Administración y control de tráfico en una red local

Autor: Teresa Domínguez Reséndiz

Tesis presentada para obtener el título de:
Ing. En Sistemas computacionales

Nombre del asesor:
Gabriel Nava Vázquez

UNIVERSIDAD VASCO DE QUIROGA

ESCUELA DE SISTEMAS COMPUTACIONALES

"ADMINISTRACIÓN Y CONTROL DE TRÁFICO EN UNA RED LOCAL"

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO EN SISTEMAS COMPUTACIONALES

P R E S E N T A:

TERESA DOMÍNGUEZ RESÉNDIZ

A S E S O R:

M.C. GABRIEL NAVA VÁZQUEZ

CLAVE: 16PSU0049F

MORELIA, MICHOACÁN **AGOSTO 2006**

Agradecimientos

A Dios:

Por darme la oportunidad de vivir y
nunca dejarme en los momentos más
dificiles.

A mis padres:

Que han sido mi apoyo y mi formación,
gracias por el aliciente para
cumplir esta meta.

A mis hermanos:

Jenny Marcela, Marcelo y Patricio Alejandro
Gracias por su cariño y por estar
siempre cuando los necesité.

A mi esposo:

Muchas gracias Gama por todo
tu amor y tu comprensión
y por darme tu mano para que
caminemos juntos.

A ti hijo:

Gracias por tu amor incondicional David,
y porque con solo verte, me motivas
a seguir hacia el frente.

A mi asesor:

M.C. Gabriel Nava Vázquez
por su paciencia y guía para
poder llevar este proyecto
a terminación.

Ing. Rodrigo Díaz Hernández:

Te agradezco la oportunidad y
el tiempo que dedicaste a llevarme
por el mundo de Linux, con tu apoyo
he podido concluir este trabajo.

Y a todas aquellas personas de quienes
recibí el apoyo para poder concluir este
proyecto.

INDICE

INTRODUCCIÓN

El objetivo principal de las redes de computadoras es compartir recursos (software, datos, dispositivos periféricos como impresoras, módems, máquinas de fax, discos duros y otros equipos para almacenamiento de datos) entre los diversos dispositivos conectados a ellas. Una red puede ser tan pequeña como dos computadoras enlazadas por un solo cable o tan grande que conecte cientos de computadoras y periféricos con distintas configuraciones.

Cuando en la década de los 80's se presenta la computadora personal (PC), el adjetivo personal parece ser el mas cercano y acertado para su descripción. Estaba dirigido a las personas que deseaban disponer de su propia computadora.

En la actualidad, el acceso a una computadora se hace más frecuente, su utilidad y aplicaciones están relacionadas con casi todas las actividades de la vida cotidiana y no solo a particulares, también a organizaciones con diversos niveles jerárquicos y que son importantes dentro de la sociedad.

Es aquí en donde se ubica la importancia del concepto de "red de computadoras", ¿Que es?, ¿Para que sirve?, ¿En que me beneficia? El uso de una red de computadoras es muy importante dentro de una organización debido a todos los beneficios que aporta. Mejorar el rendimiento, ahorrar dinero y tiempo, así como compartir información entre los equipos, son solo algunos de los objetivos que nos proporcionaría el instalar una red.

Contar con una red nos resulta muy útil, pero no podemos dejar de mencionar que su uso también acarrea ciertos riesgos y que muchas veces pasan por encima de los propios beneficios que nos proporciona. Por mencionar algunos de estos riesgos, está la seguridad de los equipos, la perdida de tiempo entre el personal cuando hacen un uso indebido de estos, la intrusión de usuarios ajenos o no deseados a nuestro equipo, los altos costos cuando se pretende comprar programas para cada equipo y sobre todo, quizás la mas importante de estas, la protección de los datos y de la información.

El implementar mecanismos para la administración de la red nos permite tener control sobre los accesos que se tienen a ésta, y también nos permite administrar el uso de los servicios de la misma.

En el contexto de las redes de computadoras se encuentran los conceptos de ancho de banda y filtrado de paquetes, cuyos servicios y control son temas principales en este proyecto. Veremos como administrar su uso y el manejo del filtrado de paquetes aplicándose dentro de un servidor Linux, ya que con su correcto uso, pueden ser muy útiles para la administración de la propia red, lo que a la larga será benéfico para la organización.

Cabe señalar que las herramientas que se utilizarán en el proyecto cuentan con funciones específicas, es decir, se elige utilizarlas puesto que cada una cubre cosas distintas pero que en conjunto nos permiten una buena administración de la red. Con el filtrado de paquetes `iptables`, se está tratando de mantener segura la red, para poder permitir o denegar el tráfico que cubra las necesidades de la organización, dando paso a permitir o no, el uso de algunas aplicaciones, y que aunado a esto, mantenga alejados a los intrusos, aún cuando por descuido o con conocimiento previo, se pudiera exponer a la red.

Además de filtrar el tráfico, también es importante que aquel que sea permitido, lo podamos regular en cuanto a uso. La herramienta que nos proporciona ese beneficio es la utileria `tc`, que en un grado más complejo se presenta como `CBQ`, el que va a limitar o regular el ancho de banda que se pudiera consumir para evitar un congestión dentro de la red.

Como se menciona, estás herramientas en conjunto nos proporcionan un servicio y una administración más completa, que aún cuando cada herramienta por separado resulta bastante útil, al complementarse, nos darán una red más segura y con mejor funcionamiento.

CAPITULO I

INTRODUCCION A LAS REDES

Debido al rápido progreso de la tecnología, la industria de las computadoras está avanzando rápidamente y las diferencias entre juntar, transportar, almacenar y procesar la información están desapareciendo con rapidez.

Aunque la industria de las computadoras es joven comparándola con otras industrias, las computadoras han logrado un progreso bastante considerable en poco tiempo.

1.1 ¿Que es una red?

Durante las dos primeras décadas de su existencia, los sistemas de cómputo eran centralizados y de proporciones muy grandes y lo que antes se conocía como un "centro de cómputo" en el que se contaba con una gran computadora a la cual los usuarios llevaban sus trabajos para ser procesados, ahora es algo completamente obsoleto.

Ese viejo modelo de una sola computadora ahora ha cambiado por uno en el cual un gran número de computadoras separadas pero interconectadas hacen el trabajo. A esto es a lo que podemos llamar una **red de computadoras**.

El concepto de red es algo complejo por lo que se podría definir de una forma más sencilla como *un conjunto de computadoras o nodos*[1] *(dos como mínimo) que se unen a través de medios físicos (hardware) y lógicos (software) para compartir información y recursos con el fin de llevar a cabo una actividad o labor de forma eficiente y eficaz.*

Cabe aclarar que el término red, no solo es para referirse a un conjunto de computadoras, ya que se puede incluir otro tipo de dispositivos y aún así tener una red.

Veamos entonces que cuando hablamos de una red nos referimos a "un conjunto de material preparado para que los nodos puedan comunicarse uno con otro"[1].

1.2 Usos de las redes

Es importante entender por que la gente está interesada en las redes de computadoras y para que puede utilizarlas.

1.2.1 Redes para las compañías

Muchas organizaciones tienen una cantidad importante de computadoras en operación, generalmente alejadas entre si. En términos generales, la cuestión aquí es **compartir recursos**[2] y la meta es hacer que todos los programas, el equipo y los datos estén disponibles para cualquiera en la red, sin importar la localización física de los recursos de los usuarios. En otras palabras el hecho de que el usuario este a muchos kilómetros de distancia de sus datos significa que los podrá ver como si fueran locales.

---

1 Por nodo hacemos referencia a computadoras, impresoras, hubs y cualquier otro dispositivo que tengamos unido a la red.

Una segunda meta es **ahorrar dinero**. Las computadoras pequeñas tienen una relación precio/rendimiento mucho mejor que las grandes. Las mainframes son aproximadamente 10 veces más rápidas que las computadoras personales, pero cuestan mil veces más.

Otra meta al establecer redes es la **escalabilidad**: la capacidad para incrementar el rendimiento del sistema gradualmente cuando la carga de trabajo crece, añadiendo solamente más procesadores.

Un objetivo más del establecimiento de una red de computadoras es proporcionar un potente **medio de comunicación** entre usuarios muy distantes. Al usar una red se hace más fácil la cooperación entre grupos de gente muy apartada y quizá, a largo plazo, el uso de las redes mejorará la comunicación entre las personas.

1.2.2 Redes para la gente

Todas las motivaciones anteriormente citadas para construir redes de computadoras son de naturaleza esencialmente económica y tecnológica.

Al iniciar la década de los 90, las redes de computadoras empezaron a prestar servicios a particulares en su hogar. Estos servicios son muy diferentes al modelo de "eficiencia corporativa", a continuación mencionaremos tres de los más importantes aspectos de esta evolución:

- acceso a información remota
- comunicación de persona a persona
- entretenimiento interactivo

El acceso a la información remota viene en muchas formas. Un área en la cual ya se esta utilizando es el acceso a instituciones financieras, mucha gente paga o administra cuentas bancarias de forma electrónica[2]. Las compras desde el hogar se están haciendo populares, los periódicos se publican en línea y algunos pueden ser personalizados. El siguiente paso más allá de los periódicos podrá ser la biblioteca digital en línea.[2]

Otra aplicación en esta categoría es el acceso a la actual red mundial (World Wide Web-WWW), la cual contiene información sobre arte, cultura, negocios, deportes, cocina, gobierno, salud, historia, etc.

La segunda categoría de redes que se usa implica la interacción persona a persona. Millones de personas utilizan ya el correo electrónico, el correo electrónico en tiempo real permite a los usuarios remotos comunicarse sin retraso, posiblemente viéndose y escuchándose hasta realizar reuniones virtuales llamadas videoconferencias.

La tercera categoría es el entretenimiento, la aplicación más interesante aquí es que es posible seleccionar cualquier película o programa de televisión y exhibirlo en pantalla de forma instantánea y algunas películas son interactivas y tenemos ya juegos de simulación en tiempo real multipersonales.

[2] A la fecha de terminación de este proyecto, los servicios mencionados en Redes para la gente ya son una realidad, y están siendo utilizados por miles de personas alrededor de todo el mundo.

La capacidad de combinar información, comunicación y entretenimiento seguramente hará surgir una nueva y gran industria basada en redes de computadoras.

1.3 Protocolos de Comunicación

Los protocolos son reglas y procedimientos para la comunicación. El término "protocolo" se utiliza en distintos contextos. Por ejemplo, los diplomáticos de un país se ajustan a las reglas del protocolo creadas para ayudarles a interactuar de forma correcta con los diplomáticos de otros países. De la misma forma se aplican las reglas del protocolo al entorno informático. Cuando dos equipos están conectados en red, las reglas y procedimientos técnicos que dictan su comunicación e interacción se denominan protocolos.

Al hablar de protocolos de red hay que recordar los siguientes tres puntos:

- **Existen muchos protocolos:** Aunque cada protocolo facilita la comunicación básica, cada uno tiene un propósito diferente y realiza distintas tareas. Cada protocolo tienes sus propias ventajas y limitaciones.
- **Algunos protocolos solo trabajan en ciertos niveles OSI:** El nivel al que trabaja un protocolo describe su función. Por ejemplo, un protocolo que trabaje en el nivel físico asegura que los paquetes de datos pasen a la tarjeta de red (NIC) y salgan al cable de red.
- **Los protocolos también pueden trabajar juntos en una jerarquía o conjunto de protocolos:** Los niveles de la jerarquía de protocolos se corresponden con los niveles del modelo OSI. Por ejemplo, el nivel de aplicación del protocolo TCP/IP se corresponde con el nivel de presentación del modelo OSI.

1.3.1 Protocolos en una arquitectura multinivel

En una red, tienen que trabajar juntos varios protocolos. Al trabajar juntos, aseguran que los datos se preparan correctamente, se transfieran al destino correspondiente y se reciban de forma apropiada.

El trabajo de los distintos protocolos tiene que estar coordinado de forma que no produzcan conflictos o realicen tareas incompletas. Los resultados de esta coordinación se conocen como *trabajo en niveles.* Para trabajar en niveles, también se debe llevar una jerarquía para los niveles. Una jerarquía de protocolos[5] es una combinación de protocolos. Cada nivel de la jerarquía especifica un protocolo diferente para la gestión de una función o de un subsistema del proceso de comunicación y cada nivel tiene su propio conjunto de reglas.

1.4 Modelo OSI

El modelo OSI (*Open System Interconection*) es la propuesta que hizo la Organización Internacional para la Estandarización (ISO)[6] para estandarizar la interconexión de

sistemas abiertos[3].

El modelo en sí mismo no puede ser considerado una arquitectura, ya que no especifica el protocolo que debe ser utilizado en cada capa, sino que suele hablarse de modelo de referencia. Este modelo esta dividido en siete capas:

1. Capa física
2. Capa de enlace de datos
3. Capa de red
4. Capa de transporte
5. Capa de sesión
6. Capa de presentación
7. Capa de aplicación

1.4.1 Capa física

La capa física del modelo OSI es la que se encarga de las conexiones físicas de la computadora hacia la red, tanto en lo que se refiere al medio (cable conductor, fibra óptica, inalámbrico), las características del medio (p.e. tipo de cable o calidad del mismo) como en la forma en la que se transmite la información (codificación de señales., modulación, niveles de corriente eléctrica).

Es la encargada de transmitir los bits de información a través del medio utilizado para la transmisión. Se ocupa de las propiedades físicas y de las características eléctricas de los diversos componentes.

Sus principales funciones se pueden resumir como:

- Definir el medio o medios físicos por los que va a viajar la comunicación: cable de pares trenzados, coaxial, guías de onda, fibra óptica, aire.
- Transmitir el flujo de bits a través del medio.
- Manejar las señales eléctricas/electromagnéticas.
- Especificar cables, conectores y componentes de interfaz con el medio de transmisión, polos en un enchufe.

1.4.2 Capa de enlace de datos

Es la segunda capa de el modelo OSI recibe peticiones de la capa de red y utiliza los servicios de la capa física

A partir de cualquier medio de transmisión debe ser capaz de proporcionar una transmisión libre de errores. Debe crear y reconocer los límites de las tramas, así como resolver los problemas derivados del deterioro, pérdida o duplicidad de las tramas. También debe incluir algún mecanismo de regulación de tráfico que evite la saturación de un receptor que sea más lento que el emisor.

---

3 Un sistema abierto se refiere a que es independiente de una arquitectura específica.

1.4.3 Capa de Red

El objetivo de la capa de red es hacer que los datos lleguen desde el origen al destino, aún cuando ambos no estén conectados directamente. Es decir, se encarga de encontrar un camino, atravesando los equipos que sea necesario para hacer llegar los datos al destino.

Adicionalmente, la capa de red es también la encargada de gestionar la congestión de una red[4].

1.4.4 Capa de Transporte

La función principal es la de aceptar los datos de la capa superior y dividirlos en unidades más pequeñas para pasarlos a la capa de red, asegurando que todos los segmentos lleguen correctamente, esto debe ser independiente del hardware en el que se encuentre.

Acepta los datos de la capa de sesión, los divide si es necesario y los pasa a la capa de red asegurándose que lleguen bien a su destino. Es la parte encargada de garantizar la calidad de transmisión de datos.

Además este nivel actúa como puente entre los tres niveles inferiores totalmente orientados a las comunicaciones y los tres niveles superiores totalmente orientados al procesamiento. Además, garantiza una entrega confiable de la información[4].

1.4.5 Capa de Sesión

El objetivo básico de la capa de sesión es gestionar una sesión de trabajo, es decir, que debe dar un servicio orientado a conexión[5] a pesar de lo que pueda tener por debajo.

Permite a los usuarios sesionar entre sí permitiendo acceder a un sistema de tiempo compartido a distancia, o transferir un archivo entre dos máquinas

1.4.6 Capa de Presentación

El objetivo de la capa de presentación es encargarse de la presentación de la información, se ocupa de los aspectos de semántica y sintaxis, de manera que aunque distintos equipos puedan tener diferentes representaciones internas de: caracteres (ASCII, unicode[6], etc), números, sonidos e imágenes; los datos lleguen de manera reconocible.

4] Fenómeno que se produce cuando la saturación de un nodo tira toda la red (similar a un embotellamiento en un cruce importante de la ciudad).

5 El servicio orientado a conexión se modelo basándose en el sistema telefónico. Así el usuario establece la conexión, la usa y luego la desconecta.

6 UNICODE es una norma de unificación de caracteres. Su objetivo es asignar a cada posible carácter de cada posible lenguaje un número y nombre *único*.

