PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
SEDE AMBATO

ESCUELA DE INGENIERÍA DE SISTEMAS

DISERTACIÓN DE GRADO PREVIA LA OBTENCIÓN DEL TITULO DE INGENIERO DE SISTEMAS

"CREACIÓN DE UN FIREWALL PARA EL CENTRO DE CÓMPUTO DE LA PUCESA"

Danilo Oldemar Luna Villacrés
Edgar Fernando Solís Acosta

DIRECTOR DE LA DISERTACIÓN: Ing. Janio Jadán.

AMBATO, 2002

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
SEDE AMBATO

ESCUELA DE INGENIERÍA DE SISTEMAS

DISERTACIÓN DE GRADO PREVIA LA OBTENCIÓN DEL TITULO DE INGENIERO DE SISTEMAS

"CREACIÓN DE UN FIREWALL PARA EL CENTRO DE CÓMPUTO DE LA PUCESA"

DIRECTOR: **Ing. Janio Jadán.**

REVISORES: **Ing. Patricio Medina.** **Ing. Natasha Bayas.**

Danilo Oldemar Luna Villacés
Edgar Fernando Solís Acosta

AMBATO, 2002

Agradecimiento

En primer lugar deseo agradecer profundamente a un gran amigo espiritual que nunca me falla Jesús, que me dio fuerzas y sabiduría en mis momentos más dificiles y me encamino en el camino del bien.

Deseo exteriorizar un agradecimiento muy especial a mis padres cuya tolerancia, educación y excelentes concejos que me han sabido impartir, por lo cuál nunca podré retribuir en toda mi vida.

Agradecer de manera particular a mi hermana Adriana Luna, quien me apoyó en los momentos difíciles ayudándome y dándome fuerzas para salir siempre adelante.

Agradecer a todos mis compañeros y amigos que siempre estuvieron en las buenas y en las malas, quienes confiaron en mi capacidad y me dieron ánimos para sobresalir con éxito en toda mi carrera universitaria.

A todos mis maestros que con paciencia, dedicación y excelentes conocimientos me guiaron al camino del triunfo. En especial a los Ingenieros Janio Jadán, Patricio Medina y Mario Holguín

Danilo Oldemar Luna Villacrés.

Agradecimiento

Quiero agradecer profundamente al amigo que nunca falla, Jesús que me concedió fuerzas en los momentos más difíciles e iluminó mi sabiduría, a Maria Santísima quien me forja el presente.

Agradecer a mis padres cuya tolerancia y educación han tenido sobre mí, nunca podré retribuir lo suficiente a lo largo de toda mi vida. No es posible expresar con simples palabras el efecto que ellos han tenido conmigo.

Agradecer a mis hermanas Rosalva e Iliana Solís Acosta, quienes me apoyaron en los momentos de más necesidad ayudándome a salir siempre adelante.

A Luis, Vinicio Jordán mis primos, Miguel, Yolanda mis mejores amigos quienes confiaron en mi dándome ánimo para sobresalir en mi vida estudiantil.

A todos mis maestros que con paciencia y dedicación me guiaron por la ruta del triunfo. En especial a los Ingenieros Janio Jadán y Patricio Medina.

Edgar Fernando Solís Acosta.

INDICE

INTRODUCCIÓN GENERAL

Los avances tecnológicos han roto muchas barreras y han superando ciertas limitaciones del hombre, con el objetivo que la vida de los seres humanos sea mucho más confortable. La información se ha convertido en uno de los recursos que ha acrecentado su valor, hasta convertirse en uno de los más importantes del nuevo milenio, por lo que su procesamiento, análisis y almacenamiento es uno de los pilares del conocimiento moderno.

El procesamiento de información en la actualidad es un recurso de altísimo valor, lo que ha originado un amplio desarrollo de la industria del Software. Industria que junto con la aparición de Internet han tenido un crecimiento explosivo en la última década. Empresas, Centros Educativos y personas naturales se han ido interconectando a la gran autopista de la información, la cual ha traído muchos beneficios con respecto a la comunicación y transferencia de información. Junto a estos avances y en forma paralela se han producido muchos incidentes de robo de información, fraudes e intrusiones que han generado y siguen generando perdidas cuantiosas en tiempo y dinero. Este fenómeno ha dado origen a la seguridad informática, una área de mucha importancia para Empresas e Instituciones que se encuentran permanentemente conectados a Internet. Con la aplicación de técnicas de seguridad dichas empresas pueden salvaguardar su información de intrusos que quieren apoderarse de ellas con fines maliciosos.

La búsqueda permanente de alternativas a la solución de los problemas que el hombre está enfrentando en la actualidad a través del uso de la tecnología, ha llevado a éste a revertir los pensamientos tradicionales, en teoría, que mediante el consenso científico, se han

Existe muchas alternativas que pueden ser aplicadas, la mayoría de ellas muy costosas, por lo que debemos buscar nuevas alternativas, y es justamente a donde está enfocado nuestra aplicación, construir un Firewall que se adapte a las necesidades del centro de cómputo de la PUCESA, así como su implantación a bajo costo.

El área de Seguridad Informática es muy extensa, motivo por el cual nuestro proyecto se delimitará al desarrollo de un Firewall para el centro de computo de la PUCESA.

INTRODUCCIÓN AL SISTEMA DE SEGURIDAD FIREWALL

La construcción de un Firewall se realiza para el control de intrusos externos que pretendan ingresar al centro de cómputo de la PUCESA. Este proyecto tiene una gran ventaja, al no permitir el paso de intrusos que pretendan vulnerar la información que compete únicamente a estudiantes y personal docente y administrativo de la institución.

El objetivo de este proyecto es presentar una documentación detallada de la construcción e implantación de un Firewall en la PUCESA, que facilite al administrador del centro de cómputo la tarea de proteger a la red informática de la escuela de ingeniería de sistemas de posibles amenazas externas, para ello se detallarán los pasos a seguir para la compilación e instalación del software, aunque la última palabra tendrá siempre la documentación de cada procedimiento en particular. En algún caso la descripción que aquí se hace pudiera ser completa y el administrador tendrá que recurrir a las instrucciones de implementación de cada uno de los procedimientos que conforman el sistema, los cambios serán mínimos y perfectamente documentados dentro de la documentación de cada procedimiento.

En un mundo tan evolutivo como Windows en todas sus versiones, Linux entre otros, el saber compilar aplicaciones llega a ser vital, ya que los programas cambian constantemente y no siempre se disponen de binarios ya compilados, estos pueden que no se adapten a nuestras necesidades. Y desde el punto de vista didáctico, se aprende mucho enfrentándose a este tipo de problemas.

Como es habitual debemos estar preparados adecuadamente para poder enfrentar y resolver problemas que se presentarán a futuro en el convivir diario de esta especialidad, una vez analizado y superado tenemos que actualizar el sistema de acuerdo a las necesidades.

Para evaluar el correcto funcionamiento del Firewall se realizará pruebas de intrusión para determinar si el sistema bloquea al intruso automáticamente, restringiéndole el ingreso a la información de la PUCESA.

Un Firewall es ideal para aplicaciones de red que requiera un funcionamiento robusto y conexión a Internet en forma continua. Este es el caso de los servidores de la PUCESA que poseen Bases de Datos, e información de acceso restringido importante únicamente para la organización en cuestión.

La robustez y eficacia de este sistema ha hecho que empresas busquen las alternativas de crear su propio Firewall para proteger información reservada a costos muy elevados. Nuestro objetivo es obtener la misma robustez y eficacia minimizando los costos.

ESTRUCTURA DEL DOCUMENTO

En el capítulo I se realiza una introducción a la seguridad informática en términos generales, una introducción a aspectos de vulnerabilidad y protección de información, así como la inversión y formas de no ser vulnerado.

En el capítulo II se realiza estudios minuciosos, detallados de Intrusos (piratas), técnicas y programas para vulnerar sistemas operativos, también se estudia generalidades sobre tipos de intrusos.

En el capítulo III se menciona a acerca de Murallas de fuego (Firewall), realizando una introducción general de cómo se puede proteger la información de posibles ataques de Hackers, Crackers, Phreakers.

En el capítulo IV se describe una documentación más completa del funcionamiento de los Firewalls con el objetivo que el administrador tenga un conocimiento apropiado para el manejo del mismo, por lo que recomendamos al usuario interesado analizar cada procedimiento que se ha creado. Además se acompaña un manual técnico debidamente documentado en donde podrá tener acceso a las referencias del sistema.

CAPITULO I

SEGURIDAD INFORMÁTICA

El término seguridad se refiere a resguardar los recursos de un posible daño o peligro de intrusión por parte de personas no autorizadas, es una fianza u obligación de indemnidad a favor de la empresa, cualquier método utilizado para la seguridad de los recursos debe realizarse en forma eficáz para proporcionar los tres principios básicos de la seguridad:

- Integridad
- Disponibilidad
- Confidencialidad

Las responsabilidades y derechos tanto de usuarios como administradores, describen lo que se va a proteger y de lo que se esta tratando de proteger, esto es el primer paso en la implantación de mecanismos de seguridad efectivos.

Atender de manera eficiente la seguridad de una red se hace cada vez más difícil. A pesar de que las herramientas se mejoran día a día, los Intrusos también aumentan su nivel de conocimientos técnicos y de sofisticación. En general, las empresas y las organizaciones son cada vez más conscientes de los riesgos y permanentemente tratan de aumentar los niveles de protección.

1.1. VULNERABILIDADES Y PROTECCIÓN

Los intrusos utilizan diversas técnicas para quebrar los sistemas de seguridad de una red informática. Básicamente buscan los puntos débiles del sistema para poder colarse en ella. El trabajo de los probadores (testers) no difiere mucho de esto. En lo que sí se diferencia, y por completo, es en los objetivos.

Mientras que los intrusos penetran en las redes para dañar o robar información, un probador que podría ser el mismo administrador del sistema lo hace para poder mejorar los sistemas de seguridad.

Al conjunto de técnicas que se utilizan para evaluar y probar la seguridad de una red se lo conoce como pruebas de penetración (Penetration Testing), uno de los recursos más poderosos con los que se cuenta hoy para generar barreras cada vez más eficaces.

En cuanto a las barreras de seguridad, un probador explica: "Están totalmente relacionadas con el tipo de información que se maneja en cada organización. Por consiguiente, según la información que deba ser protegida, se determinan la estructura y las herramientas de seguridad. No a la inversa". [RefKoasp]

La seguridad informática no es únicamente un conjunto de técnicas de hardware y software, a ella se agrega lo que se denomina "políticas de seguridad internas", que cada empresa u organización debe generar.

La explicación del por qué, viene de un dato de la realidad. Según un reciente informe de la publicación InformationWeek (http: //www.koasp.com), un porcentaje sustancial de intrusiones en las redes de las empresas (ya sean chicas, medianas o grandes) proviene de ataques internos. Es decir, los mismos empleados violentan a su propia organización. Y aquí es donde cobran especial importancia las políticas de seguridad que se establezcan, además del aspecto técnico.

El nivel de importancia que se le da a la cuestión de la seguridad se generalizó en los últimos años. Esto significa que las empresas son cada vez más conscientes del tema y no escatiman esfuerzos para evitar ser vulneradas.

Esta conclusión lleva a pensar que la seguridad creció. Pero esto no es asi, porque simultáneamente aumentó y se difundieron la técnicas y conocimientos para violentar sistemas informáticos. Por lo tanto, el nivel de inseguridad aumentó.

"En el año 1995, con la ejecución de algunas herramientas específicas de ataque y penetración, se hallaron 150 puntos vulnerables en diversos sistemas de red. En el año 2000, las mismas herramientas fueron utilizadas sobre las nuevas versiones de los sistemas operativos y el resultado fue peor: se encontraron 450 puntos débiles, pese a los avances y la mejora tecnológica del Software y Hardware". [Relkoasp]

Esto hace que las compañías de software presten cada vez más atención al problema. "El Windows 2000, por ejemplo, que salió al mercado, ya fue sometido a pruebas de este tipo y se le detectaron problemas de seguridad".

Con lo que se puede concluir que ningún sistema informático es 100% seguro, pero se pueden minimizar los riesgos invirtiendo en seguridad.

1.1.1. LA INVERSIÓN EN SEGURIDAD

Los costos de las diferentes herramientas de protección se están haciendo accesibles, en general, incluso para las organizaciones más pequeñas. Esto hace que la implementación de mecanismos de seguridad se dé prácticamente en todos los niveles. Empresas grandes, medianas, chicas y las multinacionales más grandes. Todas pueden acceder a las herramientas que necesitan y los costos (la inversión que cada empresa debe realizar) van de acuerdo con la empresa.

Pero no es sólo una cuestión de costos, los constantes cambios de la tecnología hacen que para mantener un nivel parejo de seguridad cada empresa deba actualizar permanentemente las herramientas con las que cuenta. Como los intrusos mejoran sus armas y metodologías de penetración de forma incesante, el recambio y la revisión constante en los mecanismos de seguridad se convierten en imprescindibles. Y éste es un verdadero punto crítico.

Según conocedores de la materia "el nivel de seguridad que se implante es tan importante como el tipo de elementos que se usen". Sin duda, éstos deben ser las que mejor se adapten al tipo de organización. Pero tan importante como eso es el hecho de conocer exactamente cómo funcionan y qué se puede hacer con ellos". Es prioritario saber los riesgos que una nueva tecnología trae aparejados.

Hoy es imposible hablar de un sistema ciento por ciento seguro, sencillamente porque el costo de la seguridad total es muy alto. Por eso las empresas, en general, asumen riesgos: deben optar entre perder un negocio o arriesgarse a ser violentado. La cuestión es que, en algunas organizaciones puntuales, tener un sistema de seguridad muy acotado les impediría hacer más negocios. Si un intruso quiere gastar cien mil dólares en equipos para descifrar una encriptación, lo puede hacer porque es imposible de controlarlo. Y en tratar de evitarlo se podrían gastar millones de dólares.

La solución a medias, entonces, sería acotar todo el espectro de seguridad, en lo que hace a plataformas, procedimientos y estrategias. De esta manera se puede controlar todo un conjunto de vulnerabilidades, aunque no se logre la seguridad total. Y esto significa ni más ni menos que un gran avance con respecto a unos años atrás.

1.2. PRINCIPIOS BÁSICOS DE LA SEGURIDAD INFORMÁTICA

Toda organización debe estar a la vanguardia de los procesos de cambio. Donde disponer de información continua, confiable y en tiempo, constituye una ventaja fundamental. Donde tener información es tener poder; donde la información se reconoce como:

- **Crítica**, indispensable para garantizar la continuidad operativa de la organización.
- **Valiosa**, es un activo corporativo que tiene valor en si mismo.
- **Sensitiva**, debe ser conocida por las personas que necesitan los datos.

Donde identificar los riesgos de la información es de vital importancia.

La seguridad informática debe garantizar:

- **La Disponibilidad** de los sistemas de información, acción por la cual el administrador debe disponer de sistemas de seguridad para que no sea violentado.
- **La Recuperación rápida** y completa de los sistemas de información, para que sean respaldados y protegidos nuevamente.
- **La Integridad** de la información, para que no sufra cambios de ninguna naturaleza cuando sea violentada.
- **La Confidencialidad** de la información es de importancia solo para la empresa más no para los piratas de información.

1.3. LA SEGURIDAD EN INTERNET

Al hablar sobre la "Seguridad en Internet" nos referimos al gran indice de inseguridad interna de la infraestructura informática de las empresas, así como la falta de una cultura informática necesaria para contemplar estos problemas.

El alto grado de vulnerabilidad de la información transferida por la Internet y la facilidad de ataques externos e internos que se traducen en pérdidas que ascienden hasta miles de dólares en términos de información alterada, robada o perdida.

Según una investigación realizada en 1700 empresas por la **ICSA** (Corporación Internacional de Estadísticas por Area), el 75 por ciento de estas han tenido algún problema de seguridad. De éstas el 40 por ciento ha enfrentado problemas de seguridad

debido a la falta de apoyo de la alta dirección para invertir en medidas y herramientas de seguridad y sólo el 38 por ciento se debió a la falta de herramientas adecuadas. [RefAlpworld]

Una alternativa es el uso de una llave pública y una privada mediante el protocolo de seguridad Securet Socket Layer (SSL) que autentifica tanto al usuario que envia como al que recibe la información, porque es durante este proceso de transmisión que ocurren la mayor parte de las violaciones en la seguridad. Más que un problema de tecnología, la seguridad en la transmisión de la información por la Red se debe a la falta de cultura de las organizaciones y de las personas que la integran.

El eslabón más débil de esta cadena en la seguridad la constituye el humano y no el tecnológico, lo cual destaca la importancia de tener una cultura de seguridad, porque no existe en muchas empresas un responsable de la seguridad.

A todos los usuarios se les debe divulgar las políticas de seguridad, además de hacer constantes auditorias para controlar que sean las adecuadas al momento que vive la empresa.

Lo que se necesita no es solamente prevenir un ataque en la seguridad, sino ser capaces de detectar y responder a esta agresión mientras ocurre y reaccionar ante la misma.

Es importante destacar que no existe un control de seguridad único, sino que las empresas deben contar con diversas capas de seguridad en todos los niveles de su información para

poder así detectar el problema en algunos de estos puntos antes de que llegue a la información crucial.

Otra alternativa para la protección de una red informática que se encuentra permanentemente conectada a Internet, es la aplicación de un muro de seguridad o Firewall en Inglés.

En este esquema se destacan las siguientes partes:

1. El usuario ingresa a un sitio prohibido pero el administrador del Firewall lo bloquea inmediatamente.
2. A través de la red (Internet normalmente) el Firewall determina si un usuario está debidamente autorizado por el administrador.
3. El Firewall detecta que los usuarios no ingresen a sitios o recursos prohibidos por el administrador del centro de cómputo.

En la figura 1.1 se puede apreciar toda la interacción que se realiza mediante el Firewall, mecanismo que permite describir fácilmente la forma de control de diferentes usuarios que no están debidamente autorizados para poder acceder a sitios no autorizados.

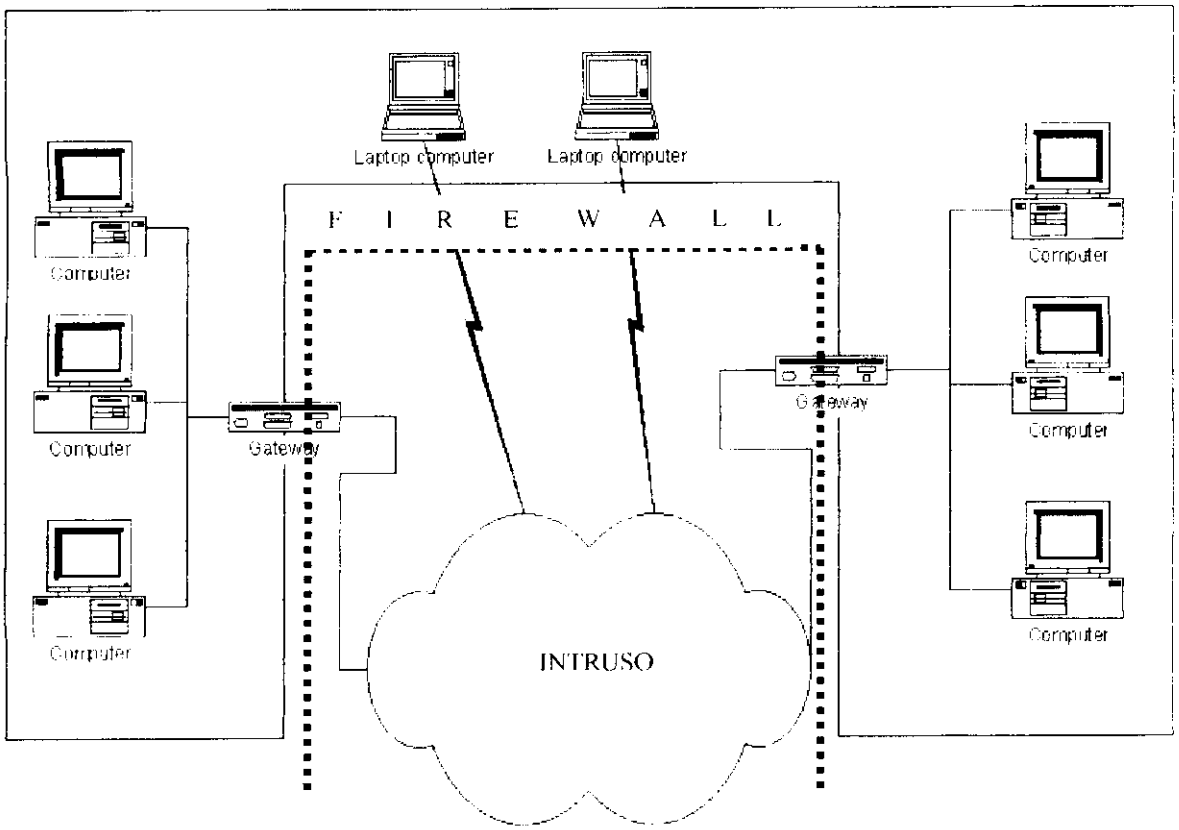


Figura 1.1 Esquema de un Firewall y Gateway en una red de computadoras

Al desarrollar este sistema de aplicación (Firewall), se protegerá la integridad y disponibilidad de la red informática, con el propósito de mantener sus recursos en forma óptima y verificar que los recursos estén debidamente asegurados.

Cuando la gente piensa en Internet, si bien la asocia al mayor recurso de información jamás imaginado, también asocia este nombre a inseguridad. Aunque tan mal merecida fama es en realidad debida al mismo mito que rodea a los intrusos, este concepto, básicamente utilizado por los gerentes de sistemas con respecto a la conexión de sus sistemas a Internet, si bien algo exagerado, no podríamos decir que es equivocado.

Conectarse implica riesgos, y esto es lo que menos quiere uno correr. En un estudio, tiempo atrás, de Ernst & Young, una consultora americana, se descubrió que cuatro de cada cinco grandes organizaciones (de más de 2500 empleados) estaban corriendo aplicaciones de misión crítica en LANs. Con el devenir de los tiempos, esa información procesada en las LANs, tanto como las LANs mismas, se han vuelto elementos vitales para el manejo de las empresas, y al mismo tiempo más vulnerables. De 61 grandes organizaciones analizadas por la **ICSA** (Corporación Internacional de Estadísticas por Area), en sólo tres meses de estudio se observaron 142 incidentes debido al hacking o a brechas de seguridad. [RefSupernet]

Por otra parte, aquellas organizaciones que permanecen desconectadas de Internet, hacen presión para establecer esa conexión, aunque más no fuese para e-mail. Incluso la presión para conectarse es tal, que algunos usuarios individuales contratan conexiones privadas sin conocimiento o autorización alguna por parte de la gente de sistemas de la empresa.

Para las empresas conectarse a Internet es como echar una brillante luz sobre los problemas de seguridad. Aquellos pequeños problemas referidos a la seguridad de la red, casi imperceptibles debido a su uso aislado, se vuelven trascendentes. Por ejemplo aquellas cuentas "guest" (invitado) apenas protegidas, o contraseñas demasiado obvias, pueden causar mucho daño cuando la red es visible desde el "exterior".

La moraleja es que si una empresa, cualquiera sea su tipo, intenta conectarse a otra u otras (mediante Internet o no), pueden pasar malas cosas si no se tiene la precaución necesaria,

para ello se necesita una fuerte combinación de políticas y tecnología dispuestas apropiadamente.

Estas cuestiones son bien comprendidas por aquellos profesionales que se dedican a administrar sistemas interconectados. Ahora, para un usuario ordinario de Internet, puede parecerle algo exagerada la preocupación expresada, dado a que puede llegar a considerar que una precaución de este tipo no es necesaria.

1.3.1. EL TRÁFICO DE INTERNET

Al referirnos al tráfico que pasa por el firewall, en realidad a lo que hacemos mención es a los datos transportados por el conjunto de protocolos TCP/IP. La figura 1.2 ilustra un diagrama de TCP/IP, que muestra cómo el protocolo se subdivide en capas, y la manera en que las direcciones son usadas. Para poder controlar el tráfico TCP/IP, debe conocerse sólidamente la estructura del mismo.

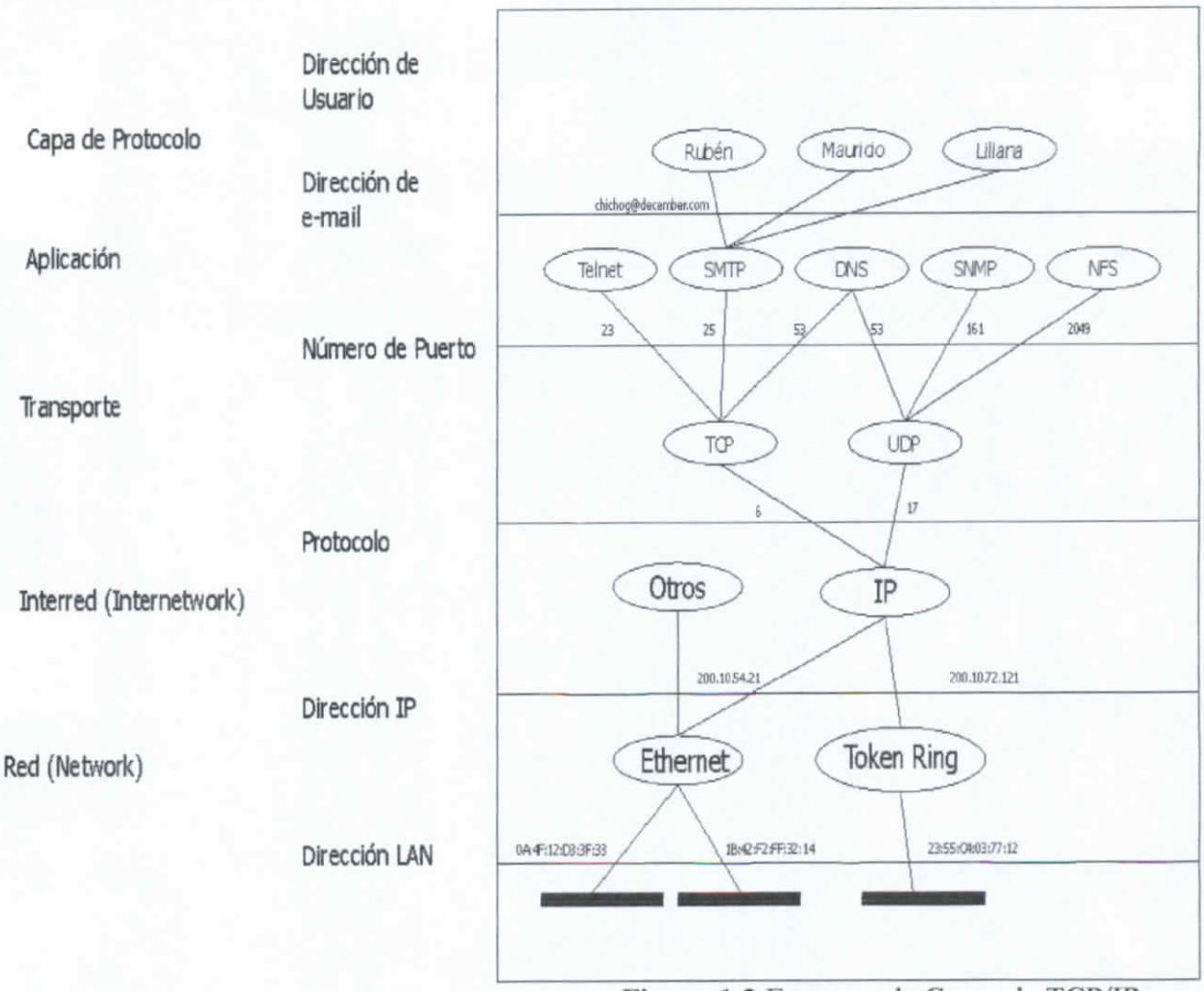


Figura 1.2 Esquema de Capas de TCP/IP

A continuación se explica lo básico acerca del protocolo TCP/IP.

Empecemos por definir qué es un protocolo. Un protocolo puede definirse como una descripción formal de mensajes para ser intercambiados, y las reglas a seguir con el objeto de que dos o más sistemas intercambien información de manera tal que permita que ambas partes lo comprendan.

La familia de protocolos TCP/IP, a la que se hace referencia oficialmente como el Conjunto de Protocolos de Internet (Internet Protocol Suite), en los documentos estándares de Internet, toma su nombre de dos de los protocolos más importantes: TCP (Transmission Control Protocol - Protocolo de Control de Transmisión) e IP (Internet Protocol - Protocolo de Internet). Las aplicaciones de red presentan los datos a TCP. Este divide los datos en trozos, llamados paquetes, y le otorga a cada uno un número.

Esos paquetes, reunidos, representan texto, imágenes, videos o sonido, cualquier cosa digital que la red pueda transmitir. La secuencia de números ayudan a asegurar que los paquetes sean correctamente re-ensamblados por el receptor. Por lo tanto, cada paquete consiste en el contenido, o datos, y la información que el protocolo necesita para su correcto funcionamiento, también llamado encabezado de protocolo.

Luego, TCP presenta los datos al Protocolo de Internet, o abreviadamente IP, con el propósito de proveer comunicación básica de host a host. IP agrega al paquete, en el encabezado de protocolo, la dirección de origen de donde vienen los datos y la dirección del sistema al que se dirigen.
El protocolo IP es técnicamente referido como un servicio de datagramas (paquetes) no confiable. Aunque en este contexto, no confiable significa simplemente que los protocolos superiores (como TCP) no dependen de IP para la entrega apropiada del paquete.

El protocolo IP hace su mejor intento con el objeto de entregar el paquete designado en su destino, pero si por algún motivo falla, solo tiende a "soltar" el paquete.

Aquí es donde el protocolo de alto nivel, TCP, entra en juego. TCP usa la secuencia de números, mencionada anteriormente, para re-ensamblar los paquetes en el orden correcto, y requerir la retransmisión de los paquetes que se pierden en el camino. Esto puede hacerse incluso si algunos de los paquetes toman diferentes rutas para llegar a su destino, lo que hace que la combinación de TCP e IP sea un protocolo muy confiable.

TCP usa otra pieza de información para asegurarse que los datos lleguen a la aplicación adecuada cuando llegan al sistema: el número de puerto. Estos puertos, con ubicaciones que van desde el 1 al 65535 (16 bits), no representan en realidad un puerto físico (como el serie o el paralelo), sino que se usan a modo de regiones de memoria. Los puertos del 1 al 1023 son reservados para aplicaciones de servidor, aunque los servidores pueden usar otros puertos altos.

En realidad los puertos altos (del 1024 en adelante) son dinámicamente asignados a aplicaciones según la necesidad. Algunas aplicaciones usan puertos estándar asignados, por ejemplo, los programas FTP siempre se conectarán al servidor FTP a través del puerto 21.

Por lo tanto los datos a ser transmitidos por TCP/IP tienen un puerto del cual vienen y al cual van, más una dirección IP de destino y origen. Los firewalls usan esas direcciones y los puertos para controlar el flujo de la información.

En el capítulo III se detallarán los protocolos que generalmente usan los Firewall.

1.3.2. LA SEGURIDAD Y LOS INTRUSOS

Junto a los avances de la informática y las comunicaciones en los últimos años, ha surgido una oleada de apasionados de estas tecnologías, que armados con sus ordenadores y conexiones a redes como Internet, ha logrado humillar a instituciones tan potencialmente seguras como el Pentágono y la NASA. La notoriedad de sus hazañas, su juventud y la capacidad de dejar en evidencia a instituciones muy poderosas, les hace aparecer ante la opinión pública rodeados de un lado de romanticismo. Pero, ¿quiénes son?, ¿son peligrosos para la sociedad?, ¿deben ser perseguidos?

Podemos encontrarnos con diferentes términos para definir a estos personajes: Hackers, *crackers*, piratas, etc., estando normalmente condicionado el calificativo a los objetivos y a los efectos de sus ataques a los sistemas.

El término Hacker, por ejemplo, se utiliza normalmente para identificar a los que únicamente acceden a un sistema protegido como si se tratara de un reto personal, sin intentar causar daños.

Los *crackers*, en cambio, tienen como principal objetivo producir daños que en muchos casos suponen un problema de extrema gravedad para el administrador del sistema.

En cuanto a los *piratas*, su actividad se centra en la obtención de información confidencial y *software* de manera ilícita.

1.4. SEGURIDAD LÓGICA Y CONFIDENCIAL

La computadora es un instrumento que estructura gran cantidad de información, la cual puede ser confidencial para individuos, empresas o instituciones, y puede ser mal utilizada o divulgada a personas que hagan mal uso de esta. También puede ocurrir robos, fraudes o sabotajes que provoquen la destrucción total o parcial de la actividad computacional.

Esta información puede ser de suma importancia, y el no tenerla en el momento preciso puede provocar retrasos sumamente costosos. Ante esta situación, en el transcurso del siglo XX, el mundo ha sido testigo de la transformación de algunos aspectos de seguridad y de derecho.

En la actualidad y principalmente en las computadoras personales, se ha dado otro factor que hay que considerar el llamado ""virus" de las computadoras, el cual aunque tiene diferentes intenciones se encuentra principalmente para paquetes que son copiados sin autorización ("piratas") y borra toda la información que se tiene en un disco.

Al auditar los sistemas se debe tener cuidado que no se tengan copias "piratas" o bien que, al conectarnos en red con otras computadoras, no exista la posibilidad de transmisión del virus.

El uso inadecuado de la computadora comienza desde la utilización de tiempo de máquina para usos ajenos de la organización, la copia de programas para fines de comercialización

sin reportar los derechos de autor hasta el acceso por vía telefónica a bases de datos a fin de modificar la información con propósitos fraudulentos.

Un método eficaz para proteger sistemas de computación es el software de control de accesos. Dicho simplemente, los paquetes de control de acceso protegen contra el acceso no autorizado, pues piden del usuario una contraseña antes de permitirle el acceso a información confidencial. Dichos paquetes han sido populares desde hace muchos años en el mundo de las computadoras grandes, y los principales proveedores ponen a disposición de clientes algunos de estos paquetes.

Un sistema integral de seguridad debe comprender:

- Elementos administrativos
- Definición de una política de seguridad
- Organización y división de responsabilidades
- Seguridad física y contra catástrofes(incendio, terremotos, etc.)
- Prácticas de seguridad del personal
- Elementos técnicos y procedimientos
- Sistemas de seguridad (de equipos y de sistemas, incluyendo todos los elementos, tanto redes como terminales.
- Aplicación de los sistemas de seguridad, incluyendo datos y archivos
- Planeación de programas de desastre y su prueba.

Se debe evaluar el nivel de riesgo que puede tener la información para poder hacer un estudio adecuado costo/beneficio entre el costo por perdida de información y el costo de un sistema de seguridad, para lo cual se debe considerar lo siguiente:

- Clasificar la instalación en términos de riesgo (alto, mediano, pequeño).
- Identificar aquellas aplicaciones que tengan un alto riesgo.
- Cuantificar el impacto en el caso de suspensión del servicio en aquellas aplicaciones con un alto riesgo.
- Formular las medidas de seguridad necesarias dependiendo del nivel de seguridad que se requiera.
- Justificar el costo de implantar las medidas de seguridad, para poder clasificar el riesgo e identificar las aplicaciones de alto riesgo, se debe preguntar lo siguiente:
    - Que sucedería si no se puede usar el sistema?
    - Si la contestación es que no, se podría seguir trabajando, esto nos sitúa en un sistema de alto riego.

Una vez que se ha definido, el grado de riesgo, hay que elaborar una lista de los sistemas con las medidas preventivas que se deben tomar, así como las correctivas en caso de desastre señalándole a cada uno su prioridad.

Hay que tener mucho cuidado con la información que sale de la oficina, su utilización y que sea borrada al momento de dejar la instalación que está dando respaldo.

Para clasificar la instalación en términos de riesgo se debe:

- Clasificar los datos, información y programas que contienen información confidencial que tenga un alto valor dentro del mercado de competencia de una organización, e información que sea de difícil recuperación.
- Identificar aquella información que tenga un gran costo financiero en caso de pérdida o bien puede provocar un gran impacto en la toma de decisiones.
- Determinar la información que tenga una gran pérdida en la organización y, consecuentemente, puedan provocar hasta la posibilidad de que no pueda sobrevivir sin esa información.

Para cuantificar el riesgo es necesario que se efectúen entrevistas con los altos niveles administrativos que sean directamente afectados por la suspensión en el procesamiento y que cuantifique el impacto que les puede causar este tipo de situaciones.

Al implantar sistemas de seguridad puede, reducirse la flexibilidad en el trabajo, pero no debe reducir la eficiencia.

1.5. SEGURIDAD FÍSICA

El objetivo es establecer políticas, procedimientos y prácticas para evitar las interrupciones prolongadas del servicio de procesamiento de datos, información debido a contingencias como incendio, inundaciones, huelgas, disturbios, sabotaje, etc. y continuar en medio de emergencia hasta que sea restaurado el servicio completo.

Entre las precauciones que se deben revisar están:

- Los ductos del aire acondicionado deben estar limpios, ya que son una de las principales causas del polvo y se habrá de contar con detectores de humo que indiquen la posible presencia de fuego.

- En las instalaciones de alto riesgo se debe tener equipo de fuente no interrumpible, tanto en la computadora como en la red y los equipos de teleproceso.

- En cuanto a los extintores, se debe revisar en número de estos, su capacidad, fácil acceso, peso y tipo de producto que utilizan. Es muy frecuente que se tengan los extintores, pero puede suceder que no se encuentren recargados o bien que sean de difícil acceso de un peso tal que sea difícil utilizarlos.

- Esto es común en lugares donde se encuentran trabajando hombres y mujeres y los extintores están a tal altura o con un peso tan grande que una mujer no puede utilizarlos.

- Otro de los problemas es la utilización de extintores inadecuados que pueden provocar mayor perjuicio a las máquinas (extintores líquidos) o que producen gases tóxicos.

- También se debe ver si el personal sabe usar los equipos contra incendio y si ha habido prácticas en cuanto a su uso.

- Se debe verificar que existan suficientes salidas de emergencia y que estén debidamente controladas para evitar robos por medio de estas salidas.

Los materiales más peligrosos son las cintas magnéticas que al quemarse, producen gases tóxicos y el papel carbón que es altamente inflamable.

1.6. POLITICAS DE SEGURIDAD

La evolución de las nuevas tecnologías, en especial el tremendo auge de las tecnologías Internet/Intranet, ha provocado la aparición de nuevas necesidades y la posibilidad de adquirir ventajas competitivas. Además hay que aclarar que lo que inicialmente partió como una opción de negocio se está transformando en una elección obligada si se desea mantener la posición en el mercado frente a los competidores; así, hasta los más reacios han debido claudicar ante la evidencia aplastante.

Los beneficios atribuibles a las nuevas tecnologías son muchos, pero también los riesgos: La pérdida de imagen (a menudo más crítica que la propia pérdida de datos), la pérdida de información, la suplantación de usuarios, el espionaje de información sensible o incluso el cumplimiento de la normativa vigente.

A pesar de lo cambiante del entorno, los requisitos de seguridad siguen siendo los mismos: Autenticación, confidencialidad, control de acceso, integridad y no repudio; aunque los objetivos y la implementación de los mismos evolucionan a velocidades vertiginosas.

Dado que el acceso a los recursos del Departamento de Computación produce necesariamente interacción entre personas, tanto a nivel computacional como personal (salas de estaciones), es necesario para la administración establecer ciertas reglas mínimas de convivencia, de manera que estos recursos sean bien aprovechados por todos.

El no cumplimiento de estas reglas puede derivar en sanciones que a la larga pueden perjudicar el desempeño académico de los usuarios.

1.6.1. POLITICAS DE USO ACEPTABLE DE LOS COMPUTADORES POR PARTE DE LOS USUARIOS

Se considera usuario del Sistema a los individuos que posean acceso a las computadoras a través de una cuenta de usuario contraseña de acceso.

A continuación se muestran algunos ejemplos de problemas y políticas de seguridad que podrían aplicarse en el acceso a las computadoras.

- Entrar a otras cuentas distintas de la propia, ni se puede compartir la cuenta. En caso que un trabajo académico lo requiera, los usuarios podrán pedir la creación de un grupo especial, de modo que pueda compartir archivos y/o directorios a través de él.

