DAVID MOISÉS TERÁN PÉREZ

ADMINISTRACIÓN Y SEGURIDAD

EN REDES DE COMPUTADORAS

Alfaomega

DAVID MOISÉS TERÁN PÉREZ

ADMINISTRACIÓN Y SEGURIDAD
EN REDES DE COMPUTADORAS

Alfaomega

Director Editorial
Marcelo Grillo Giannetto
mgrillo@alfaomega.com.mx

Jefe de Ediciones
Francisco Javier Rodríguez Cruz
jrodriguez@alfaomega.com.mx

Datos catalográficos
Terán Pérez, David Moisés

Administración y seguridad en redes
de computadoras

Primera Edición

Alfaomega Grupo Editor, S.A. de C.V. México

ISBN: 978-607-538-097-1

Formato: 17 x 23 cm Páginas 460

Administración y seguridad en redes de computadoras

Primera edición: Alfaomega Grupo Editor, México, enero 2018

Dr. Isidoro Olvera (Eje 2 sur) No. 74, Col. Doctores, C.P. 06720, Cuauhtémoc, Ciudad de México

Miembro de la Cámara Nacional de la Industria Editorial Mexicana
Registro No. 2317

Pág. Web: **http://www.alfaomega.com.co**
E-mail: **cliente@alfaomegacolombiana.com**

ISBN: 978-958-778-407-7

Empresas del grupo:

México: Alfaomega Grupo Editor, S.A. de C.V. – Dr. Isidoro Olvera No. 74, Col. Doctores, C.P. 06720, Cuauhtémoc, Cd. de Méx.
Tel.: (52-55) 5575-5022 – Fax: (52-55) 5575-2420 / 2490. Sin costo: 01-800-020-4396
E-mail: atencionalcliente@alfaomega.com.mx

Colombia: Alfaomega Colombiana S.A. – Calle 62 No. 20-46, Barrio San Luis, Bogotá, Colombia
Tels.: (57-1) 746 0102 / 210 0122 – E-mail: cliente@alfaomegacolombiana.com

Chile: Alfaomega Grupo Editor, S.A. – Av. Providencia 1443. Oficina 24, Santiago, Chile
Tel.: (56-2) 2235-4248 – Fax: (56-2) 2235-5786 – E-mail: agechile@alfaomega.cl

Argentina: Alfaomega Grupo Editor Argentino S.A. - Av. Córdoba 1215 Piso 10 - C.P. 1055
Ciudad Autónoma de Buenos Aires, Argentina
Tel/Fax: (54-11) 4811-0887 – E-mail: ventas@alfaomegaeditor.com.ar
www.alfaomegaeditor.com.ar

Acerca del autor

David Moisés Terán Pérez

Es ingeniero mecánico electricista con especialidad en sistemas eléctricos de potencia, egresado de la Universidad Nacional Autónoma de México (UNAM); maestro en Microelectrónica por la Universidad de La Sorbona (París, Francia), maestro en ciencias de la educación por la Universidad del Valle de México (UVM); y doctor en educación por la Universidad Pedagógica Nacional (UPN). Cuenta con 27 años de experiencia como docente en educación superior como la Universidad Nacional Autónoma de México (UNAM), el Instituto Tecnológico de Estudios Superiores de Monterrey (ITESM), la Universidad del Pedregal, la Universidad Tecnológica de México (UNITEC), la Universidad ICEL, la Escuela Bancaria Comercial (EBC), entre muchas otras.

A Diana, Emanuel y David

Mensaje del Editor

Una de las convicciones fundamentales de Alfaomega es que los conocimientos son esenciales en el desempeño profesional, ya que sin ellos es imposible adquirir las habilidades para competir laboralmente. El avance de la ciencia y de la técnica hace necesario actualizar continuamente esos conocimientos, y de acuerdo con esto Alfaomega publica obras actualizadas, con alto rigor científico y técnico, y escritas por los especialistas del área respectiva más destacados.

Consciente del alto nivel competitivo que debe de adquirir el estudiante durante su formación profesional, Alfaomega aporta un fondo editorial que se destaca por sus lineamientos pedagógicos que coadyuvan a desarrollar las competencias requeridas en cada profesión específica.

De acuerdo con esta misión, con el fin de facilitar la comprensión y apropiación del contenido de esta obra, cada capítulo inicia con el planteamiento de los objetivos del mismo y con una introducción en la que se plantean los antecedentes y una descripción de la estructura lógica de los temas expuestos; asimismo, a lo largo de la exposición se presentan ejemplos desarrollados con todo detalle y cada capítulo finaliza con unas conclusiones y algunos cuentan con una serie de prácticas.

Los libros de Alfaomega están diseñados para ser utilizados en los procesos de enseñanza aprendizaje, y pueden ser usados como textos en diversos cursos o como apoyo para reforzar el desarrollo profesional; de esta forma, Alfaomega espera contribuir a la formación y al desarrollo de profesionales exitosos para beneficio de la sociedad, y espera ser su compañera profesional en este viaje de por vida por el mundo del conocimiento.

Contenido

Capítulo 7

Introducción

Administración y seguridad en Redes de Computadoras presenta herramientas teóricas y prácticas que permiten a los ingenieros prepararse para las certificaciones de CISCO, las cuales evalúan los conocimientos y las habilidades que se tienen sobre del diseño y soporte de redes. Para ello se muestran una serie de prácticas y bancos de preguntas que simulan las que aplica CISCO.

La obra, como su título lo anuncia, aborda dos temas: la administración de redes de computadoras y su seguridad. En el primero, se plantea cómo configurar la red para que no existan fallas y que ésta sea funcional, dependiendo de la cantidad de usuarios que la utilizan. En la seguridad informática se desarrollan los principios y fundamentos de la teoría de la seguridad informática, además de los objetivos, políticas, procedimientos y arquitectura de la seguridad.

El libro está dividido en siete capítulos, cuenta con dos secciones de prácticas y dos de bancos de preguntas para la certificación de CISCO.

Generalidades sobre la administración de una red de computadoras

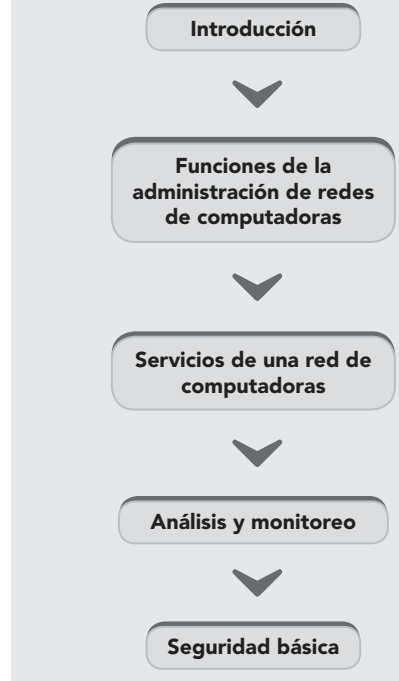
La diferencia entre la genialidad y la estupidez humana: es que la genialidad, sí tiene sus límites.

Albert Einstein

Después de estudiar este capítulo, el lector será capaz de:

- Entender qué es la administración de una red de computadoras.
- Comprender la importancia de la administración de una red de computadoras.
- Establecer en qué consiste la administración de una red de computadoras.

Para cumplir con los objetivos planteados, el siguiente diagrama de flujo presenta la forma de acceder a los contenidos:



1.1 Introducción

La evolución en las técnicas de gestión de una red de computadoras se dirige siempre con los avances en las tecnologías, lo que ha conducido al progreso de las redes de transmisión de datos en las organizaciones, en las cuales se asientan los diversos sistemas de administración, que se caracterizan por recursos en constante incremento del número, complejidad y heterogeneidad.

En este sentido, las redes de transmisión de datos surgieron como el medio de interconectar diferentes equipos que, instalados de forma remota unos con otros, han ofrecido capacidades de acceso a distintos servicios: diversas capacidades de procesamiento, bases de datos, conocimientos internacionales, compartición de grandes recursos, edición y almacenamiento de información, etcétera (Terán Pérez, 2014).

La administración de redes convergentes de computadoras abarca en general el tratamiento de datos estadísticos e información sobre el estado de distintas partes de la red, y se llevan a cabo las acciones necesarias para ocuparse de fallas y otros cambios. La técnica más primitiva para la monitorización de éstas es hacer *pinging*[1] entre los servidores críticos, método basado en un datagrama de eco (*echo*), el cual produce una réplica inmediata cuando llega al destino (Terán Pérez, 2010).

La mayoría de las implementaciones TCP/IPv4[2] (o en un futuro, el TCP/IPv6) incluyen el *ping*, por medio del cual se sabrá qué ocurre en la red: si se recibe una réplica, significa que el servidor se encuentra activo; en caso contrario, se sabrá que hay algún error. Si los *ping* a todos los servidores de una red no dan respuesta, es lógico concluir que la conexión a dicha red, o esta misma, no funciona; si solamente uno de los servidores no responde, es razonable concluir que sólo un servidor tiene algún problema (Kurose y Keith, 2005). También hay un enfoque oficial TCP/IP para llevar a cabo la monitorización: en la primera fase se usa un conjunto de protocolos SGMP (*Simple Gateway Monitoring Protocol*) y SNMP (*Simple Network Management Protocol*) ambos diseñados para recoger información y cambiar los parámetros de la configuración y otras entidades de la red. El primero se encuentra disponible para varias pasarelas (*gateways*) comerciales, así como para sistemas UNIX; además, tiene mecanismos para añadir informaciones que varían de un dispositivo a otro. Cualquier implementación SGMP necesita que se proporcione un conjunto de datos para que empiece a funcionar (Black, 1997).

Por su parte, el protocolo SNMP apareció a finales de 1988, es ligeramente más complejo y, por ende, requiere mayor información para trabajar, razón por la que usa el llamado MIB (*Management Information Base*), que es resultado de numerosas reuniones de comités formados por vendedores y usuarios. También se espera la elaboración de un equivalente de TCP/IP de CMIS (*Content Management Interoperability Services*), el servicio ISO de monitorización de redes; sin embargo, CMIS y sus protocolos CMIP (*Common Management Information Protocol*) todavía no son estándares oficiales ISO, pero se encuentran en fase experimental.

[1] El mecanismo del comando *ping* es similar al usado en el sonar, puesto que permite observar si hay conectividad entre dos computadoras y registra el tiempo que tardan en llegar los paquetes con relación en la tardanza de la respuesta.

[2] El modelo TCP/IP se utiliza para comunicaciones en red. Éste proporciona conectividad de extremo a extremo especificando la manera en la cual deben ser transmitidos, direcciones, formateados y enrutados los datos para el destinatario.

En términos generales, todos los protocolos persiguen el mismo objetivo, que es recoger información crítica de una forma estandarizada; para ello se ordena la emisión de datagramas UDP (*User Datagram Protocol*) desde un programa de administración de redes que se ejecuta en alguno de los servidores. Por lo regular, la interacción es bastante simple, se realiza con el intercambio de un par de datagramas básicos: una orden y una respuesta. El mecanismo de seguridad también es muy sencillo porque permite que se incluyan contraseñas en las órdenes (en SGMP se conoce como una sesión de nombre —*session name*—, en lugar de una contraseña). También existen métodos de seguridad más elaborados basados en la criptografía, los cuales se abordarán más adelante (Boggs, Mogul y Kent, 1988).

La existencia de dispositivos de comunicaciones dispersos que implementan e interconectan entre sí todas estas redes, obliga a disponer de sistemas de gestión para la configuración, supervisión, diagnóstico y mantenimiento de los dispositivos. Los enlaces de comunicaciones para el acceso a servicios avanzados de telecomunicaciones han permitido que los más avanzados gestionen las redes para un mejor aprovechamiento. Si bien en un principio la administración de una red significaba atención más o menos individualizada a los elementos, en la actualidad se busca una gestión única de la red (Stallings, 2010).