1.4.7 Capa de Aplicación

Ofrece a las aplicaciones la posibilidad de acceder a los servicios de las demás capas y define los protocolos que utilizan las aplicaciones para intercambiar datos, como correo electrónico, gestores de bases de datos. Hay tantos protocolos como aplicaciones distintas y debido a que continuamente se desarrollan aplicaciones distintas el número de protocolos crece sin parar.

El nivel de aplicación es casi siempre el más cercano al usuario.

Nivel de aplicación	Inicia o acepta una petición
Nivel de presentación	Añade información de formato, presentación y cifrado al paquete de datos
Nivel de sesión	Añade información del flujo de tráfico para determinar cuando se envía el paquete
Nivel de transporte	Añade información para el control de errores
Nivel de red	Se añade información de dirección y secuencia de paquetes
Nivel de enlace de datos	Añade información de comprobación de envío y prepara los datos para que vayan a la conexión física
Nivel físico	El paquete se envía como una secuencia de bits

Figura 1.1 Niveles del Modelo OSI

1.5 Protocolo TCP/IP

La familia de protocolos TCP/IP (*Transmission Control Protocol/ Internet Protocol*) caracteriza un estándar de protocolos de comunicación entre equipos informáticos. El protocolo de Internet (IP) y el protocolo de transmisión (TCP) fueron desarrollados en 1973 para un proyecto de un sistema basado en redes de conmutación de paquetes desarrollado por el gobierno estadounidense y la agencia de defensa, ARPA.

Muchas grandes redes han sido implementadas con este protocolo, una gran variedad de *universidades, dependencias gubernamentales y empresas están conectadas mediante el protocolo TCP/IP. Cualquier máquina de la red puede comunicarse con otra distinta y al tener esta conectividad se pueden enlazar redes que físicamente son independientes,* mediante una red virtual llamada Internet.

El Internet comenzó siendo una red de ARPA (llamada ARPAnet) que conectaba redes de computadoras de varias universidades y laboratorios de investigación en Estados Unidos. Cuando el TCP/IP comenzó a ser usado en máquinas Unix, la infraestructura básica del Internet apareció. Una red TCP/IP es como la carretera por donde pasará la información. Por este motivo, el Internet esta completamente montado sobre el protocolo TCP/IP.

TCP/IP proporciona la base para muchos servicios útiles como el correo electrónico, transferencia de archivos, hacer un login remoto, entre otros.

En Internet se diferencían cuatro niveles o capas en las que se agrupan los protocolos, y que se relacionan con los niveles OSI de la siguiente manera[3]:

- **Aplicación:** Se corresponde con los niveles OSI de aplicación, presentación y sesión. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico (SMTP), transferencia de ficheros (FTP), conexión remota (TELNET) y otros más recientes como el protocolo http (*Hypertext Transfer Protocol*).
- **Transporte:** Coincide con el nivel de transporte del modelo OSI. Los protocolos de este nivel, tales como TCP y UDP se encargan de manejar los datos y proporcionar la fiabilidad necesaria en el transporte de los mismos.
- **Internet:** Es el nivel de red en el modelo OSI. Incluye al protocolo IP, que se encarga de enviar los paquetes de información a sus destinos correspondientes. Es utilizado con esta finalidad por los protocolos del nivel de transporte.
- **Enlace:** Los niveles que corresponden al modelo OSI son el de enlace y el nivel físico. Los protocolos que pertenecen a este nivel son los encargados de la transmisión a través del mediò físico al que se encuentra conectado cada host.

Modelo OSI	TCP/IP
Aplicación	Aplicación
Presentación	
Sesión	
Transporte	Transporte
Red	Internet
Enlace	Enlace
Físico	

Figura 1.2 Comparación entre el modelo OSI y el TCP/IP[3]

Actualmente constituye la infraestructura tecnológica más extendida y desarrollada sobre la que circulan las comunicaciones electrónicas (datos, voz, multimedia...). Su expansión se ha debido principalmente al desarrollo exponencial de la red mundial, Internet.

Desde sus comienzos, en la red existen computadoras de diferentes tipos, por lo que se hizo necesario un protocolo común y único, de manera que todas pudieran entender e interpretar correctamente la información que por ellas circula. Este protocolo se denominó **TCP/IP,** que es un conjunto de protocolos de comunicaciones que definen como se pueden comunicar entre sí computadoras y otros dispositivos de distintos tipos.

Hay que tener en cuenta que en Internet se encuentran conectados equipos de clases muy diferentes y con hardware y software incompatibles en muchos casos. Aquí se encuentra una de las grandes ventajas de TCP/IP, pues este protocolo se encargará de que la comunicación entre todos sea posible, siendo entonces el TCP/IP la base de Internet, que sirve para enlazar computadoras que utilizan diferentes sistemas operativos incluyendo PC, mini computadoras y computadoras centrales en cualquier tipo de red.

TCP/IP no es un protocolo único sino que lo que se conoce con este nombre es en realidad el conjunto de varios protocolos y sus dos protocolos más importantes son el **TCP** que significa *Transmision Control Protocol* e **IP** que es *Internet Protocol.* Mencionando algunos de los protocolos que forman la familia TCP/IP están los siguientes:

- FTP (*File Transfer Protocol*): se usa para transferencia de archivos.
- SMTP (*Simple Mail Transfer Protocol*): es una aplicación para el correo electrónico.
- TELNET: permite la conexión a una aplicación remota desde un proceso o terminal.
- HTTP: (*Hyper Text Transfer Protocol*): Protocolo de transferencia de hipertexto utilizado en la Web.

Independientemente de su significado concreto, TCP/IP ha llegado a ser casi sinónimo de Internet y a todo lo relacionado con la red.

En principio, el **IP** es el encargado de la transmisión de los datos, que sea posible el tráfico de una computadora a otra, mientras que el **TCP** es el encargado de juntar los paquetes que se han enviado, de pedir los que faltan (en su caso) y finalmente, ordenarlos, puesto que la red no garantiza la llegada de todos los paquetes ni tampoco que su llegada sea en orden.

En realidad **TCP** se encarga de "negociar" con el equipo remoto ciertos parámetros que determinan los detalles del modo en que se realizará la transmisión.

1.6 La Capa de Red en Internet

En la capa de red, el Internet puede verse como un conjunto de subredes, o sistemas autónomos[7] interconectados. No hay una estructura real, pero existen varios *backbone* principales. Estos se construyen a partir de líneas de alto ancho de banda y enrutadores rápidos. Conectadas a los *backbone* de redes regionales (de nivel medio), y conectadas a estas redes están las LAN de muchas universidades, compañías y proveedores de servicio Internet.

1.6.1 Terminología

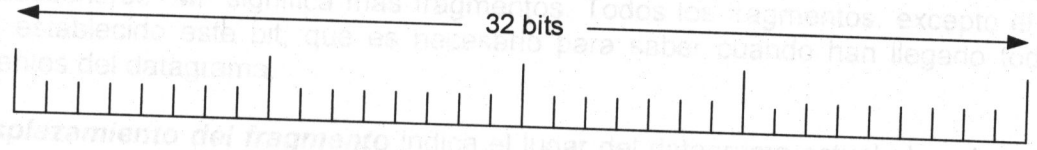
- **Backbone:** Línea de gran capacidad a la que se conectan líneas de menor capacidad a través de puntos de conexión llamados nodos. La traducción literal "columna vertebral" o "espina dorsal".
- **Datagrama:** Un datagrama es un fragmento de paquete que es enviado con la suficiente información como para que la red pueda simplemente encaminar el fragmento hacia el equipo receptor, de manera independiente a los fragmentos restantes. Esto puede provocar una recomposición desordenada o incompleta del paquete en el equipo destino.
- **Host**: Literalmente anfitrión. Es un equipo directamente conectado a una red y que efectúa y alberga servicios disponibles para otros equipos de la red. Es una aplicación informática que realiza algunas tareas en beneficio de otras aplicaciones llamadas clientes. Algunos servicios habituales son los servicios de archivos, que permiten a los usuarios almacenar y acceder a los archivos de un equipo o equipo y los servicios de aplicaciones que realiza en beneficio del usuario final.

[7] Al referirnos a un sistema autónomo se habla de un equipo el cuál no tiene una relación amo-esclavo con otro. Si una computadora puede parar, arrancar o controlar otra a voluntad, no es autónoma.

1.6.2 Protocolo IP

El pegamento que mantiene unido al Internet es el protocolo IP (*Internet Protocol*, protocolo de Internet). A diferencia de la mayoría de los protocolos, éste se diseñó desde el principio con la interconexión de redes en mente. Podemos entender que el trabajo de la capa de red es proporcionar una mejor ruta para el transporte de datagramas del origen al destino, sin importar si estas máquinas están en la misma red, o si hay otras redes en ellas.

Para comenzar con el estudio de la capa de red en Internet veremos cual es el formato de los datagramas de IP mismos. Un datagrama de IP consiste en una parte de cabecera y una parte de texto.



Versión	IHL	Tipo de Servicio	Longitud total
Identificación		D F / M F	Desplazamiento del fragmento
Tiempo de Vida	Protocolo		Suma de comprobación de la cabecera
Dirección de Origen			
Dirección de Destino			
Opciones (0 ó más palabras)			

Figura 1.3 Formato del datagrama IP (Protocolo de Internet)

El campo ***Versión*** indica a que versión del protocolo pertenece cada uno de los datagramas[7] y que pueden ser:

- IPv4: es la versión 4 del protocolo IP y esta fue la primera versión del protocolo que se implementó y forma la base del Internet.
- IPv6: es la versión 6 del protocolo de Internet y esta siendo implementada para solucionar la limitación de direcciones de red admitidas.

Dado que la longitud de la cabecera no es constante, se incluye un campo en la cabecera, ***IHL***, para indicar la longitud en palabras de 32 bits. El valor mínimo es de 5, cifra que es aplicada cuando no hay opciones.

El campo de ***tipo de servicio*** permite al *host* indicar a la subred el tipo de servicio que quiere. Son posibles varias combinaciones de confiabilidad y seguridad. Por ejemplo, para voz digitalizada es más importante la entrega rápida que la entrega precisa, y para

transmisión de archivos resulta ser más importante la transmisión libre de errores que la entrega rápida.

La ***longitud total*** incluye todo el datagrama tanto la cabecera como los datos., mide en bytes la longitud del datagrama. La longitud máxima es de 65535 bytes (2^16).

El campo de ***identificación*** es necesario para que el host de destino determine a que datagrama pertenece un fragmento recién llegado. Todos los fragmentos de un datagrama contienen el mismo valor de *identificación.*

En seguida viene un bit sin uso y después dos campos de un bit. ***DF*** significa no fragmentar. Esta es una orden para que las pasarelas no fragmenten el datagrama, por que el extremo destinatario es incapaz de poner las partes juntas nuevamente. Si el datagrama no puede pasarse a través de una red, se deberá encaminar sobre otra red o bien, desecharse. ***MF*** significa más fragmentos. Todos los fragmentos, excepto el último tienen establecido este bit, que es necesario para saber cuando han llegado todos los fragmentos del datagrama.

El ***desplazamiento del fragmento*** indica el lugar del datagrama actual al cual pertenece este fragmento. En un datagrama, todos los fragmentos, con excepción del último, deberán ser un múltiplo de 8 octetos que es la unidad elemental de fragmentación. Dado que se proporcionan 13 bits, puede haber un máximo de 8192 fragmentos de datagrama, dando una longitud máxima de datagrama de 65,536 bytes, uno más que el campo de *longitud total.*

El campo de ***tiempo de vida*** es un contador que sirve para limitar la vida de un paquete. Este contador cuenta el tiempo en segundos, permitiendo una vida máxima de 255 seg.

Una vez que la capa de red ha ensamblado un datagrama completo, necesita saber que hacer con él. El campo de ***protocolo*** especifica que protocolo de alto nivel se empleó para construir el mensaje transportado en el campo datos de datagrama IP. Algunos valores posibles son: 1 = ICMP, 6 = TCP, 17 = UDP, 88 = IGRP (Protocolo de Enrutamiento de Pasarela Interior de CISCO).

La ***suma de comprobación de la cabecera*** verifica solamente la cabecera. El algoritmo es sumar todas las medias palabras de 16 bits a medida que llegan, usando aritmética de complemento a uno, y luego obtener el complemento a uno del resultado.

La ***dirección de origen*** y la ***dirección de destino*** indican el número de red y el número de host.

El campo de ***opciones*** se diseñó para proporcionar un recurso que permitiera que las versiones siguientes del protocolo incluyeran información no presente en le diseño original y para evitar la asignación de bits de cabecera a información pocas veces necesaria.

Explicando el formato de un datagrama IP, ahora podemos hablar de la definición del protocolo IP: *Internet Protocol*, es el protocolo principal de TCP/IP encargado de la transmisión y enrutamiento de los paquetes de datos al equipo destino.

Es la base fundamental de Internet. Porta datagramas de la fuente al destino. El nivel de transporte parte el flujo de datos en datagramas. Durante su transmisión se puede partir un datagrama en fragmentos que se montan de nuevo en el destino.

Las principales características de este protocolo son:

- Protocolo no orientado a conexión
- Fragmenta paquetes si es necesario
- Direccionamiento mediante direcciones lógicas de 32 bits
- Si un paquete no es recibido, este permanecerá en la red por un tiempo finito
- Solo se realiza verificación por suma al encabezado del paquete, no a los datos que este contiene.

El protocolo de Internet proporciona un servicio de distribución de paquetes de información no orientado a conexión de manera no fiable, la no orientación a conexión significa que los paquetes de información, que será emitido a la red, son tratados independientemente, pudiendo viajar por diferentes trayectorias para llegar a su destino. El término no fiable significa más que nada que no se garantiza la recepción del paquete.

1.6.3 Puertos de Comunicación

Para identificar a las diferentes aplicaciones con las que se está trabajando, el protocolo TCP/IP envía cada paquete con un número conocido como *puerto*[8]. Se usa el concepto de *número de puerto* para identificar a las aplicaciones emisoras y receptoras. Cada lado de la conexión tiene un número asociado de puerto (16 bits sin signo, por lo que existen 65,535 puertos posibles) asignado por la aplicación emisora o receptora[9]. Los puertos definen los protocolos de aplicación y no identifican el programa de aplicación actual que está ejecutándose.

Los puertos son clasificados en tres categorías: bien conocidos, registrados y dinámicos/privados. Los puertos bien conocidos son asignados por la *Internet Assigned Numbers Authority* (IANA), van del 0 al 1023 y son usados normalmente por el sistema o procesos con privilegios. Los puertos registrados son utilizados por las aplicaciones de usuario de forma temporal cuando se conectan con los servidores, pero también pueden representar servicios que hayan sido registrados por terceros. Los puertos dinámicos/privados también pueden ser usados por las aplicaciones del usuario, pero este caso es menos común.

1.6.4 Direcciones IP

Cada host y enrutador de Internet tiene una dirección IP, que codifica su número de red y su número de host. La combinación es única: no hay dos máquinas que tengan la misma dirección de IP. Todas las direcciones IP son un número binario de 32 bits (4 octetos) de longitud, se agrupan y representan como 4 enteros que van de 0-255 y se usan en los campos de dirección de origen y de dirección de destino de los paquetes IP.

1.6.5 Clases

Existen diferentes tipos de direcciones IP. El objetivo para clasificar las redes es el de decidir cual es la parte que se encarga de la red, y que partes se dedican para los nodos.

Clase	Red (bits)	Nodos (bits)
Clase A	8	24
Clase B	16	16
Clase C	24	8
Clase D	Reservadas para uso futuro	
Clase E		

- Direcciones clase A
- Direcciones clase B
- Direcciones clase C
- Direcciones clase D
- Direcciones clase E

Las máquinas que se encuentran conectadas a varias redes tienen direcciones IP diferentes en cada red. Los formatos usados para las direcciones IP se muestran en la figura:

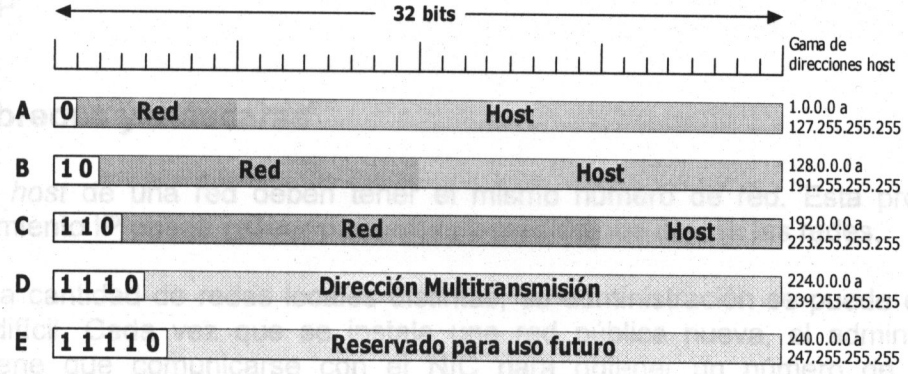


Figura 1.4 Formato de direcciones IP

Hay decenas de miles de redes ahora conectadas a Internet y la cifra se duplica cada año. Es por esto que los números de red son asignados por el **NIC** (*Network Information Center,* centro de información de redes) para evitar conflictos.