- Entrar a otros hosts en los que no se tenga acceso permitido.

- Intentar violar la seguridad del sistema local o de cualquiera accesible a través de la red.

- Interrumpir maliciosamente el normal funcionamiento del sistema.

Para los problemas anteriores se podrían tener las siguientes políticas:

- Seleccionar un buen password (clave) de acceso y mantenerla confidencial. Se considera un buen password, todo string (datos) que no pertenezca a un diccionario conocido; todo string que no pertenezca al vocabulario habitual; o todo string que no pueda ser fácilmente deducido.

- Configurar adecuadamente el acceso a la cuenta personal, de modo que no permita el acceso a ella de otros usuarios. El usuario debe preocuparse de la configuración de sus archivos sensibles: hosts, etc.

En las salas de estaciones:

- Se prohibe ingerir alimentos, bebidas y fumar.

- Se definen las siguientes prioridades (de menor a mayor):

- Usuarios no trabajando, es decir, las personas que están haciendo uso de alguno de los recursos de la red, que no impliquen directamente una responsabilidad académica.

- Usuarios trabajando (tareas, tesis, memoria, etc.)

Basados en esta prioridad, si una persona de mayor prioridad debe hacer uso de una estación, entonces puede solicitarla a una persona con menor prioridad. En ese caso, la estación debe ser cortésmente cedida.

Los usuarios pueden dirigirse a los administradores en caso de tener problemas con sus cuentas, o en el caso que deseen hacer un reclamo respecto otro usuario, o alguna situación que les parezca improcedente. En caso que el reclamo sea contra uno de los administradores, se pueden dirigir al Coordinador de Sistemas o al Director del Departamento.

1.6.2. POLITICAS DE USO ACEPTABLE DE LOS COMPUTADORES POR PARTE DE LOS ADMINISTRADORES

Los administradores estarán sujetos a las mismas reglas que los usuarios, pero además, debido a que sus funciones les otorgan el privilegio de ser superusuario en el sistema, se les exigirá un cuidado especial en lo que dice relación con la privacidad de las personas.

Los administradores, por regla general, no pueden:

- Leer el correo electrónico de los usuarios.

- Accesar programas o archivos protegidos de los usuarios.

- Interrumpir maliciosamente el normal funcionamiento del sistema.

En caso de que existan sospechas de que la seguridad del sistema o de que su normal funcionamiento están comprometidos, el administrador podrá infringir alguna de las reglas anteriores, dejando constancia del hecho, de modo que cualquier reclamo posterior, pueda justificarse adecuadamente.

- Los administradores no podrán compartir sus cuenta.

1.6.3. POLITICAS DE SEGURIDAD EN EL CORREO ELECTRÓNICO

Compilar las siguientes sugerencias para ser un usuario de correo electrónico seguro y con sentido común:

- Modificar la contraseña con frecuencia. El cambio de contraseña puede asegurar que el correo electrónico siga siendo privado. Además, las contraseñas que utilizan letras y números son más difíciles de adivinar.

- No compartir la contraseña. La mayoría de los administradores de correo electrónico no pedirán contraseña.

- No dejarse engañar por mensajes de correo electrónico maliciosos que pidan la contraseña. Éste es una treta bien conocida, aunque no demasiado común, ideada para engañarle a un usuario y hacer que comparta su contraseña. Como regla general, no compartir nunca con nadie.

- Nunca abrir archivos de datos adjuntos que provengan de un origen desconocido. Pueden contener lo que se conoce como "cartas bomba" o "virus", los cuales pueden dañar su computador.

- Recordar siempre cerrar la sesión cuando se haya terminado. Es rápido, fácil y puede impedir que intrusos no deseados tengan acceso a una cuenta de usuario.

- Se utiliza un equipo público, en un café de Internet por ejemplo, es aconsejable que se cierre el explorador que se estaba utilizando cuando se está preparado para finalizar su sesión de Internet.

- No responder a mensajes no solicitados ("spam"), u otro tipo de correo ofensivo o de acoso. Al responder, lo que único que hace es confirmar que tiene una dirección de correo electrónico activa a la que pueden enviar constantemente correo electrónico no solicitado. En su lugar, dirigir el mensaje no solicitado info@memo.com.co Para ayudar a controlar el envío masivo de correos indeseados,

proporcionamos a los usuarios "filtros" para el correo entrante. Éstos se pueden configurar fácilmente para enviar directamente a su papelera de reciclaje ciertos mensajes (como los que incluyan ciertas palabras) que no desee.

- Asegurarse de que utiliza el software de Internet más reciente (por ejemplo, un explorador como Microsoft Internet Explorer o Netscape Navigator). Las versiones más recientes a menudo ofrecen mejoras para proteger la seguridad.

- Utilizar siempre una red segura. La mayor parte de redes corporativas y proveedores de servicios Internet están protegidos por administradores que vigilan los posibles problemas de seguridad y actúan para proteger a los usuarios de los "intrusos" (usuarios malintencionados) que pueden intentar robar información personal que se transfiera a través de la red. Aunque el riesgo es pequeño, hay que tener precaución cuando se utiliza una red que no le sea familiar.

- Usar estaciones mantenidas por personal de confianza, o bien preguntar si el equipo de Internet que utiliza está protegido contra infracciones de seguridad. Utilizar el sentido común cuando se está conectado a Internet y mantener cierta dosis de escepticismo.

- Tener precaución cuando se revele información personal, tal como la dirección física, a cualquiera que conozca en el ciberespacio; incluso si afirma ser una autoridad.

1.6.4. SANCIONES

Es muy importante también definir las acciones o sanciones que se pueden tomar si no se cumple una o algunas de las políticas de seguridad antes mencionadas.

Por ejemplo un usuario tiene un password poco seguro, o una mala configuración que permite que otros usuarios se conecten en su cuenta, se le amonestara verbal o electrónicamente de modo que ajuste su configuración a lo que corresponde.

En caso de que surgiera algún problema con la cuenta de algún usuario, o con las acciones o procesos de algún usuario, los administradores podrán cerrar o bloquear temporalmente esa cuenta, investigar la situación, y luego traspasar los antecedentes a la Dirección del Departamento, quien determinará las sanciones si corresponde. En el caso que la falta lo amerite, los antecedentes pueden dirigirse a la Comisión de Etica de la Facultad.

Una vez que se ha realizado una introducción a la seguridad Informática, en el capítulo II se va a profundizar en el campo de los intrusos Informáticos.

CAPÍTULO II

INTRUSOS INFORMÁTICOS

Los piratas ya no tienen un parche en su ojo ni un garfio en reemplazo de la mano. Tampoco existen los barcos ni tesoros escondidos debajo del mar. Llegando al año 2002, los piratas se presentan con un cerebro desarrollado, curioso y con muy pocas armas: una simple computadora y una línea telefónica. Estos piratas constituyen los intrusos informáticos, que son personas que irrumpen en una red o sistema sin autorización. Los intrusos pueden ser de varios tipos, donde los menos inofensivos como los Hackers hasta los más dañinos y peligrosos como los Crakers.

2.1. TIPOS DE INTRUSOS

El objetivo de los Intrusos es infiltrarse en sistemas informáticos que están conectados mediante la red (Internet), la cual es una actividad profesional sin capacidad legal para ingresar a un sitio determinado, para proveerse de recursos informáticos. En términos generales los intrusos pueden estar clasificados como:

- Intrusos Internos (Insiders)
- Intrusos Externos (Outsiders)

Los Intrusos Internos son aquellas personas que intentan acceder a información no autorizada dentro de su misma red. Es decir que son usuarios legítimos de un sistema

informático y tratan de violentar seguridades de la misma red para acceder a información no autorizada.

Algunos ejemplos de ellos son empleados que han sido despedidos por alguna razón y tratan de vengarse haciendo daño al sistema.

Los Intrusos Externos son personas que tratan de ingresar a un sistema de cómputo ajeno. Por ejemplo de este tipo de intrusos son los denominados Hacker o Cracker.

Dentro de los intrusos Internos o Externos existe una clasificación más especifica, de acuerdo a las técnicas que utilizan. Esta clasificación se muestra a continuación.

2.1.1. HACKERS

La palabra Hackers aún no se encuentra en los diccionarios pero ya suena en todas las personas que alguna vez se interesaron por la informática o leyeron algún diario.

Hackers proviene de "hack", el sonido que hacían los técnicos de las empresas telefónicas al golpear los aparatos para que funcionen. Hoy es una palabra temida por empresarios, legisladores y autoridades que desean controlar a quienes se divierten descifrando claves para ingresar a lugares prohibidos y tener acceso a información indebida.

También se los define como usuarios de ordenadores especializados en penetrar en las bases de datos de sistemas informáticos estatales con el fin de obtener información secreta.

En la actualidad, el término se identifica con el de delincuente informático, e incluye a los cibernautas que realizan operaciones delictivas a través de las redes de ordenadores existentes.

Tradicionalmente se considera Hacker al aficionado a la informática cuya afición es buscar defectos y puertas traseras para entrar en los sistemas. Para los especialistas, la definición correcta sería: experto que puede conseguir un sistema informático.

Un Hackers es master en programación capaz de pensar y hacer cosas como si fuera "magia". Se dice que el término Hacker nació por los programadores de Massachusetts Institute of Technology (MIT), que en los 60 se llamaron a sí mismos Hackers, para hacer mención de que podían hacer programas mejores y más eficientes, o que hacían cosas que nadie había podido hacer.

Un Hacker es una persona que investiga la tecnología de una forma no convencional. Esta es la imagen que suele tener la gente de un hacker, por investigar la tecnología de forma distinta, termina metiéndose en lugares donde no estaba previsto que entrara y violando la seguridad de algunos sistemas. La definición más popular de hacker: "señor que viola sistemas de computadoras".

Sólo basta con repasar unas pocas estadísticas. Durante 1997, el 54 por ciento de las empresas sufrieron ataques de Hackers en sus sistemas. [RefKriesgam]
Las incursiones de los piratas informáticos, ocasionaron pérdidas totales de 137 millones de dólares en ese mismo año. El Pentágono, la CIA, UNICEF, La ONU y demás

organismos mundiales han sido victimas de intromisiones por parte de estas personas que tienen muchos conocimientos en la materia y también una gran capacidad para resolver los obstáculos que se les presentan.

Un hacker puede tardar meses en vulnerar un sistema ya que son cada vez más sofisticados. Pero ellos se llenan del lema, "Para toda ley creada existe una trampa".

Hacker no es entrar en un sistema, sino aprender como funciona. El principal objetivo de los Hackers no es convertirse en delincuentes sino "pelear contra un sistema injusto" utilizando como arma al propio sistema. Su guerra es silenciosa pero muy convincente.

La cultura popular define a los hackers como aquellos que, con ayuda de sus conocimientos informáticos consiguen acceder a los ordenadores de los bancos y de los negociados del gobierno. Bucean por información que no les pertenece, roban software caro y realizan transacciones de una cuenta bancaria a otra.

Los criminólogos, Donn Parker y August Bequai, por otra parte, describen a los hackers en términos menos halagadores. El primero los denomina "violadores electrónicos" y el segundo por su parte los describe como "vándalos electrónicos". Ambos, aunque aseveran que las actividades de los hackers son ilegales, eluden hábilmente llamarlos "criminales informáticos". [RefCrysof]

Donn Parker y August Bequai hacen una clara distinción entre el hacker que realiza sus actividades por diversión y el empleado que de repente decide hacer algo malo. Por tanto,

parece que tenemos una definición en la que cabe dos extremos: por un lado, el moderno ladrón de bancos y por otro el inquieto. Ambas actividades (y todas las intermedias) son calificadas con el mismo termino. Dificilmente se podría considerar esto como un ejemplo de conceptualización precisa.

Una gran parte de esta ambigüedad puede seguir desde el origen aproximadamente 20 años de vida del mencionado término. El término comenzó a usarse aplicándolo a un grupo de pioneros de la informática del MIT (Massachusetts Institute of Tecnology), a principios de la década de 1960. Desde entonces, y casi hasta finales de la década de 1970, un hacker era una persona obsesionada por conocer lo más posible sobre los sistemas informáticos.

Los diseñadores del ordenador Apple, Jobs y Wozniack, pueden considerarse hackers. Pero a principios de la década de 1980, influenciados por la difusión de la pelicula Juegos de Guerra, y el ampliamente publicado arresto de una "banda de hackers" conocida como la 414, los hackers pasaron a ser considerados como chicos, jóvenes, capaces de violar sistemas informáticos de grandes empresas y del gobierno.

El hacker tiene una actitud diferente hacia la tecnología, mira la tecnología de una forma diferente, no se conforma con leer el manual y usarla como se debe. El Hacker que desde chico empieza a desarmar el autito, es un hackercito. A ese hay que cuidarlo, no se conforma en jugar como se debe. Así empieza un hacker, desarmando autitos

La tecnología llama la atención; hace 15 o 20 años atrás no habia mucha información disponible sobre computadoras, queríamos empezar a comunicarnos; empezó a surgir el

tema de las comunicaciones de los módem, en esa época 300 baudios con acopladores acústicos, una cosa bien primitiva y no había muchas cosas disponibles para la gente, inclusive lo poco que había era solamente para empresas, entonces querían jugar con eso y la única alternativa que quedaba era usar esos canales que no estaban disponibles y tenías que vulnerar (hackear).

La actitud de un hacker es muy difícil mostrar en una película lo que hace un pirata de la información, sería muy aburrido mostrarlo porque es estar delante de una pantalla durante horas mirando un montón de números.

Para que un hacker sea mejor que otro hacker se define en función de las cosas que hacen y de cómo las hacen, lo que pasa es que no hay un organismo central de calificación de hackers que designe puntaje, pero hay una cosa tácita de que la gente conocen entre sí y obviamente existen ciertas rivalidades. Pero llama la atención un Hackers cuando hace algo de manera distinta, es la forma en que lo hace, porque le ocurrió una idea brillante, con uso fabuloso de una tecnología un Hackers realiza una acción determinada que a nadie se le hubiera ocurrido, ése es un hacker admirable.

En resumen podríamos decir que el concepto de hacker depende de cada uno. Existen tantos conceptos de hacker como internautas quieren conocer y aprender. Por ello para ser un hacker, hay que querer conocer cosas, aprender a programar (con un lenguaje de programación basta, pero puede no ser suficiente), hay que conocer sobre redes, sistemas, etc. para luego poder ponerlo en practica cuando se busque algún tipo de información. Es posible que no se tenga ningún conocimiento para acceder a la información, sin embargo,

hay que revisar si esa información es suficiente para uno y si es veraz, por lo que habrá que entrar en los sistemas para curiosear y conocer la verdad.

Un hacker no es más que alguien que quiere conocer la verdad y aprende para ello. También un hacker pone a prueba sus conocimientos, intentando entrar en sitios a los que no ha sido invitado. También intenta mejorar a los demás poniendo en conocimiento de todos, las fallas y las virtudes que ha encontrado en los sistemas y redes. [REFGTRI]

2.1.2. CRACKERS

Los crackers (crack=destruir) son aquellas personas que siempre buscan molestar a otros, piratear software protegido por leyes, destruir sistemas muy complejos mediante la transmisión de poderosos virus, etc. Esos son los crackers. Adolescentes inquietos que aprenden rápidamente este complejo oficio.

Se diferencian con los Hackers porque no poseen ningún tipo de ideología cuando realizan sus "trabajos". Los medios de comunicación masivos prefieren tildarlos a los Crackers de delincuentes que interceptan códigos de tarjetas de crédito y los utilizan para beneficio propio.

Los Crackers tienen como blanco sistemas en donde pueden sacar beneficio o simplemente donde pueden hacer daño, por ejemplo, se entrometen en los sistemas de aeropuertos produciendo un caos en los vuelos y en los horarios de los aviones.

2.1.3. PHREAKERS

El objetivo de un Phreaker no es realizar llamadas de larga distancia gratis, sino descubrir lo que la compañía telefónica no explica sobre su red.

Así, aunque un individuo tenga conocimiento especial sobre los sistemas telefónicos, cuando realiza una llamada de larga distancia gratis para cargar un juego, esta actuando como un Telepirata, en cierto modo, esto es un puro argumento semántico.

Los phreakers telefónicos, hoy en día es una actividad de uso habitual. Cuando se publicaron las aventuras de John Draper, en un articulo de la revista Esquire, en 1971. Se trata de una forma de evitar los mecanismos de facturación de las compañías telefónicas. Permite llamar a cualquier parte del mundo sin costo alguno prácticamente.
En muchos casos, también evita, o al menos inhibe, la posibilidad de que se pueda trazar el camino de la llamada hasta su origen, evitando así la posibilidad de ser atrapado. Para la mayor parte de los miembros del submundo informático, ésta es simplemente una herramienta para poder realizar llamadas de larga distancia sin tener que pagar enormes facturas.

La cantidad de personas que se consideran phreakers, contrariamente a lo que sucede con los hackers, es relativamente pequeña. Pero aquellos que si se consideran phreakers lo hacen para explorar el sistema telefónico.

La mayoria de la gente, aunque usa el teléfono, sabe muy poco acerca de él. Los phreakers, por otra parte, desean aprender mucho sobre el tema. Este deseo de conocimiento lo resume así un phreaker activo: "El sistema telefónico es la cosa más interesante y fascinante que se conoce". "Hay tantas cosas que aprender". Incluso los phreakers tienen diferentes áreas de conocimiento. Hay tantas cosas que se pueden conocer que en una tentativa puede aprenderse algo muy importante, o puede suceder lo contrario. Todo depende de como y donde obtener la información.

Trabajar para una empresa de telecomunicaciones, haciendo algo interesante, como programar una central de conmutación. Algo que no sea una tarea esclavizadora e insignificante, pero que sea divertido.

El tener acceso a cosas de estas empresas, como manuales, etc., debe ser "grandioso". La mayoría de la gente del submundo no se acerca al sistema telefónico con esa pasión. Solo están interesados en explorar sus debilidades para otros fines. En este caso, el sistema telefónico es una herramienta muy utilizada.

Con el uso de tarjetas telefónicas, abrieron la puerta para realizar este tipo de actividad a gran escala. Hoy en día no hace falta ningún equipo especial. Solo un teléfono con marcación por tonos y un número de tarjeta, y con eso se puede llamar a cualquier parte del mundo.

Lo que más le llama la atención es lo de la telefonía celular, todo lo que se puede hacer sobre telefonía celular. cuán fácilmente es, más conocemos que muchos políticos tampoco

lo saben, si no dirían las cosas que han dicho por celular, hay gente que obviamente le interesa mucho el tema Internet, evidentemente.

El uso de las herramientas que son propias no esta limitada a los phreakers, pero no es suficiente para merecer la distinción.

En particular, el mundo de los Hackers y los Phreakers están muy relacionados. Pero, de la misma forma que no debemos agrupar toda la actividad del submundo informático bajo la acepción de Hacker, tampoco debemos insistir en que nuestras definiciones sean exclusivas hasta el punto de ignorar lo que representan.

Las tipologías son amplias. Pero representan un paso más en la representación precisa, especificación e identificación de las actividades que se dan en el submundo de la informática.

La incorporación de las denominadas "redes inteligentes" podría dificultar considerablemente las actividades de los pheackers y Hackers.

El Instituto Tecnológico de Georgia, EEUU, trabaja en un proyecto de desarrollo de redes neurológicas, que probablemente aumentarán la seguridad del tráfico digital.

El nombre "red neurológica" se basa en las neuronas del cerebro humano, que aprenden de la experiencia, creando conexiones entre las distintas áreas del cerebro. Con todo, cabe

precisar que no se trata de redes que estén en condiciones de pensar, sino de sistemas capaces de identificar patrones en el flujo digital y aprender de los intentos de intrusión.

Hoy en día, los administradores de sistemas deben actualizar manualmente los sistemas de protección de las redes contra las embestidas de los sagaces piratas informáticos.

Con la incorporación de redes inteligentes se hará más previsible y fácil la contención intrusos, según escribe James Cannady, experto en el tema, en un artículo en Netsys.com.

Según Cannady, tales redes estarán incluso en condiciones de detectar máquinas que monitorizan ilegalmente el tráfico de la red para captar y apoderarse de información tal como números de tarjetas de crédito, contraseñas y otros datos confidenciales.

La novedad es que las redes neurológicas detectarán ese tipo de máquinas sin que sus operadores se percaten.

El avance de la informática ha introducido nuevos términos en el vocabulario de cada día. Una de estas palabras, hacker, phreacker, tiene que ver con los delitos informáticos.

Todos estamos familiarizados con las historias de aquellos que consiguen entrar en las corporaciones informatizadas. Pero tenemos la impresión de que el término "hacker y phreacker" es uno de los peor entendidos, aplicados y, por tanto, usados en la informática.

2.1.4. TELEPIRATERÍA

El objetivo de un Telepirata es obtener una copia del software más moderno para su ordenador.

La "telepiratería" del software. Consiste en la distribución ilegal de software protegido por los derechos de autor. No nos referimos a la copia e intercambio de diskettes que se produce entre conocidos (que es igualmente ilegal), sino a la actividad que se realiza alrededor de los sistemas BBS (Bulletin Board System) que se especializan en este tipo de trafico.

El acceso a este tipo de servicios se consigue contribuyendo, a través de un módem telefónico, con una copia de un programa comercial. Este acto delictivo permite a los usuarios copiar, o "cargar", de tres a seis programas que otros hayan aportado. Así, por el precio de una sola llamada telefónica, uno puede amontonar una gran cantidad de paquetes de software. En muchas ocasiones, incluso se evita pagar la llamada telefónica.

Nótese que al contrario que las dos actividades de hacker y phreaker, no hay ninguna consideración al margen de "prestigio" o "motivación" en la telepiratería. En este caso, el cometer los actos basta para "merecer" el título de telepirata

La telepiratería esta hecha para las masas. Al contrario de lo que sucede con los hackers y los phreakers, no requiere ninguna habilidad especial. Cualquiera que tenga un ordenador con módem y algún software dispone de los elementos necesarios para entrar en el mundo de la telepiratería. Debido a que la telepiratería no requiere conocimientos especiales, el

papel de los telepiratas no inspira ningún tipo de admiración o prestigio en el submundo informático (Una posible excepción la constituyen aquellos que son capaces de quitar la protección del software comercial). Aunque los hackers y los phreakers de la informática probablemente no desaprueben la piratería, y sin duda participen individualmente de alguna forma, son menos activos (o menos visibles) en los BBS (Bulletin Board System) que se dedican a la telepiratería. Tienden a evitarlos porque la mayoría de los telepiratas carecen de conocimientos informáticos especiales, y por tanto son conocidos por abusar en exceso de la red telefónica para conseguir el último programa de juegos.

Los medios de comunicación afirman que son únicamente los hackers los responsables de las perdidas de las grandes compañías de telecomunicaciones y de los servicios de larga distancia. Este no es el caso. Los hackers representan solo una pequeña parte de estas perdidas. El resto esta causado por "los piratas y ladrones que venden estos códigos en la calle."

Podemos mencionar que el proceso de intercambio de grandes programas comerciales por módem normalmente lleva varias horas, y son estas llamadas, y no las que realizan los "entusiastas de telecomunicaciones", las que preocupan a las compañías telefónicas.

Pero sin considerar la ausencia de conocimientos especiales, por la fama de abusar de la red, o por alguna otra razón, parece haber algún tipo de división entre los hackers phreakers y los telepiratas.

Después de haber descrito los tres papeles del submundo informático, podemos ver que la definición presentada al principio, según la cual un hacker era alguien que usaba una tarjeta de crédito telefónica robada para cargar alguno de los últimos juegos, no refleja las definiciones dadas en el propio submundo informático. Obviamente, corresponde a la descripción de un telepirata y no a las acciones propias de un hacker o un phreaker.

Los términos, "hacker", "phreaker" y "telepirata" se presentan y definen tal y como los entienden aquellos que se identifican con estos papeles.

Independientemente de que a un Hacker se le etiquete erróneamente como Telepirata, los accesos ilegales y las copias no autorizadas de software comercial van a seguir produciéndose. Pero si queremos conocer los nuevos desarrollos de la era informática, debemos identificar y reconocer los tres tipos de actividades con que nos podemos encontrar.

El agrupar los tres tipos bajo una sola etiqueta es más que impreciso, ignora las relaciones funcionales y diferencias entre ellos. Hay que admitir, de todas formas, que siempre habrá alguien que este en desacuerdo con las diferencias que se han descrito entre los grupos.

En el desarrollo de esta investigación, queda de manifiesto que los individuos que realizan actualmente estas actividades no se ponen de acuerdo en cuanto a donde están las fronteras. Las categorías y papeles, como se ha indicado previamente, no son mutuamente exclusivos.

No se quiere dar la impresión de que un individuo es un hacker, phreaker o telepirata exclusivamente. Estas categorías no son mutuamente excluyentes. De hecho, muchos individuos son capaces de actuar en más de uno de estos papeles. Se cree que la respuesta se encuentra en buscar los objetivos que se han expuesto previamente.

2.1.4.1. BBS (Bulletin Board System)

Servicio al que se conectan usuarios de ordenadores personales a través de un modem, mediante el que se pueden enviar mensajes, mantener conversaciones on-line, intercambiar software o acceder a bases de datos. Básicamente se compone de un ordenador con una potente base de datos y un sistema de conexión a través de la red telefónica conmutada y el correspondiente módem (la primera red pública existente en el mundo antes de Internet entre ordenadores independientes).

2.2. ATAQUES Y AMENAZAS

El objetivo es describir cuales son los métodos más comunes que se utilizan hoy para perpetrar ataques a la seguridad informática (confidencialidad, integridad y disponibilidad de la información) de una organización o empresa, y que armas podemos implementar para la defensa, ya que saber cómo nos pueden atacar (y desde donde), es tan importante como saber con que soluciones contamos para prevenir, detectar y reparar un siniestro de este tipo. Sin olvidar que éstas últimas siempre son una combinación de herramientas que tienen que ver con tecnología y recursos humanos (políticas, capacitación).

Los ataques pueden tener varios objetivos incluyendo fraude, extorsión, robo de información, venganza o simplemente el desafío de penetrar un sistema. Esto puede ser realizado por empleados internos que abusan de sus permisos de acceso, o por atacantes externos que acceden remotamente o interceptan el tráfico de red.

A esta altura del desarrollo de la "sociedad de la información" y de las tecnologías computacionales, los piratas informáticos ya no son novedad. Los hay prácticamente desde que surgieron las redes digitales, hace años atrás.

Sin duda a medida que el acceso a las redes de comunicación electrónica se fue generalizando, también se fue multiplicando el número de quienes ingresan "ilegalmente" a ellas, con distintos fines.

Los piratas cibernéticos que se consideran como una suerte de Robin Hood modernos y reclaman un acceso libre e irrestricto a los medios de comunicación electrónicos. Estos tipos de Genios informáticos, por lo general se lanzan desafiando programa de seguridad, captar las claves de acceso a computadoras remotas y utilizar sus cuentas para viajar por el ciberespacio, ingresar a redes de datos, sistemas de reservas aéreas, bancos, o cualquier otra "cueva" más o menos peligrosa.

Como los administradores de todos los sistemas, disponen de herramientas para controlar que "todo vaya bien", si los procesos son los normales o si hay movimientos sospechosos, por ejemplo que un usuario esté recurriendo a vías de acceso para las cuales no está

autorizado o que alguien intente ingresar repetidas veces con claves erróneas que esté probando.

Todos los movimientos del sistema son registrados en archivos, que los operadores revisan diariamente. Estos archivos se denominan bitácoras o "logs" que pueden servir para una auditoría del sistema o para detectar alguna acción maliciosa por parte de un intruso.

2.2.1. MÉTODOS Y HERRAMIENTAS DE ATAQUE

En los primeros años, los ataques involucraban poca sofisticación técnica.

Los insiders (empleados disconformes o personas externas con acceso a sistemas dentro de la empresa) utilizaban sus permisos para alterar archivos o registros.

Los outsiders (personas que atacan desde afuera de la ubicación física de la organización) ingresaban a la red simplemente averiguando una clave válida.

A través de los años se han desarrollado formas cada vez más sofisticadas de ataque para explotar "agujeros" en el diseño, configuración y operación de los sistemas. Esto permitió a los nuevos atacantes tomar control de sistemas completos, produciendo verdaderos desastres que en muchos casos llevó a la desaparición de aquellas organizaciones o

empresas con altísimo grado de dependencia tecnológica (bancos, servicios automatizados, etc).

Estos nuevos métodos de ataque han sido automatizados, por lo que en muchos casos sólo se necesita conocimiento técnico básico para realizarlos.

El aprendiz de intruso tiene acceso ahora a numerosos programas y scripts (encriptadores) de numerosos "hacker" bulletin boards y web sites, donde además encuentra todas las instrucciones para ejecutar ataques con las herramientas disponibles.

Los métodos de ataque descriptos a continuación están divididos en categorías generales que pueden estar relacionadas entre sí, ya que el uso de un método en una categoría permite el uso de otros métodos en otras. Por ejemplo: después de vulnerar (crackear) una contraseña, un intruso realiza un ingreso como usuario legítimo para navegar entre los archivos y explotar las debilidades del sistema.

Eventualmente también, el atacante puede adquirir derechos a lugares que permitan dejar un virus u otras bombas lógicas para paralizar todo un sistema antes de huir.

2.2.2 TIPOS DE ATAQUES

El objetivo de ataques por medio de Hackers, Phreaker, Crackers o Telepiratas u otros es infiltrarse en sistemas mediante la red (Internet), esta actividad profesional ilegal permite a este tipo de intrusos proveerse de recursos, causando perdidas, etc. A continuación mencionamos algunos tipos de ataques.

2.2.2.1. EAVESDROPPING Y PACKET SNIFFING

Muchas redes son vulnerables al eavesdropping, o la pasiva intercepción (sin modificación) del tráfico de red. En Internet esto es realizado por packet sniffers, que son programas que monitorean los paquetes de red que están direccionados a la computadora donde están instalados.

El sniffer puede ser colocado tanto en una estación de trabajo conectada a red, como a un equipo router o a un gateway de Internet, y esto puede ser realizado por un usuario con legítimo acceso, o por un intruso que ha ingresado por otras vías. Existen kits disponibles para facilitar su instalación.

Este método es muy utilizado para capturar loginIDs y passwords de usuarios, que generalmente viajan claros (sin encriptar) al ingresar a sistemas de acceso remoto (RAS).

También son utilizados para capturar números de tarjetas de crédito y direcciones de e-mail entrantes y salientes.

El análisis de tráfico puede ser utilizado también para determinar relaciones entre organizaciones e individuos.

2.2.2.2. SNOOPING Y DOWNLOADING

Los ataques de esta categoría tienen el mismo objetivo que el sniffing, obtener la información sin modificarla. Sin embargo los métodos son diferentes. Además de interceptar el tráfico de red, el atacante ingresa a los documentos, mensajes de e-mail y otra información guardada, realizando en la mayoría de los casos un downloading de esa información a su propia computadora.

El Snooping puede ser realizado por simple curiosidad, pero también es realizado con fines de espionaje y robo de información o software. Los casos mas resonantes de este tipo de ataques fueron: el robo de un archivo con mas de 1700 números de tarjetas de crédito desde una compañía de música mundialmente famosa, y la difusión ilegal de reportes oficiales reservados de las Naciones Unidas, acerca de la violación de derechos humanos en algunos países europeos en estado de guerra [RefWintest].

2.2.2.3. TAMPERING O DATA DIDDLING

Esta categoría se refiere a la modificación desautorizada a los datos, o al software instalado en un sistema, incluyendo borrado de archivos.

Este tipo de ataques son particularmente serios cuando el que lo realiza ha obtenido derechos de administrador o supervisor, con la capacidad de disparar cualquier comando y por ende alterar o borrar cualquier información que puede incluso terminar en la baja total del sistema en forma deliberada. O aún si no hubo intenciones de ello, el administrador posiblemente necesite dar de baja por horas o días hasta chequear y tratar de recuperar aquella información que ha sido alterada o borrada.

Como siempre, esto puede ser realizado por insiders (internos) o outsiders, (externos) generalmente con el propósito de fraude o dejar fuera de servicio un competidor.

Son innumerables los casos de este tipo como empleados (o externos) bancarios que crean falsas cuentas para derivar fondos de otras cuentas, estudiantes que modifican calificaciones de exámenes, o contribuyentes que pagan para que se les anule la deuda por impuestos en el sistema municipal.

Múltiples web sites han sido víctimas del cambio de sus home page por imágenes terroristas o humorísticas, o el reemplazo de versiones de software para download por otros con el mismo nombre pero que incorporan código malicioso (virus, troyanos).

La utilización de programas troyanos esta dentro de esta categoría, y refiere a falsas versiones de un software con el objetivo de averiguar información, borrar archivos y hasta tomar control remoto de una computadora a través de Internet como el caso de Back Orifice y NetBus, de reciente aparición.

2.2.2.4. SPOOFING

Esta técnica es utilizada para actuar en nombre de otros usuarios, usualmente para realizar tareas de snoofing o tampering. Una forma común de spoofing, es conseguir el nombre y password de un usuario legítimo para, una vez ingresado al sistema, tomar acciones en nombre de él, como puede ser el envío de falsos e-mails.

El intruso usualmente utiliza un sistema para obtener información e ingresar en otro, y luego utiliza este para entrar en otro, y en otro. Este proceso, llamado Looping, tiene la finalidad de evaporar la identificación y la ubicación del atacante. El camino tomado desde el origen hasta el destino puede tener muchas estaciones, que exceden obviamente los límites de un país.

Otra consecuencia del looping es que una compañía o gobierno pueden suponer que están siendo atacados por un competidor o una agencia de gobierno extranjera, cuando en realidad están seguramente siendo atacado por un insider, o por un estudiante a miles de km. de distancia, pero que ha tomado la identidad de otros.

El looping hace su investigación casi imposible, ya que el investigador debe contar con la colaboración de cada administrador de cada red utilizada en la ruta, que pueden ser de distintas jurisdicciones.

Los protocolos de red también son vulnerables al spoofing. Con el IP spoofing, el atacante genera paquetes de Internet con una dirección de red falsa en el campo From, pero que es aceptada por el destinatario del paquete.

El envío de falsos e-mails es otra forma de spoofing permitida por las redes. Aquí el atacante envía a nombre de otra persona e-mails con otros objetivos. Tal fue el caso de una universidad en USA que en 1998 debió reprogramar una fecha completa de exámenes ya que alguien en nombre de la secretaria había cancelado la fecha verdadera y enviado el mensaje a toda la nómina (163 estudiantes) [RefWintest].

Muchos ataques de este tipo comienzan con ingenieria social, y la falta de cultura por parte de los usuarios para facilitar a extraños sus identificaciones dentro del sistema. Esta primera información es usualmente conseguida a través de una simple llamada telefónica.

2.2.2.5. JAMMING o FLOODING

Este tipo de ataques desactivan o saturan los recursos del sistema. Por ejemplo, un atacante puede consumir toda la memoria o espacio en disco disponible, así como enviar tanto tráfico a la red que nadie más puede utilizarla.

Muchos ISPs (proveedores de Internet) han sufrido bajas temporales del servicio por ataques que explotan el protocolo TCP. Aquí el atacante satura el sistema con mensajes que requieren establecer conexión. Sin embargo, en vez de proveer la dirección IP del emisor, el mensaje contiene falsas direcciones IP (o sea que este ataque involucra también spoofing).

El sistema responde al mensaje, pero como no recibe respuesta, acumula buffers con información de las conexiones abiertas, no dejando lugar a las conexiones legítimas.

Muchos host de Internet han sido dados de baja por el "ping de la muerte", una versión trampa del comando ping. Mientras que el ping normal simplemente verifica si un sistema esta enlazado a la red, el ping de la muerte causa el reboot o el apagado instantáneo del equipo.

Otra acción común es la de enviar millares de e-mails sin sentido a todos los usuarios posibles en forma continua, saturando los distintos servers destino.

2.2.2.6. CABALLOS DE TROYA

Consiste en introducir dentro de un programa una rutina o conjunto de instrucciones, por supuesto no autorizadas y que la persona que lo ejecuta no conoce, para que dicho programa actúe de una forma diferente a como estaba previsto (Por ejemplo Formatear el disco duro, modificar un fichero, sacar un mensaje, etc.).

2.2.2.7. BOMBAS LOGICAS

Este suele ser el procedimiento de sabotaje mas comúnmente utilizado por empleados descontentos. Consiste en introducir un programa o rutina que en una fecha determinada destruirá, modificará la información o provocará que se cuelgue el sistema. Muchos de estos programas son transferidos por medio de correo electrónico con mensajes que parecen inofensivos.

2.2.2.8. INGENIERA SOCIAL

Básicamente convencer a la gente de que haga lo que en realidad no debería. Por ejemplo llamar a un usuario haciéndose pasar por administrador del sistema y requerirle el password con alguna excusa convincente. Esto es común cuando en el Centro de Computo los administradores son amigos o conocidos.

Por ejemplo un intruso podría hacerse pasar por vía telefónica por un administrador de red y solicitar temporalmente su contraseña para fines de prueba, aplicando la psicología para evitar que el usuario se de cuenta que esté depositando información sensible que puede comprometer el sistema.

2.2.2.9. DIFUSION DE VIRUS

Si bien es un ataque de tipo tampering, difiere de este porque puede ser ingresado al sistema por un dispositivo externo (diskettes) o través de la red (e-mails u otros protocolos) sin intervención directa del atacante.

Dado que el virus tiene como característica propia su autoreproducción, no necesita de mucha ayuda para propagarse a través de una LAN o WAN rápidamente, si es que no esta instalada una protección antivirus en los servidores, estaciones de trabajo, y los servidores de e-mail.

Existen distintos tipos de virus, como aquellos que infectan archivos ejecutables (.exe, .com, .bat, etc) y los sectores de boot-partición de discos y diskettes, pero aquellos que causan en estos tiempos mas problemas son los macro-virus, que están ocultos en simples documentos o planilla de cálculo, aplicaciones que utiliza cualquier usuario de PC, y cuya difusión se potencia con la posibilidad de su transmisión de un continente a otro a través de cualquier red o Internet. Y además son multiplataforma, es decir, no están atados a un

sistema operativo en particular, ya que un documento de MS-Word puede ser procesado tanto en un equipo Windows 3.x/95/98, como en una Macintosh u otras.

Cientos de virus son descubiertos mes a mes, y técnicas más complejas se desarrollan a una velocidad muy importante a medida que el avance tecnológico permite la creación de nuevas puertas de entrada. Por eso es indispensable contar con una herramienta antivirus actualizada y que pueda responder rápidamente ante cada nueva amenaza.

El ataque de virus es el más común para la mayoría de las empresas, que en un gran porcentaje responden afirmativamente cuando se les pregunta si han sido víctimas de algún virus en los últimos 5 años.

2.2.2.10. OBTENCIÓN DE PASSWORDS, CÓDIGOS Y CLAVES

Este método (usualmente denominado cracking), comprende la obtención "por fuerza bruta" de aquellas claves que permiten ingresar a servidores, aplicaciones, cuentas, etc. Muchas passwords de acceso son obtenidos fácilmente porque involucran el nombre u otro dato familiar del usuario, que además nunca la cambia. En esta caso el ataque se simplifica e involucra algún tiempo de prueba y error.

Otras veces se realizan ataques sistemáticos (incluso con varias computadoras a la vez) con la ayuda de programas especiales y "diccionarios" que prueban millones de posibles claves hasta encontrar la password correcta.

Es muy frecuente crackear una contraseña explotando agujeros en los algoritmos de encriptación utilizados, o en la administración de las claves por parte la empresa.

Por ser el uso de passwords la herramienta de seguridad más cercana a los usuarios, es aquí donde hay que poner énfasis en la parte "humana" con políticas claras (como se define un password?, a quien se esta autorizado revelar?) y una administración eficiente (cada cuanto se están cambiando?).

No muchas organizaciones están exentas de mostrar passwords escritas y pegadas en la base del monitor de sus usuarios, u obtenerlas simplemente preguntando al responsable de cualquier PC, cual es su password?.

2.2.2.11. ELIMINAR EL BLANCO

Este tipo de ataques está subdividido a continuación por otros tipos de ataques que pertenecen al mencionado anteriormente (eliminar el blanco).