La gestión de redes y de sistemas comprende las aplicaciones específicas que incorporan la posibilidad de gestión e interacción en parámetros asociados al rendimiento, así como de las plataformas web, las bases de datos y, en definitiva, los sistemas que componen la arquitectura de una organización; es decir, los servidores. Aunque por lo regular se identifican con un determinado *hardware* (incluyendo componentes, garantías y actividades de reparación) y un *software* de base estáticos, se requiere realizar actividades continuas de gestión de los mismos para así mejorar la disponibilidad, la seguridad, la integración con el resto de los elementos, la confiabilidad, el rendimiento, etcétera (Terán Pérez, 2010).

Los principales problemas relacionados con la expansión de las redes de computadoras son la administración del correcto funcionamiento día a día y la planificación estratégica de su crecimiento; de hecho, se estima que más del 70% del costo total de una red corporativa se atribuye a estos. Por todo ello, la gestión de red integrada como conjunto de actividades dedicadas al control y vigilancia de los recursos de telecomunicación, se ha convertido en un aspecto de enorme importancia en el mundo de las comunicaciones (Millán Tejedor, 2009).

1.2 Funciones de la administración de redes de computadoras

La administración de redes es una profesión muy demandada. La proliferación de las redes informáticas y las consecuencias en el uso y abuso, ha hecho que la administración de redes, se convierta en una tarea imprescindible para las organizaciones (empresas, industrias, corporativos, dependencias o instituciones). La administración de redes informáticas se vinculada con la tecnología de la información (TI). Esta última se relaciona con el diseño, el desarrollo, la implementación, el soporte y la administración de *hardware*, *software*, y los sistemas informáticos en red. Los profesionales de TI poseen un amplio conocimiento sobre sistemas de computación y sistemas operativos; además gozan de las capacidades necesarias para llevar a cabo la administración de la red.

El concepto de la administración de las redes informáticas define las diversas tareas que desarrollan los profesionales de TI en una red informática con el objetivo de brindar de forma efectiva numerosos servicios de red, garantizando la disponibilidad y la calidad que espera el usuario final. La administración de red involucra personas, *software* y *hardware*. Los administradores de redes son los profesionales de TI que garantizan el servicio continuo al usuario. El conjunto de herramientas de gestión de redes forman parte del *software* que participa en la tarea de administración. Y por ende, el *hardware* se relaciona de forma directa con los dispositivos de red usados para administrar dicha red.

En las empresas y en los centros de datos, la administración de redes de computadoras es más compleja con mayor volumen de trabajo en comparación a la administración de redes domésticas, debido al cúmulo de información que se almacena y a la infraestructura de dichas redes. La administración de redes de computadoras empresariales, se orienta a garantizar con éxito el mantenimiento y funcionamiento adecuado de los servicios de correo electrónico, la administración y la seguridad de las bases de datos y del sitio web empresarial; que por consecuencia contribuye a la satisfacción del cliente.

1.2.1 Configuración y administración

La configuración comprende las funciones de monitoreo y mantenimiento del estado de la red, las cuales incluyen

- Servicios de soporte técnico (configuraciones).
- Asegurarse que la red sea utilizada con eficiencia.
- Verificar que los objetivos de calidad del servicio sean alcanzados.

Un administrador de red sirve a los usuarios de distintas formas, crea espacios de comunicación, atiende sugerencias, mantiene a tiempo y en buena forma las herramientas requeridas, vigila el adecuado funcionamiento del *hardware* y *software* de las computadoras y redes a su cargo, también se ocupa de la documentación que describe la red; respeta la privacidad de los usuarios y promueve el buen uso de los recursos (Bertsekas y Gallager, 1992). A cambio de tantas responsabilidades, la recompensa es el buen funcionamiento de la red como un medio que vincula a las personas a través del uso de las computadoras y los programas como herramientas para agilizar algunas labores que dan tiempo y eficiencia para la productividad personal.

De igual forma, el administrador de red debe conocer las reglas de *root* (en sistemas operativos del tipo UNIX, es el nombre convencional de la cuenta de usuario que posee todos los derechos en todos los modos —mono o multiusuario—), que también es llamado súper-usuario o la cuenta del administrador. El *root* puede hacer muchas cosas que un usuario común no puede, como cambiar el dueño o los permisos de los archivos y enlazar a puertos de numeración pequeña. No es recomendable utilizar el usuario *root* para una simple sesión de uso habitual porque se pone en riesgo el sistema al garantizar acceso privilegiado a cada programa en ejecución; por ello, es preferible usar una cuenta normal y el comando su para acceder a los privilegios de *root* de las máquinas que se administra, dado que puede configurar servicios y establecer políticas que afectarán a los usuarios de la red. Algunas de las labores que sólo pueden hacerse desde la cuenta del administrador son:

- Cambiar el nombre de la cuenta que permite gestionar un sistema Linux.
- Configurar los programas que se inician junto con el sistema.
- Administrar las cuentas de usuarios; así como los programas y la documentación instalada.
- Configurar los programas y dispositivos.
- Ajustar la zona geográfica, fecha hora.
- Supervisar el espacio en discos y mantener las copias de respaldo.
- Precisar los servicios que funcionarán en la red.
- Solucionar problemas con dispositivos o programas.

Finalmente, en este contexto, el término "monitoreo" describe el uso de un sistema que revisa constantemente una red de computadoras en busca de componentes defectuosos o lentos para luego informar cuál la problemática y dar solución a la misma (Bhatti y Crowcroft, 2000). La administración de redes es un subconjunto de funciones: mientras que un sistema de detección de intrusos monitorea una red por amenazas externas, otro monitorea en busca de problemas causados por la sobrecarga en la infraestructura y fallas en los servidores, figura 1.1.

Figura. 1.1 Administración de una red de computadoras en la práctica

Para determinar el estatus de un servidor web, un *software* de monitoreo envía con periodicidad peticiones del protocolo de transferencia de hipertexto (HTTP, *Hypertext Transfer Protocol*) con el objetivo de obtener páginas para un servidor de correo electrónico y después enviar mensajes mediante el protocolo SMTP, los cuales posteriormente serán retirados mediante IMAP (*Internet Message Access Protocol*) o POP3 (*Protocolo Post Office*). Por lo general, los datos evaluados son tiempo de respuesta y disponibilidad (*uptime*), aunque también se obtienen estadísticas con consistencia y fiabilidad que han ganado popularidad.

La generalizada instalación de dispositivos de optimización para redes de áreas extensas (WAN, *Wide Area Network*) tiene un efecto adverso en la mayoría del *software* de monitoreo, en especial al intentar medir de manera precisa el tiempo de respuesta de punto a punto, dado el límite de visibilidad de ida y vuelta. Las fallas de peticiones de estado producen ciertas acciones variables desde del sistema de monitoreo. Por ejemplo, una alarma puede ser enviada al administrador como una ejecución automática de mecanismos de controles de fallas (Held, 2010).

La monitorización del servidor puede ser interna (por ejemplo, se comprueba su *software*) o externa (revisión manual); en ambas se verifican características como el uso del procesador y la memoria, el rendimiento de la red, el espacio libre en disco y las aplicaciones instaladas. A lo largo de este proceso se comprueban también los códigos HTTP enviados del servidor (definidos en la especificación HTTP RFC[3] 2 616) que son la forma más rápida de comprobar el funcionamiento. A continuación, se dan ejemplos de las herramientas de monitoreo más utilizadas:

Tcpdump. Permite monitorear a través de comandos de la consola de LINUX, todos los paquetes que atraviesan la interfaz indicada. A la par de los múltiples filtros, parámetros y opciones, *tcpdump* ofrece infinidad de combinaciones para monitorear, como el tráfico que ingresa de una IP, un servidor o una página determinada; se puede solicitar el tráfico de un puerto específico o pedirle que muestre todos los paquetes cuyo destino sea una dirección MAC determinada.

Wireshark: *sniffer.*[4] Captura las tramas y paquetes que pasan a través de una red. Cuenta con todas las características estándar de un analizador de protocolos y posee una interfaz gráfica fácil de manejar. Se usa con frecuencia en Ethernet, aunque es compatible con otras redes.

Hyperic. Aplicación que posibilita administrar infraestructuras virtuales, físicas y en la nube. Este programa autodetecta muchas tecnologías y cuenta con dos versiones: una *open source* y una comercial. Algunas de sus características son la optimización para ambientes virtuales que integran *vCenter* y *vSphere*, también cuenta con capacidad para funcionar en 75 componentes comunes como bases de datos en dispositivos y servidores de red y detecta de forma automática los componentes de cualquier aplicación virtual.

Nagios. Sistema de monitoreo que concede a cualquier empresa identificar y resolver errores críticos antes de que afecten los procesos de negocio. En caso de un error, la aplicación se encarga de alertar al grupo técnico para que rápidamente sea resuelto sin afectar a los usuarios finales.

[3] Los RFC o *Request for Comments* son publicaciones de cierto grupo de trabajo de ingeniería de Internet que describen distintos aspectos del funcionamiento de éste en protocolos y redes de computadoras.

[4] El programa informático que registra la información enviada por los periféricos, así como la actividad en una computadora.

La evolución y tendencias de las herramientas de monitoreo de redes establecen retos a los ejecutivos de la Tecnología de la Información (TI) como mostrar los datos de su operación para que los ejecutivos de la organización dispongan de elementos suficientes para reconocer y fomentar la importancia de la tecnología como un componente habilitador del negocio (Held, 2010). Dicha evolución se ha alimentado mediante la llegada de protocolos más avanzados de visualización de tráfico como Netflow, Jflow, Cflow, Sflow, IPFIX o Netstream; el propósito hoy en día es tener una perspectiva global del "todo" para categorizar adecuadamente los eventos que afectan el desempeño de un servicio o del proceso de negocio involucrado (Held, 2010). De hecho, a medida que han avanzado las tecnologías, se ha atravesado por diferentes etapas en el monitoreo de redes, las cuales se enumeran a continuación:

Primera generación. Aplicaciones propietarias para monitorear dispositivos activos o inactivos. La industria ha desarrollado un sinfín de herramientas para tratar de presentar los recursos de una forma amable y en tiempo real, éstas presentan los elementos a través de un código universal de colores:

Verde: todo funciona en orden.

Amarillo: detección de algún problema temporal que no afecta la disponibilidad; sin embargo, se deben realizar ajustes para no perder la comunicación.

Anaranjado: el problema se ha hecho persistente y requiere pronta atención para evitar afectaciones a la disponibilidad.

Rojo: el dispositivo se encuentra fuera de servicio en ese momento y requiere acciones inmediatas para su restablecimiento.

Segunda generación. Aplicaciones de análisis de parámetros de operación a profundidad; por medio de estos, las herramientas realizan un análisis detallado con el fin de evaluar los estados de los componentes dentro de los dispositivos (microprocesador, memoria, espacio de almacenamiento, paquetes enviados y recibidos, *broadcast*, *unicast*, *multicast*, etcétera); de manera que es posible ajustar los parámetros y establecer los niveles de servicio del dispositivo. Este tipo de aplicaciones se apoyan en *sniffers* y en elementos físicos distribuidos conocidos como probadores (*probes*), cuya función es, exclusivamente, colectar estadísticas del tráfico, controladas desde una consola central.

Tercera generación. Aplicaciones de análisis punta a punta con enfoque al servicio. Con mayores niveles de información sobre los dispositivos se tienen elementos adicionales de análisis, pero aún no existen suficientes parámetros para tomar decisiones. Esta generación de aplicaciones con enfoque transaccional captura flujos de tráfico e identifica cuellos de botella y latencias a lo largo de las conexiones que existen entre los componentes de un servicio, entregando información acerca de "la salud" de éste; además logra conectar todas las partes de manera más eficiente; en dicho caso, cada dispositivo sabe cuándo se debe informar a otro sin afectarlo en sus tareas, esto para no generar una sobrecarga de información, lo cual significa que existe una toma de decisiones con enfoque de repercusión, generada en los negocios.