Las direcciones de red, que son números de 32 bits, generalmente se escriben en notación decimal con puntos. En este formato, cada uno de los 4 bytes se escribe en decimal, de 0 a 255. La dirección IP menor es 0.0.0.0 y la mayor 255.255.255.255.

Las direcciones de la forma 127.xx.yy.zz se reservan para pruebas de retroalimentación. Los paquetes enviados a esa dirección se procesan localmente y se tratan como paquetes de entrada. Esto permite que los paquetes se envíen a la red local sin que el transmisor conozca su número.

1.6.6 Direcciones IP reservadas

Las direcciones IP privadas son aquellas que no se usan en maquinas conectadas a Internet y que, como su nombre lo indica, se reservan para uso exclusivo de intranets o para uso experimental.

Estas son las siguientes:

	Dirección IP/Máscara	Máscara de Subred
Localhost:	127.0.0.0	255.0.0.0
Privadas:	10.0.0.0/24 172.16.0.0/21 192.168.0.0/16	255.255.255.0 255.255.248.0 255.255.0.0
Network/Broadcast:	0.0.0.0/24 224.0.0.0/19	255.255.255.0 255.255.224.0

Figura 1.5 Direcciones IP privadas

Si está conectada una red de área local (LAN) a Internet, se debe utilizar una dirección IP de alcance local para todos y cada uno de los dispositivos que se tengan en la red[10]. No es recomendable inventarse direcciones IP, ya que lo peor que puede pasar es que simplemente nada funcione y que el sistema se caiga al empezar a interferir la IP "robada" con las comunicaciones de otra máquina que ya se encontraba utilizando esa dirección IP.

1.6.7 Subredes y máscaras

Todos los *host* de una red deben tener el mismo número de red. Esta propiedad del direccionamiento IP puede causar problemas a medida que crecen las redes.

Al crecer la cantidad de redes locales distintas, su administración se puede convertir en un tema difícil. Cada vez que se instala una red pública nueva, el administrador del sistema tiene que comunicarse con el NIC para obtener un número de red nuevo. Después este número se debe anunciar mundialmente.

Además mover una máquina de una LAN a otra requiere el cambio de su dirección IP, lo que a su vez puede significar la modificación de sus archivos de configuración y también el anuncio de la nueva dirección IP a todo el mundo.

Si a otra máquina se le da la dirección de IP recién liberada, esa máquina recibirá el correo electrónico y otros datos dirigidos a la máquina original hasta que la dirección se haya propagado por todo el mundo.

La solución a este problema es permitir la división de una red en varias partes para uso interno, pero aún actuar como una sola red ante el mundo exterior. En la literatura sobre Internet, a estas partes se les llaman **subredes**.

Para conseguir mayor funcionalidad podemos dividir nuestra red en subredes dividiendo

en dos el número de *host*, uno para identificar la subred y la otra parte para identificar la máquina, esta técnica es conocida como subnetting.

Cuando se trabaja con una red pequeña, con pocos host conectados, el administrador de red puede fácilmente configurar el rango de direcciones IP usado para conseguir un funcionamiento óptimo del sistema. Pero conforme la red crece, se hace necesaria una división en partes de la misma.

Aunque segmentemos la red, conforme aumenta el número de host aumenta el número de transmisiones de broadcast (cuando un equipo origen envía datos a todos los dispositivos de la red) llegando el momento que dicho tráfico pueda congestionar toda la red de forma inaceptable, al consumir un ancho de banda excesivo. Esto es así porque todos los host están enviando información de forma constante.

Para solventar este hecho es preciso dividir la red primaria en una serie de subredes, de tal forma que cada una de ellas va a funcionar como una red individual, aunque todas pertenezcan a la misma red principal (y por tanto, al mismo dominio). De esta forma, aunque la red en su conjunto tendrá una dirección IP única, a nivel administrativo podremos considerar subredes bien diferenciadas, consiguiendo con ello un control del tráfico de la red.

En la explicacion siguiente se considera una red pública, formada por host con direcciones IP públicas, que pueden ser vistas por todas las máquinas conectadas a Internet. Pero el desarrollo es igualmente válido para redes privadas, por lo que su aplicación práctica es válida en una red corporativa. Y para hacer más claro el desarrollo se va a partir de una red con dirección IP real.

Vamos a tomar como ejemplo una red de clase C, teniendo claro que lo que expliquemos va a ser útil para cualquier tipo de red, sea de clase A, B o C. Entonces, tenemos nuestra red, con dirección IP **210.25.2.0**, por lo que tenemos para asignar a los host de la misma todas las direcciones IP del rango 210.25.2.1 al 210.25.2.254, ya que la dirección 210.25.2.0 será la de la propia red y la 210.25.2.255 será la dirección de broadcast general.

Si expresamos nuestra dirección de red en binario tendremos:

210.25.2.0 = 11010010.00011001.00000010.**00000000**

Con lo que tenemos 24 bits para identificar la red (en negro) y 8 bits para identificar los host (negritas).

La máscara de red será:

255.255.255.0 = 11111111.11111111.11111111.00000000

Para crear subredes a partir de una dirección IP de red padre, la idea es "robar" bits a los host, pasándolos a los de identificación de red. ¿Cuántos?. Bueno, depende de las subredes que queramos obtener, teniendo en cuenta que cuántas más bits robemos, más

subredes obtendremos, pero con menos host cada una. Por lo tanto, el número de bits a robar depende de las necesidades de funcionamiento de la red final.

1.6.8 Máscara de subred

Otro elemento que deberemos calcular para cada una de las subredes es su máscara de subred, concepto análogo al de máscara de red en redes generales, y que va a ser la herramienta que utilicen luego los routers para dirigir correctamente los paquetes que circulen entre las diferentes subredes.

Para obtener la máscara de subred basta con presentar la dirección propia de la subred en binario, poner a 1 todos los bits que dejemos para la parte de red (incluyendo los robados a la porción de host), y poner a 0 todos los bits que queden para los host. Por último, pasaremos la dirección binaria resultante a formato decimal separado por puntos, y ésa será la máscara de la subred.

Por ejemplo, si tenemos la dirección de clase B:

150.10.x.x =

1	0	0	1	0	1	1	0.	0	0	0	0	1	0	1	0.	h	h	h	h	h	h	h	h.	h	h	h	h	h	h	h	h

y le quitamos 4 bits a la porción de host para crear subredes:

1	0	0	1	0	1	1	0.	0	0	0	0	1	0	1	0.	r	r	r	r	h	h	h	h.	h	h	h	h	h	h	h	h

la máscara de subred será:

1	1	1	1	1	1	1	1.	1	1	1	1	1	1	1	1.	1	1	1	1	0	0	0	0.	0	0	0	0	0	0	0	0

que pasada a decimal nos queda:

255.255.240.0

Las máscaras de subred, al igual que ocurre con las máscaras de red, son muy importantes, resultando imprescindibles para el trabajo de enrutamiento de los routers.

1.7 Tipos de redes en un área geográfica

La distancia es importante como medio de clasificación porque se usan diferentes técnicas a diferentes escalas. Veamos una descripción de ésta clasificación.

1.7.1 Redes de área local

Las **redes de área local**, son llamadas LAN (*local area networks*), y son redes de propiedad privada que están dentro de un edificio o campus de unos cuantos kilómetros de extensión. A estas redes también se les llama redes de acceso, ya que se usa para tener acceso a una red de área extendida y cuando no tiene conexión con ciudades por que no esta conectada a una red de área extendida se le llama Red Interna (intranet).

Se usan principalmente para conectar computadoras personales y estaciones de trabajo en oficinas pequeñas con el objeto de compartir recursos e información y presentan las siguientes características:

- Tamaño: las LAN están restringidas en tamaño, lo que quiere decir que el tiempo de transmisión del peor de los casos esta limitado y se conoce de antemano. Conocer este límite simplifica la administración de la red.
- Tecnología de transmisión: las LAN frecuentemente utilizan una tecnología de transmisión que consiste en un cable sencillo al cual se conectan todas la maquinas.
- Topología: Las LAN pueden tener diversas topologías, por ejemplo una red de bus (cable lineal) o sistema de difusión de anillo.

1.7.1.1 Tecnologías y velocidades en una LAN

Ethernet

Es una norma o estándar que determina la forma en que los puertos de la red envían y reciben datos sobre un medio físico compartido independiente de su configuración física. Es la tecnología de red de área local más extendida en la actualidad.

La velocidad de transmisión de Ethernet es de 10 Mbps en las configuraciones habituales pudiendo llegar a ser de 100 Mbps en las especificaciones de Fast Ethernet, y de hasta 1 Gigabit por segundo en Gigabit Ethernet[12].

El protocolo Ethernet especificado en el estándar IEEE 802.3 y se ocupa principalmente de la transferencia de datos de la capa de red en la máquina de origen a la capa de red de la máquina de destino.

Soporta varios tipos de medios físicos, como el par trenzado, cable coaxial, y tiene variantes según la velocidad de transferencia: Ethernet, Fast Ethernet y Gigabit Ethernet[14].

Gigabit Ethernet, también conocida como GigE, es una ampliación del estándar Ethernet que consigue una capacidad de transmisión de 1 gigabit por segundo que en la práctica se convierten en unos 100 megabytes útiles[13] (Fast Ethernet tiene alrededor de 10).

1.7.2 Redes de área metropolitana

Una **red de área metropolitana** o MAN (*metropolitana area network*) es básicamente una versión más grande de una LAN y esta basada, generalmente, en una tecnología similar. Podría abarcar un grupo de oficinas cercanas o una ciudad y podría ser privada o pública. Una MAN puede manejar datos y voz, solo tiene un o dos cables y no contiene elementos de conmutación lo que simplifica su diseño.

La razón principal para distinguirla de otro tipo de redes es que se ha adoptado un estándar llamado DQDB Bus Dual de Cola Distribuida (Distributed Queue Dual Bus). Este consiste en dos buses (cables) unidireccionales a los cuales están conectadas todas las computadoras. Cada bus tiene una cabeza terminal y un dispositivo que inicia la actividad de transmisión. Ofrece una alta velocidad de 2 Mbps a 300 Mbps.

1.7.3 Redes de área amplia

Una **red de área amplia** o WAN (*wide area network*), se extiende sobre un área geográfica extensa, que puede ser un país o un continente; es una colección de máquinas dedicadas a ejecutar programas de usuario.

Es un sistema de comunicación entre computadoras, que permite compartir información y recursos, con la característica de que la distancia entre las computadoras es amplia (de un país a otro, de una cuidad a otra, de un continente a otro).

1.7.3.1 Tecnologías y velocidades en una WAN

Frame Relay

Frame Relay[8] es un servicio de transmisión de datos en modo paquete (tramas) que permite ofrecer soluciones a servicios que requieren de un gran ancho de banda, es decir, es una tecnología de conmutación rápida de tramas.

Esta basado en una tecnología de conmutación rápida de tramas con una baja tasa de error. Típicamente ofrecen un ancho de banda comprendido en el rango de 56 Kbps y 1.544 Mbps[15].

1.7.4 Redes inalámbricas

También se conocen como WLAN de Red Área Local Inalámbrica (*Wireless Local Area*

8 Frame Relay proviene de la evolución de las redes X.25 y de los circuitos punto a punto.

Network). Una de las tecnologías mas prometedoras de esta década es la de poder comunicar computadoras mediante tecnología inalámbrica.

Las redes inalámbricas facilitan la operación en lugares donde la computadora no puede permanecer en un solo lugar. No se espera que las redes inalámbricas reemplacen a las redes cableadas ya que estas ofrecen velocidades de transmisión mayores que las ofrecidas por la tecnología inalámbrica.

Aunque las redes inalámbricas son fáciles de instalar, también tienes desventajas. Típicamente su tasa de transmisión es de 1 a 2 Mbps, lo cual es mucho más lento que las LAN alambradas. Además las tasas de error son a veces mucho más altas por estar expuestas a cualquier tipo de interferencia.

Utiliza la variante de Ethernet para las redes inalámbricas y es el estándar 802.11. Existen varios tipos de dispositivos 802.11, y la más popular es la conocida como 802.11b, que alcanza velocidades de transmisión de 11 Mbps[14].

Entre los principales estándares se encuentran[14]:

- IEEE 802.11: Estándar original de WLANs que soporta velocidades entre 1 y 2 Mbps.
- IEEE 802.11a: Estándar de alta velocidad que soporta velocidades de hasta 54 Mbps.
- IEEE 802.11b: Estándar dominante de WLAN (conocido también como Wi-Fi) que soporta velocidades hasta de 11 Mbps.

Las redes de este tipo se están extendiendo a un paso muy acelerado, ya que usadas en dispositivos como ordenadores portátiles y ordenadores de mano se proporciona una gran movilidad al usuario y un ancho de banda alto.

En este capítulo se hace una introducción al tema de las redes, a la importancia que tienen dentro de las organizaciones y para el propio uso personal. Debido a que este proyecto está basado en el uso de las redes, principalmente en organizaciones, se hace mención, entre otros temas, del Modelo OSI,, los protocolos TCP/IP, que nos sirven para conocer el funcionamiento de la transmisión de infomración entre los equipos y que habrán de mostrar las bases que más adelante sirvan para el correcto manejo de nuestra red.

Para el capítulo siguiente, veremos una de las herramientas que nos permiten proteger nuestra red, los cortafuegos, ya que al ser cada vez mayor su uso, también se hacen cada vez más vulnerables y es casi índispensable el implementarlas para poder mantenerlas protegidas.

CAPITULO II

CORTAFUEGOS Y SEGURIDAD PARA LA RED

2.1 Un cortafuegos, concepto principal

¿Qué es un *cortafuegos*? La traducción más acertada de este término que en inglés es *firewall*, al idioma español es la palabra **cortafuegos**. Veamos cual es la definición en el DRAE[9].

> **<<Cortafuego o cortafuegos.** (De *cortar* y *fuego*). M. *Agr.* Vereda ancha que se deja en los sembrados y montes para que no se propaguen los incendios. || **2.** *Arq.* Pared toda de fábrica, sin madera alguna, y de un grueso competente, que se eleva desde la parte inferior del edificio hasta más arriba del caballete, con el fin de que, si hay fuego en un lado, no se pueda comunicar este al otro. **>>**

Estas dos definiciones ya introducen una idea muy aproximada al significado de la palabra *cortafuegos* en términos informáticos, en ellas, aparecen términos como *lado*, lo que implica la existencia de dos o más partes y *comunicar* lo que pone en manifiesto que las partes están comunicadas.

Bien, pues casi sin analogías, eso es un cortafuegos en términos de sistemas computacionales. Un cortafuegos, es un sistema informático, que actúa como punto de conexión segura entre dos o más sistemas informáticos.

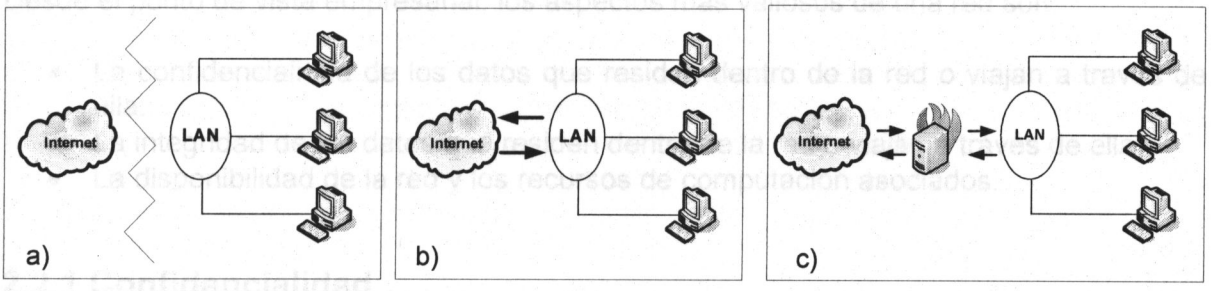


figura 2.1 a) Aislamiento de la red, b) Conexión total, c) *Cortafuegos* entre la zona riesgo y el perímetro de seguridad

Al espacio protegido se le suele llamar **perímetro de seguridad**, suele ser propiedad de la misma organización, y la protección se realiza contra una red externa, no confiable, llamada **zona de riesgo**.