2.2.2.11.1. PING MORTAL.

Algunos ataques eliminan el blanco en lugar de inundarlo con trabajo. Un ejemplo de este tipo es el ping mortal, un paquete ping ilicitamente enorme, que hace que el equipo de destino se cuelgue. Muchas implementaciones de routers, la mayoría de los Unix y todas las versiones de Windows se mostraron vulnerables a este ataque cuando se lo descubrió por primera vez hace un par de años. A pesar de que los vendedores lanzaron parches de inmediato, hay todavía cantidades significativas de hosts "no corregidos" en las redes de producción (en especial, las que corren bajo el Windows 95).

TCP/IP permite un tamaño máximo de paquete de 64 kilobytes (KB, este máximo está dividido en piezas mucho más pequeñas a través de protocolos de capas más bajas, como Ethernet o token ring, pero dentro de una computadora, paquetes mucho más grandes son posibles).

Para lidiar con un paquete de 64 KB, la cola TCP/IP asigna un buffer en memoria de 64 KB. Al recibir una cantidad ilicitamente grande de información, como un ping mortal, el buffer del equipo de destino se desborda y el sistema se puede colgar.

2.2.2.11.2. LAND

Otro método para colgar un equipo es el denominado Land attack, en el que se genera un paquete con direcciones IP y puertos de fuente y destino idénticos. Existen diferentes variantes para este ataque. Una de ellas usa idénticas direcciones IP de fuente y destino, pero no números de puertos.

Fue esta variación la que utilizó NSTL contra el primer par de productos testeados (evaluados) y los dos identificaron el tráfico como un land attack. El tercer producto que se probó, el Netranger, de Cisco, identificó a un land attack solamente (y correctamente) cuando ambas direcciones y números de puerto eran idénticos.

El ingeniero de Cisco agregó enseguida una nueva regla, que detectaba a los paquetes con direcciones idénticas nada más. Una vez más, esto pone de manifiesto la importancia de saber qué es lo que se debe buscar.

2.2.2.11.3. SUPERNUKE

Un ataque característico de los equipos con Windows es el Supernuke (llamado también a veces Winnuke), que hace que los equipos que escuchan por el puerto UDP 139 se cuelguen. Netbios es un protocolo integral para todas las versiones en red de Windows.

Para transportar Netbios por IP, Microsoft ideó el Windows Networking (Wins), un esquema que enlaza el tráfico Netbios a puertos TCP y UDP 137, 138 y 139. Al enviar a estos puertos fragmentos UDP, se pueden arruinar equipos Windows que no estén arreglados o disminuir la velocidad del equipo durante un largo tiempo. En cuanto a la inundación ICMP, todos los IDS (Intrusion Detection Systems) reconocieron a los ataques Supernuke.

2.2.2.11.4. TEARDROP 2

El ataque más reciente a nuestra base de datos, el Teardrop 2, data a fines de 1997. Al igual que el Supernuke, los ataques Teardrop 1 y Teardrop 2 afectan a fragmentos de paquetes. Algunas implementaciones de colas IP no vuelven a armar correctamente los fragmentos que se superponen, haciendo que el sistema se cuelgue. Windows NT 4.0 de Microsoft es especialmente vulnerable a este ataque, aun cuando se ha aplicado el Service Pack 3. La empresa hizo un parche del Teardrop 1 en mayo de 1997, pero se mostró vulnerable al Teardrop 2, que supuso colocar una bandera de "urgente" en la cabecera de un fragmento TCP. Hasta el lanzamiento de un hot fix en enero de 1998.

En cuanto al ataque Dig, el actual lanzamiento del Realsecure, de ISS no vio el ataque del Teardrop 2. Sí lo vio el lanzamiento beta de la versión 2.5.

2.3. PROGRAMAS UTILIZADOS PARA ATAQUES

Al igual que las herramientas de protección de sistemas informáticos, se han desarrollado una amplia gama de programas utilizados para ataques.

Uno de los programas que más se conoce como cazador de contraseñas es el **Crack** únicamente funciona bajo UNIX, pero es muy utilizado por los administradores para poder probar sus propias contraseñas.

Se tiene en cuenta que se pueden hacer dos usos distintos de los cazadores de contraseñas, uno de modo bueno, por así decirlo, y otro de modo malo. Podemos definir al modo bueno, a aquel administrador de seguridad de una empresa, el cual debe configurar y gestionar todo el sistema informático de la empresa, garantizando su seguridad frente a usos no autorizados, por ello, si el sistema tiene contraseñas, se ha de probar dichas contraseñas para asegurarse de que otros no puedan averiguar, entonces hacemos uso del cazador de contraseñas, el cual nos dirá si es o no fiable nuestra contraseña. Pero tenemos el modo malo, el cual es obvio, usar un cazador de contraseñas para averiguar las contraseñas de otros de forma no autorizada. Es evidente que el modo malo no es legal ni ético y mucho menos de ser usado por un programador que se considere honrado y serio.

A continuación en la figura 2.1 se lista los programas más conocidos para realizar ataques a sistemas informáticos.

NOMBRE DEL PROGRAMA	DESCRIPCIÓN	S.O. (Sistema Operativo)
Cracker Jack 1.4	Descodificador de Passwords de Unix. Inglés.	Dos
Brute Forece 1.1	Descodificar de passwords Unix. Inglés.	Dos
John the Ripper	Posiblemente el mejor descodificador de password Unix.	Dos
Star Cracker 1.0	Otro descodificador de pass. Unix. Ing.	Dos
Hack486	Más descodificadores de pass. Éste incluye un fichero de password para probar. Muy rápido. Ing.	Dos
[Xit]v2.0	Más descodificadores..... Ing.	Dos
Crack v5.0	Otro descodificador pero de passwords ffb X. Ing.	Unix/Linux
Magic Cracker	Otro descodificador de passwords Unix. Ing.	Win95/NT
Jill20	Complemento para el Cracker Jack.Ing.	Dos
Unix Password analizer	Busca personas bastante importantes en un fichero password de Unix. Ing.	Dos
VMS crack 1.0	Descodificador password de sistemas VMS.	-
Crack CNX	Descodifica ficheros cnx del software de infovía para Win3.x. Ing.	Dos
Glide	Dicen que descodifica los passwords .PWL de W95. No es compatible con la versión OSR2. Ing.	Dos

Figura 2.1. Programas para realizar ataques a Sistemas Informáticos

PWL Viewer	Visualizador de los ficheros .PWL. Ing.	Dos/W95
PWL Tools	Como el anterior pero todo el kit, crackeador, y visualizador. La velocidad del cual está limitada por el mal uso que se pueda hacer. Ing.	Dos/W95
PopCrack v1.0	Cracker del Popmail Password. Ing.	Dos
Toneloc 1.10	Uno de los mejores War-Dialers de todos. Ing.	Dos
Phonetag v1.3	Otro escaneador de telefonos. Ing.	Windows
THC scan v1.0	El mejor de todos. Sin ninguna duda. Pese a que es un poco dificil de configurar. Ing.	Dos
Keylog 95	Capturador de teclado. En el archivo figuran todas las teclas pulsadas. Ing.	Dos/Win95
Keylog v2.0 95/NT	Como el anterior pero mejorado. Ing.	Win95/NT
Passgrab 1.0	Otro capturador de teclado.	-
Password Thief v1.0	Un buen capturador de teclado. Sharewar.	W95
Passbios	Este programa engaña al usuario para pillar la clave de la Bios. Se simula la Bios del ordenador para engañar. Esp.	W95
L0phtCrack 2.01	Pillar passwords en NT. Ing.	W95/NT
PortScan	Escanea los puertos abiertos de un ordenador remoto. Ing.	Dos

Figura 2.1. Programas para realizar ataques a Sistemas

Winsock spy v0.91	Substituye el ficher wsock32.dll para espiar las comunicacioens de u Pc. W95. Ing.	-
Satan v1.1.1	Herramienta muy util para detectar posibles agujeros de seguridad. Ing.	Unix/Linux
Netcat v1.1.0	Herramienta que escribe y lee datos de conexiones TCP/IP. w95/NT. Ing.Versión Unix	W95/UNIX
Netpack v2.0	Conjunto de utilidades. Ing.	W95/NT
Hacker's Utility	Muchas utilidades y descodificadores de pass. Ing. o Ital.	Win95/NT
Date Dictionary Creator	Generador de listas, o diccionarios para los crackeadores de passwords. Ing.	Dos
Wordlist Maker	Creador de listas de palabras de Ing.	Win3.x.
Diccionario	Un diccionario grande para utilizarlo para los crackeadores de passwords. .	Dos
Super Diccionario	Uno de los diccionarios más grandes. !!!!3'8Mb.¡¡¡	-
Manipulador de Passwords	Descodifica y modifica el fichero /etc/passwd. Esp.	Dos
NETLAB95	Conjunto de utilidades para chequear Redes, funciones finger, ping, etc...	W95

Figura 2.1. Programas para realizar ataques a Sistemas Informáticos

CAPÍTULO III

MURALLAS DE FUEGO (FIREWALL)

Muchos de los problemas de seguridad que aparecieron con la interconexión de redes en el surgimiento de Internet pueden ser remediados o atenuados mediante el uso de determinadas técnicas y controles. Con un firewall podemos implementar un nivel de seguridad apropiado permitiendo al mismo tiempo el acceso a los vitales servicios de Internet. Un firewall es un sistema o un grupo de sistemas que implementan una política de control de acceso entre dos o más redes. Podemos imaginarlo como compuesto por dos grandes módulos; uno destinado a bloquear los accesos y el otro a permitirlos. El firewall constituye la herramienta pero se desprende que debemos tener muy claro que tipo de control de acceso debemos implementar, y a su vez esto constituye un subconjunto de la política de seguridad de la compañía.

El firewall proporciona un único check-point que preserva a la Intranet del ataque de intrusos que pudieran accederla. Nos permite monitorear la seguridad a través de sus alarmas y logs, los cuales deben ser revisados periódicamente pues debemos poder determinar hipotéticos intentos de acceso ya que el mismo firewall puede ser violado y una vez que esto sucede estamos sin protección. Como agregados a su función primordial los firewalls proveen dos funciones extras; el servicio de NAT (Network Address Translator) que permite que un servidor de mi Intranet se presente en Internet con un número de IP válido sin necesidad de reconfigurarlo, y por otro lado ofrece un punto de "logeo" y

monitoreo del uso de Internet en cuanto a requerimientos para poder determinar anchos de bandas, problemas de saturación del vínculo, etc.

Como se dijo un firewall es parte de una política y no podemos dejarle librada toda la responsabilidad cuando tenemos accesos de tipo PPP por ejemplo, o un empleado divulga la información; este tipo de casos están denotando la existencia de back-doors (puertas traseras) evidentemente muy apetecibles como objetivo de cualquier ataque.

Otro punto importante es que a pesar de que la información pase por un firewall este no nos pueda garantizar 100% la ausencia de virus, pues su inmensa variedad hace imposible que se pueda analizar cada uno de los paquetes que pasan a través de él. Por último, si bien en la actualidad el tema está lo suficientemente pulido, existen aplicaciones que pueden estar mal diseñadas y que permiten que se puedan transmitir paquetes no deseados encapsulados dentro de los mensajes que trafican; este era el caso de viejas versiones del sendmail (servicio de correo electrónico en plataforma UNIS/LINUX), por ejemplo. Por todo esto la utilización de un firewall no constituye el remedio para resolver todos los problemas de seguridad de Internet.

Podemos citar algunos ejemplos de usos más comunes de firewalls. La protección ante la utilización de servicios vulnerables; expusimos la vulnerabilidad de algunas aplicaciones, entre las cuales podemos encontrar en la actualidad el NFS (Network File System) y NIS, (Network Information Service) por dar un ejemplo. Este tipo de servicios son vulnerables a los ataques; no podemos optar por deshabilitarlos pues son muy útiles en la Intranet con lo cual la utilización de un firewall estaríamos filtrando todos los accesos externos a este tipo

de servicios. Por otro lado se puede administrar un control de acceso; esto se traduce en implementar políticas que permitan el acceso a algunos servidores y a otros no. Además la utilización de un firewall nos permite perfeccionar el control de la seguridad en cuanto a su centralización, pues además de todo el subsistema de auditoría podríamos concentrar todos los add-on de software de seguridad en un punto central en lugar de implementarlo en cada host (sumando a esto el mantenimiento que ello implica). Otras soluciones de este tipo como por ejemplo Kerberos (es un personaje de la mitología griega que por ser quien cuidaba las puertas del infierno, representa seguridad. Se podría decir que como servicio de autenticación, ahora cuida las puertas de la red, impidiendo que entren personas indeseadas) obligan a hacer actualizaciones host por host y si bien en algunos casos son las soluciones más adecuadas los firewalls tienden a simplificar esta tarea. Como una actividad secundaria podemos citar el hecho de que si todo el tráfico hacia Internet pasa por un firewall esto permite a partir de su accounting determinar el grado de uso del vínculo de red y proyectar crecimiento; este último punto debe estar soportado por otras herramientas.

A través de la realización de la presente investigación se buscan los siguientes objetivos:

- Que la investigación introduzca el concepto de firewalls. Esto incluye entender como funciona, para que sirve y el porqué de los Firewalls y su necesidad.

- Que la investigación tome conciencia de la necesidad en lo que a seguridad respecta del uso de barreras de protección a partir de que todas las redes corporativas de una organización hoy por hoy se encuentran conectadas al mundo

exterior a través de Internet, y por ende expuestas a múltiples intentos de accesos no autorizados.

- Que el investigador reconozca los diferentes tipos de firewalls en cuanto al nivel de protección.

- El tema central de la presente investigación, es que el investigador se informe de las diferentes configuraciones y arquitecturas que se pueden establecer mediante el uso de los firewalls, como así también mostrarle ventajas y desventajas de cada una de ellas. También se incluye en este punto una serie de consejos en cuanto a configuración, que si bien para un administrador experimentado puede resultar triviales, no dejan de ser relevantes como carácter informativo y de referencia para aquel que se inicia en este tema.

- Por último, dar una breve idea del estado de este tema como producto en el mercado.

3.1. DEFINICIÓN DE FIREWALLS

Originalmente el término Firewall se refiere a una técnica de construcción diseñada para prevenir la dispersión del fuego de una habitación a la otra. Ahora hablamos de Firewalls para Internetworking (conexiones entre redes).

El término Firewall es actualmente utilizado para definir un sistema o un grupo de ellos que ejerce una política de control entre dos redes. También puede definirse como un mecanismo para proteger redes de alta confianza con respecto de otras no confiables.

Los Firewalls se han convertido paulatinamente en un elemento muy importante en tantas organizaciones que conectan sus redes internas a otras externas como la Internet.

Una firewall es una barrera que controla el flujo del tráfico entre los host, los sistemas de redes, y los dominios. Existen diferentes clases, las más débiles; y las más seguras, que deberían bloquear el traspaso de cualquier tipo de datos.

Un Firewall es un tipo de tecnología que ayuda a prevenir el acceso de intrusos a una computadora, ya sea por medio de Internet o por medio de una Red Interna; además de controlar la entrada o salida de datos, no autorizada, a un sistema.

El objetivo de los cyberataques de los Hackers no es precisamente atacar a las organizaciones de seguridad; lo que ellos siempre intentan es buscar números de cuentas bancarias, claves, etc. Por otro lado, el surgimiento de las conexiones permanentes hacia

la Red, como por ejemplo, el cable módem entre otros, se está volviendo cada vez más popular, sobre todo entre los usuarios que necesitan estar la mayoría del tiempo on-line (en línea), para poder realizar su trabajo. De esta manera, el peligro de las intrusiones crece día a día, y los Hackers tienen más posibilidades de realizar más fácilmente su tarea. Por eso, para evitar todo tipo de inconvenientes, hoy se puede proteger información; bloquear los ataques de cualquier intruso, y proteger PC´s de las amenazas externas.

Ahora bien, hemos hablado mucho de los firewalls, pero en concreto ¿qué son? Existen muchas definiciones clásicas de un firewall (se pronuncia *fair-uol*) o "muro cortafuegos" en informática. Una muy interesante es la de Marcus Ranum, autor de dos "core papers" acerca de firewalls ("Thinking about firewalls" Pensando acerca de cortafuegos; y "An Internet firewall" Un muro cortafuegos de Internet), en la que lo describe de la siguiente manera: "un firewall de red es un sistema o grupo de sistemas que ejerce una política de control de acceso entre dos redes".

Una definición más específica, inclusive, la dan William Cheswick y Steve Bellovin, dos ingenieros de AT&T, autores del clásico "Firewalls and Internet security" (Adison Wesley, 1994), en la que definen al firewall como una colección de componentes o un sistema ubicado entre dos redes, que además posee las siguientes propiedades:

1.- Todo el tráfico desde dentro hacia fuera, y viceversa, debe pasar a través de él.

2.- Solo el tráfico autorizado, definido por una política local de seguridad, es permitido.

3.- El sistema en sí mismo es altamente resistente a la penetración.

Puesto de una manera simple, un firewall es un mecanismo utilizado para proteger una red confiable de una que no lo es. Un cortafuegos puede consistir en diferentes componentes, incluyendo filtros o pantallas que bloqueen la transmisión de cierto tipo de tráfico, y un gateway (se pronuncia *gueit-uey* y significa algo así como portal de salida), el cual esta compuesto por una máquina (o un conjunto de ellas) oficiando de enlace de servicios entre redes internas y externas a través de aplicaciones "proxy".[RefNlgob]

3.2. CARACTERISTICAS GENERALES DE LOS FIREWALLS

Para que un firewall trabaje apropiadamente y cumpla con su función, este debe ser parte de una consistente arquitectura de seguridad, la cual abarque a toda la organización. Las políticas deben ser realistas y reflejar el nivel de seguridad de toda la red. Un sitio con información ultra-secreta o clasificada NO necesita un firewall, lo que necesita es no estar conectado a Internet o los datos ultra-secretos o confidenciales deben ser mantenidos fuera de la red corporativa.

Un Firewall puede impedir que un usuario no autorizado acceda a un PC, independientemente de donde provenga él, es decir, puede provenir de la Web o de la red local. Bloquea algunos programas troyanos y otras aplicaciones que quieren dañar el sistema.

Mientras se mantiene conectado a la Web, constantemente se está enviando y recibiendo información en pequeñas unidades llamadas paquetes. Un paquete contiene la dirección de quien envía el mensaje, y del receptor, junto con una porción de información, una petición,

y un comando. Un firewall examina cada paquete enviado desde o hacia una máquina, para analizar si cumple con una serie de criterios; así, luego puede decidir si permite o no el paso del paquete de información. Todo tráfico externo de Internet hacia la red interna pasa a través del firewall, así puede determinar si dicho tráfico es aceptable, de acuerdo a sus políticas de seguridad.

Lógicamente un firewall es un separador, un analizador, un limitador. La implementación física varía de acuerdo al lugar. A menudo, un firewall es un conjunto de componentes de hardware: un router, un host, una combinación de routers, computadoras y redes con software apropiado. Rara vez es un simple objeto físico. Usualmente está compuesto por múltiples partes y alguna de esas partes puede realizar otras tareas, la conexión de Internet también forma parte del firewall.

Un firewall es vulnerable, él no protege de la gente que está dentro de la red interna, el firewall trabaja mejor si se complementa con una defensa interna. Un Firewall es la forma más efectiva de conectar una red a Internet para protegerla.

Los Firewalls también tienen otros usos: por ejemplo, se pueden usar para dividir partes de un sitio que tienen distintas necesidades de seguridad. [Ref3com]

3.3. LIMITACIONES DE LOS FIREWALLS

Si bien el enfoque sobre el cual queremos hacer énfasis esta dirigido a los Firewalls, nos es imposible no introducirnos dentro del tema de la seguridad informática. En casi toda publicación referida al tema, la seguridad informática se encuentran trabajando sobre conceptos equivocados y opiniones generales basadas principalmente en juicios de valor, hechos con información insuficiente. Algunas de estas equivocadas opiniones son, por ejemplo, que todos los problemas relacionados con la seguridad de las redes se resuelven simplemente desplegando un firewall. Mientras que los firewalls deben ser uno de los elementos a los que más debe prestárseles atención en el desarrollo de una conexión entre redes, éstos no son la solución total. Muchas de las amenazas caen fuera del ámbito de acción de estos sistemas. De hecho, la ingeniería social, una de las técnicas de hackeo más utilizada y efectiva, en la que se engaña a una persona para obtener información o contraseñas, compromete en gran medida la seguridad y no tiene nada que ver con los firewalls.

Estrictamente los "muros cortafuegos", solo sirven de defensa perimetral de las redes, no defienden de ataques o errores provenientes del interior.

Otra limitación de los firewalls es que no pueden resolver directamente el problema del código malicioso (léase virus y troyanos Capitulo II) Con el advenimiento de refinadas técnicas de infección y dispersión de código autorreplicante, y las más variadas cepas creadas por una multitud de programadores inescrupulosos, éste se ha vuelto en un importante riesgo a tener en cuenta. Algunos firewalls pueden chequear código entrante,

buscando virus o troyanos, pero sin embargo, estas defensas, si bien útiles, no son a toda prueba.

Los Firewalls realmente no pueden proteger contra los traidores, los usuarios que revelan información sensitiva sobre el teléfono son excelentes objetivos para la "ingeniería social". Cualquier usuario externo puede entrar a la red saltándose el firewall si encuentra un empleado "cooperador" adentro de la compañía que le de acceso a un pool (una lista) de módems.

Los Firewalls no pueden proteger contra "data-driven attacks", ataques en los cuales algo es enviado por correo o copiado a un Host interno donde luego es ejecutado.

Un firewall solo puede protege al sistema contra ciertos virus del Internet, y la gran mayoría de los virus se transmiten vía "floppy disks". Las organizaciones que están profundamente preocupadas acerca de los virus deben implementar medidas de control de virus dentro de la organización, en lugar de tratar de mantener los virus fuera del firewall, deben asegurarse que cada computadora tenga software de detección de virus el cual es ejecutado cuando la computadora es inicializada.

3.4. NECESIDAD DE IMPLANTAR FIREWALLS

El Internet como cualquier otra sociedad, está plagada de vándalos que disfrutan del equivalente electrónico de "escribir con pintura de aerosol en las paredes de otra gente".

El propósito del firewall es mantener a éstas personas fuera de nuestra red y mientras uno puede seguir haciendo su trabajo.

Frecuentemente, la parte más difícil de conectarse a Internet, en el caso de las grandes compañías, no es justificar el gasto sino convencer a la Dirección que la conexión es segura.

3.5. PROTECCIÓN DE LOS FIREWALLS

Los Firewalls generalmente nos protegen contra logins interactivos no autorizados desde el mundo "exterior".

Contra ataques que no son hechos a través del firewall. Algunos Firewalls bloquean el tráfico del exterior hacia el interior a la vez que le permiten el acceso al tráfico interior hacia el exterior.

Los Firewalls proveen un punto de choque donde la seguridad y auditoría pueden ser impuestos tanto para las personas externas como las internas. Los Firewalls proveen

reportes sumarizados sobre qué clase de tráfico pasa a través del firewall, así como los intentos que hubo para "romperlo".

Muchas organizaciones están aterrorizadas sobre la conexión a Internet y no tienen una política coherente sobre como deben ser protegidos los accesos "Dial-up" a través de los módems.

Los Firewalls es algo así como construir una puerta de acero de 6 pies para una casa de madera.

3.6. TIPOS DE FIREWALL

Los firewalls pueden diferir en su arquitectura y sus características. Actualmente, se encuentran en el mercado dos tipos de arquitecturas de firewalls:

3.6.1. FIREWALL DE FILTRADO DE PAQUETES

Este tipo de Firewall ofrece un control básico de acceso a la red basado en la información de protocolo contenida en los paquetes IP cuando éstos llegan al firewall, la información se compara con un conjunto de reglas de filtrado, que especifican las condiciones según las cuales se autoriza o deniega a los paquetes su acceso a la red.

Los firewalls de filtrado de paquetes o de sesiones emplean reglas de acceso para definir el control de acceso desde y hacia la red. Las reglas de acceso pueden soportar cualquier servicio. Por ejemplo, para soportar el servicio Telnet, una regla solamente tiene que

autorizar el acceso al puerto 23. Así mismo, esta flexibilidad también puede hacer más compleja la administración de las reglas, ya que especificar un número (de puerto) incorrecto va dejar una puerta abierta para la entrada de un intruso.

El filtro sobre puertos de TCP o UDP puede ser implementado directamente por un router (ruteador) filtrador de paquetes o por un host con capacidades de filtrado de paquetes.

Figura 3.1 muestra un ejemplo de reglas al protocolo TCP, en donde * (asterisco) significa cualquier dirección y >1023 cualquier puerto mayor a 1023.

TCP	*	123.5.6.7	> 1023	23	Permitir
TCP	*	123.5.6.8	> 1023	25	Permitir
TCP	*	123.5.6.9	> 1023	25	Permitir
TCP	130.6.48.254	123.5.6.10	> 1023	119	Permitir
UDP	*	123.5.*.*	> 1023	123	Permitir
*	*	*	*	*	Denegar

Figura 3.1 Tabla de Direcciones

La figura 3.2 Ilustra un esquema del funcionamiento del Firewall de filtrado de paquetes.

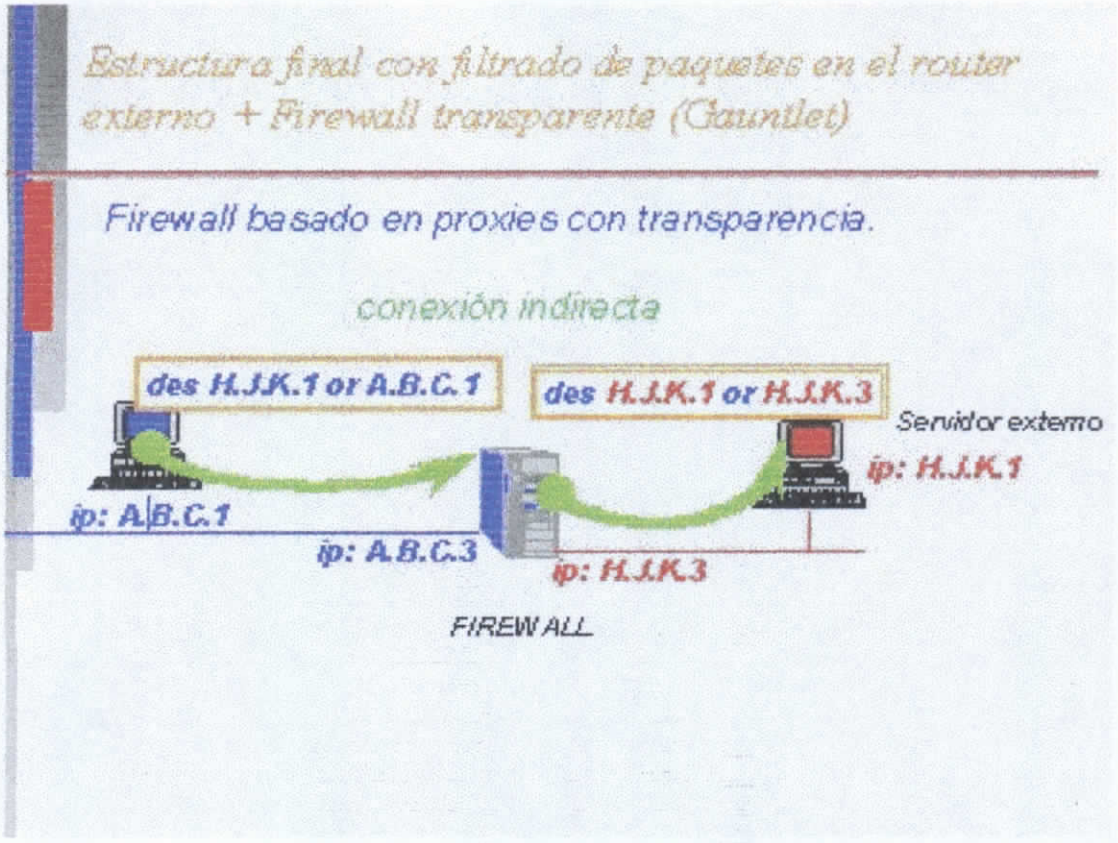


Figura 3.2 Firewall de Filtrado de Paquetes

3.6.1.1. FIREWALLS DE FILTROS DE SESIÓN

Los filtros de sesión son filtros de paquetes que mantienen información referente a cada sesión para permitir tomar decisiones más inteligentes y seguras. Los firewalls de aplicaciones no necesitan esta seguridad porque trabajan a un nivel incluso más alto. Controles de suplantación de hosts: Los controles básicos de acceso emplean una dirección IP no autenticada para identificar al solicitante. Sin embargo, esto deja una puerta abierta para los atacantes que utilicen la dirección IP de un host interno de confianza.

Las caracteristicas más comunes de un firewall que pueden ayudar a reducir la amenaza que de la suplantación de direcciones IP son al menos dos:

1.- La restricción de la "opción de la ruta origen" permite a un host controlar el ruteo del atacante para regresar a la dirección del host origen.

2.- La posibilidad de ejercer el control a través de la interfaz de la red puede ayudar también a reducir los casos de reemplazo de direcciones IP. La inclusión de la identificación de la interfaz de red en la regla o en la lista de acceso garantiza que un host de Internet no pueda pretender ser un host de la red interna.

Se debe poder permitir a un host el acceso a través del firewall en función de la interfaz de red desde la cual se recibe el paquete. Por ejemplo, permitir al host 10.X.X.X un acceso FTP (File Transfer Protocol – Protocolo de Transferencia de Archivos) a Internet cuando el paquete se recibe en la interfaz de la red interna.

La figura 3.3 muestra un esquema de los Firewall de filtrado de Sesión

Diagrama del funcionamiento del Firewall de filtros de sesión.

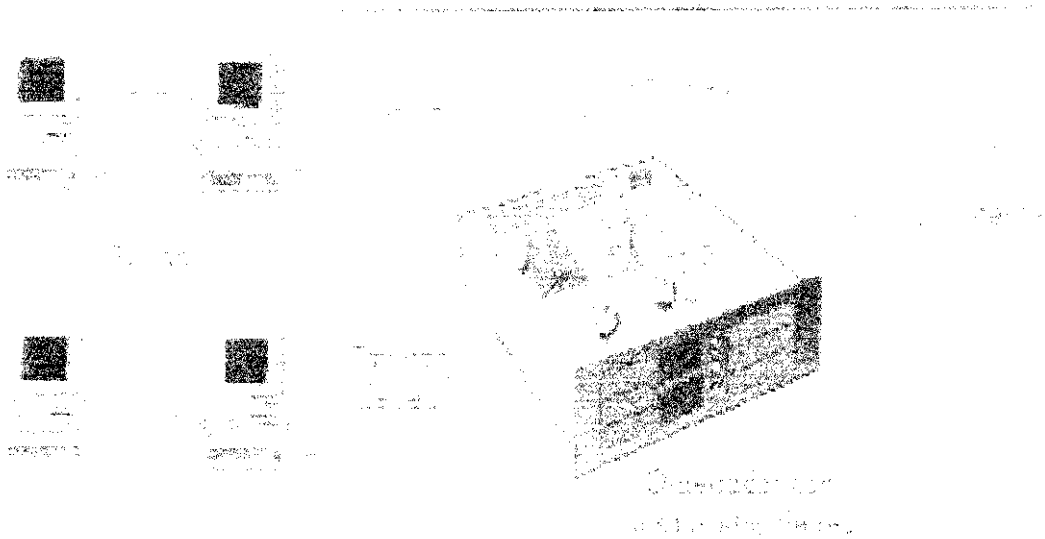


Figura 3.3 Firewall de Filtros de Sesión

3.6.1.2. FIREWALLS DE CONTROL BÁSICO DE ACCESO

La función primordial de un firewall de control básico de acceso es controlar el acceso a la red en función de la dirección del host y del servicio solicitado. Por ejemplo, puede utilizar el control básico de acceso para permitir al host utn.edu.ar (dirección de Internet) acceder a la red mediante el servicio Telnet.

Todos los firewalls disponibles ofrecen un mecanismo para controlar el acceso. También ofrecen otras características para reforzar el control de acceso, facilitar la administración del mismo y hacerlo más dificil de engañar.

La figura 3.4 ilustra el funcionamiento de un Firewall de control de acceso

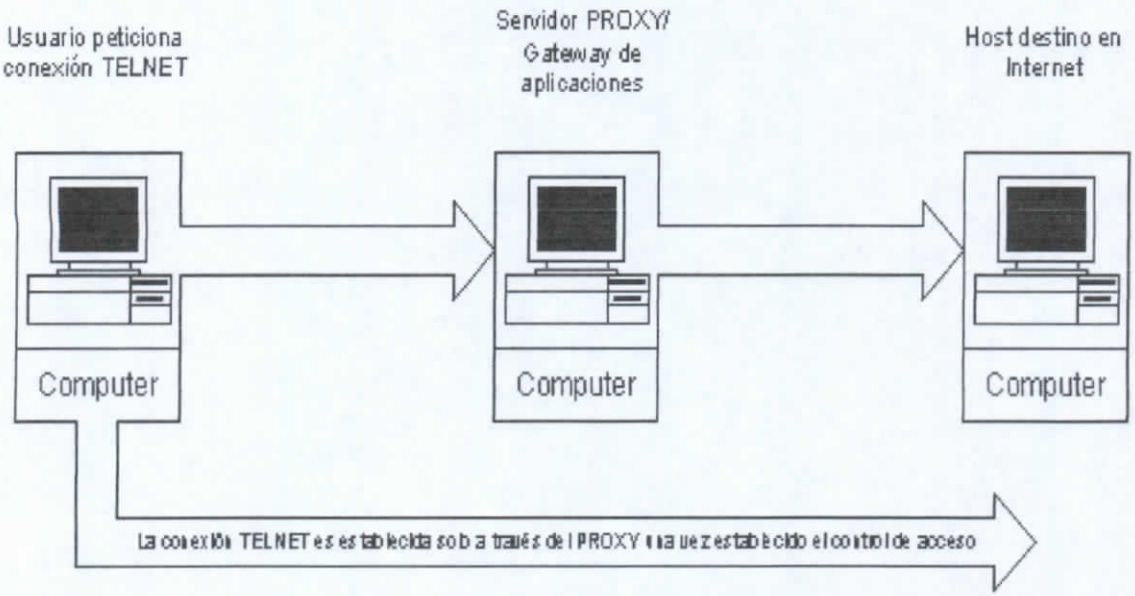


Figura 3.4 Firewall de Control de Acceso

3.6.2. FIREWALLS A NIVEL DE APLICACIÓN

Los firewalls, a nivel de aplicación, emplean dos tipos de listas de acceso, las basadas en los hosts y las basadas en los servicios: las primeras describen los conjuntos de servicios autorizados para cada host o red. Las segundas identifican los conjuntos de hosts o redes que pueden utilizar cada uno de los servicios. Las listas de acceso solamente pueden soportar políticas de seguridad simples, además resulta más sencillo comprenderlas y configurarlas. Sin embargo, muchos administradores de firewalls prefieren la flexibilidad

de las reglas de acceso, en la medida en que esté disponible una interfaz de administrador adecuada.

3.6.2.1. FIREWALL DE GATEWAY

Por su parte el firewall de gateway a nivel de aplicación también se conocen como internetworking, interrumpe la ejecución de todas las sesiones de red y crea una sesión aparte hacia el destino deseado, siempre y cuando reciba autorización para ello. A continuación, transmite la información desde la conexión original hasta la segunda conexión.

Si bien cada tipo de firewall difiere en cuanto a su funcionamiento el servicio básico ofrecido es esencialmente el mismo. Ambos actúan como un filtro entre dos redes con el fin de restringir los servicios que se ofrecen en cualquier dirección según una política preestablecida (es decir, una política de seguridad). Sin embargo, los dos tipos de firewalls difieren en lo que se refiere al nivel de control que ofrecen.

El firewall de gateway a nivel de aplicación ejerce mayor control sobre una sesión, dado que crea y mantiene la conexión actual con el exterior.

Asimismo, los firewalls presentan conjuntos de características distintos. Por ejemplo, algunos incluyen funciones de registro de actividades básicas, mientras que otros proporcionan avanzados mecanismos de alarma.

Existen variantes en los firewalls de filtrado de paquetes y en los basados en gateways a nivel de aplicación. Por ejemplo, una variante de un firewall de filtrado de paquetes mantiene información referente a las sesiones a nivel de paquete, lo que permite utilizar más información a fin de tomar decisiones más inteligentes. Esta variante se conoce en ocasiones como filtro de sesión o firewall inteligente de filtrado de paquetes.

Una variante natural de los dos tipos de firewalls descritos anteriormente es un sistema de firewall combinado que incluye un filtro de paquetes y un gateway a nivel de aplicación. El firewall CWTG (combinación de los dos anteriores) utiliza este enfoque.

En una configuración combinada los paquetes recibidos son sometidos en primer lugar a las decisiones de filtrado del filtro de paquetes. A partir de ahi, los paquetes pueden desecharse, hacerse pasar a través del kernel (núcleo del sistema operativo) hacia su destino previsto o enviarse a un proxy a fin de ser procesados posteriormente.

Un firewall combinado es la mejor solución para una Intranet que necesita la seguridad que ofrece un gateway a nivel de aplicación para ciertos servicios, velocidad y flexibilidad de un filtro de paquetes para otros tipos de servicios. No obstante, este tipo de firewalls tiende a ser más costoso, dado que proporciona más funciones y características que un dispositivo más simple, como puede ser un router (enrutador). También vale la pena tener presente que, con la combinación de routers y gateways a nivel de aplicación en una red, es posible obtener un resultado similar.

La figura 3.5 ilustra un diagrama del funcionamiento de un Firewall tipo Gateway.

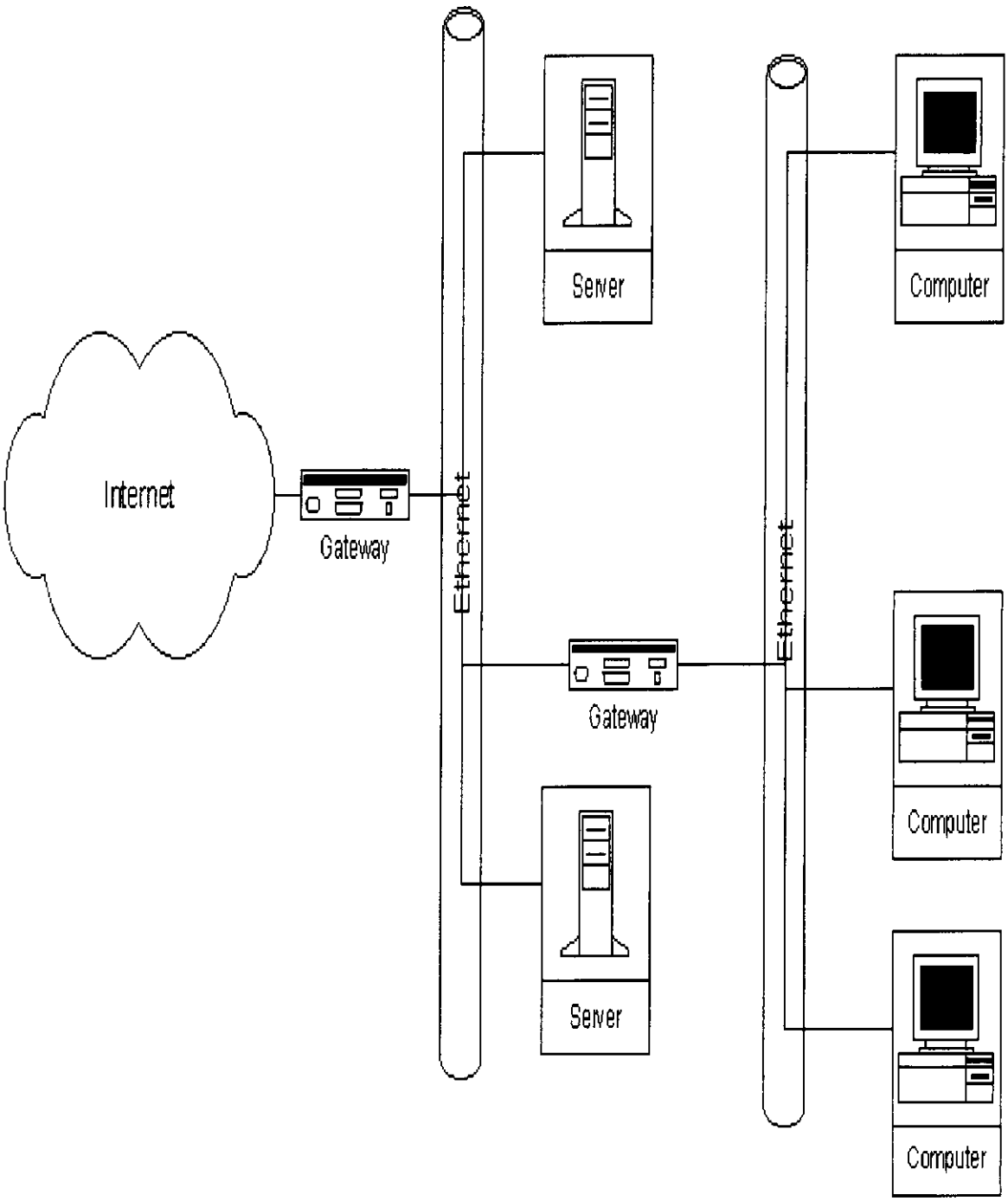


Figura 3.5 Firewall de Gateway

3.6.2.2. FIREWALLS BASADOS EN REDES

Los tipos de firewalls descritos hasta ahora se basan esencialmente en instalaciones físicas. En un firewall de este tipo, todos los componentes protectores del mismo están situados en las instalaciones de la organización a proteger. En un futuro próximo, los firewalls podrán pasar de ser un producto a un servicio que ofrecerán los proveedores de Internet. Dicho servicio se proporcionará mediante un firewall basado en red.