Cuarta generación. La personalización de indicadores de desempeño de los procesos de negocio llevando el crecimiento de las soluciones tecnológicas a los requerimientos de las organizaciones actuales dentro de las cuales están aquellas que monitorean el Desempeño de Aplicaciones (APM, *Application*

Performance Management), donde convergen elementos de tecnología (*Backend*) con los sistemas de los que forman parte para llevar a cabo las transacciones que impulsan los procesos de negocio (*Frontend*); en otras palabras, es un análisis de "punta a punta". El potencial de estas herramientas permite tener información simultánea de:

- Predicciones del desempeño.
- Modelado de escenarios (simulación y emulación).
- Análisis y planeación de capacidades.
- Funcionalidades de ajustes a las configuraciones.
- Mediciones de impacto al negocio (calidad, salud y riesgos en los servicios prestados).
- Experiencia del usuario.

En resumen, el término "configuración" tiene un significado diferente dependiendo del contexto en que sea utilizado, aun dentro del diseño y la operación de las redes de computadoras (Held, 2010). Una configuración puede ser:

- Una descripción de un sistema distribuido basado en la ubicación física y geográfica de los recursos, la cual incluye la manera en que se encuentran interconectados además de la información sobre las relaciones lógicas. Ésta puede basarse en diferentes puntos de vista (organizacional, administrativo, etcétera) o en aspectos relacionados con la seguridad.
- Un proceso para manipular la estructura del sistema distribuido donde se establecen o modifican parámetros que controlan la operación y ajustan el ambiente requerido para su funcionamiento normal.
- El resultado de un proceso de ordenamiento donde el sistema generará un conjunto de ciertos valores de los parámetros característicos para la operación normal del recurso.

La configuración, como el proceso de adaptación de los sistemas a un ambiente operativo, implica instalar nuevo *software*, actualizar el viejo, conectar dispositivos y hacer cambios en la topología de la red o en su tráfico (Day y Zimmermann, 1983); aunque ésta se acompaña de técnicas para hacer cambios e instalaciones físicas, se considera intensiva en procesos controlados por *software* y ajuste de parámetros, que incluyen, entre otros, selección de funciones, autorización, protocolos (longitud de los mensajes, tamaños de ventanas, timers, prioridades), de conexión (tipo y clases de dispositivo, velocidad de transmisión, paridad); en entradas en las tablas de enrutamiento, en servidores de nombres; en directorios; en parámetros de filtraje para cortafuegos (*firewalls*) y encaminadores (*routers*); para el algoritmo de STP (*Spanning Tree Protocol*),[5] para los enlaces conectados a los encaminadores (interfaces, ancho de banda), para máxima cuota de los usuarios, etcétera.

Los siguientes aspectos se presentan en relación con el proceso de la configuración, donde las herramientas que ayudan a disponer los componentes son:

[5] STP es un protocolo de red, nivel 2 del Modelo OSI (capa de enlace de datos); gestiona la presencia de bucles en topologías de red debido a la existencia de enlaces redundantes (necesarios en muchos casos para garantizar la disponibilidad de las conexiones), además, permite a los dispositivos de interconexión activar o desactivar en automático los enlaces de conexión de forma que se garantice la eliminación de bucles. STP es transparente a las estaciones del usuario.

Lugar. Una configuración puede hacerse en el mismo dispositivo o realizarse de forma remota; por ejemplo, por medio de un servidor DHCP. El sistema que se configura no siempre es compatible desde el equipo (servidor remoto) a configurar, esto puede deberse a razones técnicas, de organización o seguridad.

Almacenamiento. La configuración puede almacenarse en NVRAM (*Non-Volatile Random Access Memory*), en discos duros, en un *boot server* para ser "invocada" a través de los protocolos adecuados o, incluso, es posible que se recargue a través de la red.

Validez. Una configuración estática es aquella donde cada reconfiguración implica interrumpir la operación del componente (apagar y prender o reiniciar el sistema). Una dinámica, por otra parte, permite hacer cambios mientras el componente opera.

Interfaz de usuario del configurador. La calidad de la interfaz de usuario depende de la magnitud de los parámetros que pueden ser modificados con rapidez o aplicados al mismo tiempo a varios dispositivos. El sistema de configuración y la documentación de ésta deben estar protegidos de uso no autorizado (con contraseñas, en áreas físicas restringidas, con seguridad del protocolo de configuración, etcétera).

La administración de la configuración implica establecer parámetros, definir valores de umbral (*threshold*), determinar filtros, asignar nombres a los objetos administrados, proveer la documentación necesaria y cambiar las configuraciones cuando sea necesario (figura 1.2).

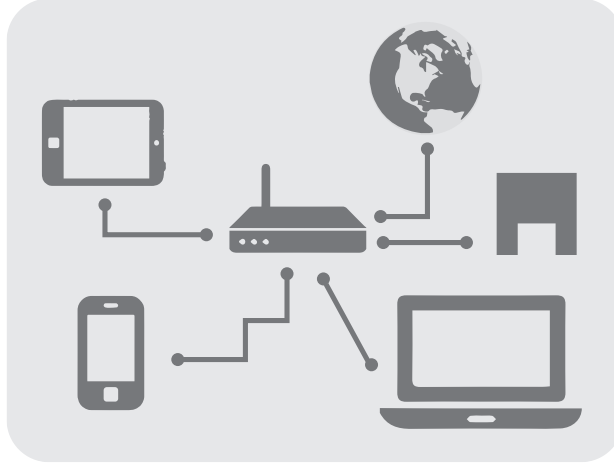


Figura. 1.2 Administración de la configuración para el uso de equipos en una red de computadoras

Las herramientas para la administración de la configuración deberían cubrir:

- Autotopología y autodescubrimiento de los componentes de red permitiendo extrapolar una descripción de la configuración a partir de un ambiente real.

- Sistemas para describir la documentación de las configuraciones o bases de datos maestras.
- Elementos para generar mapas de la configuración de una red. Es decir, el uso de MRTG (*Multi Router Traffic Grapher*) que en la práctica es un recopilador de datos del tráfico de la red, para supervisar la red. O también, el uso de Nagios que es un sistema de monitorización de redes.
- Herramientas para activar sistemas de respaldo (*back-up*).
- Instrumentos para establecer e invocar parámetros de configuración y estados del sistema.
- Materiales para distribuir *software* y controlar el uso de licencias.
- Componentes para supervisar y controlar la autorización.

1.2.2 Fallas

La administración de las fallas se relaciona con la detección, aislamiento y eliminación de comportamientos anormales del sistema (figura 1.3). La identificación y seguimiento de éstas es un problema operacional importante en todos los sistemas de procesamiento de datos (Lin y Costello, 2004). En comparación con sistemas no conectados a una red, la administración de fallas en redes de computadoras y sistemas distribuidos es más difícil por una diversidad de razones que incluyen, entre otras, el mayor número de componentes involucrados, la amplia distribución física de los recursos, la heterogeneidad de los componentes de *hardware* y *software*, así como las diferentes unidades de la organización involucradas.

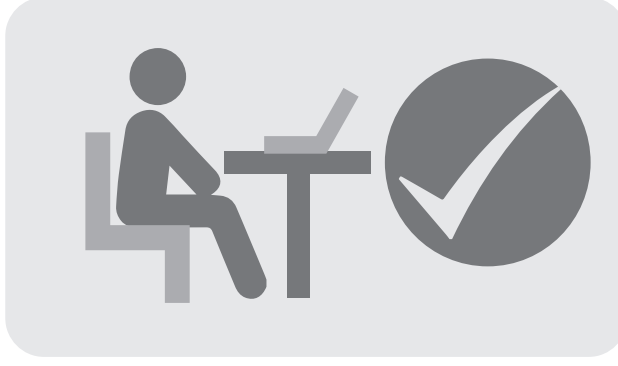


Figura 1.3 Correcta administración de las fallas en una red de computadoras

Una falla puede ser definida como una desviación de las metas operacionales establecidas con respecto en las funciones del sistema o los servicios. Los mensajes sobre las fallas son dados a conocer por el mismo componente o por los usuarios del sistema. Algunas de las fuentes más frecuentes de errores son:

- Los elementos que conforman los enlaces (cables UTP o de fibra óptica, líneas dedicadas, canales virtuales, entre otros).

- El sistema de transmisión (*transceivers*, *concentradores*, *switches*, servidores de acceso, encaminadores).
- Sistemas finales (clientes o servidores).
- El *software* de los componentes.
- La operación incorrecta por parte de los usuarios y los operadores.

La función de la administración de las fallas es detectar y corregir con rapidez los errores para asegurar un alto nivel de disponibilidad de un sistema distribuido y los servicios que éste presta; por otro lado, las tareas son:

- Monitoreo del sistema y la red.
- Respuesta y atención a alarmas.
- Diagnóstico de las causas de la falla.
- Establecer la propagación de errores.
- Presentar y evaluar medidas para recuperarse de los errores.
- Operación de sistemas de etiquetación de problemas (TTS, *Trouble Tickets Systems*)
- Proporcionar asistencia a usuarios *(Help Desk*).

Además, las siguientes capacidades técnicas pueden ayudar en el análisis de fallas:

- Autoidentificación de los componentes del sistema.
- Realización de pruebas, por separado, con los componentes del sistema.
- Facilidades de seguimiento (*tracing*).
- Disposición de una bitácora (*log*) de errores.
- Hacer eco de los mensajes en todas las capas del protocolo.
- Posibilidad de revisar los vaciados (*dumps*) de memoria.
- Métodos para generar errores a propósito en ambientes predefinidos para el sistema.
- Posibilidad de iniciar rutinas de autoverificación y transmisión de datos de prueba a puertos específicos (*test de loops*, *test remotos*), al igual que pruebas de asequibilidad con paquetes ICMP (*ping* o *traceroute*).
- Establecer opciones para valores de umbral.
- Activación de reinicios planificados (dirigidos a puertos, grupos de puertos o componentes específicos).
- Disponibilidad de sistemas para realizar tests especiales (osciloscopios, reflectómetros, probadores de interfaces, analizadores de protocolos, monitoreadores de *hardware* para supervisión de líneas).
- Soporte de mecanismos de filtraje para mensajes de fallas o alarmas y correlación de eventos para reducir el número de eventos relevantes y para análisis de las causas de los problemas.
- Interfaces de herramientas de administración de fallas a sistemas de etiquetación de problemas y soporte técnico o ayuda al usuario (*help desk*); es decir, propagación automática de notificaciones y correcciones de fallas.

1.2.3 Contabilidad (administración de los usuarios)

La administración de los usuarios comprende tareas como la gestión de los nombres y las direcciones, servicios relacionados con directorios y permisos para el uso de los recursos, así como de contabilidad (figura 1.4), pues existen costos asociados al proporcionar servicios de comunicaciones que deben ser informados a los usuarios (cargos de acceso y de utilización).

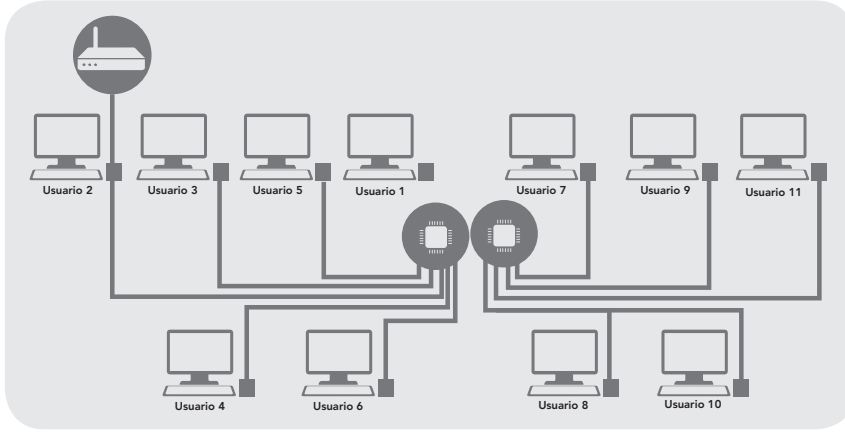


Figura 1.4 Representación de la administración de los usuarios en una red de computadoras.

