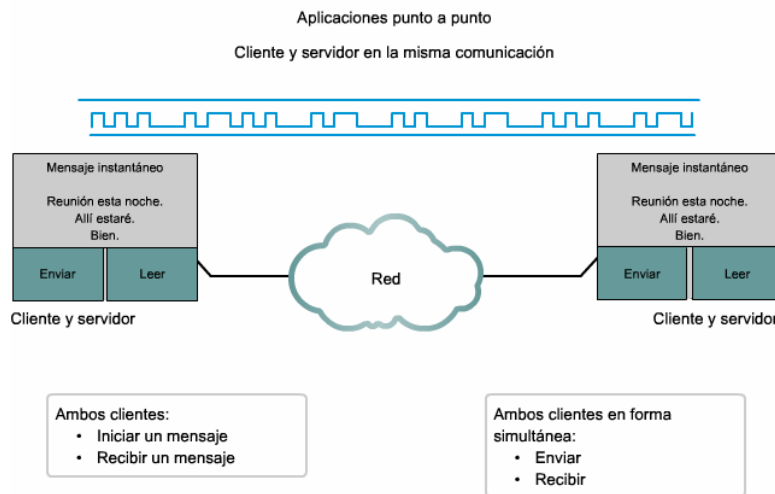


Aplicaciones punto a punto: Una aplicación punto a punto (P2P), a diferencia de una red punto a punto, permite a un dispositivo actuar como cliente o como servidor dentro de la misma comunicación. En este modelo, cada cliente es un servidor y cada servidor es un cliente. Ambos pueden iniciar una comunicación y se consideran iguales en el proceso de comunicación. Sin embargo, las aplicaciones punto a punto requieren que cada dispositivo final proporcione una interfaz de usuario y ejecute un servicio en segundo plano. Cuando inicia una aplicación punto a punto específica, ésta invoca la interfaz de usuario requerida y

los servicios en segundo plano. Después de eso, los dispositivos se pueden comunicar directamente.

Algunas aplicaciones P2P utilizan un sistema híbrido donde se descentraliza el intercambio de recursos, pero los índices que apuntan a las ubicaciones de los recursos están almacenados en un directorio centralizado. En un sistema híbrido, cada punto accede a un servidor de índice para alcanzar la ubicación de un recurso almacenado en otro punto. El servidor de índice también puede ayudar a conectar dos puntos, pero una vez conectados, la comunicación se lleva a cabo entre los dos puntos sin comunicación adicional al servidor de índice.

Las aplicaciones punto a punto pueden utilizarse en las redes punto a punto, en redes cliente-servidor y en Internet.



EJEMPLOS DE SERVICIOS Y PROTOCOLOS DE LA CAPA DE APLICACIÓN

PROTOCOLOS Y SERVICIOS DE DNS

Ahora que tenemos una mejor comprensión de cómo las aplicaciones proporcionan una interfaz para el usuario y acceso a la red, veremos algunos protocolos específicos utilizados comúnmente.

Como vimos, la capa de transporte utiliza un esquema de direccionamiento llamado número de puerto. Los números de puerto identifican las aplicaciones y los servicios de la capa de aplicación que son el origen y el destino de los datos. Los programas del servidor generalmente utilizan números de puerto predefinidos comúnmente conocidos por los clientes. Mientras examinamos los diferentes servicios y protocolos de la capa de aplicación de TCP/IP, nos referiremos a los números de puerto TCP y UDP normalmente asociados con estos servicios. Algunos de estos servicios son:

- **Sistema de nombres de dominios (DNS)** - TCP/UDP puerto 53
- **Protocolo de transferencia de hipertexto (HTTP)** - TCP puerto 80
- **Protocolo simple de transferencia de correo (SMTP)** - TCP puerto 25
- **Protocolo de oficina de correos (POP)** - TCP puerto 110
- **Telnet** - TCP puerto 23
- **Protocolo de configuración dinámica de host (DHCP)** - UDP puertos 67 y 68
- **Protocolo de transferencia de archivos (FTP)** - TCP puertos 20 y 21

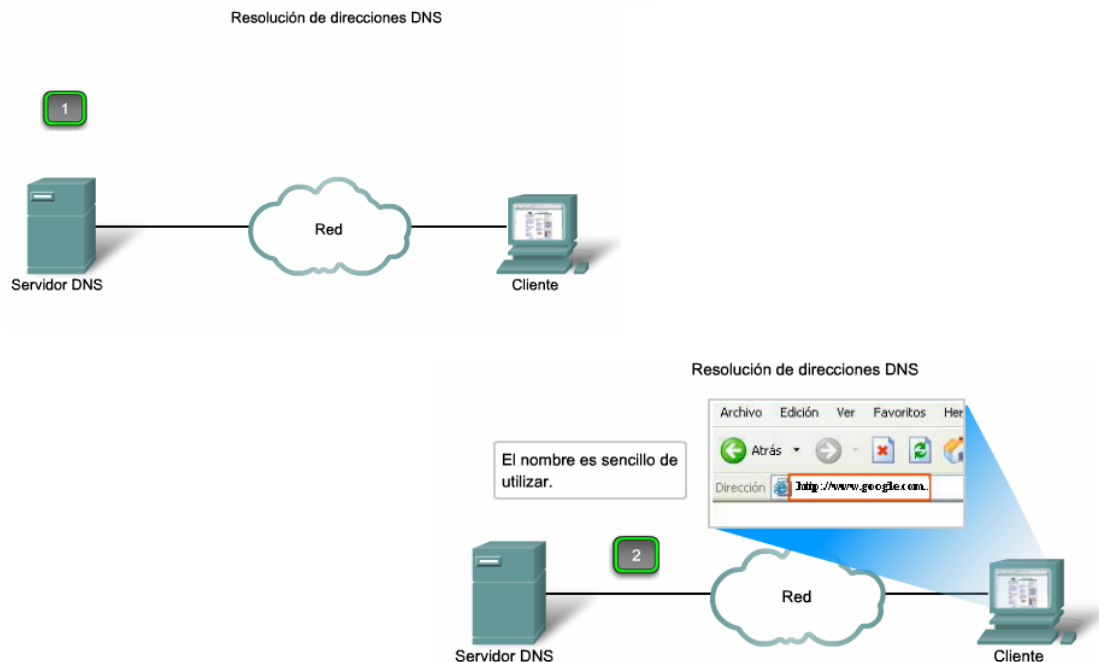
DNS

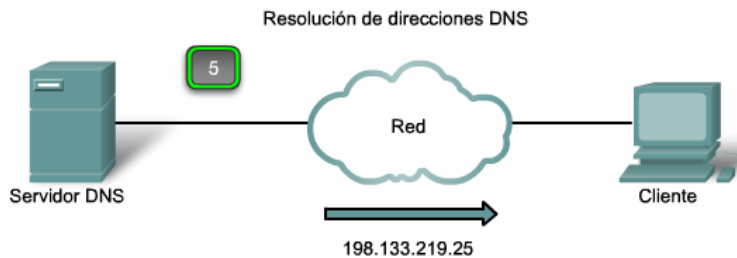
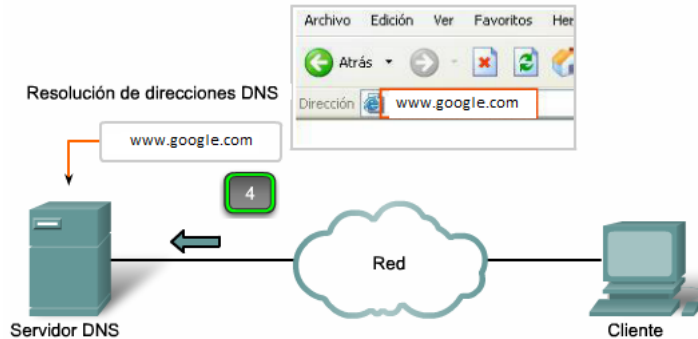
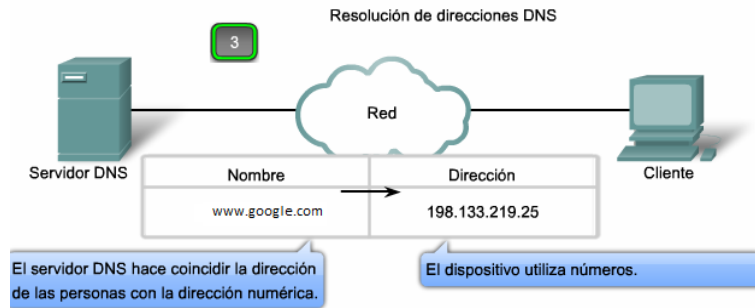
En las redes de datos, los dispositivos se etiquetan con una dirección IP numérica, de manera que pueden participar en el envío y la recepción de mensajes de la red. Sin embargo, la mayoría de las personas pasan mucho tiempo tratando de recordar estas direcciones numéricas. Por lo tanto, los nombres de dominios se crearon para convertir las direcciones numéricas en un nombre sencillo y reconocible.

