UÍDE

UNIVERSIDAD INTERNACIONAL DEL ECUADOR

FACULTAD DE CIENCIAS Y TECNOLOGÍAS APLICADAS

Tesis de grado para la obtención del título de

Ingeniero en Informática y Multimedia

TÍTULO

DESARROLLO DE UN FIREWALL PERSONALIZADO CON OPCIÓN DE AUTO-RESPALDO DE DATOS HACIA LA NUBE INFORMÁTICA PARA LAS PYMES

ALUMNO

Gabriel David Silva Donoso

DIRECTOR

Rubén Torres

Enero de 2015

Guayaquil - Ecuador

Yo, **GABRIEL DAVID SILVA DONOSO**, declaro que soy el autor exclusivo del presente trabajo, y que éste es auténtico y personal mío. Todos los efectos académicos y legales que se desprendan del presente trabajo serán de mi exclusiva responsabilidad.

Guayaquil, Enero 19 del 2015

(F)_____

GABRIEL DAVID SILVA DONOSO
C.I. 0918214511

Yo, Rubén Torres Ortega, declare que, en lo que personalmente conozco, el señor, Gabriel David Silva Donoso, es el autor exclusivo del presente trabajo y que éste es original, auténtico y personal suyo.

(F)_____

Rubén Torres
Director Técnico de Trabajo de Grado
C.I. 0909454696

AUTORIDADES DE LA UNIVERSIDAD INTERNACIONAL DEL ECUADOR

En la ciudad de Guayaquil, a los 19 días del mes Enero de 2015, se subscribe la siguiente acta de Defensa de Grado, del estudiante, GABRIEL DAVID SILVA DONOSO, de la carrera de Ingeniería en Informática y Multimedia, siendo las principales autoridades: el Ing. Xavier Fernández Orrantia, Rector de la Universidad Internacional del Ecuador, Econ. Ramiro Canelo Salazar, Vicerrector Financiero y Marisol Bermeo Valencia, Vicerrector Académico de la Universidad Internacional del Ecuador y el Ab. Aldo Maino Isaías, Director Ejecutivo-Extensión Guayaquil. Para lo cual doy fe.

Ab. Aldo Maino Isaías
Director Ejecutivo – Extensión Guayaquil

EXTENSIÓN GUAYAQUIL
UNIVERSIDAD INTERNACIONAL DEL ECUADOR

MIEMBROS DEL TRIBUNAL DE GRADO

Miembro Principal

Miembro Principal

Miembro Principal

Damos fe de la elaboración de este Trabajo de Grado, que fue presentado en la fecha: 19 de Enero de 2015.

AGRADECIMIENTO

Agradezco al Creador y Padre Celestial por darme la mejor herencia que un hijo puede tener en esta vida, la educación. A la Universidad Internacional del Ecuador, prestigiosa institución que dotó de sus mejores profesionales quienes me prepararon durante toda esta larga trayectoria para crecer profesionalmente; finalmente quiero dar un agradecimiento especial a todas las personas que no creyeron en mí, ya que de no ser por ellas, no hubiera tenido el reto de culminar esta bella etapa de mi vida y demostrarles que todo es posible cuando uno se lo propone con actitud.

GABRIEL DAVID SILVA DONOSO

DEDICATORIA

- Al Padre Celestial por darme las fuerzas necesarias para realizar este proyecto ya que me ha dado la fortaleza necesaria para avanzar a pesar de las dificultades presentadas en el desarrollo del mismo.

- A mí amada esposa Evelyn Plaza y tía materna Raquel Donoso por vuestra guía terrenal y paciencia infinita, por incondicionalmente haberme apoyado en todo este tiempo, les dedico todo mi esfuerzo puesto para la realización de esta tesis.

- A mi hija Arianne Silva, aunque aún no has nacido, espero llegue pronto ese día, crezcas y caminemos juntos por la vida para contarte todas mis vivencias.

GABRIEL DAVID SILVA DONOSO

RESUMEN

Se escogió el tema, el cual su implementación fue enfocada sólo a las empresas nacionales del Ecuador ya que, de acuerdo a un artículo publicado en periódico local del Ecuador[1], el Ecuador está en el puesto 108 de 138 países en cuanto a Tecnología Informática, según el análisis de reporte Global de Tecnología 2012-2013. Sumado a esto, se tiene las empresas que han sido expuestas a ataques de hackers informáticos dentro o fuera de la misma y sobre todo la saturación interna de la red por exceso de vulnerabilidades. Esto indirectamente impacta a la organización y crecimiento de las empresas nacionales. Se propuso proveer un servicio Firewall que permita tener un acceso seguro a usuarios, servicio redundante y disponible de la información de la empresa, organizando su más importante recurso: el informático, incluyendo una automatización de respaldo desde un servidor local hacia otro que está físicamente localizado en el extranjero (Alemania).

[1] "Diario Hoy", 24 de Marzo de 2014

Índice general

Índice de gráficos

Capítulo 2

Capítulo 3

Capítulo 4

Índice de anexos

Palabras clave

Pymes, Firewall, Tecnología informática, Nube informática, Vulnerabilidades, Ataques informáticos, Respaldo de información, Hackers, Software libre, Filtrado de paquetes, Zona desmilitarizada, Saturamiento de canal, Puertos lógicos de la computadora.

CAPÍTULO 1

1.1. Antecedentes

Con el advenimiento de Internet y la necesidad de automatizar tareas y procesos la computadora se ha transformado en la herramienta administrativa por excelencia en las Pequeñas y Medianas Empresas. Desde el software fiscal hasta el procesador de palabras; todo pasa por la computadora, de ahí que toda institución sea de mayor o menor tamaño, está en la necesidad de incorporar a su forma de trabajo, una plataforma informática, la cual debe ajustarse a la misión de la empresa y ofrezca las herramientas necesarias para el correcto desempeño de la misma de manera segura sin que se tenga la opción de que la información sensible sea vulnerable.

1.2. Planteamiento del Problema

Al tener una conexión directa a Internet sin restricciones, se tiene el riesgo de sufrir intromisiones de ataques de hackers informáticos que pondrían en riesgo la integridad del software, la información confidencial de la empresa y los datos confidenciales de los clientes de esta, a quienes se les debe brindar un servicio de calidad y seguridad. Empiezan presentado problemas de lentitud en los correos electrónicos, comportamientos inesperados en las computadoras o servidores, en la navegación a internet y en la mayoría de los casos, estas computadoras y servidores suelen llegar a perder información valiosa. La base de datos de sus sistemas contable, sufren comúnmente un daño irreparable y esta información siempre termina siendo recreada o restaurada perdiendo información de los últimos días que se ha considerado un respaldo. Un Firewall es un dispositivo que filtra el tráfico entre redes, como mínimo dos, el Firewall puede ser un dispositivo físico o un software sobre un sistema operativo. En general debemos verla como una caja con dos o más interfaces de red en la que se establecen unas reglas de filtrado con las que se decide si una conexión determinada puede establecerse o no. Incluso puede ir más allá y realizar modificaciones sobre las comunicaciones, como el servicio NAT (traducción de direcciones de red por sus siglas en Inglés).

1.3. Justificación

1.3.1. Impacto empresarial

Las empresas se verán afectadas de una manera positiva debido a que van a tener una productividad ininterrumpida por el hecho de no tener saturación en la red local de computadoras y servidores, seguridad en su información valiosa la cual no será expuesto ante los atacantes o hackers informáticos y adicionalmente pueden tener un control de cuándo deben hacer una actualización de versión en sus aplicativos internamente.

1.3.2. Impacto Social

El desarrollo de este proyecto de tesis va orientado al objetivo 11 del manual del buen vivir el cual es asegurar la soberanía y eficiencia de los sectores estratégicos para la transformación industrial y tecnológica. El Ecuador tiene una oportunidad histórica para ejercer soberanamente la gestión económica, industrial y científica, de sus sectores estratégicos. Esto permitirá generar riqueza y elevar en forma general el nivel de vida de nuestra población. Para el Gobierno de la Revolución Ciudadana, convertir la gestión de los sectores estratégicos en la punta de lanza de la transformación tecnológica e industrial del país, constituye un elemento central de ruptura con el pasado.

1.3.3. Impacto académico

Si se implementa un Firewall en las computadoras de las instituciones educativas, estos serán beneficiados ya que no tendrán que incurrir en costo mensual de servicio técnico para desinfectar o aun peor, reinstalar el sistema operativo por ataques de virus informáticos. El mismo proyecto quedará como donación a la ciencia para que puedan continuar con mejoras del mismo.

1.4. Objetivo general

Proteger la red interna de las empresas PYMES de ataques internos y externos mediante la implementación de un Firewall sobre plataforma de un sistema operativo de bajo costo pero con atributos y beneficios robustos.

1.5. Objetivos específicos

- Analizar la situación actual y los requerimientos de seguridad informática de una empresa PYME.

- Determinar las diferentes formas de ataques maliciosos que sufren las redes LAN en general y por ende la de la empresa en mención.

- Aprovechar la tecnología existente en el mercado sobre Firewalls.

- Analizar las diferentes alternativas de solución para proteger a la empresa.

- Definir políticas de seguridad de red en base a los programas y servicios con los que la empresa cuenta.

- Implementar la mejor solución que se adapte a las necesidades de una empresa PYME.

CAPÍTULO 2

2.1. Marco Referencial

2.1.1. Situación actual en el Ecuador

En el Ecuador, existen muchas empresas que están en proceso de desarrollo y crecimiento pero hay algunas que aún no han logrado adaptarse a una cultura de organizar toda su información apropiadamente sobre servidores, mediante el uso de la tecnología informática y utilizando una infraestructura segura con redundancia y disponibilidad de información inmediata ante un caso de desastre real o ante un caso de ataque de hackers informáticos. Por falta de esto, se genera una considerable pérdida de tiempo, costo de oportunidad y hasta en ciertos casos pérdida de dinero conllevando a múltiples implementaciones de hardware adicional costoso. Adicionalmente debido a:

1. **La carencia de almacenamiento sobre un servidor seguro:** Comúnmente la mayoría de los administradores de sistemas optan por hacer que los usuarios guarden su información sobre los discos duros de sus computadoras. La falta apropiada de almacenamiento de información centralizada y segura sobre un servidor que contenga toda la información vital de una empresa como archivos de presupuesto anual, roles de pagos de los empleados, carpetas o recursos compartidos, permisos de usuarios sobre una carpeta o recurso compartido, ausencia de un control de acceso y seguridad hacia la computadora.

2. **Adquisición costosa de licenciamiento de software y contratación de múltiple servicio técnico:** Para atender diferentes actividades informáticas dentro de la compañía, tales como configuración de los accesos de recursos compartidos a usuarios, configuración de sistemas físicos que están obligados a usar licenciamiento para asegurar el acceso autorizado a los usuarios, bloqueo y denegación de información a los intrusos o usuarios no autorizados, quienes una vez que hayan penetrado en su servidor se verán

obligados a sustraer o adulterar la información delicada de la empresa, en casos comunes la opción de instalar software de terceros para evitar esto.

2. **Ausencia de respaldo de la información:** El daño en los sistemas de información bancarios, afecta a varias áreas como: el sistema de gestión de caja que puede estar fuera de línea, generando incomodidad por parte del banco a clientes, debido a que el último respaldo de transacciones que se tiene, es del mes pasado.

3. Un corte de energía inesperado y la ausencia de información de la nómina de empleados para pago de sus sueldos. En ambos casos, comúnmente se genera pérdida de tiempo y en algunas ocasiones, pérdida de dinero.

4. **Vulnerabilidad de la información**: Por la ausencia de un sistema firewall personalizado que proteja toda la red privada de datos de la empresa contra los atacantes informáticos ya sea interno como externos.

2.2. Marco Legal

Delitos informáticos contemplados en la Ley Ecuatoriana

Según (Juan Mendez, 2013) Es acción antijurídica, ilegal, culpable o dolosa, sancionada con una "pena", según la gravedad de la misma. Inmemorablemente, siempre, ha sido castigada y aquellos que lo cometieron, se los denomina "delincuentes" en general; en particular, asesino porque quitó la vida a otro ser humano, violador, violó a otro ser humano y así sucesivamente. La presencia y proceso de las nuevas TIC´s en la Sociedad de la Información y Comunicación SIC, ha dado lugar al surgimiento de nuevas actividades y figuras jurídicas legales e ilegales. Insertas en éstas se encuentran, justamente, el delito informático, ¿en qué consiste? Si trasladamos la definición anterior a aquel, será el delito o crimen informático; piratería virtual, apropiación de algo ajeno con fines de lucro; spam, distribución de correo con avisos publicitarios, todos realizados a través de hardware y software, para vulnerar, dañar o destruir o invadir un sistema de propiedad ajena, sea empresa, gobierno o personal, pudiendo abarcar asuntos relacionados con la Información, comunicación personal, actividades económicas, funcionamiento con

Internet, debiéndose añadir que, al cometimiento de un delito informático se viola lo que es más privativo de un ser humano como son los bienes intangibles amparados por el Derecho de Propiedad al manipularse sus datos personales y privados considerando que desde un nombre de dominio, una creación intelectual de cualquier índole, intimidad personal, dirección virtual o convencional, cometimiento de fraude, conlleva la seguridad jurídica en la red. Y, según la gravedad se los clasifica en delitos relacionados por el contenido en sabotajes informáticos, infracciones a los derechos de la propiedad intelectual y afines, así: Phishing, muy conocido en nuestro medio, especialmente, por el perjuicio ocasionado a funcionarios públicos y que ascendieron en un aproximado a US$ 6´000.000,oo. Consiste en el envío de correos electrónicos que, aparentando originarse de fuentes fiables, ejemplo, entidades bancarias, intentan obtener datos confidenciales del usuario, valiéndose de un enlace que, al ser pulsado, lleva a páginas web falsas o falsificadas. Tampering o data diddling, modificación desautorizada de datos o al software de un sistema llegándose, incluso, a borrar cualquier información. Scanning, escudriña el contenido de un libro, periódico, en busca de algo especial para sus intereses. Pharming o cambiazo, táctica fraudulenta en los contenidos del servidor de nombres de dominio, ya sea a través de la configuración del protocolo IP o del archivo, para redirigir a los navegadores a páginas web falsas en lugar de las auténticas cuando el usuario accede a las mismas. Skimming, en lo negativo es la técnica delictiva que utiliza tecnología avanzada y facilita al ladrón o hacker robar las claves personales de los cajeros sin necesidad de estar presente, utilizando un dispositivo electrónico diseñado para este fin. Cuando el usuario se aleja, el delincuente ingresa y carga los datos en un sistema con el que puede leerlos y, posteriormente, introducirlos en una tarjeta con banda magnética sin uso, facilitándole hacer una tarjeta clon y procede a estafar. Otros también tipificados son el Tampering o Data diddling, modificación desautorizada de datos personales o al software instalado en un sistema; Sniffing, roba información de un terminal específico o de una red por medio de un apartado o cable que cumple funciones de espía; el Anonimato, referente a la habilidad de ocultar la identidad de las personas durante el uso de la red internacional de datos o páginas que se visitan por medio de servidores especializados o programas de cómputo que muestran una dirección IP que no corresponde con el equipo utilizado. Existen muchísimos otros definidos desde la legislación de Naciones Unidas.

Los sujetos o personas que realizan o acometen los delitos informáticos, según la actividad que hayan efectuado, son los Hackers, Script Kiddies o criminales informáticos, que "aprovechan sus conocimientos (experto) de la informática (redes, programación, etc.) para utilizar la vulnerabilidad de un sistema con un fin: obtener información privada. Existen muchos tipos, por ejemplo hacker de sombrero blanco o sombrero negro. El del sombrero blanco seria que avisa del peligro de un posible atentado en la red informática. El otro, lo usará con fines maliciosos"

Según (Felipe Sosa, 2013) "Crackers o vandálico virtual, programadores maliciosos", son individuos de la sociedad moderna que poseen conocimientos avanzados en el área tecnológica e informática, igual que los Hackers, invaden sistemas, descifran claves y contraseñas de programas, algoritmos de encriptación, roban datos personales, destruyen y cuando crean algo es únicamente para fines personales, son extremadamente precavidos con el manejo de la información, precisamente, para ocasionar el daño inmoral e ilegal a los sistemas informáticos. Pirata informático es quien adopta por negocio la reproducción, apropiación y distribución, con fines lucrativos, y a gran escala, a través de distintos medios y contenidos de software, videos, música, de los que no posee licencia o permiso de su autor, generalmente haciendo uso de una computadora. Siendo la de software la práctica de piratería más conocida, por ello se los clasifica como: Piratas de software, de música, de videos-juegos, de películas, de libros o artículos, todo lo cual tiene que ver con los derechos de Propiedad Intelectual. Spammers, persona o grupos dedicados a la distribución de correos electrónicos no deseados a usuarios o empresas, por lo cual, son combatidos. Esta actividad es sumamente lucrativa, y en la gran mayoría de legislaciones se la considera ilegal. Con esta ligera y breve explicación de los delitos informáticos mediante conceptos generales y universales originados desde las mismas Naciones Unidades, cuya comisión especializada, UNCITRAL o CNUDMI, elaboró la ley modelo y Ecuador la internalizó mediante la Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos, R.O. 557 de 17-abril-2002, complementado con el reglamento, -diciembre-02-, constando en el capítulo II de las Infracciones Informáticas, artículos 57 en adelante, sancionando o penalizando a los mismos, reformaron a los artículos 202, 262, 353, 415, 553, 563, 606 #19º del Código Penal del Ecuador, que en la ley especial corresponde a los siguientes artículos: 58, 59, 60, 61, 62, 63 y 64.

¿Qué se legisla? Conductas ilícitas, acceso ilegal a sistemas informáticos, interceptación ilegal de las comunicaciones, daños en sistemas informáticos, fraude electrónico, fraude en las telecomunicaciones, entre otros. En el siguiente cuadro se describe tanto las infracciones, la pena carcelaria y pecuniaria: (Alberto Espinoza, 2013)

ART. 58: DELITOS CONTRA LA INFORMACIÓN PROTEGIDA (art.202 CP):	***SANCIÓN CARCELARIA***	***SANCIÓN PECUNIARIA***
1.- Violentando claves o sistemas	6 meses a un año	US$ 500.- a US$ 1.000.-
2.- Información obtenida sobre la Seguridad nacional, secretos comerciales o industriales:	3 años	US$ 1.000.- a US$ 1.500.-
3.- Divulgación o utilización fraudulenta de los rubros anteriores:	3 a 6 años	US$ 2.000.- a US$ 10.000.-
4.- Divulgación o utilización por funcionarios a cargo de dicha información.	6 a 9 años	US$ 2.000.- a US$ 10.000.-
5.- Obtención y uso no autorizados de datos personales para cederla o utilizarla :	2 meses a 2 años	US$ 1.000.- a US$ 2.000.-
ART. 59: DESTRUCCION MALICIOSA DE DOCUMENTOS POR FUNCIONARIOS DE SERVICIO PÚBLICO (Art. 262 CP)	3 A 6 AÑOS	xxx

ART. 60: FALSIFICACIÓN ELECTRÓNICA SEGÚN EL SIGUIENTE DETALLE Y CON ÁNIMO DE LUCRO CON PERJUICIO A TERCEROS (Art. 353 CP):	Serán juzgados de acuerdo a lo que se dispone en este capítulo, o sea, 6 años	xxx
1.- Alterar un mensaje de datos		
2.- Simulación de mensaje.		
3.- Suposición de intervención en actos, declaraciones, etc.		
ART. 61: DAÑOS INFORMÁTICOS (Art. 415 CP) SEGÚN:		
1.- Daño doloso de información contenida en un sistema.	6 meses a 3 años	US$ 60,oo a US$ 150,oo
2.- Cometido por funcionario público o vinculado a la defensa nacional.	3 a 5 años	US$ 200.- a US$ 600.-
3.- Si no se tratare de un delito mayor, la destrucción, alteración o inutilización de infraestructura para la transmisión.	8 meses a 4 años	US$ 200.- a US$ 600.-

ART. 62: APROPIACIÓN ILÍCITA (Art. 553 CP) SEGÚN LO SIGUIENTE: 1.- Uso fraudulento o ilícito para apropiación de un bien ajeno, etc.	6 meses a 5 años	US$ 500.- a US$ 1.000.-
2.- Uso fraudulento mediante la utilización de los siguientes medios: 1) Inutilización de sistemas de alarma o guarda; 2) Descubrimiento descifrado de claves secretas o encriptadas; 3) de tarjetas magnéticas, carding o perforadas; 4) de controles o instrumentos de apertura a distancia y 5) violación de seguridades electrónicas u otras semejantes	Uno a cinco años	US$ 1.000.- a US$ 2.000.-
ART. 63 ESTAFA (ART.363 CP) A TRAVÉS DE MEDIOS ELECTRÓNICOS	Uno a cinco años	US$ 500.- a US$ 1.000.-
ART. 64 DERECHO A LA INTIMIDAD (art.606 #19ºCP) Si no fuere delito	La sanción aún está en sucres, equivalente a casi centavos.	De dos a cuatro días

Fuente: Informática hoy (Elvis Proaño, 2012)

2.3. Marco Teórico

2.3.1. Redes de computadoras

Una red de computadoras, llamada también red de comunicaciones de datos o red informática, es un conjunto de equipos informáticos y software conectados entre sí por medio de dispositivos físicos que envían y reciben impulsos eléctricos, ondas electromagnéticas o cualquier otro medio para el transporte de datos, con la finalidad de compartir información, recursos y ofrecer servicios. La finalidad principal para la creación de una red de computadoras es compartir los recursos y la información en la distancia, asegurar la confiabilidad y la disponibilidad de la información, aumentar la velocidad de transmisión de los datos y reducir el costo general de estas acciones. Un ejemplo es Internet, la cual es una gran red de millones de computadoras ubicadas en distintos puntos del planeta interconectadas básicamente para compartir información y recursos. (Oliva Marcillo, 2012)

2.3.2. Topología de redes

La topología de red se define como la cadena de comunicación usada por los nodos que conforman una red para comunicarse.

Según el autor (Aurelio Silvetty, 2012), expresa un ejemplo claro de esto es la topología de árbol, la cual es llamada así por su apariencia estética, por la cual puede comenzar con la inserción del servicio de internet desde el proveedor, pasando por el router, luego por un switch y este deriva a otro switch u otro router o sencillamente a los hosts (estaciones de trabajo), el resultado de esto es una red con apariencia de árbol porque desde el primer router que se tiene se ramifica la distribución de internet dando lugar a la creación de nuevas redes o subredes tanto internas como externas. Además de la topología estética, se puede dar una topología lógica a la red y eso dependerá de lo que se necesite en el momento. En algunos casos se puede usar la palabra arquitectura en un sentido relajado para hablar a la vez de la disposición física del cableado y de cómo el protocolo considera dicho cableado. Así, en un anillo con una MAU se puede decir que se tiene una topología en anillo, o que se trata de un anillo con topología en estrella. La topología de red la determina únicamente la configuración de las conexiones entre nodos. La distancia entre los nodos, las interconexiones físicas, las tasas de transmisión y los tipos de señales no pertenecen a la topología de la red, aunque pueden verse afectados por la misma.

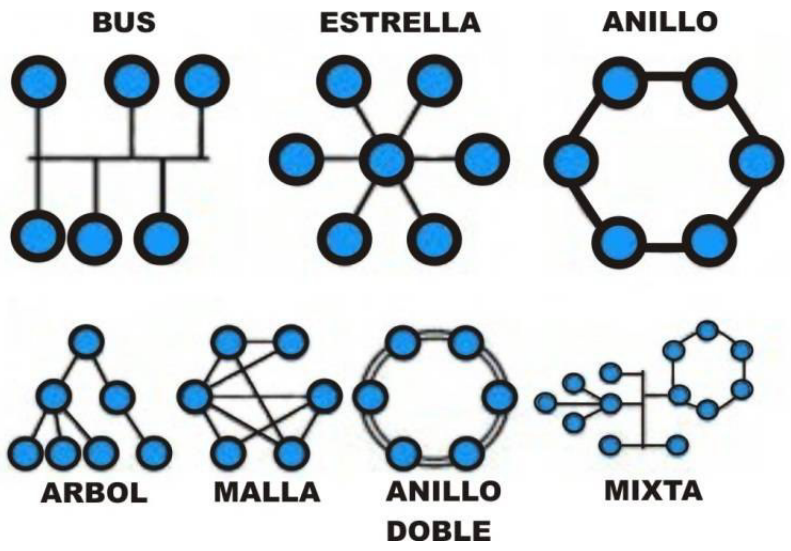
2.3.3. Tipos de arquitecturas

La topología en estrella reduce la posibilidad de fallo de red conectando todos los nodos a un nodo central. Cuando se aplica a una red basada en la topología estrella este concentrador central reenvía todas las transmisiones recibidas de cualquier nodo periférico a todos los nodos periféricos de la red, algunas veces incluso al nodo que lo envió. Todos los nodos periféricos se pueden comunicar con los demás transmitiendo o recibiendo del nodo central solamente. Un fallo en la línea de conexión de cualquier nodo con el nodo central provocaría el aislamiento de ese nodo respecto a los demás, pero el resto de sistemas permanecería intacto. El tipo de concentrador hub se utiliza en esta topología, aunque ya es muy obsoleto; se suele usar comúnmente un switch. La desventaja radica en la carga que recae sobre el nodo central. La cantidad de tráfico que deberá soportar es grande y aumentará conforme se vayan agregando más nodos periféricos, lo que le hace poco recomendable para redes de gran tamaño.

Además, un fallo en el nodo central puede dejar inoperante a toda la red. Esto último conlleva también una mayor vulnerabilidad de la red, en su conjunto, ante ataques. Si el nodo central es pasivo, el nodo origen debe ser capaz de tolerar un eco de su transmisión. Una red, en estrella activa, tiene un nodo central activo que normalmente tiene los medios para prevenir problemas relacionados con el eco.

Una topología en árbol, también conocida como topología jerárquica, puede ser vista como una colección de redes en estrella ordenadas en una jerarquía. Éste árbol tiene nodos periféricos individuales, por ejemplo hojas, que requieren transmitir a y recibir de otro nodo solamente y no necesitan actuar como repetidores o regeneradores. Al contrario que en las redes en estrella, la función del nodo central se puede distribuir. Para aliviar la cantidad de tráfico de red que se necesita para retransmitir en su totalidad, a todos los nodos, se desarrollaron nodos centrales más avanzados que permiten mantener un listado de las identidades de los diferentes sistemas conectados a la red. Éstos switches de red "aprenderían" cómo es la estructura de la red transmitiendo paquetes de datos a todos los nodos y luego observando de dónde vienen los paquetes respuesta. (Javier Heinz, 2013)

Gráfico 2. 1: Topología de redes



Fuente: (Groth & Skandier, 2005)

2.3.4. Modelo OSI

Para el autor (ZIMMERMAN, 1980), el modelo de interconexión de sistemas abiertos, también llamado OSI (en inglés Open System Interconnection) es el modelo de red descriptivo creado por la Organización Internacional para la Estandarización en el año 1984, una federación global de organizaciones que representa aproximadamente a 130 países. El núcleo de este estándar es el modelo de referencia OSI, una normativa formada por siete capas que define las diferentes fases por las que deben pasar los datos para viajar de un dispositivo a otro sobre una red de comunicaciones. Siguiendo el esquema de este modelo se crearon numerosos protocolos. El advenimiento de protocolos más flexibles donde las capas no están tan desmarcadas y la correspondencia con los niveles no era tan clara puso a este esquema en un segundo plano. Sin embargo es muy usado en la enseñanza como una manera de mostrar cómo puede estructurarse una "pila" de protocolos de comunicaciones. El modelo especifica el protocolo que debe ser usado en cada capa, y suele hablarse de modelo de referencia ya que es usado como una gran herramienta para la enseñanza de comunicación de redes.

Se trata de una normativa estandarizada útil debido a la existencia de muchas tecnologías, fabricantes y compañías dentro del mundo de las comunicaciones, y al estar en continua expansión, se tuvo que crear un método para que todos pudieran entenderse de algún modo, incluso cuando las tecnologías no coincidieran. De este modo, no importa la localización geográfica o el lenguaje utilizado. Todo el mundo debe atenerse a unas normas mínimas para poder comunicarse entre sí. Esto es sobre todo importante cuando hablamos de la red de redes, es decir, Internet.
Este modelo está dividido en siete capas:

2.3.5. Capa física

Según (Janeth Paez, 2012) es la que se encarga de las conexiones físicas de la computadora hacia la red, tanto en lo que se refiere al medio físico como a la forma en la que se transmite la información. Sus principales funciones se pueden resumir como:

- Definir el medio o medios físicos por los que va a viajar la comunicación: cable de pares trenzados o no, como en RS232/EIA232, coaxial, guías de onda, aire, fibra óptica.
- Definir las características materiales, componentes y conectores mecánicos y eléctricas (niveles de tensión) que se van a usar en la transmisión de los datos por los medios físicos.
- Definir las características funcionales de la interfaz establecimiento, mantenimiento y liberación del enlace físico.
- Transmitir el flujo de bits a través del medio.
- Manejar las señales eléctricas del medio de transmisión, polos en un enchufe, etc.
- Garantizar la conexión, aunque no la fiabilidad de dicha conexión.

2.3.6. Capa de enlace de datos

Esta capa se ocupa del direccionamiento físico, de la topología de la red, del acceso al medio, de la detección de errores, de la distribución ordenada de tramas y del control del flujo. Por lo cual es uno de los aspectos más importantes a revisar en el momento de conectar dos computadoras, ya que está entre la capa 1 y 3 como parte esencial para la creación de sus protocolos básicos (MAC, IP), para regular la forma de la conexión entre computadoras así determinando el paso de tramas (trama = unidad de medida de la información en esta capa, que no es más que la segmentación de los datos trasladándolos por medio de paquetes), verificando su integridad, y corrigiendo errores, por lo cual es importante mantener una excelente adecuación al medio físico, los más usados son el cable UTP, par trenzado o de 8 hilos, con el medio de red que redirecciona las conexiones mediante un router. Dadas estas situaciones cabe recalcar que el dispositivo que usa la capa de enlace es el Switch que se encarga de recibir los datos del router y enviar cada uno de estos a sus respectivos destinatarios, servidor - computadora cliente o algún otro dispositivo que reciba información como celulares, etc., dada esta situación se determina como el medio que se encarga de la corrección de errores, manejo de tramas, protocolización de datos se llaman protocolos a las reglas que debe seguir cualquier capa del modelo OSI. (Sergio Mendez, 2012)

2.3.7. Capa de red

Se encarga de identificar el enrutamiento existente entre una o más redes.[2] Las unidades de información se denominan paquetes, y se pueden clasificar en protocolos enrutables y protocolos de enrutamiento:

- Enrutables: viajan con los paquetes (IP, IPX, APPLETALK).
- Enrutamiento: permiten seleccionar las rutas (RIP, IGRP, EIGRP, OSPF, BGP).

El objetivo de la capa de red es hacer que los datos lleguen desde el origen al destino, aun cuando ambos no estén conectados directamente. Los dispositivos que facilitan tal tarea se denominan encaminadores, aunque es más frecuente encontrarlo con el nombre en inglés routers. Los routers trabajan en esta capa, aunque pueden actuar como switch de nivel 2 en determinados casos, dependiendo de la función que se le asigne. Los firewalls actúan sobre esta capa principalmente, para descartar direcciones de máquinas. En este nivel se realiza el direccionamiento lógico y la determinación de la ruta de los datos hasta su receptor final. (Castells, 1997)

2.3.8. Capa de transporte

Capa encargada de efectuar el transporte de los datos, que se encuentran dentro del paquete de la máquina origen a la de destino, independizándolo del tipo de red física que se esté utilizando. La PDU de la capa 4 se llama Segmento o Datagrama, dependiendo de si corresponde a TCP o UDP. Sus protocolos son TCP y UDP; el primero orientado a conexión y el otro sin conexión. Trabajan, por lo tanto, con puertos lógicos y junto con la capa red dan forma a los conocidos como Sockets IP:Puerto (191.16.200.54:80).

[2] Como estándar del modelo OSI, es la actividad principal de un dispositivo enrutador sea cual sea su marca, modelo o serie.

2.3.9. Capa de sesión

Esta capa es la que se encarga de mantener y controlar el enlace establecido entre dos computadoras que están transmitiendo datos de cualquier índole. Por lo tanto, el servicio provisto por esta capa es la capacidad de asegurar que, dada una sesión establecida entre dos máquinas, la misma se pueda efectuar para las operaciones definidas de principio a fin, reanudándolas en caso de interrupción. En muchos casos, los servicios de la capa de sesión son parcial o totalmente prescindibles. (Felipe Sosa, 2013)

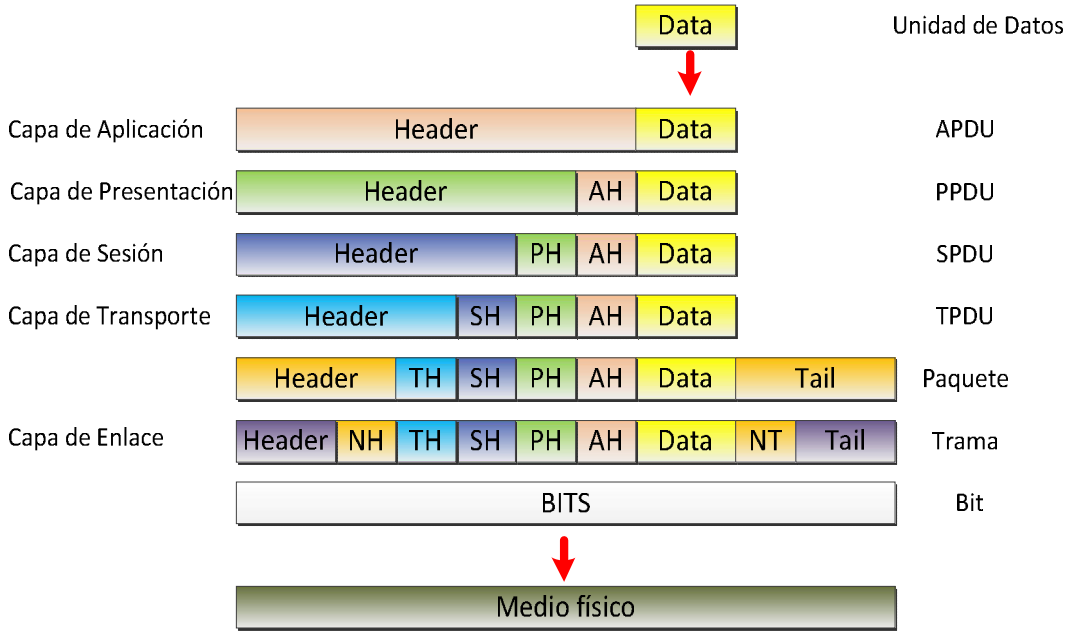
2.3.10. Capa de Presentación

El objetivo es encargarse de la representación de la información, de manera que aunque distintos equipos puedan tener diferentes representaciones internas de caracteres los datos lleguen de manera reconocible. Esta capa es la primera en trabajar más el contenido de la comunicación que el cómo se establece la misma. En ella se tratan aspectos tales como la semántica y la sintaxis de los datos transmitidos, ya que distintas computadoras pueden tener diferentes formas de manejarlas. Esta capa también permite cifrar los datos y comprimirlos. Por lo tanto, podría decirse que esta capa actúa como un traductor. (Elvis Proaño, 2012)

2.3.11. Capa de aplicación

Ofrece a las aplicaciones la posibilidad de acceder a los servicios de las demás capas y define los protocolos que utilizan las aplicaciones para intercambiar datos, como correo electrónico (Post Office Protocol y SMTP), gestores de bases de datos y servidor de ficheros (FTP), por UDP pueden viajar (DNS y Routing Information Protocol). Hay tantos protocolos como aplicaciones distintas y puesto que continuamente se desarrollan nuevas aplicaciones el número de protocolos crece sin parar.

Gráfico 2. 2: Tipos de capas



Fuente: (Groth & Skandier, 2005)

Cabe aclarar que el usuario normalmente no interactúa directamente con el nivel de aplicación. Suele interactuar con programas que a su vez interactúan con el nivel de aplicación pero ocultando la complejidad subyacente.

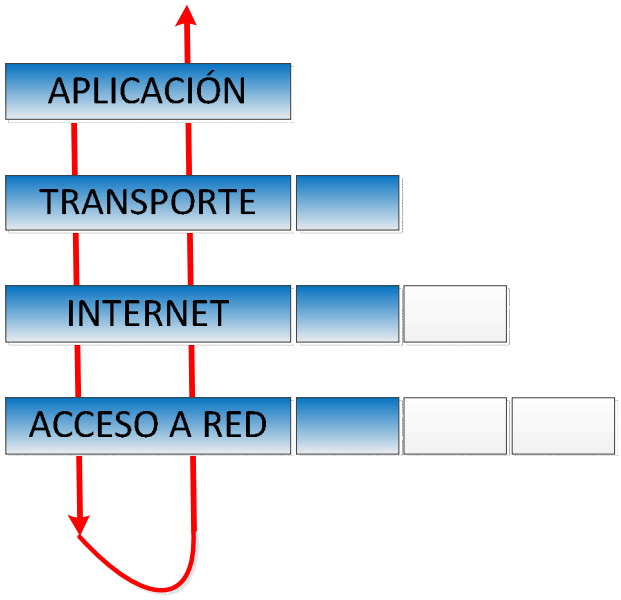
2.4. TCP/IP

Según (Postel, 1981) el modelo TCP/IP es un modelo de descripción de protocolos de red creado en la década de 1970 por DARPA, una agencia del Departamento de Defensa de los Estados Unidos. Evolucionó de ARPANET, el cual fue la primera red de área amplia y predecesora de Internet. EL modelo TCP/IP se denomina a veces como Internet Model. El modelo TCP/IP, describe un conjunto de guías generales de diseño e implementación de protocolos de red específicos para permitir que una computadora pueda comunicarse en una red. TCP/IP provee conectividad de extremo a extremo especificando como los datos deberían ser formateados, direccionados, transmitidos, enrutados y recibidos por el destinatario. Existen protocolos para los diferentes tipos de servicios de comunicación entre computadoras.

Las capas están jerarquizadas. Cada capa se construye sobre su predecesora. El número de capas y, en cada una de ellas, sus servicios y funciones son variables con cada tipo de red. Sin embargo, en cualquier red, la misión de cada capa es proveer servicios a las capas superiores haciéndoles transparentes el modo en que esos servicios se llevan a cabo. De esta manera, cada capa debe ocuparse exclusivamente de su nivel inmediatamente inferior, a quien solicita servicios, y del nivel inmediatamente superior, a quien devuelve resultados:

- **Capa 4 o capa de aplicación:** Aplicación, asimilable a las capas 5 (sesión), 6 (presentación) y 7 (aplicación) del modelo OSI. La capa de aplicación debía incluir los detalles de las capas de sesión y presentación OSI.
- Crearon una capa de aplicación que maneja aspectos de representación, codificación y control de diálogo.
- **Capa 3 o capa de transporte:** Transporte, asimilable a la capa 4 (transporte) del modelo OSI.
- **Capa 2 o capa de red: Internet:** asimilable a la capa 3 (red) del modelo OSI.
- **Capa 1 o capa de enlace:** Acceso al Medio, asimilable a la capa 1 (física) y 2 (enlace de datos) del modelo OSI.

Gráfico 2. 3: Modelo OSI



Fuente: (Postel, 1981)

2.5. Direccionamiento IP V4 - V6

2.5.1. Dirección IP

Una dirección IP es una etiqueta numérica que identifica, de manera lógica y jerárquica, a un interfaz (elemento de comunicación/conexión) de un dispositivo, habitualmente una computadora, dentro de una red que utilice el protocolo IP (Internet Protocol), que corresponde al nivel de red del protocolo TCP/IP. Dicho número no se ha de confundir con la dirección MAC que es un identificador de 48bits para identificar de forma única a la tarjeta de red y no depende del protocolo de conexión utilizado ni de la red. La dirección IP puede cambiar muy a menudo por cambios en la red o porque el dispositivo encargado dentro de la red de asignar las direcciones IP, decida asignar otra IP, por ejemplo, con el protocolo DHCP, a esta forma de asignación de dirección IP se denomina dirección IP dinámica, normalmente abreviado como IP dinámica.

Los sitios de Internet que por su naturaleza necesitan estar permanentemente conectados, generalmente tienen una dirección IP fija, esta, no cambia con el tiempo. Los servidores de correo, DNS, FTP públicos y servidores de páginas web necesariamente deben contar con una dirección IP fija o estática, ya que de esta forma se permite su localización en la red. A través de Internet las computadoras se conectan entre sí mediante sus respectivas direcciones IP. Sin embargo, a los seres humanos nos es más cómodo utilizar otra notación más fácil de recordar, como los nombres de dominio; la traducción entre unos y otros se resuelve mediante los servidores de nombres de dominio DNS, que a su vez, facilita el trabajo en caso de cambio de dirección IP, ya que basta con actualizar la información en el servidor DNS y el resto de las personas no se enterarán ya que seguirán accediendo por el nombre de dominio.

2.5.2. Direccionamiento IPV4

Las direcciones IPv4 se expresan por un número binario de 32 bits permitiendo un espacio de direcciones de 4.294.967.296 (232) direcciones posibles. Las direcciones IP se pueden expresar como números de notación decimal: se dividen los 32 bits de la dirección en cuatro octetos.

El valor decimal de cada octeto está comprendido en el rango de 0 a 255 [el número binario de 8 bits más alto es 11111111 y esos bits, de derecha a izquierda, tienen valores decimales de 1, 2, 4, 8, 16, 32, 64 y 128, lo que suma 255].