Las estrategias y procedimientos para la asignación de costos no pueden ni deben ser establecidos rígidamente por un sistema de contabilidad; además, es importante que su administración sea capaz de ajustarse a los lineamientos de una política dada (Chen y Nahrstedt, 1998).

La administración de la contabilidad también incluye la recopilación y cuidado de estadísticas de uso, definición de unidades de contabilización, asignación de cuentas y costos, mantenimiento de bitácoras de contabilidad, establecimiento y monitoreo de las cuotas concedidas y, finalmente, definición de políticas de contabilidad y tarifas que permitirán generar facturas y cargos a los usuarios. Si varios proveedores están involucrados en la prestación de los servicios, las reglas de conciliación también pertenecen a la administración de la contabilidad. Este proceso puede realizarse repartiendo ingresos mediante una tarifa plana o con un precio para cierta unidad de tráfico.

Cómo se implemente el sistema de contabilidad, qué enfoque se utilizará para recopilar los parámetros de contabilidad y cómo serán distribuidos los costos, es una decisión administrativa que puede ser influenciada por políticas de la organización. Entre más sutil sea el enfoque, más complicado e intensivo en costos es el procedimiento de contabilidad.

Los parámetros "de uso" utilizados para calcular los costos incluyen la cantidad de paquetes o *bytes* transmitidos, la duración de la conexión, el ancho de banda, la calidad del servicio o Q_OS (*Quality of Service*), la conexión, la localización de los participantes en la comunicación, los costos para servicios de conversión de protocolos vía una pasarela (*gateway*), el uso de recursos en los servidores y el de productos de *software* (control de licencias). Además de los costos variables, también se tienen en cuenta los fijos (espacio de oficina, mantenimiento, depreciación de muebles y equipos, etcétera).

1.2.4 Desempeño

En términos de los objetivos (muy concretos y específicos), la administración del desempeño se entiende como una continuación sistemática de la administración de fallas: mientras que ésta última es la responsable de asegurar que una red de comunicaciones o un sistema distribuido solamente opere, la segunda busca que el sistema, como un todo, se comporte bien; es decir, se enfoca en la calidad del servicio (figura 1.5) (Clark, Shenker y Zhang, 1992).

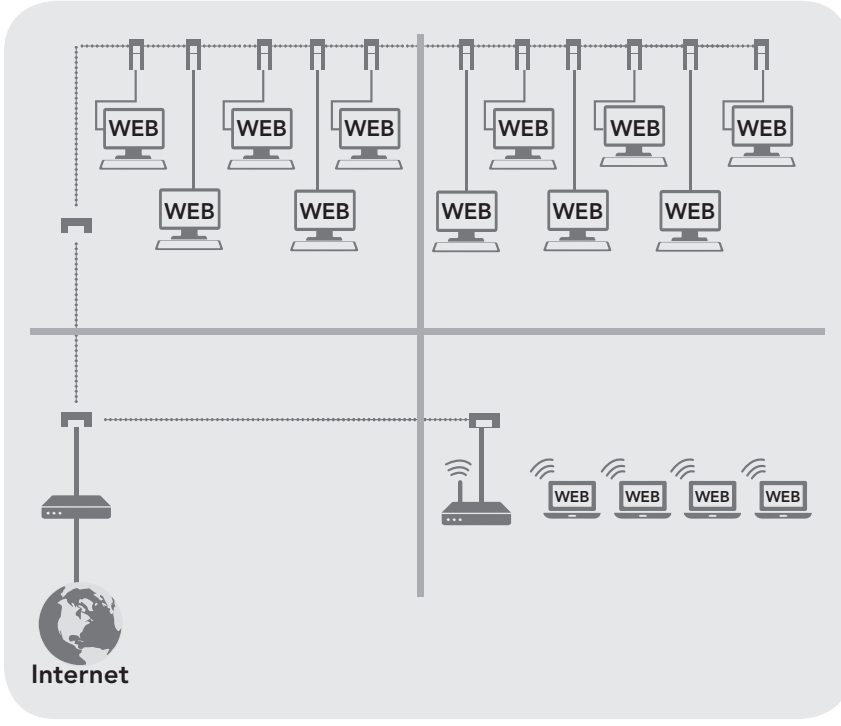


Figura 1.5 Administración del desempeño de una red de computadoras

La calidad del servicio (Q_OS) es un mecanismo para establecer una interfaz que permita "negociar" entre el proveedor y el cliente. Su importancia se incrementa a medida de que más relaciones entre estos se involucran en la implementación de redes corporativas o de sistemas distribuidos (Tompros y Denazis, 2007). Entonces, la interfaz de servicios es definida por las siguientes características:

- Especificación del servicio y del tipo de éste (determinístico, estadístico, el mejor posible).
- Definición de los parámetros de Q_OS relevantes con valores cuantificables incluyendo los de uso, promedio y límite; entre muchos otros.
- Establecimiento de las operaciones de monitoreo.
- Descripción de reacciones a cambios de los parámetros de Q_OS.

Sin embargo, es difícil y no siempre posible proporcionar una definición completa de una interfaz de servicios sobre la base de lo escrito antes (Salazar *et al.*, 2002) que tienden a aparecer las siguientes situaciones:

Problemas en el mapeo vertical de Q_OS. Gracias a que los sistemas de comunicación son sistemas por capas, los parámetros de Q_OS específicos a la capa N tienen que ser mapeados de dicha manera: (N+1) o (N-1). Por ejemplo, un parámetro de la Q_OS orientada a la aplicación (en este caso, calidad de transmisión de voz), se debe mapear a la Q_OS dependiente de la red (*jitter*). Las jerarquías de Q_OS aún no están definidas por completo para todos los protocolos o servicios. Esta situación se exacerba cuando los servicios de diferentes capas se proveen por distintos proveedores (*carriers*).

Problemas en el mapeo horizontal de Q_OS. Si más de un proveedor participa en una red corporativa, el resultado puede ser la concatenación de diferentes subredes o secciones de troncales, que son utilizadas para proveer un servicio con una calidad uniforme entre dos usuarios. Se asume que cada *carrier* tiene implementadas las mismas características de calidad de servicio o, al menos, utiliza protocolos de negociación de Q_OS, de reservación de recursos o administración estandarizados. Cuanto más complejo sea el servicio, menos probable es que se reúnan estos requerimientos.

Métodos de medición. La forma óptima para calcular la Q_OS sería aplicar métodos de medida basados en cantidades visibles en la interfaz de servicio antes que utilizar un análisis de la tecnología suministrada por el proveedor, pues ésta puede cambiar rápidamente y, además, las cantidades medidas no son de interés del cliente; por ello, para él deben ser convertidas a los parámetros de Q_OS.

En resumen, la administración del desempeño incorpora todas las medidas requeridas para asegurar que la calidad del servicio (Q_OS) cumpla con un contrato de nivel de servicios (Bhatti y Crowcroft, 2000), esto incluye:

- Establecer métricas y parámetros de calidad de servicio.
- Monitorear todos los recursos para detectar posibles y reales "cuellos de botella" en el desempeño, así como traspasos de los umbrales.
- Realizar medidas y análisis de tendencias para predecir fallas que puedan ocurrir.
- Evaluar bitácoras históricas (*logs*).
- Procesar los datos medidos y elaborar reportes de desempeño.
- Llevar a cabo planificación de desempeño y de capacidad (*capacity planning*), lo cual implica proporcionar modelos de predicción, simulados o analíticos, utilizados para verificar los resultados de nuevas aplicaciones, mensajes de afinamiento y cambios de configuración.

1.2.5 Seguridad

La seguridad se relaciona con los sistemas distribuidos y los recursos de la compañía que "vale la pena" proteger como la información, la infraestructura de tecnologías de la información (ITI) y los servicios, los cuales se encuentran expuestos a amenazas o al uso inadecuado (Berghel, 2001). Las medidas de seguridad son necesarias para prevenir

daños y pérdidas, así como para establecer acciones que busquen "enfrentar" las amenazas y riesgos obtenidos como resultado de los análisis de seguridad del sistema en red (Anderson, 2008); los cuales, por lo regular, derivan de:

Ataques pasivos. Robo de información llevado a cabo de forma oculta y a través del análisis de tráfico de la red.

Ataques activos. Accesos no autorizados por medio de elementos como virus, tras lo que se lleva a cabo, por ejemplo, suplantación de usuarios, traducida en una manipulación de secuencias de mensajes, manejo de los recursos al recargar su uso, reconfiguración no autorizada, reprogramación de sistemas, etcétera.

Mal funcionamiento de los recursos.

Comportam**iento inapropiado o deficiente y respuestas de operación incorrectas.**

Las metas y los requerimientos de seguridad se establecen a partir de análisis de amenazas y de los valores que necesitan protección (Berghel, 2001). Las políticas de seguridad definidas podrán identificar los requerimientos de seguridad; algunos ejemplos son:

- Las contraseñas (*passwords*) deben ser cambiadas cada tres semanas.
- Sólo los gerentes de segunda línea tienen acceso a los datos del personal.
- Todos los ataques que afecten la seguridad del sistema serán registrados y se les hará un seguimiento.

Dichas políticas sirven como marco de referencia para los servicios de seguridad necesarios y su implementación; por ello, la administración de seguridad comprende los siguientes puntos:

- Realizar un análisis de las posibles y probables amenazas.
- Definir e implementar políticas reales de seguridad.
- Verificación de la identidad (autenticación basada en firmas digitales y en la certificación).
- Establecer e implementar controles de acceso.
- Garantizar la confidencialidad (encriptación).
- Asegurar la integridad de los datos (autenticación de mensajes).
- Monitoreo de los sistemas para prevenir amenazas de seguridad.
- Elaborar reportes sobre el estado de la seguridad y su vulneración o de sus intentos.

Podría asumirse que un conjunto de procedimientos de seguridad reconocidos, cuya mayor parte está disponible como software de dominio público, ya existe en el área de administración de seguridad. El principal problema es encontrar la forma correcta de incorporarlos a una arquitectura de la administración y de controlarlos de una manera uniforme dentro del marco de las políticas de seguridad.

1.3 Servicios de una red de computadoras

La finalidad de los servicios de una red de computadoras, es que los usuarios de los sistemas informáticos de una organización, puedan hacer un mejor uso de los mismos, enriqueciendo de este modo el rendimiento global de la organización. Así, las organizaciones obtienen una serie de ventajas del uso de las redes en los entornos de trabajo como son:

- Mayor facilidad de comunicación.
- Mejora de la competitividad.
- Mejora de la dinámica de grupo.
- Reducción del presupuesto para el procesamiento de los datos, y convertirlos en información útil, para apoyar la toma de decisiones. Así como, para convertirse en conocimiento.
- Reducción de los costos de proceso por usuario.
- Mejoras en la administración de los paquetes y de los programas.
- Mejoras en la integridad de los datos.
- Mejora en los tiempos de respuesta.
- Flexibilidad en el proceso de datos.
- Mayor variedad en el uso de paquetes y de programas.
- Mayor facilidad de uso.
- Mejor y mayor seguridad.

Para que todo esto sea posible, la red debe prestar una serie de servicios fundamentales a los usuarios como:

- Acceso
- Ficheros
- Impresión
- Correo
- Información
- Otros (a especificar)

Para la prestación de los servicios de la red, se requiere que existan sistemas en la red con capacidad para actuar como servidores. Los servidores y los servicios de red, se basan en los sistemas operativos de la red (los cuales pueden ser centralizados o distribuidos). Un sistema operativo de red, es un conjunto de programas que permiten y controlan el uso de dispositivos de red por múltiples usuarios. Estos programas interceptan las peticiones de servicio de los usuarios, y las dirigen a los equipos servidores adecuados. Por ello, el sistema operativo de red, le permite a ésta, ofrecer capacidades de multiproceso y multiusuario.