Para que un cortafuegos pueda ser efectivo, todo tráfico de información que provenga del exterior deberá pasar a través del mismo donde podrá ser inspeccionada la información.

Para conocer bien el funcionamiento de un cortafuegos tenemos que hablar de algunas definiciones de sus partes o características para entenderlo:

[9] Diccionario de la Real Academia de la lengua Española. http://buscon.rae.es/diccionario/drae.htm

Se conoce por máquina o ***host* bastión** a un sistema especialmente asegurado, pero que puede ser vulnerable a todo tipo de ataques por estar abierto a Internet y que tiene como función ser el punto de contacto de la red interna de una organización con otras redes. El *host* bastión filtra el tráfico de entrada y salida y también esconde la configuración de la red hacia afuera.

Por **filtrado de paquetes** entendemos la acción de denegar o permitir el flujo de tramas entre dos redes de acuerdo a unas normas predefinidas. El filtrado también se conoce como *screening*, y a los dispositivos que lo implementan se les conoce como **chokes**[10].

Un **proxy** es un programa que permite o niega el acceso a una aplicación determinada entre dos redes. Los clientes *proxy* solo se comunican con los servidores *proxy* que son los que autorizan las peticiones a los servidores reales, o las deniegan y las devuelven a quién las solicitó.

2.2 Por qué son útiles los cortafuegos

Los cortafuegos protegen a las redes. Pero la razón por la que una red requiere protección puede no estar del todo clara. Más concretamente, debido a que el término *red* es algo abstracto, puede no estar claro que parte de la red requiere protección o contra qué hay que proteger la red.

Desde el punto de vista empresarial, los aspectos más valiosos de una red son:

- La confidencialidad de los datos que residen dentro de la red o viajan a través de ella.
- La integridad de los datos que residen dentro de la red o viajan a través de ella.
- La disponibilidad de la red y los recursos de computación asociados.

2.2.1 Confidencialidad

Algunos datos son valiosos porque no son muy conocidos. Si la información estuviera garantizada para protegerse contra cualquier pérdida que fuera ocasionada por su propio uso, esa información valdría muchísimo. Pero supongamos que todo el mundo pudiera acceder a la misma información y tuviera la misma garantía. En ese caso, la información carecería de valor. La revelación pública de la información compromete su valor. Por lo tanto, la confidencialidad de la información debe ser protegida.

2.2.2 Integridad

La integridad de una persona o institución es el grado de confianza que ofrece. Análogamente, la integridad de los datos es el grado en que podemos confiar que los datos estén completos y sean exactos y sin que hayan sido modificados incorrectamente.

10 el choke puede ser la máquina bastión o un elemento diferente.

La integridad de los datos es importante porque el valor de los datos disminuye generalmente si los datos cambian y dejan de ser exactos.

2.2.3 Disponibilidad

Es garantizar que los recursos estén a nuestro alcance cuando se necesiten, así como la probabilidad que tiene un sistema de ser usado en su totalidad para el cumplimiento de sus prestaciones.

2.3 Características de su diseño

Existen tres decisiones básicas en el diseño o configuración de un cortafuegos; la primera de ellas, y la más importante, hace referencia a la política de seguridad de la organización propietaria: evidentemente, la configuración y nivel de seguridad potencial será distinto en una empresa que utilice un cortafuegos para bloquear todo el tráfico externo hacia el dominio de su propiedad (excepto quizás, las consultas a su página *web*) frente a otra donde sólo se intente evitar que los usuarios internos pierdan el tiempo en la red, bloqueando, por ejemplo todos los servicios de salida al exterior excepto el correo electrónico.

Sobre esta decisión influyen, aparte de motivos de seguridad, motivos administrativos de cada organismo.

La segunda decisión de diseño a tener en cuenta es el nivel de monitorización y control deseado en la organización; una vez definida la política a seguir, hay que definir cómo implementarla en el cortafuegos indicando básicamente qué se va a permitir y qué se va a denegar.

Por último, la tercera decisión a la hora de instalar un sistema de cortafuegos es meramente económica: en función del valor estimado de lo que deseamos proteger, debemos gastar más o menos dinero, o no gastar nada. Un cortafuegos puede no implicar gastos extras para la organización, o si, necesitarse el desembolso de varios miles de pesos.

2.4 ¿Cual es su propósito y sus funciones?

El principal propósito de un cortafuegos es mantener segura a una red, separándola y actuando como punto seguro de conexión hacia el exterior, principalmente hacia el Internet, va examinando cada trama de datos que ha sido enviada y va bloqueando las que no cumplen con las reglas de seguridad establecidas.

Otro objetivo es mantener a los usuarios no autorizados fuera de la red privada ya que al ser establecidas las políticas y reglas de control de acceso, estos no podrán acceder por no cumplir con dichas reglas que se han adecuado para restringir o denegar cualquier tráfico ajeno o que no haya sido especificado.

Además su implementación permite vigilar entre la red pública y la red privada, actuando como centinela constantemente en las conexiones a Internet e interrogando a quien intenta acceder al sistema suponiendo que preguntan: "¿Quién es usted y su contraseña?" dando solo acceso cuando está satisfecho con la respuesta.

En términos más técnicos un cortafuegos monitorea el tráfico que entra y sale de la computadora. Revisa cada grupo de datos que intenta ir desde Internet a la computadora y viceversa. Cada paquete tiene una firma que lo identifica para saber quien lo envió y como se debe procesar el paquete. El cortafuegos ve esta información y luego toma decisiones de acceso, decisiones que se tomarán en base a las reglas que se le han establecido.

Monitorizar la actividad de nuestro cortafuegos es algo indispensable para la seguridad de todo el perímetro protegido; la monitorización nos facilitara información sobre los intentos de ataque que estemos sufriendo (origen, franjas horarias, tipo de acceso...), así como la existencia de tramas que aunque no supongan un ataque previo sí que son al menos sospechosas.

Una manera muy importante de llevar el control mas adecuado de lo que ocurre con nuestro cortafuegos es mediante registros. El cortafuegos genera registros o "logs" del tráfico entrante y saliente.

¿Que información debemos registrar? Además de los registros estándar -los que incluyen las estadísticas de tipos de paquetes recibidos, frecuencias o direcciones fuente y destino) recomienda verificar información de la conexión (origen y destino, nombre de usuario, hora y duración) intentos de uso de protocolos denegados, intentos de falsificación de dirección por parte de máquinas internas al perímetro de seguridad (paquetes que llegan desde la red externa con la dirección de un equipo interno) y tramas recibidas desde routers desconocidos. Evidentemente, todos estos registros deben ser leídos con frecuencia, y el administrador de la red deberá tomar medidas si se detectan actividades sospechosas para prevención o detección de ataques; si la cantidad de *logs* generada es considerable nos puede interesar el uso de herramientas que filtren dicha información.

2.5 Configuración típica de un cortafuegos

Para que un cortafuegos entre redes funcione como tal, debe tener al menos dos tarjetas de red. Esta sería la topología clásica de un cortafuegos:

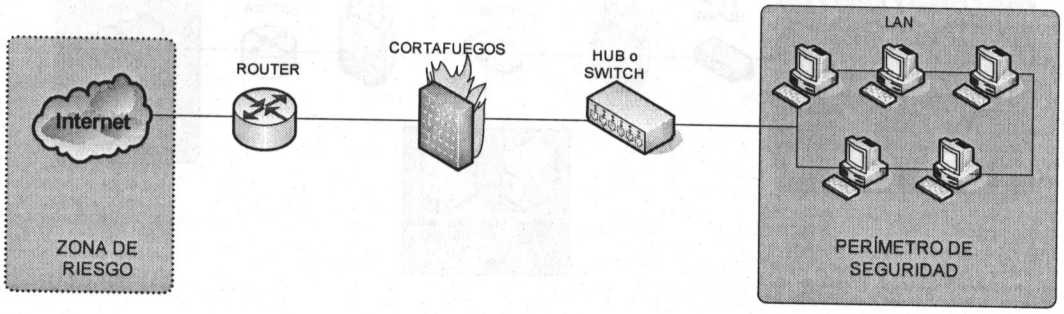


Figura 2.2 Esquema de cortafuegos típico entre la red local e Internet

Este es el esquema típico de un cortafuegos para proteger una red local conectada a Internet a través de un router. El cortafuegos debe colocarse entre el router conectado a una tarjeta de red, (generalmente llamada *eth0* en Linux, con un único cable) y la red local, con la segunda tarjeta de red, (denominada *eth1,* conectado al switch o al hub de la LAN).

Dependiendo de las necesidades de cada red, pueden ponerse uno o más cortafuegos para establecer distintos perímetros de seguridad en torno a un sistema. Es frecuente también que se necesite exponer algún servidor a Internet (como es el caso de un servidor Web, un servidor de correo, etc.), y en esos casos obviamente en principio se debe aceptar cualquier conexión a ellos. Lo que se recomienda en esa situación es situar ese servidor en un lugar aparte de la red, al que se le denomina DMZ o zona desmilitarizada. El cortafuegos tiene entonces tres entradas:

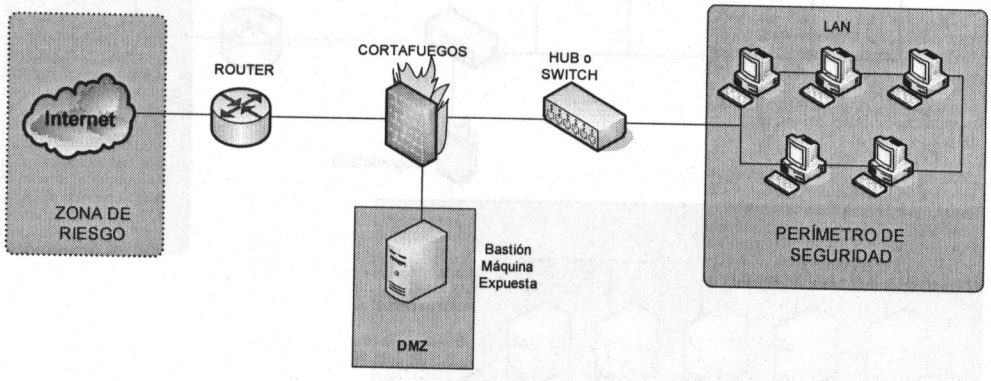


Figura 2.3 Esquema de un cortafuegos entre red local e Internet con zona DMZ para servidores expuestos

En la zona desmilitarizada se pueden poner tantos servidores como se necesiten. Con esta arquitectura, se permite que el servidor sea accesible desde Internet de tal forma que si es atacado y se gana acceso a él, la red local sigue protegida por el cortafuegos.

Esta estructura de DMZ puede hacerse también con un doble cortafuegos (aunque como se ve puede usar un único dispositivo con al menos tres interfaces de red). El esquema sería como este:

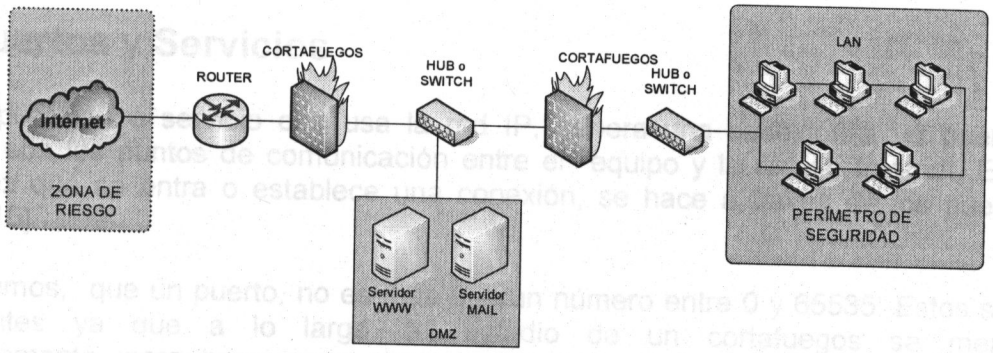


Figura 2.4 Esquema de un cortafuegos entre red local e Internet con zona DMZ para servidores expuestos creado con doble cortafuegos

Los cortafuegos se pueden usar en cualquier red. Es habitual tenerlos como protección de Internet en las empresas, aunque ahí también suelen tener una doble función: controlar los accesos externos hacia el interior y también los internos hacia el exterior; esto último se hace con el cortafuegos o frecuentemente con un proxy (que también utiliza reglas, aunque de más alto nivel).

También en empresas con muchos servidores alojados, lo normar es encontrarnos con uno o más cortafuegos ya sea filtrando toda la instalación o parte de ella:

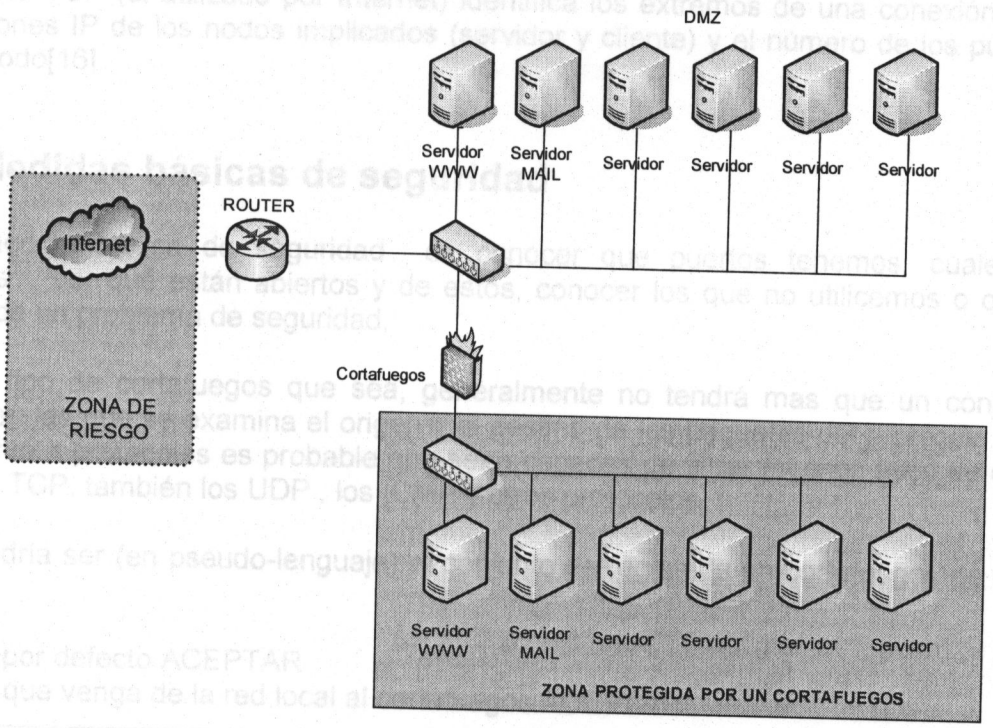


Figura 2.5 Esquema de un cortafuegos entre redes en el que solo se filtra

Este podría ser el esquema de un ISP[11] en el que se coloca el cortafuegos para proteger determinados servidores como servicio adicional. Este se pone entre el resto de la red y la zona de servidores protegidos.

2.6 Puertos y Servicios

Cada aplicación o servicio que usa la red IP, genera una conexión a un puerto. Los puertos son los puntos de comunicación entre el equipo y la red de Internet. Es decir, cada vez que se entra o establece una conexión, se hace a través de los puertos del equipo[15].

Recordemos, que un puerto, no es más que un número entre 0 y 65535. Estos son muy importantes ya que a lo largo del estudio de un cortafuegos se mencionan frecuentemente, para hacer mas clara su relación y uso dentro de los cortafuegos, revisemos el concepto *puerto*.

En el momento que nuestro equipo se conecta a Internet, éste pasa a ser un elemento más dentro de la red, es decir, forma parte de toda la red y como tal se tiene que comunicar con el resto. Para poder comunicarse, lo primero que necesita es tener una dirección electrónica y poder identificarse con los demás. Ésta es la dirección IP, pero esto no es suficiente, ya que en Internet se pueden utilizar muchos y diversos servicios y es necesario poder diferenciarlos. La forma de poder "diferenciarlos" es mediante los puertos.