En la expresión de direcciones IPv4 en decimal se separa cada octeto por un carácter único ".". Cada uno de estos octetos puede estar comprendido entre 0 y 255, salvo algunas excepciones. Los ceros iniciales, si los hubiera, se pueden obviar.

Ejemplo de representación de dirección IPv4: 010.128.001.255 o 10.128.1.255

En las primeras etapas del desarrollo del Protocolo de Internet, los administradores de Internet interpretaban las direcciones IP en dos partes, los primeros 8 bits para designar la dirección de red y el resto para individualizar la computadora dentro de la red. Este método pronto probó ser inadecuado, cuando se comenzaron a agregar nuevas redes a las ya asignadas. En 1981 el direccionamiento internet fue revisado y se introdujo la arquitectura de clases (classful network architecture). En esta arquitectura hay tres clases de direcciones IP que una organización puede recibir de parte de la Internet Corporation for Assigned Names and Numbers (ICANN): clase A, clase B y clase C.

En una red de clase A, se asigna el primer octeto para identificar la red, reservando los tres últimos octetos (24 bits) para que sean asignados a los hosts, de modo que la cantidad máxima de hosts es 224 - 2 (se excluyen la dirección reservada para difusión (últimos octetos en 255) y de red (últimos octetos en 0)), es decir, 16 777 214 hosts.

En una red de clase B, se asignan los dos primeros octetos para identificar la red, reservando los dos octetos finales (16 bits) para que sean asignados a los hosts, de modo que la cantidad máxima de hosts es 216 - 2, o 65 534 hosts.

En una red de clase C, se asignan los tres primeros octetos para identificar la red, reservando el octeto final (8 bits) para que sea asignado a los hosts, de modo que la cantidad máxima de hosts es 28 - 2, o 254 hosts.

Gráfico 2. 4: Tabla de Direccionamiento IPV4

Clase	Intervalo	No. De redes	No. De equipos por red	Máscara de red	ID de broadcast
A	0.0.0.0 - 127.255.255.255	128	16 777 214	255.0.0.0	x.255.255.255
B	128.0.0.0 - 191.255.255.255	16 384	65 534	255.255.0.0	x.x.255.255
C	192.0.0.0 - 223.255.255.255	2 097 152	254	255.255.255.0	x.x.x.255
D	224.0.0.0 - 239.255.255.255	Histórico			
E	240.0.0.0 - 255.255.255.255	Histórico			

Fuente: (Philippe Atelin, 2013)

- La dirección 0.0.0.0 es reservada por la IANA para identificación local.
- La dirección que tiene los bits de host iguales a cero sirve para definir la red en la que se ubica. Se denomina dirección de red.
- La dirección que tiene los bits correspondientes a host iguales a uno, sirve para enviar paquetes a todos los hosts de la red en la que se ubica. Se denomina dirección de difusión.
- Las direcciones 127.x.x.x se reservan para designar la propia máquina. Se denomina dirección de bucle local o loopback.

El diseño de redes de clases (classful) sirvió durante la expansión de internet, sin embargo este diseño no era escalable y frente a una gran expansión de las redes en la década de los noventa, el sistema de espacio de direcciones de clases fue reemplazado por una arquitectura de redes sin clases Classless Inter-Domain Routing (CIDR) en el año 1993. CIDR está basada en redes de longitud de máscara de subred variable (variable-length subnet masking VLSM) que permite asignar redes de longitud de prefijo arbitrario. Permitiendo una distribución de direcciones más fina y granulada, calculando las direcciones necesarias y "desperdiciando" las mínimas posibles.

2.5.3. Direcciones privadas

Hay ciertas direcciones en cada clase de dirección IP que no están asignadas y que se denominan direcciones privadas.

Las direcciones privadas pueden ser utilizadas por los hosts que usan traducción de dirección de red (NAT) para conectarse a una red pública o por los hosts que no se conectan a Internet. En una misma red no pueden existir dos direcciones iguales, pero sí se pueden repetir en dos redes privadas que no tengan conexión entre sí o que se conecten mediante el protocolo NAT. Las direcciones privadas son:

- Clase A: 10.0.0.0 a 10.255.255.255 (8 bits red, 24 bits hosts).

- Clase B: 172.16.0.0 a 172.31.255.255 (16 bits red, 16 bits hosts). 16 redes clase B contiguas, uso en universidades y grandes compañías.

- Clase C: 192.168.0.0 a 192.168.255.255 (24 bits red, 8 bits hosts). 256 redes clase C contiguas, uso de compañías medias y pequeñas además de pequeños proveedores de internet (ISP).

Muchas aplicaciones requieren conectividad dentro de una sola red, y no necesitan conectividad externa. En las redes de gran tamaño a menudo se usa TCP/IP. Por ejemplo, los bancos pueden utilizar TCP/IP para conectar los cajeros automáticos que no se conectan a la red pública, de manera que las direcciones privadas son ideales para estas circunstancias.

Las direcciones privadas también se pueden utilizar en una red en la que no hay suficientes direcciones públicas disponibles. Las direcciones privadas se pueden utilizar junto con un servidor de traducción de direcciones de red (NAT) para suministrar conectividad a todos los hosts de una red que tiene relativamente pocas direcciones públicas disponibles. Según lo acordado, cualquier tráfico que posea una dirección destino dentro de uno de los intervalos de direcciones privadas no se enrutará a través de Internet.

2.5.4. Máscara de subred

La máscara permite distinguir los bits que identifican la red y los que identifican el host de una dirección IP.

Dada la dirección de clase A 10.2.1.2 sabemos que pertenece a la red 10.0.0.0 y el host al que se refiere es el 2.1.2 dentro de la misma.

La máscara se forma poniendo a 1 los bits que identifican la red y a 0 los bits que identifican el host. De esta forma una dirección de clase A tendrá como máscara 255.0.0.0, una de clase B 255.255.0.0 y una de clase C 255.255.255.0. Los dispositivos de red realizan un AND entre la dirección IP y la máscara para obtener la dirección de red a la que pertenece el host identificado por la dirección IP dada.

Por ejemplo un router necesita saber cuál es la red a la que pertenece la dirección IP del datagrama destino para poder consultar la tabla de encaminamiento y poder enviar el datagrama por la interfaz de salida. Para esto se necesita tener cables directos. La máscara también puede ser representada de la siguiente forma 10.2.1.2/8 donde el /8 indica que los 8 bits más significativos de máscara están destinados a redes, es decir /8 = 255.0.0.0. Análogamente (/16 = 255.255.0.0) y (/24 = 255.255.255.0).

2.5.5. Creación de subredes

El espacio de direcciones de una red puede ser subdividido a su vez creando subredes autónomas separadas. Un ejemplo de uso es cuando necesitamos agrupar todos los empleados pertenecientes a un departamento de una empresa. En este caso crearíamos una subred que englobara las direcciones IP de éstos. Para conseguirlo hay que reservar bits del campo host para identificar la subred estableciendo a uno los bits de red-subred en la máscara.

Por ejemplo la dirección 172.16.1.1 con máscara 255.255.255.0 nos indica que los dos primeros octetos identifican la red, por ser una dirección de clase C, el tercer octeto identifica la subred, a 1 los bits en la máscara y el cuarto identifica el host, a 0 los bits correspondientes dentro de la máscara. Hay dos direcciones de cada subred que quedan reservadas: aquella que identifica la subred campo host a 0 y la dirección para realizar difusión en la subred, todos los bits del campo host en 1.

2.5.6. IP dinámica

De acuerdo a (Philippe Atelin, 2013), una dirección IP dinámica es una IP asignada mediante un servidor DHCP (Dynamic Host Configuration Protocol) al usuario. La IP que se obtiene tiene una duración máxima determinada. El servidor DHCP provee parámetros de configuración específicos para cada cliente que desee participar en la red IP. Entre estos parámetros se encuentra la dirección IP del cliente.

DHCP apareció como protocolo estándar en octubre de 1993. El estándar RFC 2131 especifica la última definición de DHCP (marzo de 1997). DHCP sustituye al protocolo BOOTP, que es más antiguo. Debido a la compatibilidad retroactiva de DHCP, muy pocas redes continúan usando BOOTP puro. Las IP dinámicas son las que actualmente ofrecen la mayoría de operadores. El servidor del servicio DHCP puede ser configurado para que renueve las direcciones asignadas cada tiempo determinado.

2.5.6.1. Ventajas

- Reduce los costos de operación a los proveedores de servicios de Internet (ISP).
- Reduce la cantidad de IP asignadas de forma fija inactivas.

2.5.6.2. Desventajas

- Obliga a depender de servicios que redirigen un host a una IP

2.5.6.3. Asignación de direcciones IP

Dependiendo de la implementación concreta, el servidor DHCP tiene tres métodos para asignar las direcciones IP:

- **Manualmente.-** cuando el servidor tiene a su disposición una tabla que empareja direcciones MAC con direcciones IP, creada manualmente por el administrador de la red. Sólo clientes con una dirección MAC válida recibirán una dirección IP del servidor.

- **Automáticamente.-** donde el servidor DHCP asigna permanentemente una dirección IP libre, tomada de un rango prefijado por el administrador, a cualquier cliente que solicite una.

- **Dinámicamente.-** el único método que permite la reutilización de direcciones IP. El administrador de la red asigna un rango de direcciones IP para el DHCP y cada computadora cliente de la LAN tiene su software de comunicación TCP/IP configurado para solicitar una dirección IP del servidor DHCP cuando su tarjeta de interfaz de red se inicie. El proceso es transparente para el usuario y tiene un periodo de validez limitado.

2.5.7. IP fija

Una dirección IP fija es una dirección IP asignada por el usuario de manera manual, o por el servidor de la red, ISP en el caso de internet, router o switch en caso de LAN, con base en la Dirección MAC del cliente. Mucha gente confunde IP Fija con IP Pública e IP Dinámica con IP Privada. Una IP puede ser Privada ya sea dinámica o fija como puede ser IP Pública:

- Dinámica o Fija. Una IP Pública se utiliza generalmente para montar servidores en internet y necesariamente se desea que la IP no cambie por eso siempre la IP

- Pública se la configura de manera Fija y no Dinámica, aunque si se podría.

En el caso de la IP Privada generalmente es dinámica asignada por un servidor DHCP, pero en algunos casos se configura IP Privada Fija para poder controlar el acceso a internet o a la red local, otorgando ciertos privilegios dependiendo del número de IP que tenemos, si esta cambiara (fuera dinámica) sería más complicado controlar estos privilegios.

Las IP Públicas fijas actualmente en el mercado de acceso a Internet tienen un costo adicional mensual. Estas IP son asignadas por el usuario después de haber recibido la información del proveedor o bien asignadas por el proveedor en el momento de la primera conexión. Esto permite al usuario montar servidores web, correo, FTP, etc. y dirigir un nombre de dominio a esta IP sin tener que mantener actualizado el servidor DNS cada vez que cambie la IP como ocurre con las IP Públicas dinámicas.

2.6. Direccionamiento IPV6

Según (IPV6, 2014) El Internet Protocol versión 6 (IPv6) (en español: Protocolo de Internet versión 6) es una versión del protocolo Internet Protocol (IP), y diseñada para reemplazar a Internet Protocol versión 4 (IPv4), que actualmente está implementado en la gran mayoría de dispositivos que acceden a Internet.

Diseñado por Steve Deering de Xerox PARC y Craig Mudge, IPv6 está destinado a sustituir a IPv4, cuyo límite en el número de direcciones de red admisibles está empezando a restringir el crecimiento de Internet y su uso, especialmente en China, India, y otros países asiáticos densamente poblados. El nuevo estándar mejorará el servicio globalmente; por ejemplo, proporcionará a futuras celdas telefónicas y dispositivos móviles sus direcciones propias y permanentes. A principios de 2010, quedaban menos del 10% de IP´s sin asignar. En la semana del 3 de febrero del 2011, la IANA (Agencia Internacional de Asignación de Números de Internet, por sus siglas en inglés) entregó el último bloque de direcciones disponibles (33 millones) a la organización encargada de asignar IP´s en Asia, un mercado que está en auge y no tardará en consumirlas todas. IPv4 posibilita 4.294.967.296 (232) direcciones de red diferentes, un número inadecuado para dar una dirección a cada persona del planeta, y mucho menos a cada vehículo, teléfono, PDA, etcétera. En cambio, IPv6 admite 340.282.366.920.938.463.463.374.607.431.768.211.456 (2128 o 340 sextillones de direcciones) cerca de 6,7 × 1017 (670 mil billones) de direcciones por cada milímetro cuadrado de la superficie de la tierra. (Jose Benavides, 2013)

El cambio más grande de IPv4 a IPv6 es la longitud de las direcciones de red.[3] Las direcciones IPv6, son de 128 bits; esto corresponde a 32 dígitos hexadecimales, que se utilizan normalmente para escribir las direcciones IPv6. En muchas ocasiones las direcciones IPv6 están compuestas por dos partes lógicas:

- Un prefijo de 64 bits y otra parte de 64 bits que corresponde al identificador de interfaz, que casi siempre se genera automáticamente a partir de la dirección MAC de la interfaz a la que está asignada la dirección.

2.6.1. Notación para las direcciones IPv6

Según (D-Link, 2012), las direcciones IPv6, de 128 bits de longitud, se escriben como ocho grupos de cuatro dígitos hexadecimales; por ejemplo:

- 2001:0db8:85a3:08d3:1319:8a2e:0370:7334 es una dirección IPv6 válida.

Se puede comprimir un grupo de cuatro dígitos si éste es nulo (es decir, toma el valor "0000"); por ejemplo:

- 2001:0db8:85a3:0000:1319:8a2e:0370:7344
- 2001:0db8:85a3::1319:8a2e:0370:7344

Siguiendo esta regla, si más de dos grupos consecutivos son nulos, también pueden comprimirse como "::". Si la dirección tiene más de una serie de grupos nulos consecutivos la compresión sólo se permite en uno de ellos. Así, las siguientes son representaciones posibles de una misma dirección:

- 2001:0DB8:0000:0000:0000:0000:1428:57ab
- 2001:0DB8:0000:0000:0000::1428:57ab
- 2001:0DB8:0:0:0:0:1428:57ab

[3] De acuerdo a los Ingenieros de Sistemas Carlos Montero Lucio y Sergio Flores, Telconet ya cuenta con IPV6 desde el año 2012, referencia aquí

- 2001:0DB8:0::0:1428:57ab
- 2001:0DB8::1428:57ab

Son todas válidas y significan lo mismo, pero

- 2001::25de::cade

No es válida porque no queda claro cuántos grupos nulos hay en cada lado. Los ceros iniciales en un grupo también se pueden omitir:

- 2001:0DB8:02de::0e13
- 2001:DB8:2de::e13

Si la dirección es una dirección IPv4 empotrada, los últimos 32 bits pueden escribirse en base decimal, así:

- ::ffff:192.168.89.9
- ::ffff:c0a8:5909

No se debe confundir con:

- ::192.168.89.9
- ::c0a8:5909

El formato ::ffff:1.2.3.4 se denomina dirección IPv4 mapeada, y el formato ::1.2.3.4 dirección IPv4 compatible.

Las direcciones IPv4 pueden ser transformadas fácilmente al formato IPv6; por ejemplo, si la dirección decimal IPv4 es 135.75.43.52 (en hexadecimal, 0x874B2B34), puede ser convertida a 000:0000:0000:0000:0000:0000:874B:2B34 o ::874B:2B34. Entonces, uno puede usar la notación mixta dirección IPv4 compatible, en cuyo caso la dirección debería ser::135.75.43.52. Este tipo de dirección IPv4 compatible casi no está siendo utilizada en la práctica, aunque los estándares no la han declarado obsoleta.

Cuando lo que se desea es identificar un rango de direcciones diferenciable por medio de los primeros bits, se añade este número de bits tras el carácter de barra "/". Por ejemplo:

- 2001:0DB8::1428:57AB/96 sería equivalente a 2001:0DB8::

- 2001:0DB8::874B:2B34/96 sería equivalente a 2001:0DB8:: y por supuesto también a 2001:0DB8::1428:57AB/96

2.6.2. Identificación de los tipos de direcciones

Los tipos de direcciones IPv6 pueden identificarse tomando en cuenta los rangos definidos por los primeros bits de cada dirección:

- ::/128

La dirección con todo ceros se utiliza para indicar la ausencia de dirección, y no se asigna ningún nodo:

- ::1/128

La dirección de loopback es una dirección que puede usar un nodo para enviarse paquetes a sí mismo (corresponde con 127.0.0.1 de IPv4). No puede asignarse a ninguna interfaz física:

- ::1.2.3.4/96

La dirección IPv4 compatible se usa como un mecanismo de transición en las redes duales IPv4/IPv6. Es un mecanismo que no se usa.

- ::ffff:0:0/96

La dirección IPv4 mapeada se usa como mecanismo de transición en terminales Duales:

- fe80::/10

El prefijo de enlace local (en inglés link local) específica que la dirección sólo es válida en el enlace físico local:

- fec0::

El prefijo de emplazamiento local (en inglés site-local prefix) específica que la dirección sólo es válida dentro de una organización local. La RFC 3879 lo declaró obsoleto, estableciendo que los sistemas futuros no deben implementar ningún soporte para este tipo de dirección especial. Se deben sustituir por direcciones Local IPv6 Unicast. ff00::/8

El prefijo de multicast. Se usa para las direcciones multicast. Hay que resaltar que no existen las direcciones de difusión (en inglés broadcast) en IPv6, aunque la funcionalidad que prestan puede emularse utilizando la dirección multicast FF01::1/128, denominada todos los nodos (en inglés all nodes). Un paquete en IPv6 está compuesto principalmente de dos partes: la cabecera, que tiene una parte fija y otra con las opciones y los datos.

Cabecera Fija

Los primeros 40 bytes (320 bits) son la cabecera del paquete y contiene los siguientes campos:

Gráfico 2. 5: Paquete en IPV6

Dirección Multicast	Área de Funcionamiento	Significado	Descripción
FF01::1	Nodo	Todos los nodos	Todos los nodos en la interfase local
FF01::2	Nodo	Todos los enrutadores	Todos los enrutadores en la interfase local
FF02::1	Enlace Local	Todos los nodos	Todos los nodos en el enlace local
FF02::2	Enlace Local	Todos los enrutadores	Todos los enrutadores en el enlace local
FF05::2	Sitio	Todos los enrutadores	Todos los enrutadores en un sitio

Fuente: (IPV6, 2014)

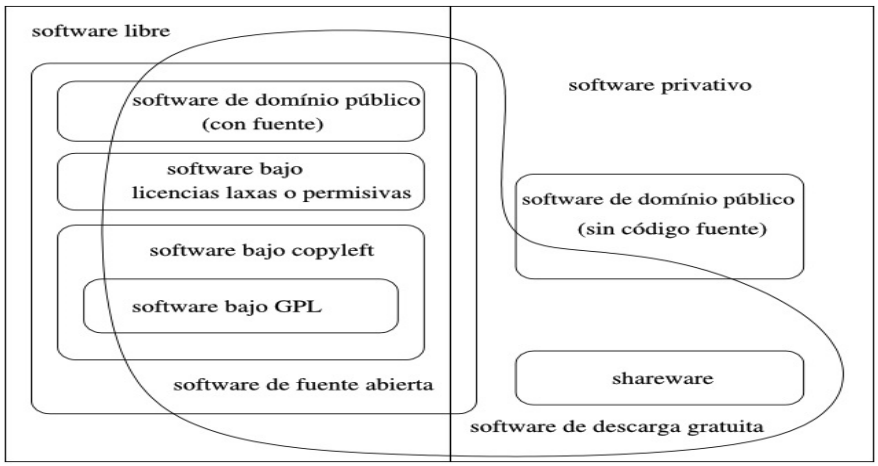
2.7. Sistemas Operativos

Según (Castells, 1997), el sistema operativo es el programa o software más importante de una computadora. Para que funcionen los otros programas, cada computadora de uso general debe tener un sistema operativo. Los sistemas operativos realizan tareas básicas, tales como reconocimiento de la conexión del teclado, enviar la información a la pantalla, no perder de vista archivos y directorios en el disco, y controlar los dispositivos periféricos tales como impresoras, escáner, etc. En sistemas grandes, el sistema operativo tiene incluso mayor responsabilidad y poder, es como un policía de tráfico, se asegura de que los programas y usuarios que están funcionando al mismo tiempo no interfieran entre ellos. El sistema operativo también es responsable de la seguridad, asegurándose de que los usuarios no autorizados no tengan acceso al sistema. Un usuario normalmente interactúa con el sistema operativo a través de un sistema de comandos, por ejemplo, el sistema operativo DOS contiene comandos como copiar y pegar para copiar y pegar archivos respectivamente. Los comandos son aceptados y ejecutados por una parte del sistema operativo llamada procesador de comandos o intérprete de la línea de comandos.

2.7.1. Sistemas Operativos Propietarios

El software propietario, mala traducción de proprietar y software, en inglés, también llamado privativo, privado, de código cerrado, cautivo o software no libre, es cualquier programa informático en el que el usuario tiene limitaciones para usarlo, modificarlo o redistribuirlo. Para la Fundación para el Software Libre (FSF) este concepto se aplica a cualquier software que no es libre o que sólo lo es parcialmente (semilibre), sea porque su uso, redistribución o modificación está prohibida, o requiere permiso expreso del titular del software. La persona física o jurídica al poseer los derechos de autor sobre un software tiene la posibilidad de controlar y restringir los derechos del usuario sobre su programa, lo que en el software no libre implica por lo general que el usuario sólo tendrá derecho a ejecutar el software bajo ciertas condiciones, comúnmente fijadas por el proveedor, que signifique la restricción de una o varias de las cuatro libertades.

Gráfico 2. 6: Libertades del software



Fuente: (Foundation, 2012)

2.7.2. Sistemas Operativos Libres

El software libre, en inglés free software, aunque esta denominación también se confunde a veces con "gratis" por la ambigüedad del término "free" en el idioma inglés, por lo que también se usa "libre software" y "logical libre", es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, modificado, y redistribuido libremente. Según la Free Software Foundation, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar el software y distribuirlo modificado. El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar software libre a "software gratuito", denominado usualmente freeware, ya que, conservando su carácter de libre, puede ser distribuido comercialmente ("software comercial"). Análogamente, el "software gratis" o "gratuito" incluye en ocasiones el código fuente; no obstante, este tipo de software no es libre en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

2.7.3. Software libre

Software libre (en inglés free software, aunque esta denominación a veces se confunde con “gratis” por la ambigüedad del término free en el idioma inglés, por lo que también se usa libre software) es la denominación del software que respeta la libertad de todos los usuarios que adquirieron el producto y, por tanto, una vez obtenido el mismo puede ser usado, copiado, estudiado, modificado, y redistribuido libremente de varias formas. Según la Free Software Foundation, el software libre se refiere a la seguridad de los usuarios para ejecutar, copiar, distribuir y estudiar el software, e incluso modificarlo y distribuirlo modificado. El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar “software libre” a “software gratuito” (denominado usualmente freeware), ya que, conservando su carácter de libre, puede ser distribuido comercialmente (software comercial). Análogamente, el software gratis o gratuito incluye en ocasiones el código fuente; no obstante, este tipo de software no es “libre” en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa. Tampoco debe confundirse software libre con “software de dominio público”. Éste último es aquel software que no requiere de licencia, pues sus derechos de explotación son para toda la humanidad, porque pertenece a todos por igual. Cualquiera puede hacer uso de él, siempre con fines legales y consignando su autoría original. Este software sería aquel cuyo autor lo dona a la humanidad o cuyos derechos de autor han expirado, tras un plazo contado desde la muerte de éste.

2.7.4. Historia del software libre

Entre los años 1960 y 1970, el software no era considerado un producto sino un añadido que los vendedores de las grandes computadoras de la época (las mainframes) aportaban a sus clientes para que éstos pudieran usarlos. En dicha cultura, era común que los programadores y desarrolladores de software compartieran libremente sus programas unos con otros.

Este comportamiento era particularmente habitual en algunos de los mayores grupos de usuarios de la época, como DECUS (grupo de usuarios de computadoras DEC). A finales de la década de 1970, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia. En 1971, cuando la informática todavía no había sufrido su gran boom, las personas que hacían uso de ella, en ámbitos universitarios y empresariales, creaban y compartían el software sin ningún tipo de restricciones. Con la llegada de los años 1980 la situación empezó a cambiar. Las computadoras más modernas comenzaban a utilizar sistemas operativos privativos, forzando a los usuarios a aceptar condiciones restrictivas que impedían realizar modificaciones a dicho software. En caso de que algún usuario o programador encontrase algún error en la aplicación, lo único que podía hacer era darlo a conocer a la empresa desarrolladora para que ésta lo solucionara. Aunque el programador estuviese capacitado para solucionar el problema y lo desease hacer sin pedir nada a cambio, el contrato le impedía que modificase el software.

El mismo Richard Matthew Stallman cuenta que por aquellos años, en el laboratorio donde trabajaba, habían recibido una impresora donada por una empresa externa. El dispositivo, que era utilizado en red por todos los trabajadores, parecía no funcionar a la perfección, dado que cada cierto tiempo el papel se atascaba. Como agravante, no se generaba ningún aviso que se enviase por red e informase a los usuarios de la situación. La pérdida de tiempo era constante, ya que en ocasiones, los trabajadores enviaban por red sus trabajos a imprimir y al ir a buscarlos se encontraban la impresora atascada y una cola enorme de trabajos pendientes. Richard Stallman decidió arreglar el problema, e implementar el envío de un aviso por red cuando la impresora se bloqueara. Para ello necesitaba tener acceso al código fuente de los controladores de la impresora. Pidió a la empresa propietaria de la impresora lo que necesitaba, comentando, sin pedir nada a cambio, qué era lo que pretendía realizar. La empresa se negó a entregarle el código fuente. En ese preciso instante, Stallman se vio en una encrucijada: debía elegir entre aceptar el nuevo software propietario firmando acuerdos de no revelación y acabar desarrollando más software propietario con licencias restrictivas, que a su vez deberían ser más adelante aceptadas por sus propios colegas.

Con este antecedente, en 1984, Richard Stallman comenzó a trabajar en el proyecto GNU, y un año más tarde fundó la Free Software Foundation (FSF). Stallman introdujo la definición de software libre y el concepto de "copyleft", que desarrolló para otorgar libertad a los usuarios y para restringir las posibilidades de apropiación del software. De acuerdo con tal definición, un software es "libre" cuando garantiza las siguientes libertades:

Libertad	Descripción
0	la libertad de usar el programa, con cualquier propósito
1	la libertad de estudiar cómo funciona el programa y modificarlo, adaptándolo a tus necesidades
2	la libertad de distribuir copias del programa, con lo cual puedes ayudar a tu prójimo
3	la libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie

Fuente: Richard Stallman, 1980

Las libertades 1 y 3 requieren acceso al código fuente porque estudiar y modificar software sin su código fuente es muy poco viable. Ciertos teóricos usan este cuarto punto (libertad 3) para justificar parcialmente las limitaciones impuestas por la licencia GNU GPL frente a otras licencias de software libre (ver Licencias GPL). Sin embargo el sentido original es más libre, abierto y menos restrictivo que el que le otorga la propia situación de incompatibilidad, que podría ser resuelta en la próxima versión 3.0 de la licencia GNU GPL, que causa en estos momentos graves perjuicios a la comunidad de programadores de software libre, ya que muchas veces no se puede reutilizar o mezclar códigos de dos licencias distintas, pese a que las libertades teóricamente lo deberían permitir. Tanto la Open Source Initiative como la Free Software Foundation, mantienen en sus webs oficiales, listados de las licencias de software libre que aprueban. El término software no libre se emplea para referirse al software distribuido bajo una licencia de software más restrictiva que no garantiza estas cuatro libertades.

Las leyes de la propiedad intelectual reservan la mayoría de los derechos de modificación, duplicación, y redistribución, para el dueño del copyright; el software dispuesto bajo una licencia de software libre rescinde específicamente la mayoría de estos derechos reservados. La definición de software libre no contempla la cuestión del precio; un eslogan frecuentemente usado es "libre como en libertad, no como en cerveza gratis" o en inglés "Free as in freedom, not as in free beer" (aludiendo a la ambigüedad del término inglés "free"), y es habitual ver a la venta CD de software libre como distribuciones Linux. Sin embargo, en esta situación, el comprador del CD tiene el derecho de copiarlo y redistribuirlo. El software gratis puede incluir restricciones que no se adaptan a la definición de software libre; por ejemplo, puede no incluir el código fuente, puede prohibir explícitamente a los distribuidores recibir una compensación a cambio, etc. Para evitar la confusión, algunas personas utilizan los términos "libre" (software libre) y "gratis" (software gratis) para evitar la ambigüedad de la palabra inglesa "free". Sin embargo, estos términos alternativos son usados únicamente dentro del movimiento del software libre, aunque están extendiéndose lentamente hacia el resto del mundo. Otros defienden el uso del término open source software (software de código abierto). La principal diferencia entre los términos "open source" y "free software" es que éste último tiene en cuenta los aspectos éticos y filosóficos de la libertad, mientras que el "open source" se basa únicamente en los aspectos técnicos. En un intento por unir los mencionados términos que se refieren a conceptos semejantes, se está extendiendo el uso de la palabra "FLOSS" con el significado de free/libre and open source software e, indirectamente, también a la comunidad que lo produce y apoya.

2.7.5. Ventajas del software libre

- **Bajo costo de adquisición:** Se trata de un software económico ya que permite un ahorro de grandes cantidades en la adquisición de las licencias.
- **Innovación tecnológica:** esto se debe a que cada usuario puede aportar sus conocimientos y su experiencia y así decidir de manera conjunta hacia donde se debe dirigir la evolución y el desarrollo del software.

- **Independencia del proveedor:** al disponer del código fuente, se garantiza una independencia del proveedor que hace que cada empresa o particular pueda seguir contribuyendo al desarrollo y los servicios del software.
- **Escrutinio público:** esto hace que la corrección de errores y la mejora del producto se lleven a cabo de manera rápida y eficaz por cada uno de los usuarios que lleguen a utilizar el producto.
- **Adaptación del software:** esta cualidad resulta de gran utilidad para empresas e industrias específicas que necesitan un software personalizado para realizar un trabajo.
- **Lenguas:** aunque el software se cree y salga al mercado en una sola lengua, el hecho de ser software libre facilita en gran medida su traducción y localización para que usuarios de diferentes partes del mundo puedan aprovechar estos beneficios.
- **Aprovechamiento más adecuado de los recursos:** muchas aplicaciones utilizadas o promovidas por las administraciones públicas son también utilizadas por otros sectores de la sociedad.
- **Fomento de la industria local:** una de las mayores ventajas del software libre es la posibilidad de desarrollar industria local de software.
- **Independencia del proveedor:** es obvio que una organización preferirá depender de un mercado en régimen de competencia que de un solo proveedor que puede imponer las condiciones en que proporciona su producto.
- **Adaptación a las necesidades exactas:** en el caso del software libre, la adaptación puede hacerse con mucha mayor facilidad, y lo que es más importante, sirviéndose de un mercado con competencia, si hace falta contratarla.

2.7.6. Desventajas del Software Libre

- Algunas aplicaciones pueden llegar a ser algo complicadas de instalar.
- Inexistencia de garantía por parte del autor, por ello existen comunidades, que ayudan y aportan tanto en código como en soluciones.
- Poca estabilidad y flexibilidad en el campo de multimedia y juegos.

- Menor compatibilidad con el hardware
- Dificultad en el intercambio de archivos: esto se da mayormente en los documentos de texto (generalmente creados con Microsoft Word), ya que si los queremos abrir con un Software Libre.

2.7.7. Formatos abiertos

Los formatos abiertos permiten al software libre mantener sus cuatro libertades y la libre difusión de todo el código y formatos utilizados, su distribución y estudio, debido a esto, los creadores de software libre desarrollan a la vez de programas libres, formatos libres para estos programas o utilizan formatos libres ya creados anteriormente. Los formatos libres permiten a los usuarios poder trabajar con programas libres aunque al ser libres pueden ser implementados y utilizados cualquier programa sea cerrado o no. Algunas compañías, como Microsoft, suelen no utilizar formatos libres en sus programas, no por impedimento si no por falta de voluntad de implementar formatos abiertos en sus programas, aun así los usuarios pueden instalar software libre en sus sistemas para trabajar con estos formatos.

2.7.8. Tipos de licencias

Una licencia es aquella autorización formal con carácter contractual que un autor de un software da a un interesado para ejercer "actos de explotación legales". Pueden existir tantas licencias como acuerdos concretos se den entre el autor y el licenciatario.

2.7.9. Licencias GPL

Una de las más utilizadas es la Licencia Pública General de GNU (GNU GPL). El autor conserva los derechos de autor (copyright), y permite la redistribución y modificación bajo términos diseñados para asegurarse de que todas las versiones modificadas del software permanecen bajo los términos más restrictivos de la propia GNU GPL.

Esto hace que sea imposible crear un producto con partes no licenciadas GPL: el conjunto tiene que ser GPL. Es decir, la licencia GNU GPL posibilita la modificación y redistribución del software, pero únicamente bajo esa misma licencia. Y añade que si se reutiliza en un mismo programa código "A" licenciado bajo licencia GNU GPL y código "B" licenciado bajo otro tipo de licencia libre, el código final "C", independientemente de la cantidad y calidad de cada uno de los códigos "A" y "B", debe estar bajo la licencia GNU GPL. En la práctica esto hace que las licencias de software libre se dividan en dos grandes grupos, aquellas que pueden ser mezcladas con código licenciado bajo GNU GPL (y que inevitablemente desaparecerán en el proceso, al ser el código resultante licenciado bajo GNU GPL) y las que no lo permiten al incluir mayores u otros requisitos que no contemplan ni admiten la GNU GPL y que por lo tanto no pueden ser enlazadas ni mezcladas con código gobernado por la licencia GNU GPL. En el sitio web oficial de GNU hay una lista de licencias que cumplen las condiciones impuestas por la GNU GPL y otras que no.

2.7.10. Licencias AGPL

La Licencia Pública General de Affero (en inglés Affero General Public License, también Affero GPL o AGPL) es una licencia copyleft derivada de la Licencia Pública General de GNU diseñada específicamente para asegurar la cooperación con la comunidad en el caso de software que corra en servidores de red. La Affero GPL es íntegramente una GNU GPL con una cláusula nueva que añade la obligación de distribuir el software si éste se ejecuta para ofrecer servicios a través de una red de computadoras. La Free Software Foundation recomienda que el uso de la GNU AGPLv3 sea considerado para cualquier software que usualmente corra sobre una red.

2.7.11. Licencias estilo BSD

Llamadas así porque se utilizan en gran cantidad de software distribuido junto a los sistemas operativos BSD.

El autor, bajo tales licencias, mantiene la protección de copyright únicamente para la renuncia de garantía y para requerir la adecuada atribución de la autoría en trabajos derivados, pero permite la libre redistribución y modificación, incluso si dichos trabajos tienen propietario. Son muy permisivas, tanto que son fácilmente absorbidas al ser mezcladas con la licencia GNU GPL con quienes son compatibles. Puede argumentarse que esta licencia asegura “verdadero” software libre, en el sentido que el usuario tiene libertad ilimitada con respecto al software, y que puede decidir incluso redistribuirlo como no libre. Otras opiniones están orientadas a destacar que este tipo de licencia no contribuye al desarrollo de más software libre (normalmente utilizando la siguiente analogía: "una licencia BSD es más libre que una GPL si y sólo si se opina también que un país que permita la esclavitud es más libre que otro que no la permite").

2.7.12. Licencias estilo MPL y derivadas

Esta licencia es de Software Libre y tiene un gran valor porque fue el instrumento que empleó Netscape Communications Corp. para liberar su Netscape Communicator 4.0 y empezar ese proyecto tan importante para el mundo del Software Libre: Mozilla. Se utilizan en gran cantidad de productos de software libre de uso cotidiano en todo tipo de sistemas operativos. La MPL es Software Libre y promueve eficazmente la colaboración evitando el efecto "viral" de la GPL (si usas código licenciado GPL, tu desarrollo final tiene que estar licenciado GPL). Desde un punto de vista del desarrollador la GPL presenta un inconveniente en este punto, y lamentablemente mucha gente se cierra en banda ante el uso de dicho código. No obstante la MPL no es tan excesivamente permisiva como las licencias tipo BSD. Estas licencias son denominadas de copyleft débil. La NPL (luego la MPL) fue la primera licencia nueva después de muchos años, que se encargaba de algunos puntos que no fueron tomados en cuenta por las licencias BSD y GNU.

2.7.13. Copyleft

Hay que hacer constar que el titular de los derechos de autor (copyright) de un software bajo licencia copyleft puede también realizar una versión modificada bajo su copyright original, y venderla bajo cualquier licencia que desee, además de distribuir la versión original como software libre. Esta técnica ha sido usada como un modelo de negocio por una serie de empresas que realizan software libre (por ejemplo MySQL); esta práctica no restringe ninguno de los derechos otorgados a los usuarios de la versión copyleft. En España, toda obra derivada está tan protegida como una original, siempre que la obra derivada parta de una autorización contractual con el autor. En el caso genérico de que el autor retire las licencias "copyleft", no afectaría de ningún modo a los productos derivados anteriores a esa retirada, ya que no tiene efecto retroactivo. En términos legales, el autor no tiene derecho a retirar el permiso de una licencia en vigencia. Si así sucediera, el conflicto entre las partes se resolvería en un pleito convencional.

2.7.14. Comparación con el software de código abierto

Aunque en la práctica el software de código abierto y el software libre comparten muchas de sus licencias, la Free Software Foundation opina que el movimiento del software de código abierto es filosóficamente diferente del movimiento del software libre. Apareció en 1998 con un grupo de personas, entre los que cabe destacar a Eric S. Raymond y Bruce Perens, que formaron la Open Source Initiative (OSI). Ellos buscaban darle mayor relevancia a los beneficios prácticos del compartir el código fuente, e interesar a las principales casas de software y otras empresas de la industria de la alta tecnología en el concepto. Por otro lado, la Free Software Foundation y Richard Stallman prefieren plantear el asunto en términos éticos empleando el término "software libre". Los defensores del término "código abierto", en inglés open source, afirman que éste evita la ambigüedad del término en ese idioma que es free en free software.

El término "código abierto" fue acuñado por Christine Peterson del think tank Foresight Institute, y se registró para actuar como marca registrada el término en inglés para los productos de software libre. Mucha gente reconoce el beneficio cualitativo del proceso de desarrollo de software cuando los desarrolladores pueden usar, modificar y redistribuir el código fuente de un programa. El movimiento del software libre hace especial énfasis en los aspectos morales o éticos del software, viendo la excelencia técnica como un producto secundario de su estándar ético. El movimiento de código abierto ve la excelencia técnica como el objetivo prioritario, siendo la compartición del código fuente un medio para dicho fin. Por dicho motivo, la FSF se distancia tanto del movimiento de código abierto como del término "Código Abierto" (en inglés Open Source).

Puesto que la OSI sólo aprueba las licencias que se ajustan a la Open Source Definition (definición de código abierto), la mayoría de la gente lo interpreta como un esquema de distribución, e intercambia libremente "código abierto" con "software libre". Aun cuando existen importantes diferencias filosóficas entre ambos términos, especialmente en términos de las motivaciones para el desarrollo y el uso de tal software, raramente suelen tener impacto en el proceso de colaboración. Aunque el término "código abierto" elimina la ambigüedad de libertad frente a precio (en el caso del inglés), introduce una nueva: entre los programas que se ajustan a la definición de código abierto, que dan a los usuarios la libertad de mejorarlos, y los programas que simplemente tiene el código fuente disponible, posiblemente con fuertes restricciones sobre el uso de dicho código fuente.

Mucha gente cree que cualquier software que tenga el código fuente disponible es de código abierto, puesto que lo pueden manipular (un ejemplo de este tipo de software sería el popular paquete de software gratuito Graphviz, inicialmente no libre pero que incluía el código fuente, aunque luego AT&T le cambió la licencia). Sin embargo, mucho de este software no da a sus usuarios la libertad de distribuir sus modificaciones, restringe el uso comercial, o en general restringe los derechos de los usuarios.

2.7.15. Implicaciones económico-políticas

Una vez que un producto de software libre ha empezado a circular, rápidamente está disponible a un costo muy bajo. Al mismo tiempo, su utilidad no decrece. El software, en general, podría ser considerado un bien de uso inagotable, tomando en cuenta que su costo marginal es pequeñísimo y que no es un bien sujeto a rivalidad (la posesión del bien por un agente económico no impide que otro lo posea). Puesto que el software libre permite el libre uso, modificación y redistribución, a menudo encuentra un hogar entre usuarios para los cuales el coste del software no libre es a veces prohibitivo, o como alternativa a la piratería. También es sencillo modificarlo localmente, lo que permite que sean posibles los esfuerzos de traducción a idiomas que no son necesariamente rentables comercialmente. Existen muchas posturas acerca de la relación entre el software libre y el actual sistema político-económico:

- Algunos consideran el software libre como un competidor contra el centralismo en empresas y gobiernos, una forma de orden espontáneo o de anarquismo práctico.
- Algunos consideran el software libre como una forma de trabajo colaborativo en un modelo de mercado, tal como se había planteado el cooperativismo.
- Algunos comparan el software libre a una economía del regalo, donde el valor de una persona está basado en lo que ésta da a los demás, sin que incurra valor monetario formal de por medio.
- Grupos como Oekonux e Hipatia consideran que todo debería producirse de esta forma y que este modelo de producción no se limita a reemplazar el modelo no libre de desarrollo del software. La cooperación basada en la libre asociación puede usarse y se usa para otros propósitos (tales como escribir enciclopedias, por ejemplo).
- Hay proyectos de desarrollo con impulso gubernamental que utilizan software libre, así como en proyectos de voluntariado en países en vías de desarrollo.