Un firewall basado en red está situado en la red del proveedor de servicios Internet, al cual la organización estará conectada. Observe que el firewall basado en red deberá contemplar múltiples políticas de seguridad, una para cada cliente, e impedir que éstos se mezclen.

Los firewalls basados en red tendrán que superar dos obstáculos principales antes de hacerse realidad.

El primero es la velocidad, debido a que el firewall de red gestionará el tráfico de múltiples Intranets, deberá soportar un elevado rendimiento total de procesamiento. El segundo es impedir que cualquier router situado entre el cliente y el firewall pueda evitar el firewall y permitir el acceso incontrolado a la Intranet de otro cliente.

Con toda probabilidad, los proveedores de servicios o proveedores de firewalls conseguirán superar estos obstáculos y así suministrar un servicio de firewall basado en red a sus clientes. Es posible que algunas organizaciones se sentirán incómodas poniendo en manos externas la responsabilidad de proteger su red de la Internet. Tal vez dichas

organizaciones preferirán agregar su propio firewall basado en instalaciones fisicas, posiblemente como complemento al ofrecido por el proveedor Internet.

3.6.2.3. FIREWALLS BASADOS EN HOST BASTIÓN

La arquitectura basada en un host bastión consiste en un host configurado para resistir los ataques procedentes del exterior. El blindaje del host bastión es importante porque éste se sitúa normalmente en un lugar expuesto directamente a Internet.

La figura 3.6 Ilustra el diagrama de funcionamiento del Firewall de filtrado de Host.

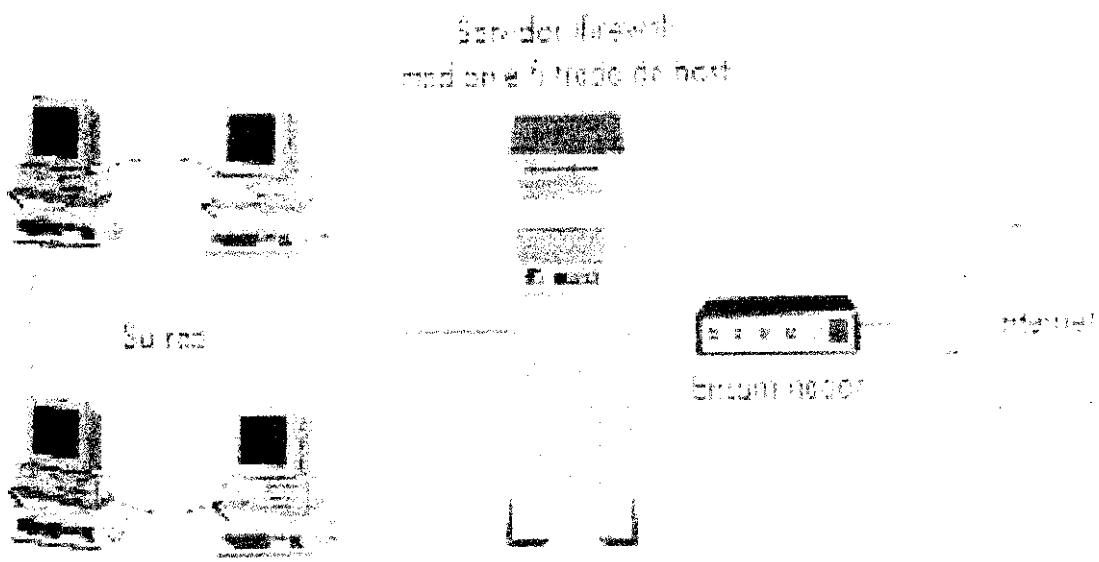


Figura 3.6 Firewall Basados en Host

Existen dos tipos de hosts bastión:

1. Firewall basado en Host Bastión de Residencia Dual
2. Firewall basado en Host Bastión de Residencia Única

3.6.2.3.1. FIREWALL BASADO EN HOST BASTIÓN DE RESIDENCIA DUAL

Al igual que la mayoría de firewalls, el host bastión de residencia dual posee una conexión a la Intranet interna y otra a la red exterior (por lo general Internet). Esta forma temprana de host bastión obligaba a los usuarios del interior a registrarse en el firewall y a efectuar desde el host bastión todas sus acciones con la red exterior.

Esta configuración aislaba a los hosts internos del exterior, pero afectaba considerablemente la capacidad de los usuarios para interaccionar con la red exterior. A medida que evolucionó la tecnología de los bastiones, fueron agregándoseles aplicaciones proxy con el fin de que actuaran en representación del usuario. El resultado de esta evolución es la arquitectura basada en un gateway a nivel de aplicación descrita anteriormente.

3.6.2.3.2. FIREWALL BASADO EN HOST BASTIÓN DE RESIDENCIA ÚNICA

Este tipo de firewall basado en un host bastión es un host de residencia única (que significa que sólo hay una conexión de red) conectado a lo que se conoce como una red de perímetro (subred).

El servicio de firewall lo proporciona una combinación de los dos routers de filtrado, la red de perímetro y el host bastión. El router exterior filtra los servicios no soportados por la red interna o por el host bastión.

El router interno, conocido en ocasiones como router de bloqueo, limita todos los servicios no soportados por la red interna y sirve de protección principal a ésta.

El host bastión proporciona a los usuarios externos servicios como un gateway de correo electrónico, FTP anónimo (es decir, un sitio que permite la transferencia de ficheros a clientes remotos), consultas del Domain Name System (DNS) o un servidor Web HTTP.

Una red de perímetro puede contener varios hosts bastión, que proporcionan cada uno de ellos uno o más servicios.

La red de perímetro, conocida también como "subred" no debe contener recursos considerados como delicados. La idea principal que hay detrás de los hosts bastión es que no son insustituibles. Están configurados para resistir los ataques, pero si son vulnerados no suponen una amenaza para la red interna. De este modo, ningún host que se encuentre dentro de ésta debe confiar en ellos.

La Figura 3.7 Ilustra un diagrama del funcionamiento de un Firewall basados en subred.

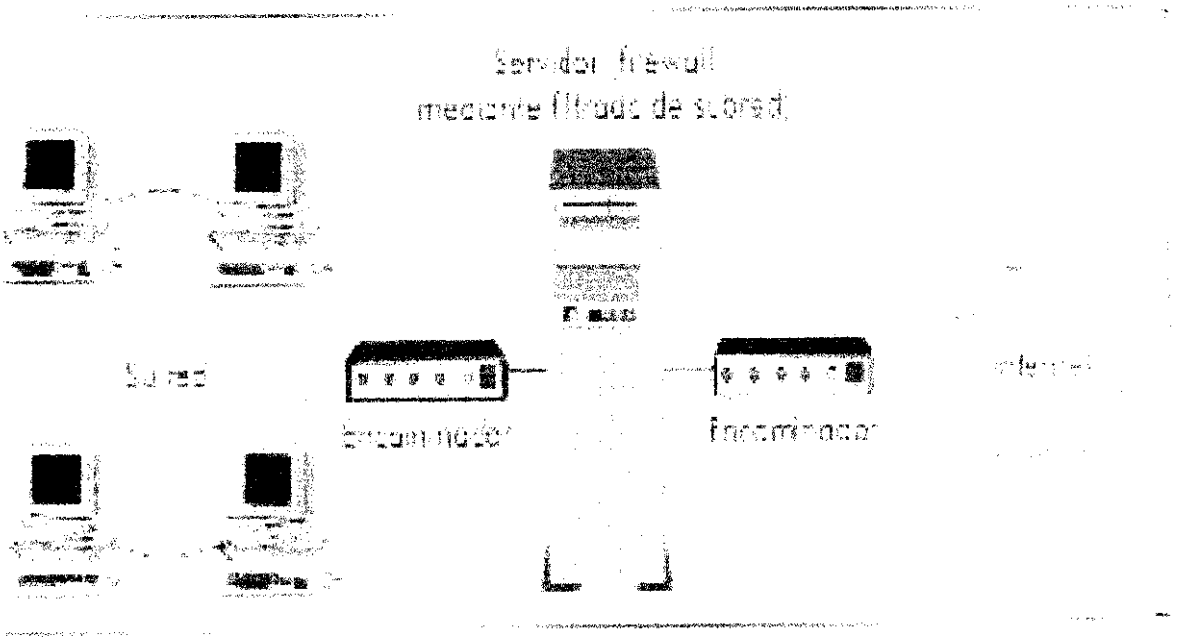


Figura 3.7 Firewall de Conexión Única de Red

3.6.3. FIREWALLS DEL FUTURO

Se encuentran en un punto entre los del Nivel de Red y los del Nivel de Aplicación. Serán firewalls con un sistema de "fast packet-screening" que auditará y llevará un archivo histórico de los datos que pasan por él.

Se incorporará la encriptación para que ellos protejan el tráfico, que pase a Internet, firewalls con encriptación "end-to-end" (punto a punto) pueden ser utilizados por las organizaciones con múltiples puntos de conectividad con Internet para utilizar a esta red

como un backbone privado sin la preocupación de que los datos o passwords están siendo vistos en Internet.

Todos los tipos de firewalls que se han descrito anteriormente se utilizan actualmente, lo que da a entender que no existe un tipo "óptimo". El firewall más adecuado a un entorno determinado depende de diversos factores, como los conocimientos de los administradores, los tipos de servicios que se pretende soportar, el presupuesto y las necesidades de la organización. Antes de decidir cuál es el firewall que mejor se adapta a las necesidades propias, deben evaluarse las amenazas a las que está sujeta la Intranet. Es necesario, asimismo, desarrollar una política de seguridad.

En el momento de determinar cuál es el firewall más adecuado, tal vez la consideración más importante deba basarse en los tipos de servicio que se pretende soportar a través del mismo. Por ejemplo, quizá se desee soportar FTP, Telnet, correo electrónico y HTTP. Cada servicio posee su propio conjunto de puntos débiles y protecciones disponibles.

Independientemente del firewall que adquiera, éste debe incluir las características adecuadas que proporcionen la suficiente protección a los servicios que se pretende soportar.

3.7. ELABORACIÓN DE BARRERAS DE PROTECCIÓN

Existen varios métodos para construir una barrera de protección. Las organizaciones con talento en la programación y recursos financieros suficientes, en general prefieren usar un método personalizado de barreras de protección para proteger la red de la organización. Si se ejecuta de manera adecuada, tal vez éste sea el método más eficaz y por supuesto el más costoso.

Otras organizaciones prefieren usar los productos comerciales existentes, así como personalizarlos y configurarlos para cumplir la política de seguridad de red de esas organizaciones.

A continuación describiremos las distintas arquitecturas con las cuales se puede implementar una barrera de protección Firewall para nuestra red.

3.8. ARQUITECTURA DE FIREWALLS

Las tecnologías de filtrado de paquetes que se emplean en los firewalls constituyen una manera eficaz y general para controlar el tráfico en la red. Tales tecnologías tienen la ventaja de no realizar ningún cambio en las aplicaciones del cliente y el servidor, pues operan en las capas IP y TCP, las cuales son independientes de los niveles de aplicación según se establece en el modelo OSI. Por otro lado, los enfoques de la filtración de paquetes no han declarado muchos requerimientos de seguridad, por la información incompleta con la que trabajan.

Sólo la información de las capas de transporte y red, como las direcciones IP, los números de puerto y las banderas TCP están disponibles para las decisiones de filtración. En muchas implementaciones de los filtros de paquete, el número de reglas puede ser limitado; además, mientras mayor sea este número, habrá una alta penalización en el desempeño, a causa del proceso adicional necesario para las reglas complementarias.

En vista de la falta de información de contexto, ciertos protocolos como el UDP (User Datagrama Protocol) y RPC (Remote Process Control) no pueden filtrarse con efectividad. Además, en muchas implemetaciones, faltan los mecanismos de intervención y alerta.

Muchas de estas implementaciones de filtros pueden requerir un alto nivel de comprensión de los protocolos de comunicación y su comportamiento, cuando se utilizan por diferentes aplicaciones.

Los dispositivos de filtración de paquetes, casi siempre se mejoran mediante otros tipos de dispositivos llamados barreras de protección. Las barreras de protección se llaman así porque operan en las capas superiores del modelo OSI y tienen información completa sobre las funciones de la aplicación en la cual basan sus decisiones. Estos constituyen la mayoría de los firewalls tal cual hoy los conocemos.

3.8.1. ARQUITECTURA DE DOS BASES

Un firewall de dos bases no es nada más y nada menos que un firewall con dos interfaces de red, que permite asilar una red interna de una red externa no confiable. Como este anfitrión no envía ningún tráfico TCP/IP, bloquea por completo cualquier tráfico IP entre las redes no confiables interna y externa.

Muchos servicios de Internet son en esencia de almacenaje y envío. Si estos servicios se ejecutan en el anfitrión, pueden configurarse para transmitir servicios de aplicación desde una red hacia la otra. Si los datos de aplicación deben cruzar la barrera, es factible configurar los agentes emisores de aplicación para hacer la ejecución en el anfitrión.

Estos agentes son programas especiales, utilizados para enviar solicitudes de aplicación entre dos redes conectadas. Otro método es permitir que los usuarios se conecten al anfitrión de dos bases y después tengan accesos a los servicios externos desde la interfaz de red externa del anfitrión.

Si se usan los emisores de aplicación, el tráfico de la aplicación no puede cruzar la barrera, a menos que el emisor de aplicación se ejecute y se configure en el servidor de barrera de protección. Esta acción es la implementación de la politica "si no está permitido de manera expresa, está prohibido". Si se autoriza a los usuarios conectarse en forma directa a la barrera de protección, puede comprometerse la seguridad de ésta, porque, la barrera es el punto central de la conexión entre la red externa y la interna. Por definición, la barrera de este tipo está en zona de riesgo. Si el usuario selecciona una contraseña débil o

compromete su cuenta de usuario (al proporcionar la contraseña), la zona de riesgo quizá se extienda a la red interna y por lo tanto eliminará el objetivo de la barrera.

Si se mantienen registros adecuados de la conexiones de usuarios, es posible rastrear las conexiones no autorizadas a la barrera, en el momento que se descubra una brecha de seguridad. En cambio si se impide que los usuarios se conecten en forma directa a la barrera, cualquier intento de conexión directa se registrará como algo notorio y como una brecha potencial de seguridad.

Este tipo de firewall, con una interfaz mirando a cada red, es la configuración básica usada en las barreras de protección. Los aspectos delicados son que el enrutamiento se encuentra inhabilitado y que la única ruta entre los segmentos de red es a través de una función de capa de aplicación. Si el enrutamiento se ha configurado de manera errónea por accidente (o por diseño) para permanecer activo, se ignorarán las funciones de la capa de aplicación de las barreras de protección.

La mayoría de estas configuraciones están montadas sobre máquinas UNIX. En algunos implementaciones de este sistema operativo, las funciones de enrutamiento se activan de manera predeterminada, por lo cual es importante verificar que dichas funciones están inhabilitadas.

3.8.2. ARQUITECTURA NO RECOMENDADA

En realidad la mayoría de los autores y los administradores con experiencia de campo en el tema de la utilización de firewalls como parte de una política general de seguridad de una organización establecen una serie de consejos en cuanto a la configuración tanto de los firewalls como de los routers de manera tal de no dejar back-doors (puerta trasera) que permitan el ingreso no autorizado y que no podamos rastrear.

Para hablar de arquitecturas no recomendadas seguramente estaremos situados en la deformación de alguna de las expuestas anteriormente.

Entre ellas encontramos por ejemplo la utilización de DMZ (zonas desmilitarizada) con servidores con más de una interfaz de red, de las cuales una apunta a la red interna, o servidores que contengan pools de módems para conexiones dial-up y pertenezcan a la red interna.

El primero de los casos se da cuando por razones de nuestro negocio debemos ofrecer servicios a la red externa, con lo cual podemos pensar en ubicar el servidor en una DMZ (zonas desmilitarizada). Hasta aquí no hay problema, pero si por alguna razón pensáramos en agregar otra tarjeta de red apunta a la red interna, para servicios de backup centralizado o lo que fuere, estamos generando un back-door (puerta trasera), que permite el acceso directo a nuestra red interna obviando el filtrado del firewall.

El segundo paso, constituye un punto crucial en la protección de nuestra red. Hoy por hoy son necesarias las conexiones dial-up para todo lo que constituye el acceso remoto de nuestros usuarios, para tareas de mantenimiento, por ejemplo, pero este tipo de acceso debe estar correctamente administrado y centralizado, de manera tal que podamos asegurarnos que todo el tráfico que se genere a partir de esos servidores sea controlado. En el ejemplo planteado basta con franquear al servidor dedicado a las conexiones dial-up, que por cierto no tiene por que tener todas las medidas de protección que puede tener un firewall, para tener acceso a toda la red interna sin someterse a los filtros de los firewalls.

En realidad se debe considerar como arquitectura no recomendada a aquellas que de alguna u otra forma evitan que el tráfico de red se someta al análisis de un firewall.

3.9. INSTALACIÓN DE FIREWALLS

Cuando se decide instalar un firewall el primer y más importante punto tiene que ver con la decisión política de cómo se quiere operar el sistema: todo aquellos no especificado se bloquea; esta política pretende que el firewall bloquee todo el tráfico, y las aplicaciones que se deseen "dejar pasar" deberán ser especificadas una por una y con el razonable fundamento del caso. Este tipo de decisión es altamente recomendada, pues crea un ambiente muy seguro en el cual solo algunos servicios "selectos" son soportados.

Por el otro lado y totalmente opuesta a la primera se encuentra la política de permitir todo aquello que no este negado. Esta política supone que el firewall dejará pasar todo el tráfico salvo aquellos servicios que se han considerado "peligrosos" y que se configurarán caso

por caso. Este tipo de decisión crea un ambiente más flexible, con más servicios disponibles para los usuarios. Probablemente la decisión tenga más que ver con cuestiones políticas que con un diseño técnico.

El segundo punto es determinar el nivel de auditoria y control a implementar y por último el tema financiero. En el aspecto financiero, podemos encontrar soluciones que no nos costarán nada en cuestión de dinero pero sí el de mantenimiento. [RefSerc]

Un firewall es a menudo instalado en el punto donde la red interna se conecta con Internet.

3.10. ADMINISTRACIÓN DE FIREWALLS

Una diferencia importante entre los diversos tipos de firewalls es el método empleado para la administración de los mismos. Como hemos mencionado, la protección que ofrece un firewall puede llegar a ser nula si no es administrado correctamente.

Una interfaz fácil de utilizar y con un número mínimo de opciones de configuración reduce la posibilidad de que se produzcan errores de administración naturalmente, un número menor de opciones de configuración puede significar también menor flexibilidad de configuración.

Existen tres clases de interfaz del administrador de firewalls:

- Administración basada en ficheros de texto.

- Administración basada en menús de texto.
- Administración basada en GUI (Usuario).

3.10.1. ADMINISTRACIÓN BASADA EN FICHEROS DE TEXTO

La interfaz basada en ficheros de texto es la de uso más extendido en lo que respecta a los routers y a los firewalls de cosecha propia. Este tipo de interfaces permiten al administrador editar un archivo específico donde puede introducir parámetros de configuración específicos. Se trata de la interfaz de elección para los administradores de sistemas UNIX/LINUX tradicionales, dado que ofrece una interfaz de control a bajo nivel con los mecanismos del firewall.

La desventaja de dicho control a bajo nivel es que resulta mucho más fácil cometer errores, ya que, al editar un fichero, pueden producirse errores de escritura u otros errores técnicos que, en un sistema basado en menús, es menos probable que ocurran.

3.10.2. ADMINISTRACIÓN BASADA EN MENÚS DE TEXTO

La interfaz de administrador basada en menús de texto presenta un menú basado en texto que reduce la probabilidad de producirse errores pero que proporciona menor capacidad de control para el administrador. Sin embargo, la posibilidad de error no queda totalmente excluida, dado que el administrador no siempre puede ver el efecto de algunos cambios.

3.10.3. ADMINISTRACIÓN BASADA EN GUI (USUARIO)

La interfaz gráfica de usuario, o GUI, para administradores incorpora ventanas, botones, menús desplegables y pantallas de ayuda que facilitan el trabajo de configuración. La mayoría de proveedores ha optado por incluir esta interfaz en sus productos, puesto que tiende a ser más fácil de utilizar y no es susceptible a muchos de los errores que pueden producirse en los otros dos tipos de interfaz.

Algunos productos firewalls ofrecen la posibilidad de realizar la administración centralmente, lo que permite configurar múltiples firewalls desde una ubicación individual remota.

Esta característica puede ser importante si existen múltiples firewalls situados en diversas ubicaciones pero se dispone solamente de un especialista en firewalls en una de esas ubicaciones.

La administración centralizada permite a este especialista configurar cada uno de los firewalls desde un sitio central. [Reflsa]

3.11. VELOCIDAD DE FIREWALLS

Con el objeto de proteger la red de los ataques procedentes de Internet, es necesario canalizar todas las comunicaciones que se hacen desde y hacia ésta a través del Firewall.

Un Firewall lento puede convertirse en un cuello de botella teniendo en cuenta la alta transmisión de paquetes con información multimedia (web).

3.12. PROTOCOLOS Y SERVICIOS SOPORTADOS POR FIREWALLS

Los servicios se refieren a los protocolos a nivel de aplicación que el firewall reconoce y autoriza. Es muy importante que los firewalls nieguen cualquier servicio que no puedan reconocer.

Los servicios se identifican mediante el número del puerto de destino TCP o UDP. Estos servicios tienen números de puertos conocidos que son fijos (por ejemplo Telnet: puerto 21).

3.12.1. TIPOS DE PROTOCOLOS SOPORTADOS POR LOS FIREWALLS

A continuación presentamos una lista de protocolos de servicios mínimos y básicos soportados por la mayoría de los firewalls y una breve descripción de los mismos:

3.12.1.1. DNS (Protocolo TCP o UDP número de puerto 53)

Por lo general, el Domain Name System. (DNS) no es un servicio que utilicen directamente los usuarios. Cuando un usuario solicita conectarse a un host, la aplicación de red llama al DNS para averiguar la dirección IP asociada al host. Si el servidor DNS local del usuario no tiene esta información, la solicita a otros servidores DNS.

Los servidores DNS comparten información. Es precisamente esta capacidad de la que debemos protegemos, debido a que no es deseable que ningún servidor DNS de Internet pueda actualizar los nombres de servidor de la red propia. En una situación de este tipo, los atacantes pueden redefinir la dirección de un host externo a la Intranet con una dirección de confianza de la red con la dirección de un host interno

El firewall debe permitir a los servidores DNS de la red propia el acceso a servidores de nombres del exterior e incluso su actualización con direcciones nuevas, negando a los externos poder actualizar los registros de los servidores de nuestra red.

Los firewalls a nivel de aplicación desprovistos de la capacidad de un proxy invisible, no necesitan soporte DNS porque realizan la consulta directamente a un servidor de nombres externo.

3.12.1.2. FINGER (Protocolo TCP Puerto 79)

El servicio finger fue desarrollado para permitir a los usuarios de una red poder localizar a otros usuarios. Gracias a finger es posible averiguar nombres de entrada en el sistema (logins), y los nombres reales de los usuarios. Esta es una información valiosa para un intruso potencial, por lo que el firewall debe prohibir cualquier solicitud de finger procedente del exterior. Una alternativa para descartar las solicitudes de finger procedentes del exterior es tener un proxy de finger que muestre un mensaje estándar como "Esta red no soporta el servicio finger".

A menudo los intrusos emplean solicitudes finger a modo de sonda. Por esta razón, algunos administradores instalan un servidor proxy de finger que responde a una solicitud finger efectuando a su vez una solicitud inversa a fin de obtener información acerca del solicitante. Sin embargo, hay que tener cuidado con esta política porque la solicitud de finger inversa puede ser atrapada por un servidor finger que haya sido diseñado también para efectuar una solicitud finger inversa. El resultado será un bucle interminable de solicitudes finger entre ambos hosts, lo cual bloqueará innecesariamente recursos de los dos sistemas.

3.12.1.3. FTP (Protocolo TCP Puerto número 21)

FTP o File Transfer Protocol (protocolo de transferencia de archivos). Se trata del protocolo estándar para transferir archivos entre sistemas que soporta una autenticación sencilla de contraseñas.

Para cada archivo FTP o transferencia de información, se establece habitualmente una conexión de red aparte desde el host de destino hacia el host desde donde se origina la conexión FTP. El firewall debe ser capaz de permitir esta segunda conexión en el sentido contrario ya que si no, no se transferirán los datos. Por lo general, un puerto origen TCP de 20 identifica la conexión de datos.

3.12.1.4. GOPHER (Protocolo TCP Puerto número 70 y otros)

El protocolo y servicio gopher proporciona un sistema sencillo de menúes textuales cuya función es ayudar a encontrar información en Internet.

Gopher es uno de los precursores del HTML, y los servidores y clientes gopher plantean las mismas amenazas que los clientes y servidores HTTP. Por ejemplo, es posible engañar a un cliente gopher para que ejecute órdenes no autorizadas en la computadora del usuario.

Al igual que HTML, los servidores gopher pueden estar disponibles en otros puertos además del puerto predeterminado.

3.12.1.5. ICMP (Protocolo ICMP)

Internet Control Message Protocol (ICPM) es un protocolo soportado por encima de IP, a nivel de TCP o UDP. IP lo utiliza para enviar mensajes de error o de prueba entre sistemas distintos. Un mensaje ICMP contiene campos de tipo y de código que indican un mensaje

predefinido como "no se puede contactar con la red" o "acceso denegado para propósitos de administración".

La conocida aplicación de prueba ping emplea el protocolo ICMP para enviar mensajes de petición de eco QCMP tipo 8, código 0, para comprobar si es posible acceder a un host. El host destino responde con un mensaje de respuesta, de echo (ICMP tipo 0, código 0). El firewall puede configurarse para permitir algunos mensajes ICMP y denegar otros. Por ejemplo, tal vez se desee impedir a los hosts de Internet hacer un "ping" en la red propia para averiguar qué hosts pueden ser atacados. Por otra parte, quizá sea deseable que los mensajes de error regresen a los hosts de la red.

3.12.1.6. IRC (Protocolo TCP Puerto número 6667)

La aplicación Internet Relay Chat (IRC) ofrece la posibilidad de participar en conferencias con múltiples usuarios en un entorno de texto. Con una aplicación IRC cliente, un usuario puede ponerse en contacto con un servidor IRC y unirse a una conversación.

La principal amenaza asociada a este servicio no es inherente al protocolo, sino que representa más bien una amenaza de ingeniería social. Entre otras cosas, la ingeniería social es el acto que puede perpetrar un atacante para obligar a un usuario o administrador a que proporcione información de autenticación o a que reduzca los controles de seguridad. Un usuario de Internet puede intentar convencer a un usuario interno para que modifique la configuración de su computadora a fin de proporcionar una característica determinada.

Dicha modificación puede ser un método del que le servirá al usuario externo para penetrar en la computadora del usuario interno. Un proxy IRC situado en el firewall no tiene una gran utilidad para Contrarrestar esta amenaza.

3.12.1.7. E-mail (Protocolo TCP Puerto número 25)

Mail, o "e-mail", es el servicio más utilizado en Internet. Permite a los usuarios enviar mensajes sin que sea necesario establecer una conexión directa entre el host remitente y el host destinatario. Un mensaje de correo puede recorrer un gran número de hosts antes de llegar a su destino. El protocolo de correo estándar que se emplea en Internet es el Simple Mail Transfer Protocol (SMTP).

3.12.1.8. MBONE (Protocolo IP)

Multicast Backbone (MBONE) se emplea para dirigir paquetes multitransmitidos a través de routers que no soportan el direccionamiento de multitransmisión. Este es un mecanismo que sirve para retransmitir el mismo paquete IP hacia varios sitios de Internet. Se emplea en servicios de conferencia en tiempo real como radio Internet y suministros de vídeo. Algunos routers no soportan el direccionamiento de multitransmisiones. MBONE encapsula un paquete multitransmitido en un paquete IP de transmisión única (estándar) para que pueda atravesar un router que no soporta la multitransmisión. El número de protocolo indicado en el campo de protocolo IP es IP (protocolo 4). Este paquete "IP incluido en IP" es recibido por otro router de multitransmisión que suprime el paquete IP multitransmitido original y lo vuelve a enviar.

La amenaza que supone permitir la entrada de paquetes MBONE en la red es nuestro desconocimiento del protocolo y servicio del paquete IP interno. Un firewall que soporte MBONE debería, como mínimo, examinar la dirección del paquete IP interno y asegurarse de que es un paquete multitransmitido. También sería útil poder realizar un filtrado según el protocolo y servicio (en los casos aplicables) del paquete IP interno.

3.12.1.9. NETWORK NEWS (Protocol TCP Puerto número 119)

Network News es otro servicio de uso bastante generalizado. Permite a los usuarios acceder a newsgroups a fin de leer información o de participar en debates. Los newsgroups constan de una serie de mensajes que tienen un tema común. Los usuarios pueden leer estos mensajes y agregar los suyos propios. Existe un newsgroup para prácticamente cualquier tema imaginable. El protocolo empleado es el Network News Transfer Protocol (NNTP).

3.12.1.10. NFS (Protocolo UDP, Puerto número 2049)

Network File System (NFS) permite a los usuarios compartir sistemas de ficheros con otros usuarios. Este tipo de característica es un estándar de las redes de PC como Novell Netware o Microsoft Windows. El NFS estándar proporciona muy poca seguridad y, por eso, es vulnerable a los ataques. La mayoría de expertos en firewall recomiendan no permitir conexiones NFS a través del firewall, aunque muchos administradores de Intranets (especialmente en entornos académicos) no están dispuesto a prescindir de este servicio.

De todos modos, los gateways a nivel de aplicación no soportan habitualmente este servicio y el filtrado de paquetes no elimina el riesgo que trae el mismo.

3.12.1.11. PORT MAPPER (Protocolo TCP o UDP Puerto número 111)

Las aplicaciones Remote Procedure Call (RPC) emplean un asignador de puerto para obtener el número de puerto TCP o UDP actual de un servicio. Si bien se dice que las aplicaciones servidor utilizan números de puerto conocidos, esto no es siempre cierto.

Algunas aplicaciones servidor pueden ejecutarse en cualquier número de puerto y dependen de la aplicación servidor de asignador de puerto para dirigir los clientes hacia el número de puerto apropiado. Esto dificulta enormemente la posibilidad de que el firewall pueda realizar el filtrado o establecer un proxy para dichas aplicaciones, puesto que el número de puerto de las últimas puede cambiar en cualquier momento.

3.12.1.12. RLOGIN (Protocolo TCP, Puerto número 513)

El comando rlogin (login remoto) son utilizados para acceder de un sistema local a otro remoto. Pero no se recomienda su uso para acceder a/o desde Internet porque la mayoría de ellos no soportan funciones adecuadas para la autenticación de usuarios.

3.12.1.13. TELNET (Protocolo TCP Puerto número 23)

TeInet es el protocolo y aplicación estándar para la entrada (login) en sistemas remotos. Proporciona una conexión entre dos sistemas basada en carácter. Todos los corta-fuegos de gateway basados en aplicaciones soportan este proxy. Muchos de ellos pueden además autentificar el usuario Telnet en el cortafuegos.

3.12.1.14. SNMP (Protocolos TCP y UDP, Puertos 161 y 162)

Simple Network Management Protocol (SNMP) es el protocolo que emplea una estación para supervisar y configurar dispositivos de red como routers, hubs y hosts. Los dispositivos de red escuchan en el puerto 161 a la espera de órdenes procedentes de la estación de gestión de la red. La estaciones de gestión de red escuchan en el puerto 162 a la espera de trampas (es decir, alarmas) procedentes de los dispositivos de la red.

La versión 1 de SNMP es un protocolo que está extensamente implementado, pero no incluye funciones robustas de autenticación. Los dispositivos de red responden a las peticiones que reciben, además de reconfigurarlas. Por este motivo, un gran número de dispositivos de red están configurados para proporcionar solamente información de estado y desautorizar cualquier intento de reconfiguración procedente de la red.

La versión 2 estándar de SNMP soportará funciones de autenticación.

3.12.1.15. WWW (Protocolo TCP Puerto número 80 y otros)

Probablemente, la World Wide Web (www) es la principal responsable del repentino interés y expansión que ha experimentado Internet actualmente. El principal protocolo de servicio empleado por la Web es el Hypertext Transfer Protocol (HTTP), que permite a los usuarios transferir documentos desde un servidor HTTP Este protocolo está soportado por aplicaciones cliente gráficas conocidas como navegadores o browsers.

Son varios los problemas de seguridad que se han relacionado con el HTTP y los servidores y navegadores asociados al mismo

3.12.1.16. X 11 (Protocolo TCP Puerto número 6.000 y superiores)

X 11 se refiere a la especificación correspondiente al entorno gráfico de usuario, de uso generalizado en estaciones de trabajo UNIX. La mayoría de servidores X soportan más de un puerto Xserver 6.001, y es posible emplear números de puerto superiores a este último. El protocolo X 11 es un servicio muy potente que permite a una aplicación remota presentar gráficos y aceptar órdenes de un ratón en una estación de trabajo X o en un PC que soporta una interfaz XWindows. Sin embargo, esta potencia se proporciona a expensas de un cierto riesgo para la seguridad. La aplicación remota puede tomar completamente el control de la pantalla, del teclado e incluso del ratón. Si tiene previsto establecer conexiones X 11 desde Internet, el firewall debe poder soportar este tipo de operaciones, además de filtrar las conexiones o comandos X 11 no deseados. [RefNetsearch]

3.13. DETECCIÓN DE INTRUSOS

La mayoría de firewall son armas defensivas que sirven para proteger la red. Pero algunos firewall están provistos de características que permiten tomar la iniciativa.

Los administradores pueden emplear estas herramientas para atraer a un posible atacante o malhechor hacia una trampa. Estas trampas se conocen en ocasiones como "cebos y trampas" o 'tarros de trúel"; proporcionan a los intrusos la impresión de que han conseguido entrar en la red, pero, en realidad, éstos son dirigidos hacia un lugar seguro del firewall o un sistema seguro, mientras dan al administrador tiempo suficiente para recoger información adicional sobre el intruso para identificarlo y tal vez perseguirlo.

No es necesario que el firewall tenga una característica especial para crear estas trampas. Es posible configurarlo para dirigir el tráfico hacia un host especial que no contiene información de valor pero que, en su lugar, contiene archivos ficticios para mantener ocupado al intruso. También puede prepararse una alarma que avise al administrador siempre que alguien accede a dicho sistema.

3.13.1 IDS (Intrusión Detection Systems) Sistema de Detección de Intrusos

Varias cualidades importantes de los IDS los ubican bastante mas allá de los "network management systems", los "routers", los "firewalls" y otros medios de protección de redes.

Todos los productos, con excepción del Sessionwall-3, de Abirnet Inc. (Dallas), constan de un monitor y una "management station" (estación de administración) que recoge información de monitores (Sessionwall-3 es manejado de forma local).

A diferencia de los productos de monitoreo remoto (RMON), los IDS no usan SNMP- que en estos momentos carece de rasgos de seguridad claves- para transmitir información del monitor al gerente. En lugar de ello, los IDS utilizan diversos medios de autenticación y codificado. Todas las interfaces de monitoreo de los productos, con excepción de ID-Trak, de Internet Tools Inc. (Fremont, California) son pasivas, de forma tal de que los agresores no estarán en condiciones de detectar nada si hay un IDS escuchando.

Los productos IDS que se probaron incluyen también rutinas predefinidas para detectar ataques específicos, y permiten a vendedores y usuarios agregar rutinas que detectan ataques nuevos apenas se los descubre. De todas maneras, existen grandes diferencias en cuanto a qué tipo de definiciones están disponibles para los usuarios.

3.14. EL MERCADO Y LOS FIREWALLS

Hoy por hoy en el mercado existen infinidad de proveedores de aplicaciones de firewalls, que están complementadas con los esfuerzos por elevar el nivel de seguridad de los distintos sistemas operativos.

Estas aplicaciones de firewalls están disponibles para los distintos sistemas operativos y si bien en líneas generales todos cumplen básicamente las mismas funciones, cada proveedor tiene sus características particulares que se van igualando de uno a otro con el correr del tiempo y el desarrollo de los distintos Firewalls.

Entre los productos más utilizados se encuentran Firewall-1, Pix y Raptor, de las empresas Chekpoint, Cisco y HP respectivamente.

Indagando un poco en el mercado local se puede comprobar que el producto de Cisco esta siendo muy utilizado, debido a su integración "nativo" en todas aquellas organizaciones que están utilizando routers Cisco. [RefSisco] [RefHp] www.cisco.com

Además de verificar las características particulares de cada uno se pueden bajar de Internet versiones de evaluación para corroborar las distintas facilidades como así también observar como se adapta a la organización donde se implementará en nuestro caso en la PUCESA.

CAPÍTULO IV

DESARROLLO DE UN FIREWALL PARA LA PUCESA

4.1. DESCRIPCIÓN GLOBAL DEL PROYECTO

Durante el proceso de investigación que se ha venido realizando durante el desarrollo de los capítulos anteriores, se ha establecido el propósito del presente proyecto, él cual es desarrollar un software de Seguridad de filtraje de paquetes IP, que actúe como Firewall (muro de seguridad) para la red de la Escuela de Ingeniería de Sistemas de la Pontificia Universidad Católica del Ecuador (PUCESA).

Luego de haber enfocado el proyecto de manera general, proseguiremos a establecer la metodología que se siguió para el desarrollo del Firewall.

4.2. METODOLOGÍA PARA EL DESARROLLO DEL SISTEMA

Para la elaboración del presente proyecto, hemos considerado sustentarnos en la metodología de cascada que ha sido aplicada en el desarrollo del Firewall de la PUCESA.

A continuación se describen los diferentes procesos que se tomaron en cuenta en la presente investigación:

a. Proceso de Pre-Desarrollo: En esta fase se realizó un estudio de las actividades, estructura y características de la red de la Escuela de Sistemas de la PUCESA; lo que significó recabar la mayor información posible de la misma, para poder optar por la mejor alternativa de solución.

b. Escogitamiento de la herramienta o lenguaje de programación: En esta fase se realizó un estudio sobre la herramienta de programación seleccionada que se ajuste al Hardware y software del Servidor de la red.

c. Proceso de Desarrollo: En esta fase se determinó qué información se debe obtener, en dónde se debe obtener, así como la estructuración de datos y algoritmos a utilizar para la elaboración del Firewall, llegar a implementarlo, y por último desarrollar el tipo de interfaz. En este proceso se siguió tres sub-procesos, los cuales fueron:

- **Proceso de Requerimientos:** Se definen las características o requerimientos específicos que debía cumplir el software.