En Internet, estos nombres de dominio, tales como www.google.com, son mucho más fáciles de recordar para la gente que algo como 198.133.219.25, el cual es la dirección numérica actual para ese servidor. Además, si google decide cambiar la dirección numérica, es transparente para el usuario, ya que el nombre de dominio seguirá siendo www.google.com. La nueva dirección simplemente estará enlazada con el nombre de dominio existente y la conectividad se mantendrá. Cuando las redes eran pequeñas, resultaba fácil mantener la asignación entre los nombres de dominios y las direcciones que representaban. Sin embargo, a medida que las redes y el número de dispositivos comenzaron a crecer, el sistema manual dejó de ser práctico.

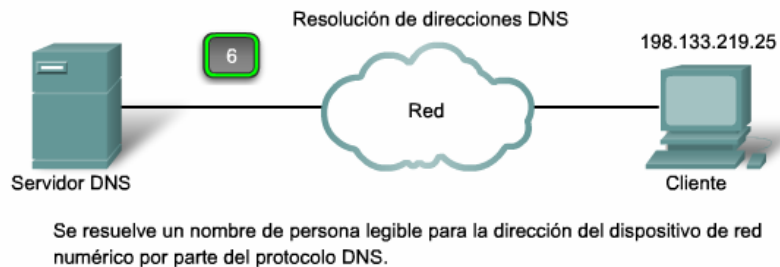
El **Sistema de nombres de dominios (DNS)** se creó para que el nombre del dominio busque soluciones para estas redes. DNS utiliza un conjunto distribuido de servidores para resolver los nombres asociados con estas direcciones numéricas.

El protocolo DNS define un servicio automatizado que coincide con nombres de recursos que tienen la dirección de red numérica solicitada. Incluye las consultas sobre formato, las respuestas y los formatos de datos. Las comunicaciones del protocolo DNS utilizan un formato simple llamado mensaje. Este formato de mensaje se utiliza para todos los tipos de solicitudes de clientes y respuestas del servidor, mensajes de error y para la transferencia de información de registro de recursos entre servidores.





El número se envía de regreso al cliente para su utilización en la realización de solicitudes del servidor.



DNS es un servicio cliente-servidor; sin embargo, difiere de los otros servicios cliente-servidor que estamos examinando. Mientras otros servicios utilizan un cliente que es una aplicación (como un explorador Web o un cliente de correo electrónico), el cliente DNS ejecuta un servicio por sí mismo. El cliente DNS, a veces denominado **resolución DNS**, admite la resolución de nombres para otras aplicaciones de red y servicios que lo necesiten.

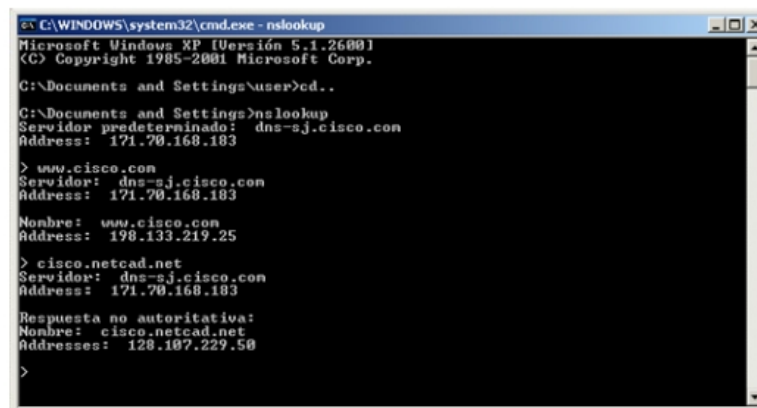
Al configurar un dispositivo de red, generalmente proporcionamos una o más direcciones del servidor DNS que el cliente DNS puede utilizar para la resolución de nombres. En general, el

proveedor de servicios de Internet provee las direcciones para utilizar con los servidores DNS. Cuando la aplicación del usuario pide conectarse a un dispositivo remoto por nombre, el cliente DNS solicitante consulta uno de estos servidores de denominación para resolver el nombre para una dirección numérica.

Los sistemas operativos computacionales también cuentan con una herramienta llamada nslookup que permite que el usuario consulte de forma manual los servidores de nombres para resolver un nombre de host dado. Esta utilidad también puede utilizarse para solucionar los problemas de resolución de nombres y verificar el estado actual de los servidores de nombres.

En la figura, cuando se ejecuta nslookup, se muestra el servidor DNS predeterminado configurado para su host. En este ejemplo, el servidor DNS es dns-sjk.cisco.com que tiene una dirección de 171.68.226.120.

Uso de nslookup



```
C:\WINDOWS\system32\cmd.exe - nslookup
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\user>cd..
C:\Documents and Settings>nslookup
Servidor predeterminado: dns-sjk.cisco.com
Address: 171.70.168.183

> www.cisco.com
Servidor: dns-sjk.cisco.com
Address: 171.70.168.183

Nombre: www.cisco.com
Address: 198.133.219.25

> cisco.netcad.net
Servidor: dns-sjk.cisco.com
Address: 171.70.168.183

Respuesta no autoritativa:
Nombre: cisco.netcad.net
Addresses: 128.107.229.50

>
```

Luego podemos escribir el nombre de un host o dominio para el cual deseamos obtener la dirección. En la primer consulta de la figura, se hace una consulta para www.cisco.com. El servidor de nombre que responde proporciona la dirección 198.133.219.25.

La consulta mostrada en la figura son sólo pruebas simples. La herramienta nslookup tiene muchas opciones disponibles para lograr una extensa verificación y prueba del proceso DNS.

Un servidor DNS proporciona la resolución de nombres utilizando el demonio de nombres que generalmente se llama named (se pronuncia name-dee).

El servidor DNS almacena diferentes tipos de registros de recursos utilizados para resolver nombres. Estos registros contienen el nombre, la dirección y el tipo de registro.

Algunos de estos tipos de registros son:

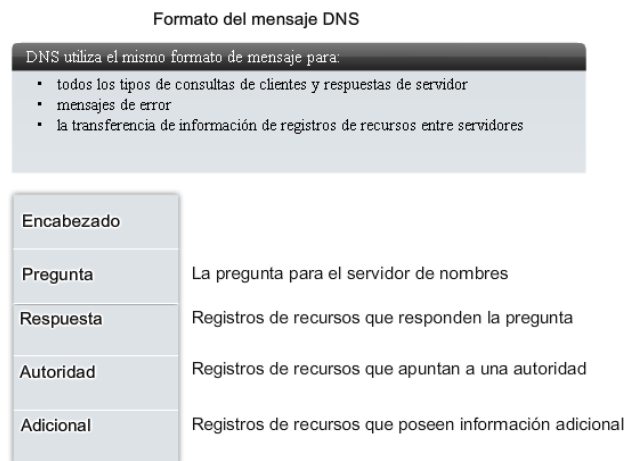
- **A:** una dirección de dispositivo final
- **NS:** un servidor de nombre autoritativo
- **CNAME:** el nombre canónico (o Nombre de dominio completamente calificado) para un alias que se utiliza cuando varios servicios tienen una dirección de red única, pero cada servicio tiene su propia entrada en el DNS

- **MX:** registro de intercambio de correos; asigna un nombre de dominio a una lista de servidores de intercambio de correos para ese dominio

Cuando un cliente hace una consulta, el proceso "nombrado" del servidor busca primero en sus propios registros para ver si puede resolver el nombre. Si no puede resolverlo con sus registros almacenados, contacta a otros servidores para hacerlo.

La solicitud puede pasar a lo largo de cierta cantidad de servidores, lo cual puede tomar más tiempo y consumir banda ancha. Una vez que se encuentra una coincidencia y se devuelve al servidor solicitante original, el servidor almacena temporalmente en la caché la dirección numerada que coincide con el nombre.

Si vuelve a solicitarse ese mismo nombre, el primer servidor puede regresar la dirección utilizando el valor almacenado en el caché de nombres. El almacenamiento en caché reduce el tráfico de la red de datos de consultas DNS y las cargas de trabajo de los servidores más altos de la jerarquía. El servicio del cliente DNS en las PC de Windows optimiza el rendimiento de la resolución de nombres DNS almacenando previamente los nombres resueltos en la memoria. El comando `ipconfig /displaydns` muestra todas las entradas DNS en caché en un sistema informático con Windows XP o 2000.



El sistema de nombres de dominios utiliza un sistema jerárquico para crear una base de datos y así proporcionar una resolución de nombres. La jerarquía es similar a un árbol invertido con la raíz en la parte superior y las ramas por debajo.

En la parte superior de la jerarquía, los servidores raíz mantienen registros sobre cómo alcanzar los servidores de dominio de nivel superior, los cuales a su vez tienen registros que apuntan a los servidores de dominio de nivel secundario y así sucesivamente.

Los diferentes dominios de primer nivel representan el tipo de organización o el país de origen. Entre los ejemplos de dominios del nivel superior se encuentran:

- .au:** Australia
- .co:** Colombia
- .com:** una empresa o industria
- .jp:** Japón
- .org:** una organización sin fines de lucro

Después de los dominios del nivel superior, se encuentran los nombres de los dominios de segundo nivel y debajo de estos hay otros dominios de nivel inferior.

Cada nombre de dominio es una ruta hacia este árbol invertido que comienza de la raíz.