1.3.1 DHCP

El protocolo de configuración dinámica de *host* o DHCP (*Dynamic Host Configuration Protocol*) permite a los clientes de una red IP obtener sus parámetros de configuración automáticamente. Se trata de un protocolo de tipo cliente/servidor que por lo regular el servidor posee una lista de direcciones IP dinámicas que se asignan a los clientes, sabiendo en todo momento quién ha estado en posesión de cada una, cuánto tiempo la ha tenido y a quién ha sido concedida después. DHCP se estableció como una extensión del protocolo *Bootstrap* (BOOTP), el cual fue extendido porque se requería intervención manual para completar la información de configuración en cada cliente sin proporcionar un mecanismo para la recuperación de las direcciones IP en desuso. En la actualidad, éste se mantiene como el estándar para redes IPv4.

En octubre de 1993, este protocolo se publicó por primera y su implementación actual se encuentra en el estándar RFC 2 131, aunque de forma original, se definió bajo 1 531; para DHCPv6 se publica el estándar RFC 3 315 (figura 1.6), aunque 3 633 le añadió un mecanismo de delegación de prefijo (Joel, 2002).

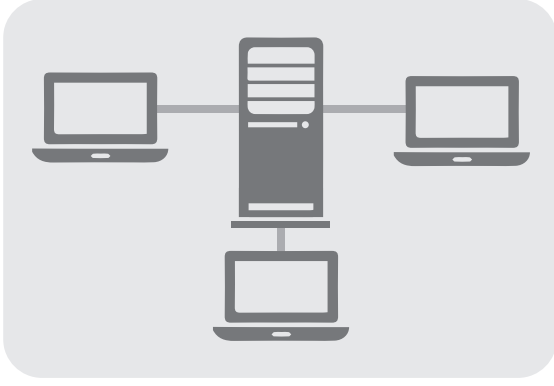


Figura 1.6 Importancia de integrar un servidor DHCP en una red de computadoras

DHCPv6 se amplió aún más para proporcionar información a los clientes con la configuración automática de direcciones sin estado en el RFC 3 736. El protocolo BOOTP, a su vez, fue definido por primera vez en el 951 como un reemplazo para la resolución de direcciones inversa o RARP (*Reverse Address Resolution Protocol*). La principal motivación para dicha sustitución fue que este último era un protocolo de la capa de enlace de datos, lo cual hizo más difícil su aplicación en muchas plataformas de servidores y requería uno presente en cada enlace de red individual.

BOOTP introdujo la innovación de un agente de retransmisión, lo que permitió el envío de paquetes BOOTP fuera de la red local utilizando enrutamiento IP estándar, por lo que un servidor central podría servir de anfitrión en muchas subredes IP (Croft, 2005).

El DHCP permite al administrador supervisar y distribuir de forma centralizada las direcciones IP necesarias, así como asignar y enviar en automático una nueva, si el dispositivo es conectado en un lugar diferente de la red (Perkins, 2002). El protocolo DHCP incluye tres métodos de asignación de direcciones IP:

Asignación manual o estática. Se proporciona una dirección IP a una máquina determinada; suele utilizarse cuando se quiere controlar la asignación de dirección IP a cada cliente y, a la par, evitar que se conecten clientes no identificados.

Asignación automática. Determina una dirección IP a una máquina cliente la primera vez que hace la solicitud al servidor DHCP y hasta que el cliente la libera. Se utiliza cuando el número de clientes no varía demasiado.

Asignación dinámica. Es el único método que permite la reutilización dinámica de las direcciones IP. El administrador de la red determina un rango de éstas y cada dispositivo conectado a la red está configurado para solicitar la suya al servidor cuando la tarjeta de interfaz de red se inicializa. El procedimiento usa un concepto muy simple en un intervalo de tiempo controlable; esto facilita la instalación de nuevas máquinas-clientes a la red.

Algunas implementaciones de DHCP pueden actualizar el sistema de nombres de dominio o DNS (*Domain Name System*), asociado con los servidores para reflejar las nuevas direcciones IP mediante el protocolo de actualización de DNS establecido en RFC 2 136 (versión en inglés). Dichas opciones configurables están definidas en RFC 2 132 (versión en inglés) y se presentan a continuación:

- Dirección del servidor y nombre DNS.
- Puerta de enlace de la dirección IP.
- Dirección de publicación masiva (*broadcast address*).
- Máscara de subred.
- Tiempo máximo de espera del protocolo de resolución de direcciones o ARP (*Adress Resolution Protocol*).
- Unidad de transferencia máxima o MTU (*Maximum Transmission Unit*) para la interfaz.
- Dominios y servidores de servicio de información de red o NIS (*Network Information Service*).
- Servidores de protocolo de tiempo de red o NTP (*Network Time Protocol*).
- Servidor SMTP.
- Servidor TFTP.
- Nombre del servidor WINS (*Windows Internet Naming Service*).

1.3.2 DNS

Al explicar las características del DNS, se debe establecer que un nombre de dominio consiste en dos o más partes, técnicamente etiquetas, separadas por puntos cuando son escritas en forma de texto. Por ejemplo, el punto final separa la etiqueta de la raíz de la jerarquía, aunque por lo general, se omite por ser puramente formal. Una muestra de nombre de dominio correcto o FQDN *(Fully Qualified Domain Name*) es "www.ejemplo.com", figura 1.7.

Figura 1.7 Símbolo universal de DNS

A la etiqueta ubicada más a la derecha se le llama dominio de nivel superior (TLD, *Top Level Domain*), como ".com" en www.ejemplo.com u ".org" en es.wikipedia.org. Cada etiqueta

a la izquierda especifica una subdivisión o subdominio.[6] Finalmente, la parte más a la izquierda expresa el nombre de la máquina (*hostname*); el resto sólo determina la manera de crear una ruta lógica a la información requerida. Por ejemplo, el dominio es.wikipedia.org tendría el nombre de la máquina "es", aunque en este caso no se refiere a una máquina física en particular. En teoría, esta subdivisión puede tener hasta 127 niveles, y es posible que cada etiqueta contenga hasta 63 caracteres restringidos para que la longitud total del nombre del dominio no exceda los 255, aunque en la práctica los dominios son casi siempre cortos; estos deben comenzar con una letra[7] y tienen "-" como único símbolo permitido.

Una vez establecido esto, cabe mencionar que el sistema de nombres de dominio (DNS, *Domain Name System*) nació de la necesidad de recordar fácilmente los nombres de todos los servidores conectados a Internet, pues la empresa estadounidense SRI International (antes sólo SRI o *Stanford Research Institute*) alojaba un archivo llamado hosts, que contenía todos los nombres de dominio conocidos, el crecimiento explosivo de la red causó que esto no resultara práctico, por lo que en 1983, Paul V. Mockapetris publicó los RFC 882 y 883 definiendo lo que hoy en día ha evolucionado hacia el DNS moderno.

El DNS se asocia a información variada con nombres de dominios asignados a cada uno de los participantes. Su función más importante es traducir nombres inteligibles para las personas en identificadores binarios asociados con los equipos conectados a la red y enfocado el propósito de localizarlos y direccionarlos a todo el mundo (Anderson, 2008). Otro de sus usos más comunes son la asignación de nombres de dominio a direcciones IP y la localización de los servidores de correo electrónico de cada uno. Por ejemplo, si la dirección IP del sitio FTP (*File Transfer Protocol*) de prox.mx es 200.64.128.4, la mayoría de la gente llega a este equipo especificando *ftp.prox.mx* y no la IP, pues, además de ser más fácil de recordar, es más fiable; sin embargo, la dirección numérica podría cambiar por muchas razones, sin que tenga que modificarse el nombre.

Para la operación práctica del sistema DNS se utilizan tres componentes principales:

Clientes fase 1. Programa cliente que se ejecuta en la computadora del usuario y que genera peticiones de resolución de nombres a un servidor DNS.

Servidores DNS. contestan las peticiones de los clientes. Los servidores recursivos tienen la capacidad de reenviar la petición a otro servidor si no disponen de la dirección solicitada.

Zonas de autoridad. porciones del espacio de nombres raros de dominio que almacenan los datos. Cada zona de autoridad abarca al menos un dominio y sus subdominios.

En el mundo real el DNS funciona de la siguiente forma: los usuarios, por lo general, no se comunican de manera directa con el servidor, sino que la resolución de nombres se hace de forma transparente por las aplicaciones del cliente (por ejemplo, navegadores, clientes de correo y otras usadas en Internet).

La mayoría de usuarios domésticos utilizan como servidor DNS el proporcionado por su proveedor de Internet, donde la dirección de servidores puede ser ajustada de forma manual o automática mediante DHCP. En otros casos, los administradores de red tienen configurados sus propios servidores DNS. En cualquiera de las dos situaciones los servidores DNS que reciben la petición buscan en primer lugar verificar si disponen de la

[6] El "subdominio" expresa dependencia relativa, no dependencia absoluta.

[7] Véase la RFC 1 035, sección 2.3.1 *Preferencia nombre de la sintaxis*.

respuesta en la memoria caché, pues de ser así, la sirven; en caso contrario, iniciarían la búsqueda de manera recursiva. Una vez encontrada la respuesta, el servidor DNS guardará el resultado en su memoria caché para futuros usos y devuelve el resultado.

1.3.3 Telnet

Telnet (*Telecommunication Network*) es el nombre de un protocolo de red que permite viajar a otra máquina para manejarla de manera remota como si se estuviera sentado delante de ella; también se trata del programa informático que implementa el cliente. Para que la conexión funcione, como en todos los servicios de Internet, la máquina a la que se accede debe tener un programa especial que reciba y gestione las conexiones (Braden, 1989). El puerto que se utiliza de manera común es 2,3 (figura 1.8).

Figura 1.8 Funcionamiento de Telnet

Telnet sólo sirve para acceder en modo terminal; es decir, sin gráficos, pero es una herramienta muy útil para arreglar fallas a distancia y para consultar datos personales en máquinas accesibles por la red. Además, se ha utilizado (y aún hoy se puede ocupar en su variante SSH) para abrir una sesión con una máquina UNIX, de modo que múltiples usuarios se conecten, abran sesión y puedan trabajar a través de ésta.

A pesar de sus beneficios, el mayor problema de Telnet es la seguridad porque todos los nombres de usuario y contraseñas necesarias para entrar en las máquinas viajan por la red como texto plano, lo cual facilita que cualquiera que espíe el tráfico pueda obtener dichos datos; por esta razón, dejó de usarse Telnet hace unos años cuando apareció y se popularizó el SSH. Algunos clientes de Telnet son mTelnet, NetRunner, Putty o Zoc.

Hoy en día, este protocolo también se usa para llegar al sistema de tablón de anuncios o BBS (*Bulletin Board System*) que al principio era accesible con sólo un módem a través de la línea telefónica. Para acceder a un BBS mediante Telnet es necesario un cliente que proporcione soporte a gráficos ANSI y los protocolos de transferencia de ficheros.[8]

[8] El protocolo de transferencia de ficheros más común con mejor funcionamiento es llamado ZModem.

En 1969 se desarrolló Telnet, la mayoría de los usuarios de computadoras en la red estaban en los servicios informáticos de instituciones académicas o en grandes instalaciones de investigación privadas y de gobierno. En este ambiente, la seguridad no fue una preocupación hasta después de la explosión del ancho de banda de los años 90 del siglo XX, pues con la subida exponencial del número de gente con acceso a Internet y, por ende, el aumento en usuarios que procuran ingresar maliciosamente a los servidores de otras personas, Telnet dejó de ser recomendado en redes con conectividad a Internet. En resumen, se hace necesario destacar tres razones principales por las cuales Telnet no se recomienda para los sistemas modernos desde el punto de vista de la seguridad:

- Los dominios de uso general de Telnet tienen varias vulnerabilidades descubiertas a lo largo de los años y varias más que aún podrían existir.
- Telnet, por defecto, no cifra ninguno de los datos enviados sobre la conexión (contraseñas inclusive), así que es fácil interferir y grabar las comunicaciones con utilidades comunes como *Tcpdump* y *Wireshark*, dando como resultado el uso de la información para propósitos maliciosos.
- Telnet carece de un esquema de autentificación que permita asegurar que la comunicación esté siendo realizada entre los dos anfitriones deseados y no interceptada entre ellos.