- **Proceso de Análisis y Diseño:** En esta fase se realizó un estudio de necesidades, las mismas que sirvieron de inicio para definir el software, para luego proceder a diseñar la estructura y arquitectura del software. Terminado el análisis y diseño correspondiente se procedió a codificar el software.

- **Procesos de Implementación:** En esta fase se determinó qué herramientas y su configuración eran necesarias en el servidor para que funcione el sistema desarrollado.

sobre todo en lo que se refiere a las versiones del software y cambio de plataformas de hardware, ya que el desarrollo no fue realizado directamente en el servidor.

Una vez explicada la metodología utilizada, a continuación se explica en forma detallada el desarrollo del Firewall de la PUCESA.

4.3 PROCESO DE PRE-DESARROLLO

Previo a la realización del Software del Firewall se realizó un levantamiento de información de la red informática de la Escuela de Ingeniería de Sistemas de la PUCESA, determinándose lo siguiente:

La red está compuesta de dos laboratorios de computadores, el uno denominado IBM y el otro COMPAQ, denominados así por las marcas de computadoras existentes para la fecha se su creación. En la actualidad se ha instalado nuevos computadores tipo "clon".

La mayoría de estos equipos en ambos laboratorios tienen el sistema Operativo Windows 98 y en algunas de ellas se ha instalado Windows Milenium y Windows XP, en dos de ellas un Sistema de servidores Windows NT y Windows 2000 Server.

Estos laboratorios esta interconectados entre si por medio de concentradores (hubs) al Centro de cómputo, en el aspecto físico con cable trenzado UTP-5 y a nivel de software de comunicaciones con el protocolo TCP/IP, es decir que cada máquina tiene una dirección IP.

En el centro de cómputo básicamente existen dos servidores, el principal y que se conecta a Internet por medio de una línea dedicada es del modelo SPARK de Sum Microsystems, su sistema operativo es Red Hat Linux v6.0, y el segundo servidor de tipo secundario tipo "clon" posee Windows Nt 4.0 como sistema operativo.

Por último cabe mencionar que en la red existen otros puntos de conexión, uno a la Secretaría de la Escuela, otro al aula PCTV, otros al departamento de proyectos, dirección y subdirección de la Escuela, uno al auditórium y finalmente otros para interconexión con las demás escuelas. La disposición de la red se muestra en la figura 4.1

En este esquema se sugiere instalar un Servidor Linux Intel que actue como Host Bastión ante los servidores de la PUCESA y los laboratorios.

Toda información fue recabada por entrevistas con el administrador del Centro de Cómputo.

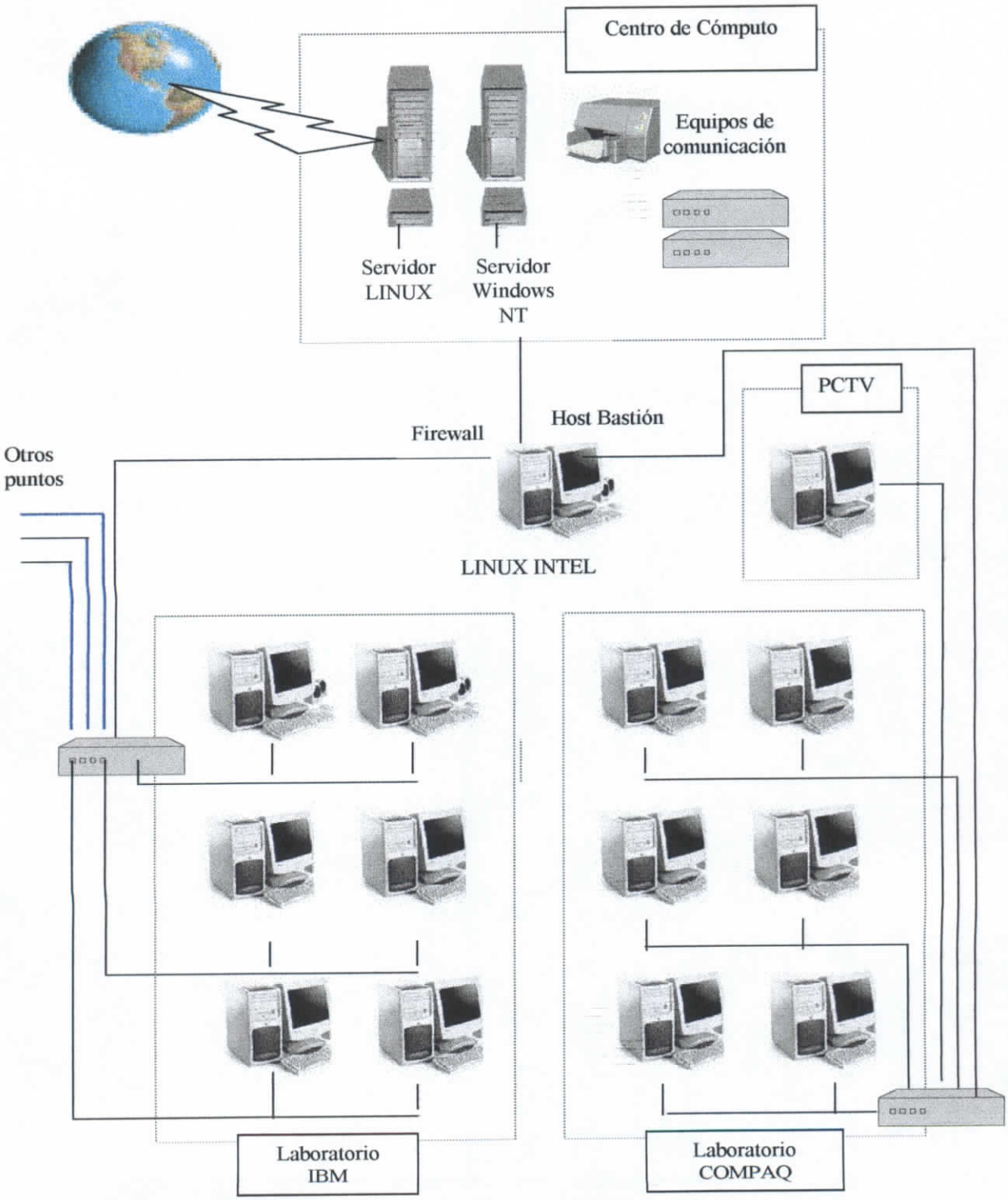


Figura 41. Estructura de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA

También se pudo recabar que la red funciona como una Intranet, en donde desde cualquier equipo interno se puede acceder al Portal de la PUCESA (www.pucesa.edu.ec) y al servicio de correo electrónico por medio de un WebMail. La parte más critica para nuestra investigación justamente está dada por la conexión permanente de la red a Internet. Conexión que es muy importante ya que la Universidad necesita mantener su portal en Internet, los estudiantes y profesores necesitan Internet para buscar información y además porque la PUCESA actúa como un pequeño proveedor de servicios Internet (ISP) para su personal.

Esta conexión permanente, que es necesaria es la puerta de entrada de intrusos si es que no se protege adecuadamente con algún mecanismo de seguridad. Por conversaciones con el administrador del Centro de Cómputo, ya se han registrado intrusiones en la red de la Escuela. De ahí nació la idea de investigar de algún mecanismo de seguridad y que pueda ayudar al administrador del Centro de Cómputo a proteger nuestra red.

Nuestra investigación en el campo de la seguridad informática ha permitido que conozcamos que existen básicamente dos tipos de intrusos, los internos y los externos. Nuestro objetivo entonces se enfocaba a proteger a la red de intrusos externos, ya que de éstos tipos de ataques han sucedido antes. De nuestra investigación se determinó que la configuración de un Firewall conocido también como muro de seguridad o muralla de fuego era una buena alternativa para este tipo de ataques y es así que se decidió optar por esta alternativa. Luego de esto era necesario saber que tipo de herramienta podíamos utilizar. Esto se describe a continuación.

4.4. SELECCIÓN DE LA HERRAMIENTA DE PROGRAMACIÓN

Nuestro objetivo era construir el Firewall en el servidor que estaba conectado a Internet permanentemente, y éste era el Servidor Linux, es por eso que se comenzó por analizar las herramientas que se podían utilizar en esta plataforma.

En los manuales de Linux encontramos que este sistema operativo traía consigo un módulo de filtraje de paquetes denominado IPCHAINS, al investigar y probar como funciona éste módulo nos percatamos que se necesitaba amplia experiencia para poder configurar un Firewall con éste paquete de filtraje de paquetes, primero porque no existe una interfaz gráfica para configurar IPCHAINS, es decir es un comando con un sinnúmero de parámetros que se los escribían manualmente en una sesión de comandos. Al realizar esto, en varias ocasiones se cometían errores de sintaxis y lógicos, además que cuando se consultaban las reglas existentes en el Firewall de Linux no se podía entender claramente lo que significaban.

Este estudio nos permitió partir de algo que ya estaba echo, poder mejorarlo e implementarlo en el Centro de Cómputo de la PUCESA, sin descartar que pueda ser también usado en cualquier estación en red con sistema operativo Linux. Nuestra aplicación entonces se basaría en construir una interfaz gráfica y amigable para la configuración de un Firewall, utilizando y aprovechando el comando IPCHAINS propio del Sistema Operativo Linux.

Se necesitaba entonces el Sistema Operativo Linux Red. Hat 6.0 con el módulo de IPCHAINS instalado. Luego de este escogitamiento había que buscar una herramienta de programación para la plataforma Linux. Para seleccionar esta herramienta se analizó que debía tener portabilidad, con el fin de poder construir un Firewall de acceso remoto con restricciones desde estaciones Windows para ayudarle al administrador ver las reglas creadas en el Firewall desde diferentes equipos. Tomando en cuenta estas necesidades se escogió el lenguaje de programación JAVA.

Java es un lenguaje de programación que se ejecuta sobre una "*máquina hipotética o virtual*" denominada ***Java Virtual Machine (JVM)***. En la que ***JVM*** interpreta un código neutro, convirtiéndolo a código particular de la CPU utilizada. Esta característica facilitaba a que podamos desarrollar la aplicación en Windows, el sistema operativo que hemos estudiado a lo largo de nuestra carrera. Este lenguaje sigue el lema: "*Write Once, Run Everywhere*".

Al programar en ***Java*** no se iba a partir desde cero, ya que cualquier aplicación que se desarrolle está soportada por un gran número de ***clases*** preexistentes. ***Java*** incorpora en el propio lenguaje muchos aspectos que en cualquier otro lenguaje son extensiones propiedad de empresas de software o fabricantes de ordenadores (threads, ejecución remota, componentes, seguridad, acceso a bases de datos, etc.). Por eso muchos expertos opinan que ***Java*** es el lenguaje ideal para aprender la informática moderna, porque incorpora todos estos conceptos de un modo estándar, mucho más sencillo y claro que con las citadas extensiones de otros lenguajes.

El principal objetivo del lenguaje ***Java*** es llegar a ser el "nexo universal" que conecte a los usuarios con la información, esté ésta situada en el ordenador local, en un servidor de ***Web***, en una base de datos o en cualquier otro lugar. En consecuencia se eligió este lenguaje porque es *"simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutra, portable, de altas prestaciones, multitarea y dinámico"*.

4.5 PROCESO DE DESARROLLO

En esta etapa se realizó un estudio de cada una de las actividades que iban a conformar nuestra aplicación, tales como: el flujo de la información y las responsabilidades; requerimientos, análisis y diseño y finalmente la implementación del nuevo sistema.

4.5.1 FLUJO DE INFORMACIÓN

El estudio del flujo de información, nos permite tener una idea clara de la creación, utilización y manejo de la información, que permiten realizar el proceso de configuración del Firewall. Para el diseño del flujo de información, se basó en los requerimientos del nuevo sistema, el mismo que se presenta en la figura 4.2, definiendo todos aquellos elementos que involucran los procesos de administración del Firewall.

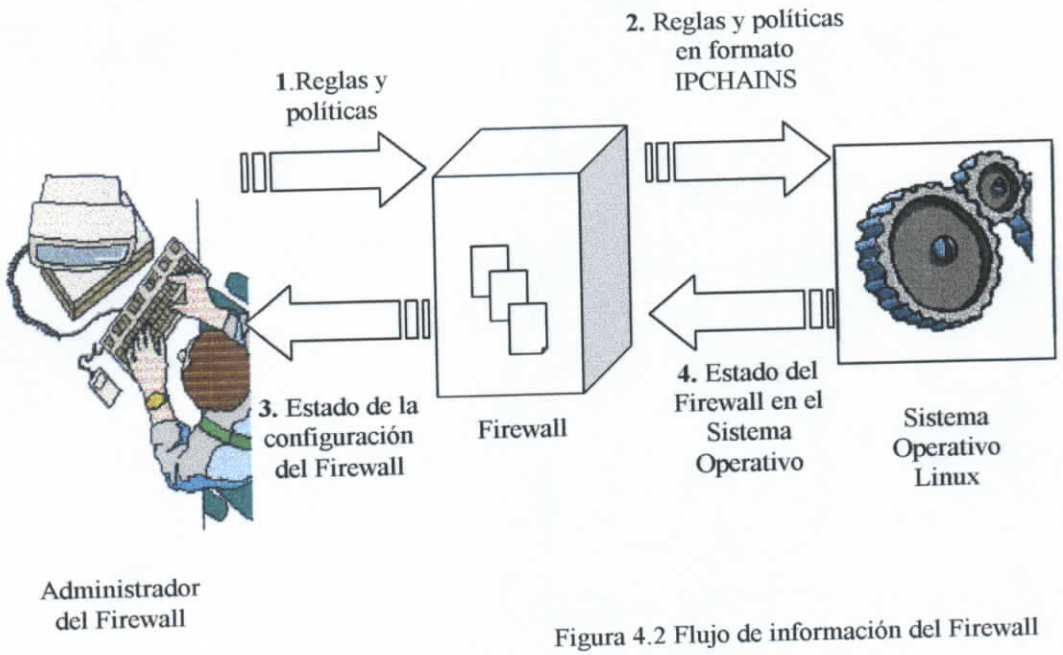


Figura 4.2 Flujo de información del Firewall

En la figura 4.2 en el punto (1.) El administrador del servidor interactua con el Firewall en el que administra un conjunto de reglas y políticas, y configuraciones que van a almacenarse en archivos de texto normal, excepto las cuentas y claves del Firewall que se almacenan en un archivo de texto encriptado.

En el paso (2.) el Firewall toma las configuraciones del Firewall suministradas por el administrador y las convierte en un formato que sea entendible por el sistema operativo, es decir las pone en formato IPCHAINS. Esta información va hacia el sistema operativo y ejecutan.

En el punto (3.) el administrador puede consultar en modo gráfico las reglas y políticas existentes en el Firewall, así como el estado del Firewall en el sistema operativo, información que va desde el punto (4.) y pasa al (3.)

4.5.2. REQUERIMIENTOS

Los requerimientos básicos que realizamos para el desarrollo del Firewall en nuestra estación de trabajo fueron:

- Instalación de una tarjeta de red
- Instalación del sistema Operativo Linux Red Hat 6.0
- Configuración de direcciones IP y dominios
- Instalación del compilador de Java JDK1.2.2 (Java Deployment Kit versión 1.2.2)
- Comprobación de la existencia del módulo IPCHAINS.

La explicación detallada de cada unos de estos requerimientos se explican más adelante en la sección de implementación.

4.5.3 ANÁLISIS Y DISEÑO DEL FIREWALL

En esta sección se explica el análisis de los procesos que ejecuta el Firewall. Debido a que se utilizó Java, un lenguaje puro en programación orientada a objetos se explicará el modelo de clases.

El Firewall está compuesto de 23 clases en su totalidad, de las cuales 14 de ellas son clases de usuario generadas por nosotros, las restantes son clases que se generan en algunas librerías que utilizamos. De estas 13 clase, una de ellas se la creó para acceso Remoto

multiplataforma. En la tabla 4.1 se muestran en conjunto las 14 clases que nos interesa explicar.

CONJUNTO DE CLASES DEL FIREWALL		
	CLASE	DESCRIPCION
1	Acercade	Utilizada para presentar información del sistema e información de autoría
2	Aplicar	Utilizada para implantar las reglas del Firewall en el sistema operativo
3	Ayuda	Muestra la ayuda del Firewall, cargando documentos HTML
4	Buscar	Busca reglas por medio de coincidencias
5	CambiarPol	Utilizada para transformar información del Firewall a IPCHAINS
6	Configurar	Configura parámetros del Firewall en archivos del sistema
7	ConsultarPR	Lista completa de políticas y reglas del Firewall
8	CrearPoliticas	Crea políticas del Firewall
9	CrearReglas	Crea reglas del Firewall
10	EliminarPol	Elimina políticas del Firewall
11	EliminarReg	Elimina reglas del Firewall
12	Estado	Muestra el estado del Firewall en el sistema operativo
13	Firewall	Clase principal que muestra la consola y realiza la llamada a las demás clases
14	Remoto	Consola cliente multiplataforma utilizada para acceso remoto

Tabla 4.1 Clases básicas del Firewall

En la figura 4.3 se muestra la organización de la arquitectura utilizada en el Firewall

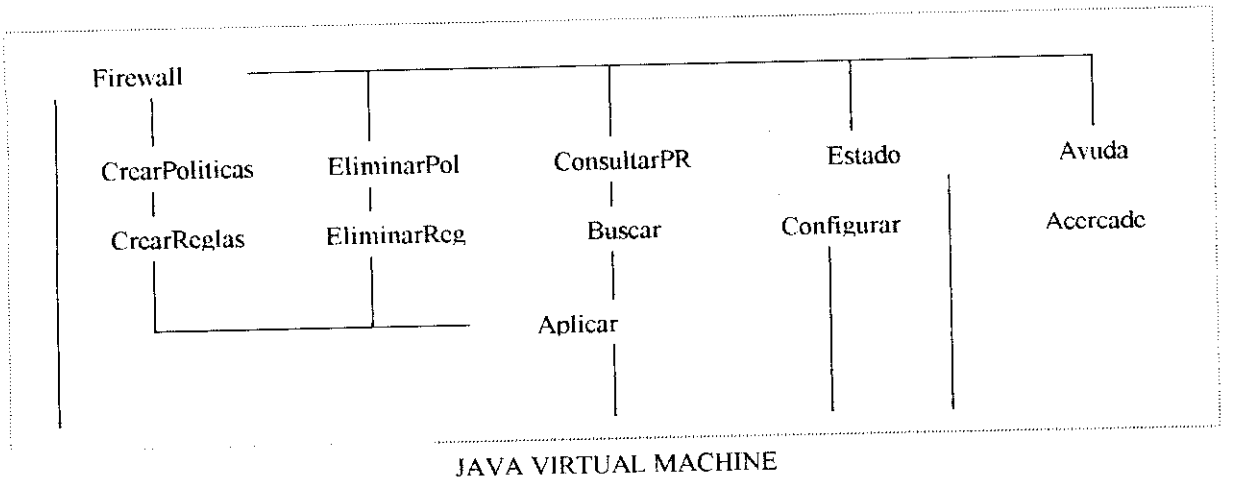


Tabla 4.3 diseño arquitectónico del Firewall

Una vez que se ha explicado en forma global el conjunto de clases que componen el Firewall, a continuación se va explicar cada una de las 14 clases, con sus respectivos métodos y atributos.

1. Acercade	
Constructor	Public Acercade()
Métodos	**Descripción**
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[ ])	Método principal que instancia a la clase

Tabla 4.2 Descripción clase Acercade

2. Aplicar	
Constructor	Public Aplicar() Public Ejecutar()
Métodos	**Descripción**
Public void init()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[ ])	Método principal que instancia a la clase
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.3 Descripción clase Aplicar

3. Ayuda	
Constructor	Public Ayuda(String docuhtml)
Métodos	**Descripción**
Private void CargaPagina (String url)	Carga página HTML en la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[ ])	Método principal que instancia a la clase

Tabla 4.4 Descripción clase Ayuda

4. Buscar	
Constructor	Public Buscar()
Métodos	**Descripción**
Public void init()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[ ])	Método principal que instancia a la clase
Public void recuperar()	Busca coincidencias en el archivo de reglas del Firewall
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.5 Descripción clase Ayuda

5. CambiarPol	
Constructor	Public CambiarPol()
Métodos	**Descripción**
Public void init()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[ ])	Método principal que instancia a la clase
Public void itemStateChanged(ItemEvent event)	Detecta el evento de cambio de politica
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.6 Descripción clase CambiarPol

6. Configurar	
Constructor	Public Configurar()
Métodos	**Descripción**
Public void init()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[ ])	Método principal que instancia a la clase
Public void recuperar()	Busca coincidencias en el archivo de reflas del Firewall
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.7 Descripción clase Configurar

7. ConsultarPR	
Constructor	Public ConsultarPR()
Métodos	**Descripción**
Public void init()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[])	Método principal que instancia a la clase
Public void recuperar()	Busca coincidencias en el archivo de reflas del Firewall
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.8 Descripción clase ConsultarPR

8. CrearPolitica	
Constructor	Public CrearPolitica()
Métodos	**Descripción**
Public void init()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[])	Método principal que instancia a la clase
Public void itemStateChanged(ItemEvent event)	Detecta de algún cambio en la lista de políticas
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.9 Descripción clase CrearPolitica

9. CrearRegla	
Constructor	Public CrearRegla
Métodos	**Descripción**
Public void InitialPositionSet()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[])	Método principal que instancia a la clase
Public void itemStateChanged(ItemEvent event)	Detecta de algún cambio en las listas
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.10 Descripción clase CrearRegla

10. EliminarPol	
Constructor	Public EliminarPol()
Métodos	**Descripción**
Public void init()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[])	Método principal que instancia a la clase
Public void recuperar()	Extrae reglas del Firewall desde el archivo de reglas
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.11 Descripción clase EliminarPol

11. EliminarReg	
Constructor	Public EliminarReg()
Métodos	**Descripción**
Public void init()	Inicializa variables y objetos de la interfaz
Public void actionPerformed(ActionEvent e)	Método para eventos ocurridos en la interfaz
Public static void main(String args[])	Método principal que instancia a la clase
Public void recuperar()	Extrae reglas del Firewall desde el archivo de reglas
Public String LeeVariable(String variable)	Lee variables del archivo de configuraciones: config.cfg

Tabla 4.12 Descripción clase EliminarReg

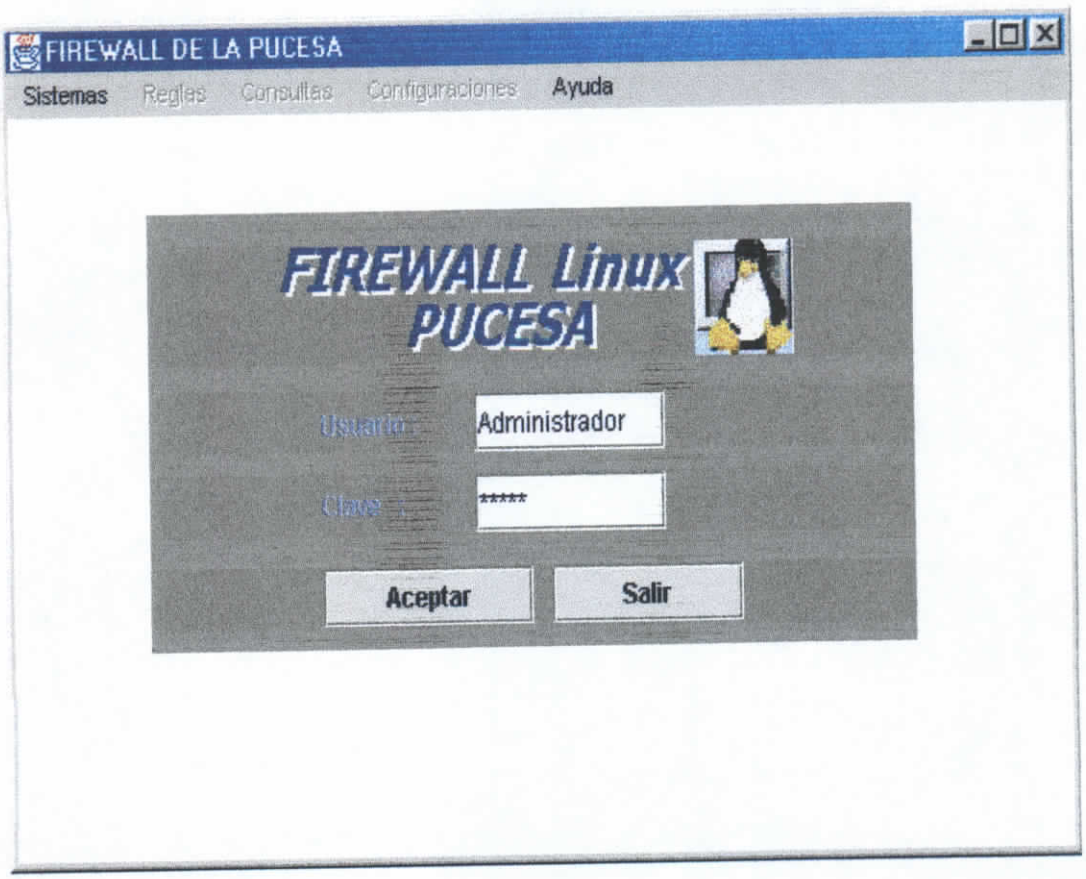


Figura 4.4 Pantalla principal del Firewall

En la figura 4.4 se muestra la pantalla inicial del Firewall, cuando en el sistema se ejecuta la clase Firewall() de la siguiente manera "java Firewall". Esta pantalla antes de nada solicita se ingrese un logín y un contraseña para poder acceder a utilizar los demás procesos del Firewall. Esta pantalla es la encargada de llamar a las demás.

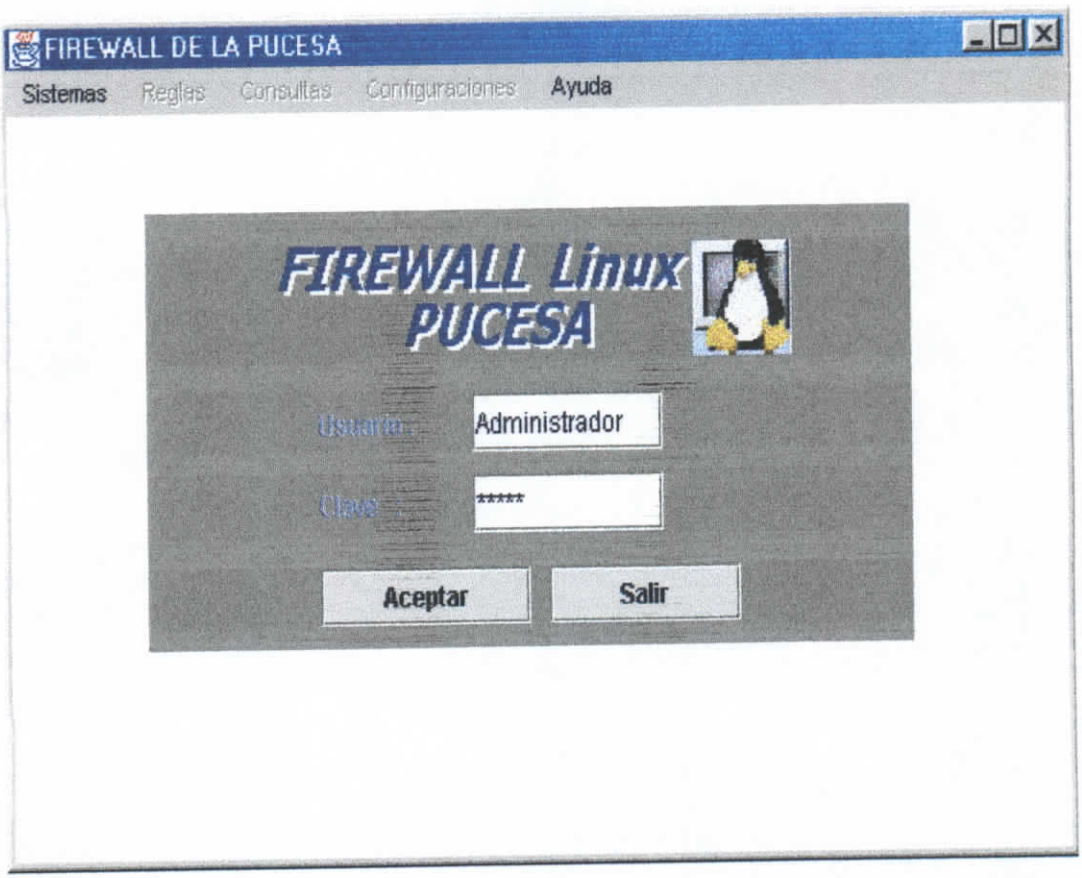


Figura 4.4 Pantalla principal del Firewall

En la figura 4.4 se muestra la pantalla inicial del Firewall, cuando en el sistema se ejecuta la clase Firewall() de la siguiente manera "java Firewall". Esta pantalla antes de nada solicita se ingrese un logín y un contraseña para poder acceder a utilizar los demás procesos del Firewall. Esta pantalla es la encargada de llamar a las demás.

La figura 4.5 se muestra la pantalla de la clave del Firewall, En esta pantalla se realiza el cambio de clave, digitando la clave anterior, posteriormente se digita la nueva clave y finalmente hay que confirmar y aplicar.

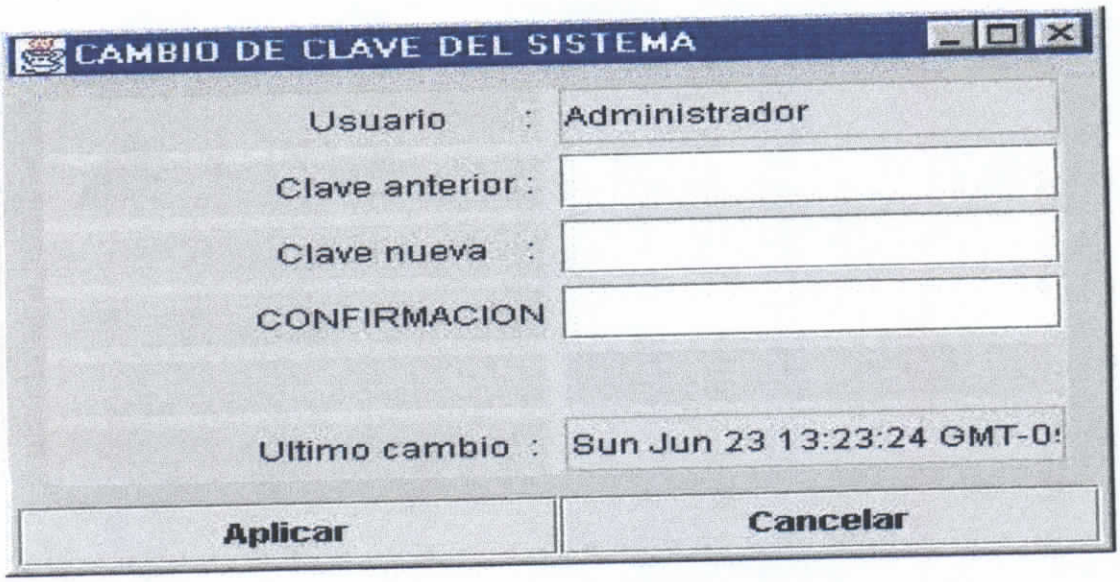


Figura 4.5 Cambio de Clave del Sistema

En la figura 4.6 se muestra la configuración del sistema donde se encuentra ubicado el archivo reglas.bat

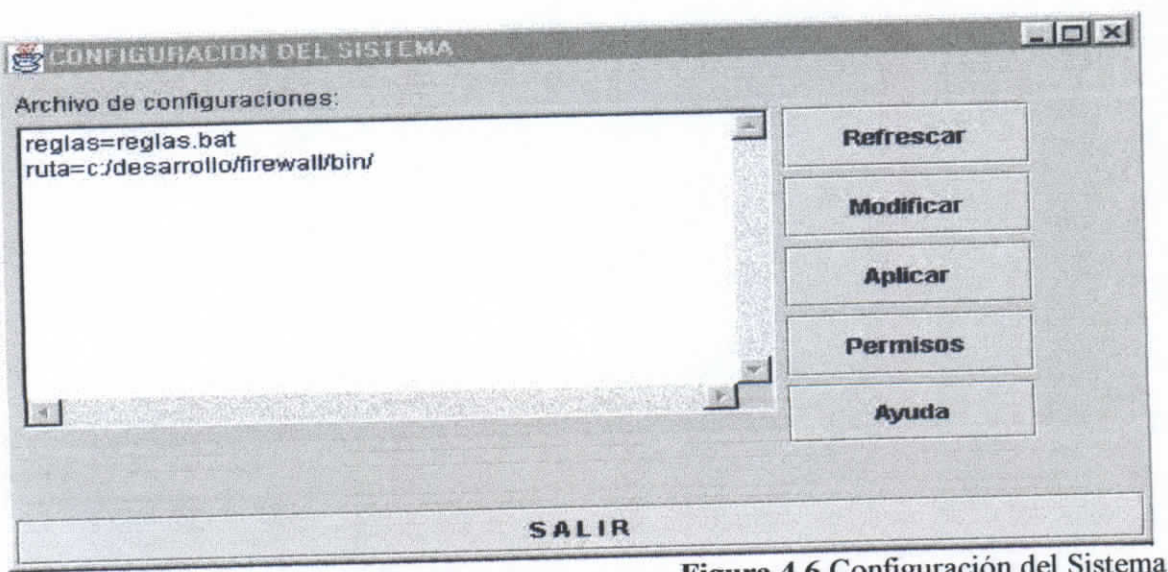


Figura 4.6 Configuración del Sistema

La pantalla 4.7 muestra el menú de opciones que posee el Firewall como: Sistema, Reglas, Consultas, Configuraciones, Ayuda.

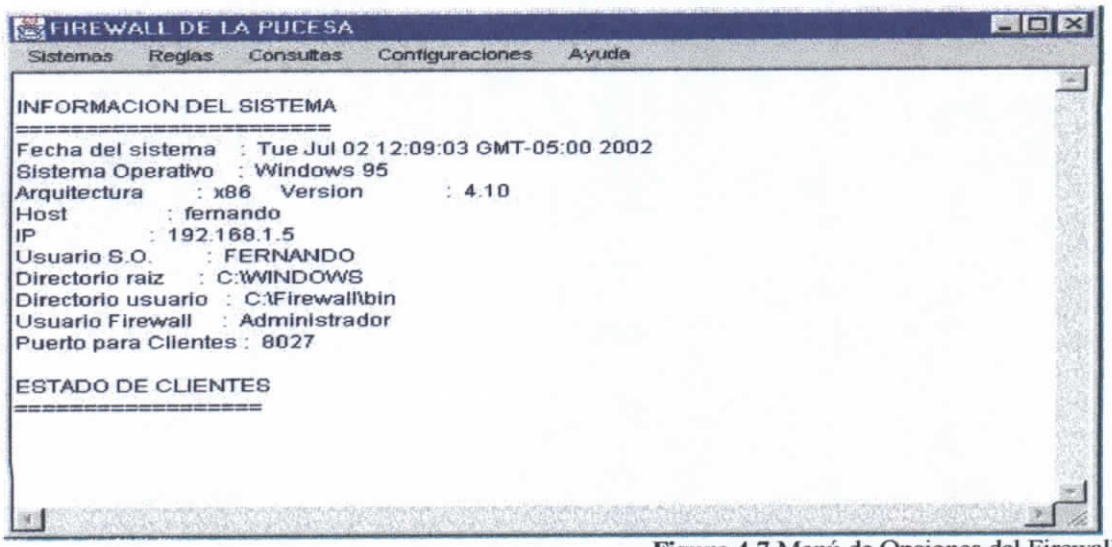


Figura 4.7 Menú de Opciones del Firewall

En la pantalla 4.8 se presenta el estado del Firewall es decir todas las reglas y políticas que se están ejecutando en el programa.

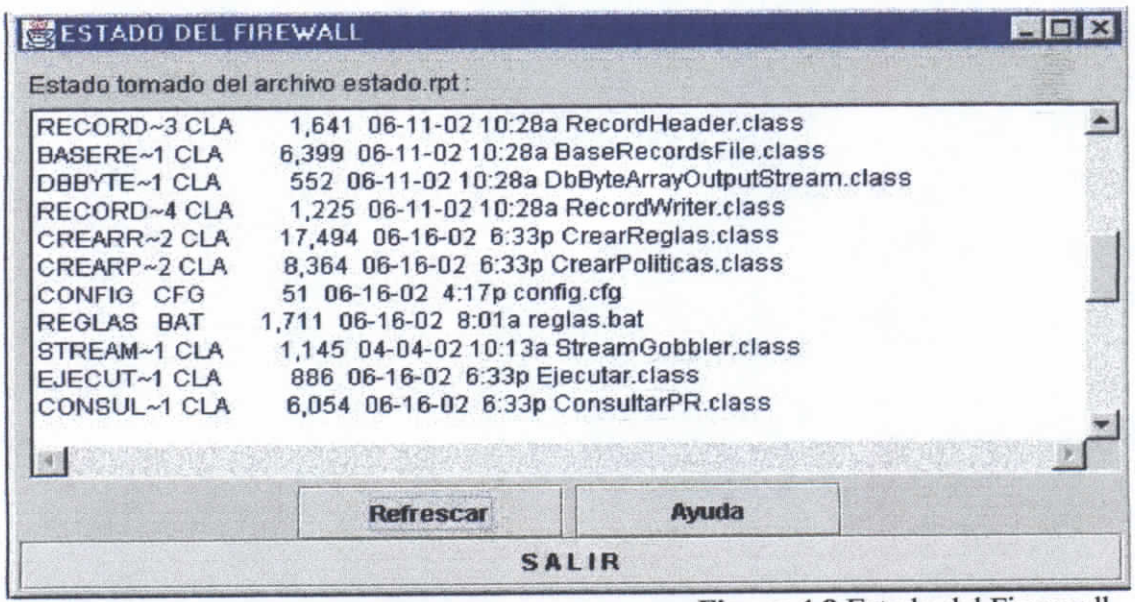


Figura 4.8 Estado del Fiurewall

En la figura 4.9 del Firewall se presenta en forma de consulta general todas las reglas y políticas que se han creado para obtener la máxima seguridad.

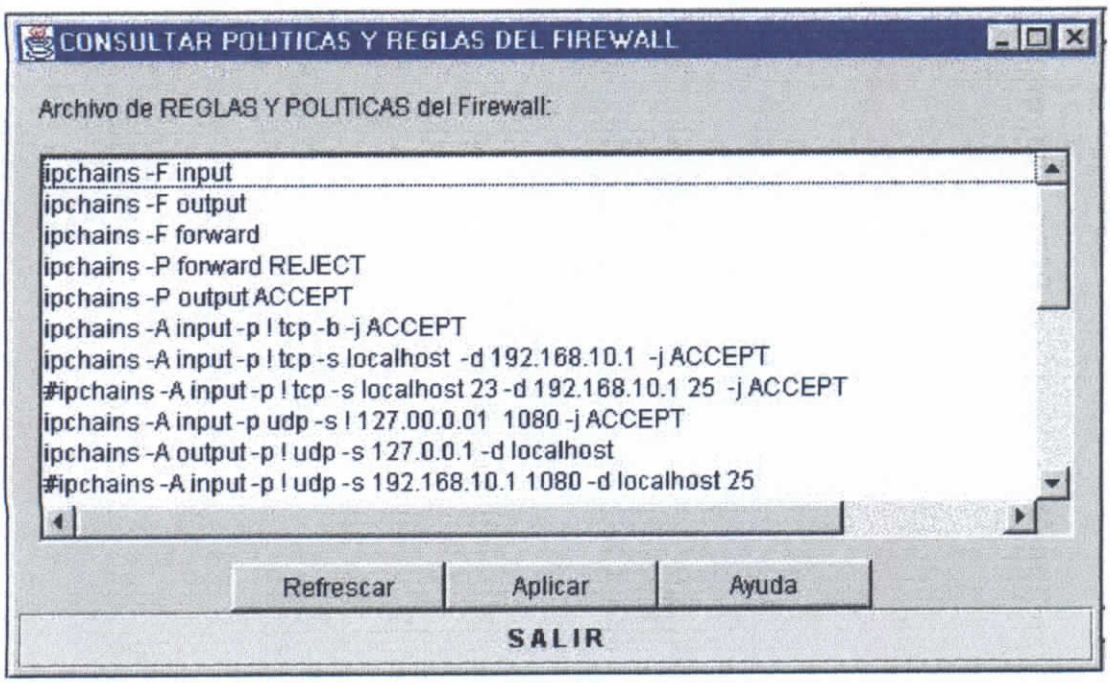


Figura 4.9 Consultas de Políticas y Reglas del Firewall

La pantalla 4.10 muestra la eliminación de políticas.