Por ejemplo, como se muestra en la figura, el servidor DNS raíz puede no saber exactamente dónde se ubica el servidor de correo electrónico mail.cisco.com, pero conserva un registro para el dominio "com" dentro del dominio de nivel superior. Asimismo, los servidores dentro del dominio "com" pueden no tener un registro de mail.cisco.com, pero sí tienen un registro para el dominio "cisco.com". Los servidores dentro del dominio cisco.com tienen un registro (un registro MX para ser precisos) para mail.cisco.com.

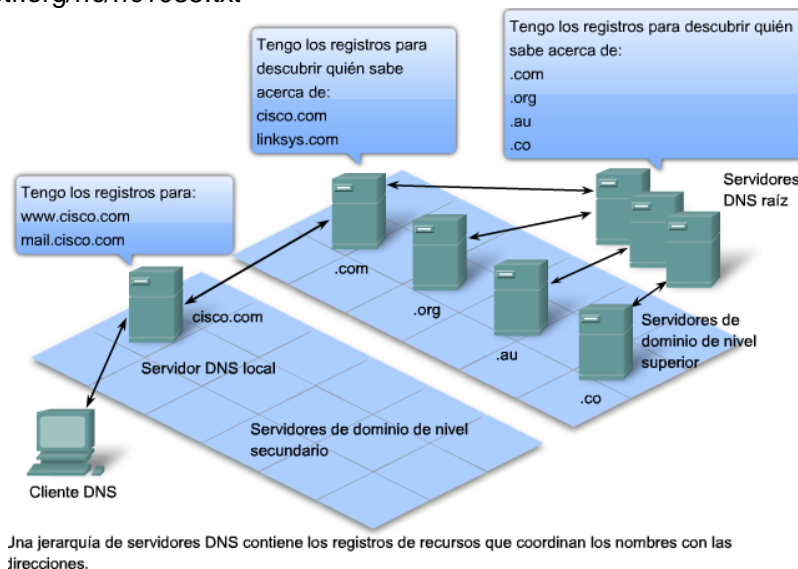
El DNS depende de esta jerarquía de servidores descentralizados para almacenar y mantener estos registros de recursos. Los registros de recursos enumeran nombres de dominios que el servidor puede resolver y servidores alternativos que también pueden procesar solicitudes. Si un servidor dado tiene registros de recursos que corresponden a su nivel en la jerarquía de dominios, se dice que es autoritativo para dichos registros.

Por ejemplo, un servidor de nombres en el dominio cisco.netacad.net no sería autoritativo para el registro mail.cisco.com porque dicho registro se mantiene en un servidor de nivel de dominio superior, específicamente el servidor de nombres en el dominio cisco.com.

Enlaces

<http://www.ietf.org/rfc/rfc1034.txt>

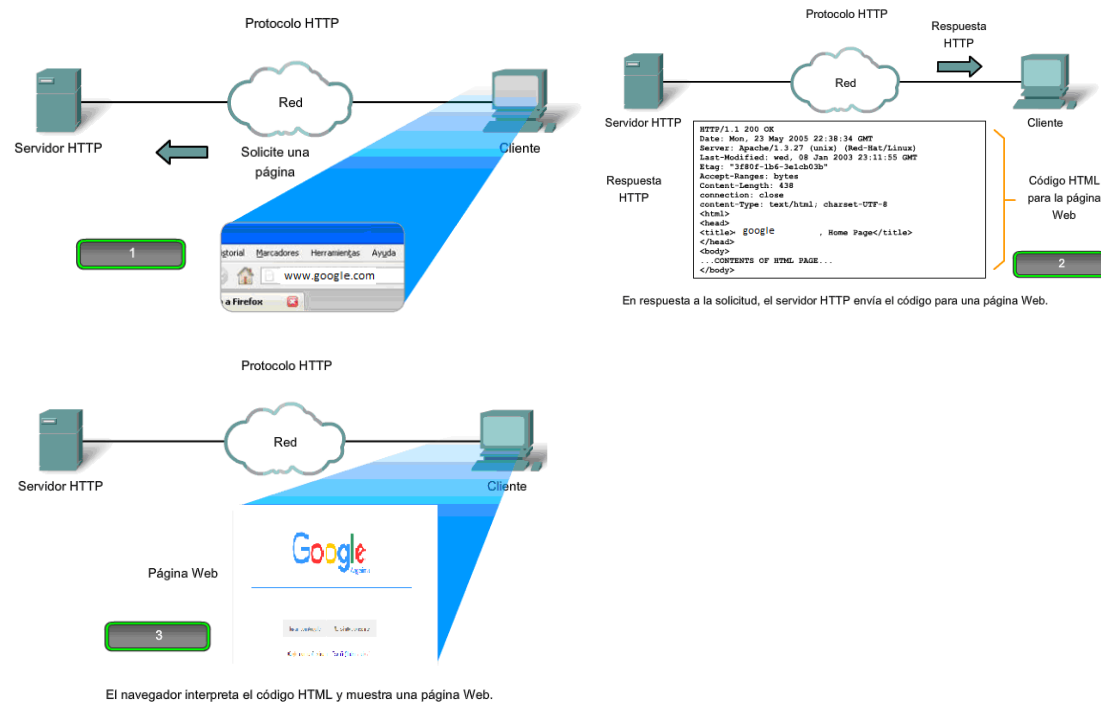
<http://www.ietf.org/rfc/rfc1035.txt>



SERVICIO WWW y http

Cuando se escribe una dirección Web (o URL) en un explorador de Internet, el explorador establece una conexión con el servicio Web del servidor que utiliza el protocolo HTTP. URL (o Localizador uniforme de recursos) y URI (Identificador uniforme de recursos) son los nombres que la mayoría de las personas asocian con las direcciones Web.

El URL `http://www.google.com/index.html` es un ejemplo que se refiere a un recurso específico, una página Web llamada `index.html` en un servidor identificado como `google.com` (en la figura se visualiza los pasos que utiliza el HTTP).



Los exploradores Web son las aplicaciones cliente que utilizan nuestras computadoras para conectarse a la World Wide Web y acceder a recursos almacenados en un servidor Web. Al igual que con la mayoría de los procesos de servidores, el servidor Web funciona como un servicio básico y genera diferentes tipos de archivos disponibles.

Para acceder al contenido, los clientes Web realizan conexiones al servidor y solicitan los recursos deseados. El servidor responde con el recurso y, al recibirlo, el explorador interpreta los datos y los presenta al usuario.

Los buscadores pueden interpretar y presentar muchos tipos de datos, como texto sin cifrar o Lenguaje de marcas de hipertexto (HTML, el lenguaje en el que se crean las páginas Web). Otros tipos de datos, sin embargo, requieren de otro servicio o programa. Generalmente se les conoce como plug-ins o complementos. Para ayudar al explorador a determinar qué tipo de archivo está recibiendo, el servidor especifica qué clase de datos contiene el archivo.

Para comprender mejor cómo interactúan el explorador Web y el cliente Web, podemos analizar cómo se abre una página Web en un explorador. Para este ejemplo, utilizaremos la dirección URL: <http://www.google.com/web-server.htm>.

Primero, el explorador interpreta las tres partes del URL:

1. http (el protocolo o esquema)
2. www.google.com (el nombre del servidor)
3. web-server.htm (el nombre de archivo específico solicitado).

Después, el explorador verifica con un servidor de nombres para convertir a www.google.com en una dirección numérica que utilizará para conectarse con el servidor. Al utilizar los requerimientos del protocolo HTTP, el explorador envía una solicitud GET al servidor y pide el archivo web-server.htm. El servidor, a su vez, envía al explorador el código

HTML de esta página Web. Finalmente, el explorador descifra el código HTML y da formato a la página para la ventana del explorador.

El protocolo de transferencia de hipertexto (HTTP), uno de los protocolos del grupo TCP/IP, se desarrolló en sus comienzos para publicar y recuperar las páginas HTML, y en la actualidad se utiliza para sistemas de información distribuidos y de colaboración. HTTP se utiliza a través de la World Wide Web para transferencia de datos y es uno de los protocolos de aplicación más utilizados.

HTTP especifica un protocolo de solicitud/respuesta. Cuando un cliente, generalmente un explorador Web, envía un mensaje de solicitud a un servidor, el protocolo HTTP define los tipos de mensajes que el cliente utiliza para solicitar la página Web y envía los tipos de mensajes que el servidor utiliza para responder. Los tres tipos de mensajes comunes son GET, POST y PUT.

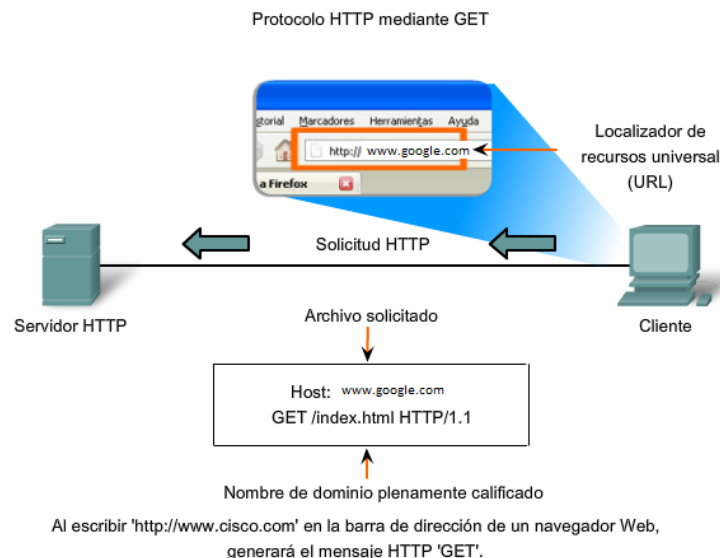
GET es una solicitud de datos por parte del cliente. Un explorador Web envía el mensaje GET para solicitar las páginas desde un servidor Web. Como se muestra en la figura, una vez que el servidor recibe la solicitud GET, responde con una línea de estado, como HTTP/1.1 200 OK, y un mensaje propio, el cuerpo del cual puede ser el archivo solicitado, un mensaje de error u otra información.