Las implicaciones políticas y económicas del software libre, o su afinidad con el antiautoritarismo, son discutidas. Mientras para unos estas implicaciones son notorias y representan un factor importante a tomarse en cuenta, para otros si bien podría existir una leve relación, no tiene suficiente relevancia.

2.7.16. Modelo de negocio

El negocio detrás del software libre se caracteriza por la oferta de servicios adicionales al software como: la personalización y/o instalación del mismo, soporte técnico, donaciones, patrocinios o como un elemento de responsabilidad social corporativa; en contraposición al modelo de negocio basado en licencias predominante en el software de código cerrado.

2.7.17. Seguridad relativa

Existe una cierta controversia sobre la seguridad del software libre frente al software no libre (siendo uno de los mayores asuntos la seguridad por oscuridad). Un método usado de forma habitual para determinar la seguridad relativa de los productos es determinar cuántos fallos de seguridad no parcheados existen en cada uno de los productos involucrados. Por lo general los usuarios de este método recomiendan que cuando un producto no proporcione un método de parchear los fallos de seguridad, no se use dicho producto, al menos hasta que no esté disponible un arreglo.

2.7.18. Software libre en la Administración Pública

Existe una serie de países en los cuales, sus administraciones públicas, han mostrado apoyo al software libre, sea migrando total o parcialmente sus servidores y sistemas de escritorio, sea subvencionándolo. Como ejemplos de ello se tiene a Alemania, Argentina, Brasil, Cuba, Chile, China, Ecuador, España, Francia, México, República Dominicana, y Venezuela.

Además de lo anterior, la Administración Pública tiene una cierta función de escaparate y/o guía de la industria que la hace tener un gran impacto, que debería dirigirse a la creación de un tejido tecnológico generador de riqueza nacional. Ésta puede crearse fomentando empresas, cuyo negocio sea en parte el desarrollo de nuevo software libre para la Administración, el mantenimiento y la adaptación del software existente, etc. En España en el año 2009, el Centro Nacional de Referencia de Aplicación de las TIC basadas en Fuentes Abiertas (CENATIC), elaboró un informe junto a la Universidad Rey Juan Carlos (Grupo GsyC/LibreSoft) y Telefónica I+D, con el fin de analizar el estado en que se encuentra el proceso de implantación del software de fuentes abiertas en la Administración Pública española. En México, el Software Libre nació en las universidades y los centros de investigación. Es por eso que, desde hace tres décadas, los estudiantes y los profesores usan software libre para fines didácticos y de investigación. Las universidades suelen optar por el uso de software libre en vez de utilizar software privativo, porque satisface de una mejor manera sus necesidades de cómputo, dada su naturaleza de apertura del código y la libertad de compartir los resultados obtenidos. De forma colateral, no se tienen gastos adicionales derivados del pago de licenciamientos. El software libre no se limita a ser gratuito o de muy bajo coste, porque también tiene un valor social fundamental, puesto que la única restricción que tiene es la de conservarse libre, lo cual quiere decir que puede ser explorado, verificado, reproducido, y extendido, en todas sus capacidades, para beneficio de todos, de forma muy similar a la naturaleza de la producción de la ciencia. Computólogos, físicos, químicos, matemáticos, y otros profesionistas y científicos, utilizan software libre como herramienta de investigación y creación. Un claro ejemplo de ello es la llamada Delta Metropolitana, que es una red de súper computadoras que están en varios puntos de la Ciudad de México, en el CINESTAV, el IPN, la UAM, y la UNAM. Esa red de supe cómputo utiliza software libre para consolidar sus recursos, hacer investigación, y generar conocimiento.

2.7.19. Motivaciones del software libre

La motivación ética, abanderada por la Free Software Foundation, heredera de la cultura hacker, y partidaria del apelativo libre, que argumenta que el software es conocimiento y debe poderse difundir sin trabas.

Su ocultación es una actitud antisocial y la posibilidad de modificar programas es una forma de libertad de expresión, aunque sin olvidar una estructura jerarquizada por la meritocracia. La motivación pragmática, abanderada por la Open Source Initiative y partidaria del apelativo abierto, que argumenta ventajas técnicas y económicas, con respecto a evitar una tragedia de los anticomunes mejorando los incentivos. Aparte de estas dos grandes motivaciones, la gente que trabaja en software libre suele hacerlo por muchas otras razones, que van desde la diversión a la mera retribución económica, que es posible debido a modelos de negocio sustentables.

2.7.20. Regulación España

La Orden EDU/2341/2009, de 27 de agosto, por la que se crea el Centro Nacional de Desarrollo Curricular en Sistemas no Propietarios, tiene como finalidad el diseño, el desarrollo y la promoción de contenidos educativos digitales para colectivos educativos específicos, en el ámbito de las Tecnologías de la Información y la Comunicación, que se centra en promocionar y aplicar estrategias dirigidas a poner a disposición de los centros escolares recursos y contenidos digitales de calidad, desarrollados en software libre.

2.7.21. Regulación Venezuela

El Decreto presidencial 3390 de fecha 23 de diciembre de 2004 y publicado en La Gaceta Oficial de Venezuela nº 38095 el 28 de diciembre de 2004, establece textualmente en su artículo 1 que "La Administración Pública Nacional empleará prioritariamente Software Libre desarrollado con Estándares Abiertos, en sus sistemas, proyectos y servicios informáticos. A tales fines, todos los órganos y entes de la Administración Pública Nacional iniciarán los procesos de migración gradual y progresiva de éstos hacia el Software Libre desarrollado con Estándares Abiertos"

2.7.22. Regulación Ecuador

El Presidente Rafael Correa Delgado, el día jueves 10 de abril del 2008, firmó el Decreto N° 1014; en el cual ordena, que el software usado por las administraciones públicas del país sea software libre e implícitamente basado en estándares abiertos.

2.7.23. Regulación Uruguay

En Uruguay, a partir de 2003 (Comisión de Constitución, Códigos, Legislación General y Administración Carpeta Nº 3565 de 2003 Repartido Nº 1510 de noviembre de 2003) estudió una "Ley de Software Libre y Formatos Abiertos en el Estado", la primera versión planteaba el uso de formatos abiertos en todo el Estado y Software Libre en la educación, con una segunda versión presentada en el 2006, que ya planteaba dar preferencia al uso de Software Libre en todos los organismos del Estado. Luego dicha versión con modificaciones, fue la que recibió media sanción en la Cámara de Diputados el 19 de diciembre de 2012, la cual fue apoyada por la comunidad de software libre uruguaya. La Ley de Software Libre y Formatos Abiertos fue aprobada en diciembre de 2013. La misma establece:

1. Que el Estado deberá preferir la inversión y desarrollo en software libre sobre el privativo, salvo cuando éste no cumpla las necesidades técnicas requeridas.

2. En caso de que el Estado decida invertir en software privativo, deberá justificar las razones del gasto y argumentar su elección.

3. El Estado deberá distribuir y aceptar toda información en al menos un formato abierto, estándar y libre.

4. El intercambio de información a través de Internet deberá ser posible en al menos un programa licenciado como software libre.

2.7.24. Regulación Argentina

En la Argentina, en la Provincia de Río Negro, el 08/03/2012 el Parlamento aprobó la Ley 4747/12 que establece el empleo obligatorio del sistema de Software Libre en los tres Poderes del Estado, entes descentralizados y empresas con participación estatal.

2.7.25. Regulación Bolivia

El presidente Evo Morales Ayma, el día lunes 8 de agosto del 2011, reglamentó la Ley Nº 164 de Telecomunicaciones y TIC's para el Desarrollo de Tecnologías de Información y Comunicación.

2.7.26. Free Software Foundation

La Free Software Foundation (Fundación para el software libre) es una organización creada en octubre de 1985 por Richard Stallman y otros entusiastas del software libre con el propósito de difundir este movimiento. La Fundación para el software libre (FSF) se dedica a eliminar las restricciones sobre la copia, redistribución, entendimiento, y modificación de programas de computadoras. Con este objeto, promociona el desarrollo y uso del software libre en todas las áreas de la computación, pero muy particularmente, ayudando a desarrollar el sistema operativo GNU. En sus inicios, la FSF destinaba sus fondos principalmente a contratar programadores para que escribiesen software libre. A partir de mediados de la década de 1990 existen ya muchas compañías y autores individuales que escriben software libre, por ello los empleados y voluntarios de la FSF han centrado su trabajo fundamentalmente en asuntos legales, organizativos y promocionales en beneficio de la comunidad de usuarios de software libre. La FSF se creó con la idea original de promover el software libre. La organización desarrolla el sistema operativo **Las Licencias GNU.** La FSF elabora, mantiene y defiende la Licencia Pública General GNU (GNU GPL), la licencia de software libre más utilizada, cuya última versión es la GPLv3 que fue publicada en forma definitiva en junio de 2007.

Aparte la FSF también es responsable de la [GNU LGPL|Licencia Pública General Reducida GNU] (GNU LGPL) y la Licencia de documentación libre GNU (GNU iFDL). La FSF tiene recursos y voluntad para hacer cumplir las licencias que elabora. Pero solo puede presentar demandas, sobre software del cual posea derechos de autor. La fundación se enfrenta cada año a unas 50 violaciones de la GPL y siempre trata de evitar llegar a los tribunales.

2.7.27. Alojamiento de proyectos

La FSF aloja proyectos de software libre en su sitio web Savannah. Ofrece una de interfaz web para el hosting y el mantenimiento de las páginas web de los proyectos, seguimiento de errores, CVS, FTP, y listas de correo. Hospeda más de 2.800 proyectos.

2.7.28. Formación legal

La FSF organiza seminarios sobre los aspectos legales a tener en cuenta cuando se usa la licencia GPL.

2.7.29. Free Software Directory

Es un directorio con más de 5.000 programas que se ha comprobado que son software libre. La Unesco ayuda en la financiación de este proyecto.

2.7.30. Premios y reconocimientos

FSF Award for the Advancement of Free Software "Premio para el Avance del Software Libre de la Fundación para el Software Libre" que otorga la fundación a una persona que haya hecho una gran contribución al progreso del software libre. Free Software Award for Projects of Social Benefit.

2.7.31. GNU Press

El departamento de publicaciones de la FSF es el responsable de "publicar libros asequibles sobre informática usando licencias de libre distribución"

2.7.32. Campañas

La FSF promueve numerosas campañas en defensa y promoción del software libre:

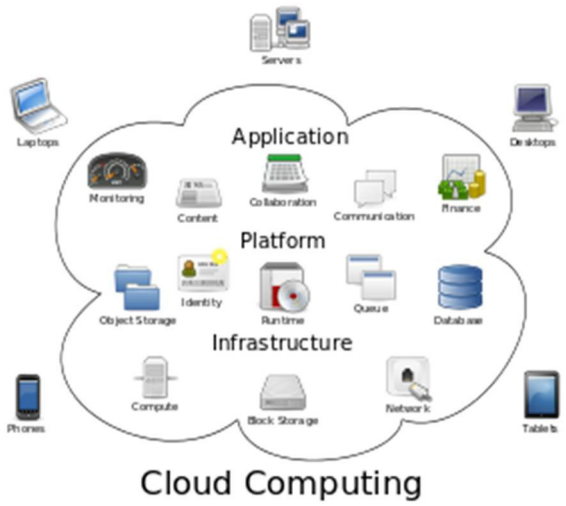
- DefectiveByDesign: Campaña para eliminar el DRM.
- PlayOgg.org: Para promocionar Ogg, una alternativa a formatos privativos como MP3 y AAC.
- Resist the imposition of Digital Restrictions Management: Campaña para "Resistir a la imposición de la gestión de Restricciones sobre lo Digital".Esto además del DRM también incluye la llamada Trusted Computing (que en la FSF denominan Computación traidora).
- Free BIOS: "BIOS libre". Campaña de apoyo al proyecto de creación de un BIOS libre.
- Hardware devices that support free software: Campaña para promocionar la compra de "hardware que de soporte al software libre".
- High Priority Free Software Projects: para llamar la atención sobre los "proyectos de software libre prioritarios".
- Encourage governments to adopt OpenDocument: Para "Promover en los gobiernos la adopción de OpenDocument"
- High Priority Free Software Projects centra la atención en proyectos que, de terminarse, podrían favorecer la adopción de software libre
- Campaign for Hardware that Supports Free Software, para promover el hardware que apoye la libertad ejecutar software libre.

La Fundación se ha posicionado en numerosas ocasiones contra las patentes de software.

2.8. Computación en la nube (Cloud Computing)

Según (Dragon JAR, 2013), la computación en la nube, concepto conocido también bajo los términos servicios en la nube, informática en la nube, nube de cómputo o nube de conceptos, del inglés Cloud Computing, es un paradigma que permite ofrecer servicios de computación a través de Internet.

Gráfico 2. 7: Computación en la nube



Fuente: (Dragon JAR, 2013)

2.8.1. Definición del Cloud Computing

El nuevo concepto de negocio en Internet también conocido como "computación en la nube". El Cloud Computing consiste en la posibilidad de ofrecer servicios a través de Internet. La computación en nube es una tecnología nueva que busca tener todos nuestros archivos e información en Internet y sin depender de poseer la capacidad suficiente para almacenar información.

El Cloud Computing explica las nuevas posibilidades de forma de negocio actual, ofreciendo servicios a través de Internet, conocidos como e-Business (negocios por Internet).

Gráfico 2. 8: Cloud Computing

Fuente: (Dragon JAR, 2013)

2.8.2. Procedimiento

Toda la información, procesos, datos, etc. se localizan dentro de la red de internet, como en una nube, así todo el mundo puede acceder a la información completa, sin poseer una gran infraestructura.

2.8.3. Ventajas del Cloud Computing

- Bajo coste: Productos gratuitos o pagos mensuales fijos por utilización, sin costes adicionales, dado que no hay que invertir en gran infraestructura, ni en licencias.
- Seguridad: Los datos siempre están seguros.
- No hay necesidades de poseer una gran capacidad de almacenamiento.
- Mayor rapidez en el trabajo al estar basado en web.
- Información a tiempo real.
- Fuerte inversión en innovación.
- Acceso a toda la información.
- Acceso cuando quiera y donde quiera, sólo con una conexión a Internet.

2.8.4. Introducción al Cloud Computing

En este tipo de computación todo lo que puede ofrecer un sistema informático se ofrece como servicio, de modo que los usuarios puedan acceder a los servicios disponibles "en la nube de Internet" sin conocimientos (o, al menos sin ser expertos) en la gestión de los recursos que usan.

Según la IEEE Computer Society, es un paradigma en el que la información se almacena de manera permanente en servidores de Internet y se envía a cachés temporales de cliente, lo que incluye computadoras de escritorio, centros de ocio, portátiles, etc. La computación en la nube son servidores desde Internet encargados de atender las peticiones en cualquier momento. Se puede tener acceso a su información o servicio, mediante una conexión a internet desde cualquier dispositivo móvil o fijo ubicado en cualquier lugar. Sirven a sus usuarios desde varios proveedores de alojamiento repartidos frecuentemente por todo el mundo. Esta medida reduce los costes, garantiza un mejor tiempo de actividad y que los sitios web sean invulnerables a los hackers, a los gobiernos locales y a sus redadas policiales. "Cloud Computing" es un nuevo modelo de prestación de servicios de negocio y tecnología, que permite incluso al usuario acceder a un catálogo de servicios estandarizados y responder con ellos a las necesidades de su negocio, de forma flexible y adaptativa, en caso de demandas no previsibles o de picos de trabajo, pagando únicamente por el consumo efectuado, o incluso gratuitamente en caso de proveedores que se financian mediante publicidad o de organizaciones sin ánimo de lucro. El cambio que ofrece la computación desde la nube es que permite aumentar el número de servicios basados en la red. Esto genera beneficios tanto para los proveedores, que pueden ofrecer, de forma más rápida y eficiente, un mayor número de servicios, como para los usuarios que tienen la posibilidad de acceder a ellos, disfrutando de la “transparencia” e inmediatez del sistema y de un modelo de pago por consumo. Así mismo, el consumidor ahorra los costes salariales o los costes en inversión económica (locales, material especializado, etc.). Computación en nube consigue aportar estas ventajas, apoyándose sobre una infraestructura tecnológica dinámica que se caracteriza, entre otros factores, por un alto grado de automatización, una rápida movilización de los recursos, una elevada capacidad de adaptación para atender a una demanda variable, así como virtualización avanzada y un precio flexible en función del consumo realizado, evitando además el uso fraudulento del software y la piratería.

La computación en nube es un concepto que incorpora el software como servicio, como en la Web 2.0 y otros conceptos recientes, también conocidos como tendencias tecnológicas, que tienen en común el que confían en Internet para satisfacer las necesidades de cómputo de los usuarios.

2.8.5. Comienzos del Cloud Computing

El concepto de la computación en la nube empezó en proveedores de servicio de Internet a gran escala, como Google, Amazon AWS, Microsoft y otros que construyeron su propia infraestructura. De entre todos ellos emergió una arquitectura: un sistema de recursos distribuidos horizontalmente, introducidos como servicios virtuales de TI escalados masivamente y manejados como recursos configurados y mancomunados de manera continua. Este modelo de arquitectura fue inmortalizado por George Gilder en su artículo de octubre 2006 en la revista Wired titulado “Las fábricas de información”. Las granjas de servidores, sobre las que escribió Gilder, eran similares en su arquitectura al procesamiento “grid” (red, parrilla), pero mientras que las redes se utilizan para aplicaciones de procesamiento técnico débilmente acoplados (loosely coupled), un sistema compuesto de subsistemas con cierta autonomía de acción, que mantienen una interrelación continua entre ellos, este nuevo modelo de nube se estaba aplicando a los servicios de Internet.

2.8.6. Historia del Cloud Computing

El concepto fundamental de la entrega de los recursos informáticos a través de una red global tiene sus raíces en los años sesenta. La idea de una "red de computadoras intergaláctico" fue introducido en los años sesenta por JCR Licklider, quien era responsable de permitir el desarrollo de ARPANET (Advanced Research Projects Agency Network) en 1969. Su visión era que todo el mundo pudiese estar interconectado y poder acceder a los programas y datos desde cualquier lugar, explicó Margaret Lewis, directora de marketing de producto de AMD. "Es una visión que se parece mucho a lo que llamamos Cloud Computing". Otros expertos atribuyen el concepto científico de la computación en nube a John McCarthy, quien propuso la idea de la computación como un servicio público, de forma similar a las empresas de servicios que se remontan a los años sesenta. John McCarthy, 1960: "Algún día la computación podrá ser organizada como un servicio público". Desde los años sesenta, la computación en nube se ha desarrollado a lo largo de una serie de líneas. La Web 2.0 es la evolución más reciente.

Sin embargo, como Internet no empezó a ofrecer ancho de banda significativo hasta los años noventa, la computación en la nube ha sufrido algo así como un desarrollo tardío. Uno de los primeros hitos de la computación en nube es la llegada de Salesforce.com en 1999, que fue pionero en el concepto de la entrega de aplicaciones empresariales a través de una página web simple. La firma de servicios allanó el camino para que tanto especialistas como empresas tradicionales de software pudiesen publicar sus aplicaciones a través de Internet. El siguiente desarrollo fue Amazon Web Services en 2002, que prevé un conjunto de servicios basados en la nube, incluyendo almacenamiento, computación e incluso la inteligencia humana a través del Amazon Mechanical Turk. Posteriormente en 2006, Amazon lanzó su Elastic Compute Cloud (EC2) como un servicio comercial que permite a las pequeñas empresas y los particulares alquilar computadoras en los que se ejecuten sus propias aplicaciones informáticas. 8 "Amazon EC2/S3 fue el que ofreció primero servicios de infraestructura en la nube totalmente accesibles", dijo Jeremy Allaire, CEO de Brightcove, que proporciona su plataforma SaaS de vídeo en línea a las estaciones de televisión de Reino Unido y periódicos. George Gilder, 2006: "El PC de escritorio está muerto. Bienvenido a la nube de Internet, donde un número enorme de instalaciones a lo largo de todo el planeta almacenarán todos los datos que usted podrá usar alguna vez en su vida". Otro hito importante se produjo en 2009, cuando Google entre otros, empezaron a ofrecer aplicaciones basadas en navegador. Servicios, como Google Apps. "La contribución más importante a la computación en nube ha sido la aparición de "aplicaciones asesinas" de los gigantes de tecnología como Microsoft y Google. Cuando dichas compañías llevan a cabo sus servicios de una manera que resulta segura y sencilla para el consumidor, el efecto "pasar la pelota" en sí, crea un sentimiento de mayor aceptación de los servicios online", dijo Dan Germain, jefe de la oficina de tecnología en IT proveedor de servicios Cobweb Solutions. Otro de los factores clave que han permitido evolucionar a la computación en la nube según el británico y pionero en computación en la nube Jamie Turner, han sido la tecnologías de virtualización, el desarrollo del universal de alta velocidad de ancho de banda, y normas universales de interoperabilidad de software. Turner añadió: "A medida que la computación en nube se extiende, su alcance va más allá de un puñado de usuarios de Google Docs. Sólo podemos empezar a imaginar su ámbito de aplicación y alcance. Casi cualquier cosa puede ser utilizado en la nube".

2.8.7. Características del Cloud Computing

La computación en nube presenta las siguientes características clave:

- **Agilidad** mejora con la capacidad de los usuarios para volver a la provisión de recursos de infraestructura tecnológica.

- **Coste**: los proveedores de computación en la nube afirman que los costes se reducen. Un modelo de prestación pública en la nube convierte los gastos de capital en gastos de funcionamiento. Ello reduce barreras de entrada, ya que la infraestructura se proporciona típicamente por una tercera parte y no tiene que ser adquirida por una única sola vez o tareas informáticas intensivas infrecuentes.

- **Escalabilidad y elasticidad** aprovisionamiento de recursos sobre una base de autoservicio en casi en tiempo real, sin que los usuarios necesiten cargas de alta duración.

- Dispositivo e **independencia de la ubicación** permite a los usuarios acceder a los sistemas utilizando un navegador web, independientemente de su ubicación o del dispositivo que utilice (por ejemplo, PC, teléfono móvil).

- La **tecnología de virtualización** permite compartir servidores y dispositivos de almacenamiento y una mayor utilización. Las aplicaciones pueden ser fácilmente migradas de un servidor físico a otro.

- **Rendimiento**. Los sistemas en la nube controlan y optimizan el uso de los recursos de manera automática, dicha característica permite un seguimiento, control y notificación del mismo. Esta capacidad aporta transparencia tanto para el consumidor o el proveedor de servicio.

- La **seguridad** puede mejorar debido a la centralización de los datos. La seguridad es a menudo tan bueno o mejor que otros sistemas tradicionales, en parte porque los proveedores son capaces de dedicar recursos a la solución de los problemas de seguridad que muchos clientes no pueden permitirse el lujo de abordar.

- **Mantenimiento** de las aplicaciones de computación en la nube es más sencillo, ya que no necesitan ser instalados en la computadora de cada usuario y se puede acceder desde diferentes lugares.

2.8.8. Beneficios del Cloud Computing

- Integración probada de servicios Red. Por su naturaleza, la tecnología de Cloud Computing se puede integrar con mucha mayor facilidad y rapidez con el resto de las aplicaciones empresariales (tanto software tradicional como Cloud Computing basado en infraestructuras), ya sean desarrolladas de manera interna o externa.

- Prestación de servicios a nivel mundial. Las infraestructuras de Cloud Computing proporcionan mayor capacidad de adaptación, recuperación completa de pérdida de datos (con copias de seguridad) y reducción al mínimo de los tiempos de inactividad.

- Una infraestructura 100% de Cloud Computing permite al proveedor de contenidos o servicios en la nube prescindir de instalar cualquier tipo de software, ya que éste es provisto por el proveedor de la infraestructura o la plataforma en la nube. Un gran beneficio del Cloud Computing es la simplicidad y el hecho de que requiera mucha menor inversión para empezar a trabajar.

- Implementación más rápida y con menos riesgos, ya que se comienza a trabajar más rápido y no es necesaria una gran inversión. Las aplicaciones del Cloud Computing suelen estar disponibles en cuestión de días u horas en lugar de semanas o meses, incluso con un nivel considerable de personalización o integración.

- Actualizaciones automáticas que no afectan negativamente a los recursos de TI. Al actualizar a la última versión de las aplicaciones, el usuario se ve obligado a dedicar tiempo y recursos para volver a personalizar e integrar la aplicación. Con el Cloud Computing no hay que decidir entre actualizar y

conservar el trabajo, dado que esas personalizaciones e integraciones se conservan automáticamente durante la actualización.

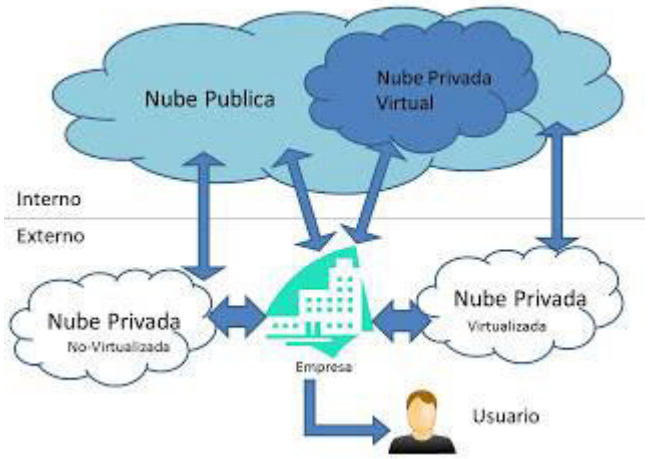
- Contribuye al uso eficiente de la energía. En este caso, a la energía requerida para el funcionamiento de la infraestructura. En los datacenters tradicionales, los servidores consumen mucha más energía de la requerida realmente. En cambio, en las nubes, la energía consumida es sólo la necesaria, reduciendo notablemente el desperdicio.

2.8.9. Desventajas del Cloud Computing

- La centralización de las aplicaciones y el almacenamiento de los datos origina una interdependencia de los proveedores de servicios.
- La disponibilidad de las aplicaciones está sujeta a la disponibilidad de acceso a Internet.
- Los datos "sensibles" del negocio no residen en las instalaciones de las empresas, lo que podría generar un contexto de alta vulnerabilidad para la sustracción o robo de información.
- La confiabilidad de los servicios depende de la "salud" tecnológica y financiera de los proveedores de servicios en nube. Empresas emergentes o alianzas entre empresas podrían crear un ambiente propicio para el monopolio y el crecimiento exagerado en los servicios.
- La madurez funcional de las aplicaciones hace que continuamente estén modificando sus interfaces, por lo cual la curva de aprendizaje en empresas de orientación no tecnológica tenga unas pendientes significativas, así como su consumo automático por aplicaciones.
- Seguridad. La información de la empresa debe recorrer diferentes nodos para llegar a su destino, cada uno de ellos (y sus canales) son un foco de inseguridad. Si se utilizan protocolos seguros, HTTPS por ejemplo, la velocidad total disminuye debido a la sobrecarga que éstos requieren.
- Escalabilidad a largo plazo. A medida que más usuarios empiecen a compartir la infraestructura de la nube, la sobrecarga en los servidores de los proveedores aumentará, si la empresa no posee un esquema de crecimiento óptimo puede llevar a degradaciones en el servicio o altos niveles de jitter.
- Privacidad. La información queda expuesta a terceros que pueden copiarla o acceder a ella.

2.8.10. Tipos de nubes

Gráfico 2. 9: Tipos de nubes



Fuente: E-conomic 2014, basado en el concepto de Cloud Computing

- Una nube pública es una nube computacional mantenida y gestionada por terceras personas no vinculadas con la organización. En este tipo de nubes tanto los datos como los procesos de varios clientes se mezclan en los servidores, sistemas de almacenamiento y otras infraestructuras de la nube. Los usuarios finales de la nube no conocen que trabajos de otros clientes pueden estar corriendo en el mismo servidor, red, sistemas de almacenamiento, etc aplicaciones, almacenamiento y otros recursos están disponibles al público a través el proveedor de servicios que es propietario de toda la infraestructura en sus centros de datos; el acceso a los servicios solo se ofrece de manera remota, normalmente a través de Internet.

- Las nubes privadas son una buena opción para las compañías que necesitan alta protección de datos y ediciones a nivel de servicio. Las nubes privadas están en una infraestructura bajo demanda gestionada para un solo cliente que controla qué aplicaciones debe ejecutarse y dónde. Son propietarios del servidor, red, y disco y pueden decidir qué usuarios están autorizados a utilizar la infraestructura. Al administrar internamente estos servicios, las empresas tienen la ventaja de mantener la privacidad de su información y permitir unificar el acceso a las aplicaciones corporativas de sus usuarios.

- Las nubes híbridas combinan los modelos de nubes públicas y privadas. Usted es propietario de unas partes y comparte otras, aunque de una manera controlada. Las nubes híbridas ofrecen la promesa del escalado aprovisionada externamente, en-demanda, pero añaden la complejidad de determinar cómo distribuir las aplicaciones a través de estos ambientes diferentes. Las empresas pueden sentir cierta atracción por la promesa de una nube híbrida, pero esta opción, al menos inicialmente, estará probablemente reservada a aplicaciones simples sin condicionantes, que no requieran de ninguna sincronización o necesiten bases de datos complejas. Se unen mediante la tecnología pues permiten enviar datos o aplicaciones entre ellas. Un ejemplo son los sistemas de correo electrónico empresarial.

- Nube comunitaria. De acuerdo con Joyanes Aguilar, 2012, el Instituto Nacional de Estándares y tecnología (NITS por sus siglas en inglés) define este modelo como aquel que se organiza con la finalidad de servir a una función o propósito común (seguridad, política…), y son administradas por las organizaciones constituyentes o terceras partes.

2.8.11. Aspectos de seguridad

La seguridad en la computación en la nube, puede ser tan buena o mejor que la que disponíamos en los sistemas tradicionales, porque los proveedores son capaces de proporcionar recursos, que resuelvan problemas de seguridad que muchos clientes no pueden afrontar. Sin embargo, la seguridad todavía sigue siendo un asunto importante, cuando los datos tienen un matiz confidencial. Esto atrasa la adopción de la computación en la nube hasta cierto punto.

- **Seguridad como servicio:** En el entorno de la nube, la seguridad provista por los proveedores, se pueden distinguir dos métodos: El primer método, es que cualquiera puede cambiar sus métodos de entrega incluidos en los servicios de la nube. El segundo método es que los proveedores de servicio de la nube proveen seguridad solo como servicio en la nube, con información de seguridad de las compañías.

- **Seguridad del explorador:** En el entorno de la nube, los servidores remotos son usados para la computación. Los nodos del cliente se usan solo para entrada/salida de operaciones, y para la autorización y autenticación de la información en la nube. Un navegador web estándar es una plataforma normalmente utilizada para todos los usuarios del mundo. Esto puede ser catalogado en dos tipos diferentes: Software como servicio (SaaS), Aplicaciones Web, o Web 2.0. Transport Layer Security (TLS), se suele emplear para la encriptación de datos y la autentificación del host.

2.8.12. Autenticación

En el entorno de la nube, la base para el control de acceso es la autenticación, el control de acceso es más importante que nunca desde que la nube y todos sus datos son accesibles para todo el mundo a través de internet. Trusted Platform Module (TPM) es extensamente utilizado y un sistema de autenticación más fuerte que el nombre de usuario y la contraseña. Trusted Computing Groups (TCG's) es un estándar sobre la autorización de usuarios y otras herramientas de seguridad de comunicación en tiempo real entre el proveedor y el cliente.

2.8.13. Pérdida de gobernanza

En las infraestructuras de la nube, el cliente necesariamente cede el control al proveedor (Cloud Provider) en un número de asuntos, los cuáles afectan a la seguridad. Al mismo tiempo, el acuerdo de nivel de servicio no suele tener el cometido de surtir este tipo de servicios en la parte del proveedor de la nube, dejando una brecha en las defensas de seguridad.

2.8.14. Lock-In

Esta es una pequeña oferta en este tipo de herramientas, los procedimientos o estándares de formatos de datos o interfaces de servicios que podrían garantizar los datos, las aplicaciones y el servicio de portabilidad.

Esto puede hacer difícil para el cliente migrar de un proveedor a otro, o migrar los datos y servicios de nuevo a otro entorno informático. Esto introduce una particular dependencia en el proveedor de la nube para la provisión del servicio, especialmente a la portabilidad de los datos, el aspecto más fundamental.

2.8.15. Protección de los datos

La computación en la nube pone en riesgo la protección de datos para los usuarios de la nube y sus proveedores. En muchos casos, ocasiona dificultades para el proveedor (in el rol del controlador de la información) para asegurar la efectividad práctica del manejo de los datos del proveedor de la nube y para cerciorar que los datos van por el camino correcto.

Este problema se suele agravar en casos de múltiples transferencias de datos, por ejemplo entre sistemas federados. Por otra parte, algunos proveedores de la nube, proporcionan información de sus prácticas de cercenamiento de datos.

También hay algunas ofertas de certificaciones en el procesamiento de datos, las actividades de seguridad, y los controles de datos que tienen lugar; ejemplo, la certificación SAS70. Las corrientes de datos de internet, están unidas al malware y de paquetes señuelo para meter al usuario en una desconocida participación en actividades delictivas.

2.8.16. Limitaciones

Algunas limitaciones que están retrasando un poco a la computación en la nube son algunas de las siguientes:

- **Pérdidas de datos /Fuga:** Los esfuerzos para controlar la seguridad de los datos de la computación en la nube no son muy buenos; acordadamente con el acceso de control API y la generación de las claves, almacenamiento y configuración de deficiencias, permiten resultados en pérdidas de datos, y también permiten una escasa política de destrucción de datos. La fuga, es la causa de la escasa vital política de destrucción de datos.

- **Dificultad de valorar la fiabilidad de los proveedores:** El proveedor de servicio de computación en la nube, controla la fuerza con la que se pueden realizar los esfuerzos, que actualmente se solían usar para controlar los accesos a los datos, los cuáles son diferentes en muchos proveedores y en estas circunstancias; pero no todo es suficiente, las compañías necesitan una evaluación de los proveedores y proponer qué y cómo filtran el programa personal.
- **Los mecanismos de autentificación no son muy fuertes:** En la nube, hay muchísimos datos, aplicaciones y recursos almacenados. La computación en la nube es muy débil en los mecanismos de autentificación, por lo tanto el atacante puede fácilmente obtener la cuenta de usuario cliente y acceder a la máquina virtual .
- **Investigación:** Multitud de universidades, institutos, proveedores e instituciones gubernamentales están invirtiendo en computación en la nube:

- En Octubre de 2007, la Inicitativa Académica de Computación en la Nube (ACCI) se anunció como un proyecto multi-universitario dedicado a orientar técnicamente a estudiantes en sus desafios con la computación en la nube

- En Abril de 2009, UC Santa Barbara lanzó la primera plataforma de codigo abierto, AppScale, capaz de ejecutar aplicaciones de Google App Engine a escala en multitud de infraestructuras.

- En Abril de 2009, surgió el laboratario de computación en la nube de St Andrews, enfocado en la investigación de esta nueva área. Único en el Reino Unido, StaCC pretende convertirse en un centro internacional de excelencia para la investigación y docencia en computación en la nube, además, proporciona consejo e información a empresas interesadas en servicios en la nube.

- En Enero de 2011, IRMOS EU financió el desarrollo de una plataforma en la nube en tiempo real, permitiendo aplicaciones interactivas en infraestructuras de la nube.

- En Diciembre de 2010, el proyecto de investigación TrustCloud fue iniciado por los laboratorios HP Singapur para abordar la transparencia y la rendición de cuentas de la computación en nube a través de detectives, los enfoques centrados en los datos encapsulados en un TrustCloud marco de cinco capas. El equipo identificó la necesidad de monitorizar los ciclos de vida y las transferencias en la nube, que conduce al abordaje de cuestiones esenciales de seguridad, como las fugas de datos, la rendición de cuentas y las transferencias de datos entre países mediante transacciones en la nube

- En Junio de 2011, dos universidades de la India University of Petroleum and Energy Studies y University of Technology and Management introdujeron una asignatura de computación en la nube en colaboración con IBM.

- En Julio 2011, se dio inicio al proyecto de alto rendimiento de computación en la nube (HPCCLoud) con el objetivo de investigar mejoras en el rendimiento en entornos de aplicaciones científicas en la nube.

- En Julio de 2011, la asociación de la industria en telecomunicaciones elaboró un documento para analizar los desafíos de integración y oportunidades entre los servicios en la nube y los servicios de comunicación tradicionales en los Estados Unidos.

- En Diciembre de 2011, el proyecto VISION Cloud financiado por la UE propuso una arquitectura y una implementación para los servicios de uso intensivo de datos con el objetivo de proporcionar una infraestructura de almacenamiento virtualizada.

- En Octubre de 2012, el Centro de desarrollo para la Computación Avanzada publicó un software llamado "Meghdoot" de código abierto, de servicio en la nube.

- En Febrero de 2013, el proyecto BonFire lanzó un centro de experimentación y pruebas en la nube. La instalación ofrece acceso transparente a los recursos de la nube, con el control y la observabilidad necesaria para diseñar las futuras tecnologías en la nube.

Aplicaciones que utilizan tecnología Cloud Computing

- Salesforce.com – Desarrollado por Salesforce.com Inc
- Box (Sitio Web) – Desarrollado por Box Inc
- OwnCloud - desarrollado por OwnCloud
- Dropbox - desarrollado por Dropbox
- Google Drive - desarrollado por Google
- Wuala - desarrollado por LaCie
- iCloud - desarrollado por Apple
- OneDrive - desarrollado por Microsoft (Antes SkyDrive)
- Campaign Cloud - desarrollado por ElectionMall Technologies
- Ubuntu One - desarrollado por Canonical
- Doitle ajaxplorer - desarrollado por Doitle
- SugarSync - desarrollado por SugarSync

2.8.17. Lenguajes de programación

Existen varias interfaces de programación de aplicaciones que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Pascal, Delphi (vía dbExpress), Eiffel, Smalltalk, Java (con una implementación nativa del driver de Java), Lisp, Perl, PHP, Python, Ruby, Gambas, REALbasic (Mac y Linux), (x)Harbour (Eagle1), FreeBASIC, y Tcl; cada uno de estos utiliza una interfaz de programación de aplicaciones específica.

También existe una interfaz ODBC, llamado MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL. También se puede acceder desde el sistema SAP, lenguaje ABAP.

2.9. MySQL

Para (Fredd Judge, 2012), MySQL es un sistema de administración de bases de datos. Una base de datos es una colección estructurada de tablas que contienen datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto volumen de información en una red corporativa. Para agregar, acceder a y procesar datos guardados en un computador, usted necesita un administrador como MySQL Server. Dado que las computadoras son muy buenas manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.

MySQL es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido. MySQL es software de fuente abierta. Fuente abierta significa que es posible para cualquier persona usarlo y modificarlo. Cualquier persona puede bajar el código fuente de MySQL y usarlo sin pagar. Cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir qué puede hacer y qué no puede hacer con el software en diferentes situaciones. MySQL es usado por muchos sitios web grandes y populares, como Wikipedia, Google (aunque no para búsquedas), Facebook, Twitter, Flickr y YouTube

2.9.1. Aplicaciones de MySQL

MySQL es muy utilizado en aplicaciones web, como Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación.

2.9.2. Plataformas en donde funciona MySQL

MySQL funciona sobre múltiples plataformas, incluyendo:

- AIX
- BSD y FreeBSD
- HP-UX
- Kurisu OS
- GNU/Linux
- Mac OS X
- NetBSD
- OpenBSD
- OS/2 Warp
- QNX
- SGI IRIX
- SunOS y Solaris
- SCO OpenServer y UnixWare
- Tru64
- eBD
- Windows XP, Windows 7, Windows 8 y Windows Server (2000, 2003, 2008 y 2012).
- OpenVMS18

2.9.3. Características adicionales de MySQL

- Usa GNU Automake, Autoconf, y Libtool para portabilidad
- Uso de multihilos mediante hilos del kernel.
- Usa tablas en disco b-tree para búsquedas rápidas con compresión de índice
- Tablas hash en memoria temporales
- El código MySQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL.
- Completo soporte para operadores y funciones en cláusulas select y where.
- Completo soporte para cláusulas group by y order by, soporte de funciones de agrupación
- Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
- Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).
- Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma y sistema. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix.
- En MySQL 5.0, los clientes y servidores Windows se pueden conectar usando memoria compartida.
- MySQL contiene su propio paquete de pruebas de rendimiento proporcionado con el código fuente de la distribución de MySQL.
- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferentes velocidades de operación, soporte físico, capacidad, distribución geográfica, transacciones.
- Transacciones y claves foráneas.
- Replicación y búsqueda de indexación de campos de texto.