Estos defectos han causado el abandono y depreciación del protocolo Telnet rápidamente a favor de uno más seguro y funcional como SSH lanzado en 1995. Éste proporciona toda la utilidad presente en Telnet, pero agregó el cifrado fuerte para evitar que los datos sensibles sean interceptados y la autentificación mediante llave pública para asegurar que el equipo remoto sea realmente el autorizado. Con base en esto, los expertos en seguridad computacional como el Instituto SANS (*SysAdmin Audit, Networking and Security Institute*) y los miembros del *newsgroup* de *comp.os.linux.security* recomiendan que el uso de Telnet para las conexiones remotas sea descontinuado bajo cualquier circunstancia normal (Braden, 1989).

1.3.4 SSH

SSH (*Secure Shell*) o intérprete de órdenes seguro, es el nombre de un protocolo y del programa que lo implementa, el cual sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos, así como redirigir el tráfico de "X" para ejecutar programas gráficos si se tiene un servidor "Y" corriendo en sistemas Unix y Windows.

Además, SSH permite copiar datos de forma segura, como simular sesiones FTP cifradas y gestionar claves RSA[9] (*Rivest, Shamir and Adleman*). Es el primero y más utilizado algoritmo de este tipo, válido tanto para cifrar como para firmar digitalmente.

La seguridad de éste radica en el problema de la factorización de números enteros, pues los mensajes enviados se representan mediante números y el funcionamiento se basa en el producto conocido de dos números primos grandes elegidos al azar y mantenidos en secreto para no escribir claves al conectar a los dispositivos y pasar los datos de cualquier otra aplicación por un canal seguro tunelizado mediante SSH (figura 1.9). En la actualidad, estos primos son del orden de 10^{200} y se prevé que dicho tamaño crezca con el aumento de la capacidad de cálculo de las computadoras.

[9] Sistema criptográfico de clave pública desarrollado en 1977.

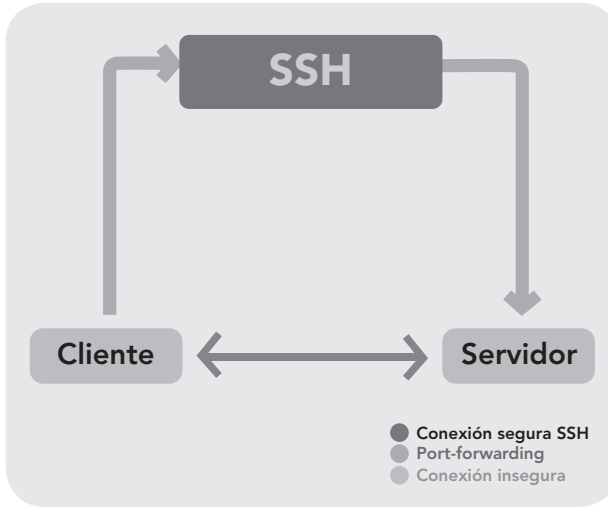


Figura 1.9 Funcionamiento de SSH

Al principio, sólo existían los *r-commands*, que se basaban en el programa *rlogin*, el cual funciona de forma similar a Telnet. La primera versión del protocolo y el programa eran libres y los creó un finlandés llamado Tatu Ylönen, pero su licencia fue cambiando y terminó apareciendo la compañía *SSH Communications Security,* que lo ofrecía de manera gratuita para uso doméstico y académico, pero exigía el pago a otras empresas. En 1997, (dos años después de que se creara la primera versión), se propuso como borrador en la IETF (*Internet Engineering Task Force*), en español llamada Grupo de Trabajo de Ingeniería Abierta (Abransom, 2000).

Por otro lado, a principios de 1999, se empezó a escribir una versión que se convertiría en la implementación libre por excelencia llamada OpenSSH de OpenBSD. En la actualidad existen dos versiones de SSH: la primera utiliza muchos algoritmos de cifrado patentados y es vulnerable a un agujero de seguridad que potencialmente permite a un intruso insertar datos en la corriente de comunicación. La segunda versión suite OpenSSH, bajo Red Hat Enterprise Linux que tiene un algoritmo de intercambio de claves mejorado que no es vulnerable al agujero de seguridad de la versión 1; sin embargo, ésta también soporta las conexiones de la primera.

1.3.5 FTP y TFTP

FTP (*File Transfer Protocol*) es un protocolo de red de transferencia de archivos entre sistemas conectados a una red TCP (*Transmission Control Protocol*) basado en la arquitectura cliente-servidor, independiente al sistema operativo utilizado en cada equipo. El servicio FTP es ofrecido por la capa de aplicación del modelo de capas de red TCP/IP al usuario, utilizando los puertos de red 20 y 21.

Un problema básico de FTP es que está pensado para ofrecer la máxima velocidad en la conexión, pero no la máxima seguridad porque todo el intercambio de información (desde el *login* y el password del usuario en el servidor hasta la transferencia de cualquier archivo) se realiza en texto plano sin ningún tipo de cifrado con lo que un posible atacante

puede capturar este tráfico, acceder al servidor y/o apropiarse de los archivos transferidos. Para solucionar dicha situación son de gran utilidad aplicaciones como SCP y SFTP incluidas en el servicio SSH y que permiten transferir archivos cifrando toda la actividad. En el modelo, el intérprete de protocolo de usuario inicia la conexión de control en el puerto 21 (Tanenbaum, 2006). La figura 1.10 muestra la operación de FTP.

Figura 1.10 Operación de FTP

El funcionamiento de FTP es el siguiente: las órdenes estándar las genera la IP de usuario y se transmiten al proceso del servidor a través de la conexión de control. Después, las respuestas estándar se envían desde la IP del servidor al usuario por la conexión de control como respuesta a las órdenes, las cuales especifican parámetros como puerto, modo de transferencia, tipo de representación y estructura; así como la naturaleza de la operación sobre el sistema de archivos como almacenar, recuperar, añadir, borrar, etcétera.

El proceso de transferencia de datos (DTP, *Dynamic Trunking Protocol*) de usuario, u otro en su lugar, debe esperar a que el servidor inicie la conexión al puerto de datos especificado para transferirlos en función de los parámetros determinados.

También hay que destacar que la conexión de datos es bidireccional; es decir, que se puede usar simultáneamente para enviar y recibir. Ésta al principio tenía un problema: la localización de los servidores en la red; es decir, el usuario que quería descargar algún archivo mediante FTP debía conocer en qué máquina estaba ubicado. En ese momento, la única herramienta de búsqueda de información que existía era *Gopher* (o sea, literalmente, "lanzarse sobre" la información) con todas sus limitaciones: se trata de un servicio cuyo objetivo es la localización de archivos a partir del título, el cual consiste en un conjunto de menús de recursos ubicados en diferentes máquinas intercomunicadas. Cada una sirve un área de información, pero la organización interna permite que todas ellas funcionen como si se tratase de una sola. De acuerdo con esto, el usuario navega a través de los menús hasta localizar la información buscada y desconoce exactamente de qué máquina se está descargando. Con la llegada de Internet, los potentes motores de búsqueda como *Google* dejaron de lado el servicio *Gopher* y la localización de los servidores FTP pasó a ser un problema olvidado.

Algunos clientes de FTP básicos en modo consola vienen integrados en los sistemas operativos incluyendo Microsoft Windows, DOS, GNU/Linux y Unix; sin embargo, hay clientes disponibles con opciones añadidas e interfaz gráfica que más confiables a la hora de conectarse con servidores FTP no anónimos (Balacheff *et al.*, 2003), pues los anónimos

ofrecen sus servicios libremente a todos los usuarios y permiten acceder a sus archivos sin necesidad de tener un nombre de usuario o una cuenta, aunque se tendrá que introducir una contraseña momentánea (que es la dirección de correo electrónico propia); empero, la conexión se realiza con menos privilegios que un usuario normal: sólo se podrán leer y copiar los archivos públicos, así indicados por el administrador. Por otro lado, se utiliza un servidor FTP anónimo para depositar grandes archivos que no tienen utilidad y se reservan los servidores de páginas web para almacenar información textual destinada a la lectura en línea.

Si se desea tener privilegios como acceso a cualquier parte del sistema de archivos, modificación de los existentes y la posibilidad de subir los propios, por lo general se realiza mediante una cuenta de usuario: en el servidor se guarda la información de las distintas cuentas que pueden acceder a él, de forma que para iniciar una sesión FTP se debe introducir una autentificación (*login*) y una contraseña (*password*) que identifica al cliente/usuario unívocamente.

Por otro lado, al disponer de un cliente FTP basado en web se puede acceder al servidor FTP remoto como si se estuviera realizando cualquier otro tipo de navegación web: se podrán crear, copiar, renombrar y eliminar directorios; igualmente, existirá la posibilidad de cambiar permisos, editar, ver, subir y descargar archivos, así como cualquier otra función del protocolo FTP que el servidor remoto permita. Sin embargo, la entrada sin restricciones que proporcionan las cuentas de usuario implica problemas de seguridad, lo que ha dado lugar a un tercer tipo de acceso FTP denominado invitado (*guest*), que se puede considerar como la mezcla de los dos anteriores (McConell, 1996). La idea de este mecanismo es la siguiente: se trata de permitir que cada usuario se conecte a la máquina mediante su *login* y *password*, pero evitando que tenga acceso a partes del sistema de archivos que no necesita para realizar su trabajo; de esta forma, llegará a un entorno restringido, algo muy similar a lo que sucede en los accesos anónimos, pero teniendo más privilegios.

FTP admite dos modos de conexión del cliente denominados activo o estándar (envía comandos tipo "PORT" al servidor por el canal de control al establecer la conexión) y pasivo (envía comandos tipo "PASV"). En ambos modos, el cliente lleva a cabo una conexión con el servidor mediante el puerto 21, que establece el canal de control. Pero en modo activo, el servidor siempre crea el canal de datos en el puerto 20, mientras que en el lado del cliente el canal de datos se asocia a un puerto aleatorio mayor que 1 024. Para ello, el cliente manda un comando "PORT" al servidor por el canal de control indicándole el número de puerto, de manera que aquél pueda abrirle una conexión de datos por donde se transferirán los archivos y los listados en el puerto especificado (Stallings, 2005).

Lo anterior presente una grave problemática de seguridad y es que la máquina cliente debe estar dispuesta a aceptar cualquier conexión de entrada en un puerto superior al 1 024, con los problemas que ello implica si se tiene el equipo conectado a una red insegura como Internet. De hecho, los cortafuegos que se instalen en el equipo para evitar ataques seguramente rechazarán esas conexiones aleatorias; para solucionar el problema anterior, se desarrolló el modo pasivo: cuando el cliente envía un comando "PASV" sobre el canal de control, el servidor FTP indica el puerto al que debe conectarse el cliente (mayor al 1 023; por ejemplo, 2 040); éste inicia una conexión desde el puerto que le sigue al puerto de control (por ejemplo, 1 036) hacia el del servidor especificado anteriormente (2 040).

Antes de cada nueva transferencia, tanto en el modo activo como en el pasivo, el cliente debe enviar otra vez un comando de control, tras lo que el servidor recibirá esa conexión de datos en un nuevo puerto aleatorio (si está en modo pasivo) o por el puerto 20 (si está en modo activo).

Por otro lado, en el protocolo FTP existen dos tipos de transferencia: en ASCII y binarios. Es muy importante conocer estos, pues si no se utilizan las opciones adecuadas, se puede destruir la información del archivo. Por ello, al ejecutar la aplicación FTP se debe utilizar uno de estos comandos (o poner la correspondiente opción en un programa con interfaz gráfica).

Tipo ASCII. Adecuado para transferir archivos que sólo contengan caracteres imprimibles (los archivos ASCII no son archivos resultantes de un procesador de texto); por ejemplo, páginas HTML (*HyperText Markup Language*), pero sin las imágenes que puedan contener.

Tipo binario. Usado cuando se trata de archivos comprimidos, ejecutables para computadoras personales, imágenes y archivos de audio, entre otras aplicaciones.