POST y PUT se utilizan para enviar mensajes que cargan datos en el servidor Web. Por ejemplo, cuando el usuario ingresa información en un formato incluido en una página Web, POST incluye la información en el mensaje enviado al servidor.

PUT carga los recursos o el contenido en el servidor Web.

Aunque es muy flexible, HTTP no es un protocolo seguro. Los mensajes POST cargan información al servidor en un texto sin formato que se puede interceptar y leer. De forma similar, las respuestas del servidor, generalmente páginas HTML, también se descifran.

Para una comunicación segura a través de Internet, se utiliza el protocolo HTTP seguro (HTTPS) para acceder o subir información al servidor Web. HTTPS puede utilizar autenticación y encriptación para asegurar los datos cuando viajan entre el cliente y el servidor. HTTPS especifica reglas adicionales para pasar los datos entre la capa de aplicación y la capa de transporte.

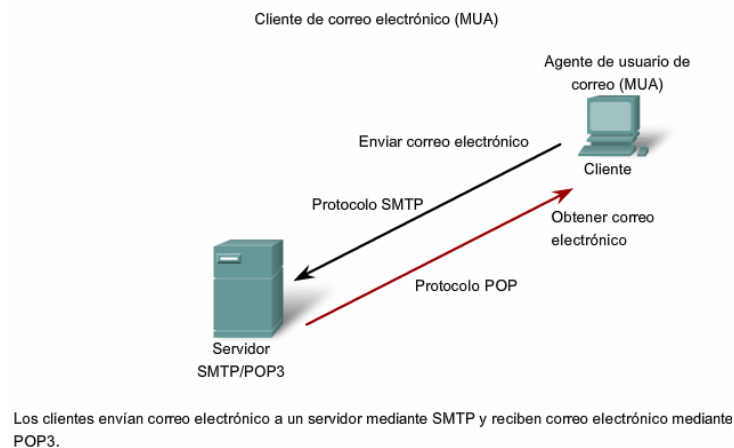


SERVICIO DE CORREO ELECTRÓNICO Y PROTOCOLOS SMTP/POP

Correo electrónico, el servidor de red más conocido, ha revolucionado la manera en que nos comunicamos, por su simpleza y velocidad. Incluso para ejecutarse en una computadora o en otro dispositivo, los correos electrónicos requieren de diversos servicios y aplicaciones. Dos ejemplos de protocolos de capa de aplicación son el Protocolo de oficina de correos (POP) y el Protocolo simple de transferencia de correo (SMTP), que aparecen en la figura. Como con el HTTP, estos protocolos definen los procesos de cliente-servidor.

Cuando la gente redacta mensajes de correo electrónico, generalmente utilizan una aplicación llamada Agente de usuario de correo (MUA), o un cliente de correo electrónico. MUA permite enviar los mensajes y colocar los recibidos en el buzón del cliente; ambos procesos son diferentes.

Para recibir correos electrónicos desde un servidor de correo, el cliente de correo electrónico puede utilizar un POP. Al enviar un correo electrónico desde un cliente o un servidor se utilizan formatos de mensajes y cadenas de comando definidas por el protocolo SMTP. En general, un cliente de correo electrónico proporciona la funcionalidad de ambos protocolos dentro de una aplicación.

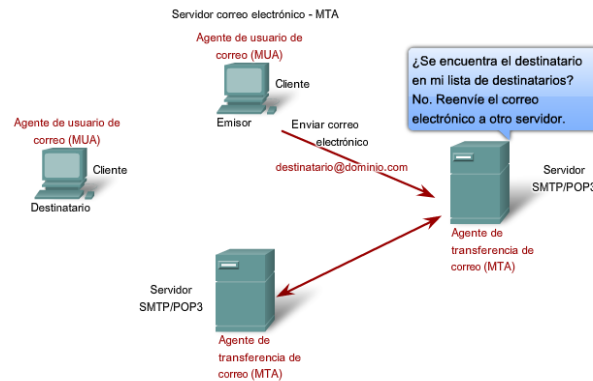


Procesos del servidor de correo electrónico: **MTA** y **MDA**

El servidor de correo electrónico utiliza dos procesos independientes:

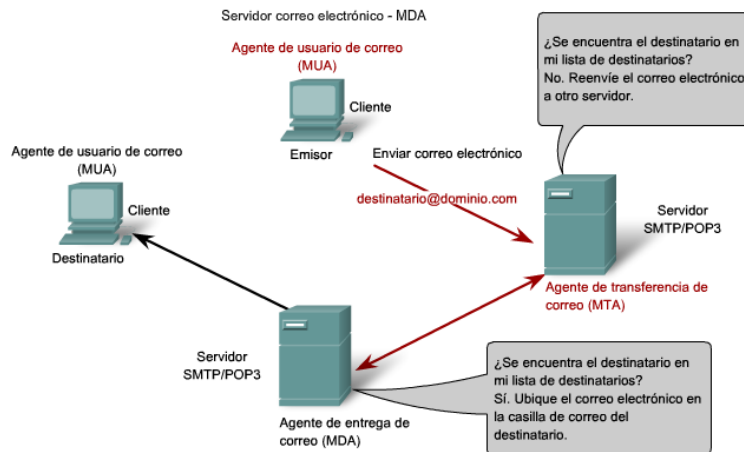
- Agente de transferencia de correo (**MTA**)
- Agente de entrega de correo (**MDA**)

El proceso Agente de transferencia de correo (MTA) se utiliza para enviar correo electrónico. Como se muestra en la figura, el MTA recibe mensajes desde el MUA u otro MTA en otro servidor de correo electrónico. Según el encabezado del mensaje, determina cómo debe reenviarse un mensaje para llegar al destino. Si el correo está dirigido a un usuario cuyo buzón está en el servidor local, el correo se pasa al MDA. Si el correo es para un usuario que no está en el servidor local, el MTA enruta el correo electrónico al MTA en el servidor correspondiente.



El proceso de agente de transferencia de correo riga el manejo de correo electrónico entre servidores.

En la figura, vemos que el Agente de entrega de correo (MDA) acepta una parte del correo electrónico desde un Agente de transferencia de correo (MTA) y realiza el envío real. El MDA recibe todo el correo entrante desde el MTA y lo coloca en los buzones de los usuarios correspondientes. El MDA también puede resolver temas de entrega final, como análisis de virus, correo no deseado filtrado y manejo de acuses de recibo. La mayoría de las comunicaciones de correo electrónico utilizan las aplicaciones MUA, MTA y MDA. Sin embargo, hay otras alternativas para el envío de correo electrónico.

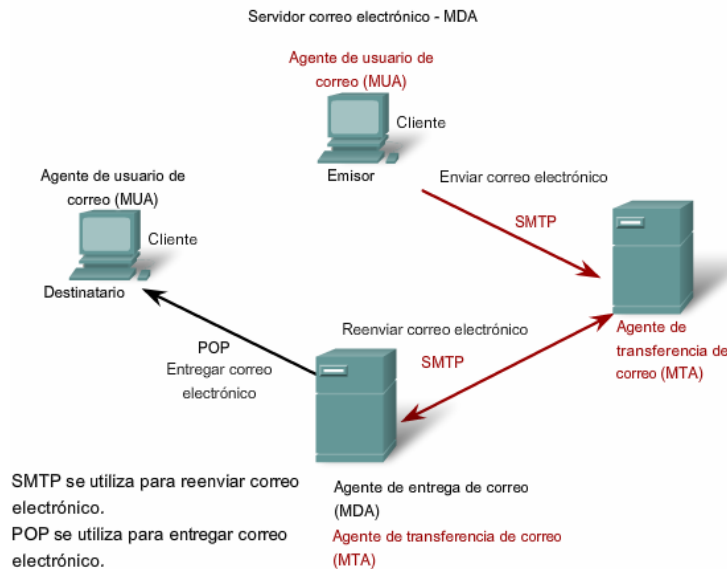


El proceso de agente de entrega de correo riga la entrega de correo electrónico entre servidores y clientes.

Un cliente puede estar conectado a un sistema de correo electrónico corporativo, como Lotus Notes de IBM, Groupwise de Novell o Exchange de Microsoft. Estos sistemas con frecuencia tienen su propio formato interno de correo electrónico, y sus clientes generalmente se comunican con el servidor de correo electrónico mediante un protocolo propietario. El servidor envía o recibe correo electrónico por medio de Internet a través del gateway de correo de Internet del producto, el cual realiza cualquier reformato necesario. Como segunda alternativa, las computadoras que no tienen un MUA pueden conectarse a un servicio de correo en un explorador Web para así recuperar y enviar mensajes. Algunas computadoras pueden ejecutar su propio MTA y administrar correos electrónicos de dominio interno. Si, por ejemplo, dos personas que trabajan para la misma empresa intercambian correos electrónicos entre ellos utilizando un protocolo propietario, los mensajes pueden permanecer completamente dentro del sistema de correos corporativo de la empresa.