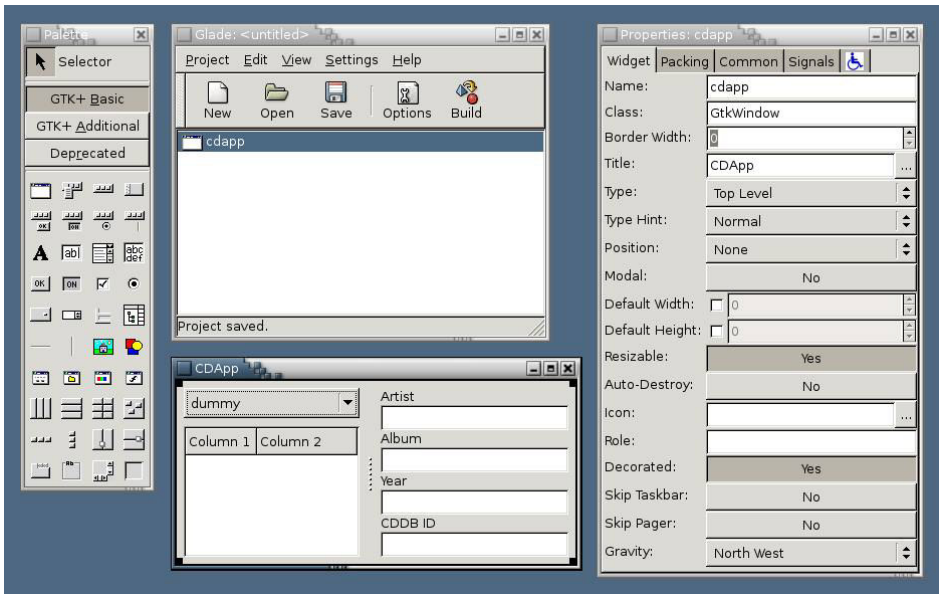
2.9.4. Licencia MySQL

La licencia GNU GPL de MySQL obliga a que la distribución de cualquier producto derivado (aplicación) se haga bajo esa misma licencia. Si un desarrollador desea incorporar MySQL en su producto pero desea distribuirlo bajo otra licencia que no sea la GNU GPL, puede adquirir una licencia comercial de MySQL que le permite hacer justamente eso.

2.10. Glade

Según (Groth & Skandier, 2005) Glade o Glade Interface Designer, que significa "Diseñador de interfaces Glade", es una herramienta de desarrollo visual de interfaces gráficas mediante GTK/GNOME. Es independiente del lenguaje de programación y predeterminadamente no genera código fuente sino un archivo XML (ver sección GtkBuilder). La posibilidad de generar automáticamente código fuente fue descontinuada desde Glade versión 3. Aunque tradicionalmente se ha utilizado de forma independiente, está totalmente integrado en Anjuta 2. Se encuentra bajo la licencia GPL. Para QT existe un proyecto similar, QtDesigner

Gráfico 2. 10: Glade



Fuente: (Groth & Skandier, 2005)

El primer lanzamiento de Glade, la versión 0.1, se hizo el 18 de abril de 1998. Y Glade-3 se lanzó el 12 de agosto de 2006. Según el sitio web de Glade, las diferencias más notorias para el usuario final son:

- “Deshacer” y “rehacer” disponible para todas las operaciones.
- Permite abrir varios proyectos simultáneamente.
- Remoción de la generación automática de código fuente.
- Ayuda contextual mediante DevHelp.

Sin embargo, la mayoría de las diferencias son internas. Glade-3 fue reescrito completamente, para poder tomar ventaja de las nuevas características de GTK+ 2 y el sistema GObject (Glade-3 comenzó a escribirse antes de que Glade-2 fuese portado a GTK+ 2). Por lo tanto el código principal de Glade-3 es más pequeño y permite nuevas cosas interesantes, incluyendo:

- Catálogo de widgets "enchufables" ("pluggable" widgets). Esto significa que bibliotecas externas pueden proveer su conjunto de widgets en tiempo de ejecución y Glade los detectará. De hecho, Glade-3 soporta sólo widgets estándar de GTK+; los widgets GNOME UI y DB son provistos por separado.

- Las herramientas de Glade (paleta, editor, etc.) son implementadas como widgets. Esto permite una fácil integración con IDEs como Anjuta o Scaffold, y hace que sea más fácil cambiar la interfaz.

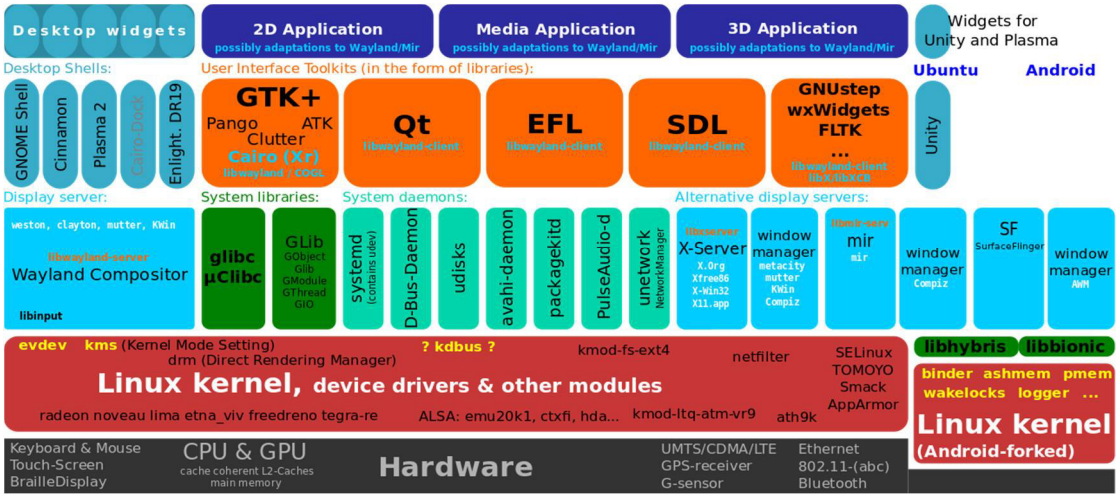
2.11. GtkBuilder

GtkBuilder es un formato XML que Glade usa para almacenar los elementos de las interfaces diseñadas. Estos archivos pueden emplearse para construirla en tiempo de ejecución mediante el objeto GtkBuilder de GTK+. GladeXML era el formato que se usaba en conjunto con la biblioteca *libglade* (ambos obsoletos en favor de GtkBuilder).

2.11.1. GTK+

GTK+ o **The GIMP Toolkit** es un conjunto de bibliotecas multiplataforma para desarrollar interfaces gráficas de usuario (GUI), principalmente para los entornos gráficos GNOME, XFCE y ROX aunque también se puede usar en el escritorio de Windows, Mac OS y otros. Inicialmente fueron creadas para desarrollar el programa de edición de imagen GIMP, sin embargo actualmente se usan bastante por muchos otros programas en los sistemas GNU/Linux. Junto a Qt es una de las bibliotecas más populares para X Window System. GTK+ se ha diseñado para permitir programar con lenguajes como C, C++, C#, Fortran, Java, Ruby, Perl, PHP o Python. Licenciado bajo los términos de LGPL, GTK+ es software libre y es parte del proyecto GNU.

Gráfico 2. 11: Estructura GTK +



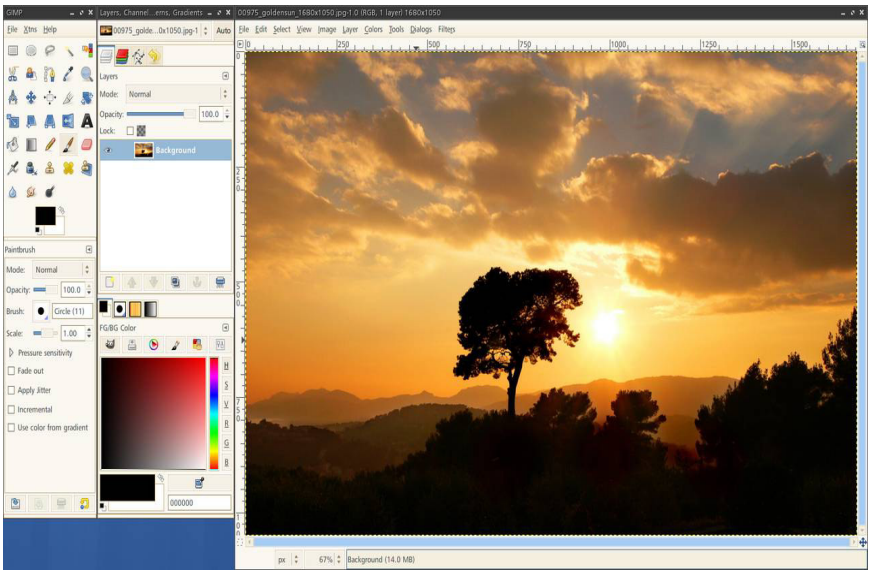
Fuente: GTK+

2.11.2. Bibliotecas de GTK+

GTK+ se basa en varias bibliotecas desarrolladas por el equipo de GTK+ y de GNOME:

- **GLib**. Biblioteca de bajo nivel estructura básica de GTK+ y GNOME. Proporciona manejo de estructura de datos para C, portabilidad, interfaces para funcionalidades de tiempo de ejecución como ciclos, hilos, carga dinámica o un sistema de objetos.
- Screenshot de GIMP 2.0. GTK+ es responsable de administrar los componentes de la interfaz del programa, incluyendo los menús, botones, campos de entrada, etc.
- **GTK**. Biblioteca la cual realmente contiene los objetos y funciones para crear la interfaz de usuario. Maneja widgets como ventanas, botones, menús, etiquetas, deslizadores, pestañas, etc.
- **GDK**. Biblioteca que actúa como intermediario entre gráficos de bajo nivel y gráficos de alto nivel.
- **ATK**. Biblioteca para crear interfaces con características de una gran accesibilidad muy importante para personas discapacitadas o minusválidas. Pueden usarse utilerías como lupas de aumento, lectores de pantalla, o entradas de datos alternativas al clásico teclado o ratón.
- **Pango**. Biblioteca para el diseño y renderizado de texto, hace hincapié especialmente en la internacionalización. Es el núcleo para manejar las fuentes y el texto de GTK+2.
- **Cairo**. Biblioteca de renderizado avanzado de controles de aplicación.

Gráfico 2. 12: Bibliotecas de GTK+



Fuente: Autor

2.11.3. Aplicaciones que usan GTK+, Entornos que utilizan GTK+

- GNOME está basado en GTK+, lo que significa que los programas de GNOME usan GTK+
- Xfce está basado en GTK+
- LXDE está basado en GTK+, significa "Lightweight X11
- Desktop Environment"
- ROX Desktop un escritorio ligero, con características de la GUI de RISC OS
- GPE Palmtop Environment
- Maemo (Nokia's Internet-tablet framework)
- Access Linux Platform (sucesor de la plataforma Palm OS PDA)
- One Laptop Per Child usa GTK+ y PyGTK

Los entornos de escritorio no son necesarios para ejecutar los programas GTK+. Si las bibliotecas que requiere el programa están instaladas, un programa GTK+ puede ser ejecutado por encima de otros entornos basadas en X11 como KDE o cualquier otro entorno, lo que incluye Mac OS X, si X11.app está instalado. GTK+ también puede ejecutarse en Microsoft Windows, es utilizado por algunas aplicaciones populares multiplataforma como Pidgin y GIMP. wxWidgets, un toolkit gráfico multiplataforma usa GTK+ en sistemas tipo Unix. Algunos de los ports más inusuales incluyen directfb y ncurses.

2.11.4. Decoradores de ventanas

- Metacity hasta su versión 2.32 y Xfwm4 usan GTK+ 2.
- Metacity desde la versión 2.34 y la versión 3 de GNOME en adelante usan GTK+3

2.11.5. Aplicaciones de GTK+

Algunas aplicaciones que usan GTK+ para desarrollar sus interfaces de usuario incluyen:

- AbiWord - Procesador de textos.
- CinePaint (ex FilmGimp) - Editor de gráficos animados en HDR.
- Ekiga (ex GnomeMeeting) - Software telefónico VoIP H.323/SIP.
- Evolution - Cliente de correo electrónico.
- Firefox - Navegador web.
- GIMP - Editor de gráficos.
- Gnumeric - Programa de hoja de cálculo.
- Chromium - Navegador Web basado en WebKit y desarrollado en gran medida por Google.
- GRAMPS - Software de genealogía.
- Inkscape - Editor de gráficos vectoriales SVG.
- K-3D - Programa de modelado 3D libre.
- Marionnet - Un simulador de red interactivo.
- Midori - Navegador Web ligero,forma parte del proyecto XFCE.
- Nero Linux - Un programa para la edición de discos.
- Pidgin - Cliente de mensajería instantánea.
- VMware Player - Máquina virtual.
- Wireshark - Capturador y analizador de paquetes de redes computacionales.

2.12. Python

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma.

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License, que es compatible con la Licencia pública general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores. Python fue creado a finales de los ochenta por Guido van Rossum en el Centro para las Matemáticas y la Informática (CWI, Centrum Wiskunde & Informática), en los Países Bajos, como un sucesor del lenguaje de programación ABC, capaz de manejar excepciones e interactuar con el sistema operativo Amoeba. El nombre del lenguaje proviene de la afición de su creador por los humoristas británicos Monty Python. Van Rossum es el principal autor de Python, y su continuo rol central en decidir la dirección de Python es reconocido, refiriéndose a él como Benevolente Dictador Vitalicio (en inglés: Benevolent Dictator for Life, BDFL). Python alcanzó la versión 1.0 en enero de 1994. Una característica de este lanzamiento fueron las herramientas de la programación funcional: lambda, reduce, filter y map. Van Rossum explicó que "hace 12 años, Python adquirió lambda, reduce(), filter() y map(), cortesía de un pirata informático de Lisp que las extrañaba y que envió parches". El donante fue Amrit Prem; no se hace ninguna mención específica de cualquier herencia de Lisp en las notas de lanzamiento. La última versión liberada proveniente de CWI fue Python 1.2. En 1995, van Rossum continuó su trabajo en Python en la Corporation for National Research Initiatives (CNRI) en Reston, Virginia, donde lanzó varias versiones del software.

En el año 2000, el equipo principal de desarrolladores de Python se cambió a BeOpen.com para formar el equipo BeOpen PythonLabs. CNRI pidió que la versión 1.6 fuera pública, continuando su desarrollo hasta que el equipo de desarrollo abandonó CNRI; su programa de lanzamiento y el de la versión 2.0 tenían una significativa cantidad de traslapo. Python 2.0 fue el primer y único lanzamiento de BeOpen.com. Después que Python 2.0 fuera publicado por BeOpen.com, Guido van Rossum y los otros desarrolladores de PythonLabs se unieron en Digital Creations. Adicionalmente, tomó una característica mayor del lenguaje de programación funcional Haskell: listas por comprensión. La sintaxis de Python para esta construcción es muy similar a la de Haskell, salvo por la preferencia de los caracteres de puntuación en Haskell, y la preferencia de Python por palabras claves alfabéticas.

Python 2.0 introdujo además un sistema de recolección de basura capaz de recolectar referencias cíclicas. Posterior a este doble lanzamiento, y después que van Rossum dejó CNRI para trabajar con desarrolladores de software comercial, quedó claro que la opción de usar Python con software disponible bajo GNU GPL era muy deseable. La licencia usada entonces, la Python License, incluía una cláusula estipulando que la licencia estaba gobernada por el estado de Virginia, por lo que, bajo la óptica de los abogados de Free Software Foundation (FSF), se hacía incompatible con GPL. CNRI y FSF se relacionaron para cambiar la licencia de software libre de Python para hacerla compatible con GPL. En el año 2001, van Rossum fue premiado con FSF Award for the Advancement of Free Software. Python 1.6.1 es esencialmente el mismo que Python 1.6, con unos pocos arreglos de bugs, y con una nueva licencia compatible con GPL.

Gráfico 2. 13: Python

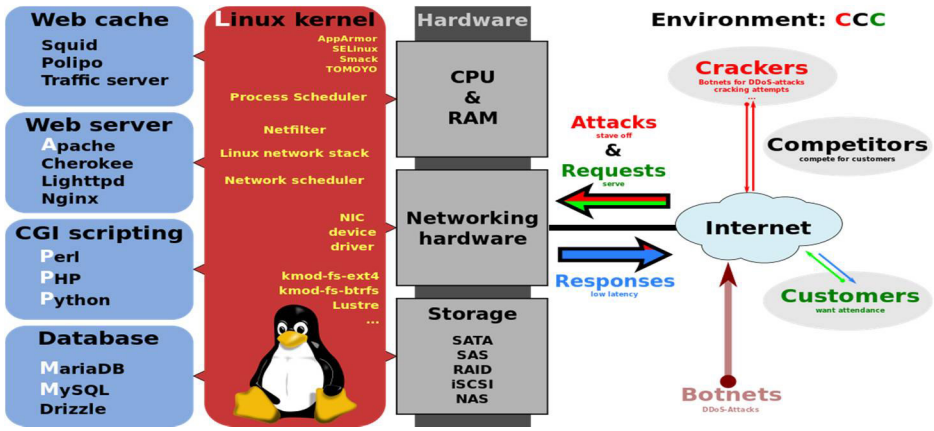
```
def add5(x):
   return x+5

def dotwrite(ast):
   nodename = getNodename()
   label=symbol.sym_name.get(int(ast[0]),ast[0])
   print '    %s [label="%s' % (nodename, label),
   if isinstance(ast[1], str):
      if ast[1].strip():
         print '= %s"];' % ast[1]
      else:
         print '"]'
   else:
      print '"];'
      children = []
      for  in n, childenumerate(ast[1:]):
         children.append(dotwrite(child))
      print ,'    %s -> {' % nodename
      for  in :namechildren
         print '%s' % name,
```

Fuente: Python

Otro objetivo del diseño del lenguaje es la facilidad de extensión. Se pueden escribir nuevos módulos fácilmente en C o C++. Python puede incluirse en aplicaciones que necesitan una interfaz programable. Los usuarios de Python se refieren a menudo a la Filosofía Python que es bastante análoga a la filosofía de Unix. El código que sigue los principios de Python de legibilidad y transparencia se dice que es "pythonico". Contrariamente, el código opaco u ofuscado es bautizado como "no pythonico" ("unpythonic" en inglés). Estos principios fueron famosamente descritos por el desarrollador de Python Tim Peters en El Zen de Python:

- Bello es mejor que feo.
- Explícito es mejor que implícito.
- Simple es mejor que complejo.
- Complejo es mejor que complicado.
- Plano es mejor que anidado.
- Disperso es mejor que denso.
- La legibilidad cuenta.
- Los casos especiales no son tan especiales como para quebrantar las reglas.
- Aunque lo práctico gana a la pureza.
- Los errores nunca deberían dejarse pasar silenciosamente.
- A menos que hayan sido silenciados explícitamente.
- Frente a la ambigüedad, rechaza la tentación de adivinar.
- Debería haber una -y preferiblemente sólo una- manera obvia de hacerlo.
- Aunque esa manera puede no ser obvia al principio a menos que usted sea holandés.
- Ahora es mejor que nunca.
- Aunque nunca es a menudo mejor que ya mismo.
- Si la implementación es difícil de explicar, es una mala idea.
- Si la implementación es fácil de explicar, puede que sea una buena idea.
- Los espacios de nombres (*namespaces*) son una gran idea ¡Hagamos más de esas cosas!

Gráfico 2. 14: LAMP

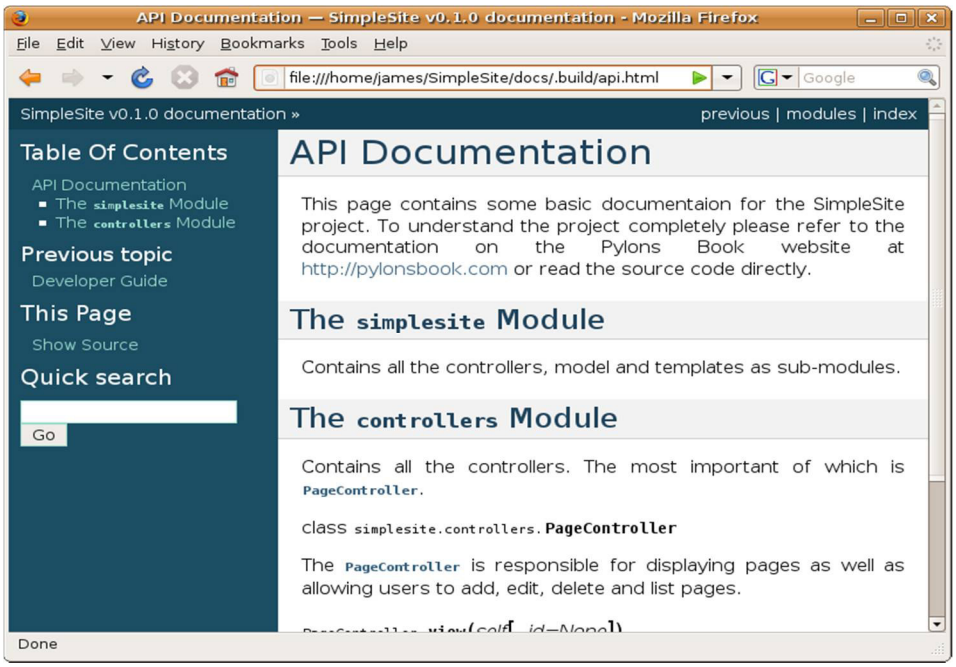


Fuente: LAMP comprende Python (aquí con Squid)

2.13. Sphinx (generador de documentación)

Sphinx es un software generador de documentación que convierte ficheros reStructuredText en sitios web HTML y otros formatos, incluyendo PDF, EPub y man. Saca provecho de la naturaleza extensible de reStructuredText y sus extensiones (ej. para generar automáticamente documentación desde código fuente, escribir notación matemática o resalzar código). El primer lanzamiento público, la versión 0.1.61611, se hizo el 21 de marzo de 2008. Se desarrolló y usó extensivamente por y para el Proyecto de documentación Python.

Gráfico 2. 15: Sphinx



Fuente: Autor

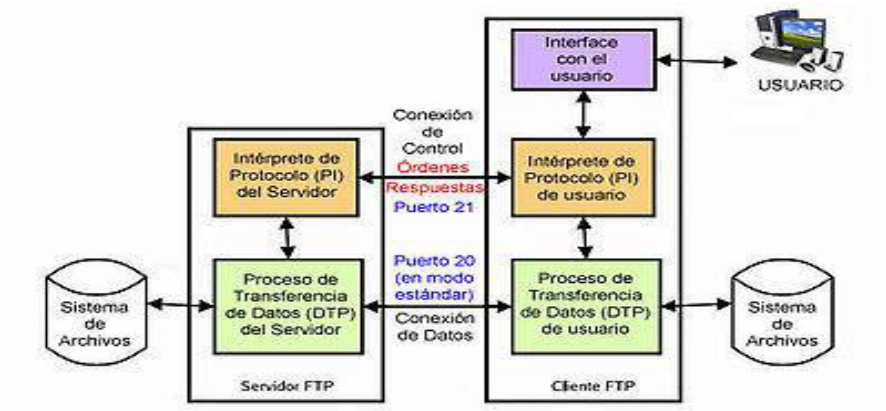
Desde su introducción en 2008, Sphinx ha sido adoptado por muchos otros proyectos Python importantes, como Bazaar, SQLAlchemy, MayaVi, Sage, SciPy, Django y Pylons; también se usa para documentar la API Python de Blender. El proyecto Read the Docs, que automatiza el proceso de construir y subir documentación Sphinx después de cada commit, se creó para hacer más sencillo el mantenimiento de la documentación. Sphinx está patrocinado por la Python Software Foundation.

2.14. FTP (File Transfer Protocol)

FTP (siglas en inglés de File Transfer Protocol, "Protocolo de Transferencia de Archivos") en informática, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la arquitectura cliente-servidor. Desde una computadora cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos, independientemente del sistema operativo utilizado en cada computadora.

El servicio FTP es ofrecido por la capa de aplicación del modelo de capas de red TCP/IP al usuario, utilizando normalmente el puerto de red 20 y el 21. Un problema básico de FTP es que está pensado para ofrecer la máxima velocidad en la conexión, pero no la máxima seguridad, ya que todo el intercambio de información, desde el login y password del usuario en el servidor hasta la transferencia de cualquier archivo, se realiza en texto plano sin ningún tipo de cifrado, con lo que un posible atacante puede capturar este tráfico, acceder al servidor y/o apropiarse de los archivos transferidos. Para solucionar este problema son de gran utilidad aplicaciones como scp y sftp, incluidas en el paquete SSH, que permiten transferir archivos pero cifrando todo el tráfico. El protocolo FTP se empezó a utilizar en abril de 1971, publicado como el RFC 114, antes de que existiera la pila TCP/IP. La estructura general fue establecida en 1973. Fue modificado varias veces, añadiendo nuevos comandos y funcionalidades. Al final se publicó el RFC 959 en octubre de 1985, que es la que se utiliza actualmente.

Gráfico 2. 16: El Modelo FTP



Fuente: Fundamentos de FTP

El siguiente modelo representa el diagrama de un servicio FTP. En el modelo, el intérprete de protocolo (IP) de usuario inicia la conexión de control en el puerto 21. Las órdenes FTP estándar las genera el IP de usuario y se transmiten al proceso servidor a través de la conexión de control. Las respuestas estándar se envían desde la IP del servidor la IP de usuario por la conexión de control como respuesta a las órdenes. Estas órdenes FTP especifican parámetros para la conexión de datos (puerto de datos, modo de transferencia, tipo de representación y estructura) y la naturaleza de la operación sobre el sistema de archivos (almacenar, recuperar, añadir, borrar, etc.). El proceso de transferencia de datos (DTP) de usuario u otro proceso en su lugar, debe esperar a que el servidor inicie la conexión al puerto de datos especificado (puerto 20 en modo activo o estándar) y transferir los datos en función de los parámetros que se hayan especificado. Con respecto a la comunicación entre cliente y servidor, es independiente del sistema de archivos utilizado en cada computadora, de manera que no importa que sus sistemas operativos sean distintos, porque las entidades que se comunican entre sí son los PI y los DTP, que usan el mismo protocolo estandarizado: el FTP.

También hay que destacar que la conexión de datos es bidireccional, es decir, se puede usar simultáneamente para enviar y para recibir, y no tiene por qué existir todo el tiempo que dura la conexión FTP. Pero tenía en sus comienzos un problema, y era la localización de los servidores en la red. Es decir, el usuario que quería descargar algún archivo mediante FTP debía conocer en qué máquina estaba ubicado. La única herramienta de búsqueda de información que existía era Gopher, con todas sus limitaciones. Un servidor FTP es un programa especial que se ejecuta en una computadora servidor normalmente conectado a Internet (aunque puede estar conectado a otros tipos de redes, LAN, MAN, etc.).

Su función es permitir el intercambio de datos entre diferentes servidores/computadoras. Por lo general, los programas servidores FTP no suelen encontrarse en las computadoras personales, por lo que un usuario normalmente utilizará el FTP para conectarse remotamente a uno y así intercambiar información con él.

Las aplicaciones más comunes de los servidores FTP suelen ser el alojamiento web, en el que sus clientes utilizan el servicio para subir sus páginas web y sus archivos correspondientes; o como servidor de backup (copia de seguridad) de los archivos importantes que pueda tener una empresa. Para ello, existen protocolos de comunicación FTP para que los datos se transmitan cifrados, como el SFTP (Secure File Transfer Protocol). Cuando un navegador no está equipado con la función FTP, o si se quiere cargar archivos en una computadora remota, se necesitará utilizar un programa cliente FTP. Un cliente FTP es un programa que se instala en la computadora del usuario, y que emplea el protocolo FTP para conectarse a un servidor FTP y transferir archivos, ya sea para descargarlos o para subirlos. Para utilizar un cliente FTP, se necesita conocer el nombre del archivo, la computadora en que reside (servidor, en el caso de descarga de archivos), la computadora al que se quiere transferir el archivo (en caso de querer subirlo nosotros al servidor), y la carpeta en la que se encuentra. Algunos clientes de FTP básicos en modo consola vienen integrados en los sistemas operativos, incluyendo Microsoft Windows, DOS, GNU/Linux y Unix. Sin embargo, hay disponibles clientes con opciones añadidas e interfaz gráfica. Aunque muchos navegadores tienen ya integrado FTP, es más confiable a la hora de conectarse con servidores FTP no anónimos utilizar un programa cliente.

Los servidores FTP anónimos ofrecen sus servicios libremente a todos los usuarios, permiten acceder a sus archivos sin necesidad de tener un “USER ID” o una cuenta de usuario. Es la manera más cómoda fuera del servicio web de permitir que todo el mundo tenga acceso a cierta información sin que para ello el administrador de un sistema tenga que crear una cuenta para cada usuario. Si un servidor posee servicio “FTP anonymous” solamente con teclear la palabra “anonymous”, cuando pregunte por tu usuario tendrás acceso a ese sistema. No se necesita ninguna contraseña preestablecida, aunque tendrás que introducir una sólo para ese momento, normalmente se suele utilizar la dirección de correo electrónico propia. Solamente con eso se consigue acceso a los archivos del FTP, aunque con menos privilegios que un usuario normal. Normalmente solo podrás leer y copiar los archivos que sean públicos, así indicados por el administrador del servidor al que nos queramos conectar.

Normalmente, se utiliza un servidor FTP anónimo para depositar grandes archivos que no tienen utilidad si no son transferidos a la máquina del usuario, como por ejemplo programas, y se reservan los servidores de páginas web (HTTP) para almacenar información textual destinada a la lectura en línea. Si se desea tener privilegios de acceso a cualquier parte del sistema de archivos del servidor FTP, de modificación de archivos existentes, y de posibilidad de subir nuestros propios archivos, generalmente se suele realizar mediante una cuenta de usuario. En el servidor se guarda la información de las distintas cuentas de usuario que pueden acceder a él, de manera que para iniciar una sesión FTP debemos introducir una autentificación (en inglés: login) y una contraseña (en inglés: password) que nos identifica unívocamente. Un "cliente FTP basado en Web" no es más que un cliente FTP al cual podemos acceder a través de nuestro navegador web sin necesidad de tener otra aplicación para ello. El usuario accede a un servidor web (HTTP) que lista los contenidos de un servidor FTP. El usuario se conecta mediante HTTP a un servidor web, y el servidor web se conecta mediante FTP al servidor FTP.

El servidor web actúa de intermediario haciendo pasar la información desde el servidor FTP en los puertos 20 y 21 hacia el puerto 80 HTTP que ve el usuario. Siempre hay momentos en que nos encontramos fuera de casa, no llevamos la computadora portátil encima y necesitamos realizar alguna tarea urgente desde una computadora de acceso público, de un amigo, del trabajo, la universidad, etc. Otras veces estamos detrás de un proxy o firewall que no nos permite acceder a servidores FTP externos. Al disponer de un cliente FTP basado en Web podemos acceder al servidor FTP remoto como si estuviéramos realizando cualquier otro tipo de navegación web. A través de un cliente FTP basado en Web podrás, crear, copiar, renombrar y eliminar archivos y directorios. Cambiar permisos, editar, ver, subir y descargar archivos, así como cualquier otra función del protocolo FTP que el servidor FTP remoto permita. El acceso sin restricciones al servidor que proporcionan las cuentas de usuario implica problemas de seguridad, lo que ha dado lugar a un tercer tipo de acceso FTP denominado invitado (guest), que se puede contemplar como una mezcla de los dos anteriores.

La idea de este mecanismo es la siguiente: se trata de permitir que cada usuario conecte a la máquina mediante su login y su password, pero evitando que tenga acceso a partes del sistema de archivos que no necesita para realizar su trabajo, de esta forma accederá a un entorno restringido, algo muy similar a lo que sucede en los accesos anónimos, pero con más privilegios. FTP admite dos modos de conexión del cliente. Estos modos se denominan *activo* (o Estándar, o PORT, debido a que el cliente envía comandos tipo PORT al servidor por el canal de control al establecer la conexión) y *pasivo* (o PASV, porque en este caso envía comandos tipo PASV). Cuando el cliente envía un comando PASV sobre el canal de control, el servidor FTP le indica por el canal de control, el puerto (mayor a 1023 del servidor. Ejemplo: 2040) al que debe conectarse el cliente. El cliente inicia una conexión desde el puerto siguiente al puerto de control (Ejemplo: 1036) hacia el puerto del servidor especificado anteriormente (Ejemplo: 2040).

Antes de cada nueva transferencia tanto en el modo Activo como en el Pasivo, el cliente debe enviar otra vez un comando de control (PORT o PASV, según el modo en el que haya conectado), y el servidor recibirá esa conexión de datos en un nuevo puerto aleatorio (si está en modo pasivo) o por el puerto 20 (si está en modo activo). En el protocolo FTP existen 2 tipos de transferencia en ASCII y en binarios. Es importante conocer cómo debemos transportar un archivo a lo largo de la red. Si no utilizamos las opciones adecuadas podemos destruir la información del archivo. Por eso, al ejecutar la aplicación FTP, debemos acordarnos de utilizar uno de estos comandos (o poner la correspondiente opción en un programa con interfaz gráfica):

- Tipo ASCII

Adecuado para transferir archivos que sólo contengan caracteres imprimibles (archivos ASCII, no archivos resultantes de un procesador de texto), por ejemplo páginas HTML, pero no las imágenes que puedan contener.

- Tipo Binario

Este tipo es usado cuando se trata de archivos comprimidos, ejecutables para PC, imágenes y archivos de audio.

2.15. Nmap

Es un programa de código abierto que sirve para efectuar rastreo de puertos escrito originalmente por Gordon Lyon (más conocido por su alias Fyodor Vaskovich). Se usa para evaluar la seguridad de sistemas informáticos, así como para descubrir servicios o servidores en una red informática. Nmap apareció en septiembre de 1997, en un artículo de la revista Phrack Magazine. El código fuente venía incluido. Otros desarrollos incluyeron mejores algoritmos para determinar qué servicios estaban funcionando, reescritura de código de C a C++, se agregaron tipos de scan adicionales y nuevos protocolos como IPv6. Nmap 3.5 apareció en febrero de 2004, y la versión 4.0 en enero de 2006, con cientos de mejoras. Los cambios de cada versión se pueden encontrar en el listado de cambios de Nmap.

2.15.1. Características de Nmap

- Descubrimiento de servidores: Identifica computadoras en una red, por ejemplo listando aquellas que responden ping.
- Identifica puertos abiertos en una computadora objetivo y qué servicios está ejecutando la misma.
- Determinar qué sistema operativo y versión utiliza dicha computadora, (esta técnica es también conocida como *fingerprinting*).
- Obtiene algunas características del hardware de red de la máquina objeto de la prueba.

2.15.2. Aplicaciones típicas de Nmap

Ha llegado a ser uno de las herramientas imprescindibles para todo administrador de sistema, y es usado para pruebas de penetración y tareas de seguridad informática en general. Como muchas herramientas usadas en el campo de la seguridad informática, es también una herramienta muy utilizada para hacking. Los administradores de sistema pueden utilizarlo para verificar la presencia de posibles aplicaciones no autorizadas ejecutándose en el servidor, así como los crackers pueden usarlo para descubrir objetivos potenciales. Nmap permite hacer el inventario y el mantenimiento del inventario de computadoras de una red.

Se puede usar entonces para auditar la seguridad de una red, mediante la identificación de todo nuevo servidor que se conecte. Nmap es a menudo confundido con herramientas para verificación de vulnerabilidades como Nessus. Nmap es difícilmente detectable, ha sido creado para evadir los Sistema de detección de intrusos (IDS) e interfiere lo menos posible con las operaciones normales de las redes y de las computadoras que son analizadas.

2.15.3. Entornos de trabajo

Nmap puede funcionar en sistemas operativos basados en Unix (GNU/Linux, Solaris, BSD y Mac OS X), y también en otros Sistemas Operativos como Microsoft Windows y Amiga OS.

2.15.4. Interfaces gráficas

La interfaz usuario oficial es *nmapfe*, escrita originalmente por Zach Smith, y Nmap lo integra desde la versión 2.2.

Existen otras interfaces basadas en navegadores Web. Algunos ejemplos son

LOCALSCAN, **nmap-web**, y **Nmap-CGI**.

NMapW es una interfaz sobre Microsoft Windows escrita por Syhunt.

NMapWin es otra interfaz para Windows. Sin embargo, no ha sido actualizada desde la versión 1.4.0 lanzada en junio de 2003.

Una plataforma completa Nmap con capacidades para funcionar sobre distintos OS se encuentra en UMIT. Su autor es Adriano Monteiro Marques.

Zenmap es la interfaz oficial para sistemas operativos GNU Linux.

2.15.5. Controversia

De manera análoga a la mayoría de herramientas utilizadas en seguridad informática, Nmap puede usarse para bien o para mal. Puede usarse solo o para preparar otro ataque, con otra herramienta de intrusión.

Pero los mismos administradores de sistemas lo utilizan para buscar fallas en sus propias redes, o bien para detectar computadoras que no cumplen con los requisitos mínimos de seguridad de la organización. (Nótese que Nmap por sí solo sólo dará una indicación básica de la vulnerabilidad de una computadora, y que normalmente es usado en conjunto con otras herramientas y tests)

Nmap es a menudo confundido con herramientas de investigación de vulnerabilidad como Nessus, las cuales van más lejos en su exploración de sus objetivos.

2.15.6. Cultura Popular

Nmap ha sido también usado en el film *The Matrix reloaded* por el personaje Trinity para penetrar en el sistema de la central eléctrica, mediante la explotación de vulnerabilidades en el servidor SSH y en el Control de redundancia cíclica, (descubiertas en el 2001). La interfaz gráfica de Nmap en la película suscitó el interés de las discusiones en Internet., y fue comentado como una aparición bastante realista de las herramientas de hacking. En esas discusiones, algunos piensan que el personaje Trinity utilizó el ataque Control de redundancia cíclica (descubierto en 2001) para obtener el acceso, luego de que Nmap revelara la existencia de un servicio SSH. Nmap y NmapFE fueron también usados en The Listening, una película de 2006 sobre un ex funcionario de la NSA estadounidense, que deserta y organiza una estación de contraespionaje en los alpes italianos. Partes del código fuente de Nmap pueden verse en la película *Battle Royale*. Imágenes extraídas de películas y otras alusiones a Nmap pueden verse en la página "Nmap in the News" del sitio web oficial de Nmap.

2.16. Ataques informáticos

Un ataque informático es un método por el cual un individuo, mediante un sistema informático, intenta tomar el control para desestabilizar o dañar otro sistema informático (computadora, servidor, red privada, etc.) ya sea presencial o remotamente.

2.16.1. Tipos de ataques informáticos

Scanning (Búsqueda): El escaneo, como método de descubrir canales de comunicación susceptibles de ser explotados, lleva en uso mucho tiempo. La idea es recorrer (escanear) tantos puertos de escucha como sea posible, y guardar información de aquellos que sean receptivos o de utilidad para cada necesidad en particular. Muchas utilidades de auditoría también se basan en este paradigma. El escaneo de puertos pertenece a la seguridad informática desde que era utilizado en los sistemas de telefonía. Dado que actualmente existen millones de números de teléfono a los que se pueden acceder con una simple llamada, la solución lógica para encontrar números que puedan interesar es intentar conectarlos a todos. La idea básica es llamar a un número y si el módem devuelve un mensaje de conectado, grabar el número. En otro caso, la computadora cuelga el teléfono y llama al siguiente número. Escanear puertos implica las mismas técnicas de fuerza bruta. Se envía una serie de paquetes para varios protocolos y se deduce que servicios están "escuchando" por las respuestas recibidas o no recibidas. Existen diversos tipos de Scanning según las técnicas, puertos y protocolos explotados:

TCP Connect Scanning: Esta es la forma básica del escaneo de puertos TCP. Si el puerto está escuchando, devolverá una respuesta de éxito; cualquier otro caso significará que el puerto no está abierto o que no se puede establecer conexión con a él. Las ventajas que caracterizan esta técnica es que no necesita de privilegios especiales y su gran velocidad. Su principal desventaja es que este método es fácilmente detectable por el Administrador del sistema. Se verá un gran número de conexiones y mensajes de error para los servicios en los que se ha conseguido conectar la máquina que lanza el scanner e inmediatamente se ha desconectado.

TCP SYN Scanning: Cuando dos procesos establecen una comunicación usan el modelo Cliente/Servidor para establecer la conexión. La aplicación del Servidor "escucha" todo lo que ingresa por los puertos. La identificación del Servidor se efectúa a través de la dirección IP del sistema en el que se ejecuta y del número de puerto del que depende para la conexión. El Cliente establece la conexión con el Servidor a través del puerto disponible para luego intercambiar datos.

La información de control llamada HandShake (es como un saludo) se intercambia entre el Cliente y el Servidor para establecer un dialogo antes de transmitir datos. Los "paquetes" o segmentos TCP tienen banderas que indican el estado del mismo. El protocolo TCP de Internet, sobre el que se basa la mayoría de los servicios (incluyendo el correo electrónico, el web y el IRC) implica esta conexión entre dos máquinas. El establecimiento de dicha conexión se realiza mediante lo que se llama Three-Way Handshake ("conexión en tres pasos") ya que intercambian tres segmentos. En forma esquemática se tiene: El programa Cliente (C) pide conexión al Servidor (S) enviándole un segmento SYN (Synchronize Sequence Number). Este segmento le dice a S que C desea establecer una conexión. S si está abierto y escuchando, al recibir este segmento SYN, activa su indicador SYN y envía una autentificación ACK (Acknowlegde) de manera de acuse de recibo a C. Si S está cerrado envía un indicador RST. C entonces ACKea (autentifica) a S. Ahora ya puede tener lugar la transferencia de datos. Cuando las aplicaciones conectadas terminan la transferencia, realizaran otra negociación a tres bandas con segmentos FIN en vez SYN. La técnica TCP SYN Scanning, se implementa un escaneo de "media-apertura", dado que nunca se abre una sesión TCP completa. Se envía un paquete SYN como si se fuera a usar una conexión real y se espera por la respuesta. Al recibir un SYN/ACK se envía, inmediatamente, un RST para terminar la conexión y se registra este puerto como abierto. La principal ventaja de esta técnica de escaneo es que pocos sitios están preparados para registrarlos. La desventaja es que en algunos sistemas Unix, se necesitan privilegios de Administrador para construir estos paquetes SYN.