Los puertos son los puntos de enganche para cada conexión de red que se realiza. El protocolo TCP (el utilizado por Internet) identifica los extremos de una conexión por las direcciones IP de los nodos implicados (servidor y cliente) y el número de los puertos de cada nodo[16].

2.7 Medidas básicas de seguridad

Una medida básica de seguridad es conocer que puertos tenemos, cuales están abiertos[12], por qué están abiertos y de estos, conocer los que no utilicemos o que sean fuente de un problema de seguridad.

Sea el tipo de cortafuegos que sea, generalmente no tendrá mas que un conjunto de reglas en las que se examina el origen y el destino de los paquetes del protocolo TCP/IP. En cuanto a protocolos es probable que sean capaces de filtrar muchos tipos de ellos, no solo los TCP, también los UDP , los ICMP y otros protocolos.

Este podría ser (en pseudo-lenguaje) el conjunto de reglas de un cortafuegos de la figura 2.2:

Política por defecto ACEPTAR
Todo lo que venga de la red local al cortafuegos ACEPTAR

11 ISP es el acrónimo de Internet Service Provider , Proveedor de Servicios de Internet

12 Un puerto abierto significa que está preparado para establecer una conexión o ya la está estableciendo con una dirección remota

Todo lo que venga de la IP de mi PC al puerto tcp 22 ACEPTAR
Todo lo que venga de la IP de la PC del jefe al puerto tcp 1863 ACEPTAR

En resumen lo que hace es:

- Habilita el acceso a los puertos de administración a determinadas direcciones IP privilegiadas
- Deniega el acceso desde el exterior a puertos de administración y a todo lo que este entre los puertos 1 y 1024.

Para la implementación de un cortafuegos hay dos formas:

1) Política por defecto ACEPTAR: en un principio todo lo que entra y sale por el cortafuegos se aceptará y solo se denegará lo que se diga explícitamente.
2) Política por defecto DENEGAR: todo esta denegado, y solo se permitirá pasar por el cortafuegos aquello que se permita explícitamente.

Cabe mencionar que la primera política facilita mucho la gestión del cortafuegos, ya que simplemente nos tenemos que preocupar de proteger aquellos puertos o direcciones que sabemos que nos interesan; el resto no importa tanto y se deja pasar. El problema se presenta cuando no controlemos lo que esta abierto o que en algún momento se instale un software nuevo que abra un determinado puerto. Si la política por defecto es ACEPTAR y no se protege explícitamente, podemos poner en riesgo la seguridad de la red.

Ahora bien, si la política por defecto es DENEGAR, el cortafuegos se convierte en un auténtico muro infranqueable. El problema es que es mucho más difícil preparar un cortafuegos así y se debe tener mucho cuidado en lo que se tiene que abrir sin caer en la tentación de empezar a poner reglas demasiado permisivas.

Algo importante es que el orden en que se ponen las reglas del cortafuegos es determinante. Normalmente cuando hay que decidir que se hace con un paquete se va comparando con cada regla hasta que se encuentra un que le afecta, y se hace lo que dicte esta regla (aceptar o denegar); después de eso **no se mirarán más las reglas** para ese paquete. ¿Cuál es el problema? Si agregamos reglas muy permisivas entre las primeras reglas del firewall, puede que las siguientes no se apliquen y no sirva de nada nuestra protección.

2.8 Como están relacionados con la seguridad de una red

Los sucesos que amenazan la confiabilidad, la integridad y la disponibilidad de los datos pueden surgir accidentalmente o a propósito, generalmente con pretensiones maliciosas.

Ciertamente, una estrategia eficaz de defensa de la red debe ser capaz de enfrentarse a ataques novedosos. Sin embargo, la mayoría de los ataques son relativamente toscos y predecibles. Garantizando que la red es inmune (o al menos muy resistente) a estos ataques se puede dirigir la atención a amenazas menos familiares.

2.8.1 El nivel de amenaza

Históricamente, la mayor amenaza a la seguridad de la información de una organización ha surgido de la propia organización. Empleados poco honestos o descontentos han sido los responsables de la mayoría de los crímenes relacionados con la informática. Teniendo en cuenta el valor económico de la s pérdidas, los delitos cometidos por gente de dentro puede continuar siendo la mayor amenaza. Sin embargo, si los delitos los medimos por número de ataques intentados y logrados, la gente externa parece haberse convertido en la mayor amenaza.

Otra razón para el aumento de los delitos informáticos es, naturalmente, la llegada del Internet. Antes del Internet un aspirante a atacante requería acceso físico a los recursos de información de una organización para ocasionar daños. Los recursos de información de muchas organizaciones ahora están disponibles a través de Internet y en la Web.

Es en estos puntos en donde esta la estrecha relación con los cortafuegos:

- La información importante y sensible puede estar protegida por contraseñas y algunos otros dispositivos para que no sea accesible al público, sin embargo, un atacante hábil podría sortear estas medidas preventivas.
- El delito informático ya no está representado por los actos de alguien de dentro que actúa por venganza o por disgusto, ahora los ataques pueden surgir de cualquier lugar del mundo.

Podemos encontrar la importancia de la aplicación o implementación de un cortafuegos en los puntos señalados anteriormente, que si bien, no son los únicos, son los más claros en la representación del por qué protegerse.

Vemos que los ataques no son únicamente externos, también ocurren desde el interior de una organización, por eso es conveniente que la implementación de un cortafuegos, no solo proteja, sino que también debe ser capaz de mantener alejado el tráfico de ciertas partes de nuestra red que se consideren más sensibles o que contengan información muy privilegiada.

Los cortafuegos son cada vez más necesarios en nuestras redes, pero todos los expertos recomiendan que no se use **en lugar de** otras herramientas, sino **junto** a ellas. Mencionamos algunas:

Squid:

Squid es un programa que sirve de Proxy-Caché de Internet, lo que significa que si se accede más de una vez a una página, ésta se almacena en el disco duro, si no la encuentra, la buscará en Internet.

Esto acelera la navegación a Internet y normalmente se usa en servidores que se conectan a Internet para que naveguen algunos equipos a través de una conexión.

Dansguardian:

Es un módulo de seguridad permite configurar a un *proxy* con filtrado de contenidos.

Danguardian cuenta con una serie de listas para prohibir o permitir el paso de páginas web. Estos archivos se caracterizan por clasificar los contenidos según palabras clave o incluso frases completas, pero también facilita que cualquier característica de la arquitectura de Internet se pueda usar para realizar con más precisión esta tarea. Así, es posible filtrar por IP, dominio (o partes de uno), URL.

La configuración de Dansguardian y Squid permite configurar una máquina GNU/Linux para que actúe como una pasarela segura hacia los contenidos de Internet.

En el segundo capítulo se habla acerca de que es un cortafuegos, así como sus partes y características principales. Además, se mencionan los aspectos más valiosos de una organización, y que desde el punto de vista empresarial son; la confidencialidad, integridad y disponibilidad.

Existen distintos esquemas y configuraciones para aplicar un cortafuegos, estos dependerán de las necesidades y servicios que se manejan dentro de la propia organización y que se aplican de acuerdo al grado de protección que se desea tener.

También está el tema de los puertos y servicios ya que cada aplicación o servicio que se utiliza, genera una conexión a un puerto.Estos son muy importantes, ya que dentro de el estudio y aplicación de un cortafuegos son utilizados frecuentemente.

En el siguiente capitulo se habla mas detalladamente de los cortafuegos basados en la herramienta de linux, Netfilter, específicamente con iptables. Estos serán parte del objeto del proyecto y veremos sus directivas para la seleccion de paquetes.

CAPITULO III

NETFILTER

3.1 ¿Qué es Netfilter?

El kernel de Linux presenta un subsistema de red muy poderoso llamado *Netfilter.* El subsistema Netfilter proporciona un filtrado de paquetes con vigilancia continua o sin ella. También tiene la habilidad de cortar la información IP de cabecera para enrutamiento avanzado y gestión del estado en que se encuentra la conexión y es controlado a través de la utilidad *iptables.*

El poder y flexibilidad de Netfilter es implementado a través de la interfaz de *iptables.* Esta herramienta de línea de comandos es similar en sintaxis a su predecesor *ipchains*; sin embargo, iptables utiliza el subsistema Netfilter para mejorar la conexión de la red, inspección y procesamiento, además, presenta funcionalidades como: registro avanzado, acciones previas y posteriores al enrutamiento, traducciones de direcciones de red y reenvío de puertos, todo en una interfaz de línea de comandos[17].

Netfilter e Iptables están incorporados en el kernel o núcleo de Linux 2.4.x y superiores. Estas herramientas están habilitadas para hacer filtrado de paquetes, traducción de direcciones de red y puertos (*network address [and port] translation)* NA[P]T y la manipulación de otros paquetes (*mangling). iptables* es una modificación y un fuerte sucesor del sistema anterior de Linux 2.2.x en donde era utilizado *ipchains*[18].

3.1.1 Un poco de historia

El sistema operativo Linux, ha contado con herramientas de filtrado de paquetes (IPFW) incorporadas en su núcleo desde la versión del kernel 1.1. Se mejoró el sistema de filtrado de paquetes para las series 2.0 del kernel y se introdujo la utilidad de configuración *ipfwadm.*

A mediados de 1998, de la mano de Michael Neuling y Rusty Russell aparece la herramienta *ipchains*, incorporada en los kernels de la serie 2.2 y que todavía hoy es utilizada en gran parte de los sistemas Linux.

De nuevo Rusty Russell, a mediados de 1999, aparece con una nueva herramienta de filtrado de paquetes *iptables.* Como lo fue *ipchains* sobre ipfw, *iptables* es una modificación que permite la construcción de reglas más precisas y un mejor aprovechamiento de los recursos.

3.1.2 La arquitectura de Netfilter

Netfilter es una serie de "ganchos" en varios puntos de la pila de un protocolo (a estas alturas IPv4, IPv6). El diagrama de recorrido (idealizado) de IPv4[13] se parece a lo siguiente:

[13] **IPv4** es la versión 4 del Protocolo IP (Internet Protocol). Esta fue la primer versión del protocolo que se implemento extensamente, y forma la base de Internet.

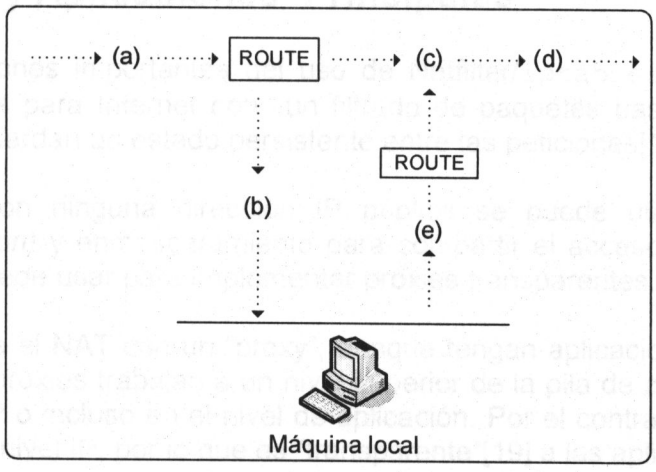


figura 3.1 Un paquete atravesando el sistema Netfilter

Los paquetes entran desde la izquierda: tras haber pasado las sencillas comprobaciones, que no haya sido truncado o detenido, que la suma de control IP es correcta, son pasados al gancho `NF_IP_PRE_ROUTING (a)` del sistema Netfilter.

Luego entran al enrutamiento, en donde se decide si el paquete está destinado a otra interfaz o a un proceso local. El código de enrutamiento puede rechazar paquetes que no se pueden enrutar.

Si está destinado a la propia máquina, se llama de nuevo al sistema Netfilter para el gancho `NF_IP_LOCAL_IN (b)`, antes de ser enviado al proceso (si hay alguno).

Si en cambio, está destinado a otra interfaz, se llama al sistema Netfilter para el gancho `NF_IP_FORWARD (c)`.

Luego el paquete pasa por un gancho final, el gancho `NF_IP_POST_ROUTING (d)`, antes de ser enviado de nuevo al cable.

Para los paquetes creados localmente, se llama al gancho `NF_IP_LOCAL_OUT (e)`. Aquí se puede ver que el enrutamiento ocurre después de haber llamado a este gancho: de hecho, se llama primero al código de enrutamiento (para averiguar la dirección IP y algunas opciones IP), y luego se le llama otra vez si el paquete ha sido alterado.

3.1.3 Selección de paquetes: `iptables`

Se ha construido un sistema de selección de paquetes llamado `iptables` sobre el sistema Netfilter. Es un descendiente directo de `ipchains`. Los módulos del kernel pueden registrar una tabla nueva, e indicarle a un paquete que atraviese una tabla dada. Este método de selección de paquetes se utiliza para el filtrado de paquetes (la tabla 'filter'), para la traducción de direcciones de red (la tabla 'nat') y para la manipulación general de paquetes de enrutamiento (la tabla 'mangle').

3.2 Funciones y Aplicaciones Principales

Una de las aplicaciones importantes del uso de Netfilter/`iptables` es que se pueden construir cortafuegos para Internet con un filtrado de paquetes basado en sistemas o protocolos que no guardan un estado persistente entre las peticiones[18].

Si no se cuenta con ninguna dirección IP pública se puede utilizar NAT (*Network Addresses Translation)* y enmascaramiento para compartir el acceso a Internet, además de que el NAT se puede usar para implementar proxies transparentes.

No debe confundirse el NAT con un "proxy", aunque tengan aplicaciones parecidas. NAT no es un proxy, los proxies trabajan a un nivel superior de la pila de protocolos, ya sea en el nivel de TCP/UDP o incluso en el nivel de aplicación. Por el contrario, el NAT trabaja a un nivel más bajo, a nivel IP, por lo que es "transparente"[19] a las aplicaciones.

Con el uso de las herramientas del sistema "`tc`" e "`iproute2`" ayuda a construir sofisticadas políticas para filtrado y QoS[14].

3.3 Reglas de Filtrado

Las reglas de filtrado se basan en revisar la información que poseen los paquetes en su encabezado, lo que hace posible su desplazamiento en un proceso de IP. Esta información consiste en la dirección IP fuente, en la dirección IP destino, el protocolo de encapsulado (TCP, UDP, ICMP) el puerto fuente, el puerto destino, el tipo de mensaje ICMP, la interfaz de entrada y la de salida del paquete.

Si se encuentra la correspondencia y las reglas permiten el paso del paquete, este será desplazado se acuerdo a la información de la tabla de ruteo, si se encuentra la correspondencia y las reglas niegan el paso, el paquete es descartado. Si estos no corresponden a las reglas, un parámetro configurable por incumplimiento determina descartar o desplazar el paquete[20].

3.3.1 `iptables`

Mencionemos que `iptables` es la interfaz de usuario de Netfilter, el complejo del núcleo de Linux que habilita el filtrado de paquetes. `iptables` proporciona un cortafuegos de filtrado de paquetes con estado con potentes capacidades de inspección y opciones de registro bastante flexibles.

3.3.2 Como funciona `iptables`

El complejo iptables tiene seis cadenas principales que se agrupan en tres tablas:

- `filter`
- `nat`
- `mangle`

[14] QoS, *Quality of Service*

3.3.2.1 La tabla `filter`

Esta tabla 'filter', nunca altera los paquetes: solo filtra.

Una de las ventajas de iptables sobre ipchains es que es pequeño y rápido, y se engancha a Netfilter en los puntos NF_IP_LOCAL_IN, NF_IP_FORWARD y NF_IP_LOCAL_OUT. Esto significa que para cualquier paquete dado, existe un (y sólo un) posible lugar donde pueda ser filtrado. Esto hace las cosas mucho más sencillas. Además, el hecho de que el sistema Netfilter proporcione dos interfaces de entrada (input) y de salida (output) para el gancho NF_IP_FORWARD significa que hay bastantes tipos de filtrado que se simplifican mucho.

La tabla filter efectúa las operaciones que comprueban el contenido de los paquetes y los acepta o bloquea según las directivas. La tabla `filter` tiene tres cadenas:

- FORWARD
- INPUT
- OUTPUT

La cadena FORWARD de `iptables` comprueba los paquetes que están siendo reenviados desde una interfaz de red a otra. Las cadenas INPUT de `iptables` se utilizan sólo para paquetes enviados al host cortafuegos. Análogamente, la cadena OUTPUT se utiliza sólo para paquetes enviados por el host cortafuegos.