Como se mencionó anteriormente, los correos electrónicos pueden utilizar los protocolos POP y SMTP (vea la figura para saber cómo funcionan). POP y POP3 (Protocolo de oficina

de correos v.3) son protocolos de envío de correo entrante y protocolos cliente-servidor típicos. Envían correos electrónicos desde el servidor correspondiente al cliente (MUA). El MDA escucha cuando un cliente se conecta a un servidor. Una vez establecida la conexión, el servidor puede enviar el correo electrónico al cliente.



El Protocolo simple de transferencia de correo (SMTP), por el contrario, rige la transferencia de correos salientes desde el cliente emisor al servidor de correos (MDA), así como también el transporte de correos entre servidores de correo electrónico (MTA). SMTP permite transportar correos por las redes de datos entre diferentes tipos de software de cliente y servidor, y hace posible el intercambio de correos en Internet.

El formato de mensajes del protocolo SMTP utiliza un conjunto rígido de comandos y respuestas. Estos comandos dan soporte a los procedimientos que se utilizan en el SMTP, como inicio de sesión, transacción de correo, envío de correo, verificación de nombres de buzones de correo, expansión de listas de correo e intercambios de apertura y cierre.

Algunos de los comandos que se especifican en el protocolo SMTP son:

HELO: identifica el proceso del cliente SMTP para el proceso del servidor SMTP

EHLO: es una nueva versión del HELO, que incluye extensiones de servicios

MAIL FROM: identifica el emisor

RCPT TO: identifica el receptor

DATA: identifica el cuerpo del mensaje

FTP (Protocolo de transferencia de archivos)

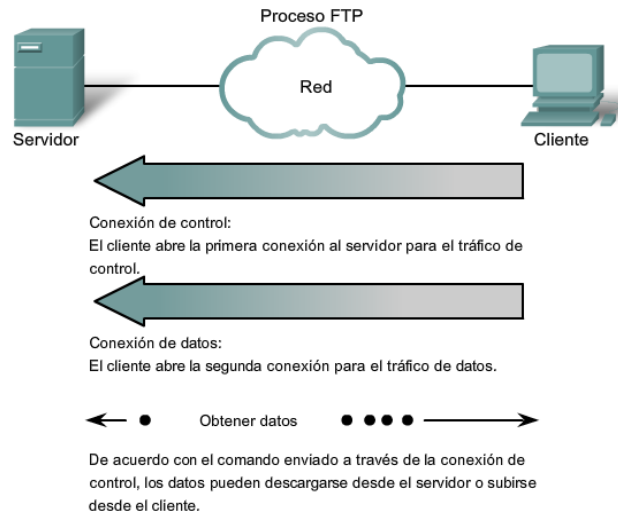
El **Protocolo de transferencia de archivos (FTP)** es otro protocolo de la capa de aplicación de uso común. El FTP se desarrolló para permitir las transferencias de archivos entre un cliente y un servidor. Un cliente FTP es una aplicación que se ejecuta en una computadora y que carga y descarga archivos de un servidor que ejecuta el demonio FTP (FTPd).

El FTP necesita dos conexiones entre el cliente y el servidor para transferir archivos de forma exitosa: una para comandos y respuestas, otra para la transferencia real de archivos.

El cliente establece la primera conexión con el servidor en TCP puerto 21. Esta conexión se utiliza para controlar el tráfico, que consiste en comandos del cliente y respuestas del servidor.

El cliente establece la segunda conexión con el servidor en TCP puerto 20. Esta conexión es para la transferencia real de archivos y se crea cada vez que se transfiere un archivo.

La transferencia de archivos puede producirse en ambas direcciones. El cliente puede descargar (bajar) un archivo desde el servidor o el cliente puede cargar (subir) un archivo en el servidor.



DHCP (Protocolo de configuración dinámica de host)

El servicio del **Protocolo de configuración dinámica de host (DHCP)** permite a los dispositivos de una red obtener direcciones IP y otra información de un servidor DHCP. Este servicio automatiza la asignación de direcciones IP, máscaras de subred, gateway y otros parámetros de networking del IP.

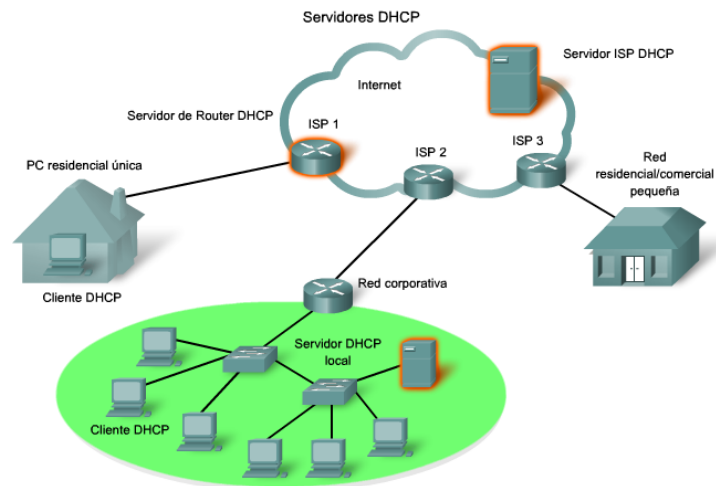
DHCP permite a un host obtener una dirección IP de forma dinámica cuando se conecta a la red. Se realiza el contacto con el servidor de DHCP y se solicita una dirección. El servidor DHCP elige una dirección del rango configurado llamado pool y la asigna ("alquila") para el host por un tiempo establecido.

En redes locales más grandes, o donde los usuarios cambien con frecuencia, se prefiere el DHCP. Los nuevos usuarios llegan con computadoras portátiles y necesitan una conexión. Otros tienen nuevas estaciones de trabajo que necesitan conexión. En lugar de que el administrador de red asigne direcciones IP para cada estación de trabajo, es más eficaz que las direcciones IP se asignen automáticamente mediante el DHCP.

Las direcciones distribuidas por DHCP no se asignan de forma permanente a los hosts, sino que sólo se alquilan por un periodo de tiempo. Si el host se apaga o se desconecta de la red, la dirección regresa al pool para volver a utilizarse. Esto es especialmente útil para los usuarios móviles que entran y salen de la red. Los usuarios pueden moverse libremente desde una ubicación a otra y volver a establecer las conexiones de red. El host puede obtener una dirección IP cuando se conecte el hardware, ya sea por cables o por LAN inalámbrica.

DHCP le permite el acceso a Internet por medio de Internet utilizando zonas de cobertura inalámbrica en aeropuertos o cafeterías. Una vez que ingresa al área, el cliente de DHCP de la computadora portátil contacta al servidor de DHCP mediante una conexión inalámbrica. El servidor de DHCP asigna una dirección IP a la computadora portátil.

Como lo muestra la figura, varios tipos de dispositivos pueden ser servidores de DHCP cuando ejecutan software de servicio de DHCP. En la mayoría de las redes medianas a grandes, el servidor de DHCP generalmente es un servidor local dedicado con base en una PC.



Con las redes domésticas, el servidor de DHCP se ubica en el ISP y un host de la red doméstica recibe la configuración IP directamente desde el ISP.

DHCP puede representar un riesgo a la seguridad porque cualquier dispositivo conectado a la red puede recibir una dirección. Este riesgo hace que la seguridad física sea un factor importante al determinar si se utiliza el direccionamiento dinámico o manual.

Ambos direccionamientos tienen su lugar en los diseños de red. Muchas redes utilizan tanto el direccionamiento estático como el DHCP. DHCP se utiliza para hosts de propósitos generales, como los dispositivos de usuario final, y las direcciones fijas se utilizan para dispositivos de red como gateways, switches, servidores e impresoras.

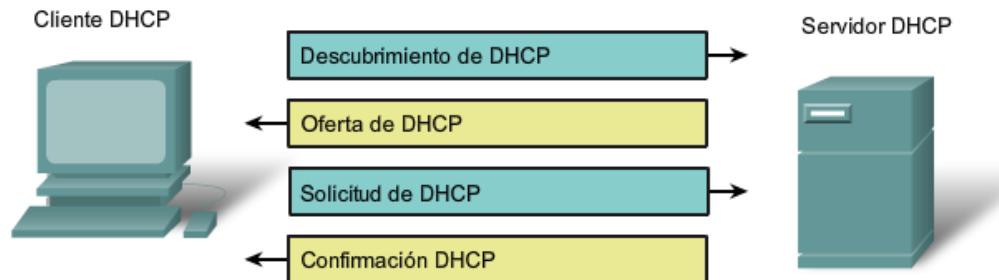
Sin DHCP los usuarios tienen que ingresar manualmente la dirección IP, la máscara de subred y otras configuraciones para poder unirse a la red. El servidor de DHCP mantiene un pool de las direcciones IP y alquila una dirección a cualquier cliente habilitado por DHCP cuando el cliente está activado. Debido a que las direcciones IP son dinámicas (alquiladas) en lugar de estáticas (asignadas en forma permanente), las direcciones en desuso regresan automáticamente al pool para volver a asignarse. Cuando un dispositivo configurado por DHCP se inicia o conecta a la red, el cliente envía un paquete **DESCUBRIMIENTO** de DHCP para identificar cualquier servidor de DHCP disponible en la red. Un servidor de DHCP responde con una **OFERTA DE DHCP**, la cual es un mensaje de oferta de alquiler con información asignada de dirección IP, máscara de subred, servidor DNS y gateway predeterminado, así como la duración del alquiler.