TCP FIN Scanning- Stealth Port Scanning: Hay veces en que incluso el escaneo SYN no es lo suficientemente "clandestino" o limpio. Algunos sistemas (Firewalls y filtros de paquetes) monitorizan la red en busca de paquetes SYN a puertos restringidos. Para subsanar este inconveniente los paquetes FIN, en cambio, podrían ser capaces de pasar sin ser advertidos. Este tipo de Escaneo está basado en la idea de que los puertos cerrados tienden a responder a los paquetes FIN con el RST correspondiente. Los puertos abiertos, en cambio, suelen ignorar el paquete en cuestión.

Este es un comportamiento correcto del protocolo TCP, aunque algunos sistemas no cumplen con este requerimiento, enviando paquetes RST siempre, independientemente de si el puerto está abierto o cerrado. Como resultado, no son vulnerables a este tipo de escaneo. Sin embargo, es posible realizarlo en otros sistemas Unix.

Fragmentation Scanning: Esta no es una nueva técnica de escaneo como tal, sino una modificación de las anteriores. En lugar de enviar paquetes completos de sondeo, los mismos se particionan en un par de pequeños fragmentos IP. Así, se logra partir una cabecera IP en distintos paquetes para hacerlo más difícil de monitorizar por los filtros que pudieran estar ejecutándose en la máquina objetivo.
Sin embargo, algunas implementaciones de estas técnicas tienen problemas con la gestión de este tipo de paquetes tan pequeños, causando una caída de rendimiento en el sistema del intruso o en el de la víctima. Problemas de esta índole convierte en detectables a este tipo de ataque.

Eavesdropping-Packet Sniffing: Muchas redes son vulnerables al Eavesdropping, o a la pasiva intercepción sin modificación del tráfico de red. Esto se realiza con Packet Sniffers, los cuales son programas que monitorean los paquetes que circulan por la red. Los Sniffers pueden ser colocados tanto en una estación de trabajo conectada a la 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. En la cabecera de los paquetes enviados a través de una red, entre otros datos, se tiene, la dirección del emisor y la del destinatario. De esta forma, independientemente de protocolo usado, las tramas llegan a su destino. Cada máquina conectada a la red mediante una placa con una dirección única verifica la dirección destino del paquete. Si estas direcciones son iguales asume que el paquete enviado es para ella, caso contrario libera el paquete para que otras placas lo analicen. Un Sniffers consiste en colocar a la placa de red en un modo llamado promiscuo, el cual desactiva el filtro de verificación de direcciones y por lo tanto todos los paquetes enviados a la red llegan a esta computadora donde está instalado el Sniffer. Inicialmente este tipo de software, era únicamente utilizado por los Administradores de redes locales, aunque con el tiempo llegó a convertirse en una herramienta muy usada por los intrusos.

Actualmente existen Sniffers para capturar cualquier tipo de información específica. Por ejemplo passwords de un recurso compartido o de acceso a una cuenta, que generalmente viajan sin encriptar al ingresar a sistemas de acceso remoto. También son utilizados para capturar números de tarjetas de crédito y direcciones de e-mails entrantes y salientes. El análisis de tráfico puede ser utilizado también para determinar relaciones entre organizaciones e individuos. Para realizar estas funciones se analizan las tramas de un segmento de red, y presentan al usuario sólo las que interesan. Normalmente, los buenos Sniffers, no se pueden detectar, aunque la inmensa mayoría, y debido a que están demasiado relacionados con el protocolo TCP/IP, si pueden ser detectados con algunos trucos.

Snooping-Downloading: Los ataques de esta categoría tienen el mismo objetivo que el Sniffing: obtener la información sin modificarla, sin embargo los métodos son diferentes. Aquí, además de interceptar el tráfico de red, el atacante ingresa a los documentos, mensajes de correo electrónico y otra información guardada, realizando en la mayoría de los casos un downloading (copia de documentos) de esa información a su propia computadora, para luego hacer un análisis exhaustivo de la misma. El Snooping puede ser realizado por simple curiosidad, pero también es realizado con fines de espionaje y robo de información o software. Los casos más resonantes de este tipo de ataques fueron: el robo de un archivo con más 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.

2.16.2. Ataques de autentificación

Este tipo de ataque tiene como objetivo engañar al sistema de la víctima para ingresar al mismo. Generalmente este engaño se realiza tomando las sesiones ya establecidas por la víctima u obteniendo su nombre de usuario y password.

Spoofing-Looping*:* Spoofing puede traducirse como "hacerse pasar por otro" y el objetivo de esta técnica, justamente, es actuar en nombre de otros usuarios, usualmente para realizar tareas de Snooping 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. El intruso usualmente utiliza un sistema para obtener información e ingresar en otro, y luego utiliza este para entrar en otro, y así sucesivamente. Este proceso, llamado Looping, y 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 Inssider, o por un estudiante a miles de Km de distancia, pero que ha tomado la identidad de otros. La investigación de procedencia de un Looping es casi imposible, ya que el investigador debe contar con la colaboración de cada Administrador de cada red utilizada en la ruta. El envío de falsos e-mails es otra forma de Spoofing que las redes permiten. Aquí el atacante envía E-Mails a nombre de otra persona con cualquier motivo y objetivo. Muchos ataques de este tipo comienzan con Ingeniería Social y los usuarios, por falta de cultura, facilitan a extraños sus identificaciones dentro del sistema usualmente través de una simple llamada telefónica.

Spoofing: Este tipo de ataques sobre protocolos suele implicar un buen conocimiento del protocolo en el que se va a basar el ataque. Los ataques tipo Spoofing bastante conocidos son el IP Spoofing, el DNS Spoofing y el Web Spoofing.

IP 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. Su utilización más común es enviar los paquetes con la dirección de un tercero, de forma que la víctima "ve" un ataque proveniente de esa tercera red, y no la dirección real del intruso.

DNS Spoofing*:* Este ataque se consigue mediante la manipulación de paquetes UDP pudiéndose comprometer el servidor de nombres de dominios (Domain Name Server-DNS) de Windows NT(c). Si se permite el método de recursión en la resolución de "Nombre" "Dirección IP" en el DNS.

La recursión consiste en la capacidad de un servidor de nombres para resolver una petición de dirección IP a partir de un nombre que no figura en su base de datos. Este es el método típico y por defecto de funcionamiento.

Web Spoofing: En el caso Web Spoofing el atacante crea un sitio web completo falso y similar al que la víctima desea entrar. Los accesos a este sitio están dirigidos por el atacante, permitiéndole monitorizar todas las acciones de la víctima, desde sus datos hasta las passwords, números de tarjeta de créditos, etc. El atacante también es libre de modificar cualquier dato que se esté transmitiendo entre el servidor original y la víctima o viceversa.

IP Splicing-Hijacking: Se produce cuando un atacante consigue interceptar una sesión ya establecida. El atacante espera a que la víctima se identifique ante el sistema y tras ello le suplanta como usuario autorizado.

Utilización de BackDoors: Las puertas traseras son trozos de código en un programa que permiten a quien las conoce saltarse los métodos usuales de autentificación para realizar ciertas tareas. Habitualmente son insertados por los programadores del sistema para agilizar la tarea de probar código durante la fase de desarrollo. Esta situación se convierte en una falla de seguridad si se mantiene, involuntaria o intencionalmente, una vez terminado el producto ya que cualquiera que conozca el agujero o lo encuentre en su código podrá saltarse los mecanismos de control normales.

Utilización de Exploits: Es muy frecuente ingresar a un sistema explotando agujeros en los algoritmos de encriptación utilizados, en la administración de las claves por parte la empresa, o simplemente encontrado un error en los programas utilizados. Los programas para explotar estos "agujeros" reciben el nombre de Exploits y lo que realizan es aprovechar la debilidad, fallo o error hallado en el sistema (hardware o software) para ingresar al mismo. Nuevos Exploits explotando nuevos errores en los sistemas se publican cada día por lo que mantenerse informado de los mismos y de las herramientas para combatirlos es de vital importancia.

Obtención de Passwords: Este método comprende la obtención por "Fuerza Bruta" de aquellas claves que permiten ingresar a los sistemas, aplicaciones, cuentas, etc. atacados. Muchas passwords de acceso son obtenidas fácilmente porque involucran el nombre u otro dato familiar del usuario y, además, esta nunca o rara vez se cambia. En este 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.

Uso de Diccionarios: Los Diccionarios son archivos con millones de palabras, las cuales pueden ser passwords utilizadas por los usuarios. Este archivo es utilizado para descubrir dicha password en pruebas de fuerza bruta. El programa encargado de probar cada una de las palabras encripta cada una de ellas mediante el algoritmo utilizado por el sistema atacado y compara la palabra encriptada contra el archivo de passwords del sistema atacado (previamente obtenido). Si coinciden se ha encontrado la clave de acceso al sistema mediante el usuario correspondiente a la clave hallada. Actualmente es posible encontrar diccionarios de gran tamaño orientados y acuerdo al tipo de organización que se esté atacando.

Denial of service (DOS): Los protocolos existentes actualmente fueron diseñados para ser empleados en una comunidad abierta y con una relación de confianza mutua. La realidad indica que es más fácil desorganizar el funcionamiento de un sistema que acceder al mismo; así los ataques de Negación de Servicio tienen como objetivo saturar los recursos de la víctima de forma tal que se inhabilita los servicios brindados por la misma.

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 pueda utilizarla. 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 (usando Spoofing y Looping). 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 ISPs (proveedores de Internet) han sufrido bajas temporales del servicio por ataques que explotan el protocolo TCP. Muchos Hosts 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 bloqueo instantáneo del equipo. Esta vulnerabilidad ha sido ampliamente utilizada en el pasado pero, aún hoy pueden encontrarse sistemas vulnerables.

Syn Flood: Como ya se explicó en el TCP SYN Scanning el protocolo TCP se basa en una conexión en tres pasos. Si el paso final no llega a establecerse, la conexión permanece en un estado denominado "semiabierto". El Syn Flood es el más famoso de los ataques del tipo Denial of Service, publicado por primera vez en la revista Phrack. Se basa en un "saludo" incompleto entre los dos hosts. El Cliente envía un paquete SYN pero no responde al paquete ACK ocasionando que la pila TCP/IP espere cierta cantidad de tiempo a que el host hostil responda antes de cerrar la conexión. Si se crean muchas peticiones incompletas de conexión, no se responde a ninguna, el Servidor estará inactivo mucho tiempo esperando respuesta. Esto ocasiona la lentitud en los demás servicios. El problema es que muchos sistemas operativos tienen un límite muy bajo en el número de conexiones "semiabiertas" que pueden manejar en un momento determinado. Si se supera ese límite, el servidor sencillamente dejará de responder a las nuevas peticiones de conexión que le vayan llegando. Las conexiones "semiabiertas" van caducando tras un tiempo, liberando "huecos" para nuevas conexiones, pero mientras el atacante mantenga el Syn Flood, la probabilidad de que una conexión recién liberada sea capturada por un nuevo SYN malicioso es muy alta. La potencia de este ataque reside en que muchos sistemas operativos fijan un límite del orden de 5 a 30 conexiones "semiabiertas", y que éstas caducan al cabo de un par de minutos. Para mantener el servidor fuera de servicio, un atacante sólo necesita enviar un paquete SYN cada 4 segundos, algo al alcance de, incluso, un módem de 300 baudios. Este ataque suele combinarse también con el IP Spoofing, de forma de ocultar el origen del ataque.

Connection Flood: La mayoría de las empresas que brindan servicios de Internet (ISP) tienen un límite máximo en el número de conexiones simultáneas. Una vez que se alcanza ese límite, no se admitirán conexiones nuevas.

Así, por ejemplo, un servidor Web puede tener, por ejemplo, capacidad para atender a mil usuarios simultáneos. Si un atacante establece mil conexiones y no realiza ninguna petición sobre ellas, monopolizará la capacidad del servidor.

Net Flood: En estos casos, la red víctima no puede hacer nada. Aunque filtre el tráfico en sus sistemas, sus líneas estarán saturadas con tráfico malicioso, incapacitándolas para cursar tráfico útil. Un ejemplo habitual es el de un teléfono: si alguien quiere molestar, sólo tiene que llamar, de forma continua. Si se descuelga el teléfono hará que deje de molestar, tampoco se puede recibir llamadas de otras personas. En el caso de Net Flooding ocurre algo similar. El atacante envía tantos paquetes de solicitud de conexión que las conexiones auténticas simplemente no pueden competir. En casos así el primer paso a realizar es el ponerse en contacto con el Proveedor del servicio para que intente determinar la fuente del ataque y, como medida provisional, filtre el ataque en su extremo de la línea. El siguiente paso consiste en localizar las fuentes del ataque e informar a sus Administradores, ya que seguramente se estarán usando sus recursos sin su conocimiento y consentimiento.

Land Attack: Este ataque consiste en un Bug (error) en la implementación de la pila TCP/IP de las plataformas Windows. El ataque consiste en mandar a algún puerto abierto de un servidor generalmente al 113 o al 139 un paquete, maliciosamente construido, con la dirección y puerto origen igual que la dirección y puerto destino. Por ejemplo se envían un mensaje desde la dirección 10.0.0.1:139 hacia ella misma. El resultado obtenido es que luego de cierta cantidad de mensajes enviados-recibidos la máquina termina colgándose. Existen ciertas variantes a este método consistente, por ejemplo, en enviar el mensaje a una dirección específica sin especificar el puerto Smurf o Broadcast Storm. Este ataque es bastante simple y a su vez devastador. Consiste en recolectar una serie de direcciones para a continuación mandar una petición ICMP simulando un Ping a cada una de ellas en serie, varias veces, falsificando la dirección IP de origen. Este paquete maliciosamente manipulado, será repetido en la difusión, y cientos o miles de hosts según la lista de direcciones de difusión disponible, mandarán una respuesta a la víctima cuya dirección IP figura en el paquete ICMP.

Supernuke o Winnuke: Un ataque característico y quizás el más común de las computadoras con Windows es el Nuke, que hace que las computadoras que escuchan por el puerto UDP 137 a 139 utilizados por los protocolos Netbios de Wins, queden fuera de servicio o disminuyan su rendimientos al enviarle paquetes UDP manipulados. Generalmente se envían fragmentos de paquetes, que la máquina víctima detecta como inválidos pasando a un estado inestable.

Teardrop I y II-Newtear-Bonk-Boink: Al igual que el Supernuke, los ataques Teardrop I y Teardrop II 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. Los ataque tipo Teardrop son especialmente peligrosos ya que existen multitud de implementaciones, algunas de ellas forman paquetes, que explotan esta debilidad. Las más conocidas son aquellas con el nombre Newtear, Bonk y Boink.

E-Mail Bombing-Spamming: El E-Mail Bombing consiste en enviar muchas veces un mensaje idéntico a una misma dirección, saturando así mailbox del destinatario. El Spamming, en cambio se refiere a enviar el e-mail miles de usuarios, haya estos solicitados el mensaje o no. Es muy utilizado por las empresas para publicitar sus productos. El Spamming está siendo actualmente tratado por las leyes europeas como una violación de los derechos de privacidad del usuario.

2.16.3. Ataques de modificación-daño

Tampering o Data Diddling: Esta categoría se refiere a la modificación desautorizada de los datos o el software instalado en el sistema víctima, incluyendo borrado de archivos. 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. Aun así, si no hubo intenciones de "bajar" el sistema por parte del atacante; el Administrador posiblemente necesite darlo 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 u Outsiders, generalmente con el propósito de fraude o de dejar fuera de servicio a un competidor. Son innumerables los casos de este tipo: 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 una deuda impositiva. [4]

Borrado de Huellas: El borrado de huellas es una de las tareas más importantes que debe realizar el intruso después de ingresar en un sistema, ya que si se detecta su ingreso el Administrador buscará como conseguir "tapar el hueco" de seguridad, evitar ataques futuros e incluso rastrear al atacante. Las Huellas son todas las tareas que realizó el intruso en el sistema y por lo general son almacenadas en Logs por el sistema operativo. Los archivos Logs son una de las principales herramientas con las que cuenta un Administrador para conocer los detalles de las tareas realizadas en el sistema y la detección de intrusos.

Ataques Mediante Java Applets: Java es un lenguaje de programación interpretado desarrollado inicialmente por SUN. Su mayor popularidad la merece en su alto grado de seguridad. Los más usados navegadores actuales, implementan Máquinas Virtuales Java (MVJ) para ser capaces de ejecutar programas (Applets) de Java. Estos Applets, al fin y al cabo no son más que código ejecutable y como tal, susceptible de ser manipulado por intrusos. Pero, partiendo del diseño, Java siempre ha pensado en la seguridad del sistema. Las restricciones a las que somete a los Applets son de tal envergadura, imposibilidad de trabajar con ficheros a no ser que el usuario especifique lo contrario, imposibilidad de acceso a zonas de memoria y disco directamente, firma digital, etc., que es muy difícil lanzar ataques. Sin embargo, existe un grupo de expertos especializados en descubrir fallas de seguridad en las implementaciones de las MVJ (Máquina virtual de Java).

Ataques Mediante JavaScript y VBScript: JavaScript (de empresa Netscape) y VBScript (de Microsoft) son dos lenguajes usados por los diseñadores de sitios Web evitando el uso de Java.

---

[4] Dependiendo del tipo de sistema de detección de intrusos (IDS), el resultado del origen del ataque o acceso no autorizado puede llegar a saberse más pronto de lo esperado.

Aunque los fallos son mucho más numerosos en versiones antiguas de JavaScript, cuando apareció JavaScript, éste permitía el envío de mensajes de correo electrónico sin el reconocimiento del usuario, la lectura del historial de páginas visitadas, la lectura de directorios y de archivos. Estas fueron razón más que suficiente para que cientos de intrusos informáticos se aprovecharan de estas debilidades. El problema más importante apareció en Netscape 2.0 y fue bautizado como "Stuck On Load". Lo que sucedía es que se podía crear una ventana de 1*1 pixeles, por la cual los intrusos podían seguir extrayendo información sin que el usuario se enterase y aun cuando éste hubiese salido de la página, ya que esta ventana, un simple punto en la pantalla, era imperceptible para el usuario.

Ataques Mediante ActiveX: ActiveX es una de las tecnologías más potentes que ha desarrollado Microsoft. Mediante ActiveX es posible reutilizar código, descargar código totalmente funcional de un sitio remoto, etc. Esta tecnología es considerada la respuesta de Microsoft a Java. ActiveX soluciona los problemas de seguridad mediante certificados y firmas digitales. Una Autoridad Certificadora (AC) expende un certificado que acompaña a los controles activos y a una firma digital del programador. Si el usuario acepta el control, éste puede pasar a ejecutarse sin ningún tipo de restricciones. Es decir, la responsabilidad de la seguridad del sistema se deja en manos del usuario, ya sea este un experto cibernauta consciente de los riesgos que puede acarrear la acción o un perfecto novato en la materia. Esta última características es el mayor punto débil de los controles ActiveX ya que la mayoría de los usuarios aceptan el certificado sin siquiera leerlo, pudiendo ser esta la fuente de un ataque con un control dañino. La filosofía ActiveX es que las Autoridades de Certificación se fían de la palabra del programador del control. Es decir, el programador se compromete a firmar un documento que asegura que el control no es nocivo. Evidentemente siempre hay programadores con pocos escrúpulos o con ganas de experimentar. Así, un conocido grupo de hackers alemanes, desarrolló un control ActiveX maligno que modificaba el programa de Gestión Bancaria Personal Quicken95 de tal manera que si un usuario aceptaba el control, éste realizaba la tarea que supuestamente tenía que hacer y además modificaba el Quicken, para que la próxima vez que la víctima se conectara a su banco, se iniciara automáticamente una transferencia a una cuenta del grupo alemán.

Otro control ActiveX muy especialmente "malévolo" es aquel que manipula el código de ciertos exploradores, para que éste no solicite confirmación al usuario a la hora de descargar otro control activo de la Web. Es decir, deja totalmente descubierto a ataques con tecnología ActiveX el sistema de la víctima.

Ataques por Vulnerabilidades en los Navegadores: Generalmente los navegadores no fallan por fallos intrínsecos, sino que fallan las tecnologías que implementan, aunque en este punto analizaremos realmente fallos intrínsecos de los navegadores, como pueden ser los "Buffer Overflow". Los "Buffer Overflows" consisten en explotar una debilidad relacionada con los buffers que la aplicación usa para almacenar las entradas de usuario. Por ejemplo, cuando el usuario escribe una dirección en formato URL ésta se guarda en un buffer para luego procesarla. Si no se realizan las oportunas operaciones de comprobación, un usuario podría manipular estas direcciones. Los protocolos usados pueden ser HTTP, pero también otros menos conocidos, internos de cada explorador, como el "res:" o el "mk:". Precisamente existen fallos de seguridad del tipo "Buffer Overflow" en la implementación de estos dos protocolos. Para poder lanzar este tipo de ataques hay que tener un buen conocimiento de lenguaje Assembler y de la estructura interna de la memoria del Sistema Operativo utilizado. También se puede citar el fallo de seguridad descubierto por Cybersnot Industries relativo a los ficheros ".lnk" y ".url"de Windows 95 y NT respectivamente. Algunas versiones de Microsoft Internet Explorer podían ser utilizadas para ejecutar la aplicación que se deseara siempre que existiera en la computadora de la víctima, por ejemplo el tan conocido y temido format.com.

2.16.4. Explotación de errores de diseño, implementación y operación

Muchos sistemas están expuestos a "agujeros" de seguridad que son explotados para acceder a archivos, obtener privilegios o realizar sabotaje.

Estas vulnerabilidades ocurren por variadas razones, y miles de "puertas invisibles" son descubiertas en sistemas operativos, aplicaciones de software, protocolos de red, browsers de Internet, correo electrónico y toda clase de servicios informático disponible. Los Sistemas operativos abiertos como Unix y Linux tienen agujeros más conocidos y controlados que aquellos que existen en sistemas operativos cerrados como Windows. La importancia y ventaja del código abierto radica en miles de usuarios analizan dicho código en busca de posibles bugs y ayudan a obtener soluciones en forma inmediata. Constantemente se encuentra en Internet avisos de nuevos descubrimientos de problemas de seguridad y herramientas de Hacking que los explotan, por lo que hoy también se hace indispensable contar con productos que conocen esas debilidades, puedan diagnosticarlas y actualizar el programa afectado con el parche adecuado.

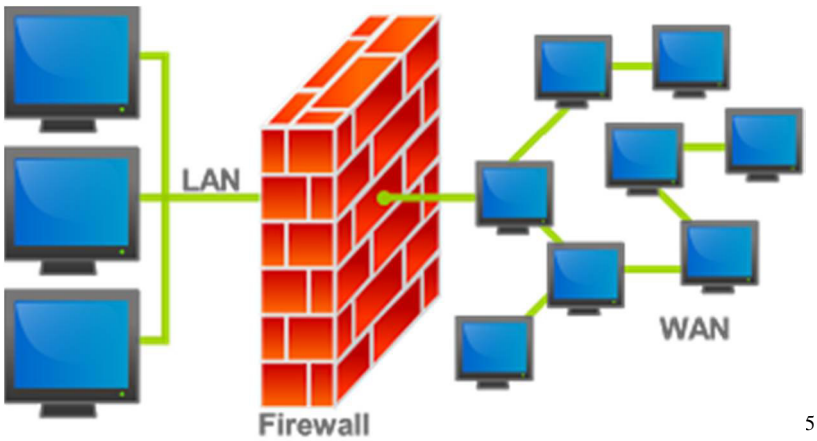
2.17. Teoría sobre el firewall

2.17.1. Concepto de Firewall

Un firewall es un software o una máquina segura y confiable que se asienta entre una red privada y una red pública. La máquina firewall se configura con un conjunto de reglas que determinan a qué tráfico de red se le permitirá pasar y cuál será bloqueado o rechazado. En algunas organizaciones grandes, puede que encuentre un firewall localizado dentro de la red corporativa para separar áreas sensibles de la organización de otros empleados. Algunos casos de criminalidad informática acontecen dentro de la misma organización, no sólo provienen de fuera. Se pueden construir un firewall en una variedad de maneras. La configuración más sofisticada involucra un número de máquinas separadas y se conoce como red perimetral. Dos máquinas, denominadas estranguladoras actúan como "filtros" para permitir pasar sólo ciertos tipos de tráfico de red, y entre estos estranguladores residen servidores de red como una pasarela de correo o un servidor intermediario de “World Wide Web”. Esta configuración puede resultar muy segura y permite de forma fácil un amplio rango de control sobre quién puede conectarse tanto desde dentro hacia fuera cómo desde fuera hacia dentro. Este tipo de configuración debería ser el utilizado por las grandes organizaciones.

Sin embargo, típicamente los firewalls son máquinas únicas que sirven todas estas funciones. Esto es algo menos seguro, porque si hay alguna debilidad en la propia máquina del cortafuego que le permita a alguien conseguir el acceso al mismo firewall, la seguridad de toda la red habrá sido comprometida. Sin embargo, estos tipos de firewall son más baratos y fáciles de mantener que la configuración más sofisticada descrita arriba.

Gráfico 2. 17: Esquema de un Firewall



[5]

Fuente: (Malf Kirch, 2000)

2.17.2. Tipos de Firewalls

Los firewalls tradicionales son de hardware, es decir, un dispositivo específico instalado en una red para levantar una defensa y proteger a la red del exterior. Son los utilizados en entorno profesionales: el administrador de red define una serie de reglas para permitir el acceso y detiene los intentos de conexión no permitidos. Los firewalls personales son programas que filtran el tráfico que entra y sale de una computadora. Una vez instalados, el usuario debe definir el nivel de seguridad: permite o deniega el acceso de determinados programas a Internet de forma temporal o definitiva y autoriza o no los accesos desde el exterior.

[5] LAN (Local Area Network o Red de área local), WAN (Wide Area Network o Red de área extendida)

Sus ventajas más notorias:

- **Protege de intrusiones.-** Solamente entran a la red las personas autorizadas basadas en la política de la red en base a las configuraciones.
- **Optimización de acceso.-** Identifica los elementos de la red internos y optimiza que la comunicación entre ellos sea más directa si así se desea. Esto ayuda a reconfigurar rápida y fácilmente los parámetros de seguridad.
- **Protección de información privada.-** Permite el acceso solamente a quien tenga privilegios a la información de cierta área o sector de la red.
- **Protección contra virus.-** Evita que la red se vea infestada por nuevos virus que sean liberados.

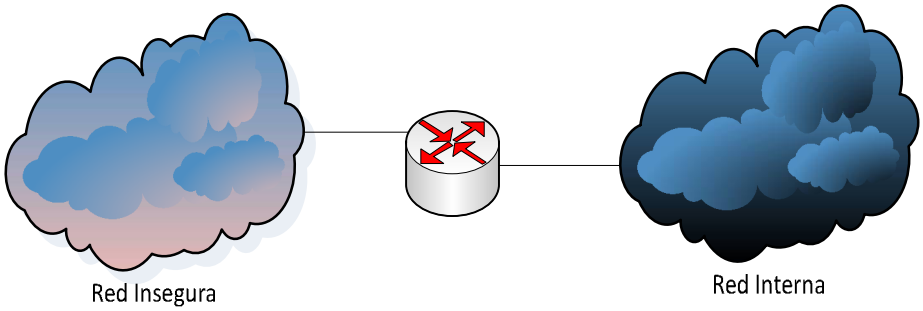
Aunque el usuario medio pueda creer que eso de los ataques no es algo que le pueda suceder en su casa a su computadora, el firewall se convierte un elemento imprescindible si se utiliza mucho la computadora y se está conectada permanentemente mediante ADSL o cable. El firewall evitará la entrada de los programas que rastrean direcciones IP, a la caza de conexiones por banda ancha que parasitar, a la vez que frustrará los intentos de los programas espía de robar datos del PC y de los troyanos de abrir brechas de seguridad. En Internet se pueden encontrar versiones no profesionales de firewalls, suficientes para el usuario doméstico, que se pueden descargar de forma gratuita. También hay firewalls integrados en los programas antivirus o en el propio sistema operativo. El que viene con Windows 7 no es demasiado seguro porque, al contrario que otros firewalls, sólo vigila las conexiones entrantes, mientras que el tráfico de salida no está restringido. Los firewalls, por defecto, se activan siempre que se enciende la computadora. Hay que configurarlo con cuidado, pues puede ocurrir que no funcione el correo electrónico o no se abran páginas web en el navegador porque el “firewall” no permite a estos programas acceder a Internet. Para eso concentran todo el flujo entrante y saliente entre la PC e Internet y bloquea los pedidos de enlaces no solicitados por el usuario potencialmente inseguro, instalaciones clandestinas de programas y algunos hasta bloquean pop ups, publicidades, etc.

2.17.3. Topologías de Firewalls

Existen diferentes formas de proteger una red mediante firewalls, como se dijo anteriormente, cada implementación depende de forma específica de la organización y sus necesidades, por tanto no existe una topología única que garantice la seguridad de cualquier red, sino una política de seguridad que tras un estudio profundo de la organización en cuestión, debe generar los lineamientos que aporten al diseño más adecuado, en este caso, de la topología de firewall a implementar:

Router ACLs: Constituye la forma más sencilla e insegura de topología, en la cual el router desempeña funciones de enrutamiento y filtrado de paquetes, es decir, es el dispositivo encargado de las comunicaciones y seguridad de la red. Este tipo de diseño presenta la desventaja de exponer la red a un gran número de ataques debido a su naturaleza stateless. Fue una solución aceptable en su época, que en la actualidad no debería implementarse por ningún motivo.

Gráfico 2. 18: Filtrado mediante Router ACLs

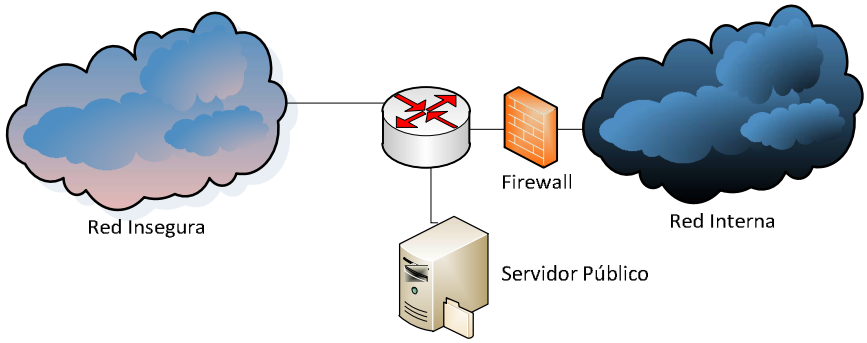


Fuente: (Dragon JAR, 2013)

Servidores conectados directamente a Red Insegura: Servidores que ofrecen servicios al exterior de la red se encuentran conectados directamente al router y no al firewall, que protege la red interna. Toda la seguridad de los servidores depende de sí mismos, es decir, deben implementar diferentes mecanismos y técnicas de endurecimiento para soportar ataques desde la red insegura y continuar funcionando.

Como recomendación deben tener el mínimo de paquetes instalados y servicios habilitados, así como endurecimiento a nivel de sistema operativo y servicios. El acceso desde los servidores públicos hacia la red interna es estrictamente prohibido debido a que pueden ser utilizados como salto hacia la red interna.

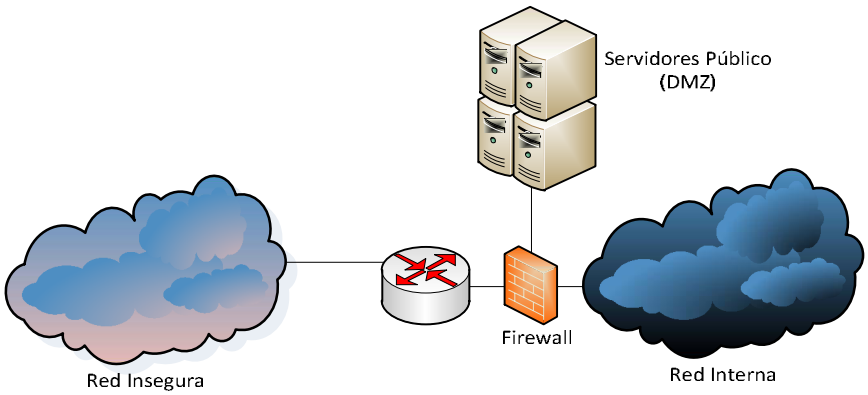
Gráfico 2. 19: Servidores conectados directamente a la red insegura



Fuente: (Dragon JAR, 2013)

Demilitarized Zone (Single Firewall): Tanto los servidores públicos como la red interna son protegidos por un firewall. El área en la que se ubican los servidores públicos se conoce como zona desmilitarizada, que puede definirse como un área pública protegida que ofrece servicios al interior y exterior de la red. La red interna se comunica con la DMZ mediante enrutamiento no deberían compartir el mismo segmento de red por razones obvias de seguridad. Este tipo de diseño es muy común, debido a su fácil implementación, seguridad y control del flujo de tráfico.

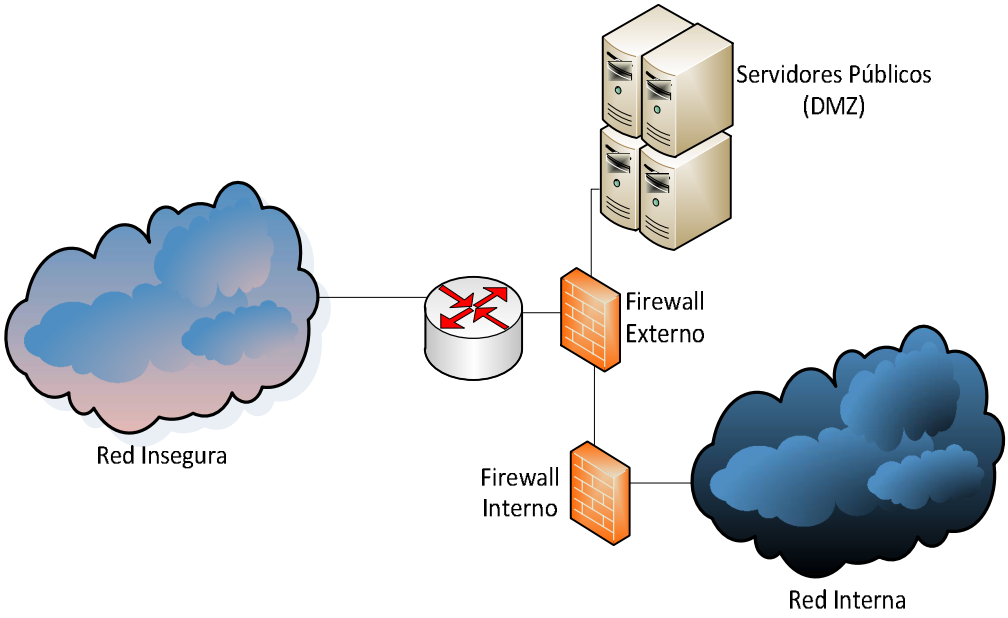
Gráfico 2. 20: Zona desmilitarizada con un simple firewall



Fuente: (Dragon JAR, 2013)

Demilitarized Zone (Dual Firewall): El diseño de firewall dual adiciona un gateway firewall para controlar y proteger la red interna, una de las razones es la protección de los servidores públicos frente a ataques provenientes de la red interna, como es bien conocido la mayor cantidad de ataques son generados desde dentro de la red. Ofrece mayor seguridad para la red interna al no contar con un punto único de ataque como en las anteriores topologías. Como desventajas puede decirse que tiene mayor dificultad de configuración y monitoreo, así como mayores costos en hardware y software. Su implementación es adecuada para redes grandes en las que hay flujo de tráfico importante entre la red interna y la DMZ, donde también se demanda mayor seguridad para la red LAN. Algunos expertos en seguridad afirman que para esta arquitectura deberían implementarse dos tipos diferentes de firewalls (fabricantes distintos), debido a que si un atacante logra pasar el firewall exterior, ya tendría suficiente información para superar el firewall interno (asumiendo que es de la misma clase), ya que tendrían similar configuración al firewall ya violentado.

Gráfico 2. 21: Zona desmilitarizada con doble firewall



Fuente: (Dragon JAR, 2013)

2.18. Tecnología de protección

2.18.1. Firewalls por software

Es el más común y utilizado en la mayoría de las PYMES[6] ecuatorianas. Se trata de un software que instalamos en nuestra computadora, por lo que sólo va a proteger la computadora en la que está instalada. Tal como vimos en la definición, un Firewall tiene la misión de filtrar el tráfico de nuestra red, ya sea de entrada o de salida, para evitar intrusiones no deseadas en nuestra computadora o bien la salida de datos del mismo. En el mercado hay una gran variedad de programas de este tipo, tanto independientes como formando parte de un paquete junto a un antivirus. El propio Windows 7 incorpora un Firewall que es bastante bueno, y especialmente el que incorpora el nuevo Windows 8. Un firewall gratuito es un Software que se puede instalar y utilizar libremente, o no, en la computadora. Son firewalls básicos que monitorean y bloquean, siempre que necesario, el tráfico de Internet.

Las características de un firewall por software son:

- Los gratuitos se incluyen con el sistema operativo y normalmente son para uso personal
- Pueden ser fácilmente integrados con otros productos de seguridad.
- No necesita de hardware adicional para instalarlo en la computadora.
- Un firewall de este tipo es el básico que debe existir en una computadora y no hay razones que justifiquen la no utilización de, por lo menos, un desktop firewall.
- EL firewall por software más común es el que viene incluido en el Sistema Operativo Windows, el cual muestra alertas al usuario de su activación o desactivación, pero al pertenecer a un sistema propietario las configuraciones y cambios que se pueden hacer al mismo son mínimas.

[6] PYMES, Siglas que significa "Pequeñas y medianas empresas"

2.18.2. Firewalls por hardware

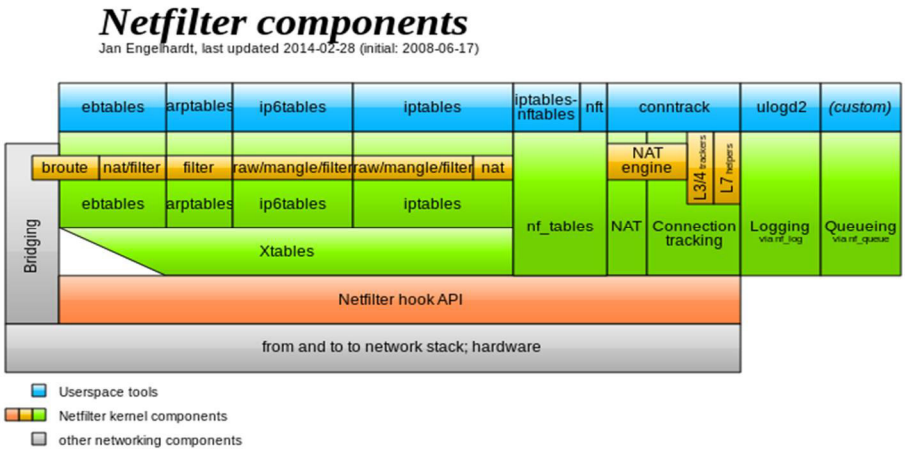
Un firewall por Hardware[7] viene normalmente instalado en los routers que utilizamos para acceder a Internet, lo que significa que todas las computadoras que estén detrás del router estarán protegidas por un firewall que está incluido en el dispositivo. La configuración de un firewall por hardware es más complicada que una instalación de un firewall por software y es normalmente realizada a través del navegador que se utiliza para acceder a Internet. La diferencia de precio entre un router con firewall y un router sin firewall es muy pequeña, por eso es recomendable comprar un firewall con esta protección.

2.19. Iptables

2.19.1. Historia

Netfilter es un framework disponible en el núcleo Linux que permite interceptar y manipular paquetes de red. Dicho framework permite realizar el manejo de paquetes en diferentes estados del procesamiento. Netfilter es también el nombre que recibe el proyecto que se encarga de ofrecer herramientas libres para firewalls basados en Linux.

Gráfico 2. 22: Netfilter



Fuente: Netfilter

[7] Posteriormente en el siguiente capítulo, se detallara de una mejor manera las características y funcionalidades de ambos tipos de Firewall

El componente más popular construido sobre Netfilter es iptables, una herramienta de firewall que permite no solamente filtrar paquetes, sino también realizar traducción de direcciones de red (NAT) para IPv4 o mantener registros de log. El proyecto Netfilter no sólo ofrece componentes disponibles como módulos del núcleo sino que también ofrece herramientas de espacio de usuario y librerías. Iptables es el nombre de la herramienta de espacio de usuario mediante la cual el administrador puede definir políticas de filtrado del tráfico que circula por la red. El nombre iptables se utiliza frecuentemente de forma errónea para referirse a toda la infraestructura ofrecida por el proyecto Netfilter. Sin embargo, el proyecto ofrece otros subsistemas independientes de iptables tales como el connection tracking system o sistema de seguimiento de conexiones, que permite encolar paquetes para que sean tratados desde espacio de usuario. Iptables es un software disponible en prácticamente todas las distribuciones de Linux actuales.