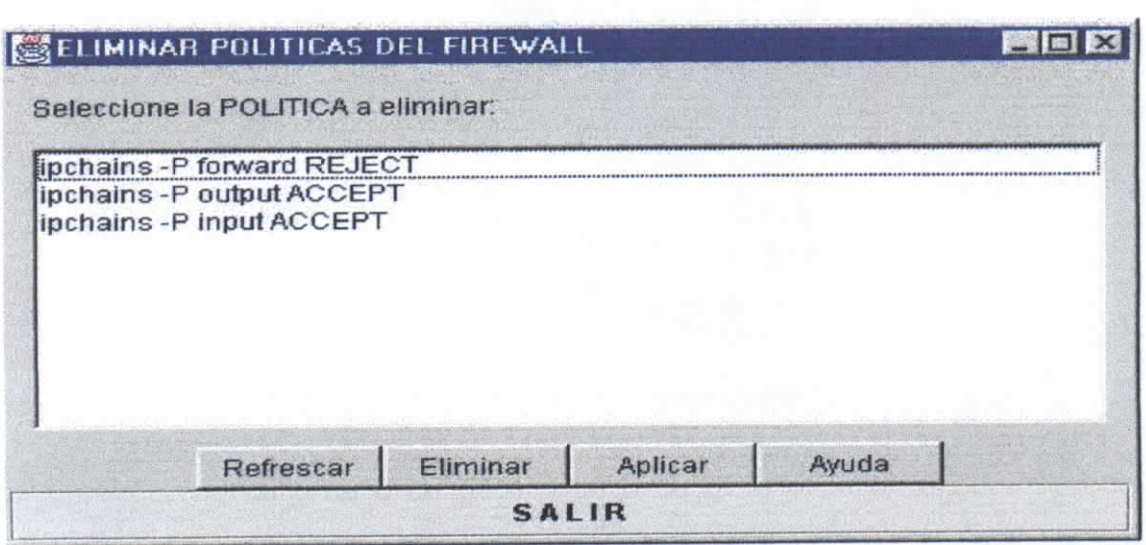


Figura 4.10 Eliminación de Políticas del Firewall

En la pantalla 4.11 se muestra la eliminación de reglas existentes en el Firewall.

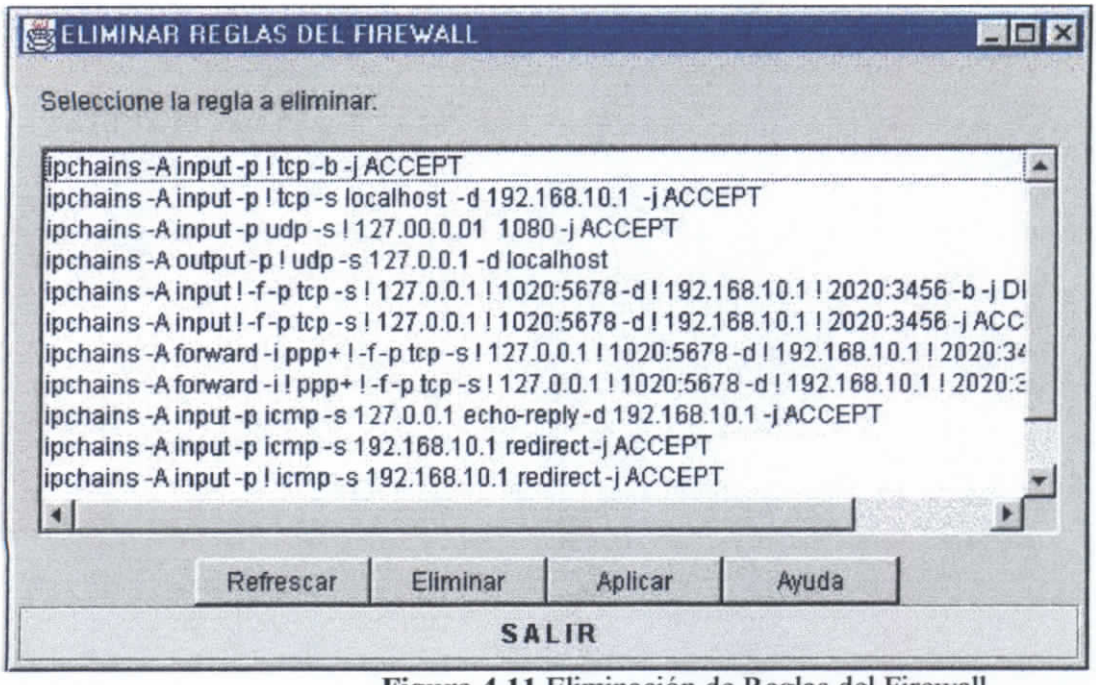


Figura 4.11 Eliminación de Reglas del Firewall

La siguiente pantalla 4.12 muestra la creación de políticas que será guardada en el archivo reglas.bat

Figura 4.12 Creación de Políticas

En la pantalla 4.13 se realiza la creación de reglas que servirá para la protección de intrusos, dichas reglas serán guardadas en el archivo antes mencionado.

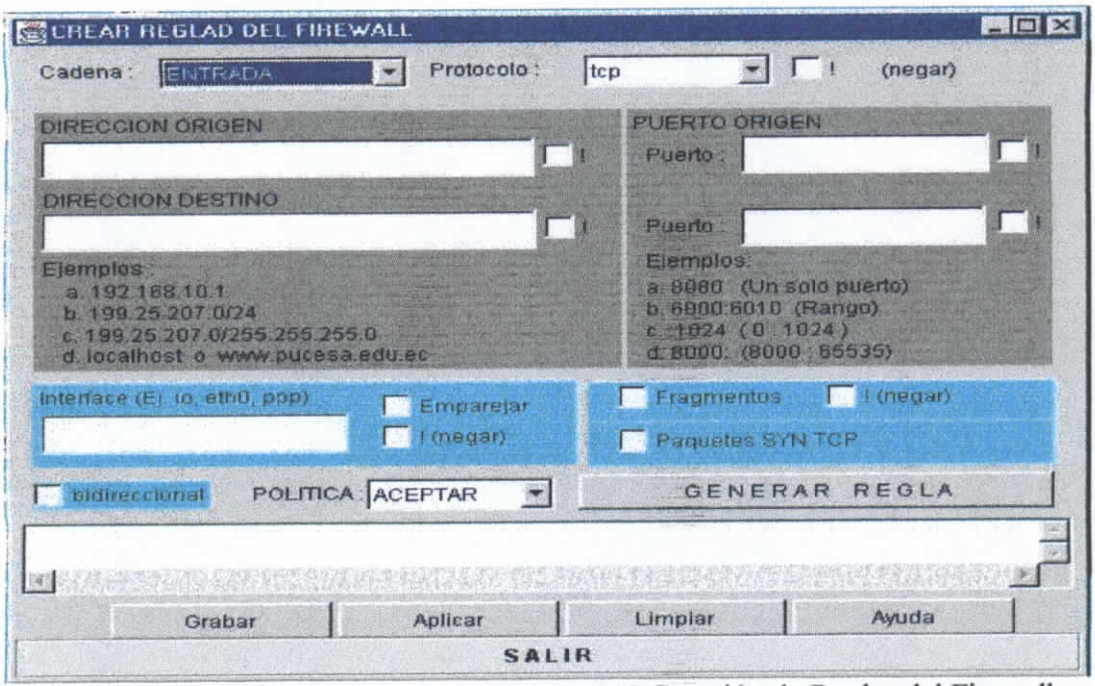


Figura 4.13 Creación de Reglas del Firewall

En la pantalla 4.14 se realiza una consulta de búsqueda de reglas, políticas mediante el ingreso de parámetros.

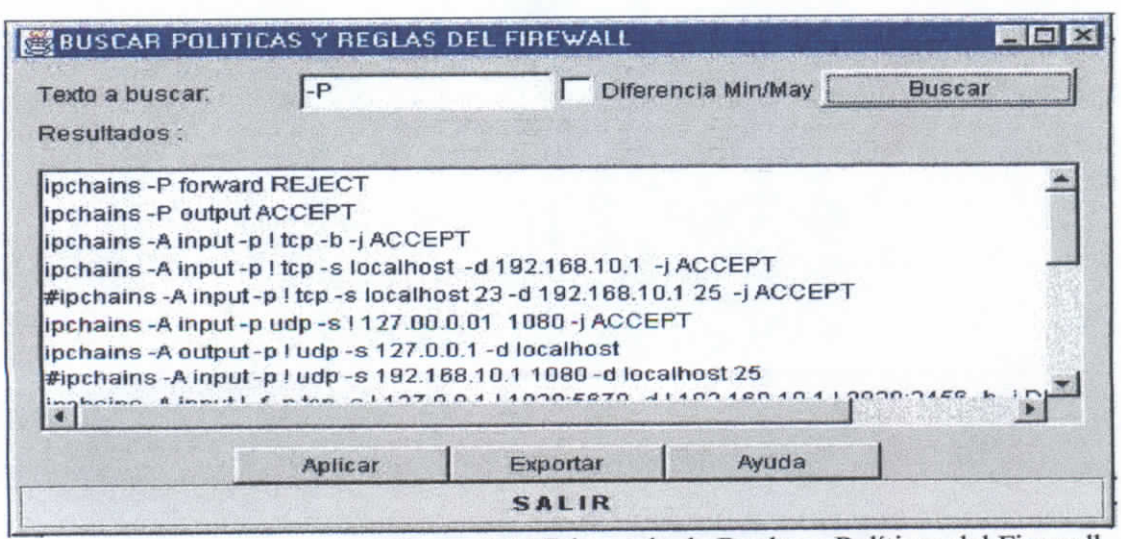


Figura 4.14 Búsqueda de Reglas y Políticas del Firewall

La pantalla 4.15 llamada acerca del sistema presenta los nombres de los creadores del Firewall, director de tesis y los revisores, además incluye el nombre de la unidad y también la versión del programa.

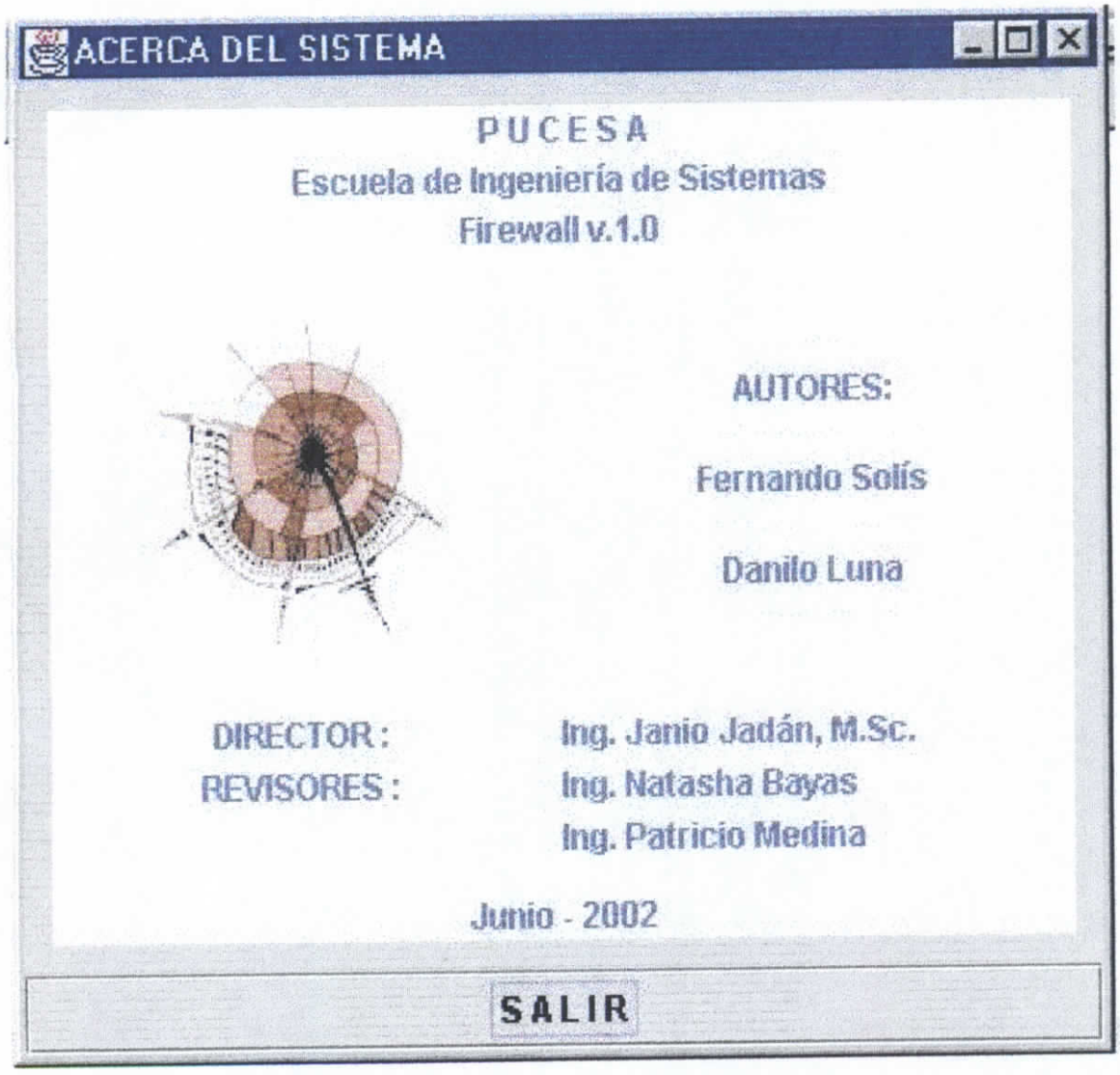


Figura 4.15 Pantalla Acerca del Sistema

4.6 IMPLEMENTACIÓN

Una vez que se ha terminado el análisis y desarrollo del Firewall se procedió a implementarlo en el sistema de desarrollo, para lo cual procederemos a explicar desde la instalación de las herramientas que se utilizaron.

A continuación se explicarán las instalaciones de Red Hat Linux V6.0 Java Deploymet Kit V1.2.2, IPCHAINS. Y el desarrollo para la PUCESA.

4.6.1 LINUX RED HAT 6.0

Red Hat Linux es un sistema operativo potente, bien acabado. Si no son expertos en Linux, va a necesitar documentación adicional para sacar el máximo partido a su sistema Red Hat Linux.

Aunque instalar Red Hat Linux es un proceso sencillo, tomarse su tiempo antes de comenzar la instalación puede hacer que las cosas salgan mucho mejor. En este capítulo, se discutirán los pasos a tomar antes de comenzar la instalación.

Si actualmente utiliza una versión 2.0 (o superior) de un sistema Red Hat Linux, puede actualizarlo. El proceso de actualización empieza de forma idéntica al proceso de instalación; se le pedirá que elija instalar o actualizar tras arrancar el programa de instalación y contestar unas cuantas preguntas.

Hay cinco cosas que se debería hacer antes de instalar Red Hat Linux:

1. Comprobar que cuenta con documentación suficiente para poder usar su sistema Red Hat Linux después de la instalación.
2. Comprobar que tiene acceso a los componentes de Red Hat Linux requeridos para la instalación.
3. Conocer la configuración del hardware de su ordenador y la información de su configuración de red.
4. Basarse en las dos primeras tareas, el método que utilizará para instalar Red Hat Linux.
5. Ubicar dónde residirán los discos duros destinados a Red Hat Linux.

4.6.1.1. INTRODUCCIÓN DE LA INSTALACIÓN

Ahora es el momento de comenzar a instalar Red Hat Linux. Para iniciar la instalación, es necesario en primer lugar arrancar el programa de instalación:

4.6.1.2. ARRANQUE DEL PROCESO DE INSTALACIÓN

Para comenzar a instalar Red Hat Linux, inserte el disquete de arranque en la primera unidad de disquetes de su ordenador y reiniciar el sistema (o arranque desde el CD-ROM

de Red Hat Linux,). Puede que necesite modificar la configuración de la BIOS para permitirle arrancar desde el disquete o el CD-ROM.

Después de un corto espacio de tiempo, debería aparecer una pantalla que contiene el prompt boot:. La pantalla contiene información acerca de varias opciones de arranque.

Cada una de ellas también tiene una o más pantallas de ayuda asociadas. Para acceder a una determinada de ellas, se pulsa la tecla de función correspondiente, tal y como se indica en la línea de la parte inferior de la pantalla. Debemos tener en cuenta dos cosas principales:

- La pantalla inicial arrancará automáticamente el programa de instalación.
- Normalmente, sólo se necesitará pulsar enter para arrancar. Es necesario observar los mensajes de arranque para ver si el núcleo Linux detecta el hardware. Si no lo hace correctamente, puede que se necesite volver a ejecutar la instalación en modo ``experto".

El modo experto deshabilita la mayor parte de los análisis del hardware, y le da la opción de introducir opciones para los controladores que se cargan durante la instalación. Para entrar en el modo experto se debe usar el comando de arranque.

Los mensajes iniciales de arranque no contendrán ninguna referencia a tarjetas SCSI o de red. Estos dispositivos están soportados por módulos que son cargados durante el proceso de instalación.

Después de introducir las opciones, se debe pulsar Intro para arrancar usando todas ellas.

Se necesitan especificar opciones de arranque para identificar su hardware, es necesario tomar nota de ellas.

[*Intel Systems:* Instalación sin usar un disquete de arranque El CD-ROM de Red Hat Linux/Intel también puede arrancarse en ordenadores modernos que soporten arranque desde CD-ROMs. No todos los ordenadores soportan esta característica, de modo que si es posible no puede hacerlo, hay otra manera de comenzar la instalación sin utilizar un disquete de arranque.El siguiente método es específico para ordenadores basados en procesadores Intel.]

Si tenemos instalado MS-DOS en su sistema, puede arrancar el sistema directamente desde el CD sin utilizar ningún disquete.

Si el computador no puede arrancar directamente desde CD-ROM (no puede usar el autoarranque basado en DOS), se tendrá que utilizar un disquete de arranque para comenzar el proceso.

Después de arrancar, el programa de instalación comenzará mostrando un mensaje de bienvenida. Se debe pulsar enter para comenzar la instalación.

Después del mensaje de bienvenida, el programa de instalación pedirá que se seleccione el idioma a utilizar durante el proceso de instalación, tal como se muestra en la figura 4.16

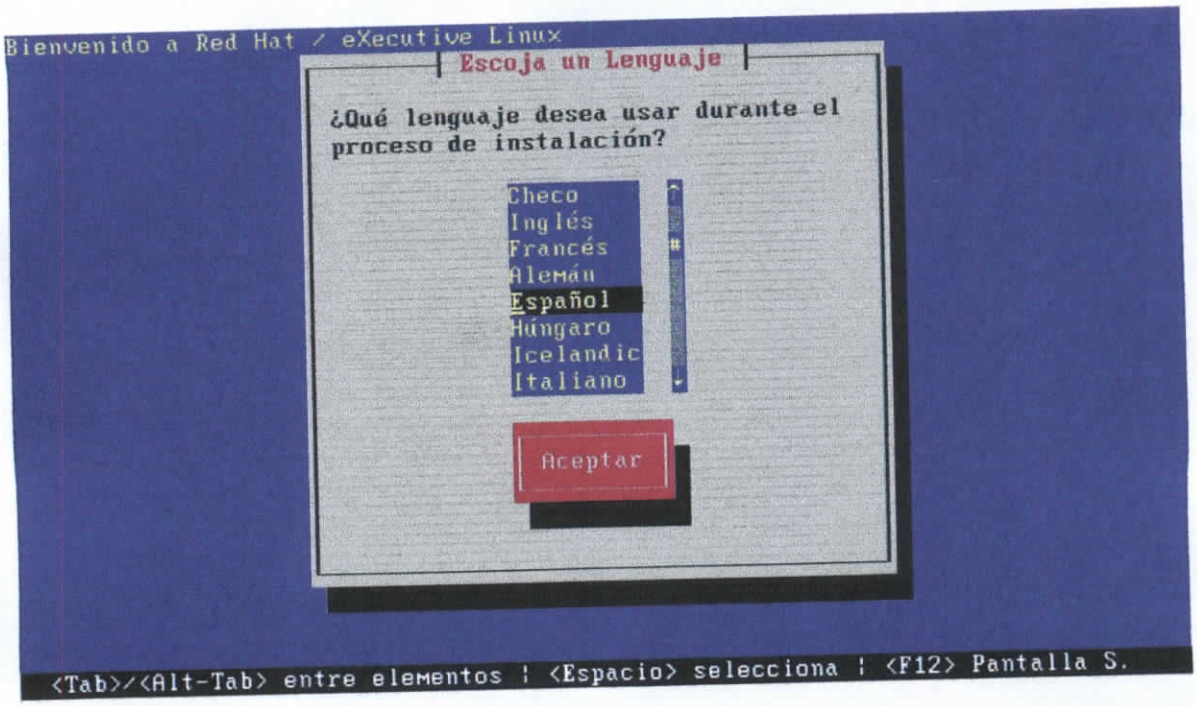


Figura 4.16. Selección del Idioma

A continuación, el programa de instalación le da la oportunidad de seleccionar un tipo de teclado. Se puede navegar en ese cuadro de diálogo del mismo modo que se hizo en el de selección de idioma, tal como se muestra en la figura 4.17.

Después de seleccionar el tipo apropiado de teclado, se debe pulsar enter, el tipo de teclado que se seleccione se cargará automáticamente tanto para el resto del proceso de instalación como para cada vez que arranque su sistema Red Hat Linux.

Se desea cambiar el tipo de teclado después de que haya instalado el sistema Red Hat Linux, se puede utilizar el comando `/usr/sbin/kbdconfig` o se puede teclear `setup` en el prompt de `root`.

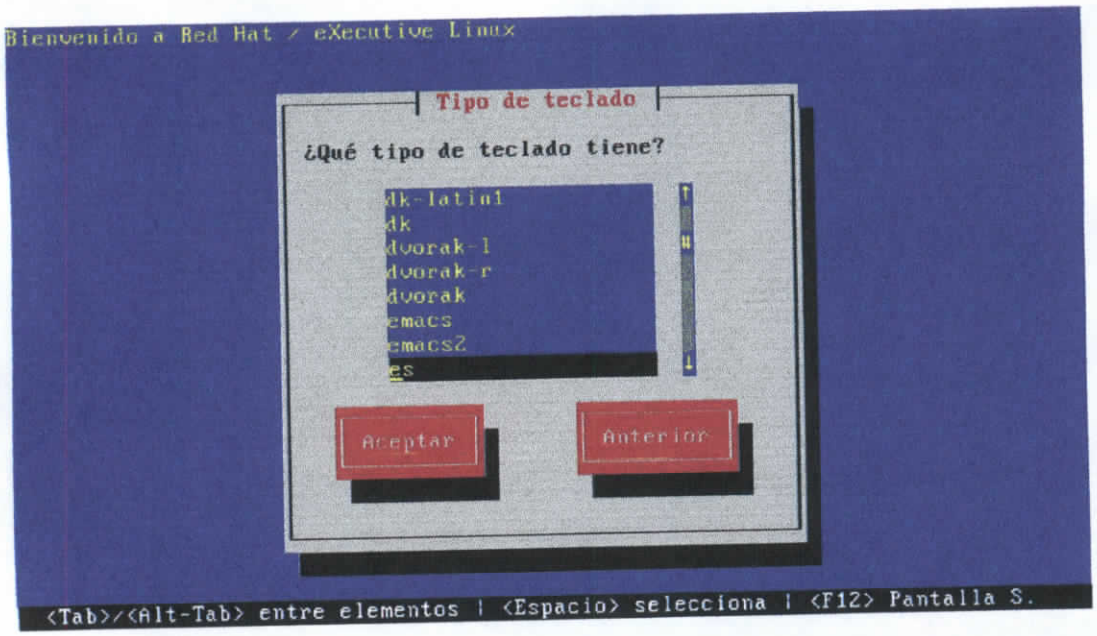


Figura 4.17. Selección del Teclado

A continuación, se le preguntará qué tipo de instalación desea usar. Se puede instalar Red Hat Linux a través de cinco métodos básicos.

- CD-ROM
- DISCO DURO
- IMAGEN NFS
- FTP
- HTTP

4.6.1.3. INSTALACIÓN DESDE UN CD-ROM

Si se va a instalar Red Hat Linux desde CD-ROM, se debe elijir ``CD-ROM", y seleccionar Aceptar [Ok]. El programa de instalación pedirá entonces que se inserte el CD de Red Hat Linux en el lector de CD-ROM. Cuando se haya hecho, se selecciona Aceptar [Ok], y presionar enter tal como se muestra en la figura 4.18.

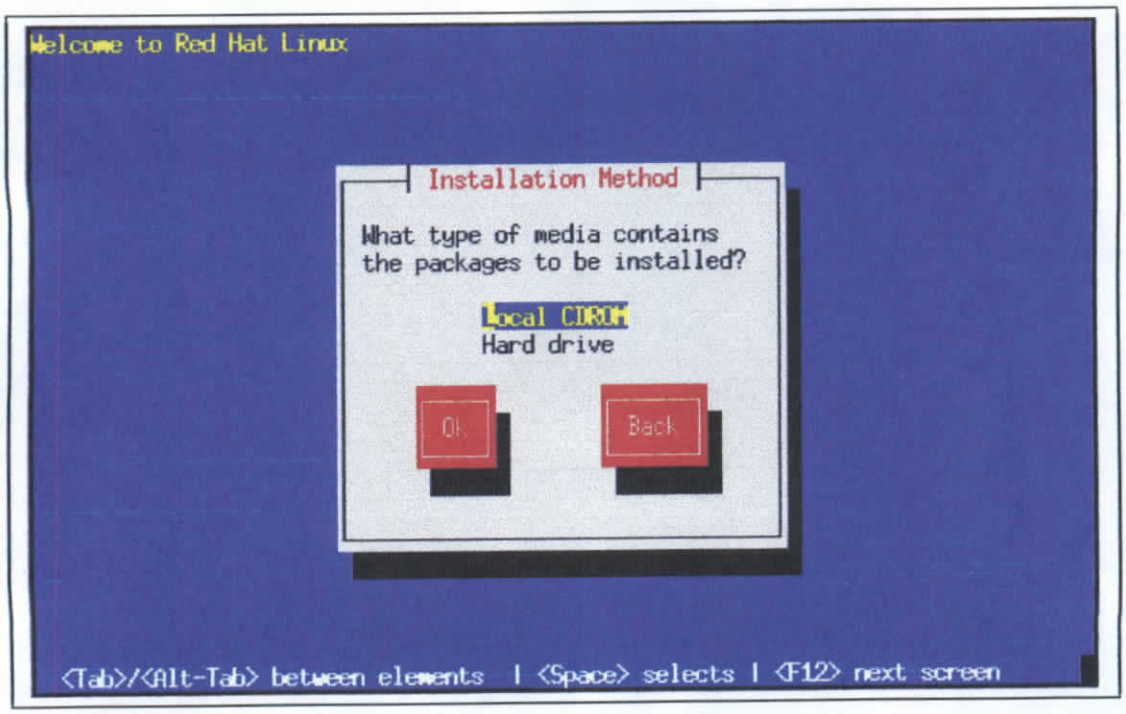


Figura 4.18. Instalación desde un CD-ROM

Después de elegir un método de instalación, el programa de instalación preguntará si se desea instalar [*install*] o actualizar [*upgrade*], tal como se muestra en la figura 4.19.

Figura 4.19. Actualizar o Instalar

Normalmente se instala Red Hat Linux en una partición o conjunto de particiones de disco limpias, o sobre otra instalación de Linux.

Al instalar Red Hat Linux sobre otra instalación de Linux (incluyendo Red Hat Linux) *no* se conserva ninguna información (ficheros o datos) de anteriores instalaciones.

El proceso de instalación para Red Hat Linux 6.0 incluye la posibilidad de actualizar desde versiones anteriores de Red Hat Linux (versión 2.0 y posteriores) que estén basadas en tecnología RPM. Al actualizar el sistema se instala el núcleo modular 2.2.*x* además de versiones actuales de los paquetes que están actualmente instalados en la máquina.

El proceso de actualización preserva los ficheros de configuración existentes, renombrándolos utilizando una extensión .rpmsave y deja un registro diciendo qué acciones realizó en /tmp/upgrade.log.

Al evolucionar el software, los formatos de los ficheros de configuración pueden cambiar, así que debería comparar cuidadosamente sus ficheros de configuración originales con los nuevos ficheros, antes de integrar los cambios. Si se desea actualizar el sistema Red Hat Linux, se debe elijir Actualizar [Upgrade].

4.6.1.4. CLASES DE INSTALACIÓN

Red Hat Linux incluye tres clases distintas, o tipos de instalaciones, estas son:

- Estación de trabajo (Workstation)
- Servidor (Server)
- Personalizada (Custom)

Estas clases le dan la opción de simplificar el proceso de instalación (con pérdida de algo de flexibilidad en la configuración), o de mantener toda la flexibilidad con un proceso de instalación ligeramente más complejo.

4.6.1.5. INSTALACIÓN DE TIPO WORKSTATION

Una instalación de tipo workstation es la más apropiada para un usuario nuevo en el mundo de Linux, con el fin de que pueda probarlo.

Una instalación de la clase workstation borra todas las particiones de tipo Linux en todos los discos duros instalados (y usa todo el espacio libre no particionado).

4.6.1.6. INSTALACIÓN DE TIPO SERVIDOR

Una instalación de tipo servidor es la más adecuada si se desea que el sistema funcione como un servidor basado en Linux, y no desea personalizar en gran medida la configuración del sistema.

Una instalación de tipo servidor borrará *toda* partición existente en todos los discos duros instalados, así que se debe elijir este tipo de instalación únicamente si se está seguro de no tener nada que se quiera conservar.

4.6.1.7. INSTALACIÓN PERSONALIZADA

La instalación personalizada pone todo el énfasis en la flexibilidad. Durante una instalación personalizada, depende cómo se particione el disco duro. Tiene control absoluto sobre los paquetes que se instalarán en el sistema. También se puede determinar si usará LILO para arrancar el sistema la pantalla del tipo de instalación se muestra en la figura 4.20.

Si se realiza una instalación completa, el programa de instalación le pedirá que elija un tipo de instalación.

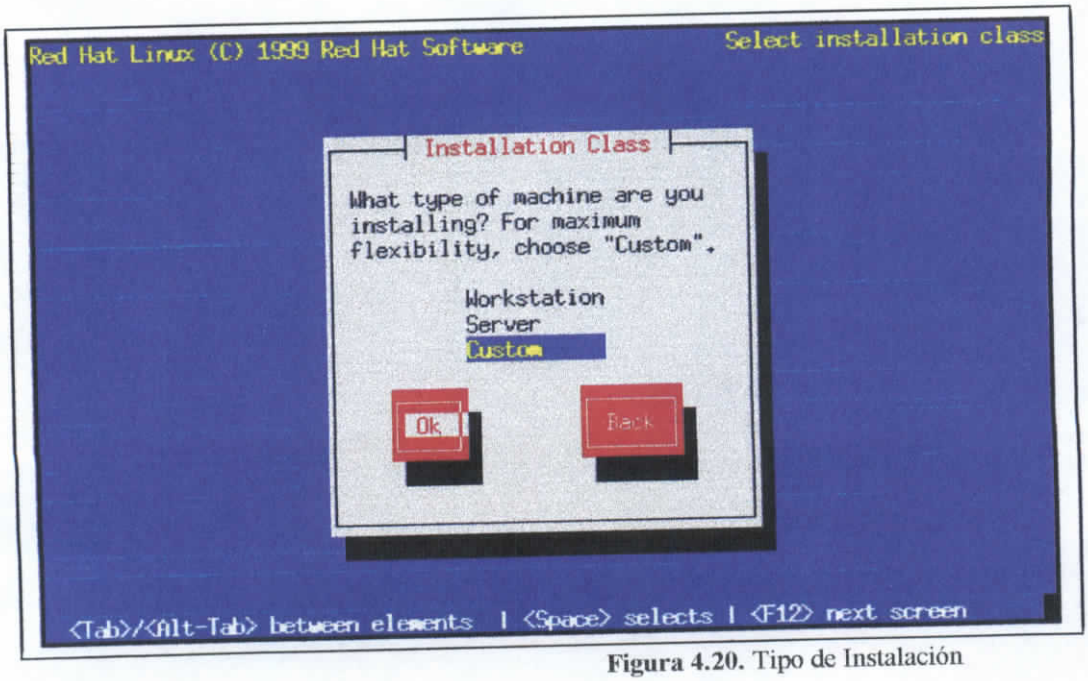


Figura 4.20. Tipo de Instalación

4.6.1.8. INSTALACIÓN DE RED HAT LINUX PASO A PASO

A continuación se explica paso a paso el proceso de instalación de Red Hat Linux 6.0. Cubriremos las siguientes áreas:

- Interfaz de usuario del programa de instalación;
- Inicio del programa de instalación;
- Selección de un método de instalación

4.6.1.9. INTERFAZ DE USUARIO DEL PROGRAMA DE INSTALACIÓN

El programa de instalación de Red Hat Linux utiliza una interfaz de pantalla completa en modo texto que incluye la mayoría de los "widgets" que se encuentran normalmente en los interfaces gráficos de usuario. Pueden parecer ligeramente diferentes a sus correspondientes gráficos. A continuación incluimos una lista de los "widgets" más importantes:

- Ventana
- Entrada de texto
- Casilla de verificación
- Zona de texto
- Barra de desplazamiento
- Botón
- Cursor

La figura 4.21 y 4.22 muestran la pantalla con los "Widgets"

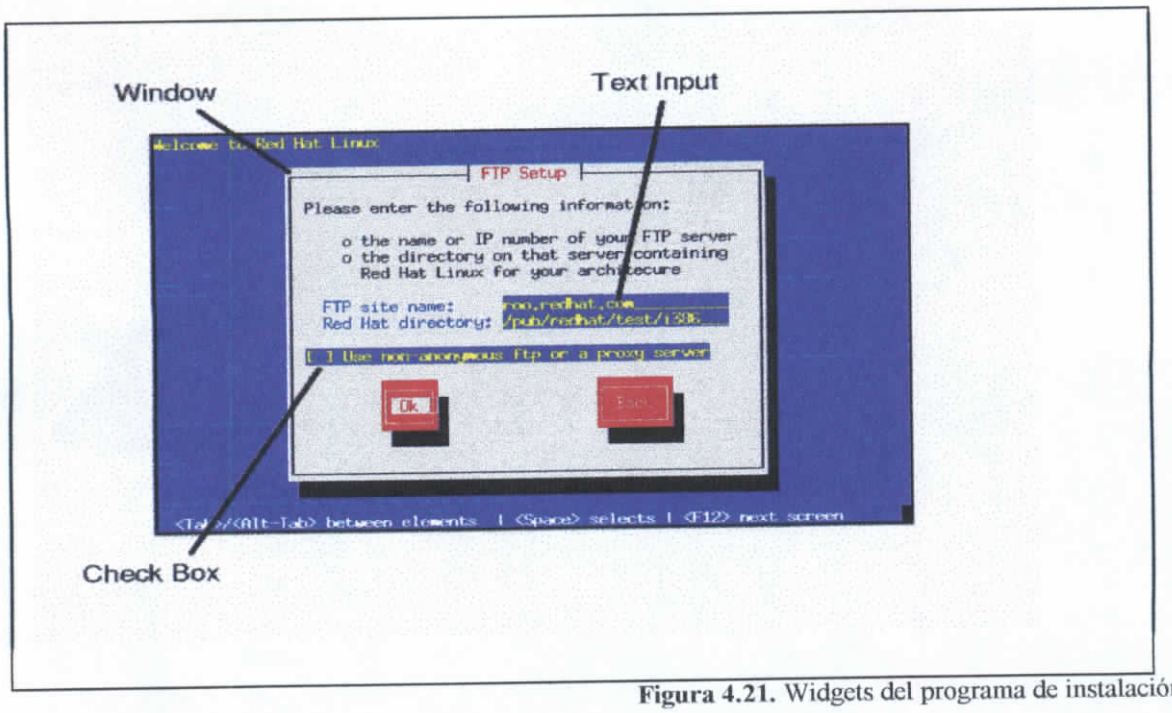


Figura 4.21. Widgets del programa de instalación

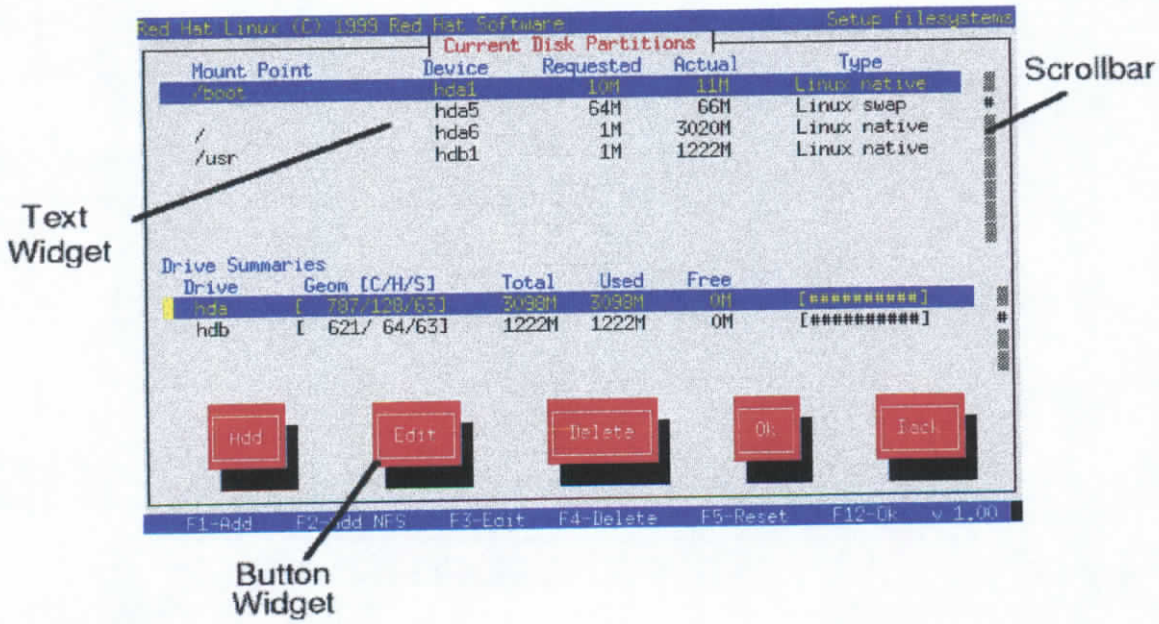


Figura 4.22. widgets del programa de instalación

A continuación, el sistema intentará localizar un ratón. Algunos ratones son detectados automáticamente; en cuyo caso, se visualiza una ventana de diálogo mostrando el puerto en el que se encontró el ratón. El sistema solicitará además que introduzca la información adicional, tal como, si tiene un ratón de dos-botones, y si le gustaría que se emulase a uno de tres-botones, como se muestra en la figura 4.23.