El cliente puede recibir múltiples paquetes de **OFERTA DE DHCP** si hay más de un servidor de DHCP en la red local, así que debe elegir entre ellos y enviar un paquete de **SOLICITUD DE DHCP** que identifique el servidor explícito y la oferta de alquiler que el cliente acepta. Un cliente puede elegir solicitar una dirección previamente asignada por el servidor.

Teniendo en cuenta que la dirección IP solicitada por el cliente, u ofrecida por el servidor, aún es válida, el servidor devolverá un mensaje ACK DHCP que le informa al cliente que finalizó el alquiler. Si la oferta ya no es válida, quizás debido al tiempo o que a otro cliente se le asignó el alquiler, el servidor seleccionado responderá con un mensaje NAK DHCP (acuse de recibo negativo). Si un mensaje NAK DHCP se devuelve, entonces el proceso de selección debe volver a comenzar con la transmisión de un mensaje nuevo de DESCUBRIMIENTO DE DHCP.

Una vez que el cliente tenga el alquiler, se debe renovar mediante otro mensaje de SOLICITUD DE DHCP, antes de que termine el alquiler.

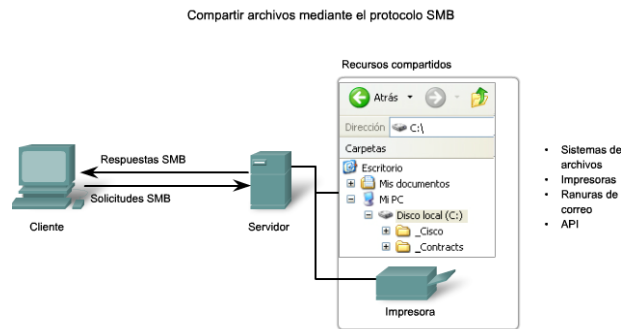
El servidor de DHCP asegura que las direcciones IP sean únicas (una dirección IP no se puede asignar a dos dispositivos de red diferentes de forma simultánea). Usar DHCP permite a los administradores de red volver a configurar fácilmente las direcciones IP del cliente sin tener que realizar cambios a los clientes en forma manual. La mayoría de los proveedores de Internet utilizan DHCP para asignar direcciones a los clientes que no necesitan una dirección estática.



PROTOCOLO SMB y SERVICIOS PARA COMPARTIR ARCHIVOS

El Bloque de mensajes del servidor (SMB) es un protocolo cliente-servidor para compartir archivos. IBM desarrolló el Bloque de mensajes del servidor (SMB) a fines de la década de los 80 para describir la estructura de recursos de red compartidos, como directorios, archivos, impresoras y puertos seriales. Es un protocolo de solicitud-respuesta. A diferencia del protocolo para compartir archivos respaldado por FTP, los clientes establecen una conexión a largo plazo con los servidores. Una vez establecida la conexión, el usuario del cliente puede acceder a los recursos en el servidor como si el recurso fuera local para el host del cliente.

El intercambio de archivos SMB y los servicios de impresión se han transformado en el pilar de networking de Microsoft. Con la presentación de la serie Windows 2000 del software, Microsoft cambió la estructura subyacente para el uso del SMB. En versiones anteriores de los productos de Microsoft, los servicios de SMB utilizaron un protocolo que no es TCP/IP para implementar la resolución de nombres. Comenzando con Windows 2000, todos los productos subsiguientes de Microsoft utilizan denominación DNS. Esto permite que los protocolos TCP/IP den soporte directamente al intercambio de recursos SMB, como se muestra en la figura.



SMB es un protocolo de solicitud-respuesta y cliente-servidor. Los servidores pueden poner sus recursos a disposición de los clientes en la red.

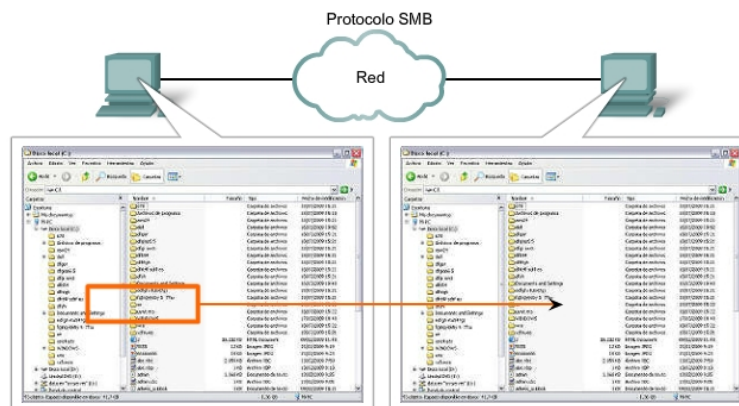
Los sistemas operativos LINUX y UNIX también proporcionan un método de intercambio de recursos con redes de Microsoft mediante una versión del SMB llamado SAMBA. Los sistemas operativos Macintosh de Apple también admiten recursos compartidos por medio del protocolo SMB.

El protocolo SMB describe el acceso al sistema de archivos y la manera en que los clientes hacen solicitudes de archivos. Además describe la comunicación entre procesos del protocolo SMB. Todos los mensajes SMB comparten un mismo formato. Este formato utiliza un encabezado de tamaño fijo seguido por un parámetro de tamaño variable y un componente de datos.

Los mensajes de SMB pueden:

- Iniciar, autenticar y terminar sesiones
- Controlar el acceso a los archivos y a la impresora
- Autorizar una aplicación para enviar o recibir mensajes para o de otro dispositivo

El intercambio de archivos de SMB se muestra en la figura.



Puede copiarse un archivo desde una PC a otra con Windows Explorer mediante el protocolo SMB.

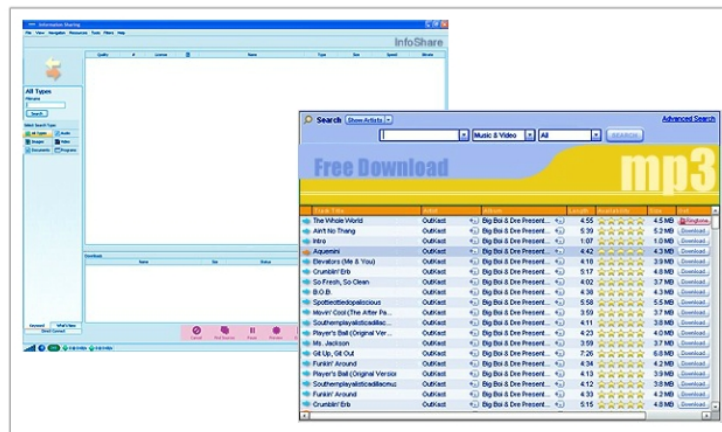
PROTOCOLO GNUTELLA y SERVICIOS P2P

Aprendimos acerca del FTP y del SMB como formas de obtener archivos, aquí presentamos otro protocolo de aplicación. Compartir archivos en Internet se ha transformado en algo muy popular. Con las aplicaciones P2P basadas en el protocolo Gnutella, las personas pueden colocar archivos en sus discos rígidos para que otros los descarguen. El software del cliente compatible con Gnutella permite a los usuarios conectarse con los

servicios Gnutella en Internet y ubicar y acceder a los recursos compartidos por otros pares Gnutella.

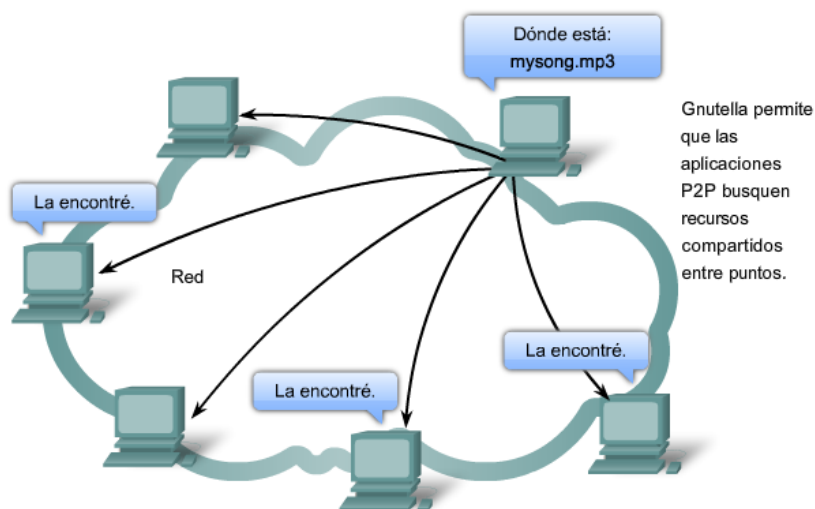
Muchas aplicaciones del cliente están disponibles para acceder en la red Gnutella, entre ellas: BearShare, Gnucleus, LimeWire, Morpheus, WinMX y XoloX (consulte una captura de pantalla de LimeWire en la figura). Mientras que el Foro de desarrolladores de Gnutella mantiene el protocolo básico, los proveedores de las aplicaciones generalmente desarrollan extensiones para lograr que el protocolo funcione mejor en dichas aplicaciones.