Antes de iptables, los programas más usados para crear firewalls en Linux eran ipchains en el núcleo Linux 2.2 e ipfwadm en el núcleo Linux 2.0, que a su vez se basaba en ipfw de BSD. Tanto ipchains como ipfwadm alteran el código de red para poder manipular los paquetes, ya que no existía un framework general para el manejo de paquetes hasta la aparición de netfilter. Iptables mantiene la idea básica introducida en Linux con ipfwadm: listas de reglas en las que se especifica qué matchear dentro de un paquete y qué hacer con ese paquete. Ipchains agrega el concepto de cadenas de reglas (chains) e iptables extendió esto a la idea de tablas: se consultaba una tabla para decidir si había que NAT-ear un paquete, y se consultaba otra para decidir cómo filtrar un paquete. Adicionalmente, se modificaron los tres puntos en los que se realiza el filtrado en el viaje de un paquete, de modo que un paquete pase solo por un punto de filtrado. Mientras que ipchains e ipfwadm combinan filtrado de paquetes y NAT, específicamente tres tipos de NAT, llamados masquerading o enmascaramiento de IP, port forwarding o redireccionamiento de puertos, y redirection o redirección, netfilter hace posible por su parte separar las operaciones sobre los paquetes en tres partes: packet filtering (filtrado de paquetes), connection tracking (seguimiento de conexiones) y Network Address Translation (NAT o traducción de direcciones de red).

Cada parte se conecta a las herramientas de netfilter en diferentes puntos para acceder a los paquetes. Los subsistemas de seguimiento de conexiones y NAT son más generales y poderosos que los que realizaban ipchains e ipfwadm. Esta división permite a iptables, a su vez, usar la información que la capa de seguimiento de conexiones ha determinado acerca del paquete: esta información estaba antes asociada a NAT. Esto hace a iptables superior a ipchains, ya que tiene la habilidad de monitorizar el estado de una conexión y redirigir, modificar o detener los paquetes de datos basados en el estado de la conexión y no solamente por el origen, destino o contenido del paquete. Un firewall que utilice iptables de este modo se llama firewall stateful, contrario a ipchains que solo permite crear un firewall stateless.

Podemos decir entonces que ipchains no está al tanto del contexto completo en el cual un paquete surge, mientras que iptables sí y por lo tanto iptables puede hacer mejores decisiones sobre el futuro de los paquetes y las conexiones. Iptables, el subsistema NAT y el subsistema de seguimiento de conexiones son también extensibles, y muchas extensiones ya están incluidas en el paquete básico de iptables, tal como la extensión ya mencionada que permite la consulta del estado de la conexión. Extensiones adicionales se distribuyen junto a la utilidad iptables, como parches al código fuente del núcleo junto con una herramienta llamada patcho-matic que permite aplicar los parches.

2.19.2. Funcionamiento

Iptables permite al administrador del sistema definir reglas acerca de qué hacer con los paquetes de red. Las reglas se agrupan en cadenas: cada cadena es una lista ordenada de reglas. Las cadenas se agrupan en tablas: cada tabla está asociada con un tipo diferente de procesamiento de paquetes. Cada regla especifica qué paquetes la cumplen (match) y un objetivo que indica qué hacer con el paquete si éste cumple la regla. Cada paquete de red que llega a una computadora o que se envía desde una computadora recorre por lo menos una cadena y cada regla de esa cadena se comprueba con el paquete. Si la regla cumple con el datagrama, el recorrido se detiene y el destino de la regla dicta lo que se debe hacer con el paquete.

Si el paquete alcanza el fin de una cadena predefinida sin haberse correspondido con ninguna regla de la cadena, la política de destino de la cadena dicta qué hacer con el paquete. Si el paquete alcanza el fin de una cadena definida por el usuario sin haber cumplido ninguna regla de la cadena o si la cadena definida por el usuario está vacía, el recorrido continúa en la cadena que hizo la llamada, lo que se denomina implicit target RETURN o RETORNO de destino implícito. Solo las cadenas predefinidas tienen políticas. En iptables, las reglas se agrupan en cadenas. Una cadena es un conjunto de reglas para paquetes IP, que determinan lo que se debe hacer con ellos. Cada regla puede desechar el paquete de la cadena (cortocircuito), con lo cual otras cadenas no serán consideradas. Una cadena puede contener un enlace a otra cadena: si el paquete pasa a través de esa cadena entera o si cumple una regla de destino de retorno, va a continuar en la primera cadena. No hay un límite respecto de cuán anidadas pueden estar las cadenas. Hay tres cadenas básicas (INPUT, OUTPUT y FORWARD: ENTRADA, SALIDA y REENVÍO) y el usuario puede crear tantas como desee. Una regla puede ser simplemente un puntero a una cadena.

2.19.3. Tablas

Hay tres tablas ya incorporadas, cada una de las cuales contiene ciertas cadenas predefinidas. Es posible crear nuevas tablas mediante módulos de extensión. El administrador puede crear y eliminar cadenas definidas por usuarios dentro de cualquier tabla. Inicialmente, todas las cadenas están vacías y tienen una política de destino que permite que todos los paquetes pasen sin ser bloqueados o alterados.

Filter Table (Tabla de filtros): Esta tabla es la responsable del filtrado, es decir, de bloquear o permitir que un paquete continúe su camino. Todos los paquetes pasan a través de la tabla de filtros.

Contiene las siguientes cadenas predefinidas y cualquier paquete pasará por una de ellas:

- INPUT chain (Cadena de ENTRADA) — Todos los paquetes destinados a este sistema atraviesan esta cadena (y por esto se la llama algunas veces LOCAL_INPUT o ENTRADA_LOCAL)

- OUTPUT chain (Cadena de SALIDA) — Todos los paquetes creados por este sistema atraviesan esta cadena (a la que también se la conoce como LOCAL_OUTPUT o SALIDA_LOCAL)

- FORWARD chain (Cadena de REDIRECCIÓN) — Todos los paquetes que meramente pasan por este sistema para ser encaminados a su destino recorren esta cadena

Tabla de traducción de direcciones de red: Esta tabla es la responsable de configurar las reglas de reescritura de direcciones o de puertos de los paquetes. El primer paquete en cualquier conexión pasa a través de esta tabla; los veredictos determinan como van a reescribirse todos los paquetes de esa conexión. Contiene las siguientes cadenas redefinidas:

- **PREROUTING chain (Cadena de PRERUTEO):** Los paquetes entrantes pasan a través de esta cadena antes de que se consulte la tabla de ruteo local, principalmente para DNAT (destination-NAT o traducción de direcciones de red de destino)
- **POSTROUTING chain (Cadena de POSRUTEO):** Los paquetes salientes pasan por esta cadena después de haberse tomado la decisión del ruteo, principalmente para SNAT (source-NAT o traducción de direcciones de red de origen)
- **OUTPUT chain (Cadena de SALIDA):** Permite hacer un DNAT limitado en paquetes generados localmente

Mangle table (Tabla de destrozo): Esta tabla es la responsable de ajustar las opciones de los paquetes, como por ejemplo la calidad de servicio. Todos los paquetes pasan por esta tabla. Debido a que está diseñada para efectos avanzados, contiene todas las cadenas predefinidas posibles:

- **PREROUTING chain (Cadena de PRERUTEO):** Todos los paquetes que logran entrar a este sistema, antes de que el ruteo decida si el paquete debe ser reenviado (cadena de REENVÍO) o si tiene destino local (cadena de ENTRADA)
- **INPUT chain (Cadena de ENTRADA):** Todos los paquetes destinados para este sistema pasan a través de esta cadena
- **FORWARD chain (Cadena de REDIRECCIÓN): Todos** los paquetes que exactamente pasan por este sistema pasan a través de esta cadena
- **OUTPUT chain (Cadena de SALIDA):** Todos los paquetes creados en este sistema pasan a través de esta cadena
- **POSTROUTING chain (Cadena de POSRUTEO):** Todos los paquetes que abandonan este sistema pasan a través de esta cadena

Además de las cadenas ya incorporadas, el usuario puede crear todas las cadenas definidas por el usuario que quiera dentro de cada tabla, las cuales permiten agrupar las reglas en forma lógica. Cada cadena contiene una lista de reglas. Cuando un paquete se envía a una cadena, se lo compara, en orden, contra cada regla en la cadena. La regla especifica qué propiedades debe tener el paquete para que la regla coincida, como número de puerto o dirección IP. Si la regla no coincide, el procesamiento continúa con la regla siguiente. Si la regla, por el contrario, coincide con el paquete, las instrucciones de destino de las reglas se siguen (y cualquier otro procesamiento de la cadena normalmente se aborta). Algunas propiedades de los paquetes solo pueden examinarse en ciertas cadenas (por ejemplo, la interfaz de red de salida no es válida en la cadena de ENTRADA). Algunos destinos solo pueden usarse en ciertas cadenas y/o en ciertas tablas (por ejemplo, el destino SNAT solo puede usarse en la cadena de POSRUTEO de la tabla de traducción de direcciones de red). El destino de una regla puede ser el nombre de una cadena definida por el usuario o uno de los destinos ya incorporados ACCEPT, DROP, QUEUE, o RETURN (aceptar, descartar, encolar o retornar, respectivamente). Cuando un destino es el nombre de una cadena definida por el usuario, al paquete se lo dirige a esa cadena para que sea procesado.

Si el paquete consigue atravesar la cadena definida por el usuario sin que ninguna de las reglas de esa cadena actúe sobre él, el procesamiento del paquete continúa donde había quedado en la cadena actual. Estas llamadas entre cadenas se pueden anidar hasta cualquier nivel deseado. Existen los siguientes destinos ya incorporados:

Accept (aceptar): Este destino hace que netfilter acepte el paquete. El significado de esto depende de cuál sea la cadena realizando esta aceptación. Un paquete que se acepta en la cadena de ENTRADA se le permite ser recibido por el sistema (host), un paquete que se acepta en la cadena de SALIDA se le permite abandonar el sistema y un paquete que se acepta en la cadena de REDIRECCIÓN se le permite ser encaminado (routing) a través del sistema.

Drop (descartar): Este destino hace que netfilter descarte el paquete sin ningún otro tipo de procesamiento. El paquete simplemente desaparece sin ningún tipo de indicación al sistema o aplicación de origen, de que fue descartado en el sistema de destino. Esto se refleja en el sistema que envía el paquete a menudo, como un communication timeout (alcance del máximo tiempo de espera en la comunicación), lo que puede causar confusión, aunque el descarte de paquetes entrantes no deseados se considera a veces una buena política de seguridad, pues no da ni siquiera el indicio a un posible atacante de que el sistema destino existe.

Queue (encolar): Este destino hace que el paquete sea enviado a una cola en el espacio de usuario. Una aplicación puede usar la biblioteca libipq, también parte del proyecto netfilter/iptables, para alterar el paquete. Si no hay ninguna aplicación que lea la cola, este destino es equivalente a DROP.

Return (retorno): Hace que el paquete en cuestión deje de circular por la cadena en cuya regla se ejecutó el destino RETURN. Si dicha cadena es una subcadena de otra, el paquete continuará por la cadena superior como si nada hubiera pasado. Si por el contrario la cadena es una cadena principal (por ejemplo la cadena INPUT), al paquete se le aplicará la política por defecto de la cadena en cuestión (ACCEPT, DROP o similar). Hay muchos destinos de extensión disponibles. Algunos de los más comunes son:

Reject (rechazo): Este destino tiene el mismo efecto que "DROP", salvo que envía un paquete de error a quien envió originalmente. Se usa principalmente en las cadenas de ENTRADA y de REDIRECCIÓN de la tabla de filtrado. El tipo de paquete se puede controlar a través del parámetro "--reject-with". Un paquete de rechazo puede indicar explícitamente que la conexión ha sido filtrada (un paquete ICMP filtrado administrativamente por conexión), aunque la mayoría de los usuarios prefieren que el paquete indique simplemente que la computadora no acepta ese tipo de conexión (tal paquete será un paquete tcp-reset para conexiones TCP denegadas, un icmpport-unreachable para sesiones UDP denegadas o un icmp-protocol-unreachable para paquetes no TCP y no UDP). Si el parámetro "--reject-with" no se especifica, el paquete de rechazo por defecto es siempre icmp-port-unreachable.
LOG (bitácora). Este destino lleva un log o bitácora del paquete. Puede usarse en cualquier cadena en cualquier tabla, y muchas veces se usa para debuggear (análisis de fallos, como ser la verificación de qué paquetes están siendo descartados).

Ulog: Este destino lleva un log o bitácora del paquete, pero no de la misma manera que el destino LOG. El destino LOG le envía información al log del núcleo, pero ULOG hace multidifusión de los paquetes que coincidan con esta regla a través de un socket netlink, de manera que programas del espacio de usuario puedan recibir este paquete conectándose al socket.

Dnat: Este destino hace que la dirección y opcionalmente el puerto de destino del paquete sean reescritos para traducción de dirección de red. Mediante la opción "--todestination" debe indicarse el destino a usar. Esto es válido solamente en las cadenas de SALIDA y PRERUTEO dentro de la tabla de nat. Esta decisión se recuerda para todos los paquetes futuros que pertenecen a la misma conexión y las respuestas tendrán su dirección y puerto de origen cambiado al original, es decir, la inversa de este paquete.

Snat: Este destino hace que la dirección (y opcionalmente el puerto) de origen del paquete sean reescritos para traducción de dirección de red. Mediante la opción "--tosource" debe indicarse el origen a usar. Esto es válido solamente en la cadena de POSRUTEO dentro de la tabla de NAT y, como DNAT, se recuerda para todos los paquetes que pertenecen a la misma conexión.

Masquerade: Esta es una forma especial, restringida de SNAT para direcciones IP dinámicas, como las que proveen la mayoría de los proveedores de servicios de Internet (ISPs) para módems o línea de abonado digital (DSL). En vez de cambiar la regla de SNAT cada vez que la dirección IP cambia, se calcula la dirección IP de origen a la cual hacer NAT fijándose en la dirección IP de la interfaz de salida cuando un paquete coincide con esta regla. Adicionalmente, recuerda cuales conexiones usan Masquerade y si la dirección de la interfaz cambia (por ejemplo, por reconectarse al ISP), todas las conexiones que hacen NAT a la dirección vieja se olvidan. Iptables es una aplicación en espacio de usuario que le permite a un administrador de sistema configurar las tablas, cadenas y reglas de netfilter (descritas más arriba). Debido a que iptables requiere privilegios elevados para operar, el único que puede ejecutarlo es el superusuario. En la mayoría de los sistemas Linux, iptables está instalado como /sbin/iptables.

Opciones comunes: En cada una de las formas de invocación de iptables que se muestra a continuación, las siguientes opciones comunes están disponibles:

- T tabla: Hace que el comando se aplique a la tabla especificada. Si esta opción se omite, el comando se aplica a la tabla filter por defecto.
- V: Produce una salida con detalles (del inglés, verbose).
- N: Produce una salida numérica (es decir, números de puerto en lugar de nombres de servicio y direcciones IP en lugar de nombres de dominio).
- Line-numbers: Cuando se listan reglas, agrega números de línea al comienzo de cada regla, correspondientes a la posición de esa regla en su cadena.

2.19.4. Especificación de las reglas de seguridad

La mayoría de las formas de comandos de iptables requieren que se les indiquen una especificación de reglas, que es usada para comparar un subconjunto particular del tráfico de paquetes de red procesados por una cadena. La especificación de regla incluye también un destino que especifica qué hacer con paquetes que son comparados por la regla. Las siguientes opciones se usan frecuentemente combinadas unas con otras para crear especificaciones de reglas.

- J destino
- Jump destino

Especifica el destino de una regla. El destino es el nombre de una cadena definida por el usuario, creada usando la opción -N, uno de los destinos ya incorporados, ACCEPT, DROP, QUEUE, o RETURN, o un destino de extensión, como REJECT, LOG, DNAT, o SNAT. Si esta opción es omitida en una regla, entonces el comparado de la regla no tendrá efecto en el destino de un paquete, pero los contadores en la regla se incrementarán.

- i [!] in-interface
- in-interface [!] in-interface

Nombre de una interfaz a través de la cual un paquete va a ser recibido solo para paquetes entrando en las cadenas de INPUT, FORWARD y PREROUTING. Cuando se usa el argumento "!" antes del nombre de la interfaz, el significado se invierte. Si el nombre de la interfaz termina con "+", entonces cualquier interfaz que comience con este nombre será comparada. Si esta opción se omite, se comparará todo nombre de interfaz.

- [!] out-interface
- out-interface [!] out-interface

Nombre de una interfaz a través de la cual un paquete va a ser enviado (para paquetes entrando en las cadenas de FORWARD, OUTPUT y POSTROUTING). Cuando se usa el argumento "!" antes del nombre de la interfaz, el significado se invierte. Si el nombre de la interfaz termina con "+", entonces cualquier interfaz que comience con este nombre será comparada. Si esta opción se omite, se comparará todo nombre de interfaz.

- p [!] protocol
- protocol [!] protocol

Compara paquetes del nombre de protocolo especificado. Si "!" precede el nombre de protocolo, se comparan todos los paquetes que no son el protocolo especificado. Nombres de protocolo válidos son icmp, udp, tcp... Una lista de todos los protocolos válidos puede encontrarse en el archivo /etc/protocols.

- s [!] origen[/prefijo]
- source [!] origen[/prefijo]

Compara paquetes IP viniendo de la dirección de origen especificada. La dirección de origen puede ser una dirección IP, una dirección IP con un prefijo de red asociado, o un nombre de terminal (hostname). Si "!" precede al origen, se comparan todos los paquetes que no vienen del origen especificado.

- d [!] destino[/prefijo]
- destination [!] destino[/prefijo]

Compara paquetes IP yendo a la dirección de destino especificada. La dirección de destino puede ser una dirección IP, una dirección IP con un prefijo de red asociado, o un nombre de terminal (hostname). Si "!" precede al origen, se matchean todos los paquetes que no van al destino especificado.

- destination-port [!] [puerto[[:puerto]]
- dport [!] [puerto[[:puerto]]

Matchea paquetes TCP o UDP (dependiendo del argumento a la opción -p) destinados a los puertos o rango de puertos (cuando se usa la forma puerto:puerto) especificados. Si "!" precede la especificación de puertos, se matchean todos los paquetes TCP o UDP que no están destinados a los puertos o rango de puertos especificados.

- source-port [!] [puerto[[:puerto]]
- sport [!] [puerto[[:puerto]]

Matchea paquetes TCP o UDP (dependiendo del argumento a la opción -p) que vienen de los puertos o rango de puertos (cuando se usa la forma puerto:puerto) especificados. Si "!" precede la especificación de puertos, se matchean todos los paquetes TCP o UDP que no vienen de los puertos o rango de puertos especificados.

- tcp-flags [!] mask comp

Matchea paquetes TCP que tienen marcadas o desmarcadas ciertas banderas del protocolo TCP. El primer argumento especifica las banderas a examinar en cada paquete TCP, escritas en una lista separada por comas (no se permiten espacios). El segundo argumento es otra lista separada por comas de banderas que deben estar marcadas dentro de las que se debe examinar. Estas banderas son: SYN, ACK, FIN, RST, URG, PSH, ALL, y NONE. Por lo tanto, la opción "--tcp-flags SYN, ACK, FIN, RST SYN" solo va a matchear paquetes con la bandera SYN marcada y las banderas ACK, FIN y RST desmarcadas.

- [!] –syn

Matchea paquetes TCP que tienen la bandera SYN marcada y las banderas ACK, FIN y RST desmarcadas. Estos paquetes son los que se usan para iniciar conexiones TCP. Al bloquear tales paquetes en la cadena de INPUT, se previenen conexiones TCP entrantes, pero conexiones TCP salientes no serán afectadas. Esta opción puede combinarse con otras, como --source, para bloquear o dejar pasar conexiones TCP entrantes solo de ciertas terminales o redes. Esta opción es equivalente a "--tcp-flags SYN, RST, ACK SYN". Si "!" precede a --syn, el significado de la opción se invierte.

2.20. Políticas de seguridad

La globalización de la economía ha exigido que las empresas implementen plataformas tecnológicas que soporten la nueva forma de hacer negocios. El uso de Internet para este fin, conlleva a que se desarrollen proyectos de seguridad informática que garanticen la integridad, disponibilidad y accesibilidad de la información.

La creación de políticas de seguridad es una labor fundamental que involucra las personas, los procesos y los recursos de la compañía. Los mismos se organizan siguiendo el esquema, normativo de seguridad, ISO17799 (mejores prácticas de seguridad) y que a continuación se presenta:

2.20.1. Seguridad Organizacional

Dentro de este, se establece el marco formal de seguridad que debe sustentar la institución, incluyendo servicios o contrataciones externas a la infraestructura de seguridad, Integrando el recurso humano con la tecnología, denotando responsabilidades y actividades complementarias como respuesta ante situaciones anómalas a la seguridad.

Nivel de Seguridad Organizativo:

- Seguridad Organizacional
- Políticas de Seguridad
- Excepciones de Responsabilidad
- Clasificación y Control de Activos
- Responsabilidad por los Activos
- Clasificación de la Información
- Seguridad Ligada al Personal
- Capacitación de Usuarios
- Respuestas a Incidentes y Anomalías de Seguridad

2.20.2. Seguridad Lógica

Trata de establecer e integrar los mecanismos y procedimientos, que permitan monitorear el acceso a los activos de información, que incluyen los procedimientos de administración de usuarios, definición de responsabilidades, perfiles de seguridad, control de acceso a las aplicaciones y documentación sobre sistemas, que van desde el control de cambios en la configuración de los equipos, manejo de incidentes, selección y aceptación de sistemas, hasta el control de software malicioso.

Nivel de Seguridad Lógico:

- Control de Accesos
- Administración del Acceso de Usuarios
- Seguridad en Acceso de Terceros
- Control de Acceso a la Red
- Control de Acceso a las Aplicaciones
- Monitoreo del Acceso y Uso del Sistema

2.20.3. Seguridad Física

Identifica los límites mínimos que se deben cumplir en cuanto a perímetros de seguridad, de forma que se puedan establecer controles en el manejo de equipos, transferencia de información y control de los accesos a las distintas áreas con base en la importancia de los activos.

Nivel de Seguridad Física:

- Seguridad Física
- Seguridad Física y Ambiental
- Seguridad de los Equipos
- Controles Generales

2.20.4. Seguridad Legal

Integra los requerimientos de seguridad que deben cumplir todos los empleados, socios y usuarios de la red institucional bajo la reglamentación de la normativa interna de políticas y manuales de procedimientos de la Universidad de Oriente en cuanto al recurso humano, sanciones aplicables ante faltas cometidas, así como cuestiones relacionadas con la legislación del país y contrataciones externas.

Los niveles de seguridad fueron organizados constatando un enfoque objetivo de la situación real de la institución, desarrollando cada política con sumo cuidado sobre qué activo proteger, de qué protegerlo cómo protegerlo y por qué protegerlo.

Nivel de Seguridad Legal:

- Seguridad Legal
- Conformidad con la Legislación
- Cumplimiento de Requisitos Legales
- Revisión de Políticas de Seguridad y Cumplimiento Técnico
- Consideraciones Sobre Auditorias de Sistemas

2.21. Honey Pots como complemento de firewall

Se llama honeypot (en inglés, tarro de miel) a una herramienta usada en el ámbito de la seguridad informática para atraer y analizar el comportamiento de los atacantes en Internet. Parece una contradicción, puesto que la función habitual de las herramientas de seguridad es exactamente la contraria: mantener alejados a los atacantes o impedir sus ataques. Sin embargo, desde hace unos años, se utilizan los honeypots para atraer a atacantes hacia un entorno controlado, e intentar conocer más detalles sobre cómo estos realizan sus ataques, e incluso descubrir nuevas vulnerabilidades. Lance Spitzner, consultor y analista informático experto en seguridad, construyó a comienzos del año 2000 una red de seis computadoras en su propia casa. Esta red la diseñó para estudiar el comportamiento y formas de actuación de los atacantes. Fue de los primeros investigadores en adoptar la idea, y hoy es uno de los mayores expertos en honeypots, precursor del proyecto honeynet (www.honeynet.org), en marcha desde 1999, y autor del libro "Honeypots: Tracking Hackers". Su sistema estuvo durante casi un año de prueba, desde abril del 2000 a febrero de 2001, guardando toda la información que se generaba. Los resultados hablaban por sí solos: en los momentos de mayor intensidad de los ataques, comprobaba que las vías de acceso más comunes a las computadoras de su casa eran escaneadas, desde el exterior de su red, hasta 14 veces al día, utilizando herramientas de ataque automatizadas. Desde entonces, se ha creado toda una comunidad de desarrolladores aglutinados alrededor de honeynet.org que ofrecen todo tipo de herramientas y consejos para utilizar estas herramientas.

2.21.1. Clasificación

Un honeypot puede ser tan simple como una computadora que ejecuta un programa, que analiza el tráfico que entra y sale de una computadora hacia Internet, "*escuchando*" en cualquier número de puertos. El procedimiento consiste en mantener una debilidad o vulnerabilidad en un programa, en el sistema operativo, en el protocolo, o en cualquier otro elemento del equipo susceptible de ser atacado, que motive al atacante a usarlo, de manera que se muestre dispuesto a emplear todas sus habilidades para explotar dicha debilidad y obtener acceso al sistema. Por otro lado, un honeypot puede ser tan complejo como una completa red de computadoras completamente funcionales, funcionando bajo distintos sistemas operativos y ofreciendo gran cantidad de servicios. Cuando algún sistema que está incluido en dicha red sea atacado de alguna forma, se advierte al administrador. Otra opción muy utilizada es crear honeypots completamente virtuales: programas específicamente diseñados para simular una red, engañar al atacante con direcciones falsas, IP fingidas y computadoras inexistentes, con el único fin de confundirlo o alimentar el ataque para analizar nuevos métodos. Si algo tienen en común los honeypots es que no guardan ninguna información relevante, y si lo parece, si se muestran contraseñas o datos de usuario, son completamente ficticios.

Gráfico 2. 23: Honeypot sitio Web

Fuente: Honeypot

Los honeypots son clasificados según diferentes categorías:

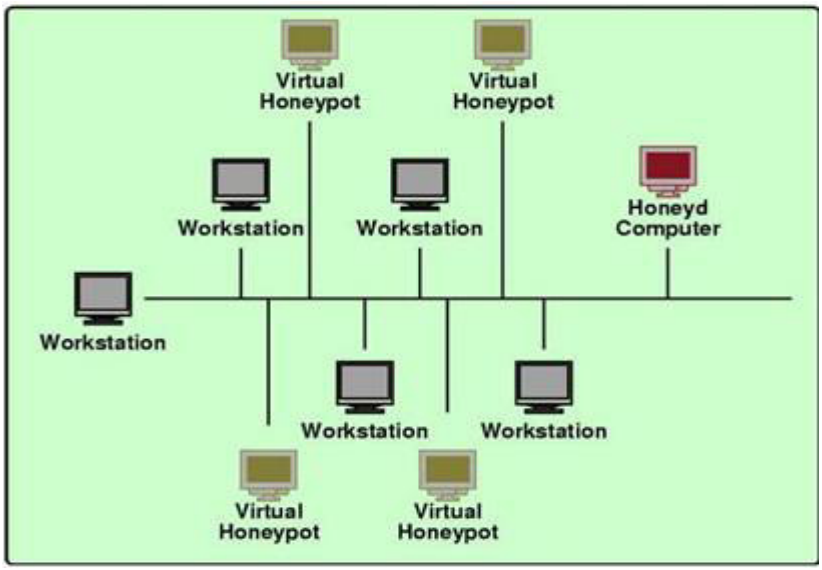
2.21.2. Honeypots de alta interacción

Suelen ser usados por las compañías en sus redes internas. Estos honeypots están construidos con máquinas reales, o consisten en una sola máquina real con un sistema operativo "normal", como el que podría utilizar cualquier usuario. Se colocan en la red interna en producción. Si están bien configurados, cualquier intento de acceso a ellos debe suponer una alerta a tener en cuenta. Puesto que no tienen ninguna utilidad más que la de ser atacados, el hecho de que de alguna forma se intente acceder a ese recurso significa por definición que algo no va bien. Cada interacción con ese honeypot se considera sospechosa por definición. Todo este tráfico debe ser convenientemente monitorizado y almacenado en una zona segura de la red, y a la que un potencial atacante no tenga acceso. Esto es así porque, si se tratase de un ataque real, el intruso podría a su vez borrar todo el tráfico generado por él mismo, las señales que ha ido dejando, con lo que el ataque pasaría desapercibido y el honeypot no tendría utilidad real. Las ventajas que ofrecen los honeypots de alta interacción es que pueden prevenir ataques de todo tipo. Tanto los conocidos como los desconocidos.

Al tratarse de un sistema real, contiene todos los fallos de software conocidos y desconocidos que pueda albergar cualquier otro sistema. Si un atacante intenta aprovechar un fallo desconocido hasta el momento (llamados en el argot "*0 day*"), será la propia interacción con la máquina, para intentar explotar el fallo, lo que alerte del problema y ayude a descubrir ese nuevo fallo. En contraposición, por ejemplo un detector de intrusos (IDS) basado en firmas, podría alertar en la red de solo intentos de aprovechar fallos o ataques ya conocidos, para los que tiene firmas que le permiten reconocerlos. La ventaja del honeypot es que, sea el ataque nuevo o no, el intento de ataque alertará al administrador, y esto le permitirá estar alerta cuanto antes del potencial peligro. En este sentido, los honeypots se usan para mitigar los riesgos de las compañías, en el sentido tradicional de uso de las conocidas herramientas defensivas. Lo que la diferencia de los tradicionales firewalls o detectores de intrusos es su naturaleza "activa" en vez de pasiva.

De modo figurado un honeypot se muestra como un anzuelo, no como un muro de contención para evitar ataques, muy al contrario, busca dichos ataques y se encarga de "entretenerlos". Muchas compañías lo usan como un valor añadido más a sus elementos de seguridad, como complemento a sus herramientas típicas. Se obtiene así una fácil detección y reconocimiento de los ataques, de forma que pueden elaborar con esos datos estadísticas que ayudan a configurar de manera más efectiva sus herramientas pasivas. Conociendo cuanto antes los problemas de seguridad a los que más se atacan o los nuevos objetivos, más eficazmente podrá defenderse una compañía concreta contra ellos. Como toda herramienta destinada a mejorar la seguridad, los honeypots tienen sus ventajas e inconvenientes. Su mayor utilidad radica en su simpleza. Al ser un mecanismo cuyo único fin consiste en que intenten aprovechar sus debilidades, no realiza ningún servicio real, y el tráfico que transita a través de él va a ser muy pequeño. Si se detecta tráfico que va o viene hacia el sistema, casi con toda probabilidad va a ser una prueba, escaneo o ataque. El tráfico registrado en un sistema de este tipo es sospechoso por naturaleza, por lo que su gestión y estudio se simplifica en gran medida. Aunque, por supuesto, ocurran "falsos positivos", expresión que, en este caso, invierte su significado. Si un falso positivo se produce normalmente cuando una actividad sospechosa tomada como ataque no resulta serlo, en el ambiente de los honeypots, el falso positivo sería el tráfico gestionado por la máquina que no representa una amenaza.

Gráfico 2. 24: Honeypot de alta interacción



Fuente: Honeypots

Entre los problemas que se pueden producir por el trabajo con honeypots, destaca la posibilidad de que se vuelva en contra del administrador. Si no se diseña de una manera absolutamente estudiada, si no se ata cada cabo, si no se aísla convenientemente, el atacante puede acabar comprometiendo un sistema real y llegar a datos valiosos conectados al honeypot.

2.21.3. Honeypots de baja interacción

Suelen ser creados y gestionados por organizaciones dedicadas a la investigación del fraude en Internet, o cualquier tipo de organización que necesite investigar sobre las nuevas amenazas en la red. Son mucho más complejos de administrar y mantener, y la información que reciben debe ser lo más extensa posible, ésta debe ser organizada y analizada para que sea de utilidad. Se suelen tratar de sistemas específicos que emulan servicios, redes, pilas TCP o cualquier otro aspecto de un sistema real, pero sin serlo. Existe un "meta-sistema" detrás, invisible para el atacante, que está simulando ser cualquier cosa para la que esté programado ser. No tienen que implementar un comportamiento completo de un sistema o servicio. Normalmente simulan ser un servicio, y ofrecen respuesta a un subconjunto de respuestas simple. Por ejemplo, un honeypot que simule ser un servidor de correo, puede simular aceptar conexiones y permitir que se escriba en ellas un correo, aunque nunca llegará a enviarlo realmente.

Normalmente este tipo de honeypots no está destinado a "atrapar" atacantes reales, sino herramientas automatizadas. Un ser humano podría detectar rápidamente si se trata de un servidor real o no, bien por su experiencia o por otras características que le hagan sospechar que no se encuentra en un entorno real. Sin embargo, sistemas automatizados como programas de explotación automática, gusanos, virus, etc., programados específicamente para realizar una acción sobre un servicio, no detectarán nada extraño. Harán su trabajo intentando explotar alguna vulnerabilidad, el honeypot simulará ser explotado, y el administrador del honeypot obtendrá la información que desea. Este tipo de honeypots, tienen el problema de que en ellos resulta más complejo descubrir nuevos tipos de ataques.

Están preparados para simular ciertos servicios que se saben atacados, y a responder de cierta manera para que el ataque crea que ha conseguido su objetivo. Pero en ningún caso puede comportarse de formas para las que no está programado, por ejemplo para simular la explotación de nuevos tipos de amenazas. Se utilizan entre muchas otras posibilidades, para generar estadísticas de explotación, detectar patrones de ataque y detectar nuevo tipo de malware. Este último punto resulta especialmente interesante. En la mayoría de las ocasiones, el malware aprovecha vulnerabilidades para descargar archivos (virus) desde un servidor. Para intentar evadir a los antivirus y pasar lo más desapercibido posible, este archivo descargado es muy variable, y pueden aparecer nuevas versiones cada pocas horas.

Un honeypot puede simular el aprovechamiento de esa vulnerabilidad y permite que se descargue ese nuevo archivo. De esta forma un honeypot puede resultar un excelente recolector de nuevas versiones de virus y malware en general de forma sistemática y automatizada. Al igual que los de alta interacción, estos sistemas deben estar muy bien protegidos para que no se vuelvan en contra del administrador del honeypot. Un atacante, ya sea de forma automática o manual,) podría llegar de alguna forma al "meta-sistema" que aloja el servicio simulado, y atacarlo.

Alta interacción	Baja interacción
Servicios reales, sistemas operativos o aplicaciones	Emulan servicios, vulnerabilidades, etc.
El riesgo que corren es mayor	El riesgo que corren es menor
Capturan menos información, pero más valiosa	Capturan mucha información. Dependen de su sistema de clasificación y análisis

Fuente: (Even, 2000)

Un honeypot puede ser diseñado como un servidor en vez de como una computadora es decir, como un sistema (honeypot que hace de servidor) que espera ser contactado con un cliente (computadora).

Desde el momento en el que uno de los objetivos de un honeypot es recabar información sobre ataques, surgió un concepto de un honeypot "cliente" que no espere a recibir ataques sino que los genere activamente. Se les llama honeymonkeys. Desde hace varios años, el vector de ataque más utilizado en Internet es el navegador. Las medidas de seguridad han aumentado y cada vez resulta más difícil aprovechar vulnerabilidades en clientes –programas- de correo electrónico, que venía siendo el vector de ataque más usado. El uso popular de firewalls también hizo que cada vez fuese más complicado para atacantes aprovechar vulnerabilidades en el propio sistema operativo. Por tanto, con el traslado a la web de los servicios (foros, chats, etc.), el navegador se convirtió en el objetivo favorito de los atacantes. Con solo visitar una web, se intenta aprovechar todo tipo de vulnerabilidades en el navegador para ejecutar código en el cliente e infectarlo. Tras este concepto surgen los honeymonkeys. Su función principal, al igual que la de los honeypots, es igualmente detectar nuevos tipos de ataques y fórmulas de infección e igual que los honeypots, están formados por un módulo de "exploración" y un módulo de recogida de datos.

Sin embargo en el caso de los honeymonkeys, la exploración se hace activamente a través de navegadores. El honeymonkey funciona como un sistema automático de navegación que visita toda clase de páginas web con el fin de que alguna de ellas intente aprovechar vulnerabilidades en el navegador. Poseen una naturaleza mucho más activa que el honeypot, en el sentido en el que "patrullan" la red como si fueran un usuario visitando enlaces compulsivamente. Fue Microsoft quien los bautizó. "Monkey" (mono, en inglés) hace alusión a los saltos y el dinamismo del tipo de acción que realizan. Con este método, al igual que los honeypots, se pueden encontrar nuevos exploits, gusanos, etc., siempre que se analice y procese convenientemente toda la información recogida.

2.21.4. Honeypots y honeynets

El propósito de las honeynet es, al igual que el honeypot, investigar el uso de las técnicas y herramientas que hacen los atacantes en Internet.

Se diferencia básicamente de un honeypot en que no supone una sola máquina, sino múltiples sistemas y aplicaciones que emulan otras tantas, imitan vulnerabilidades o servicios conocidos o crean entornos "jaula" donde es posible una mejor observación y análisis de los ataques. Los requerimientos básicos e imprescindibles para construir una honeynet son dos, los llamados: Data Control (control de datos) y Data Capture (captura de datos).

Data Control

Suponen la contención controlada de la información y las conexiones. Lidiar con atacantes siempre supone un riesgo que hay que reducir al máximo, por lo que es preciso asegurarse que, una vez comprometido el honeypot, no se comprometerán sistemas legítimos. El reto consiste en mantener un absoluto control del flujo de datos sin que el atacante lo note. No se puede cerrar un sistema por completo para evitar el tráfico innecesario. Una vez comprometido el sistema, el atacante intentará realizar distintos tipos de conexiones para continuar su ataque, probablemente necesite bajar programas por FTP, correo o conexiones SSH. Si no se le permite esta flexibilidad de acciones, además de levantar sus sospechas, no se podrán estudiar otros pasos más importantes que valdría la pena analizar. En los primeros intentos de los investigadores de poner en marcha proyectos de honeynet, no se permitieron ningún tipo de conexiones salientes para evitar ser plataforma de nuevos ataques. Pero sólo les llevaba a los atacantes unos minutos ver que algo andaba mal, y abandonar el intento de ataque. Los resultados así eran muy pobres. De esto se deduce una disyuntiva en la que reside el arte de construir una honeynet útil.

Data Capture

Es el rastreo y almacenamiento de la información que se persigue, esto es, los logs (registros de datos) de sus actos, y que serán analizados a posteriori. Se debe capturar tanta información como sea posible aislada del tráfico legal, evitando la posibilidad de que el atacante sepa que se le están recogiendo sus acciones. Lo más importante para conseguir esto es evitar el almacenamiento de resultados localmente en el propio honeypot, puesto que pueden ser potencialmente detectados y borrados con la lógica intención de no dejar huellas del ataque.

La información debe ser almacenada remotamente y en capas. No se puede limitar al registro de una simple capa de información, sino tomarla de la mayor variedad posible de recursos.

Herramientas virtuales

Las herramientas para construir un honeypot o una honeynet son muy variadas, pero el método más frecuente es el uso de máquinas físicas o virtuales para construir el honeypot. Dado el potencial peligro del uso de honeypots, y a su propia naturaleza, el uso de herramientas virtuales resulta muy conveniente y es ampliamente aceptado. Las ventajas de un sistema virtual sobre uno físico son evidentes:

- **Permiten ser restauradas en cuestión de minutos en caso de accidente**, desastre o compromiso: la mayoría de sistemas virtuales permiten almacenar un estado "ideal" y volver a él en cualquier momento de manera mucho más rápida que si hubiese que restaurar un sistema físico y devolverlo a un estado anterior.

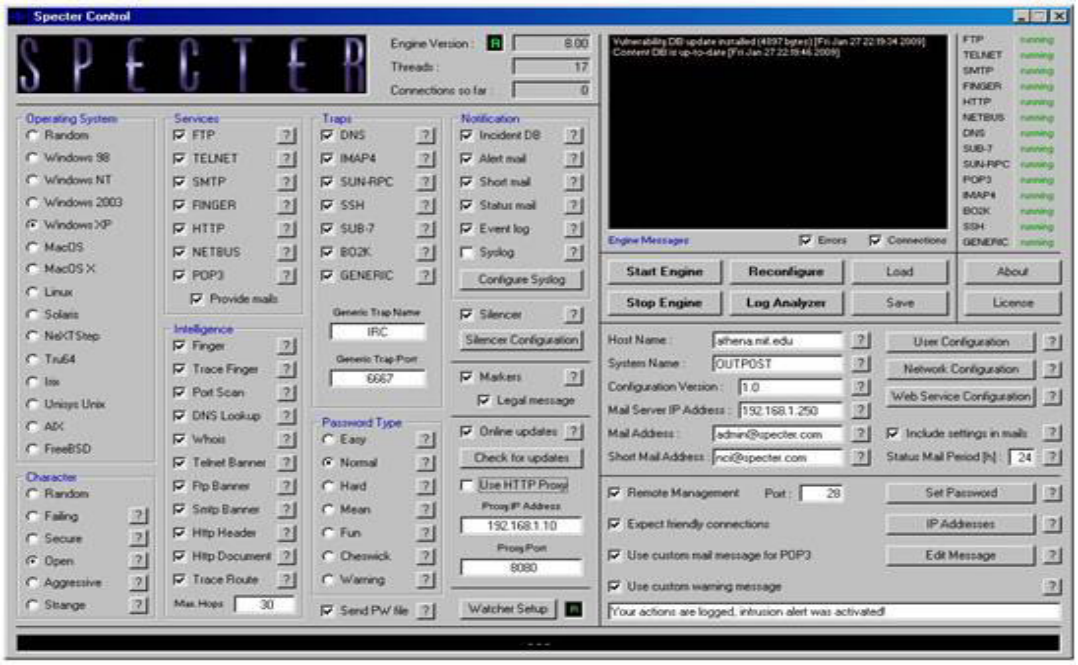
- **Permiten ser portadas a diferentes máquinas físicas que la alojan**: los sistemas virtuales, por definición, se ejecutan por igual en cualquier máquina física, que emulan el entorno necesario a través de un programa para poder reproducir el sistema virtual.