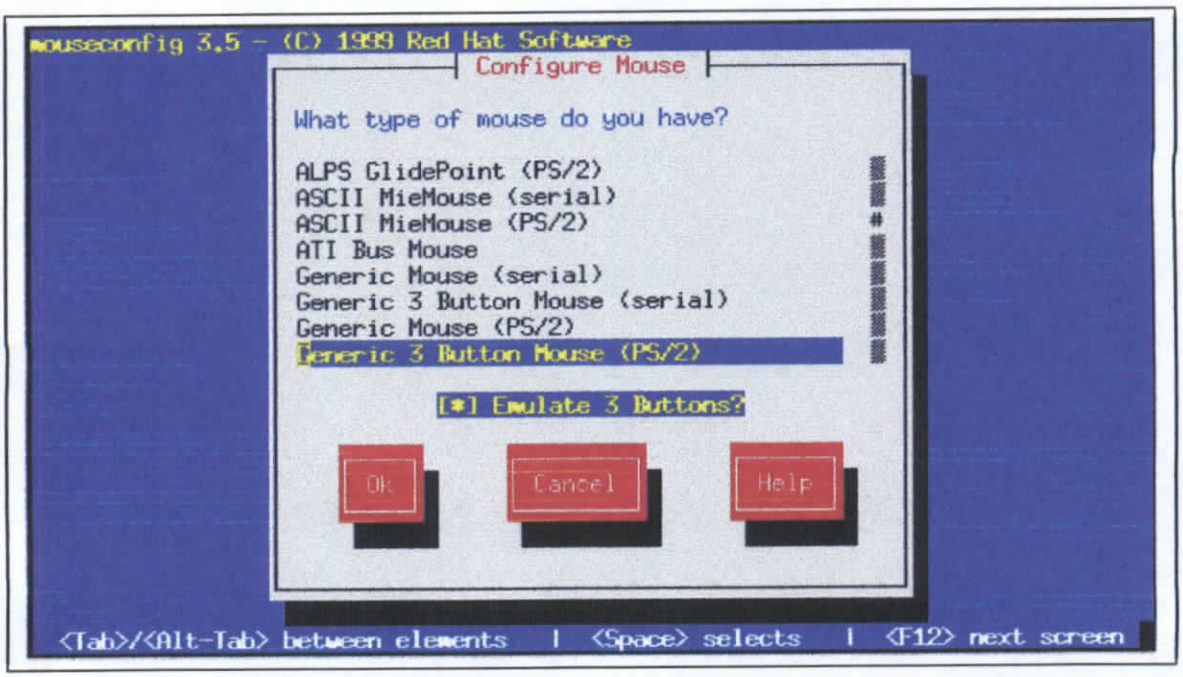


Figura 4.23. Configuración del ratón

Si se encuentra en la lista una opción *exacta* de el ratón se debe escoger esa emtrada.

La opción Emular 3 Botones le permite usar un ratón de dos-botones como si fuera un ratón de tres-botones. En general, es más fácil utilizar el sistema de ventanas-X si tiene un ratón de tres-botones. Si se selecciona esta opción, se podrá emular un tercer botón además el instalador solicita el puerto de conexión del ratón, tal como se muestra en la figura 4.24.

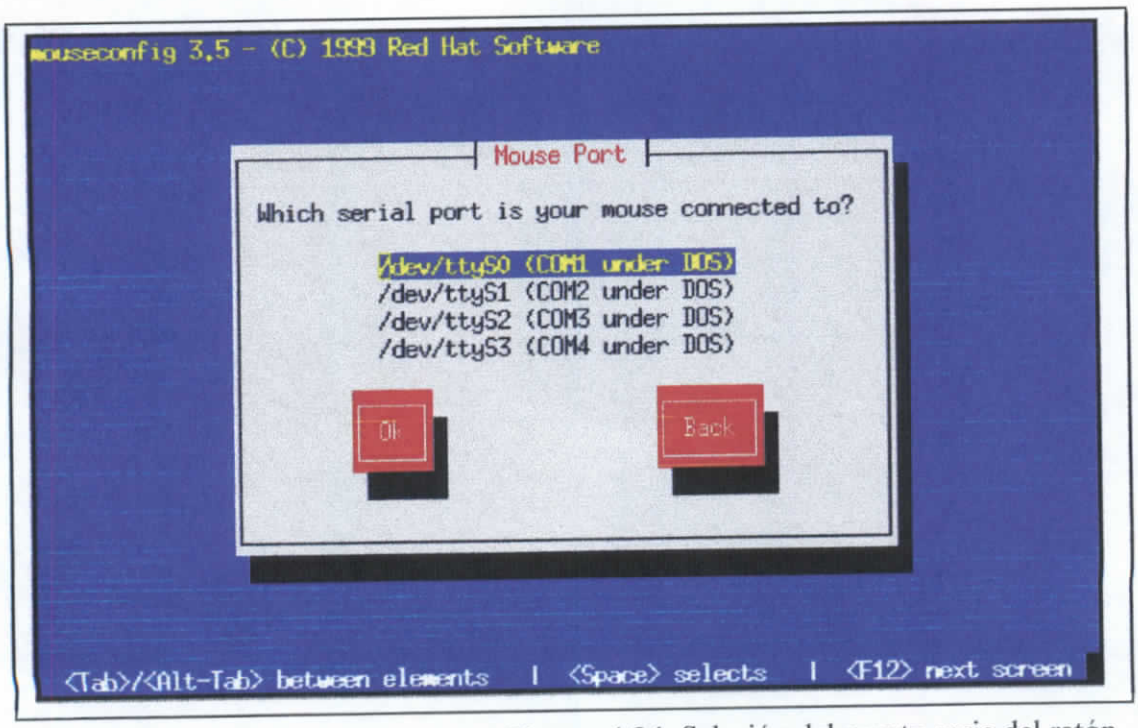


Figura 4.24. Seleción del puerto serie del ratón

Después del proceso anterior no queda nada por hacer hasta que todos los paquetes se hayan instalado. La rapidez con que esto sucede depende del número de paquetes que haya seleccionado y de la velocidad del equipo.

Si desea cambiar la configuración del ratón una vez inicializado el sistema Red Hat Linux, se podría utilizar el comando /usr/sbin/mouseconfig.

4.6.1.10. CONFIGURACIÓN DE LINUXCONF

Linuxconf es una utilidad que permite configurar y controlar varios aspectos del sistema, y es capaz de manejar un amplio rango de programas y tareas, describir de forma precisa la localización de pantallas específicas dentro de Linuxconf es sencillo, pero bastante largo debido a la naturaleza jerárquica de Linuxconf.

Si la estructura fuese un árbol genealógico, la mayoría de las pantallas de entrada de datos estarían en la cuarta generación.

4.6.1.11. EJECUCIÓN DE LINUXCONF

Para ejecutar Linuxconf es necesario tener privilegios de root. Linuxconf tiene las siguientes interfaces de usuario:

- Línea de órdenes
- Modo texto
- Basada en X Window
- Basada en Web

Linuxconf comenzará en el modo texto o en el modo X, dependiendo de la variable de entorno DISPLAY. La primera vez que se ejecute Linuxconf, será mostrado un mensaje introductorio, que aunque sólo aparece una vez, podrá acceder a la misma información a través de la ayuda de la pantalla principal.

4.6.1.12. INTERFAZ DE ÁRBOL DE MENÚS

La nueva versión de Linuxconf viene con una interfaz de árbol de menús, tal como se muestra en la figura 4.25.

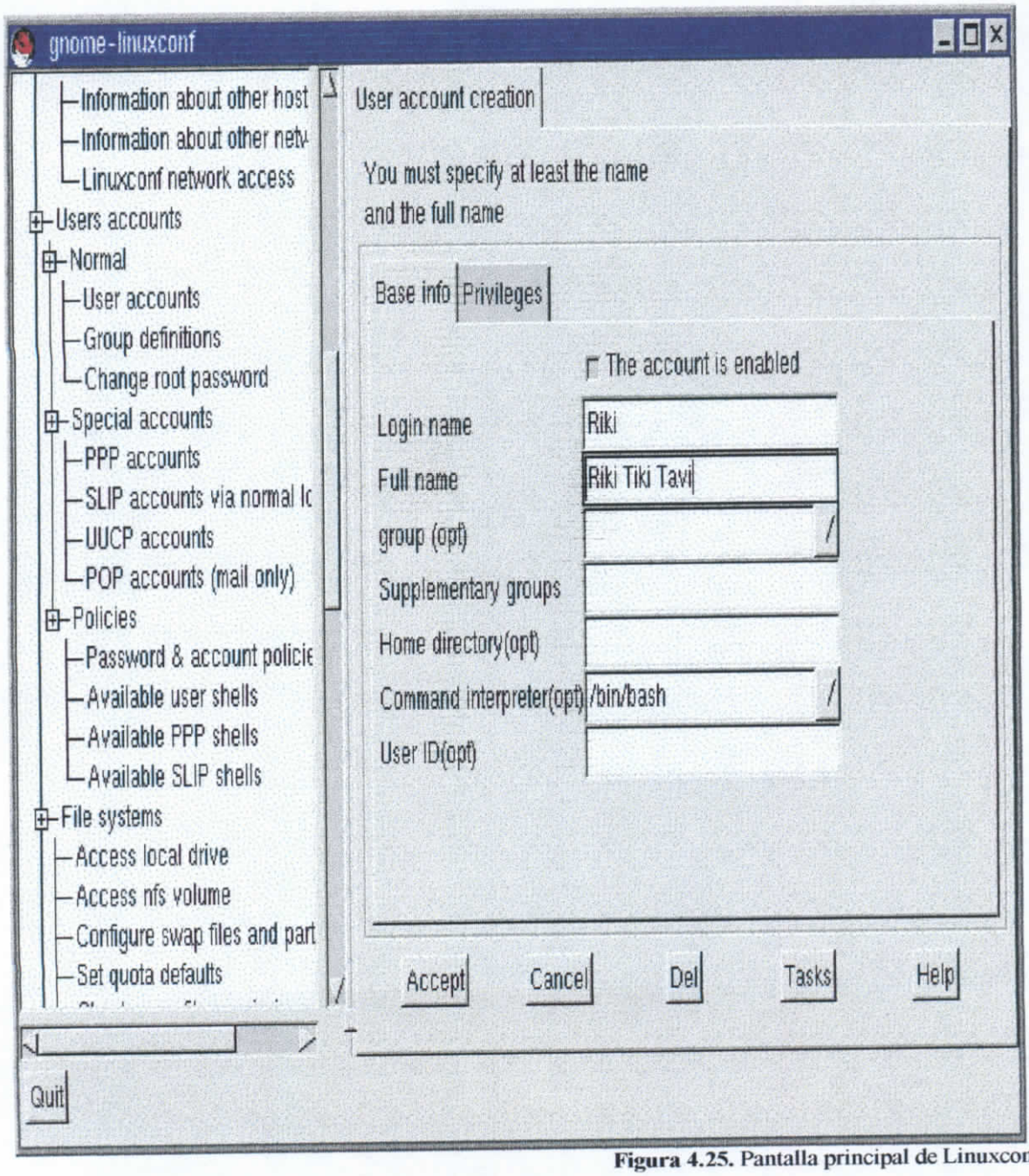


Figura 4.25. Pantalla principal de Linuxconf

4.6.1.13. INSTALACIÓN DEL LILO

Si se realizó una instalación de tipo estación de trabajo o de tipo servidor, esta parte del proceso de instalación se realiza automáticamente. Para poder arrancar el sistema Red Hat Linux, normalmente se necesita instalar LILO (el cargador de linux).

Es recomendado si ya se utiliza otro cargador de arranque en su sistema (tal como OS/2's Boot Manager Windows 2000). En este caso, el otro cargador de arranque tomará primero el control.

Una caja de dialogo aparecerá permitiéndole elegir el tipo de instalación de LILO que se desea, debe seleccionar la localización donde se desea instalar LILO y pulsar Ok. Si no se desea instalar LILO, se debe pulsar pasar.

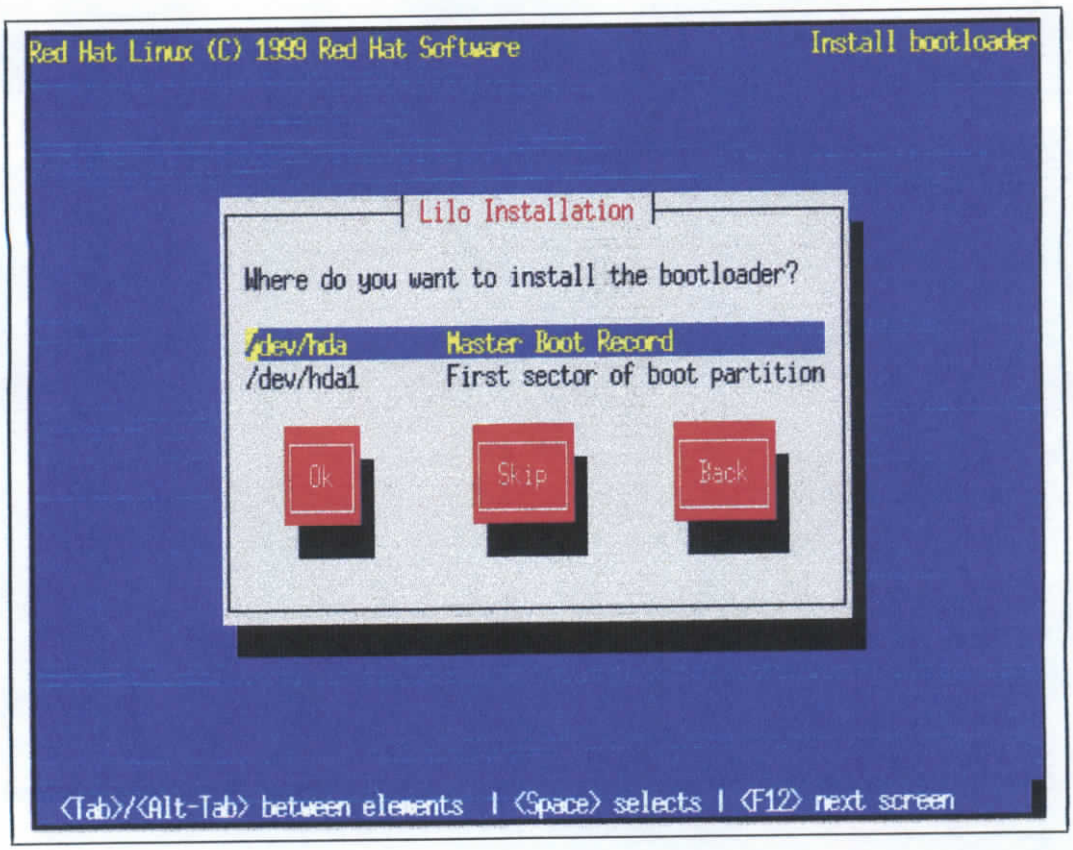


Figura 4.26. Instalación del LILO

Si se elige Pasar, no se podrá arrancar el sistema Red Hat Linux directamente, y se necesitará usar otro método de arranque (tal como un disquete de arranque). La pantalla para escoger la instalación del LILO Boot se muestra en la figura 4.26.

4.6.1.14. PLACAS BASE SMP Y LILO

En esta sección se especifica únicamente las placas base SMP. Es decir si el programa instalador detecta una placa base SMP en el sistema, automáticamente creará dos entradas en lilo.conf en contraposición a la habitual entrada simple:

Una entrada se llamará smp y la otra linux. La entrada smp será la que arranque por defecto. A menudo, si se tiene problemas con el núcleo smp, se podrá arrancar en su lugar con la entrada linux.

4.6.1.15. OPCIONES A LA LINEA DE COMANDOS DE LILO

A continuación, el programa de instalación le preguntará si se desea añadir opciones por defecto al comando de arranque de LILO, tal como se muestra en la figura 4.27.

Algunas de las opciones introducidas se pasarán al núcleo de linux cada vez que se arranque, mientras se revisa los parámetros de la BIOS de la computadora, y si encuentra que la computadora accede al disco duro en modo LBA, se debe elejir la opción Usar modo lineal.

Figura4.27. Opciones a la Línea de Comandos de Lilo

4.6.2. JAVA DEPLOYMENT KIT

Java surgió en 1991 cuando un grupo de ingenieros de *Sun Microsystems* trataron de diseñar un nuevo lenguaje de programación destinado a electrodomésticos. La reducida potencia de cálculo y memoria de los electrodomésticos llevó a desarrollar un lenguaje sencillo capaz de generar código de tamaño muy reducido.

Debido a la existencia de distintos tipos de CPUs y a los continuos cambios, era importante conseguir una herramienta independiente del tipo de CPU utilizada.

Desarrollaron un código "neutro" que no dependía del tipo de electrodoméstico, el cual se ejecutaba sobre una *"máquina hipotética o virtual*" denominada ***Java Virtual Machine (JVM***). Era la ***JVM*** quien interpretaba el código neutro convirtiéndolo a código particular de la CPU utilizada. Esto permitía lo que luego se ha convertido en el principal lema del lenguaje: *"Write Once, Run Everywhere*". A pesar de los esfuerzos realizados por sus creadores, ninguna empresa de electrodomésticos se interesó por el nuevo lenguaje.

Como lenguaje de programación para computadoras, ***Java*** se introdujo a finales de 1995. La clave fue la incorporación de un intérprete ***Java*** en la versión 2.0 del programa Netscape Navigator, produciendo una verdadera revolución en Internet. ***Java 1.1*** apareció a principios de 1997, mejorando sustancialmente la primera versión del lenguaje. ***Java 1.***2, más tarde rebautizado como ***Java*** 2, nació a finales de 1998.

Al programar en ***Java*** no se parte de cero. Cualquier aplicación que se desarrolle "cuelga" (o se apoya, según como se quiera ver) en un gran número de ***clases*** preexistentes. Algunas

de ellas las ha podido hacer el propio usuario, otras pueden ser comerciales, pero siempre hay un número muy importante de clases que forman parte del propio lenguaje (el ***API*** o ***Application Programming Interface*** de ***Jav***a). ***Java*** incorpora en el propio lenguaje muchos aspectos que en cualquier otro lenguaje son extensiones propiedad de empresas de software o fabricantes de ordenadores (threads, ejecución remota, componentes, seguridad, acceso a bases de datos, etc.). Por eso muchos expertos opinan que ***Java*** es el lenguaje ideal para aprender la informática moderna, porque incorpora todos estos conceptos de un modo estándar, mucho más sencillo y claro que con las citadas extensiones de otros lenguajes. Esto es consecuencia de haber sido diseñado más recientemente y por un único equipo.

El principal objetivo del lenguaje ***Java*** es llegar a ser el "nexo universal" que conecte a los usuarios con la información, esté ésta situada en el ordenador local, en un servidor de ***Web***, en una base de datos o en cualquier otro lugar.

Java es un lenguaje muy completo (de hecho se está convirtiendo en un macro-lenguaje: ***Java 1.0*** tenía 12 packages; ***Java 1.1*** tenía 23 y ***Java 1.2*** tiene 59). En cierta forma casi todo depende de casi todo. Por ello, conviene aprenderlo de modo *iterativo*: primero una visión muy general, que se va refinando en sucesivas iteraciones.

Una forma de hacerlo es empezar con un ejemplo completo en el que ya aparecen algunas de las características más importantes.

La compañía **Sun** describe el lenguaje **Java** como "simple, orientado a objetos, distribuido, interpretado, robusto, seguro, de arquitectura neutra, portable, de altas prestaciones,

multitarea y dinámico". Además de una serie de halagos por parte de ***Sun*** hacia su propia criatura, el hecho es que todo ello describe bastante bien el lenguaje ***Java***, aunque en algunas de esas características el lenguaje sea todavía bastante mejorable.

4.6.2.1. INTRODUCCIÓN A JAVA

Java 2 (antes llamado **Java 1.2** o **JDK 1.2**) es la tercera versión importante del lenguaje de programación ***Java***.

No hay cambios conceptuales importantes respecto a ***Java 1.1*** (en ***Java 1.1*** sí los hubo respecto a ***Java 1.0***), sino extensiones y ampliaciones, lo cual hace que a muchos efectos por ejemplo, para esta introducción sea casi lo mismo trabajar con ***Java 1.1*** o con ***Java 1.2.2***.

Los programas desarrollados en ***Java*** presentan diversas ventajas frente a los desarrollados en otros lenguajes como C/C++. La ejecución de programas en ***Java*** tiene muchas posibilidades: ejecución como aplicación independiente ***(Stand-alone Application)***, ejecución como **applet**, ejecución como **servlet,** etc.

Un ***applet*** es una aplicación especial que se ejecuta dentro de un navegador o browser (por ejemplo Netscape Navigator o Internet Explorer) al cargar una página HTML desde un servidor **Web**. El **applet** se descarga desde el servidor y no requiere instalación en el ordenador donde se encuentra el browser.

Un **servlet** es una aplicación sin interfaces gráfica que se ejecuta en un servidor de Internet.

La ejecución como aplicación independiente es análoga a los programas desarrollados con otros lenguajes.

Además de incorporar la ejecución como **Applet**, ***Java*** permite fácilmente el desarrollo tanto de arquitectura cliente-servidor como de aplicaciones distribuidas, consistentes en crear aplicaciones capaces de conectarse a otros ordenadores y ejecutar tareas en varios ordenadores simultáneamente, repartiendo por lo tanto el trabajo, aunque también otros lenguajes de programación permiten crear aplicaciones de este tipo, ***Java*** incorpora en su propio ***API*** estas funcionalidades.

4.6.2.2. ENTORNO DE DESARROLLO DE JAVA

Existen distintos programas comerciales que permiten desarrollar código ***Jav***a. La compañía ***Su***n, creadora de ***Jav***a, distribuye gratuitamente el *Java(tm) Development Kit* ***(JD***K). Se trata de un conjunto de programas y librerías que permiten desarrollar, compilar y ejecutar programas en ***Jav***a. Incorpora además la posibilidad de ejecutar parcialmente el programa, deteniendo la ejecución en el punto deseado y estudiando en cada momento el valor de cada una de las variables (con el denominado ***Debugge***r).

Cualquier programador con un mínimo de experiencia sabe que una parte muy importante (muchas veces la mayor parte) del tiempo destinado a la elaboración de un programa se destina a la detección y corrección de errores. Existe también una versión reducida del

*JD*K, denominada ***JRE*** *(Java Runtime Environment)* destinada únicamente a ejecutar código ***Java*** (no permite compilar).

Los *ID*Es *(Integrated Development Environment)*, tal y como su nombre indica, son entornos de desarrollo integrados.

En un mismo programa es posible escribir el código ***Jav***a, compilarlo y ejecutarlo sin tener que cambiar de aplicación. Algunos incluyen una herramienta para realizar ***Debug*** gráficamente, frente a la versión que incorpora el ***JDK*** basada en la utilización de una consola (denominada habitualmente ventana de comandos de MS-DOS, en ***Windows NT/95/9***8) bastante difícil y pesada de utilizar. Estos entornos integrados permiten desarrollar las aplicaciones de forma mucho más rápida, incorporando en muchos casos librerías con ***componentes*** ya desarrollados, los cuales se incorporan al proyecto o programa.

Como inconvenientes se pueden señalar algunos fallos de compatibilidad entre plataformas, y ficheros resultantes de mayor tamaño que los basados en clases estándar.

4.6.2.3. COMPILADOR DE JAVA

Se trata de una de las herramientas de desarrollo incluidas en el *JD*K. Realiza un análisis de sintaxis del código escrito en los ficheros fuente de ***Java*** (con extensión ****.jav***a). Si no encuentra errores en el código genera los ficheros compilados (con extensión ****.clas***s). En el ***JDK1.2.2*** dicho compilador **javac** Tiene numerosas opciones.

4.6.2.4. VARIABLES DEL ENTORNO JAVA

El desarrollo y ejecución de aplicaciones en ***Java*** exige que las herramientas para compilar ***(javac)*** y ejecutar ***(java)*** se encuentren accesibles. El ordenador, desde una ventana de sesión, sólo es capaz de ejecutar los programas que se encuentran en los directorios indicados del ordenador (o en el directorio activo).

Si desea compilar o ejecutar código en ***Jav***a, el directorio donde se encuentran estos programas deberá encontrarse en el ***PAT***H. Tecleando EXPORT ***PATH = $PATH:/bin*** en una ventana de sesión de Linux.

Al momento de compilar **Java *genera archivos de extensión CLASS***, la cual determina dónde buscar tanto las clases o librerías de ***Java*** (el ***API*** de ***Jav***a) como otras clases de usuario.

La variable ***CLASS*** puede incluir la ruta de directorios o donde se encuentran los programas ****.class***s.

Un fichero llamado ayuda***.class***, sería como sigue:

Export PATH=$PATH:/jdk1.2.2/bin

lo cual sería válido en el caso de que el ***JDK1.2.2*** estuviera situado en el directorio **/jdk1.2.2/bin**

Si no se desea tener que ejecutar este fichero cada vez que se abre una consola de Linux es necesario indicar estos cambios de forma "permanente". La forma de hacerlo difiere ***entre Windows y Linux*** es necesario modificar la ruta en C:\, añadiendo las líneas antes mencionadas.

Una vez abierta una sesión el ordenador ejecutará la línea que a continuación presentamos:

PATH=$PATH:/jdk1.2.2/bin

4.6.3 IPCHAINS

Esta sección describe lo que realmente necesitamos saber para construir un filtro de paquetes que llene nuestras necesidades.

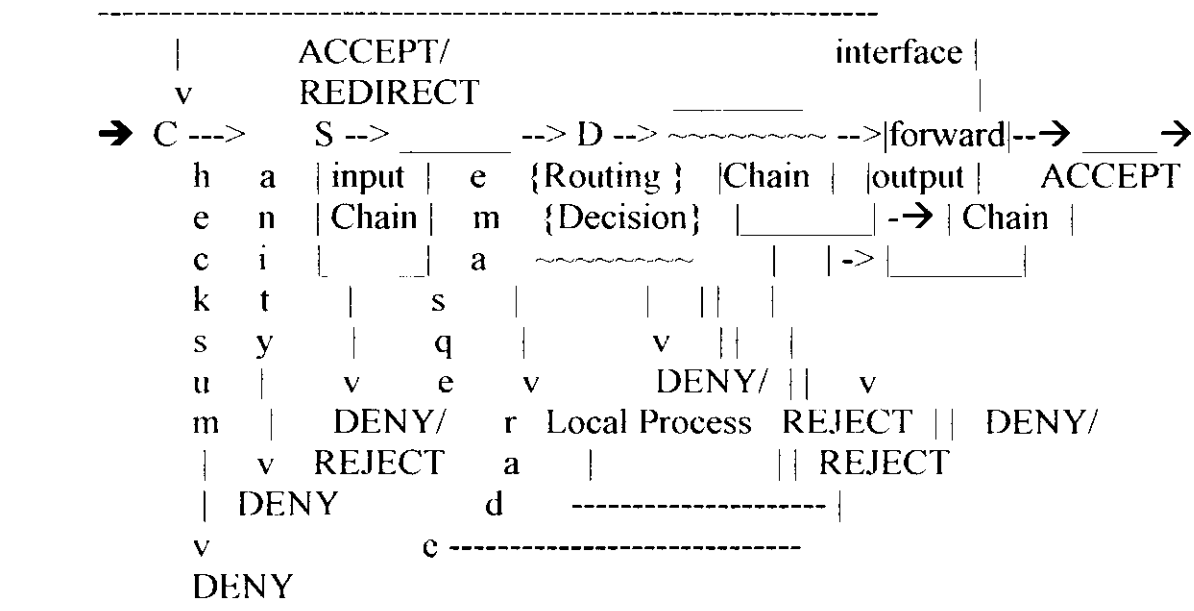
4.6.3.1. PAQUETES QUE ATRAVIESAN FILTROS.

El Kernel comienza con tres listas de reglas, estas listas son llamadas cadenas (chains) de cortafuegos o simplemente cadenas. Las tres cadenas son llamadas **entrada (input), salida (output) y envio (forward)**. Cuando un paquete entra (digamos, a través de la tarjeta Ethernet) el kernel usa la cadena **input** para decidir su destino. Si sobrevive este paso, entonces el kernel decide dónde enviar el paquete (esto se llama enrutamiento --**routing**--). Si el destino es otra máquina, consulta la cadena **forward.** Finalmente, justo antes de que el paquete salga, el kernel consulta la cadena **output.**

Una cadena es una lista (de chequeo -- checklist) de **reglas**. Cada regla dice 'si el encabezado del paquete se ve como esto, entonces esto es lo que deseo hacer con el paquete'.

Si la regla no empareja con el paquete, entonces se consulta la próxima regla en la cadena. Finalmente, si no hay ninguna regla más por consultar, entonces el kernel mira la **política** de la cadena para decidir qué hacer. En un sistema de seguridad-consciente, esta política normalmente le dice al kernel que rechace o deniegue el paquete.

Para los entusiastas del arte ASCII, esto muestra el camino completo que sigue un paquete que entra en una máquina.



- **CHECKSUM**

En esta fase se comprueba que el paquete no se ha adulterado de alguna manera. Si así pasa, se deniega.

➢ SANIDAD

Realmente hay uno de estos verificadores de sanidad antes de cada cadena del cortafuegos, pero la cadena de entrada (input) es más importante. Algunos paquetes malformados podrían confundir el código de verificación de la regla, y éstos son denegados aquí (un mensaje queda en el syslog si esto pasa).

➢ CADENA DE ENTRADA (INPUT CHAIN)

Ésta es la primera cadena que el cortafuegos probará contra el paquete. Si el resultado de la cadena es no denegar o no rechazar, el paquete sigue adelante.

➢ DESENMASCARAMIENTO (DEMASQUERADE)

Si el paquete es una respuesta a un paquete previamente enmascarado, es desenmascarado, y pasa directamente a la cadena de salida (output chain). Si no usa Ip Masquerading, puede borrar este diagrama mental.

➢ DECISIÓN DE ENRUTAMIENTO (ROUTING)

El campo del destino es examinado por el código de enrutamiento, para decidir si este paquete debe ser enviado a un proceso local (veáse proceso Local más abajo) o remitido a una máquina remota (veáse Cadena de envio -- forward -- más abajo).

➢ PROCESO LOCAL

Un proceso que corre en la máquina puede recibir paquetes después de la fase de decisión de enrutamiento, y puede enviar paquetes (que pasan por la cadena de salida --output chain-- después a la cadena de entrada --input chain-- por medio de la interface 'lo' si están destinados para un proceso local, caso contrario solo atraviesan la cadena de salida). Este cruce entre cadenas por inter-proceso no se muestra totalmente en el diagrama, es demasiado gráfico..

➢ LOCAL

Si el paquete no fue creado por un proceso local, entonces se verifica la cadena de envio (forward chain), por otra parte el paquete va directamente a la cadena de salida.

➢ CADENA DE ENVIO (FORWARD CHAIN)

Esta cadena es atravezada por cualquier paquete que esté intentando pasar a través de esta máquina a otra.

➢ CADENA DE SALIDA (OUTPUT CHAIN)

Esta cadena es atravesada justo antes de ser enviados.

4.6.3.2. USANDO IPCHAINS

Primero, verifique que usted tiene la versión de ipchains a la que este documento se refiere:

$ ipchains --version

ipchains 1.3.5, 26-June-1998 .

El ipchains tiene una página del manual bastante detallada (man ipchains), y si necesita más detalles particulares, puede comprobar la interface de programación (man 4 ipfw), o el archivo net/ipv4/ip_fw.c en el fuente del kernel 2.1.x que es (obviamente) autoritario.

Hay varias cosas diferentes que puede hacer con ipchains. Primero las operaciones para manejar cadenas enteras. Empieza con la construccion de tres cadenas input, output y forward, las cuales no puede borrar.

- Crear una nueva cadena (-N).
- Borrar una cadena vacía (-X).
- Cambiar la politica para una cadena construida. (-P).
- Listar las reglas de una cadena (-L).
- Vaciar las reglas fuera de una cadena (-F).
- Poner a cero el contador de paquete y de byte en todas las reglas en una cadena (-Z).
- Hay varias formas de manipular reglas dentro de una cadena:
- Añadir una nueva regla a una cadena (-UN).
- Insertar una nueva regla en alguna posición en una cadena (-yo).
- Reemplazar una regla en alguna posición en una cadena (-R).
- Anular una regla en alguna posición en una cadena (-D).

- Anular la primera regla que empareja en una cadena (-D).

Hay unas pocas operaciones para enmascarar, que se encuentran en ipchains por la necesidad de un buen sitio para ponerlas:

- Listar las conexiones enmascaradas actualmente (-M -L).
- Asignar valores de timeout de enmascaramiento (-M -S).

La función final (y quizás la más útil) es la que permite verificar lo que le pasaría a un paquete dado si este atraviesa por una cadena dada.

4.6.3.3. OPERACIONES EN UNA SOLA REGLA

Esto es lo práctico de ipchains; manipulación de reglas. En la mayoría de casos, usted probablemente usará los comandos de adicionar (-UN) y borrar (-D).

Los otros (-I para insertar y -R para remplazar) son simples extensiones de estos conceptos.

Cada regla especifica un juego de condiciones que el paquete debe cumplir, y qué hacer si se las encuentra (un objetivo -- target--).

Por ejemplo, usted podría querer denegar todos los paquetes de ICMP que vienen de la dirección IP 127.0.0.1. Así que, nuestras condiciones son: que el protocolo debe ser ICMP y que la dirección origen debe ser 127.0.0.1. Nuestro objetivo es denegar `DENY '.

127.0.0.1 es la interface de retorno --loopback-- que tendrá aun cuando no tenga ninguna conexión real de red. Usted puede usar el programa `ping ' para generar tales paquetes (envía un ICMP de tipo 8 (echo request) al que todos los host coperativos deben obligatoriamente responder con un paquete ICMP de tipo 0 (echo reply)). Esto es útil para probar:.

```
# ping -c 1 127.0.0.1
PING 127.0.0.1 (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.2 ms

--- 127.0.0.1 ping statistics ---
 packets transmitted,
1 packets received, 0% packet loss round-trip min/avg/max = 0.2/0.2/0.2 ms

# ipchains -A input -s 127.0.0.1 -p icmp -j DENY
# ping -c 1 127.0.0.1

PING 127.0.0.1 (127.0.0.1): 56 data bytes

--- 127.0.0.1 ping statistics ---

1 packets transmitted, 0 packets received, 100% packet loss
#
```

Puede ver aquí que el primer ping fue exitoso (el ` -c 1 ' indica a ping que envie sólo un paquete).

Luego adicionamos (-UN) una cadena `input ', una regla que especifica que para los paquetes que provienen de 127.0.0.1 (-s 127.0.0.1') con protocolo ICMP (-p ICMP) debemos pasar a DENEGAR (` -j DENY').

Luego probamos nuestra regla, usando el segundo ping. Habrá una pausa antes de que el programa deje de esperar por una respuesta que nunca vendrá.

Podemos borrar la regla de cualquiera de dos formas. Primero, puesto que sabemos que es la única regla en la cadena input, podemos usar un borrado numerado, así:

```
# ipchains -D input 1
#
```

Para borrar la regla número 1 en la cadena input.

La segunda forma es a través del comando -A, pero reemplazando el -A con -D. Esto es útil cuando tiene una cadena compleja de reglas y no quiere tener que contarlos para concluir que es la regla 37 de la que desea librarse. En este caso, usamos:

```
# ipchains -D input -s 127.0.0.1 -p icmp -j DENY
#
```

La sintaxis de -D debe tener exactamente las mismas opciones como en el comando -A (o -I o -R). Si hay multiples reglas idénticas en la misma cadena, sólo la primero se borrará.

4.6.3.4. ESPECIFICACIONES DE FILTRADO

Hemos visto el uso de ` -p ' para especificar protocolo, y ` -s ' para especificar dirección del origen, pero hay otras opciones que podemos uar para especificar las características del paquete. Lo que sigue es un compendio exhaustivo.

4.6.3.5. ESPECIFICANDO DIRECCIONES IP ORÍGEN Y DESTINO

Direcciones IP de Origen (-s) y destino (-d) pueden especificarse de cuatro maneras. La manera más común es usar el nombre completo, como `localhost ' o `www.linuxhq.com '. La segunda manera es especificar la dirección IP tal como `127.0.0.1 '.

La tercera y cuarta formas permiten especificar un grupo de direcciones IP, como `199.95.207.0/24 ' o `199.95.207.0/255.255.255.0 '. Estas dos especifican cualquier IP del rango desde 192.95.207.0 hasta 192.95.207.255; los digitos después del ` / ' dicen qué partes de la dirección de IP son significantes. ` /32 ' o ` /255.255.255.255 ' es el valor por defecto (todas las direcciones de IP). Para especificar cualquier IP se puede usar ` /0 ' , así:

```
# ipchains -A input -s 0/0 -j DENY
#
```

Esto rara vez se usa, puesto que el efecto es el mismo que no especificar la opción -s '.

4.6.3.6. ESPECIFICANDO INVERSIÓN (CONTRARIOS)

Muchas marcas, incluso las marcas ` -s ' y ` -d ' pueden tener sus argumentos precedidos por `! ' (`pronunciado NOT') para referir a direcciones que no son iguales a las dadas. Por ejemplo ` -s ! localhost' se refiere a cualquier paquete que no proviene del localhost.

4.6.3.7. ESPECIFICANDO PROTOCOLO

El protocolo puede especificarse con la marca ` -p '. El protocolo puede ser un número (si usted conoce los valores numéricos de protocolos para IP) o un nombre para los casos especiales de `TCP ', `UDP ' o `ICMP '. No importa mínusculas o mayúsculas, `tcp ' hace lo mismo que `TCP '.

El nombre de protocolo puede estar precedido por '!', para invertirlo, tal como ' -p ! TCP'.

4.6.3.8. ESPECIFICANDO PUERTOS UDP Y TCP

Para el caso especial donde un protocolo de TCP o UDP se especifica, puede haber un argumento extra que indica el puerto TCP o UDP, o un rango de puertos.

Un rango es representado usando el caracter ":", como "6000:6010" que cubre 11 números de puerto, desde 6000 hasta 6010. Si se omite el limite inferior, se asigna por defecto 0. Si se omite el limite superior, se asigna por defecto 65535.

Así pues, para especificar conexiones de TCP en los puertos por debajo del 1024, la sintáxis seria ' -p TCP -s 0.0.0.0/0 :1024 ' . Los números de puerto pueden ser especificados usando nombre, ej. 'www'.

Note que el puerto especificado puede estar precedido por '!' para indicar lo contrario. Así pues, para especificar todos los paquetes TCP excepto los WWW, lo indicará así:

-p TCP -d 0.0.0.0/0 ! www

Es importante comprender que la especificación

-p TCP -d ! 192.168.1.1 www

es muy diferente de

-p TCP -d 192.168.1.1 ! www

La primera especifica cualquier paquete de TCP al puerto de WWW en cualquier máquina excepto 192.168.1.1. El segundo especifica alguna conexión de TCP a cualquier puerto en 192.168.1.1 excepto el puerto de WWW.

Finalmente, este caso significa que no a los de puerto WWW y no a los de dirección 192.168.1.1:

-p TCP -d ! 192.168.1.1 ! www

4.6.3.9. ESPECIFICANDO ICMP TIPO & CÓDIGO

ICMP también permite un argumento opcional, puesto que ICMP no tiene puertos, (ICMP tiene un **tipo** y un **código**) los cuales tienen un significado diferente.

Puede especificarlos como nombres ICMP (use ipchains -h icmp para listar los nombres) después de la opción ` -s ', o como un tipo y código número ICMP, donde el tipo sigue a la opción '-s' y el código sigue a la opción '-d'.

Los nombres de ICMP son bastante largos: sólo necesita usar las letras suficientes para hacer que un nombre sea distinto de otro.

Aquí está una pequeña tabla de algunos de los más comunes paquetes ICMP:

Número	Nombre	Requerido por
0	echo-reply	ping
3	destination-unreachable Any	TCP/UDP traffic.
5	redirect	routing if not running routing daemon
8	echo-request	ping
11	time-exceeded	traceroute

Note que por el momento los nombres de ICMP no pueden ser precedidos por `! '. NO.

4.6.3.10. ESPECIFICANDO UNA INTERFACE

La opción ` -i ' especifica el nombre de una **interface** para comparar. Una interfaz es el dispositivo fisico por el cual un paquete llega o sale. Puede usar el comando ifconfig para listar las interfaces activas. (ie. que estén funcionando en el momento).

La interfaz para los paquetes entrantes (ie. paquetes que atraviesan la cadena input) es considerada la interfaz por donde ellos entran. Lógicamente, la interfaz para los paquetes salientes (paquetes que atraviesan la cadena output) es la interfaz por donde ellos saldrán. La interfaz para que los paquetes atraviesen la cadena forward es también la interfaz por donde ellos saldrán; una decisión bastante arbitraria a nuestro modo de ver.

Es absolutamente legal especificar una interfaz que actualmente no existe; la regla no emparejará nada hasta la interface sea activada. Esto es sumamente útil para enlaces telefónicos ppp (normalmente interface ppp0) y similares.