Por otro lado, TFTP son las siglas en inglés de *Trivial File Transfer Protocol*, que significa protocolo de transferencia de archivos trivial; el cual consiste en una versión básica de FTP. Sin embargo, TFTP a menudo se utiliza para enviar pequeños archivos entre computadoras en una red, algunas de sus características son:

- Utiliza UDP (en el puerto 69) como protocolo de transporte.
- No puede listar el contenido de los directorios.
- No existen mecanismos de autenticación o cifrado.
- Se utiliza para leer o escribir archivos de un servidor remoto.
- Soporta tres modos diferentes de transferencia: netascii, octet y mail, donde los dos primeros corresponden a los modos ASCII e imagen (binario) del protocolo FTP.

Puesto que TFTP utiliza UDP, no hay una definición formal de sesión, cliente y servidor, aunque se considera servidor a aquel que abre el puerto, y cliente a quien se conecta; sin embargo, cada archivo transferido vía TFTP constituye un intercambio independiente de paquetes, en el cual existe una relación cliente-servidor informal: la máquina A, que inicia la comunicación, envía un paquete de petición de lectura (RRQ, *Read Request*) o de escritura (WRQ, *Write Request*) a la máquina B con el nombre del archivo y el modo de transferencia; posteriormente, esta última responde con un paquete de confirmación (ACK, *Acknowledgement*), que también sirve para informar acerca del puerto al que tendrá que enviar los paquetes restantes la B. La máquina origen envía todos los datos numerados a la de destino, excepto el último, que contiene 512 bytes de datos; entonces, ésta responde con paquetes ACK numerados para todos los paquetes de datos, donde el último debe contener menos de 512 bytes de datos. Si el tamaño del archivo transferido es un múltiplo exacto de este valor, obtiene como respuesta del origen el envío de un paquete final que contiene 0 bytes de datos.

1.3.6 WWW: HTTP y HTTPS

WWW

En informática, la red informática mundial (WWW, *World Wide Web*), comúnmente conocida como web, es un sistema de distribución de documentos de hipertexto o hipermedios interconectados y accesibles vía Internet (figura 1.11). Con un navegador web, un usuario visualiza sitios compuestos de páginas que pueden contener texto, imágenes, videos u otros contenidos multimedia; además, puede navegar a través de estos usando hiperenlaces.

Entre marzo de 1989 y diciembre de 1990, el británico Tim Berners-Lee con la ayuda del belga Robert Cailliau desarrollaron la web por mientras trabajaban en el Centro Europeo de Investigaciones Nucleares (CERN) en Ginebra, Suiza; y se publicó el 7 de agosto de 1991. Berners y Cailliau propusieron utilizar el hipertexto "para vincular y acceder a información de diversos tipos como una red de nodos en que el usuario puede navegar a voluntad."

Figura 1.11 Símbolo de la web

El funcionamiento de la WWW es el siguiente: el primer paso consiste en traducir la parte del nombre del servidor de la URL en una dirección IP usando DNS. Lo siguiente es enviar una petición HTTP al servidor web solicitando el recurso. En el caso de una página web típica, primero se solicita el texto HTML, que de inmediato es analizado por el navegador, el cual hace otras peticiones para los gráficos y otros ficheros que formen parte de la página: al recibir desde el servidor web lo que se ha solicitado, el navegador busca y establece la página y cómo se describe en el código HTML, el CSS y otros lenguajes web. Al final, se incorporan las imágenes y otros recursos para producir la página completa que el usuario despliega en la pantalla (Berners-Lee; Cailliau; Loutonen; Nielsen y Secret, 1994).

Por otro lado, la web como se conoce hoy en día, ha permitido un flujo de comunicación global a una escala sin precedentes en la historia humana. Las personas separadas en el tiempo y el espacio pueden usarla para intercambiar o desarrollar sus pensamientos más íntimos o, de manera alternativa sus actividades y deseos cotidianos. Todo puede ser compartido y diseminado digitalmente con el menor esfuerzo posible, haciéndolo llegar casi de forma inmediata a cualquier otro punto del Planeta.

Desde 2007, Berners-Lee dirige el *World Wide Web Consortium* (W3C), el cual desarrolla y mantiene los siguientes y otros estándares que permiten a las computadoras de la web almacenar y comunicar con eficacia los diferentes tipos de información (Berners-Lee, 2009).

- El identificador de recurso uniforme (URI, *Uniform Resource Identifier*), que es un sistema universal para referenciar recursos como páginas web.
- El protocolo de transferencia de hipertexto (HTTP), que especifica cómo se comunican el navegador y el servidor entre ellos.
- El lenguaje de marcas de hipertexto (HTML), usado para definir la estructura y contenido de documentos de este tipo.
- El lenguaje de marcado extensible (XML, *eXtensible Markup Language*), usado para describir la estructura de los documentos de texto.

Aunque la existencia y uso de la web se basa en tecnología material que tiene a su vez algunas desventajas, esta información no utiliza recursos físicos como las bibliotecas o la prensa escrita, razón por la cual su propagación vía Internet no está limitada por el movimiento de volúmenes físicos o por copias manuales o materiales de datos. Gracias a su carácter virtual puede ser buscada fácil, eficaz y más rápido de lo que una persona podría recabarla por sí misma a través de un viaje, correo, teléfono, telégrafo o cualquier otro medio de comunicación.

La web es el medio de mayor difusión de intercambio personal aparecido en la historia de la humanidad, muy por delante de la imprenta (Berners- Lee; Bray; Connolly; Cotton; Fielding; Jeckle; Lilley; Mendelsohn; Orchard; Walsh y Williams, 2004). Como bien se ha descrito en este apartado, el alcance de la red en la actualidad es difícil de cuantificar. En

total, según las estimaciones de 2010, el número total de páginas web (bien de acceso directo mediante URL, o bien a través de enlace) era más de 27 000 millones en ese año; es decir, aproximadamente tres páginas por cada persona viva en el Planeta.

A su vez, la difusión del contenido es que en poco más de 10 años se han codificado medio billón de versiones de la historia colectiva y se han puesto frente a 1 900 millones de personas. Es en definitiva la consecución de una de las mayores ambiciones del hombre: desde la antigua Mongolia, pasando por la Biblioteca de Alejandría o la mismísima Enciclopedia de Rousseau y Diderot, el hombre ha tratado de recopilar en un mismo tiempo y lugar todo el saber acumulado desde sus inicios hasta ese momento; sueño que se ha hecho posible gracias al hipertexto. Además de todo lo reseñado, la red ha propiciado otro logro sin precedentes en la comunicación, como es la adopción de una lengua franca: el inglés, vehículo a través del cual se hace posible el intercambio de información.

HTTP

En cuanto al protocolo de transferencia de hipertexto (HTTP, *Hypertext Transfer Protocol*), se habla del elemento usado en cada transacción de la WWW; éste fue desarrollado por el W3C y la Internet Engineering Task Force, colaboración que culminó en 1999 con la publicación de una serie de RFC, donde el más importante es 2 616 que especifica la versión 1.1 (Fielding, Gettys, Mogul, Frystyk, Masinter y Berners-Lee, 1999). Es decir, HTTP define la sintaxis y la semántica que utilizan los elementos de *software* de la arquitectura web (clientes, servidores, *proxies*) para comunicarse. Este protocolo se orienta a transacciones y sigue el esquema petición–respuesta entre un cliente y un servidor, donde al cliente que efectúa la petición (un navegador web o un *spider*) se le conoce como user agent (agente del usuario). Por otra parte, a la información transmitida se la llama recurso y se le identifica mediante un localizador uniforme de recursos (URL, *Uniform Resource Locator*) que pueden ser archivos, el resultado de la ejecución de un programa, la consulta a una base de datos, la traducción automática de un documento, etcétera. La figura 1.12 muestra el símbolo del HTTP.

Figura 1.12 Símbolo del hipertexto HTTP

HTTP es un protocolo sin estado; es decir, que no guarda ninguna información sobre conexiones anteriores. El desarrollo de las aplicaciones web necesita con frecuencia mantener un estado, para ello se usan las *cookies*, que consiste en la información que un servidor puede almacenar en el sistema cliente por tiempo indeterminado; esto le permite a las aplicaciones web instituir la noción de "sesión" y rastrear usuarios.

Una transacción HTTP está formada por un encabezado seguido, opcionalmente, por una línea en blanco y algún dato. El encabezado es un bloque de datos que precede a la información propiamente dicha, por lo que muchas veces se hace referencia a él como *metadato*. Éste especificará cosas como la acción requerida del servidor, el tipo de dato retornado o el código de estado; su uso le proporciona gran flexibilidad al protocolo, permite que se envíe información descriptiva en la transacción, posibilitando la autenticación, cifrado e identificación de usuario. Si se reciben líneas de encabezado del cliente, el servidor las coloca en las variables de entorno de CGI con el prefijo HTTP_ seguido del nombre, donde un carácter guion (-) se convierte a "_". El servidor puede excluir cualquier encabezado que ya esté procesado, como *Authorization, Content-Type y Content-Length*, aparte de excluir alguno o todos los encabezados si se excede algún límite del entorno de sistema. Ejemplos de esto son las siguientes variables:

HTTP_ACCEPT. Tipos MIME (*Multipurpose Internet Mail Extensions*) que el cliente aceptará, dados los encabezados HTTP; otros protocolos quizás necesiten obtener esta información de otro lugar. Los elementos de esta lista deben estar separados por una coma, como se dice en la especificación HTTP: Tipo, tipo.

HTTP_USER_AGENT. Navegador que utiliza el cliente para realizar la petición. El formato general para esta variable es *software*/versión biblioteca/versión.

Por su parte, el servidor envía al cliente:

- Un código de estado que indica si la petición fue correcta o no: los códigos de error típicos indican que el archivo solicitado no se encontró, que la petición no se realizó de forma adecuada o que se requiere autenticación para acceder al archivo.
- La información propiamente dicha. Como HTTP permite enviar documentos de todo tipo y formato, es ideal para transmitir multimedia como gráficos, audio y video; una de sus mayores ventajas.
- Información sobre el objeto que retorna.

Hay que tener en cuenta que la lista anterior no incluye todos los campos de encabezado y que algunos de ellos sólo tienen sentido en una dirección. Finalmente, HTTP define ocho métodos (algunas veces referidos como "verbos") que indican la acción que se desea efectuar sobre el recurso identificado, el cual a menudo corresponde a un archivo o a la salida de un ejecutable que reside en el servidor como:

HEAD. Pide una respuesta idéntica a la que correspondería a una petición GET, pero sin el cuerpo. Esto es útil para recuperar metainformación escrita en los encabezados de respuesta sin tener que transportar todo el contenido.

GET. Pide una representación del recurso especificado. Por seguridad no debería ser usado por aplicaciones que causen efectos porque transmite información a través de la URI (*Uniform Resource Identifier*) agregando parámetros a la URL. Ejemplo: GET/images/logo.png, donde HTTP/1.1 obtiene un recurso llamado logo.png. Una muestra con parámetros es /index.php?page=main&lang=es.

POST. Envía los datos para que sean procesados por el recurso identificado. Los datos se incluirán en el cuerpo de la petición, lo cual puede resultar en la creación de nuevos recursos, de las actualizaciones de estos o ambas cosas.

PUT. Realiza un *upload* (o carga) de un recurso especificado. Es el camino más eficiente para subir archivos a un servidor; esto porque en POST se utiliza un mensaje multiparte que es decodificado por el servidor. En contraste, el método PUT permite escribir un archivo en una conexión *socket* establecida con el servidor. Su desventaja es que los servidores de *hosting* compartido no lo tienen habilitado. Un ejemplo es PUT/path/filename.html HTTP/1.1.

DELETE. Borra el recurso especificado.

TRACE. Este método solicita al servidor que envíe de vuelta en la sección del cuerpo de entidad de un mensaje de respuesta todos los datos que éste reciba como solicitud. Se utiliza con fines de comprobación y diagnóstico.