- **Permiten ahorrar costes**: una misma máquina física puede alojar un número indeterminado de máquinas virtuales, tantas como le permitan sus recursos, y con tantos sistemas operativos como se desee.

2.21.5. Ejemplos de Honeypots

Existen pocas herramientas comerciales que cubran este mercado de honeypots, sin embargo, en el mundo del código libre, se ofrecen muchas utilidades que pueden servir como honeypots, tanto a empresas como a particulares. Uno de los Honeypots comerciales más conocidos es **Specter**.

Gráfico 2. 25: Specter



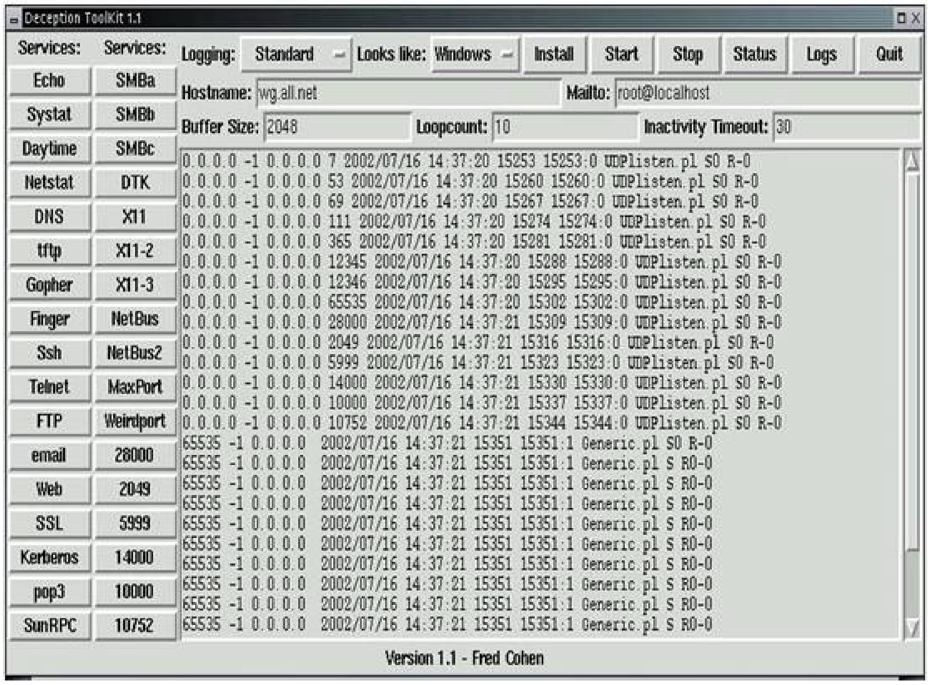
Fuente: (Even, 2000)

Es capaz de simular hasta 14 sistemas operativos diferentes, y funciona bajo Windows. Su principal atractivo es su facilidad de uso.

KFSensor también es un honeypot comercial que actúa como honeypot e IDS para sistemas Windows. En el mundo del código abierto, se pueden encontrar muchos ejemplos de Honeypots que cubren todos los aspectos de estas herramientas:

Bubblegum Proxypot, Jackpot, BackOfficer Friendly, Bigeye. HoneyWeb, Deception Toolkit, LaBrea Tarpit, Honeyd, Sendmail SPAM Trap, etc.

Gráfico 2. 26: KFSensor



Fuente: (Even, 2000)

2.22. Las PYMES en el Ecuador

Según (Dolph Lundgreen, 2011), se conoce como PYMES al conjunto de pequeñas y medianas empresas que de acuerdo a su volumen de ventas, capital social, cantidad de trabajadores, y su nivel de producción o activos presentan características propias de este tipo de entidades económicas. Por lo general en nuestro país las pequeñas y medianas empresas que se han formado realizan diferentes tipos de actividades económicas entre las que destacamos las siguientes:

- Comercio al por mayor y al por menor.
- Agricultura, silvicultura y pesca.
- Industrias manufactureras.
- Construcción.
- Transporte, almacenamiento, y comunicaciones.
- Bienes inmuebles y servicios prestados a las empresas.
- Servicios comunales, sociales y personales.

2.23. Fortalezas de las PYMES

- Representan el 95% de las unidades productivas
- Generan el 60% del empleo
- Participan del 50% de la producción
- Amplio potencial redistributivo
- Capacidad de generación de empleo
- Amplia capacidad de adaptación
- Flexibilidad frente a los cambios
- Estructuras empresariales horizontales

2.24. Debilidades de las PYMES

- Insuficiente y/o inadecuada tecnología y maquinaria para la fabricación de productos
- Insuficiente capacitación del talento humano.
- Insuficiencia de financiamiento.
- Insuficiente cantidad productiva
- Inadecuación de la maquinaria y procedimientos propios a las normativas de calidad exigidas en otros países.

2.25. Características de las PYMES

Las PYMES en nuestro país se encuentran en particular en la producción de bienes y servicios, siendo la base del desarrollo social tanto produciendo, demandando y comprando productos o añadiendo valor agregado, por lo que se constituyen en un actor fundamental en la generación de riqueza y empleo. Al ser una empresa en desarrollo sus principales características consisten:

• Requieren de exigencias técnicas, de calidad y legales.
• Escasa capacidad de negociación.
• Inexistencia de estrategias globales de internacionalización

• Débiles encadenamientos productivos-materias primas
• Costos elevados por desperdicio de materia prima.
• Insuficiente cantidad productiva para exportar.
• Inadecuación de la maquinaria y procedimientos propios a las normativas de calidad exigidas en empresas grandes.

De las pequeñas industrias en el comercio internacional, se deben emprender en acciones conjuntas entre gobierno, gremios y empresarios, encaminados a:

- Intensivos programas de capacitación en administración.
- Las empresas deben entrar en un mejoramiento continuo de la calidad, para lo cual se requiere el apoyo del gobierno y la asistencia técnica de la cooperación internacional
- Negociar con proveedores confiables, que aseguren la entrega de materias primas e insumos de calidad y a tiempo
- Cumplir con las normas de producción limpia, esto da seguridad en el acceso a mercados internacionales
- Las empresas deben trabajar con una producción especializada, esto asegura eficiencia, calidad y competitividad.
- Deben procurar la asociación con otras empresas afines y complementarias, esto dará más certeza en el cumplimiento de las cantidades, normas y tiempos de exportación.
- Utilizar canales de distribución reconocidos y confiables
- Utilizar el internet para las ventas
- Hacer un trabajo de calidad y a tiempo

2.26. Mejores prácticas para configurar las reglas del Firewall en una PYME

La primera regla que debe tener clara todo director de IT, es que el firewall no es suficiente para proteger la red y la información de su compañía, y aunque este no tiene que ver directamente con la configuración del firewall hace parte de las buenas prácticas de seguridad informática que se debe tener en toda empresa.

Entrando en el tema, configurar el firewall no es tarea fácil y se debe pasar por un proceso de configuración exhaustivo de las reglas y/o políticas para garantizar la máxima seguridad que este nos pueda ofrecer. No utilice “any” o cualquiera en una regla específica del firewall. Esto le evitará problemas de seguridad y de control de flujo de tráfico. Por ejemplo una regla que diga cualquier servicio de cualquier fuente a cualquier destino. Una regla como esta no permite que el firewall actúe como un punto de control fuerte para el tráfico que fluye a través de él. Así que debe asegurarse que el firewall solo permita el tráfico necesario. Añadir comentarios a las reglas del firewall. Con el tiempo las reglas del firewall crecen y se va haciendo difícil recordar porque una regla se creó y que se supone que esta hace. Un comentario ayudaría mucho en este caso. Los comentarios evitan crear reglas redundantes. Además si otro administrador inicia sesión estos comentarios le ayudaran a entender rápidamente porque esta regla se creó y cuál es su propósito. Planificar la adición de nuevas reglas. Cada vez que desee implementar una política nueva en el firewall o si necesita cambiar una política de firewall actual, un paso importante en el proceso de implementación de los cambios en las reglas del firewall es programar el cambio en el momento oportuno. Cambios no programados de reglas puede afectar el flujo de tráfico. Si las conexiones se caen, algunos servicios se vea afectado y si estos servicios son esenciales, entonces las operaciones del negocio, se verán demasiado afectadas. De hecho, para minimizar el impacto de un cambio de reglas de firewall, la programación de cambios en las reglas no se debe realizar durante horas laborales o días pico de trabajo. Mantener actualizado el firmware del Firewall en caso que se tenga una solución de Firewall basado en Hardware. La forma como los fabricantes corrigen los problemas de configuración, de seguridad o mejoras en la misma es a través de parches o nuevas versiones de firmware. Por eso es muy importante mantener instalado la última versión de Firmware liberada por el fabricante. Firewall de inspección profunda de paquetes de estado y de red, Firewall de aplicaciones, Firewall basado en capa 8 o identidad de usuario: permite crear políticas basadas en los usuarios y no por IP. Proporciona una seguridad integrada combinando seguridad, conectividad y productividad ya que integra funciones y módulos VPN, IPS, Anti-Virus & Anti-Spyware, Anti-Spam, Web Filtering, Administración Ancho de Banda, Administración de Enlaces Múltiples entre otros.

CAPÍTULO 3

3. Alternativas de solución

3.1. Características y precios en diferentes tipos de Firewalls

En el presente capítulo se realizó una comparación de las variadas soluciones que se ofrecen en el mercado en lo que respecta a dispositivos de seguridad o Firewalls, siendo estos en su mayoría de dos tipos: Hardware y software.

3.2. Firewalls por hardware

Los firewalls basados en hardware protegen todos los equipos de la red. Un firewall basado en hardware es más fácil de mantener y gestionar que los firewalls de software individuales. La solución ideal para las pequeñas empresas es un firewall de hardware integrado en una solución de seguridad completa. Además de un firewall, la solución debe incluir soporte para red privada virtual (VPN), antivirus, antispam, antispyware, filtrado de contenido y otras tecnologías de seguridad. A continuación se presentan algunos de los firewalls por hardware que reúnen varias de las características previamente indicadas y son de las marcas más conocidas en el mercado de la seguridad informática.

Gráfico 3. 1: D-Link NetDefend DFL-260 VPN/Firewall

Fuente: (D-Link, 2012)

EL firewall DFL-260E NetDefend Unified Threat Management (UTM) de D-Link es una potente solución de seguridad y está diseñado para proteger a las pequeñas y medianas oficinas de una amplia gama de amenazas que se encuentran en la red. El firewall proporciona enrutamiento remoto, Network Address Translation (NAT), Red Privada Virtual (VPN), seguridad de red proactiva, Sistema de prevención de intrusos (IPS), Filtrado de contenidos Web (WCF), Protección Anti-Virus (AV), balanceo de carga de tráfico y Administración de ancho de banda, todo en un compacto chasis de escritorio, de fácil integración a la red existente. Las principales características técnicas de este dispositivo son las siguientes, cuenta con 5 puertos LAN Ethernet 10/100/1000 Mbps, un puerto Wan 10/100/1000 Base-T Autosensing. Un puerto para DMZ 10/100/1000 Base-T Autosensing (configurable) un puerto de Consola, y 2 Puertos USB 2.0. El precio referencial del equipo es de $246.49 dólares, y el administrador del mismo debe tener experiencia en configuración de equipos de esta marca y de los conceptos de seguridad de red, ya que se debe administrar de forma correcta todas las funcionalidades que ofrece el dispositivo para que la inversión no sea en vano.

Gráfico 3. 2: CISCO861-K9 861 Ethernet Security Router

Fuente: (Cisco, 2012)

Los Routers Cisco Small Business ofrecen una amplia gama de características y funciones, entre ellas, conectividad VPN segura que utiliza cifrado para ofrecer a los trabajadores remotos acceso a su red y datos. Puertos de switch Fast Ethernet, funciones de redundancia de enlace y equilibrio de carga para mejorar el tiempo de actividad y el rendimiento del sistema mediante el uso de múltiples puertos para conectarse a Internet.

Gráfico 3. 3: D-Link NetDefend DFL-260 VPN/Firewall - 6 Port

Fuente: (Cisco, 2012)

El RV220W provee una fuerte seguridad con un firewall de Inspección de Estado de paquetes (SPI), además de seguridad avanzada para el wireless para mantener los activos de la empresa a salvo. El firewall ofrece activación y escaneo de puertos, prevención de denegación de servicios (Denial of Service (DoS), y configuración de DMZ basado en software. El precio de mercado de este equipo se encuentra en $294.57, es configurable mediante web browser conectándose directamente al equipo, y presenta varias soluciones de propietario si se desea ampliar la cobertura de seguridad.

Gráfico 3. 4: Cisco-Linksys BEFSX41 EtherFast Cable/DSL Firewall Router

Fuente: (D-Link, 2012)

El Router Firewall Linksys EtherFastCable/DSL con conmutador de 4 puertos es la solución perfecta para conectar un pequeño grupo de PCs a una conexión de banda ancha de Internet o una red Ethernet 10/100. El router puede ser configurado para limitar el acceso de los usuarios internos a Internet basado en URLs o por periodos de tiempo predefinidos además de filtración de URLs. Para una mayor protección contra intrusos de Internet, el router dispone de un firewall avanzado con Stateful Packet Inspection (Inspección de estado de paquetes).

Puede utilizar el Router para crear segmentos de VPN, con lo que se puede conectar de forma segura desde su hogar a la oficina o desde cualquier lugar en que se encuentre. EL router provee un puerto dedicado para la DMZ y actúa como el único gateway reconocido para la red LAN.

Gráfico 3. 5: Cisco-Linksys BEFSX41 EtherFast Cable/DSL Firewall Router

Fuente: (D-Link, 2012)

Cuenta con un precio referencial en el mercado de $289.99, este router necesita de configuración mediante web browser con pasas que se indica en un manual que viene adjunto al equipo.

3.3. Firewalls por software

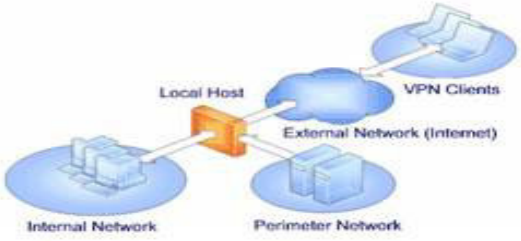
Los firewalls por software constituyen una buena opción común cuando se requiere proteger una sola computadora. Sus ventajas incluyen el no necesitar hardware adicional, ni cables extra, pero por otro lado, el costo de los mismos puede ser una desventaja, ya que se debe pagar diferentes tipos de licencias (anuales, por número de dispositivos). A más de que es necesario tener una copia del software por dispositivo, dicha copia debe ser específica para cada sistema operativo, por ejemplo un firewall de software para Windows no funcionará en Mac y viceversa. Se revisa una serie de opciones en el mercado para este tipo de seguridad que puede ser implementada en una oficina pequeña para asegurar la misma.

3.3.1. Firewall de Windows

Si utiliza Windows 7 o XP Service Pack 2 (SP2), ya dispone de un firewall integrado y activado de forma predeterminada como parte de los beneficios que le brindan estos Sistemas Operativos. Si utiliza XP pero no lo tiene actualizado con el SP2, sigue teniendo acceso al Firewall pero tendrá que activarlo.

3.3.2. ISA Server

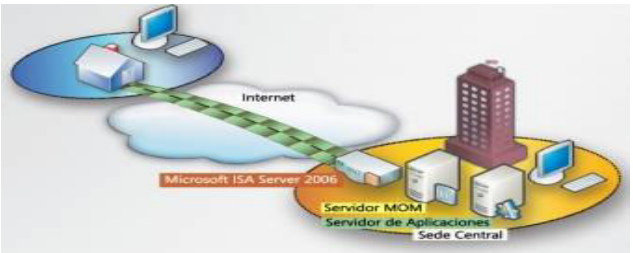
Gráfico 3. 6: Interfaz de Isa server 2004



Fuente: (D-Link, 2012)

ISA Server es un Gateway integrado de seguridad perimetral que protege su entorno de IT frente a amenazas basadas en Internet y permite a los usuarios un acceso remoto rápido y seguro a las aplicaciones y los datos. La publicación segura de aplicaciones con ISA Server 2006 permite el acceso de forma protegida a aplicaciones Exchange, SharePoint y otros servidores de aplicaciones Web. Los usuarios remotos pueden acceder a ellas desde fuera de la red corporativa con el mismo nivel de seguridad y privacidad que desde dentro. ISA Server 2006 puede utilizarse como Gateway para redes de oficinas. Permite conectar y proteger los enlaces entre oficinas remotas y departamentos centrales y optimizar el uso del ancho de banda

Gráfico 3. 7: Protección del acceso a la web ISA server

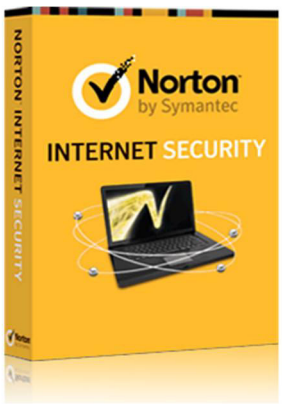


Fuente: (D-Link, 2012)

La protección del acceso a la Web que proporciona ISA Server 2006 aumenta la seguridad a los entornos de IT corporativos frente a amenazas internas y externas basadas en Internet. La versión ISA Server 2006 Ed. Estándar tiene un precio comercial de 1.499 USD por procesador, lo que le deja en muchas ocasiones fuera del alcance de empresas pequeñas, las cuales no pueden permitirse esos costos de inversión en seguridad muy elevados.

3.3.3. Norton Internet Security

Gráfico 3. 8: Norton Internet Security



Fuente: Norton By Symantec

El paquete de protección Norton Internet Security de la empresa Symantec ofrece diferentes opciones de protección que también varían en precio, de igual forma se diferencia por la cantidad de usuarios que pueden acceder a una licencia, desde 1 solo computador por un precio de $ 62.99 y valido por un año, hasta una instalación completa de 10 computadoras por un año a un precio de $179.99. Consta de:

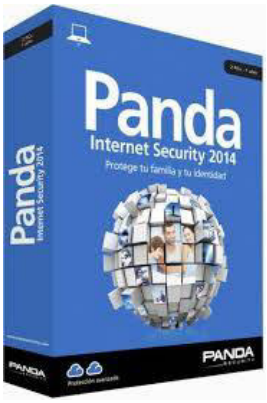
- Control parental
- Protección en Facebook
- Protección anti Phishing
- Protección de navegación
- Firewall inteligente de dos vías

- Antivirus
- Antispyware
- Antispam
- Monitoreo de red

A parte de los costos y la administración que representa este tipo de solución, hay que tomar en cuenta que los mismos sólo son válidos por un año, al finalizar dicho periodo se debe volver a contratarlos por un valor igual o superior al ya abonado, para un nuevo año de protección.

3.3.4. Panda Internet Security 2014

Gráfico 3. 9: Panda Internet Security 2014



Fuente: Panda Internet Security

La tecnología cloud de Panda Security ofrece un nuevo modelo de protección basado en la comunidad de usuarios en la que todo el mundo contribuye. La computadora está siempre protegida, siempre actualizada. La computadora no limita la capacidad de detección ya que la gran base de datos de detección de virus está alojada en Internet (la nube):

- Protección contra Spyware (software espía), Phishing (fraude online), Rootkits (Técnicas de ocultación) y Troyanos Bancarios.
- Protección en tiempo real

- Nuevo Filtro Web que permite navegar por Internet de forma segura
- Protección completa contra ataques de virus conocidos y desconocidos.
- Vacuna tus unidades USB contra infecciones.
- Firewall. Bloquea intrusos y piratas, incluso de la red inalámbrica.

El paquete de protección de Panda, tiene validez por un año, con licencias para 5 computadoras, por un precio de $249.75 el cual si se requiere renovar, de sebe volver a pagar por el mismo.

3.3.5. Avira Professional Security 2014

Avira Professional Security proporciona una sólida protección para Windows y Unix. Realiza actualizaciones automáticas y la administración centralizada, con lo que se libera de carga de trabajo al personal del departamento de informática:

- System Scanner detecta rápidamente los últimos virus, gusanos y troyanos conocidos.
- AntiPhishing impide que datos confidenciales se expongan a estafas.
- AntiAd/Spyware detiene los programas espías y las molestas ventanas emergentes online.
- Rootkit Protection atrapa el malware oculto que pasa desapercibido para los antivirus convencionales y AntiBot detiene los procesos automatizados activados por piratas informáticos.
- Real-Time Protection protege contra el malware entre escaneos del sistema.
- FireWall le da el control del tráfico online y crea una barrera en su red para mantener a raya a los piratas informáticos.
- Web Protection evita que los usuarios lleguen a páginas web con códigos maliciosos.
- Mail Protection impide que los emails infectados con malware dañen las computadoras o se extiendan entre los contactos de los usuarios.

Por un paquete de licencias para 5 PCs por un año de validez, el precio comercial es de $240,20

Gráfico 3. 10: Avira Professional Security

Fuente: Avira Professional Security

3.3.6. Zone Alarm Free Firewall 2014

Gráfico 3. 11: Zone Alarm Free Firewall

Fuente: Zone Alarm Free Firewall

Firewall bidireccional (tráfico entrante y saliente), protege activamente contra los ataques entrantes y salientes, al tiempo que le mantiene invisible para los piratas informáticos. DefenseNet™ obtiene en tiempo real los datos sobre amenazas procedentes de una comunidad de millones de usuarios del firewall ZoneAlarm, para ofrecer una respuesta rápida a las amenazas recién descubiertas y proteger su PC de los ataques más recientes.

El firewall avanzado de ZoneAlarm vigila las acciones que tienen lugar en su propia computadora para detectar y detener incluso los ataques más sofisticados y recientes capaces de eludir el control de los paquetes de seguridad y antivirus tradicionales.

Además cuenta con funcionalidades extra tales como:

- Protección avanzada contra descargas
- Protección contra la suplantación de la identidad
- Servicios de protección de la identidad
- Copia de seguridad online
- Privacy & Security Toolbar
- Facilidad de uso
- Funcionamiento automático.

Tiene una versión gratuita con opciones básicas a más de las versiones de pago que incluyen soporte técnico.

3.4. Comparación en las alternativas de solución

En resumen, un firewall por hardware es un dispositivo físico que contiene algún tipo de software propietario en su mayoría, el cual necesita de un nivel mínimo de configuración, ya que viene previamente configurados para ser conectados a la red, viene con su propia conexión de energía y puertos de red, en los cuales se conectan los dispositivos que comprenden el centro de cómputo al cual se debe proteger de las amenazas de la red. Un firewall de software da un nivel inferior de protección para mantener la computadora a salvo de los hackers y otros intrusos no deseados, porque el software a pesar de ser más fácil de personalizar para los usuarios principiantes, pueden dejar agujeros en la seguridad de la computadora de los cuales el usuario no tenga conocimiento, ya que al ser un software con reglas y configuraciones predefinidas, no permite la libertad de otros sistemas abiertos, para una configuración personalizada.

Otro punto a tomar en cuenta con los firewalls por software, es que se deben instalar de forma independiente en cada computadora conectada a la red, lo cual dificulta la administración de los mismos, por cuanto se debe tener claro las diferencias entre cada tipo y marca de computadora, así como la compatibilidad entre versiones de software, parches y aplicaciones en cada una de ellas. Una alternativa de protección que puede integrar la seguridad de un firewall por hardware y la libertad de personalización del mismo, y con una inversión inicial mínima, por cuanto se utilizará software libre, mismo que se puede descargar en varias versiones según la necesidad del usuario, además de incluir una de las herramientas significativas en lo que respecta a seguridad de tráfico y navegación en internet, IPTABLES. Iptables es un conjunto de herramientas (comandos) que le permiten al usuario enviar mensajes al kernel del Sistema Operativo Linux. El kernel tiene todo el manejo de paquetes TCP/IP metido dentro de él, no es algo aparte como lo es en otros sistemas operativos, por lo tanto todos los paquetes que van destinados a un Linux o lo atraviesan son manejados por el mismo kernel.

Entonces, iptables es una forma de indicarle al kernel algunas cosas que debe hacer con cada paquete, esto se hace en base a las características de un paquete en particular. Los paquetes de red tienen muchas características, algunas pueden ser los valores que tienen en sus encabezados (a donde se dirigen, de donde vienen, números de puertos, etc.), otra puede ser el contenido de dicho paquete (la parte de datos), y existen otras características que no tienen que ver con un paquete en particular sino con una sumatoria de ellos. La idea es lograr identificar un paquete y hacer algo con el mismo. Además, de que el hecho de esta alternativa va a controlar el tráfico de toda la red local de computadoras y servidores, independientemente de cuantas computadoras estén conectadas al mismo y de los diferentes que puedan ser entre sí, ya que las reglas que se establecen en IPTABLES se aplican a los paquetes que entran y salen de nuestra red.

CAPÍTULO 4

4. Características que debe tener un buen Firewall

En base a lo expuesto en el capítulo anterior, las características que debe tener el Firewall ideal son:

- Debe ser compatible con el estándar del modelo OSI
- Debe tener la capacidad de reconocer el tipo de tráfico que viene desde la red externa hacia la interna
- Debe tener la capacidad de reconocer el tipo de tráfico que viene desde la red interna hacia la externa

4.1. Presentando Firebuilder

Firebuilder es una herramienta que funciona sobre cualquier plataforma Linux con Kernel 2.6, tiene bajo costo y fue desarrollado para generar el script de Firewall que va proteger las redes corporativas de las pequeñas y medianas empresas. La versión 1.0 diseñada por David Silva, tiene como valor agregado, un potente escaneador de vulnerabilidades para identificar las computadoras que están actualmente vulnerables en su red. Ideal para que el administrador de infraestructura sepa que acción tomar ante esto a fin de tener una red totalmente protegida.

Gráfico 4. 1: Firebuilder

Fuente: Autor

Logra detener todas las conexiones externas no autorizadas que requerirán transmitir tráfico hacia la red local de una compañía, protegiendo tanto computadoras como servidores.

Logra detener todas las conexiones internas no autorizadas que requerirán transmitir tráfico hacia fuera de la red local de una compañía, evitando exponer las computadoras como servidores.

Logra detener la saturación del canal de datos dentro de una red local de una compañía por motivos de vulnerabilidades.

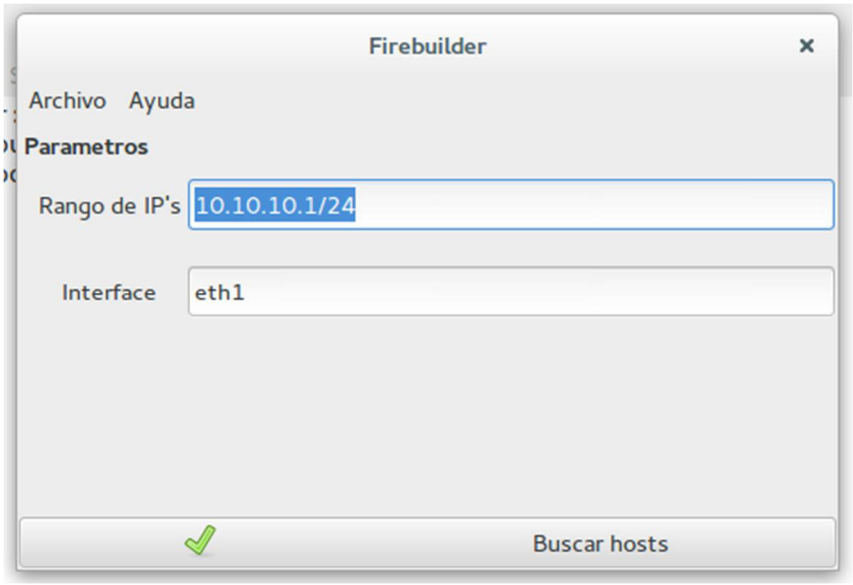
FireBuilder hará las mismas funciones que un vigilante de tránsito: autorizar quien pasa de un lado a otro y viceversa. Tomando en cuenta esta analogía, FireBuilder validará quienes son las conexiones que deberá pasar su tráfico desde fuera hacia la red local de una compañía y también controlará el tráfico que va desde la red local interna hacia fuera de esta (nube).

Para su funcionamiento se ingresan los parámetros de escaneo los cuales son:

Rango IP´s *Puede ser una IP o un rango específico*

Interface *Tarjeta o tarjetas de red en donde va empezar la búsqueda o rastreo*

Gráfico 4. 2: Funcionamiento Firebuilder

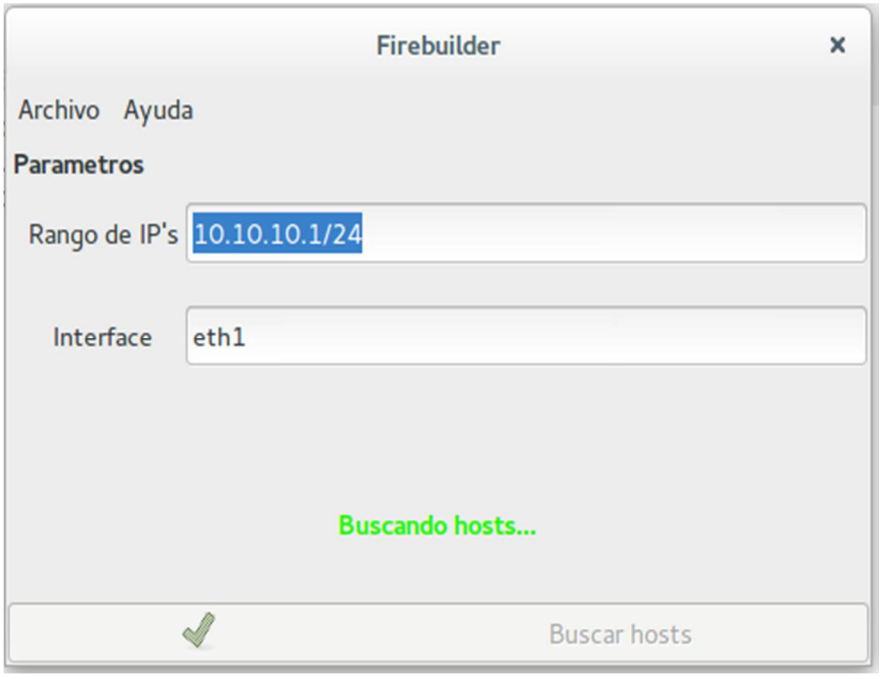


Fuente: Autor

Gráfico 4. 3: Funcionamiento 2 Firebuilder

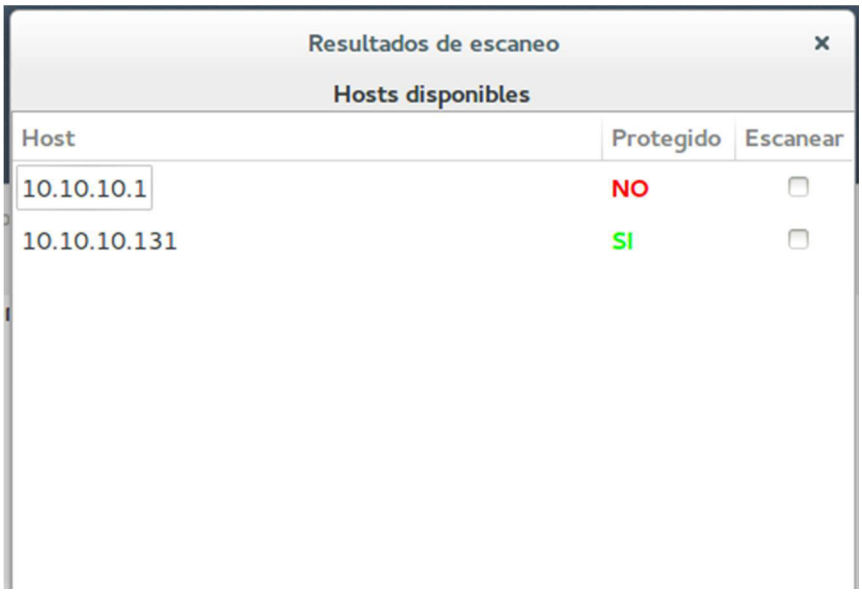
Fuente: Autor

Gráfico 4. 4: Funcionamiento 3 Firebuilder



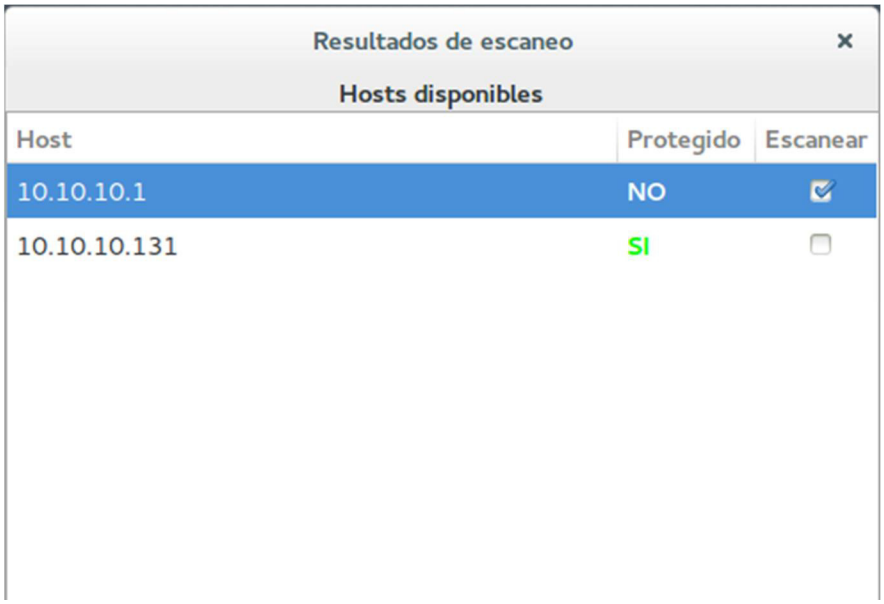
Fuente: Autor

Gráfico 4. 5: Funcionamiento 4 Firebuilder

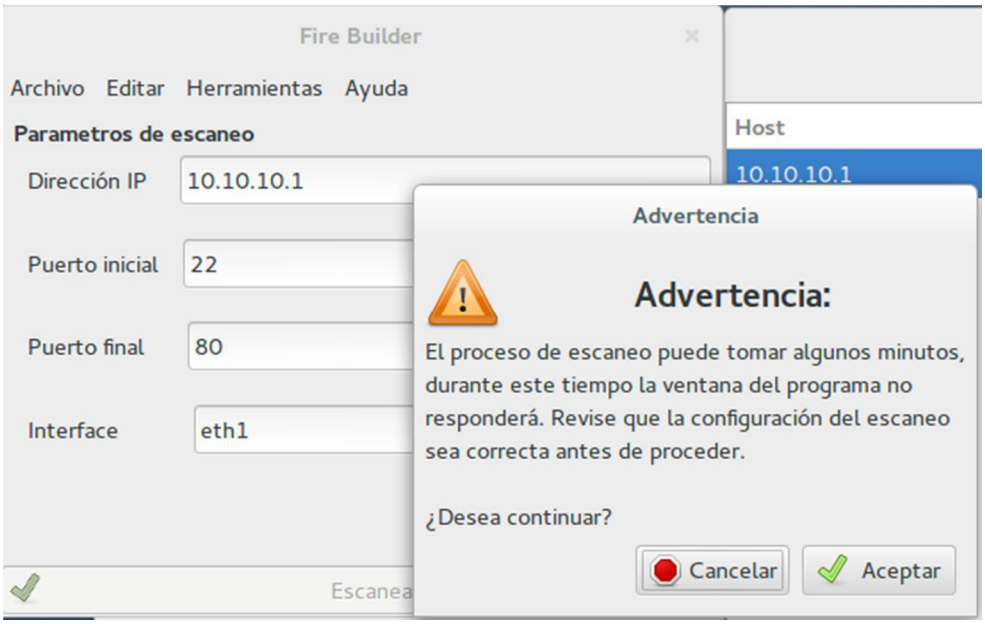


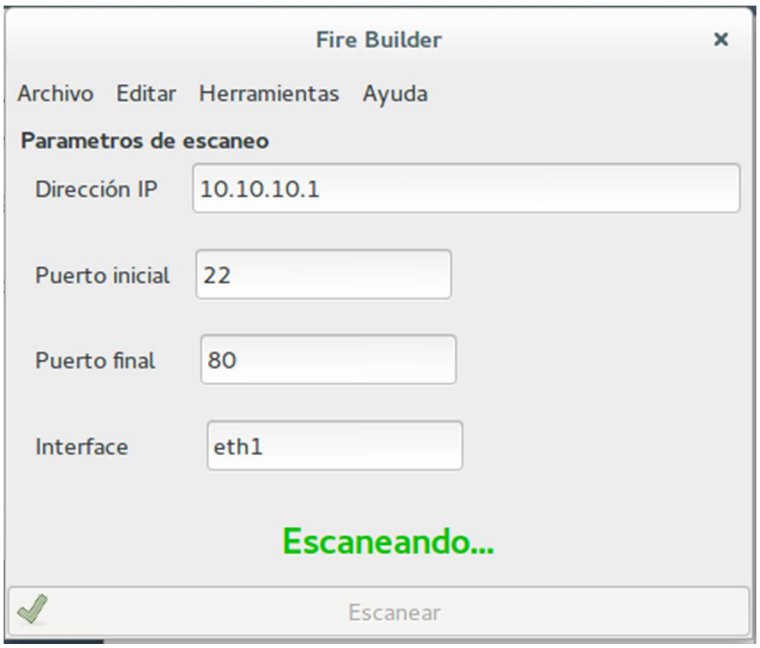
Fuente: Autor

Gráfico 4. 6: Funcionamiento 5 Firebuilder



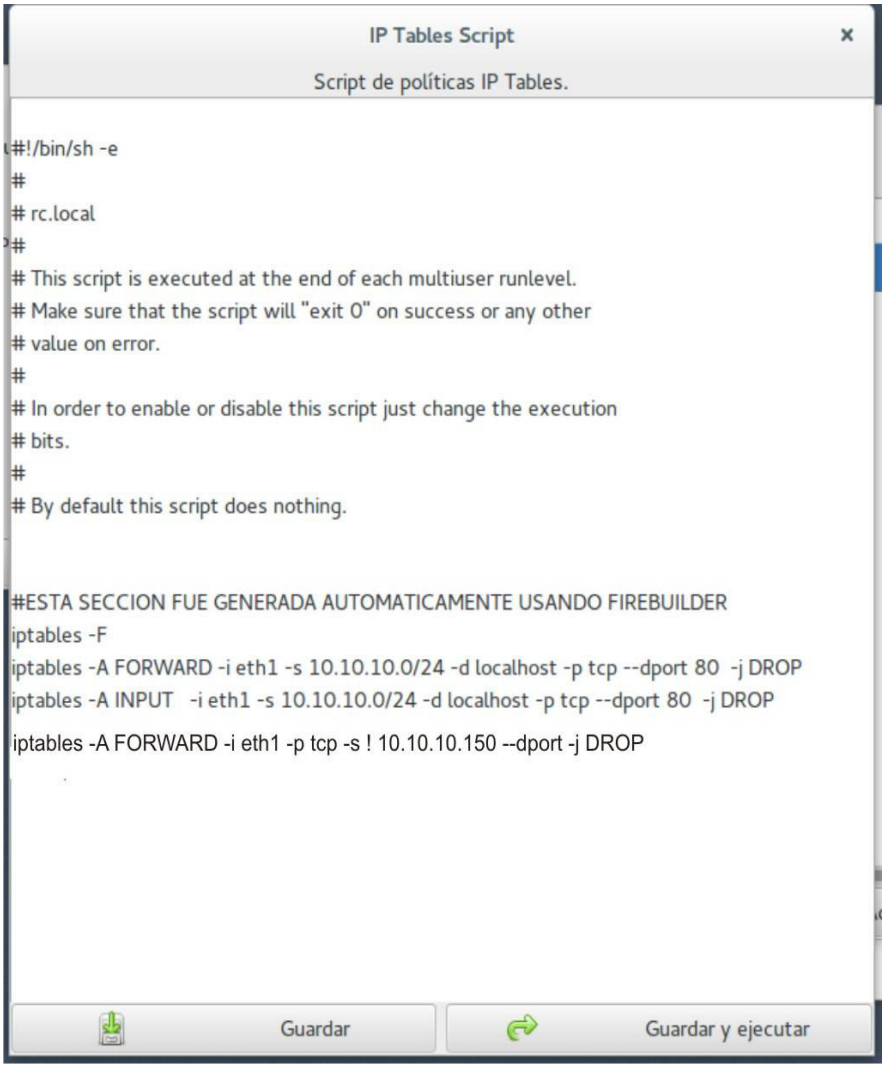

Fire Builder
Archivo Editar Herramientas Ayuda
Parametros de escaneo
Dirección IP
10.10.10.1
Puerto inicial
22
Puerto final
80
Interface
eth1
Escanear