Como un caso especial, un nombre de interfaz que termine con un `+' emparejará todas las interfaces (si ellas existen actualmente o no) qué empieza con esa cadena. Por ejemplo, para especificar una regla que se refiere a todas las interfaces PPP entonces se usaría -i ppp+.

El nombre de la interface puede ir precedido por `! ' para indicar un paquete que no empareja con la interface(s) especificada.

4.6.3.11. ESPECIFICANDO PAQUETES SYN TCP SOLAMENTE

A veces es útil permitir conexiones de TCP en un sentido, pero no en el otro. Por ejemplo, podría permitir conexiones hacia un servidor de WWW externo, pero no las que provienen desde él.

El acercamiento ingenuo sería bloquear paquetes de TCP que vienen del servidor. Desafortunadamente, para que trabajen las conexiones de TCP requieren paquetes que van en ambos sentidos.

La solución es bloquear sólo los paquetes de petición de conexión. A estos paquetes se les llama paquetes **SYN** (ok, técnicamente son aquellos paquetes con la marca SYN activada (SYN flag) y las marcas FIN y ACK limpias, pero nosotros los llamaremos paquetes SYN).

Impidiendo sólo estos paquetes, podemos detener los intentos de conexión .

La marca ` -y ' se usa para esto: es sólo válida para reglas que especifican TCP como su protocolo. Por ejemplo, especificar intentos de conexión TCP desde 192.168.1.1:

-p TCP -s 192.168.1.1 -y

Una vez más, esta marca puede ser invertida precediéndola con un `! ' que significa todos los paquetes restantes a los que inician conexión.

4.6.3.12. MANEJANDO FRAGMENTOS.

A veces un paquete es demasiado grande para alcanzar en un sola transmisión todo de una vez. Cuando esto pasa, el paquete es dividido en **fragmentos** , y se envía como múltiples paquetes. En el otro extremo se reensamblan los fragmentos para reconstruir el paquete entero.

El problema con fragmentos es que algunas de las especificaciones que se mencionaron anteriormente (en particular, puerto orígen, puerto destino, tipo ICMP, código ICMP, o marca TCP SYN) requieren que el kernel mire al inicio del paquete, los cuales son contenidos unicamente en el primer fragmento.

Si su máquina es la única conexión a una red externa, entonces puede indicar al kernel que ensamble los fragmentos que lo atraviesan, seteando a 'Y' la opción IP: always defragment. Esto envita el problema limpiamente..

Por otra parte, es importante entender cómo los fragmentos son tratados por las reglas de filtración. Alguna regla que nos pregunte por información que no se tiene, no emparejará. Esto significa que el primer fragmento se trata como cualquier otro paquete. El segundo y siguientes fragmentos no lo serán. Así pues, una regla (especificando un puerto de origen www) nunca emparejará con un fragmento (más que con el primer fragmento) Segundo y los fragmentos extensos no serán. Así la regla - **TCP -s 192.168.1.1 www** (especificando un puerto de la fuente de `www ') nunca emparejará un fragmento (otra cosa que el primer fragmento).

La regla op afectará al primer fragmento, de ese modo previene reensamblar en el host destino, sin embargo, se han detectado bugs que hacen que la máquina falle simplemente por enviar fragmentos.

Note que los encabezados de red: paquetes malformados (paquetes TCP, UDP e ICMP demasiado cortos para que el codigo del cortafuegos lea los puertos o los tipos o códigos ICMP) también son tratados como fragmentos. Sólo los fragmentos TCP que empiezan en la posición 8 son desechados explicitamente por el código de cortafuegos (un mensaje debe aparecer en el syslog si esto ocurre).

Como un ejemplo, la siguiente regla desecha cualquier fragmento que va a 192.168.1.1:

```
# ipchains -A output -f -D 192.168.1.1 -j DENY

#
```

4.6.3.13. EFECTOS LATERALES DEL FILTRADO

OK, hasta ahora sabemos todas las formas como podemos emparejar un paquete usando una regla. Si un paquete empareja con una regla, pasa lo siguiente:

- El contador de byte para esa regla es aumentado por el tamaño del paquete (encabezado y todo lo demás). El contador de paquetes para esa regla se incrementa. Si la regla lo requiere, el paquete es anotado.

- Si la regla lo requiere, el campo Type Of Service es cambiado.

- Si la regla lo requiere, el paquete es marcado (no en las series 2.0 del kernel).

- El objetivo de la regla es examinado para determinar que se hace después con el paquete.

Para variar, está puesto en orden de importancia.

4.6.3.14. ESPECIFICANDO UN OBJETIVO

Un objetivo --**target**-- indica al kernel que hacer con un paquete que empareja con una regla. El ipchains usa '-j' (saltar a) para especificar el objetivo.

El caso más simple es cuando no se ha especificado el objetivo. Este tipo de regla (a menudo llamada regla de "accounting") es útil simplemente para contar un cierto tipo de paquetes. Empareje o no con la regla, el kernel examina la próxima regla en la cadena. Por ejemplo, para contar el número de paquetes provenientes de 192.168.1.1, podemos hacer esto:

```
# ipchains -A input -s 192.168.1.1
#
```

(Usando `ipchains -L -v ' podemos ver el contador de bytes y de paquetes asociados con cada regla).

Hay seis objetivos especiales. Los tres primeros, son bastante simples.

ACEPTAR (ACCEPT) RECHAZAR (REJECT) y DENEGAR (DENY). ACCEPT permite que el paquete lo atraviese.

DENY desecha el paquete como si nunca hubiera sido recibido. REJECT desecha el paquete, pero (si no es un paquete ICMP) genera una respuesta de ICMP al orígen que dice que el destino fue inalcanzable.

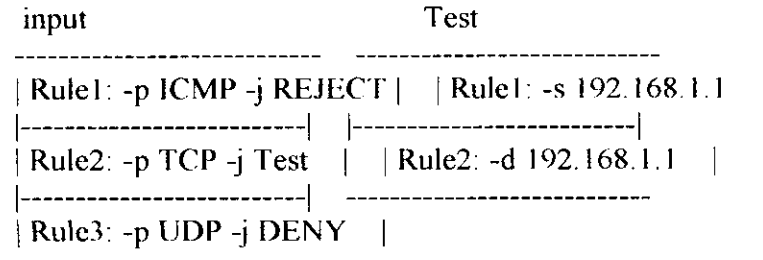
El próximo objetivo, MASQ le dice al kernel que enmascare el paquete. Para que esto funcione, su kernel debe ser compilado con Ip Masquerading habilitado. Para mas detalles, ver el IP-Masquerading-HOWTO y el Apéndice Diferencia entre ipchains e ipfwadm <HOWTO-7.html> Este objetivo solo es válido para los paquetes que atraviesan la cadena forward.

El otro objetivo especial es REDIRECT que le dice al kernel que envie el paquete a un puerto local en lugar de un sitio cualquiera donde fuera dirigido. Puede ser especificado solo para reglas que contienen protocolos TCP y UDP. Opcionalmente, un puerto (nombre o número) puede especificarse después de '-j REDIRECT' lo cual hace que el paquete sea redirigido a un puerto particular incluso si fue direccionado a otro puerto. Este objetivo es válido solo para paquetes que atraviesan la cadena input.

El último objetivo especial es RETURN el cual es similar a descender de la cadena inmediatamente.

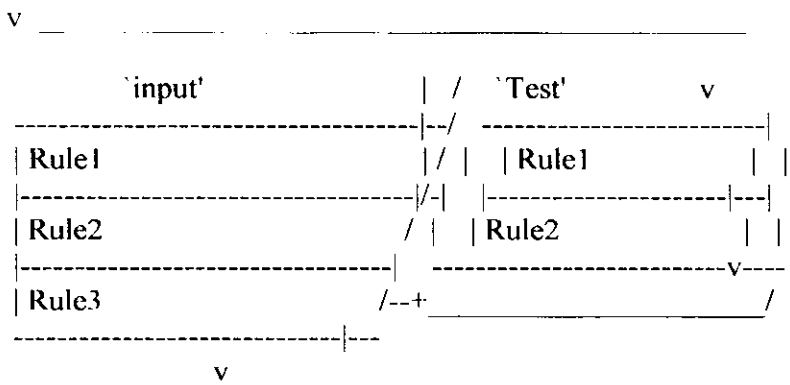
Cualquier otro objetivo indica una cadena definida por el usuario. El paquete empezará atravesando las reglas en esa cadena. Si esa cadena no decide el destino del paquete, entonces una vez que la atravesia por esa cadena ha terminado, la travesía se reasume en la próxima regla en la cadena actual.

Considere dos cadenas: input (la cadena construida --built-in--) y Test (una cadena definida por el usuario --user-defined--).



Considere un paquete de TCP que viene desde 192.168.1.1, que va hacia 1.2.3.4. Entra en la cadena input, y se prueba contra la Regla 1: no concuerda. Regla 2: concuerda, y el objetivo es Test, tal que la siguiente regla es la primera de Test. La regla 1 en test concuerda, pero no especifica un objetivo, tal que la próxima regla es examinada, Regla 2. Esta no concuerda, tal que hemos alcanzado el final de la cadena. Regresamos a la cadena input, ahora a la Regla 3, la cual no coincide.

Así que la ruta del paquete es:



Este es un efecto lateral que puede tener el emparejar una regla; puede tener la anotación del paquete emparejado usando la marca '-l'. Usualmente no querrá esto para los paquetes rutinarios, pero es una característica muy útil si desea observar eventos fuera de lo común.

El kernel anota esta información de esta forma:

Packet log: input DENY eth0 PROTO=17 192.168.2.1:53 192.168.1.1:1025

L=34 S=0x00 I=18 F=0x0000 T=254

Este mensaje de anotación es diseñado para ser conciso, y contener información técnica útil para los gurus de las redes, pero puede ser útil para el resto de nosotros. Se interpreta así:

`INPUT" es la cadena que contiene la regla la cual se ha emparejado con el paquete, causando el mensaje de anotación.

`DENY' es lo que la regla le dice al paquete. Si es '-' entonces la regla no afecta absolutamente al paquete (en reglas de accounting).

`eth0' es el nombre de la interface. Puesto que fue la cadena input, significa que el paquete llegó por 'eth0'.

`PROTO=17' significa que el paquete fue del protocolo 17. Una lista de los números de protocolos se encuentra en '/etc/protocols'. Los más comúnes son 1 (ICMP), 6 (TCP) y 17 (UDP).

`192.168.2.1' significa que la dirección IP de orígen del paquete fue 192.168.2.1.

`:53' significa que el puerto de orígen fue el 53. Mirando en '/etc/services' observa que este es el puerto 'domain' (ej. Es probablemente el puerto de respuesta del DNS).

Para UDP y TCP, este número es el puerto de origen. Para ICMP, es el tipo ICMP. Para otros, será 65535.

`192.168.1.1' es la dirección IP de destino

`:1025' significa que el puerto destino fue el 1025. Para UDP y TCP, este número es el puerto destino. Par ICMP, es el código ICMP. Para otros, será 65535.

`L=34' significa que el paquete tenia una longitud total de 34 bytes.

`S=0x00' significa que el campo Type Of Service (dividir por 4 para obtener el Tipo de Servicio tal como lo usa ipchains).

`I=18' es el ID de IP.

`F=0x0000' es el desplazamiento del fragmento de 16 bits más banderas. Un valor que comience con '0x4' o '0x5' significa que el bit de fragmento no está activo. '0x2' o '0x3' significa que el bit 'More Fragments' (Más fragmentos) está activo; se esperan más fragmentos después de este. El resto de números es el desplazamiento de este fragmento, dividido por 8.

`T=254' es el tiempo de vida (Time To Live) del paquete. Es deducido desde este valor para todo salto, y comunmente empieza en 15 o 255.

En sistemas Linux estándard, esta salida del kernel es capturada por klogd (el demonio de anotación del kernel) el cual lo entrega a syslogd (el demonio de anotación del sistema). El archivo '/etc/syslog.conf' controla el comportamiento de syslogd, porque especifica un destino para cada 'establecimiento -- facility' (en nuestro caso, el establecimiento es "kernel") y 'nivel -- level ' (para ipchains, el nivel usado es "info").

Por ejemplo, mi (Debian) /etc/syslog.conf contiene dos lineas que concuerdan con "kern.info":

```
kern.*-/var/log/kern.log
*.=info;*.=notice;*.=warn;\
        auth,authpriv.none;\
        cron,daemon.none;\
        mail,news.none          -/var/log/messages
```

Esto significa que los mensajes son duplicados en '/var/log/kern.log' y '/var/log/messages'. Para más detalles, ver 'man syslog.conf'.

4.6.3.15. ESTABLECIENDO LA POLÍTICA

Anteriormente no se dijo lo que pasa cuando un paquete alcanza el final de una cadena construida. En este caso, la **política** de la cadena determina el destino del paquete.

Sólo las cadenas construidas (input output forward) tiene políticas, porque si un paquete llega al final de una cadena definida por el usuario, pasaría a la cadena anterior.

La política puede ser cualquiera de los objetivos especiales: ACCEPT, DENY REJECT MASQ. MASQ es sólo válido para la cadena `forward '.

Note los problemas al poner valores de timeout en No puedo poner timeouts para enmascaramiento <HOWTO-6.html>.

4.6.3.17. CREACIÓN DE MULTIPLES REGLAS

A veces una simple línea de comando puede ocasionar que múltiples reglas sean afectadas. Esto se hace de dos formas: Primro, si especifica un hostname el cual se resuelve (usando DNS) a múltiples direcciones IP, ipchains actuará como si hubiese tecleado múltiples comandos con cada combinación de direcciones.

Así pues, si el hostname `www.foo.com ' se resuelve a tres direcciones IP, y el hostname 'www.bar.com' se resuelve a dos direcciones IP, entonces el comando `ipchains -A input -j reject -s www.bar.com -d www.foo.com' adicionaría seis reglas a la cadena input.

La otra forma de que ipchains realice acciones múltiples es usar la marca bidireccional (` -b '). Esta marca hace que ipchains se comporte como si usted hubiera tecleado dos veces, en la segunda los argumentos ` -s ' y ` -d ' se invierten. Así, para evitar remisiones (forwarding) hasta o desde 192.168.1.1, podría hacer a lo siguiente:

```
# ipchains -b -A forward -j reject -s 192.168.1.1
#
```

La opción -b puede usarse con las ordenes de inserción (` -I '), borrado (` -D ') (excepto en las variantes que toman un número de la regla), adición (` -A ') y Chequeo (` -C ').

Otra marca útil es ` -v ' (verbose) qué muestra exactamente lo que ipchains está haciendo con los comandos. Es útil cuando esta trabajando con órdenes que pueden afectar a múltiples reglas. Por ejemplo:.

```
# ipchains -v -b -C input -p tcp -f -s 192.168.1.1 -d 192.168.1.2 -i lo
 tcp opt ---f- tos 0xFF 0x00 via lo 192.168.1.1 -\062 192.168.1.2 * -\062 *
packet accepted
 tcp opt ---f- tos 0xFF 0x00 via lo 192.168.1.2 -\062 192.168.1.1 * -\062 *
packet accepted
```

4.6.3.18. EJEMPLOS

Tengo una conexión vía telefónica (-i ppp0). Tomo noticias (-p TCP -s news.virtual.net.au nntp) y correo (-p TCP -s mail.virtual.net.au pop-3) cada vez que me conecto. Uso el método ftp de Debian para actualizar mi máquina regularmente (-p TCP -y -s ftp.debian.org.au ftp-data). Navego en la web a través del proxy de mi ISP mientras tanto (-p TCP -d proxy.virtual.net.au 8080) , pero odio los anuncios de doubleclick.net en el archivo Dilbert (-p TCP -y -d 199.95.207.0/24 & -p TCP -y -d 199.95.208.0/24).

Esto es llamado comúnmente IP Spoofing, y hay una mejor manera de protegerse a si mismo, desde el kernel 2.1.x y superiores. Ver Cómo seteo la protección de Ip spoofing <HOWTO-5.html>?<HOWTO-5.html>.

Esta configuración es bastante simple, ya que no hay ninguna otra máquina en la red interna.

No se quiere que ningún proceso local (ie. Netscape, lince etc.) se conecte con doubleclick.net:

```
# ipchains -A output -d 199.95.207.0/24 -j REJECT
# ipchains -A output -d 199.95.208.0/24 -j REJECT
```

Ahora se quiere poner prioridades en varios paquetes salientes (no se gana mucho haciendolo en los paquetes entrantes). Puesto que tengo un número justo de estas reglas, tiene sentido ponerlos todos en una sola cadena, llamada. ppp-out

```
# ipchains -N ppp-out
# ipchains -A output -i ppp0 -j ppp-out
```

Retraso mínimo para tráfico de web & telnet.

```
# ipchains -A ppp-out -p TCP -d proxy.virtual.net.au 8080 -t 0x00 0x10
# ipchains -A ppp-out -p TCP -d 0.0.0.0 telnet -t 0x00 0x10
```

Prioridad baja para los datos del ftp, nntp, pop-3,:

```
# ipchains -A ppp-out -p TCP -d 0.0.0.0/0 ftp-data -t 0x00 0x02
# ipchains -A ppp-out -p TCP -d 0.0.0.0/0 nntp -t 0x00 0x02
# ipchains -A ppp-out -p TCP -d 0.0.0.0/0 pop-3 -t 0x00 0x02
```

Hay unas restricciones en paquetes que entran por la interface ppp0: creemos una cadena llamada `ppp-in ':

ipchains -N ppp-in

ipchains -A input -i ppp0 -j ppp-in

Ahora, ningún paquete que entra por ppp0 debe estar exigiendo una dirección de origen 192.168.1. *, así que lo anotamos y los denegamos:

ipchains -A ppp-in -s 192.168.1.0/24 -l -j DENY

#

Permitir conexiones DNS (remito todas las demandas a 203.29.16.1, así que espero respuesta DNS TCP sólo desde ellos), ftp, y retorno ftp-data solamente (las cuales solo deben ir a un puerto superior a 1023, y no a los puertos X11 alrededor de 6000)

ipchains -A ppp-in -p TCP -s 203.29.16.1 -d $LOCALIP dns -j ACCEPT

ipchains -A ppp-in -p TCP -s 0.0.0.0/0 ftp-data -d $LOCALIP 1024:5999 -j ACCEPT

ipchains -A ppp-in -p TCP -s 0.0.0.0/0 ftp-data -d $LOCALIP 6010: -j ACCEPT

ipchains -A ppp-in -p TCP -d $LOCALIP ftp -j ACCEPT

Finalmente, los paquetes local-a-local están OK:

ipchains -A input -i lo -j ACCEPT

Ahora, mi política predefinida en la cadena input es DENY, de tal forma que todo lo demás sea desechado.

ipchains -P input DENY

NOTA: No se debe configurar cadenas en este orden, mientras que los paquetes cruzan en el momento que se esta configurando. Lo más seguro es poner la política de DENY primero, luego insertar las reglas. Por supuesto, que si sus reglas exigen ver al DNS, podría tener problemas.

4.6.3.19. PAQUETES DE ICMP

Se usa paquetes de ICMP (entre otras cosas) para indicar fracaso para otros protocolos (como TCP y UDP). Paquetes `destino-inalcanzable ' en particular. Bloquear estos paquetes significa que nunca obtendrá errores de `Host unreachable' o `No route to host'. Cualquier conexión esperará por una respuesta que nunca vendrá. Esto es irritante, pero raramente fatal.

Un problema peor es el rol de los paquetes de ICMP en descubrimiento de MTU. Todas las buenas implementaciones de TCP (incluido Linux) usan el MTU, para intentar deducir que tan largos pueden ir los paquetes a un destino sin fragmentarse (la fragmentación ralentiza el rendimiento, sobre todo cuando ocasionalmente los fragmentos son perdidos).

El MTU trabaja enviando paquetes con el bit "Don't fragment" seteado, y luego enviando paquetes más pequeños si obtiene un paquete ICMP indicando "Fragmentation needed but DF set" (necesidad de fragmentación). Este es un tipo de paquete "destination-unreachable", y si nunca se recibe, el host local nunca reducirá la MTU.

Todas las aplicaciones de TCP buenas (Linux incluyó) use descubrimiento de MTU para intentar deducir eso que el paquete más grande que puede conseguir a un destino sin fragmentarse (la fragmentación retarda actuación, sobre todo cuando los fragmentos ocasionales están perdidos). El descubrimiento de MTU trabaja enviando paquetes con el "no Fragmenta" el pedazo puso, y entonces enviando paquetes más pequeños si consigue un paquete de ICMP que indica "la Fragmentación necesitó pero DF puso" (`fragmentation--ed '). Éste es un tipo de `destination-inalcanzable ' el paquete, y si nunca se recibe, el organizador local no reducirá MTU, y el rendimiento será abismal o no existente.

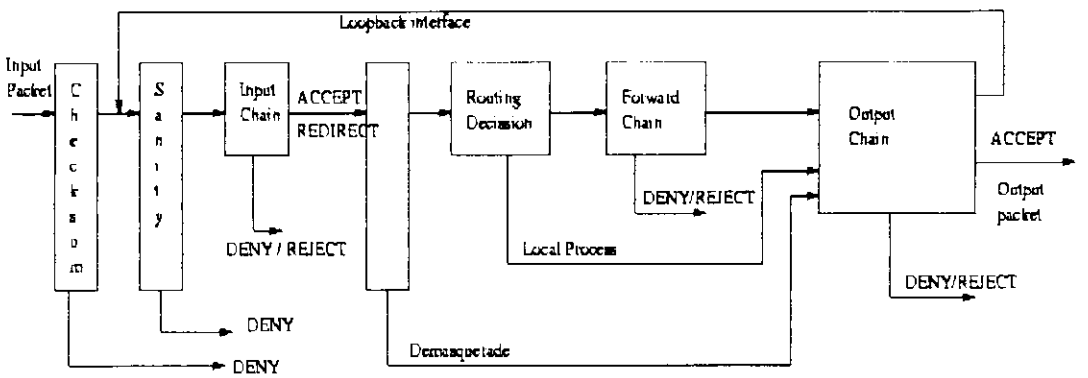


Figura 4.28. Filtrado de Paquetes

Además el usuario puede definir sus propias cadenas de filtros.

La figura 14.28 ilustra este esquema. Por cada paquete entrante, se analizan en orden cada una de las reglas definidas, y en su caso se toma una decisión (aceptar, denegar o rechazar), o bien se salta a una nueva regla. Si analizamos el dibujo, tenemos los siguientes elementos:

- **Checksum**

  Control de integridad del paquete

- **Sanity**

  Control de coherencia del paquete

- **Input Chain**

  Cadena de filtros de entrada

- **Demasquerade**

  Funciones de desenmascaramiento de direcciones IP

- **Routing decision**

  Cálculo del enrutamiento necesario

- **Local Process**

  En algunos casos se puede implementar la política de routing mediante otros programas, en lugar de usar el kernel (por ejemplo: el programa socks). Este es el punto donde se insertan los comandos adecuados en la cadena de filtros

- **Loopback interface**

  Camino alternativo que siguen los paquetes que tienen origen y destino locales

- **Local**

  Punto donde se toma la decisión sobre si el paquete tiene destino local o debe hacerse forwarding

- **Forward chain**

  Cadena de filtros de control del forwarding

- **Output chain**

  Cadena de filtros de control de salida de paquetes

4.6.3.20. CONFIGURACIÓN DEL FIREWALL

firewall debe ser ejecutado desde el *xinted* y debe ser algo parecido a esto:

Configurarlo con la politica standard

ipchains - P forward DENY

Permitir solo la conexión destinada al servidor web

ipchains - A forward - j ACCEPT - p TCP - i eth0 - d 192.168.1.100 http

Permitir la respuesta del web server

ipchains - A forward - j ACCEPT - p TCP - i eth1 - s 192.168.1.100 ! - y

4.6.4 FIREWALL PUCESA

Para la ejecución del firewall es necesario instalar Java versión jdk1.2.2. Crear la siguiente carpeta en la raíz: Firewall dentro de esta carpeta las siguientes: Bin, Fuentes, Inagenes, Docs.

El directorio /Firewall/Bin contiene todos los programas compilados que serán ejecutados.

/Firewall/Fuentes posee los archivos fuentes que hemos programado para el desarrollo del Firewall.

/Firewall/Imágenes Este directorio contiene los gráficos que se presentan en el Firewall.

/Firewall/Docs La carpeta Docs contiene la documentación existente en el proyecto.

4.6.4.1 COMPILACIÓN DEL FIREWALL

CLASS: no basta poner el ***PATH*** o directorio en el que se encuentra. Por ejemplo, si se desea compilar y ejecutar el fichero Firewall.***java***, y éste necesitara la librería de clases ***/Firewall/Bin*** ***la*** forma de compilar y ejecutar sería:

javac –d ../bin Firewall.java

java Firewall

4.7 APLICACIÓN DEL SISTEMA EN EL SERVIDOR DE LA PUCESA

Al momento de instalar el Firewall en la PUCESA nos damos cuenta que por lo menos vamos a tener cierta garantía para no ser vulnerado.

Esta aplicación restringe ciertas direcciones o máquinas que no se desea que ingrese al servidor, por tal motivo nuestro Firewall es una herramienta de aplicación poderosa capaz de soportar cierta vulnerabilidad.

4.8 ANÁLISIS Y EVALUACIÓN DE LA IMPLEMENTACIÓN

El análisis y evaluación del firewall en el servidor de la PUCESA nos muestra muchas alternativas para ejecutarlo; debido a que la Universidad no posee cierta seguridad al momento de ingreso de datos del exterior, por tal motivo la aplicación que dejamos en la universidad garantiza el ingreso y envío de datos al exterior.

Para analizar el funcionamiento del Firewall realizamos múltiples pruebas llegando a la conclusión que todo sistema es vulnerable en el mundo, por tal motivo nuestro Firewall no garantiza en su totalidad ser seguro, pero por lo menos eficaz.

CONCLUSIONES

Al finalizar el desarrollo de esta disertación, hemos podido extraer las siguientes conclusiones:

- Todos los elementos del sistema se pueden obtener de manera gratuita, por lo que el montaje inicial del sistema y uso no conllevan a mayor inversión.

- La aplicación del software Firewall, constituye una alternativa contra intrusos, para así llenar ciertos vacíos de seguridad en la red de la Escuela de Ingeniería de Sistemas de la PUCESA, y redes de este tipo.

- El estudio y manejo de plataformas no tradicionales, como es el sistema operativo Linux, representa la aplicación de nuevas tecnologías rompiendo los paradigmas impuestos.

- Se ha escogido Java Versión JDK1.2.2 como programa base, porque a los usuarios nos permite trabajar con distintas plataformas es decir Windows y Linux, direcciones de memoria e instrucciones claras y precisas, y por encima de todo consigue un rendimiento óptimo para la ejecución en las plataformas antes mencionadas.

- El gestor es el jdk1.2.2, tiene la facilidad de correr en Linux como en Windows cuyas características principales es la velocidad y la robustez.

- La eficacia del lenguaje de programación en su estructura orientada a objetos, ya que esto facilita crear módulos independientes pero que se interelacionen entre sí.

- Los sistemas informáticos han evolucionado a pasos agigantados y los intrusos inventan nuevas formas de ingreso a redes.

- Con los lenguajes de programación los intrusos se han convertido en el medio más popular para vulnerar sistemas esto puede significar perdida para la empresa.

- La implantación de un sistema automatizado que es capaz de trabajar en un entorno de Intranet, con las mismas facilidades de hacerlo en el ámbito de Internet.

- La elaboración de un sistema para evitar el ingreso de intrusos hacia el centro de cómputo de la PUCESA, facilitará la protección de los componentes del laboratorio.

- De las pruebas realizadas con un equipo de desarrollo como el sistema operativo Linux y un equipo cliente con sistema operativo Windows el Firewall creó las reglas utilizando IPCHAIN cumpliendo su cometido. Se probó con diferentes puertos y protocolos.

- Un Firewall es una muy buena herramienta para evitar el ingreso de intrusos externos, pero no garantiza de que dentro de correos electrónicos por ejemplo puedan ingresar archivos tipo caballos de troya como virus y vulnerar el sistema.

- El comando IPCHAINS es una poderosa herramienta para ser administrada por usuarios expertos, pero con el Firewall que hemos desarrollado se facilita esta tarea.

- El comando IPCHAINS facilita la prevención de intrusos, bloqueando el ingreso al sistema

- El Firewall que hemos desarrollado nos provee la garantía de salvaguardar los datos de la PUCESA al 100%.

- Debido a la estructura del servidor SPARK de la PUCESA se recomienda implantar un servidor LINUX INTEL con la metodología Host Bastión.

- Al finalizar este trabajo podremos dar por cumplidos nuestros objetivos propuestos inicialmente, hemos realizado una investigación adquiriendo nuevos conocimientos perfeccionando la utilización de nuevas herramientas las cuales rompan con los paradigmas establecidos, que el mejor método de enseñanza es el ser autodidacta.

- Es prioritario la utilización del Internet como respuesta a la necesidad de la comunidad ávida de información, lo que permite enfrentar los retos tecnológicos de su desarrollo, acorde al nuevo milenio.

RECOMENDACIONES

Las recomendaciones que se presentan a continuación se debe considerar como resultado de un análisis profundo en el desarrollo del Firewall de la PUCESA.

- Este tipo de nuevas aplicaciones no puede ser restringidas sólo a un grupo de personas por lo que se recomienda que este sistema sea publicado en los portales de Linux para que de esta manera sirva de fuente de consulta para cualquier persona interesada en incursionar en el mundo del Software libre y como retribución a todos los usuarios de Internet de quienes hemos obtenido gran fuente de información y ayuda.

- Para este tipo de aplicaciones es necesario tener nociones de programación, ya que la programación en java jdk1.2.2 es un lenguaje estructurado orientado a objetos, distribuido y multitarea.

- Para el desarrollo de este tipo de aplicaciones es necesario que el computador posea Linux o Windows y java jdk1.2.2 para comenzar la programación.

- Aprovechar el trabajo teórico como herramienta de consulta, lo que permitirá el desarrollo de aplicaciones mucho más complejas.

- Se recomienda que la PUCESA utilice un servidor con plataforma Linux/Intel para que la interfase con el servidor Linux Sparc 64 bits posea seguridad óptima y pueda funcionar el Firewall.

- Para investigaciones posteriores se recomienda mejorar este Firewall para que funcione en múltiples plataformas ya que actualmente corre en Linux Intel o Windows, pero las reglas solo se ejecutan en Linux por lo que se usa el comando IPCHAINS.

ACRÓNIMOS

ARP: Address Resolution Protocol

DMZ: Zona Desmilitarizada

FTP: File Transfer Protocol

IAP: Internet Access Provider

ICMP: Internet Control Messaje Protocol

IP: Internet Protocol

NFS: Network File System

NIC: Network Information Center

NIS: Network Information Service

OSI: Organization Standars International

RPC: Remote Process Control

SGID: Switch Group Identification

SUID: Switch User Identification

TCP : Transmission Control Protocol

UDP: User Datagrama Protocol

VPN: Virtual Private Networks.

WWW: World Wide Web

GLOSARIO

Administrador: Persona que se encarga de todas las tareas de mantenimiento de un sistema informático.

ATAPI. Sigla de *AT Attachement Packet Interface* [interfaz de paquetes para conectar a AT]. ATAPI es el protocolo mediante el cual las unidades de CD-ROM se comunican con la computadora sobre la interfaz IDE. BIOS. Sigla de *Basic Input/Output System* [sistema de entrada/salida básico].

Backdoor: Puerta de entrada trasera a una computadora, programa o sistema en general. Sirve para acceder sin usar un procedimiento normal

Bajar o Download: Extraer un programa de un BBS vía módem.

Black Box: Aparato que engaña a la central telefónica haciéndole creer que no se levantó el teléfono cuando en realidad se está produciendo una comunicación

Blue Box: Aparato (o programa de computadora) que emite tonos multifrecuencias que permite controlar las centrales telefónicas. Se utiliza para lograr comunicaciones gratuitas, entre otras cosas.

Boxes: Circuitos preparados para realizar phreaking. Destacan:

- Bluebox => Para llamar gratis
- Redbox => Emula la introducción de monedas en teléfonos públicos
- Blackbox => El que llame a un teléfono con este dispositivo no pagará la llamada.

de virus, pero pueden instalarse troyanos que proporcionen passwords nuevos. También consiste en llevar una vida acorde con el hackmode.

Hackmode: Modo de actuar del hacker. No tiene por qué estar relacionado con las computadoras, es más bien un modo de interpretar la vida. Consiste en:

- No pagar lo que no es estrictamente necesario o pagar de forma "poco corriente".
- Ser un poco "paranoico".
- Actuar acorde con costumbres rigurosamente calculadas.

Handle: Seudónimo usado en vez del nombre verdadero.

IDE. Sigla de *I*ntegrated *D*rive *E*lectronics [electrónica de unidad integrada], que denota la interfaz estándar usada para conectar fundamentalmente unidades de disco y CD-ROM a un ordenador. Vea también «EIDE» y «ATAPI».

Intel. Compañía responsable de la producción de los microprocesadores más usuales en las computadoras personales compatibles con PC. Estos procesadores incluyen el 80386, 80486, Pentium, Pentium Pro, y Pentium II Pentium III Pentium IV.

Ingeniería social: Arte de convencer a la gente de entregar información que no corresponde.

Lamer: Tonto, persona con pocos conocimientos. Principiante

LAN (Local Area Network): Red de área local. Red de computadoras dispersas sobre un área relativamente pequeña. Muchas LANs están confinadas a un solo edificio o grupo de edificios. Sin embargo una LAN puede ser conectada a otras LANs mediante enlaces de radio o líneas telefónicas, a cualquier distancia.

Outdial: Modem de salida dentro de una misma red, que permite a un usuario de la misma salir a la red telefónica convencional. Los que permiten hacer llamadas a larga distancia se llaman 'global Outdial' (Outdial globales) o GOD.

PAM. Sigla de *P*luggable *A*uthentication *M*odules [Módulos enchufables de autenticación]. PAM es un sistema de autenticación que controla el acceso a Red Hat Linux.

Packet switching: Conmutación de paquetes.

Password: Clave. Palabra que sirve para verificar que un usuario es realmente quien dice ser. Por eso mismo, el único que debe conocerla es ese mismo usuario.

PBX: Private Branch Exchange. Centrales telefónicas internas de empresas

Pasarela En términos de redes, se refiere al dispositivo que conecta uno o más ordenadores de PCMCIA. Sigla de *P*ersonal *C*omputer *M*emory *C*ard *I*nternational *A*ssociation [Asociación Internacional Tarjetas de Memoria para Computadoras Personales].

Patch o Parche: Modificación de un programa ejecutable para solucionar un problema o para cambiar su comportamiento.

Payload: Efecto visible de un software maligno.

Petar: Anular. Este término se utiliza en el supuesto de que los sistemas utilizados para 'tracear' de un BBS, se hayan anulado o caducado.

PCMCIA. Disquete necesario para las instalaciones de Red Hat Linux que requieren el uso de un dispositivo PCMCIA durante la instalación. En inglés: «PCMCIA Support Diskette».

Phreaking: Acto de llamar por teléfono gratuitamente y la realización de modificaciones a los aparatos telefónicos con el fin de obtener algún tipo de beneficio.

Protocolo: Formato pre-acordado para la transmisión de datos entre dos dispositivos. Un protocolo determina lo siguiente: el tipo de chequeo de error a utilizar; el método de compresión de datos, si existiese; cómo el dispositivo que envía datos informa que ha finalizado el mensaje; y cómo el dispositivo que recibe el mensaje indica que lo ha recibido satisfactoriamente.

Permisos. El conjunto de identificadores que controlan el acceso a los ficheros.

PLIP. Sigla de *P*arallel *L*ine *I*nternet *P*rotocol [Protocolo de Internet para líneas paralelas]. PLIP es un protocolo que permite comunicaciones TCP/IP sobre el puerto paralelo de la computadora, mediante el uso de un cable especialmente diseñado.

Raíz. (N. del T.) Traducción de la palabra «*root*». En determinados contextos se usa en castellano (ej.: «el directorio raíz»).

Root (raíz) El nombre de la cuenta de ingreso que da acceso completo y total a todos los recursos del sistema. También se usa para describir el directorio denominado con «/», como en la expresión «el directorio raíz».

Router: Dispositivo capaz de conectar un determinado número de LANs. Alpha La computadora de arquitectura RISC (*R*educed *I*nstruction *S*et *C*omputer) [Computadora con juego de instrucciones reducido] desarrollada por Digital Equipment Corporation.

Arranque dual. El acto de configurar un ordenador para que pueda arrancar más de un sistema operativo. En inglés: «*Dual Boot*».

Servidor Proxy: Es un servidor que se sitúa entre una aplicación (programa) cliente, como un browser de web, y el servidor original. Intercepta todos los pedidos hacia el servidor

real, para ver si puede satisfacerlos por si mismo; sinó se encarga de efectuar el pedido al servidor real y lo reenvía al cliente. Los servidores proxy tienen dos funciones principales: mejorar la performance (pedido de información local), y filtrar los pedidos de los clientes.

Subir o Upload: Enviar un programa a un BBS vía módem.

Setgid. Llamada al sistema que puede usarse para asignar el GID de un proceso.

Setuid. Llamada al sistema que se usa para asignar el UID de un proceso.

SILO. Cargador que se usa generalmente para sistemas Linux basados en el procesador.

Tracear: Seguimiento exhaustivo. Se utiliza cuando se intenta desproteger un programa y se tiene instalado un Debugger. Este término también es utilizado en caso de que la línea telefónica esté pinchada por la policía.

Trader: Persona que 'sube' y 'baja' continuamente programas y juegos de BBS.

UID. Abreviatura de *User ID* [ID de usuario]. Es el medio por el cual se identifica a un usuario en las distintas partes de un sistema Red Hat Linux.

UNIX.Conjunto de sistemas operativos del estilo de Linux.

Virii: Suele encontrarse en textos en inglés. Es la acción de crear virus.

Warez: Programas comerciales ofrecidos gratuitamente. Lo que se conoce popularmente como "pirateo".

Widget. Representación estandarizada en pantalla de un control que el usuario puede manipular. Ejemplos de «widgets» son las barras de desplazamiento, los botones y las cajas de texto.

X Window System. [Sistema de ventanas X] También denominado «X», esta interfaz gráfica de usuario proporciona la bien conocida metáfora de «ventanas sobre un escritorio», común a la mayoría de los sistemas hoy en día. Bajo X, los programas de aplicación actúan como clientes y acceden al servidor X que gestiona toda la actividad en pantalla.

BIBLIOGRAFIA

TEXTOS

Linux Red Hat Linux

Firewall and Internet Security (Adison Wesley)

Manual Básico de Linux (Leopoldo vazques)

Aprenda JAVA como si estuviera en Primero (Javier Garcia de Jalón)

REFERENCIAS WEB DEDICADAS A SEGURIDADES

VULNERABILIDAD Y PROTECCIÓN

- [RefKoasp] http: //www.koasp.com

SEGURIDAD EN EL INTERNET

- [RefSupernet] http://www.supernet.net/cwsapps/cwsa.html
- [RefAlpworld] http://www.alpworld.com/infinity/void

SEGURIDAD EN EL CORREO ELECTRÓNICO

- [Ref3com] http://www.correomemo.com.co/politicas_de_seguridad.htm
- [Ref3com] http://www.iclave.com/politica.asp.

INTRUSOS

- [RefGtri] http://www.gtri.gatech.edu/res-news/rchnews.html
- [RefKriesgam] http://www.kriesgam.com/
- [RefCrysof] http://www.crysoft.com/cursos/cursos.htm

TIPOS DE ATAQUES

- [RefWintest] http://www.windows2000test.com/"

FIREWALLS

- [RefNIgob] http://www.nl.gob.mx/pagina/Enlaces/ciapem/prototipo/firewall/tsld002.htm
- [RefSisco] www.cisco.com
- [RefScrc] www.scrc.ncsl.nist.gov
- [RefHp] www.hp.com
- [RefNetsearch] www.icsa.com

INSTALACIÓN DE LINUX RED HAT 6.0

- http://lucas.hispalinux.es/Articulos-periodisticos/jantonio/ipchains/ipchains.html

IPCHAINS

- www.rustcorp.com/linux/ipchain
- http://www.ibiblio.org/mdw/HOWTO/IPCHAINS-HOWTO.html