Aplicaciones punto a punto



Muchas de las aplicaciones P2P no utilizan una base de datos central para registrar todos los archivos disponibles en los puntos. Por el contrario, los dispositivos en la red se indican entre ellos qué archivos están disponibles cuando hay una consulta, y utilizan el protocolo Gnutella y los servicios para respaldar los recursos ubicados. Observe la figura.

Gnutella admite las aplicaciones P2P



Cuando un usuario se conecta a un servicio Gnutella, las aplicaciones del cliente buscan otros nodos Gnutella para conectarse. Estos nodos manejan las consultas para las ubicaciones de los recursos y responden a dichas solicitudes. Además, gobiernan los mensajes de control que ayudan al servicio a descubrir otros nodos. Las verdaderas transferencias de archivos generalmente dependen de los servicios HTTP.

El protocolo Gnutella define cinco tipos de paquetes diferentes:

- **ping:** para el descubrimiento del dispositivo
- **pong:** como respuesta a un ping
- **query:** para encontrar un archivo
- **query hit:** como respuesta a una consulta
- **push:** como una solicitud de descarga

PROTOCOLO y SERVICIOS TELNET

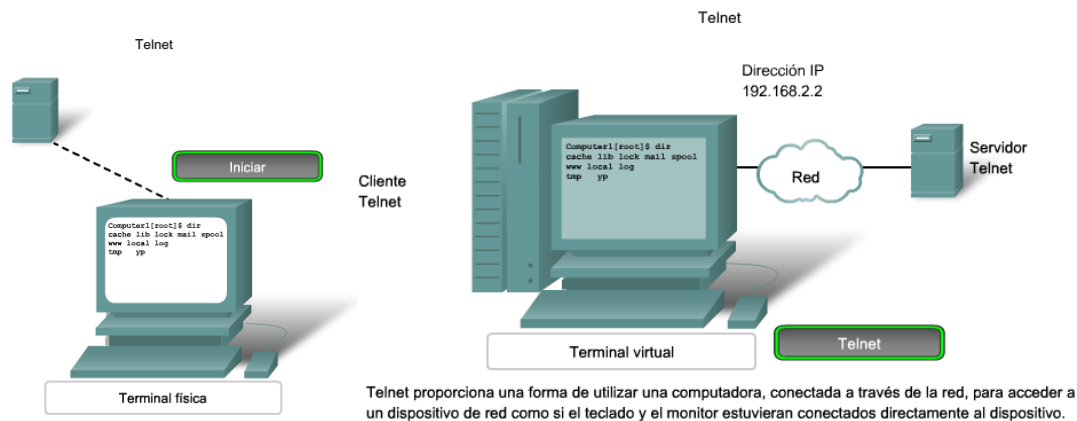
Mucho antes de que existieran las computadoras de escritorio con interfaces gráficas sofisticadas, las personas utilizaban sistemas basados en textos que eran simplemente terminales conectadas físicamente a una computadora central. Una vez que las redes estaban disponibles, las personas necesitaban acceder en forma remota a los sistemas informáticos de la misma manera en que lo hacían con las terminales conectadas directamente.

Telnet se desarrolló para satisfacer esta necesidad. Telnet se remonta a principios de la década de los 70 y se encuentra entre los servicios y protocolos de capa de aplicación más antiguo dentro del grupo TCP/IP. Telnet proporciona un método estándar de emulación de dispositivos de terminal con base en texto en la red de datos. El protocolo y el software del cliente que implementa son conocidos como Telnet.

De un modo adecuado, una conexión que utiliza Telnet se llama sesión o conexión de terminal virtual (VTY). En lugar de utilizar un dispositivo físico para conectarse al servidor, Telnet utiliza software para crear un dispositivo virtual que proporcione las mismas características de una sesión de terminal con acceso a la interfaz de línea de comandos (CLI) del servidor.

Para admitir conexiones del cliente a Telnet, el servidor ejecuta un servicio llamado demonio de Telnet. Se establece una conexión de terminal virtual desde un dispositivo final utilizando una aplicación del cliente Telnet. La mayoría de los sistemas operativos incluye un cliente de Telnet de la capa de aplicación. Telnet puede ejecutarse desde el indicador del sistema en una PC de Microsoft Windows. Otras aplicaciones de terminal comunes que ejecutan clientes Telnet son HyperTerminal, Minicom y TeraTerm.

Una vez establecida una conexión Telnet, los usuarios pueden realizar cualquier función autorizada en el servidor, como si utilizaran una sesión de línea de comandos en el servidor mismo. Si están autorizados, pueden iniciar y detener procesos, configurar el dispositivo e inclusive apagar el sistema.



Telnet es un protocolo cliente-servidor y especifica cómo se establece y se termina una sesión VTY. Además proporciona la sintaxis y el orden de los comandos utilizados para iniciar la sesión Telnet, así como también los comandos de control que pueden ejecutarse durante una sesión. Cada comando Telnet consiste en por lo menos dos bytes. El primer byte es un caracter especial denominado Interpretar como comando (IAC). Como su nombre lo indica, el IAC define el byte siguiente como un comando en lugar de un texto.

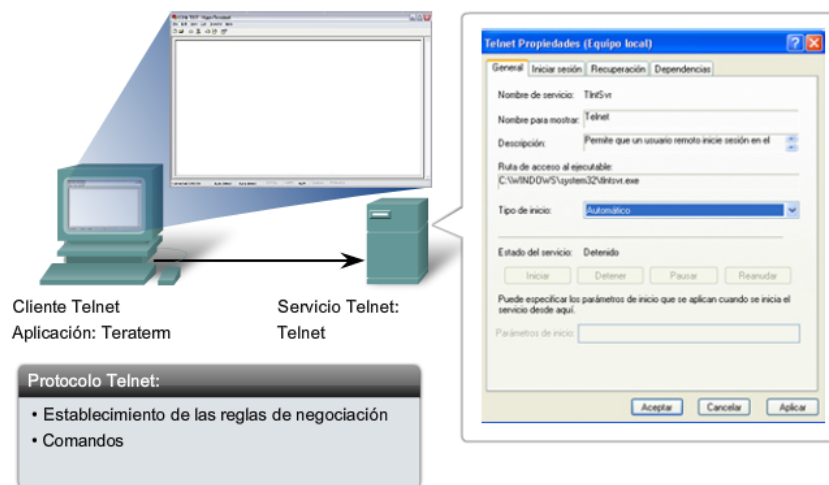
Algunas muestras de comandos del protocolo Telnet incluyen:

- **Are You There (AYT):** permite al usuario solicitar que aparezca algo en la pantalla de la terminal para indicar que la sesión VTY está activa.
- **Erase Line (EL):** elimina todo el texto de la línea actual.
- **Interrupt Process (IP):** suspende, interrumpe, aborta o termina el proceso al cual se conectó la terminal virtual. Por ejemplo, si un usuario inició un programa en el servidor Telnet por medio de VTY, puede enviar un comando IP para detener el programa.

Aunque el protocolo Telnet admite autenticación de usuario, no admite el transporte de datos encriptados. Todos los datos intercambiados durante una sesión Telnet se transporta como texto sin formato por la red. Esto significa que los datos se pueden interceptar y entender fácilmente.

Si la seguridad es un problema, el Protocolo shell seguro (SSH) ofrece un método seguro y alternativo para acceder al servidor. SSH proporciona la estructura para un inicio de sesión remoto seguro y otros servicios de red seguros. Además, proporciona mayor autenticación que Telnet y admite el transporte de datos de sesión con la autenticación. Como una mejor práctica, los profesionales de red deberían utilizar siempre SSH en lugar de Telnet, cada vez que sea posible.

Telnet: Aplicación, servicio y protocolo



PREGUNTAS y RESPUESTA REPASO

1) ¿Enumere los siete pasos del proceso para convertir comunicaciones humanas en datos?

- El usuario ingresa los datos mediante una interfaz de hardware.
- El software y el hardware convierten los datos a un formato digital.
- Los servicios de la aplicación inician la transferencia de datos.
- Las capas OSI encapsulan los datos hacia abajo en stack
- Los datos encapsulados se transportan a través de los medios hacia el destino
- Las capas OSI en el destino desencapsulan los datos hacia arriba en stack
- Los datos están listos para su procedimiento por parte del dispositivo final

2) ¿Describa las dos formas de software de capa de Aplicación y el propósito de cada una?

El software de aplicación tiene dos formas: **aplicaciones** y **servicios**.

Las aplicaciones están diseñadas para interactuar con nosotros. La aplicación es un software de usuario. Si el dispositivo es una computadora, la aplicación se iniciará generalmente por parte del usuario. Si bien puede haber varias capas de soporte por debajo, el software de aplicación proporciona una interfaz entre las personas y el hardware. La aplicación iniciará el proceso de transferencia de datos cuando el usuario presione el botón Enviar o realice una acción similar.