3.3.2.2 La tabla `nat`

El dominio de la tabla `nat`, se alimenta de paquetes mediante tres ganchos de Netfilter: para los paquetes no locales, los ganchos NF_IP_PRE_ROUTING y NF_IP_POST_ROUTING son perfectos para alteraciones en el destino y el origen, respectivamente. Para alterar el destino de los paquetes locales, se utiliza el gancho NF_IP_LOCAL_OUT.

Esta tabla es ligeramente distinta a la tabla `filter` en el sentido de que sólo el primer paquete de una conexión nueva atravesará la tabla: el resultado de este recorrido se aplica luego a todos los paquetes futuros de la misma conexión[21].

La tabla `nat` efectúa las operaciones de Traducción de Direcciones de Red (NAT), incluyendo NAT de destino, NAT de origen y enmascaramiento. La cadena consta de dos cadenas:

- PREROUTING
- POSTROUTING

La cadena PREROUTING efectúa las operaciones de NAT de destino, mientras que la cadena POSTROUTNG se ocupa de las operaciones de NAT de origen y enmascaramiento.

3.3.2.3 La tabla `mangle`

La tabla `mangle` permite modificar cualquiera de los dos campos de cabecera de paquete: tipo de servicio (Type of Service) y tiempo de vida (Time to Live). Además permite marcar paquetes para que sean reconocidos por reglas de cortafuegos posteriores y módulos del núcleo Linux. La tabla `mangle` solo tiene la cadena `PREROUTING`.

3.3.3 Comandos `iptables`

El comando `iptables` proporciona operaciones sobre cadenas y reglas. Las operaciones de reglas son:

- Añadir una regla al principio de una cadena
- Añadir una regla al final de una cadena
- Eliminar una regla
- Reemplazar una regla

Las operaciones de cadenas son:

- Enumerar las reglas asociadas a la cadena
- Vaciar una cadena (es decir, eliminar sus reglas)
- Poner a cero los contadores asociados a una cadena

3.3.4 Reglas iptables

La siguiente figura muestra la estructura de una regla iptables:

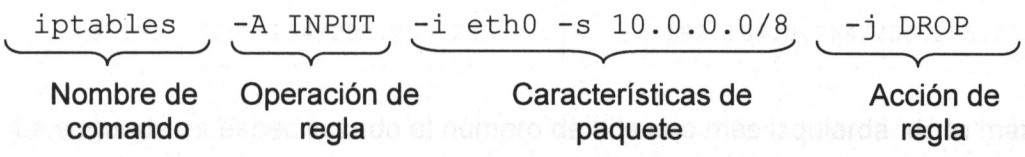


Figura 3.2 Regla iptables

`iptables` permite especificar las siguientes características de paquete relacionadas con la cabecera de paquete IP:

- Protocolo
- Dirección IP de origen
- Dirección IP de destino
- Interfaz de entrada Interfaz de salida
- Indicador de fragmento

Además podemos especificar las siguientes características de cabecera de los tipos de paquetes indicados:

- Puerto de origen
- Puerto de destino

Protocolo

Para especificar el protocolo asociado con un paquete, se usa el indicador `-p` seguido de un determinante de protocolo. El determinante de protocolo puede tomar una de estas formas:

- El nombre del protocolo : `tcp`, `udp` o `icmp`
- El número asignado a un protocolo
- El número cero o la palabra all; ambas opciones designas todos los protocolos

Podemos especificar una serie de protocolos separando cada uno por una coma: `-p tcp, udp` lo que indica que un paquete puede tener un protocolo TCP o UDP.

Dirección IP de origen y de destino

Para especificar la dirección de origen se utiliza el indicador `-s` seguido de la dirección IP. La dirección de destino se especifica de forma muy parecida a la dirección de origen, pero con el indicador `-d`. Si la dirección es la de una red en lugar de un solo host, podemos especificar la máscara de red de dos formas.

La primera es escribiendo después de la dirección IP una diagonal y la máscara de red:

IP de Origen	IP de Destino
`-s 192.168.0.254/255.255.255.0`	`-d 192.168.0.254/255.255.255.0`

La segunda es especificando el número de bits uno más izquierda de la máscara de red:

IP de Origen	IP de Destino
`-s 192.168.0.254/24`	`-d 192.168.0.1/24`

Interfaz de entrada

Para especificar la interfaz de entrada a la que llega un paquete, se usa el indicador `-i` seguido del nombre del dispositivo. Por ejemplo, para especificar que un paquete llegó a la interfaz `eht0`, se utiliza el siguiente determinante:

```
-i eth0
```

Podemos usar el determinante de interfaz de entrada sólo en tres cadenas:

- INPUT
- FORWARD
- PREROUTING

Si se utiliza el determinante en cualquier otro lugar, el comando iptables falla.

Interfaz de salida

La interfaz de salida se especifica de modo muy parecido a la interfaz de entrada. Para especificar la interfaz de salida a la que hay que enviar un paquete se utiliza el indicador `-o` seguido del nombre del dispositivo. Por ejemplo, para especificar que un paquete será enviado a la interfaz `eth0`, se usa el determinante:

```
-o eth0
```

Sólo podemos utilizar el determinante de interfaz de salida en tres cadenas:

- OUTPUT
- FORWARD
- POSTROUTING

Indicador de fragmento

Para especificar que está establecido el indicador de fragmento en la cabecera IP, se utiliza el indicador `-f`. Cuando un paquete está fragmentado, todos los datagramas excepto el primero llevan el indicador de fragmento.

Puerto de origen

Los datagramas `TCP` y `UDP` tienen asociado un puerto de origen. Para especificar el puerto de origen de un paquete, se usa el determinante `--sport`. Por ejemplo, para especificar que el puerto de origen de un paquete es UDP 53, se usan estos determinantes:

```
-p udp --sport 53
```

El determinante `--sport` no es reconocido a menos que se combine con un determinante de protocolo que especifique el protocolo `TCP` o `UDP`. Otra opción es especificar un rango de puertos, escribiendo el puerto inicial, el signo de dos puntos y el puerto final. Por ejemplo, el segundo determinante coincide con los puertos de origen TCP y UDP del 0 al 1023:

```
-p tcp,udp --sport 0:1023
```

Puerto de destino

El puerto de destino se especifica de forma muy parecida al puerto de origen. Para especificar el puerto de destino de un paquete, se usa determinante `--dport`. Por ejemplo, para especificar que el puerto de destino de un paquete es UDP 53, se usan estos determinantes:

```
-p udp --dport 53
```

El determinante `--dport` no es reconocido a menos que se combine con un determinante de protocolo que especifique el protocolo `TCP` o `UDP`. Otra opción es especificar un rango de puertos, escribiendo el puerto inicial, el signo de dos puntos y el puerto final. Por ejemplo, el segundo determinante coincide con los puertos de origen TCP y UDP del 0 al 1023:

```
-p tcp,udp --dport 0:1023
```

3.3.5 Acciones `iptables`

Las reglas iptables pueden invocar a muchas más acciones y estas se invocan con el indicador `-j`. por ejemplo, la acción `ACCEPT` se invoca de esta forma:

```
-j ACCEPT
```

Las acciónes `iptables`, tal como se llaman las acciones de cortafuegos `iptables`, son:

- `ACCEPT`
- `DROP`
- `REJECT`

Además, `iptables` proporciona acciónes que prestan soporte a Traducción de Direcciones de Red (NAT):

- `DNAT`
- `MASQ`
- `REDIRECT`
- `SNAT`

`ACCEPT`

La acción `ACCEPT` provoca que un paquete pase a la siguiente cadena en la ruta de paquetes. La acción no significa que el paquete sea aceptado, pues una cadena posterior puede bloquear ese paquete.

`DROP`

La acción `DROP` provoca que un paquete sea bloqueado. El paquete no atraviesa ninguna cadena posterior de la ruta de paquetes. No se envía un paquete de error al remitente del

paquete desechado.

La acción REJECT, que sí envía un paquete de error al remitente, es un método más cortés para bloquear paquetes, pues el paquete de error evita que el remitente vuelva a intentar establecer la conexión enviando mensajes adicionales o aguardando que termine el tiempo de espera. Sin embargo, en la mayoría de los casos, se prefiere configurar los hosts sensibles en el llamado modo silencioso. La acción DROP es apropiada en tales casos.

REJECT

La acción REJECT provoca que un paquete sea bloqueado y se envíe un paquete de error al host que envió el paquete rechazado. La acción REJECT sólo es válida en las siguientes cadenas integradas:

- INPUT
- FORWARD
- OUTPUT

3.3.6 Acciones de traducción de direcciones de red

Acción DNAT

La acción DNAT especifica la traducción de direcciones de red de destino. Se utiliza con el indicador `--to-destination`, que determina la dirección IP de destino a sustituir en el paquete.

Acción MASQUERADE

La acción MASQUERADE especifica el enmascaramiento, una forma especial de traducción de direcciones de red de origen. A menudo se utiliza con el indicador `--to-ports`, que determina el puerto de origen a sustituir en el paquete.

Acción REDIRECT

La acción REDIRECT especifica una forma especial de traducción de direcciones de red de destino que redirige los paquetes al host cortafuegos. A menudo se utiliza con el indicador `--to-ports`, que determina el puerto de destino a sustituir en el paquete.

Acción SNAT

La acción SNAT especifica la traducción de direcciones de red de origen. Se utiliza con el indicador `--to-source`, que determina la dirección IP de origen a sustituir en el paquete.

3.3.7 Un cortafuegos `iptables` simple

A continuación se muestra un cortafuegos iptables simple pero completo:

```
#####################################################################
#!/bin/sh

echo -n Aplicando Reglas del cortafuegos

## FLUSH de reglas
iptables -F
iptables -X

## Establecemos políticas por defecto
iptables -P INPUT ACCEPT
iptables -P OUTPUT      ACCEPT
iptables -P FORWARD     ACCEPT

## Se empieza a filtrar
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -p udp -s 192.168.0.12/24 --sport 53 -j ACCEPT
iptables -A INPUT -p udp - j REJECT

#####################################################################
```

El funcionamiento de este cortafuegos es que bloquea todos los paquetes UDP que provengan de la dirección de origen `192.168.0.12` con una máscara de subred de 24 bits, excepto los que viene del puerto UDP 53, que podrían ser respuestas a las consultas hechas a un servidor de nombres de dominio.

Netfiler viene presente en el kernel de Linux, este proporciona un filtrado de paquetes y es controlado a través de la utilidad iptables. Vimos como funciona iptables, cual esla fomra para utilizar sus reglas, en las que que se puede elegir entre permitir el paso de ciertos paquetes provenientes de alguna direccion IP especifica, o de algun puerto que elijamos, así como de la interfaz que se decida permitir su slaida o entrada de tráfico.

Es muy importante mencionar que el grado de dificultad de su configuración no va acompañado de ser precisamente lo mas seguro posible, el cortafuegos se ha de manejar acorde a las políticas internas de cada organización, por lo que aun cuando fuera un cortafuegos simple, puede ser bastante seguro, siempre y cuando cubra las necesidades requeridas.

Para el próximo capitulo, está la administración de ancho de banda. Aunque un cortafuegos puede ser util para la seguridad de la red, tiene funciones limitadas, puesto que su finalidad es la de filtrar paquetes y decidir si los acepta o no, de acuerdo a las politicas que se han designado previamente. Ahora bien, el ancho de banda puede llegar a ser utilizado sin ningun tipo de medida, aun cuando la red pueda ser muy segura con el cortafuegos, por lo que para complementar una buena administración, podemos manejar el ancho de banda y así limitar su consumo.

4.1 Definición de Ancho de Banda

Este parece ser un tema crítico, especialmente cuando estamos hablando de la administración de una red y la importancia de conectarse o tener acceso a Internet. Se supone que cuanto más ancho de banda, más rapidez de acceso se tiene, pero ¿realmente que significa esto? El ancho de banda se suele asemejar al diámetro de una tubería que sirve para canalizar el flujo de datos. Pero esta simplificación suele ser excesiva. En principio, el ancho de banda es la capacidad de una línea para transmitir información[22].

> *El ancho de banda es la máxima cantidad de datos que se pueden transmitir por un canal de comunicación en un momento dado, normalmente medido en segundos. Cuanto mayor sea el ancho de banda, más datos podrán circular por ella en un segundo[23].*

Los enlaces y conexiones a Internet hoy en día son muy veloces, y es muy común que nos encontremos con enlaces multipropósitos, es decir, con conexiones que nos sirven de soporte para varios tipos de servicio, como por ejemplo, Internet, correo corporativo, video conferencia, etc. Lo anterior implica tratar de dividir el ancho de banda que es utilizado por cada uno de estos servicios y tener así una mejor distribución y administración de su uso.

Aunque las redes estén configuradas para trabajar, por ejemplo, a 100 Mbps o 10 Mbps, no siempre es posible alcanzar estas velocidades de transmisión, debido a la gran cantidad de colisiones que presenta *ethernet*, que es la topología más utilizada en la actualidad en redes de área local . Debido a esto es conveniente tener bien definido que servicios son los fundamentales y cuanto ancho de banda requieren.

No sólo es importante definir las prioridades en cuanto a los servicios, también es importante definir las prioridades que tienen las diferentes subredes de la empresa, por citar un ejemplo, no se puede asignar el mismo ancho de banda a áreas como desarrollo o la gerencia y a áreas como atención al cliente o archivo.

No se puede permitir que el gerente de la empresa participe en un video conferencia "entrecortada" porque un trabajador aburrido en el archivo baja los últimos MP3 desde Internet. Y supongamos además, que el trabajador aburrido decide enviárselos a sus amigos dentro de la empresa por correo corporativo, la red comenzará a estar saturada debido al excesivo uso de ancho de banda ocupado por los correos electrónicos llenos de archivos de gran tamaño. Para evitar todo esto, es necesario implementar una buena administración de recursos que resulte conveniente.

4.1.1 Herramientas de Administración

Para lograr el objetivo, que es el de lograr una buena administración de ancho de banda no se va a utilizar ningún software especial ni costoso, sólo se va a utilizar una herramienta que viene incluida en la últimas distribuciones de Linux (kernel 2.4.x y superiores). Esta herramienta es un controlador de tráfico llamado *tc*.

Este tipo de herramientas, por formar parte del propio sistema operativo tiene un elevado nivel de eficiencia y precisión. Además de ser muy eficiente, es muy sencillo anular su efecto ya que con una simple línea, se anulan los efectos producidos.

4.2 La utilería `tc`

El kernel de Linux contiene muchas utilerías para poder llevar a cabo diferentes tareas con objetivos que nos sean útiles para tareas específicas. Para poder controlar el ancho de banda, que es el objetivo de este capítulo, podemos encontrar la utilería `tc` - *traffic control.*

La utilería `tc`, manipula las opciones de control de tráfico. Básicamente `tc` es utilizada para configurar el control de tráfico en el kernel de Linux.

4.2.1 Nombre

`tc` - muestra / manipula las opciones de control de trafico.

4.2.2 Descripción

`tc` es utilizado para configurar el Control de Trafico en el núcleo de Linux. El control de tráfico consiste en lo siguiente:

- **SHAPING (Controlar o Ajustar):** Cuando el trafico es controlado (shaped), este rango de transmisión está bajo control. Controlar puede ser más que limitar el ancho de banda disponible – esto es utilizado también para aligerar las cargas en el tráfico para un mejor comportamiento de la red. También se conoce como el proceso de retrasar paquetes antes de que salgan para hacer que el tráfico sea conforme a una tasa máxima configurada. El control ocurre en la salida de tráfico.

- **SCHEDULING (Calendarización):** Por medio de la programación en la transmisión de paquetes es posible mejorar la interactividad de tráfico que se necesita, mientras se puede garantizar el manejo del ancho de banda de manera libre. Es decir, una qdisc[15] puede, con la ayuda de un clasificador, decidir que algunos paquetes salgan antes que otros. Esto es el proceso que se le denomina *scheduling* y también se le denomina *reordenamiento.*

- **POLICING (Políticas):**Cuando se realiza el control para repartir la transmisión de tráfico, las políticas se aplican o pertenecen al tráfico de entrada. La aplicación de las políticas es en la llegada del tráfico. Se utiliza para descartar paquetes para que el tráfico se mantenga por debajo de un ancho de banda configurado.

- **DROPPING (Rechazar):**El tráfico que exceda el ancho de banda es rechazado o denegado inmediatamente, y esto puede ocurrir en la salida o llegada de tráfico.