OPTIONS. Devuelve los métodos HTTP que el servidor soporta para un URL específico. Esto puede ser utilizado para comprobar la funcionalidad de un servidor web mediante petición en lugar de un recurso determinado.

CONNECT. Se utiliza para saber si se tiene acceso a un *host*, aunque la petición no llega necesariamente al servidor. Este método se utiliza para saber si un *proxy* da acceso a un host bajo condiciones especiales como, por ejemplo, "corrientes" de datos bidireccionales encriptadas (como lo requiere SSL).

HTTPS

El protocolo seguro de transferencia de hipertexto (HTTPS, *Hypertext Transfer Protocol Secure*) se trata de la versión segura de HTTP, que es utilizada principalmente por entidades bancarias, tiendas en línea y cualquier tipo de servicio que requiera el envío de datos personales, contraseñas y realización de operaciones financieras.

El sistema HTTPS utiliza un cifrado basado en SSL/TLS[10] para crear un canal codificado apropiado para el tráfico de información más sensible que el protocolo HTTP, cuyo nivel de encriptación depende del servidor remoto y navegador utilizado por el cliente. De este modo, se consigue que la información sensible (usuario y claves de paso) no puedan ser usadas por un atacante que haya conseguido interceptar la transferencia de datos de la conexión, pues lo único que obtendrá será un flujo de información cifrada que le resultará imposible decodificar. El puerto estándar para este protocolo es 443.

El nivel de protección depende de la exactitud de la implementación del navegador web, del *software* del servidor y de los algoritmos de cifrado actualmente soportados; esto significa que HTTPS es vulnerable cuando se aplica a contenido estático de publicación disponible. El sitio entero puede ser indexado usando una araña web y es posible que la URI del recurso cifrado sea adivinada conociendo sólo el tamaño de la petición/respuesta. Esto permite a un atacante tener acceso al contenido estático de publicación y a la versión cifrada, facilitando un ataque criptográfico. Debido a que SSL opera bajo HTTP y no tiene conocimiento de protocolos de nivel más alto, sus servidores solamente pueden presentar estrictamente un certificado para una combinación de puerto/IP en particular. Esto quiere decir que en la mayoría de los casos no es recomendable usar un *hosting virtual* (*name-based*) con HTTPS.

Figura 1.13 Símbolo del protocolo de hipertexto HTTPS

Existe una solución llamada *Server Name Indication* (SNI) que envía el host-name al servidor antes de que la conexión sea cifrada; sin embargo, muchos navegadores antiguos no soportan esta extensión, aunque para las versiones más actualizadas la opción está incluida de origen.[11] La figura 1.13 muestra el símbolo de hipertexto seguro HTTPS.

1.3.7 NFS

El sistema de archivos de red (NFS, *Network File System*) es un protocolo a nivel de aplicación, según el Modelo OSI, que se utiliza para sistemas de archivos distribuidos en un entorno de red de computadoras de área local. Éste posibilita que distintos sistemas

[10] *Secure Sockets/Transport Layer Security* o capa de puertos seguros/seguridad de la capa de transporte.

[11] El soporte para SNI está disponible desde Firefox 2, Opera 8 e Internet Explorer 7 sobre Windows Vista.

conectados a una misma red accedan a ficheros remotos como si se tratara de locales (Huerta Villalón, s/a). En 1984, se desarrolló por *Sun Microsystems* con el objetivo de que fuera independiente de la máquina, el sistema operativo y el protocolo de transporte, lo cual fue posible gracias a que se implementó sobre los protocolos XDR (presentación) y ONC RPC (sesión). La figura 1.14 muestra la operación de NFS.

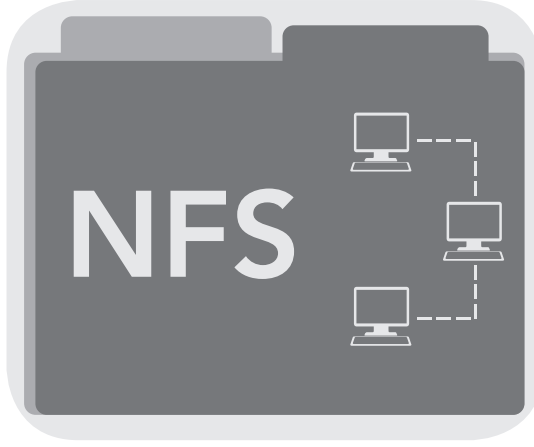


Figura 1.14 Operación de NFS

El protocolo NFS está incluido por defecto en los sistemas operativos UNIX y en la mayoría de distribuciones Linux. Las características más importantes son las siguientes:

- Está dividido al menos en dos partes principales: un servidor y uno o más clientes, donde estos últimos acceden de forma remota a los datos que se encuentran almacenados en el primero.
- Las estaciones de trabajo locales utilizan menos espacio de disco debido a que los datos se encuentran centralizados en un lugar único, pero pueden ser accedidos y modificados por varios usuarios, de forma que no es necesario replicar la información.
- Los usuarios no necesitan disponer de un directorio *home* en cada una de las máquinas de la organización, pues estos pueden crearse en el servidor de NFS para posteriormente acceder a ellos desde cualquier máquina a través de la infraestructura de red.
- También se pueden compartir a través de la red dispositivos de almacenamiento como CD-ROM y unidades ZIP; lo cual puede reducir la inversión en dichos dispositivos y mejorar el aprovechamiento del *hardware* existente en la organización.

Todas las operaciones sobre ficheros son síncronas; lo cual significa que retornan cuando el servidor ha completado todo el trabajo. En caso de una solicitud de escritura, el servidor plasmará físicamente los datos en el disco y, si es necesario, actualizará la estructura de directorios antes de devolver una respuesta al cliente, lo que garantiza la integridad de los ficheros. Al principio NFS soportaba 18 procedimientos para todas las operaciones básicas de E/S. Por su parte, los comandos de la versión 2 del protocolo son los siguientes:

NULL. Sirve para hacer *ping* al servidor y medir tiempos.

CREAT. Crea un nuevo archivo.

LOOKUP. Busca un fichero en el directorio actual y, si lo encuentra, devuelve un descriptor con más información sobre sus atributos.

READ y WRITE. Instrucciones básicas para acceder al fichero.

RENAME. Renombra un fichero.

REMOVE. Borra un fichero.

MKDIR y RMDIR. Creación y borrado de subdirectorios.

READDIR. Lee la lista de directorios.

GETATTR y SETATTR. Devuelve conjuntos de atributos de ficheros.

LINK. Crea un archivo como enlace a otro en un directorio especificado.

SYMLINK y READLINK. Llevan a cabo la creación y lectura, respectivamente, de enlaces simbólicos (en un *string*) a un archivo en un directorio.

STATFS. Devuelve información del sistema de archivos.

ROOT. Sirve para ir a la raíz (obsoleto en la versión 2).

WRITECACHE. Reservado para un uso futuro.

En la versión 3 del protocolo se eliminan los comandos "STATFS", "ROOT" y "WRITECACHE" y se agregaron los siguientes:

ACCESS. Verifica permisos de acceso.

MKNOD. Crea un dispositivo especial.

READDIRPLUS. Versión mejorada de "READDIR".

FSSTAT. Devuelve información del sistema de archivos en forma dinámica.

FSINFO. Retorna información del sistema de archivos en forma estática.

PATHCONF. Recupera información POSIX.

COMMIT. Envía datos de caché sobre un servidor con un sistema de almacenamiento estable.

Dichos comandos se corresponden con la mayoría de primitivas de E/S usadas en el sistema operativo local para acceder a ficheros locales. De hecho, una vez que se ha montado el directorio remoto, el sistema operativo local tiene que "reencaminar" las primitivas de E/S al *host* remoto; esto hace que todas sus operaciones sobre ficheros tengan el mismo aspecto, sin importar si es local o remoto.

El usuario puede trabajar con los comandos y programas habituales en ambos tipos de ficheros, ya que el protocolo NFS es completamente transparente al usuario. La versión 4 fue publicada en abril de 2003, no es compatible con las versiones anteriores y soporta 41 comandos.[12]

[12] NULL, COMPOUND, ACCESS, CLOSE, COMMIT, CREATE, DELEGPURGE, DELEGRETURN, GETATTR, GETFH, LINK, LOCK, LOCKT, LOCKU, LOOKUP, LOOKUPP, NVERIFY, OPEN, OPENATTR, OPEN_CONFIRM, OPEN_DOWNGRADE, PUTFH, PUTPUBFH, PUTROOTFH, READ, READDIR, READLINK, REMOVE, RENAME, RENEW, RESTOREFH, SAVEFH, SECINFO, SETATTR, SETCLIENTID, SETCLIENTID_CONFIRM, VERIFY, WRITE, RELEASE_LOCKOWNER, ILLEGAL.

En resumen, en la actualidad hay tres versiones de NFS en uso:

- La versión 2 (NFSv2), que es la más antigua y está ampliamente soportada por muchos sistemas operativos.
- La versión 3 (NFSv3) tiene más características, incluyendo manejo de archivos de tamaño variable y mejores facilidades de informes de errores, pero no es completamente compatible con los clientes NFSv2.
- NFS versión 4 (NFSv4) incluye seguridad *Kerberos*, trabaja con cortafuegos, permite ACL y utiliza operaciones con descripción del estado (Neuman y Ts´O, 1994).

1.3.8 CIFS

Server Message Block (SMB) es un protocolo de red que pertenece a la capa de aplicación en el Modelo OSI, el cual permite compartir archivos, impresoras y otros elementos entre nodos de una red. Es utilizado en computadores con Microsoft Windows y DOS. La figura 1.15 muestra un diagrama general del uso de CIFS.

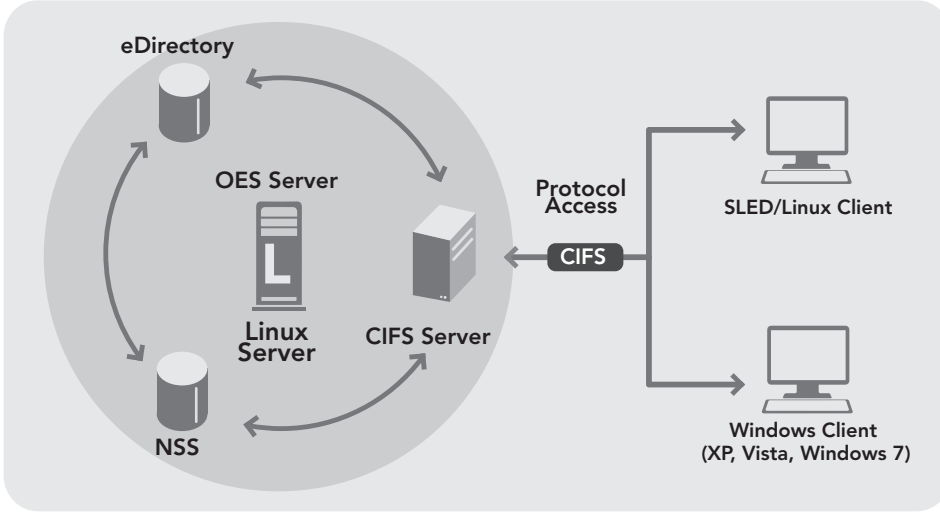


Figura 1.15 Uso de CIFS

Los servicios de impresión y el SMB para compartir archivos se han transformado en el pilar de las redes de Microsoft: en las versiones antiguas de sus productos los servicios de SMB utilizaron un protocolo que no es TCP/IP para implementar la resolución de nombres de dominio; luego, a partir de Windows 2000, se comenzó a utilizar la denominación DNS, la cual permite a los protocolos TCP/IP compartir recursos SMB (inventados por IBM), aunque la versión más común hoy en día es la modificada por Microsoft en 1998, que renombró como *Common Internet File System* (CIFS) a la cual se le añadieron características que incluyen soporte para enlaces simbólicos, enlaces duros (*hard links*) y mayores tamaños de archivo.

También existe Samba, que es una implementación libre del protocolo SMB con las extensiones de Microsoft, la cual funciona sobre sistemas operativos GNU/Linux y en UNIX.