Los servicios son programas de fondo que realizan una función particular en la red de datos. Los servicios se invocan mediante un dispositivo conectado a la red o mediante una aplicación. Por ejemplo, un servicio de red puede proporcionar funciones de transmisión de datos u ofrecer conversión de datos en una red. En general, el usuario final no puede ver ni acceder directamente a los servicios. Éstos proporcionan la conexión entre una aplicación y la red.

3) ¿Explique el significado de los términos servidor y cliente en el contexto de las redes de datos?

Se hace referencia al extremo de origen de la comunicación de datos como "servidor" y al extremo receptor se le denomina "cliente". Los procesos de cliente y servidor son servicios de capa de aplicación que proporcionan las bases para la conectividad de red de datos.

En algunos casos los "servidores" y "clientes" son dispositivos que realizan dicha función en forma específica y exclusiva. Por ejemplo:

Un servidor central de archivos puede incluir los archivos de datos comerciales de una organización a los que acceden los empleados mediante sus estaciones de trabajo de cliente únicamente.

Los ejemplos basados en Internet incluyen servidores Web y servidores de correo donde muchos usuarios acceden a un servidor suministrado en forma centralizada.

En otras situaciones, como compartir archivos en una red residencial, los dispositivos individuales pueden realizar roles de servidor y cliente en diferentes momentos.

Los servidores son un depósito y una fuente de información a la vez, como archivos de texto, bases de datos, imágenes, videos o archivos de audio que se registraron previamente.

La función del servidor puede ser administrar las comunicaciones a medida que se producen. Esto se denomina comunicación "en tiempo real". Los ejemplos incluyen un servidor de inscripción de estudiantes universitarios en el que muchos usuarios pueden acceder a la misma base de datos al mismo tiempo, pero todos requieren la misma información actualizada; o, un servidor de comunicaciones que configura una comunicación telefónica IP en la que las direcciones de red del dispositivo deben coincidir con el número telefónico marcado.

El proceso del servidor puede llamarse "daemon de servidor" y normalmente se ejecuta de fondo en lugar de hacerlo bajo el control directo de un usuario final. Estos procesos de servidor ponen los datos de la comunicación a disposición de la red de datos. Se dice que los procesos de servidor "escuchan" la solicitud de un cliente. Cuando un servidor "recibe" una solicitud de un cliente, intercambia los mensajes apropiados con el cliente de acuerdo con lo requerido por el protocolo en uso y luego envía los datos solicitados. Los procesos del

cliente en el otro extremo de la comunicación a través de la red de datos permiten al usuario realizar solicitudes para obtener los datos de un servidor. El software del cliente normalmente utiliza un programa iniciado por un usuario. El cliente inicia el flujo de datos de comunicación desde el servidor al enviar solicitudes de datos al servidor. El servidor responde iniciando el envío de una o más transmisiones de datos al cliente. Además de la transferencia de datos real, este intercambio puede incluir la autenticación del usuario y la identificación del archivo de datos que se transferirá.

Si bien normalmente se considera que los datos fluyen del servidor al cliente, siempre se produce cierto flujo del cliente al servidor. A la transferencia de datos del cliente al servidor se le denomina carga y a la transferencia de datos de un servidor se le denomina descarga.

Los ejemplos de servicios cliente-servidor comunes incluyen:

- **DNS** (Servicios de nombres de dominios)
- **FTP** (Servicio de transferencia de archivos)
- **HTTP** (Protocolo de transporte de hipertexto)
- **Telnet** (Servicio de red de teletipo)

Es a través de los servicios cliente de la que la mayoría de los usuarios experimentan la red de datos, por lo tanto es importante entender esta área del networking.

4) Compare y diferencie la transferencia de datos cliente – servidor con la de punto a punto a través de las redes

La transferencia de datos cliente/servidor hace referencia específica al extremo de origen centralizado de la comunicación de datos como el servidor y al extremo receptor como cliente.

Con la transferencia de datos punto a punto, los servicios de cliente y servidor se utilizan dentro de la misma conversación. Cualquier extremo de la comunicación puede iniciar el intercambio y ambos dispositivos se consideran iguales en el proceso de comunicación. Los dispositivos en cualquier extremo de la comunicación se denominan puntos.

En contraste con el **modelo cliente/servidor**, en el que un servidor es normalmente el **depósito centralizado** y responde las solicitudes de varios clientes, **una red punto a punto tiene datos distribuidos**. Además, una vez que se estableció la comunicación, los puntos se comunican directamente; los datos no se procesan en la capa de Aplicación por parte de otro dispositivo en la red.

5) ¿Enumere cinco funciones generales que especifican los protocolos de la capa de aplicación?

Las funciones especificadas por los protocolos de la capa de Aplicación incluyen:

1. **Los procesos que se llevan a cabo en cualquier extremo de la comunicación:** Esto incluye lo que debe ocurrir con los datos y cómo debe estructurarse la Unidad de datos del protocolo. La PDU de la capa de Aplicación utilizada en este curso se denomina "datos".

2. **Los tipos de mensajes:** pueden incluir solicitudes, acuses de recibo, mensajes de datos, mensajes de estado y mensajes de error.

3. **La sintaxis del mensaje:** proporciona el orden esperado de la información (campos) en un mensaje.

4. **El significado de los campos** dentro de los tipos específicos de mensaje debe ser constante para que los servicios puedan actuar en forma correcta de acuerdo con la información.

5. **Los diálogos del mensaje:** determinan qué respuesta produce cada mensaje para que se invoquen los servicios correctos y tenga lugar la transferencia de datos.

6) ¿Suministre los objetivos específicos de los protocolos de la capa de aplicaciones DNS, HTTP, SMB y SMTP/POP?

Todos estos protocolos utilizan un **proceso cliente/servidor**.

Sistema de nombres de dominio (DNS) proporciona a los usuarios un servicio automatizado que conecta o resuelve nombres de recursos y dominios de correo electrónico con la dirección de red de dispositivo numérico adecuada. Este servicio está disponible para cualquier usuario conectado a Internet que ejecute una aplicación de capa de Aplicación tal como un navegador Web o programa de cliente correo electrónico.

El Protocolo de transferencia de hipertexto (HTTP) se desarrolló originalmente para publicar y recuperar páginas de Lenguaje de marcado de hipertexto (HTML) y actualmente se utiliza para sistemas de información de hipermedia distribuidos y de colaboración. HTTP es utilizado por World Wide Web (WWW) para transferir datos de servidores Web a clientes Web.

El Bloque de mensajes del servidor (SMB) describe la estructura para compartir recursos de red, como directorios, archivos, impresoras y puertos seriales entre computadoras.

El Protocolo simple de transferencia de correo (SMTP) transfiere correo electrónico salientes de cliente correo electrónico al servidor de correo electrónico y transporta correo electrónico entre servidores de correo electrónico y así habilita el intercambio de correo a través de Internet.

POP, o El POP3 (Protocolo de oficina de correos versión 3), entrega correo electrónico desde el servidor de correo electrónico al cliente.

7) ¿Compare y diferencie los mensajes que los protocolos de la capa de aplicación, como DNS, HTTP, SMB y SMTP/POP, intercambian entre dispositivos para habilitar la realización de la transferencia de datos?

DNS incluye consultas, respuestas y formatos de datos estándar. Las comunicaciones del protocolo DNS se realizan en un formato único llamado mensaje. Este formato de mensaje se utiliza para todos los tipos de consultas de clientes y repuestas de servidores, mensajes de error y para la transferencia de información de registros de recursos entre servidores.

HTTP es un protocolo de solicitud/respuesta:

Una aplicación de capa de aplicación de cliente, normalmente un navegador Web, envía un mensaje de solicitud al servidor.

El servidor responde con el mensaje apropiado.

El protocolo también incluye mensajes para subir datos al servidor, como al completar un formulario en línea. Los mensajes

SMB utilizan un formato común para:

iniciar, autenticar y finalizar sesiones

controlar el acceso a archivos e impresoras

permitir a una aplicación enviar o recibir mensajes desde o hacia otro dispositivo

SMTP especifica comandos y respuestas relacionadas con el inicio de sesión, transacción de correo, reenvío de correo, verificación de nombres de buzón de correo, expansión de listas de correo e intercambios de apertura y cierre.

POP es un protocolo representativo de cliente/servidor en el que el servidor espera las conexiones del cliente y el cliente inicia la conexión al servidor. Luego, el servidor puede transferir el correo electrónico.

Todos los protocolos anteriores utilizan mensajes de solicitud/respuesta de servidor/cliente. Mientras que los usuarios ven las aplicaciones que utilizan **HTTP** (un navegador Web), **SMB** (administrador de archivos) y **SMTP/POP** (cliente correo electrónico), el funcionamiento de DNS se lleva a cabo en forma subyacente a estas aplicaciones y es completamente transparente para el usuario.

Bibliografía:

- Cisco Systems: Academia de Networking de Cisco Systems: Guía del primer año. — 2ª ed. — Madrid: Pearson Educación, 2002. ISBN 8420532967
- Curso Cisco Systems: Academia de Networking de Cisco Systems – Version 4.0
- Curso visual y práctico: Administrador de Redes Instalación y configuración de hardware y software. — USERS-CISCO.
- Stallings, William: Comunicación y Redes de Computadores / W. Stallings. — 7ª. — Madrid: Pearson Educación, 2004. ISBN 9788420541105