[15] `qdisc`: es un algoritmo que controla la cola de un dispositivo, sea de entrada (ingress) o de salida (egress)

El procesamiento del tráfico es controlado por tres tipos de objetos:

- qdiscs
- classes
- filters

QDISCS

qdiscs es la abreviación de *queueing discipline* (disciplina de colas) y es elemental para entender el control de tráfico. Siempre que el kernel necesite enviar un paquete para una interfaz, esta es encolada (enqueued) para que el qdisc la configure para la interfaz. Inmediatamente después, el kernel trata de tomar tantos paquetes desde qdisc como le sea posible, para dárselos a la interfaz de red. Una qdisc funciona como un buffer de tráfico cuando las interfaces de red no pueden atenderlas momentáneamente.

CLASSES

Algunas qdiscs pueden contener clases, las cuales contienen otras qdiscs, el tráfico se puede entonces encolar(enqueued) en cualquiera de las qdiscs internas. Cuando el kernel trata de desencolar(dequeued) un paquete de una qdisc con clase este puede venir de cualquiera de las clases. Una qdisc puede por ejemplo, dar prioridad a ciertas clases de tráfico intentando desencolarlos(dequeue) de ciertas clases antes de otras.

FILTERS

Un filtro es utilizado por una qdisc con clase[16] para determinarse en cuál clase un paquete podría ser encolado (enqueued). Siempre que llegue el tráfico en una clase con subclases, este necesita ser clasificado. Se pueden emplear varios métodos para hacerlo, así pues, uno de éstos son los filtros.

Todos los filtros unidos a la clase son invocados, mientras que uno de ellos regresa con el veredicto. Si ningún veredicto fue hecho, otros criterios pueden estar disponibles.

Es importante notar que los filtros residen dentro de las qdiscs y no deciden lo qué sucede.

4.2.3 Principios de Funcionamiento

Antes de hablar sobre sintaxis es conveniente explicar los términos y los elementos que hacen que la organización del "limitador" sea sencilla y eficiente.

Es muy importante antes de empezar a ajustar algo, saber los límites propios de lo que

[16] Una qdisc con clases tiene múltiples clases. Algunas de ella contienen otras qdisc que a su vez pueden tener otras clases, pero no necesariamente.

queremos tasar, por citar un ejemplo, no tendría sentido ajustar la velocidad final de un automóvil en 200 km/hr cuando su velocidad máxima es de 140 km/hr. En este caso pasa exactamente lo mismo, si se cuenta con un enlace cuya velocidad es de 64 Kbps, de nada serviría ajustar el limitador en 128 Kbps o en 256 Kbps, ya que la línea seguiría saturada siempre[25].

Hecha la aclaración, se puede empezar a dar definiciones sobre cuanto ancho de banda se habilitará para determinados servicios, usuarios o subredes.

4.2.4 Configurando `QDISCS` sin clases

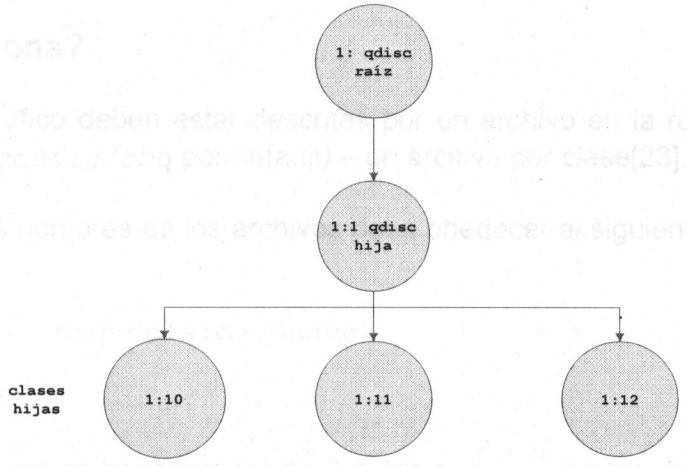
En la ausencia de `qdiscs` con clases, las `qdiscs` sin clase pueden únicamente ser unidas o adjuntarse a la raíz de un dispositivo. Su sintaxis completa es:

```
tc qdiscs add dev DEV root QDISC
```

donde:

`DEV`: es la abreviación de `DEVICE` y es el dispositivo por que el que atraviesa el tráfico. Ejemplo: `eth0, eth1 , ..., ethX`

`QDISC`: son los controladores y consisten en dos partes, un número mayor y un número menor `<mayor>:<menor>`, esto sirve para clasificar o jerarquizar las clases.



Para remover, se utiliza

```
tc qdisc del dev DEV root
```

donde:

`DEV`: es la abreviación de DEVICE y es el dispositivo por que el que atraviesa el tráfico. Ejemplo: eth0, eth1 , ..., ethX

4.2.5 QDISCS con clases

Las qdiscs con clases son muy útiles si se tienen diferentes tipos de tráfico a los que se quiere dar una un tratamiento separado. Una de las qdiscs con clases se denomina *CBQ* "*Class Based Queueing*", generalmente se le menciona de manera frecuente y la gente suele identificar el encolado con clases solo con CBQ.

4.2.6 El flujo dentro de las qdiscs con clases

Cuando entra tráfico dentro de una qdisc con clases, hay que enviarlo a alguna de las clases que contienen (se necesita "clasificarlo"). Para determinar qué hay que hacer con un paquete, se consulta a los "filtros". Es importante saber que los filtros se llaman desde dentro de una qdisc y no al revés.

Los filtros asociados a esa qdisc devuelven entonces una decisión, y la qdisc la usa para encolar el paquete en una de las clases. Cada subclase puede probar otros filtros para ver si se imparten más instrucciones.

4.3 La qdisc CBQ

CBQ es la qdisc más compleja disponible y probablemente la mas difícil de configurar correctamente. Esto se debe a que el algoritmo CBQ no es tan preciso.

4.3.1 ¿Como funciona?

Todas las clases de tráfico deben estar descritas por un archivo en la ruta $CBQ_PATH *directorio* (*/etc/sysconfig/cbq* por default) – un archivo por clase[23].

La configuración de los nombres de los archivos debe obedecer al siguiente mandato para el formato:

```
cbq-<clsid>.<name>
```

donde:

- clsid es un numero hexadecimal de 2 bytes que esta dentro del rango <0002-FFFF> (que de hecho es el ID de una clase CBQ)
- <name> es el nombre de la clase, cualquiera que ayude a distinguir la configuración de los archivos.

Ejemplo de un nombre de configuración válido: cbq-1280.mi_primer_script

Los archivos de configuración pueden contener algunos de los siguientes parámetros:

```
##  Parámetros de los dispositivos

##  DEVICE=<ifname>,<bandwidth>,[<weight>]        obligatorio
DEVICE=eth0,10Mbit,1Mbit
```

<ifname> es el nombre de la interfaz de la que se quiere controlar el tráfico que entra, ejemplo: eth0

<bandwith> es el ancho de banda físico del dispositivo, por ejemplo: ethernet 10Mbit o 100Mbit.

<weigth> es el parámetro para regular y debe ser proporcional al ancho de banda <bandwith>.
La regla general es: <weight> = <bandwith> / 10.

```
## Parámetros de clases

## RATE=<speed>                                    obligatorio
RATE=5Mbit
```

Es el ancho de banda asignado a la clase. El tráfico que atraviesa la clase es ajustado para conformar el rango especificado. Se puede utilizar Kbit, Mbits, bps, Kbps y Mbps. Si no se especifica una unidad, bits/sec son utilizados.Hay que notar que bps quiere decir bytes por segundo, no bits.

```
## WEIGHT=<speed>                                  obligatorio
WEIGHT=500Kbit
```

El parámetro para regular tiene que ser proporcional al rango RATE. La regla general es:
WEIGHT ~= RATE / 10

```
## PRIO=<1-8>                                  opcional,default 5
PRIO=5
```

Es la prioridad del tráfico de la clase. Más alto es el número o más bajo según la prioridad. La prioridad ajustada en 5 es la más adecuada.

```
## PARENT=<clsid>              opcional, no fijado por default
PARENT=1280
```

Especifica el ID de la clase padre a la cual se desea que esta clase sea adjuntada. Utilizando este parámetro y ordenando cuidadosamente los archivos de configuración, es posible crear las estructuras jerárquicas simples de las clases de CBQ.

```
## Parámetros de filtrado

## RULE= [[saddr[/prefix]][:port[/mask]],]
        [daddr[/prefix]][:port[/mask]]
```

Se pueden utilizar múltiples campos `RULE` para su configuración. La opción `port mask` solo debe ser utilizada por usuarios avanzados que conozcan y entiendan el funcionamiento del trabajo de los filtros.

Ejemplos:

```
RULE=10.1.1.0/24:80
```

selecciona el tráfico que venga del puerto 80 en la red `10.1.1.0`

```
RULE=10.2.2.5
```

selecciona el tráfico que venga de cualquier puerto en un solo host, `10.2.2.5`

```
RULE=:25,10.2.2.128/26:5000
```

selecciona el tráfico que venga de cualquier IP dentro de los puertos 25 al 5000 en la red `10.2.2.128`

Ejemplo de un archivo de configuración:

```
# cbq-1280.mi_primer_script
#
# --------------------------------------------------------------
DEVICE=eth0,10Mbit,1Mbit
RATE=128Kbit
WEIGHT=10Kbit
PRIO=5
RULE=192.168.1.0/24
# --------------------------------------------------------------
#
```

Citando el ejemplo anterior, La configuración nos muestra que controlaremos el tráfico en `10Mbit` en el dispositivo `eth0` y el tráfico vendrá de la red `192.168.1.0` y será procesada con una prioridad 5 y su rango será controlado o ajustado en `128Kbit`.

El ancho de banda es la capacidad de una línea de transmitir información, pero esto no solo es importante saberlo, se hace necesario definir que servicios tienen mayor prioridad, dentro de las organizaciones es algo complejo decidir a que o quienes se les permite un mayor o menor uso del ancho de banda.

Sin embargo, es necesario implementar una buena administración de recursos para que su uso resulte conveniente. Para este caso se va a utillizar la herramienta tc que viene incluida en las ultimas distribuciones de linux, está ayuda a manipular las opciones para que el control de tráfico sea más eficiente.

Se habla también de CBQ que es el que nos permite controlar las clases de tráfico, así como la manera en que funciona y los parametros que habrán de utilizarse dentro de los scripts de configuración.Para finalizar, el quinto capitulo tiene el caso práctico donde aplicaremos lo mencionado en los capitulos anteriores.

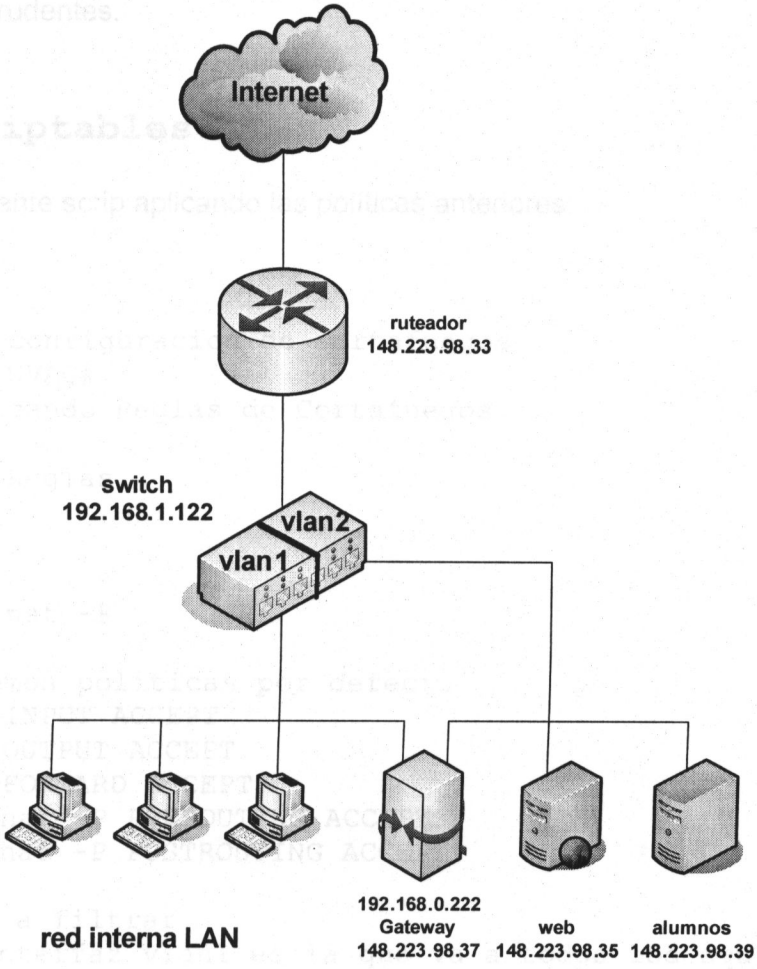
CAPITULO V

CASO PRÁCTICO

5.1 Descripción de la red

La red de la Universidad Vasco de Quiroga se encuentra conectada a Internet a través de un router con una velocidad de 256 kbps que es la proporcionada por el proveedor del servicio d einternet. Adicionalmente, el router se conecta a un switch el cual tiene dos redes virtuales `vlan1` y `vlan2`.

En la `vlan1` se encuentran conectados todos los equipos de la red local (LAN) 192.168.x.x , así como una interfaz del gateway con IP 192.168.0.222, hacia la `vlan2` está la otra interfaz del gateway con dirección 148.223.98.37, el servidor web con una dirección IP 148.223.98.35 y un servidor para alumnos 148.223.98.39.



5.2 Servicios

Los principales servicios utilizados son la navegación en Internet, el uso de mensajería instantánea, así como algunos programas para descargar archivos o música. Siendo lo más utilizado la consulta a páginas web o el correo electrónico personal. También es utilizado un servidor de mensajería instantánea para comunicarse dentro de la red local y el servicio para subir archivos o documentos del personal académico para una posterior consulta por parte de los alumnos.

5.3 Políticas

- Se va a controlar el acceso a los puertos del Squid (3128) a través del cual nos proporciona un proxy para la navegación en Internet.
- Se deniega los posibles ping a `localhost` del servidor.
- Se deniega también, la conexión al messenger de hotmail.
- Los accesos a Samba se deniegan únicamente para pruebas.

Todas las pruebas anteriores quedan aplicadas unicamente mientras se han obtenido resultados, ya que el bloqueo se hace para toda la red, con la finalidad de no perjudicar el acceso a los servicios que se brindan, siendo esto, temporal ya que es posible ir agregando restricciones unicamente para los equipos que el propio administrador de la red considere prudentes.

5.4 Reglas `iptables`

Se creo el siguiente scrip aplicando las políticas anteriores:

```
# !/bin/sh
# Script de configuracion de cortafuegos
# Red Local UVAQ#
echo -n Aplicando Reglas de Cortafuegos...

## Flush de Reglas
iptables -F
iptables -X
iptables -Z
iptables -t nat -F

## Establecemos politicas por defecto
iptables -P INPUT ACCEPT
iptables -P OUTPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -t nat -P PREROUTING ACCEPT
iptables -t nat -P POSTROUTING ACCEPT

## Empezamos a filtrar
# Nota: la interfaz vlan1 es la que va a todos los equipos (PC)
# de la red local LAN

iptables -A INPUT -s 127.0.0.1 -p icmp -j DROP && echo " denegar-
lo ok "

# Reglas para redireccionar peticiones del servicio web el puerto
3128 Squid (proxy transparente)
iptables -t nat -A PREROUTING -i vlan1 -p tcp --dport 80 -j
REDIRECT --to-port 3128  && echo "REDIRECCIONAR A PROXY (WEB) OK "
```

```
#Regla para bloquear el acceso al Samba en servidor
#iptables -A INPUT -s 192.168.0.0 -d IP DEL SERVIDOR -p tcp --
dport 111 -j DROP && echo " DENEGAR  rpcbin OK "
#iptables -A INPUT -s 192.168.0.0 -d IP DEL SERVIDOR -p tcp --
dport 139 -j DROP && echo " DENEGAR  netbios-ssn OK "
#iptables -A INPUT -s 192.168.0.0 -d IP  DEL SERVIDOR -p tcp --
dport 445 -j DROP && echo " DENEGAR  microsoft-ds OK "

#ejemplo para bloquear msn de hotmail
iptables -t mangle -A PREROUTING -p tcp --dport 1863 -j DROP
iptables -t mangle -A PREROUTING -d 63.208.13.126 -j DROP
iptables -t mangle -A PREROUTING -d 64.4.12.200 -j DROP
iptables -t mangle -A PREROUTING -d 64.4.12.201 -j DROP
iptables -t mangle -A PREROUTING -d 65.54.131.249 -j DROP
iptables -t mangle -A PREROUTING -d 65.54.194.118 -j DROP
iptables -t mangle -A PREROUTING -d 65.54.211.61 -j DROP
iptables -t mangle -A PREROUTING -d 207.46.104.20 -j DROP
iptables -t mangle -A PREROUTING -d 207.46.110.2 -j DROP
##


echo "Verifique lo que se aplica con:"
iptables -L -n

########################  Fin de script  ########################
```

5.5 Ancho de banda

El control del ancho de banda se hizo bajo el siguiente criterio:

- Mantener todo el tráfico que llegue hacia la red 192.168.0.0/16, que venga desde Internet y pase a través de la interfaz llamada `vlan1` a una velocidad de 200 Kbit.

- El tráfico que se genere de cualquier IP de la red 192.168.0.0/16 será de 1 Mbit.

Este criterio está basado en el hecho de que el tráfico que sea generado desde el interior, es decir, desde cualquier IP de la red interna, debe tener una velocidad alta, puesto que se hace necesario accesar a diversos servicios que no necesariamente estan ligados al uso del Internet, por ejemplo un servidor de mensajeria instantanea, el servidor de alumnos, así como la trasferencia de archivos entre los equipos.

El tráfico proveniente de Internet ha sido limitado con una velocidad más baja ya que este servicio es muy utilizado, por tanto siempre se encuentraba hasta el máximo de ancho de banda que proporciona el proveedor del servicio de Internet, para navegación y otros servicios que proporciona la web.

En los siguientes scripts está la configuración que se ha aplicado al servidor para que pueda mantener el control del ancho de banda que es consumido.

cbq-240.web

```
# Script para el tráfico que viene de Internet y va a la red
# 192.168.0.0/16 a tráves de vlan1

DEVICE=vlan1,100Mbit,10Mbit
RATE=240Kbit
WEIGHT=24Kbit
PRIO=5
RULE=192.168.0.0/16     #direccion destino
```

cbq-1000.lan

```
# Script para el tráfico que viene de la red local
# 192.168.0.0/16 a tráves de vlan1

DEVICE=vlan2,100Mbit,10Mbit
RATE=10Mbit
WEIGHT=1Mbit
PRIO=5
RULE=192.168.0.0/16,     #direccion de origen
```

5.6 Resultados obtenidos

La figura 5.1 nos muestra como es el comportamiento del tráfico antes de aplicar las pruebas del control de ancho de banda, como se observa el tráfico entrante siempre es igual al saliente, puesto que el tráfico que atraviesa, siempre entra y sale por el gateway a la misma velocidad y unicamente se tenía un script para controlar el ancho de banda; aproximadamente a las 17:00 hrs, se empezó a aplicar el control, por lo que en la figura 5.2 y encerrado en un cuadro rojo, se muestra el momento en el cuál disminuye el tráfico.

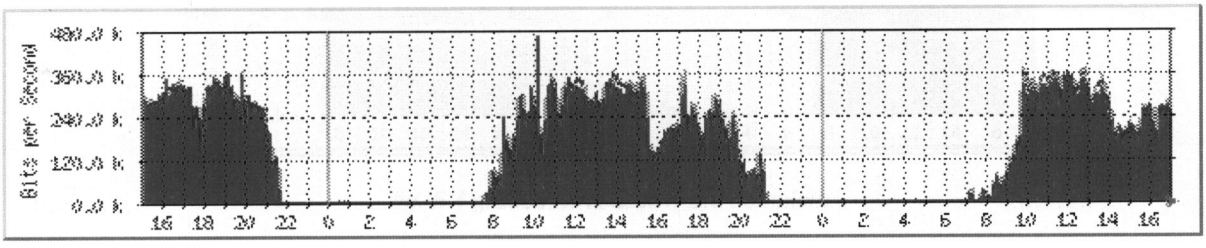


fig. 5.1 gráfica MRTG del tráfico antes de las pruebas
07-Abril-06 17 hrs.

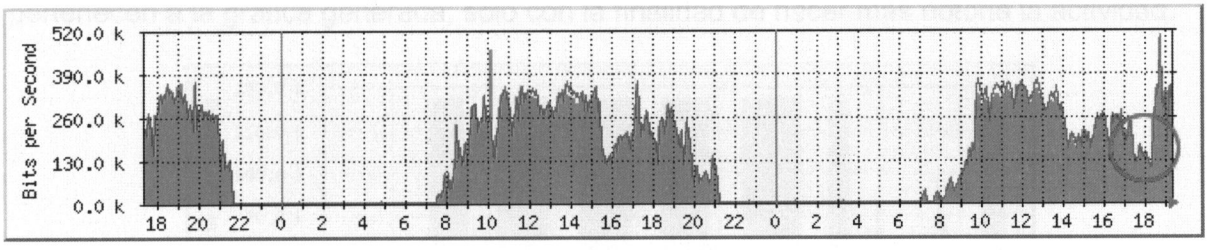


fig. 5.2 gráfica MRTG del tráfico con control de ancho de banda
07-Abril-06 19 hrs.

La siguiente gráfica nos muestra como el consumo de ancho de banda se encuentra hasta el límite, de manera constante aproximadamente a partir de las 7 hrs (1), que es cuando comienza la actividad, permanece hasta las 15 hrs.(2) y comienza a bajar, debido a la hora de salida a comer, a partir de las 16 hrs.(3) vuelve a subir su consumo, quedandose así hasta el momento en que se aplica el control(4), siendo aproximadamente las 18 hrs (se muestra una línea roja, que es independiente del grafico, solo para hacer mas notoria la aplicación del control). El filtro se aplica unicamente durante algunos minutos, ya que se está usando el servidor real de la Universidad.

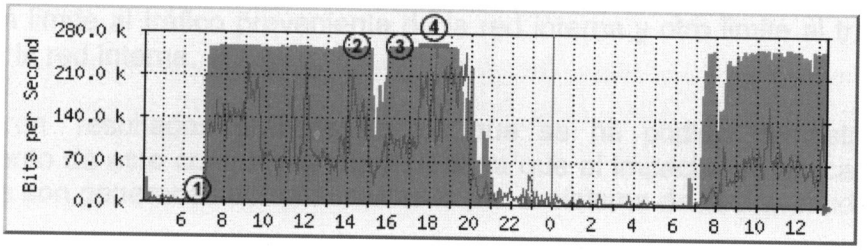


fig. 5.3 gráfica MRTG de medio día

En está gráfica podemos observar nuevamente el control del ancho de banda aproximadamente a las 16 hrs. Para ver más clara la actividad del tráfico en la parte (A) que inicia a partir de las 7 hrs se observa que es constante, con una línea roja, que es independiente de la gráfica, y en donde , aproximadamente a las 18:00 hrs se aplica el control para poder ver, la baja en l actividad del tráfico mismo. En la parte (B), que inicia aproximandose a las 7 hrs. empiezan a aparecer huecos en el tráfico, esto es originado porque se hicieron las pruebas durante tiempos cortos, iniciando y deteniendo el servicio para poder ver en las imágenes el comportamiento de las mismas en un día.

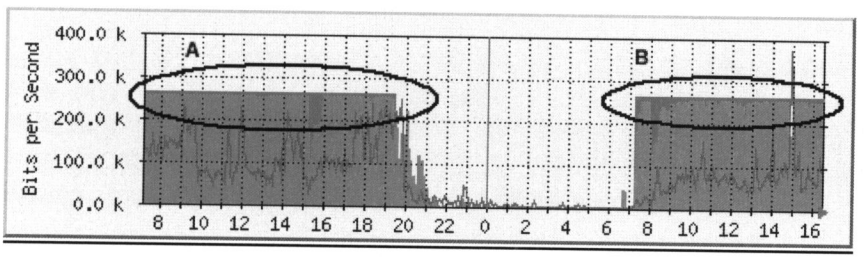


fig. 5.4 gráfica MRTG de 1 día completo

Para la siguiente imágen, el servicio se activa y desactiva durante los 5 días siguientes al inicio de la prueba, para que se guarde el registro y pueda ser mostrado el cambio que ocurrió en el comportamiento del tráfico. Se muestran líneas en color rojo y que no pertenecen a la gráfica generada, solo con la finalidad de hacer mas notoria la actividad.

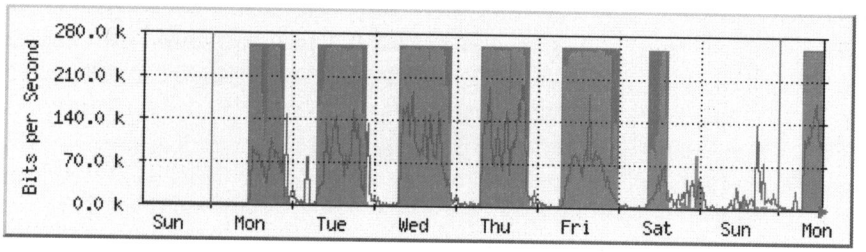


fig. 5.5 gráfica MRTG semana 1

En este caso práctico dentro de la Universidad Vasco de Quiroga, se ha podido aplicar un cortafuegos en el que se han denegado puertos muy comunes, como son: los de messenger de hotmail, los puertos utilizados para el servidor samba, para no permitir hacer ping al localhost, estos son unicamente pruebas puesto que no son puertos que ocasionan problemas, solo fueron elegidos para demostrar el funcionamiento del cortafuegos.

También se comenzó a manipular el control del tráfico en la red, ya que el tráfico siempre ocasionaba una saturación del ancho de banda, y se tomó la decision de asignarle un limite al tráfico proveniente de la red interna y otro limite al tráfico que fuera destinado a la red interna.

Se obtuvieron resultados gráficos en los que se ha podido demostrar el correcto funcionamiento de este control y la manera en la que el impacta su aplicación, dentro de graficas que son generadas automaticamente por el sistema de la Universidad.

CONCLUSIONES

Derivado de la investigación anterior se concluye que en la actualidad las herramientas para controlar y administrar una red son muy útiles para el correcto uso de estas mismas. Debido a la imposibilidad de controlar a cada individuo dentro de una organización, se hace casi indispensable implementar o aplicar medidas que mantengan segura una red puesto que su contacto con el exterior no siempre es del todo seguro y puede ser , hasta cierto punto, incierto.

Para este proyecto se implementa y utiliza el control bajo un cortafuegos y un control de ancho de banda que han resultado benéficos para evitar el uso desmedido de los recursos que se proporcionan en la red, sin caer en la restricción total, es decir, sin bloquear completamente un equipo interno y sin mantenerlo desconectado de la red, dandole unicamente los permisos para hacer uso de los recursos que su trabajo le exige.

Son medidas que se hacen necesarias puesto que no toda las personas que trabajan o que se encuentran dentro de un organismo, hacen un uso adecuado de los servicios, haciendo esto de manera consciente o en ocasiones por algun descuido.

Todo esto se ha podido evitar controlando el ancho de banda, para darles unicamente el rango que les sea necesario, ya que al tener la conexión completamente libre se hizo presente la saturación de la red, ya que son demasiados los usuarios que acceden y navegan en Internet, sin contar aquellos que hicieron descargas de música y videos. Ha sido un resultado favorable puesto que ahora para el acceso a los servicios se cuenta con un control adecuado y el tráfico en la red ahora es menor, no por que esto implique que ya no se use, sino limitando su consumo para poder evitar la saturación.

Cabe mencionar que el implementar un control de ancho de banda con la herramienta cbq, que es la que se utilizó en este proyecto, no es una tarea muy sencilla, hay muy pocas referencias y no se encuentra mucha información al respecto, por lo que dificulta un poco el trabajo, obviamente, cuando no se es del todo experto en esto, lo que implica pruebas y errores, hasta que se obtenga el resultado esperado.

Sin embargo, aunque no hay tanta información al respecto, el HOWTO es útil para poder configurar nuestro servidor Linux, para limitar el ancho de banda y el tráfico entrante y nos ayuda para saber como usar la conección a internet para que sea más eficiente.

Además, se aplicó un cortafuegos que permite mantener alejados a los servidores de aquellos usuarios intrusos. Se considera que estos servidores deben protegerse porque guardan la información mas importante de la Universidad, y también actuan de *proxies* para la navegación en Internet, ademas, de que nos han de servir para evitar el uso de aquellas aplicaciones que puedan generar más tráfico o que no esten permitidas por se ajenas a las labores propias de la institución.

Para concluir, se puede decir que utilizar las herramientas que nos proporciona Linux para la administración de la red, y que para este caso específico fueron *cbq* para controlar el ancho de banda e *iptables* para el cortafuegos, nos han dado resultados óptimos para un buen control y sin estar obligados a pagar por licencias para el uso de software.

REFERENCIAS

[1] Ricardo J. Cárdenes Medina, Linux Networking-concepts HOWTO
http://www.insflug.org/COMOs/conceptos-de-redes-COMO/conceptos-de-redes-COMO-2.html, Abril de 2005.

[2] Andrew S. Tanenbaum, Redes de Computadoras, Editorial Pearson Prentice Hall, Tercera Edición. ISBN 968-880-958-6

[3] Comparación entre el Modelo OSI y el TCP/IP
http://www.bufoland.cl/apuntes/redes/resumen_TCPIP.php
Septiembre de 2005.

[4] *http://www.monografias.com/trabajos13/modosi/modosi.shtml#NIVEL*

[5] Modelo OSI
http://web.frm.utn.edu.ar/comunicaciones/modelo_osi.html#5
Julio de 2005.

[6] Modelo OSI
http://es.wikipedia.org/wikiModelo_OSI
Julio de 2005.

[7] Miguel Alejandro Soto, Protolos TCP/IP,
http://usuarios.lycos.es/janjo/janjo1.html
Mayo de 2005.

[8] Concepto de Puertos
https://www.redes.unb.br/material/APRC_OSDI/UDP.pdf
Septiembre de 2005.

[9] Puertos TCP
http://es.wikipedia.org/wiki/TCP#Puertos_TCP
Septiembre de 2005.

[10] Robert Hart, Direcciones IP reservadas,
http://www.insflug.org/COMOs/PPP-Como/PPP-Como-2.html
Julio de 2005.

[11] Luciano Moreno, Subredes
http://www.htmlweb.net/redes/subredes/subredes_1.html

[12] Redes Frame Relay
http://www.alipso.com.ar/monografias/framerelay/
Agosto de 2005.

[13] Gigabit Ethernet
http://es.wikipedia.org/wiki/Gigabit_Ethernet
Septiembre de 2005.

[14] Introducción a las normas IEEE 802.3 y 802.11
http://gsyc.escet.urjc.es/~esoriano/memoria/node5.html
Septiembre de 2005.

[15] Puertos. Seguridad. Firewall
http://www.apymes.es/cursointinf19.htm
Septiembre de 2005.

[16] ¿Que son los puertos?
http://www.zonagratuita.com/servicios/seguridad/puertos.html
Septiembre de 2005.

[17] *Red Hat Enterprise Linux 4: Manual de seguridad*
http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-sg-es-4/ch-fw.html
Septiembre de 2005.

[18] Netfilter,firewalling, NAT and packet mangling for Linux
http://www.netfilter.org
Septiembre de 2005.

[19] David Pascual Serral. *El módulo Netfilter de Linux: Iptables*
http:// www.redes-linux.all-inone.net/manuales/seguridad/IR-iptables.pdf
Septiembre de 2005.

[20] Firewalls y Seguridad en Internet
http://www.monografias.com/trabajos3/firewalls/firewalls.shtml
Septiembre de 2005.

[21] Rusty Russell, *Linux Netfilter Haking COMO*
http://es.tldp.org/COMO-INSFLUG/COMOs/netfilter-hacking-COMO/netfilter-hacking-COMO-3.html
Septiembre de 2005.

[22] Glosario, Ancho de Banda
http://www.learnthenet.com/spanish/glossary/bandwth.htm
Octubre de 2005.

[23] Antonio Cervantes, *Ancho de Banda*
http://www.caravantes.com/cv/ancho.htm
Octubre de 2005.

[24] cbq.init
http://easynews.dl.sourceforge.net/sourceforge/cbqinit/cbq.init-v0.7.3

25] *Herramientas para el control de ancho de banda en servidores Linux*
http://www.monografias.com/trabajos17/ancho-de-banda/ancho-de-banda.shtml#herram

---