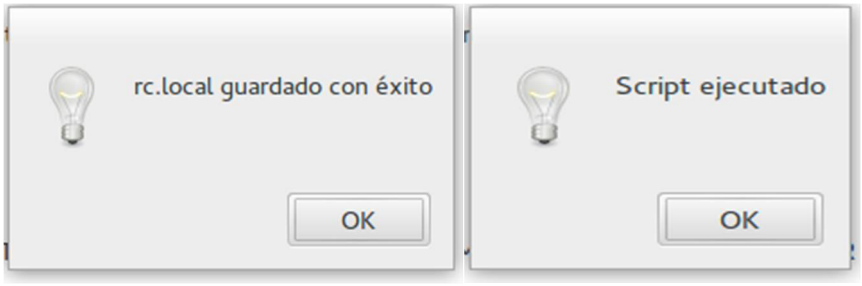

Fire Builder
Archivo Editar Herramientas Ayuda
Parametros de escaneo
Dirección IP
10.10.10.1
Puerto inicial
22
Puerto final
80
Interface
eth1
Escanea
Host
10.10.10.1
Advertencia
Advertencia:
El proceso de escaneo puede tomar algunos minutos, durante este tiempo la ventana del programa no responderá. Revise que la configuración del escaneo sea correcta antes de proceder.
¿Desea continuar?
Cancelar
Aceptar


Fire Builder
Archivo Editar Herramientas Ayuda
Parametros de escaneo
Dirección IP 10.10.10.1
Puerto inicial 22
Puerto final 80
Interface eth1
Escaneando...
Escanear


Resultados de escaneo
Resultados del escaneo
Host Protocolo Puerto Estado Servicio Bloquear Excluir
10.10.10.1 tcp 80 open http 10.10.10.150

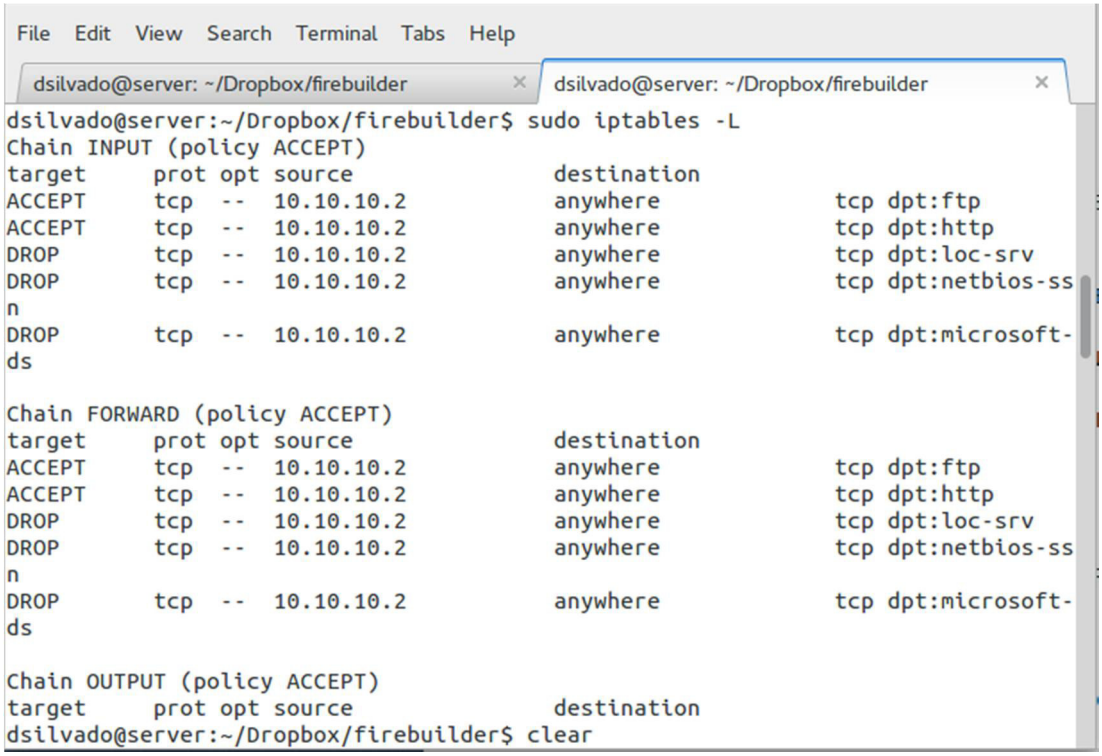

Advertencia
Advertencia:
Para volver a escanear otros hosts presione Aceptar, caso contrario presiona Cancelar para generar y ejecutar el Script de Firewall
¿Desea continuar?
Cancelar
Aceptar





Fuente: Autor

Observación: Para comprobar si el script de firewall se ha generado con éxito, se debe ejecutar el comando sudo iptables –L



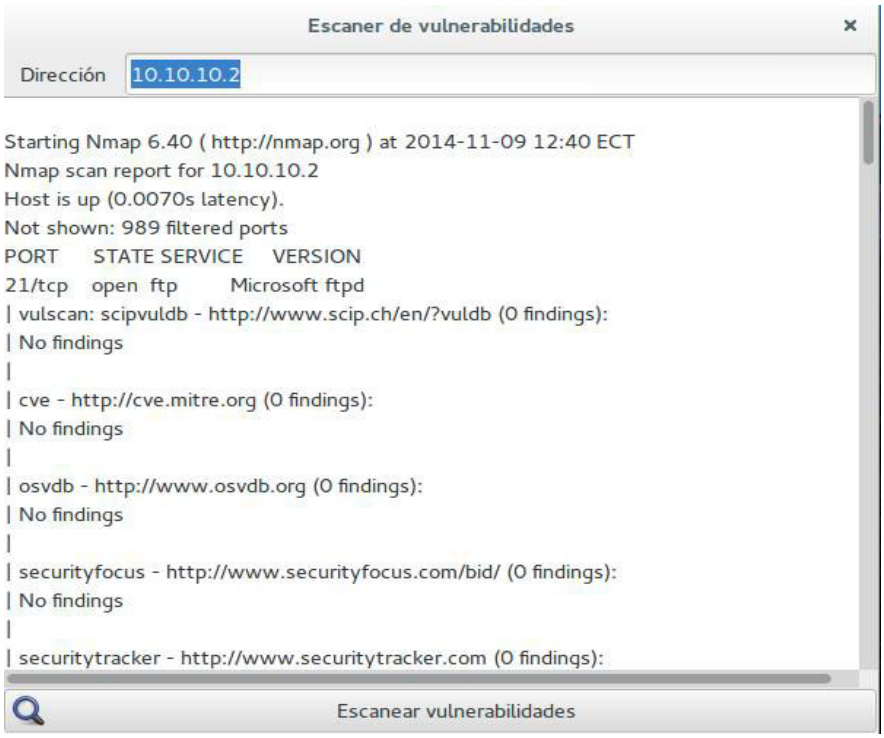
Otras cosas que pude hacer firebuilder

Gráfico 4. 7: Funcionamiento 6 Firebuilder

Fuente: Autor

Como valor agregado, Firebuilder cuenta con un escaneo de vulnerabilidades en los puertos del host de destino (Vulscan)

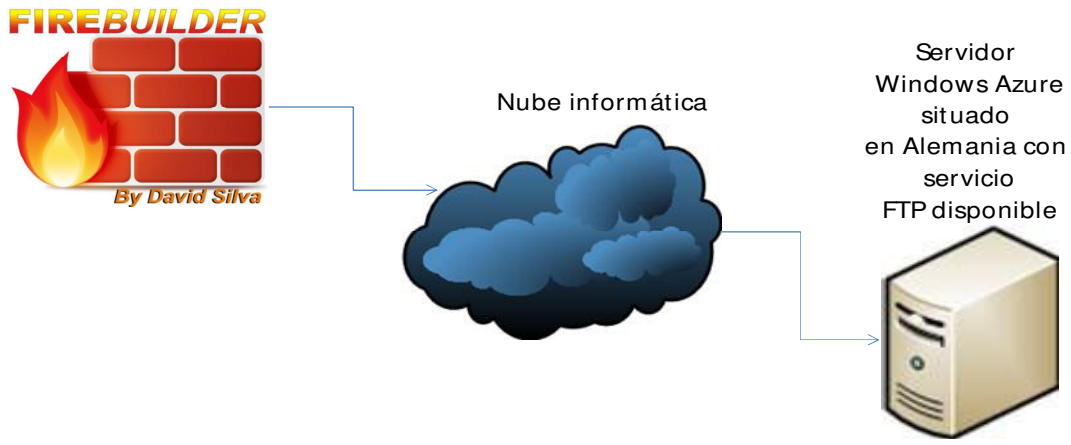
Gráfico 4. 8: Funcionamiento 7 Firebuilder



Fuente: Autor

4.2. Respaldo de la información

Gráfico 4. 9: Respaldo de la información

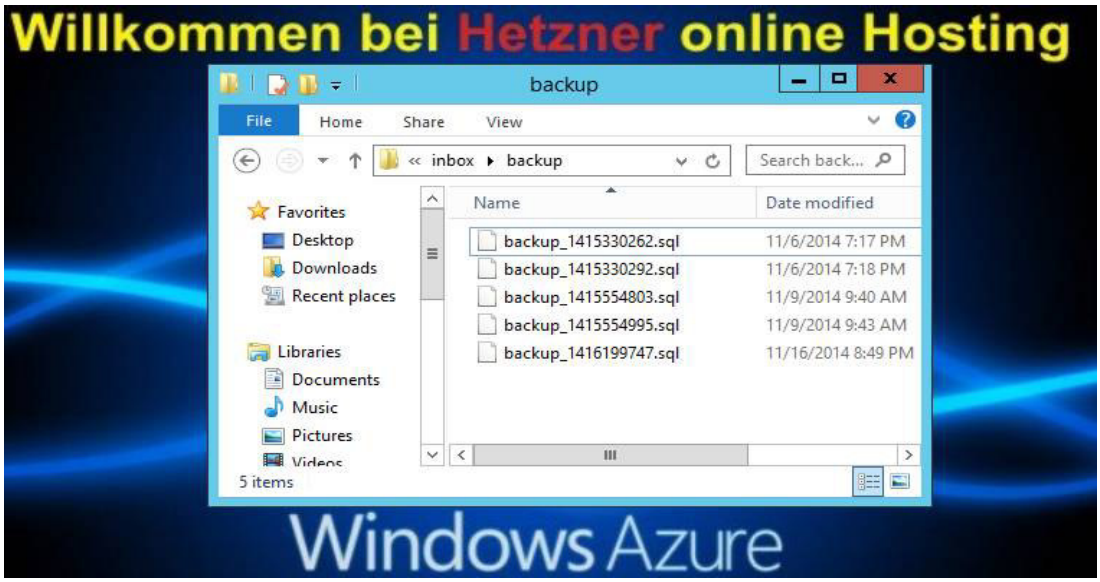


Fuente: Autor

FireBuilder genera un archivo el cual contiene los parámetros de configuración realizado + el resultado del escaneo de los puertos para la generación

Del script de Firewall + resultado de escaneo de vulnerabilidades de puertos de un servidor. Esta información es enviada mediante un archivo de extensión SQL y se lo transfiere hacia un servidor con Sistema Operativo Windows Azure mediante el servicio FTP (File Transfer Protocol)

Gráfico 4. 10: Respaldo de la información 2



Fuente: Autor

4.3. Competencia Firebuilder

Tal como se mencionó anteriormente, existen en el mercado informático dos tipos de Firewall:

A nivel de hardware, el cual lo implementan en corporaciones y su costo promedio es de $3000 (incluido licencia de uso).

A nivel de software, viene normalmente con los sistemas operativos pero con funciones básicas. Recordemos que hay sistemas operativos propietarios que también tienen costo (Ejemplo: Microsoft Windows).

FireBuilder es un generador del Script Firewall el cual tendrá la capacidad de un firewall a nivel de Hardware, fácil de descargar y sobre todo de implementar. Esto hace que no tenga competencia.

4.4. Costo de implementación

FileBuilder incurre en un bajo costo de implementación debido a que está desarrollado con componentes de licenciamiento GNU/GPL (Software Libre)

Gráfico 4. 11: Costo de implementación

Fuente: Autor

Su versión 1.0 está disponible desde el sitio Web: http://www.firebuilder-ec.com

Para el costo de la inversión, incurrirá en un solo pago inicial de $640 y un pago anual de $30. Con esto, mantendrá su red protegida de ataques cibernéticos desde dentro y fuera de su red local de la empresa. No necesita mantenimiento y como valor agregado, tiene una visita técnica sin costo para entrenamiento.

Detalle	Duración / inversión en horas	Valor	Subtotal
Desarrollo de pantallas y codificación	12	$ 20,00	$ 240,00
Hardware: CPU con 4Gb de RAM, 512Gb de disco duro y 2 tarjetas de red	1	$ 350,00	$ 350,00
Hosting para backup en Alemania	1	$ 30,00	$ 30,00
Servicio técnico para ejecución	1	$ 20,00	$ 20,00
		Total	**$ 640,00**
1. Luego de este pago anualmente se tendrá que pagar $30 por Hosting 2. No necesita mantenimiento 3. Incluye 1 visita técnica de 1 hora sin costo para entrenamiento			

4.5. Desarrollo Firebuilder

FireBuilder fue desarrollado con aplicaciones y componentes de software libre para lo cual se detalla:

- Iptables (herramienta firewall que trabaja sobre el Kernel 2.6 del sistema operativo Linux)
- Nmap (Restreador de puertos para validar su estado actual)
- GTK+ (Conjunto de bibliotecas multiplataformas para desarrollar interfaces gráficas de usuario)
- Glade + GTK builder (Desarrollador de pantallas)
- Python (Lenguaje de programación)
- MySQL (Versión libre de base de datos SQL)
- Sphinx (Generador de documentación Pyhton)

4.5.1. Requerimientos mínimos

Existe en el mercado informático una gama de servidores desde básicos hasta complejos como los HP Proliant, Superdomes o IBM Blades. Lo bueno es que Firebuilder puede trabajar con una computadora básica que cumpla con las siguientes características mínimas:

- Computadora Clone con procesador X86
- 25Mb de espacio en disco
- 512Mb de Memoria RAM
- 2 tarjetas de red (de preferencia del mismo fabricante)
- Sistema operativo Linux (de cualquier versión o distribución)
- Tener los permisos administrativos de la cuenta root

4.5.2. Instrucciones de implementación Firebuilder

Se debe instalar el CPU con el sistema operativo Linux instalado (de cualquier distribución), configurar sus interfaces eth0 y eth1.

El cable de red que viene de la red WAN se la debe conectar en la interface eth0 y el cable que viene de la red local en la interface eth1. Las direcciones IP que se van a instalar en las interfaces de red deben coincidir con el rango que está utilizando sus respectivas redes WAN y LAN. Este CPU va ser el que va estar en medio de ambas redes y controlará el tráfico tanto de entrada como de salida y viceversa.

Descargar el archivo firebuilder.rar desde el sitio del autor, desempaquetar el archivo y ejecutar: firebuilder.sh

El resto se configurará automáticamente.

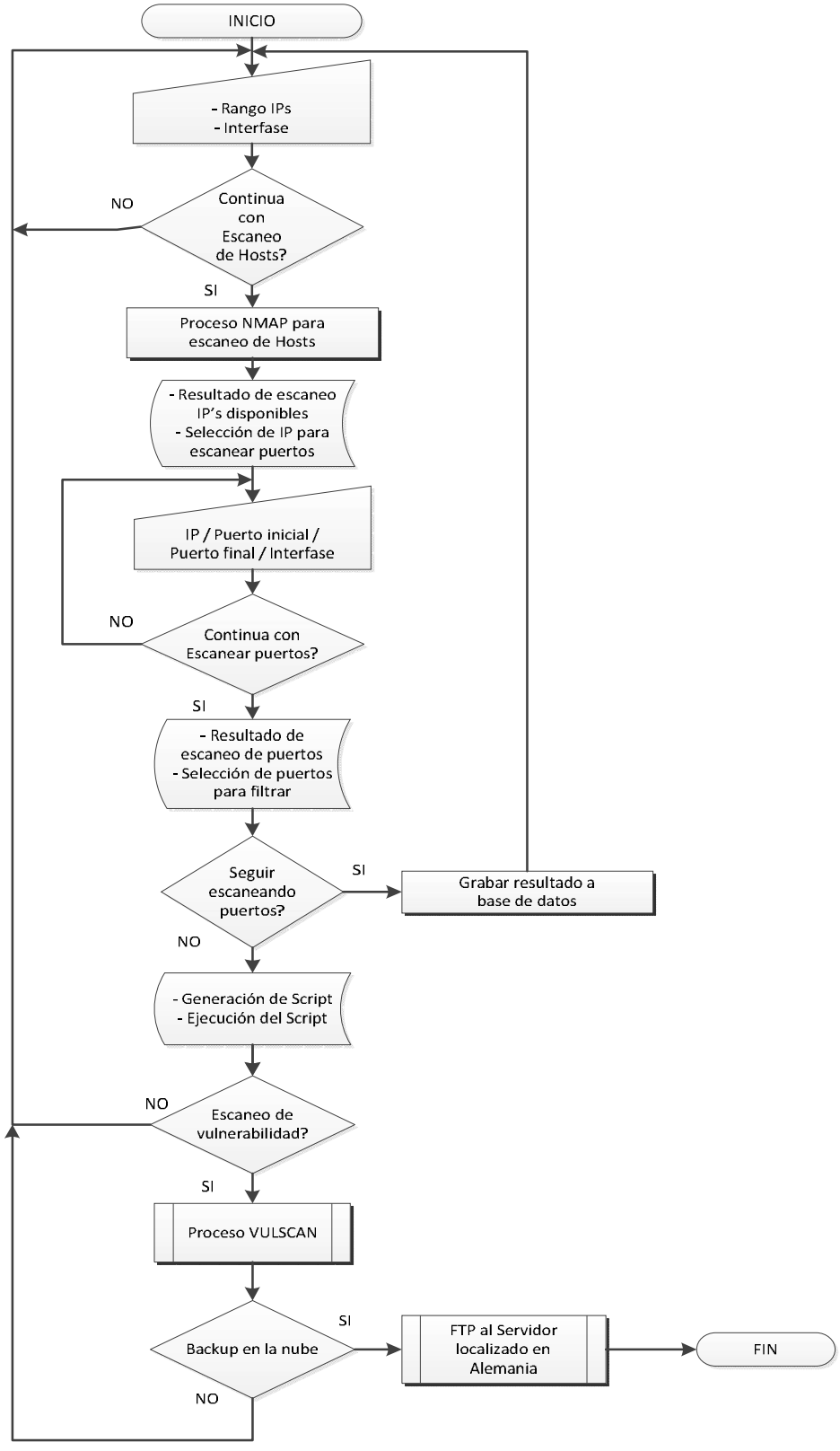
Tenemos como ejemplo:

Gráfico 4. 12: Implementación firebuilder

Fuente: Autor

4.6. Entendiendo el proceso Firebuilder

Gráfico 4. 13: Entendiendo el proceso Firebuilder

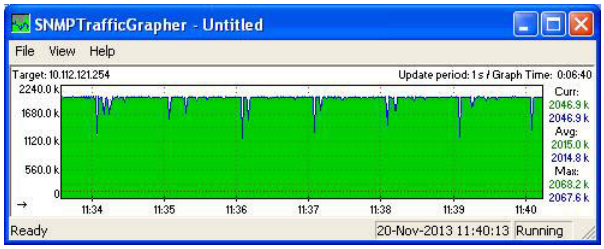


Fuente: Autor

4.7. Caso de éxito

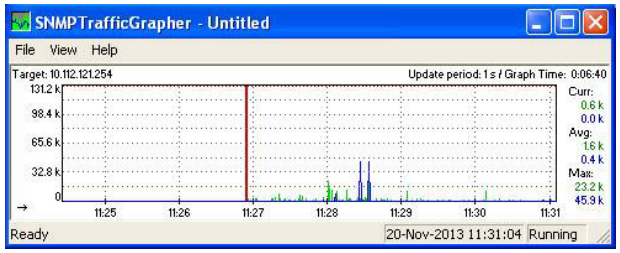
FireBuilder fue implementado en la compañía SERVITECH[8] (servicio técnico de la compañía de servicios informáticos Cartimex) ante el requerimiento de la empresa del no querer verse involucrada en temas de vulnerabilidades internas o saturación de red. La empresa SERVITECH, es una mediana empresa creada con el fin de proporcionar servicios de venta de suministros de computación a diferentes empresas de forma independiente, ubicado en el norte de la ciudad de Guayaquil, cuenta con una infraestructura que consta de varias computadoras y servidores con conexión a Internet, por medio del proveedor Ecuador Telecom, cuenta con una única oficina matriz, en la cual se encuentra toda la infraestructura informática utilizada para el funcionamiento de la empresa. Antes de esto, el consumo del canal de datos subía hasta en un 80% al punto de llegar a la saturación total. Luego de su implementación, se hizo la medición validando que el canal se mantiene en un 12% de utilización y como resultado un bloqueo total de conexiones no autorizadas las cuales amenazaban con la seguridad de la red local. Medición fue realizada por el aplicativo SMTP Traffic Grapher.

Gráfico 4. 14: Antes de implementación Firebuilder



Fuente: Autor

Gráfico 4. 15: Después de implementación Firebuilder



Fuente: Autor

[8] Servitech está gerenciado por el Ing. Admond Mondavi Sobbi

CAPÍTULO 5

5.1. Conclusiones y Recomendaciones

5.1.1. Conclusiones

El sistema planteado reúne los requisitos de ser un sistema de uso simple desde el punto de vista del usuario, pero reúne la complejidad suficiente dentro de sus procesos internos, para ser una solución lo suficientemente segura.

Todos los procesos adicionales que se realizarán dentro del sistema de seguridad, son totalmente transparentes para el usuario, es decir el usuario no se percatará que dentro del sistema se realizan verificaciones adicionales de seguridad. Iptables es una herramienta flexible, aun cuando su entendimiento conlleva algo de complejidad; una vez comprendida su filosofía de funcionamiento, la herramienta se vuelve muy versátil y permite realizar implementaciones modulares.

Se dispondrá de mayor seguridad al emplear herramientas de software abierto y de libre acceso, ya que al haber más personas que lo utilizan y lo revisan, ayudarán a encontrar posibles errores o fallas de seguridad que podrían afectar el funcionamiento del sistema.

La filosofía de distribución de código libre, busca que el conocimiento del desarrollo de las aplicaciones, sea libre para quien quiera emplearlas, de esta forma si el usuario llegase a encontrar algún problema de seguridad o de otra índole en la aplicación, podrá editar directamente el código fuente o informar al desarrollador de la aplicación, para que en una futura versión se corrija el problema.

5.1.2. Recomendaciones

En el presente proyecto, al estar basado en una infraestructura que emplea como elementos principales servidores, es recomendable establecer políticas de respaldo de la información, de los equipos más críticos, así como también de una política de respaldo continua de la información de base de datos y configuraciones de los equipos.

Tener un servidor en la nube (otra localidad u otro país) es una buena práctica, ya que en el momento de un desastre real a nivel de infraestructura, los tiempos de respuestas jugarán un papel muy importante. De igual forma, de presentarse una actualización de las configuraciones realizadas en la presente solución, se debe guardar un respaldo de las versiones previas, no sobreescribirlas o eliminarlas a fin de preservar un control de los cambios realizados al firewall. El lugar de instalación de los equipos debe poseer una apropiada instalación eléctrica, con puesta a tierra, así como también debe poseer elementos de respaldo como fuentes de alimentación ininterrumpida de por lo menos una hora de abastecimiento de energía.

Es importante para la óptima operación del sistema, que el administrador mantenga un estricto cumplimiento de las políticas de seguridad descritas e implementadas como parte de la solución presentada, ya que el incumplimiento de alguna de las mismas puede dar lugar a un incremento de la vulnerabilidad del sistema y este puede ser un blanco fácil para que usuarios mal intencionados hagan un mal uso de este.

Como sugerencia final se recomienda la revisión periódica de las vulnerabilidades que tienen las computadoras dentro de su red corporativa debido a la criticidad de sus funciones. Una red limpia de virus y segura es un dolor de cabeza menos y sobre todo mucha optimización monetaria.

Bibliografía

Autores de Libros:

- Castells, M. (1997). *La era de la información. Economía, sociedad y cultura (Vol I: La sociedad red).* Madrid: Alianza Editorial.
- Gaspel, C. (2009). *Enciclopedia práctica de la Pequeña y Mediana Empresa PYME 2da edicion.* Bogota: Oceano - Centrum.
- Groth, D., & Skandier, T. (2005). *Guía del estudio de redes, (4ª edición).* Mexico: Sybex, Inc.
- Malf Kirch, G. D. (2000). *Guía de Administración de Redes con Linux.* United States: O'Reilly & Associates.
- Philippe Atelin, J. D. (2013). *TCP/IP y protocolos de internet.* Chile: ENI ediciones.
- Postel, J. (1981). *NCP/TCP transition plan.* Colombia: McGraw Hill.
- Zimerman, H. (1980). *OSI Reference Model – The ISO Model of Architecture for Open Systems Interconnection.* United States Of America: IEEE Transactions on Communications.

Medios electrónicos:

- SearchSecurity.com. (10 de Julio de 2006). What is a distributed denial of service attack. Mexico.
- AJPDSOFT. (26 de Noviembre de 2010). Obtenido de AJPDSOFT.COM.
- Diego Lopez. (10 de Abril de 2012). Transparencias de Redes. Mexico, Monnterrey, Mexico.
- D-Link. (25 de Septiembre de 2012). Firewall. Monterrey, Mexico.
- Dragon JAR. (23 de Noviembre de 2013). Manizaldes Caldas, Colombia.

- Foundation, F. S. (1 de Enero de 2012). La ofensiva del software libre. Estados Unidos: GNU project.
- Gartner, Inc. (26 de Junio de 2008). Gartner Says Cloud Computing Will Be As Influential As E-business. United States.
- Informatica Hoy. (21 de Junio de 2010). Tipos de firewall.
- Ing. Pello Xabier Altadill Izura . (2011). *IPTABLES manual practico.* Argentina: UPV - EHU.
- IPV6. (17 de Febrero de 2014). *NIC Mexico*. Recuperado el 17 de Febrero de 2014, de http://www.ipv6.mx/index.php/informacion/fundamentos/ipv6
- Masadelante.com. (1 de Enero de 2012). ¿Qué es un sistema operativo? Madrid, España.
- Universidad del Oriente . (1 de Febrero de 2008). *Normas y políticas de seguridad informatica.* San Miguel - El Salvador: Attribution Non-commercial Share Alike.
- Alberto Espinoza. (3 de Abril de 2013). *Alegsa*. Recuperado el 17 de Junio de 2013, de Alegsa: http://www.alegsa.com.ar/Dic/spammer.php
- Aurelio Silvetty. (12 de Diciembre de 2012). *REDES TIPOS TOPOLOGIA*. Recuperado el 27 de Octubre de 2013, de REDES TIPOS TOPOLOGIA: http://redestipostopologias.blogspot.com/2009/03/topologia-de-redes.html
- Dolph Lundgreen. (25 de Febrero de 2011). *PYME*. Recuperado el 29 de Septiembre de 2013, de PYME: http://definicion.de/pyme/#ixzz2vc26cjBd
- Elvis Proaño. (3 de Marzo de 2012). *Informatica Hoy*. Recuperado el 16 de Noviembre de 2013, de Informatica Hoy: http://www.informatica-hoy.com.ar/aprender-informatica/Que-es-un-Cracker.php
- Felipe Sosa. (4 de Enero de 2013). *Un Jubilado*. Recuperado el 5 de Febrero de 2013, de Un Jubilado: http://www.unjubilado.info/2010/03/28/piratas-informaticos/

- Fredd Judge. (1 de Junio de 2012). *Mysql*. Recuperado el 4 de Julio de 2013, de Mysql: www.mysql.com

- Janeth Paez. (1 de Diciembre de 2012). *Belarmino*. Recuperado el 30 de Noviembre de 2013, de Belarmino: http://belarmino.galeon.com/

- Javier Heinz. (24 de Julio de 2013). *ISIS*. Recuperado el 13 de Octubre de 2013, de ISIS: http://wwwisis.ufg.edu.sv/wwwisis/documentos/TE/004.67-G633d/004.67-G633d-CAPITULO%20II.pdf

- Jose Benavides. (29 de Abril de 2013). *Zator*. Recuperado el 30 de Mayo de 2013, de Zator : http://www.zator.com/Hardware/H12_2.htm

- Juan Mendez. (16 de Septiembre de 2013). *Consulta Yahoo*. Recuperado el 18 de Octubre de 2013, de Consulta Yahoo: http://es.answers.yahoo.com/question/index?qid=20080511053743AASWpUU

- Oliva Marcillo. (16 de Junio de 2012). *UFG*. Recuperado el 30 de Mayo de 2013, de UFG: http://wwwisis.ufg.edu.sv/wwwisis/documentos/TE/004.67-G633d/004.67-G633d-CAPITULO%20II.pdf

- Sergio Mendez. (27 de Junio de 2012). *ORACLE*. Recuperado el 15 de Agosto de 2013, de ORACLE: https://docs.oracle.com/cd/E19957-01/820-2981/ipv6-overview-10/

ANEXOS

Anexo1: Instalación de Centos 6.5 (64 Bits)

Centos 6.5 es una distribución de Linux [9]que sigue los pasos comenzados por Red Hat. Esta distribución alcanzó un gran éxito con su versión 7 de Red Hat, llegando a la distribución 9 cuando se crearon dos líneas, una llamada Fedora Core, que con su versión inicial seguía la idea de distribución completamente abierta, y otra que seguía teniendo el nombre Red Hat, que se comercializaba siendo apoyada por un gran equipo de soporte. Muchos han catalogado a Centos de distribución light, de distribución no válida para cosas serias, si bien es cierto que es una distribución tan válida como el resto para estar instalada en grandes servidores. Centos 6.5 tiene un gran aliciente frente a otras distribuciones, y es que anaconda, su programa de instalación (que comparte con otras distribuciones), es muy sencillo de enseñar. Por una razón o por otra, Centos, en sus diferentes versiones, es una distribución muy utilizada en computadoras personales, si bien es cierto que también se ve en grandes instalaciones, aunque en estas ubicaciones da un poco lo mismo muchas veces la distribución instalada, ya que apenas se instalan aplicaciones además del propio kernel, aplicaciones que realmente diferencian unas distribuciones de otras. La distribución Centos 6.5 se la puede adquirir, vía web, ftp, jigdo o torrent de la página web de Centos (http://centosproject.org), si bien existen muchos mirrors (computadoras espejo, que guardan copias del original), de los que se puede bajar el software. Al entrar en el directorio que hace referencia a la arquitectura de nuestra computadora (i386, ppc o x86 64) podremos encontrar las imágenes de instalación en el directorio ISO que encontraremos. Hoy en día es una buena opción bajarnos la imagen del DVD al tener una velocidad de acceso a Internet aceptable, que nos permite bajar estas imágenes en un tiempo razonable y que, dado que la mayoría de los dispositivos tienen lector de DVD, es la opción de instalación más rápida. Una vez hayamos grabado la imagen con nuestro programa favorito, arrancamos la computadora con el DVD cargado en el lector, asegurándonos de que nuestra BIOS está configurada para arrancar desde DVD. En la primera pantalla que aparece, se muestran las diferentes opciones que se nos ofrece desde este DVD, como realizar un test de memoria, arrancar desde el disco duro, recuperar un sistema dañado, actualizar un sistema que ya tenga Centos (una versión anterior) o instalar Centos.

[9] Centos es la versión económica que no utiliza licencia como RHEL (Red Hat Enterprise Linux)

Estas dos últimas opciones se pueden realizar mediante un menú gráfico, sencillo de utilizar, o en modo no gráfico. Se elige, por tanto, la opción Install or upgrade an existing system, es decir, instalar o actualizar un sistema existente.

Gráfico Anexo 1. 1: Menú de arranque de Centos 6.5

Fuente: Centos

Nada más arrancar el DVD, se muestra la posibilidad de comprobar si el DVD está bien grabado. Si bien es una acción que emplea cierto tiempo, y puede resultar engorrosa, es muy útil la primera vez que utilicemos el DVD, pues si se ha bajado bien el archivo ISO, o si no se ha grabado correctamente, se puede realizar una instalación errónea y el tiempo que perdido será mucho mayor que el que supone esta comprobación. Se pulsa por lo tanto Entrar cuando esté resaltado el botón OK, teniendo en cuenta que para pasar de un botón a otro tenemos que pulsar el tabulador.

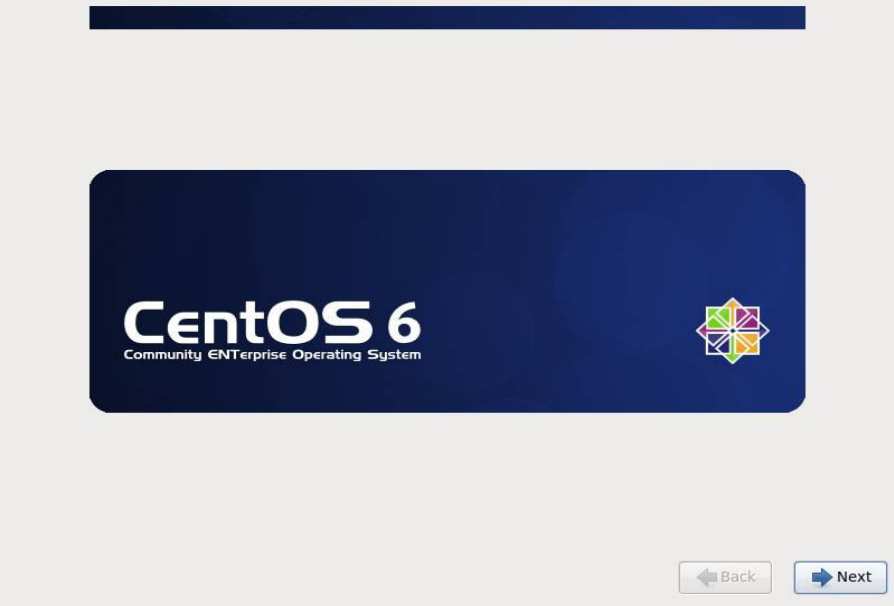
Gráfico Anexo 1. 2: Elegiremos comprobar el DVD de instalación

Fuente: Centos

La comprobación tardará un cierto tiempo, y se muestra una barra de progreso para poder tener una idea de cuánto falta.

Una vez finalizada la comprobación, se despliega un mensaje que indicará si la imagen es satisfactoria o no. En el caso de que no fuera correcta, en el caso de que el test hubiera dado como resultado que la imagen está mal, se debe bajar y/o grabar el DVD de nuevo para poder partir de un DVD correcto. Una vez aceptado este mensaje obtenido, se ofrecerá la posibilidad de comprobar más soportes. En nuestro caso no se requiere realizar ninguna comprobación adicional, por lo tanto presionar Continue y aceptar pulsando Entrar. En este momento entonces se encuentra la pantalla de bienvenida de Centos 6.5, en la que ya se puede utilizar el ratón sin mayor problema, y donde se pulsa Next para poder seguir con la instalación. A continuación se despliegan diferentes idiomas para seguir la instalación, es decir, elegiremos Spanish (Español) como el idioma que debe utilizar el programa de instalación para dirigirse a nosotros, haciendo por tanto más sencilla la instalación de Centos 6.5.

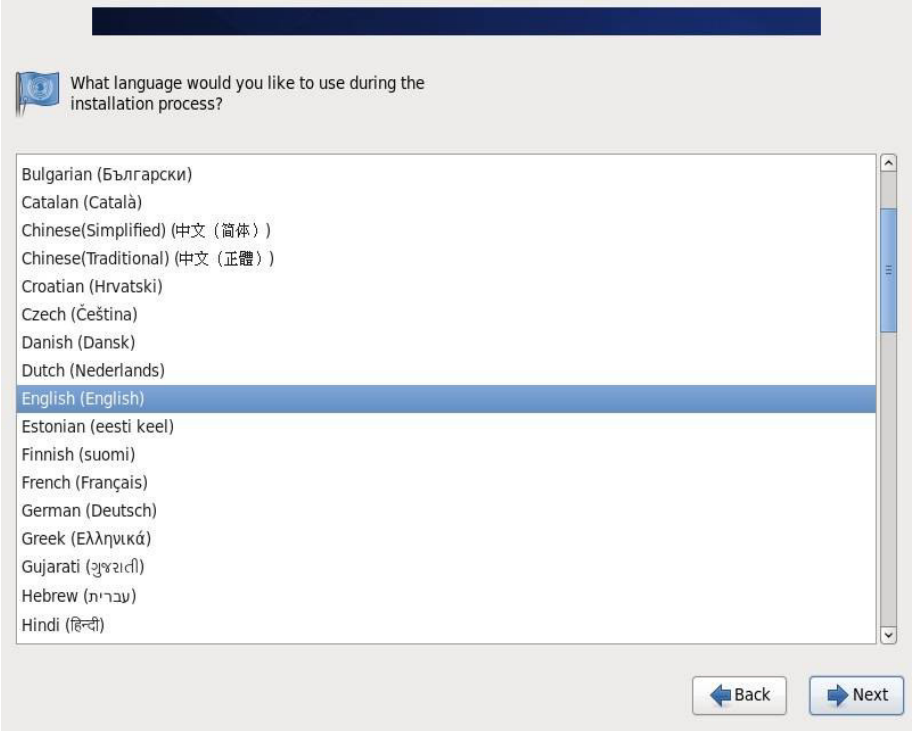
Gráfico Anexo 1. 3: Pantalla de bienvenida de Centos 6.5.



Fuente: Centos

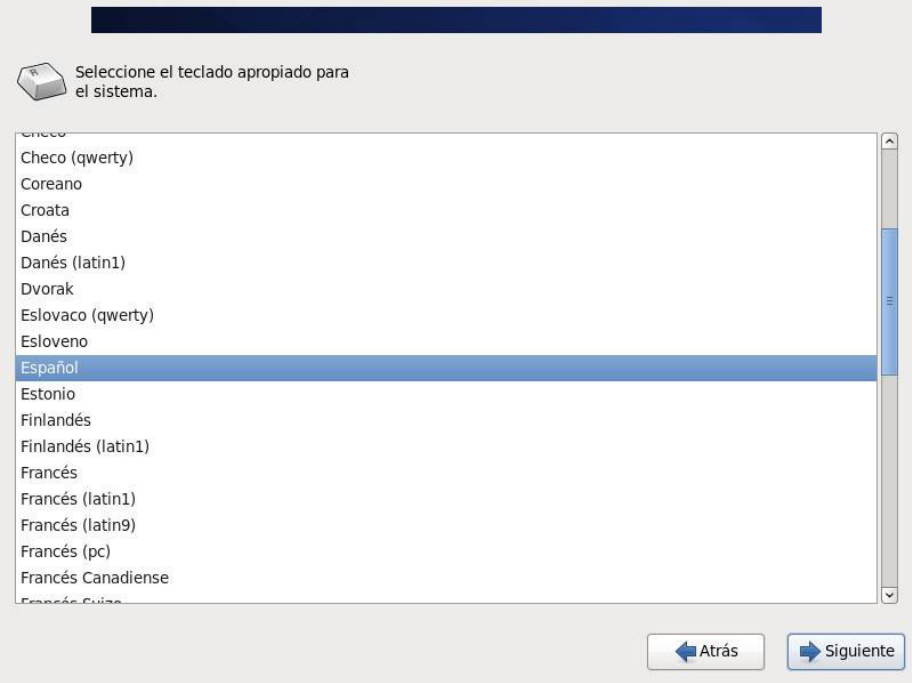
A continuación indicaremos qué tecleado estamos empleando (9). En nuestro caso lo más normal es utilizar un teclado Español, pero es posible que estemos instalando este sistema operativo en una computadora cuyo tecleado sea americano, inglés, francés, etc. de tal forma que tendríamos que elegir el correcto en cada caso.

Gráfico Anexo 1. 4: Selección de idioma en castellano



Fuente: Centos

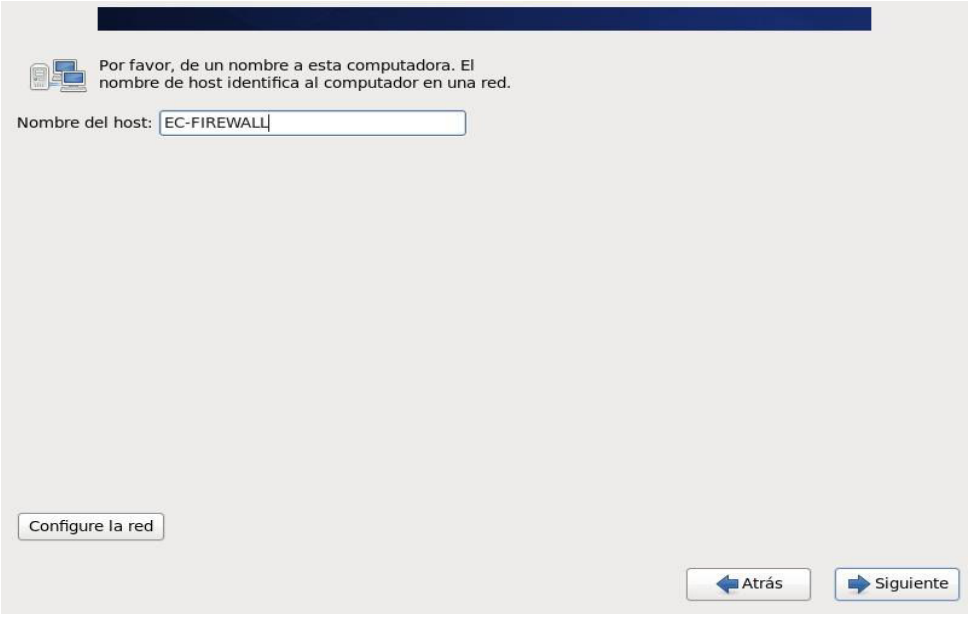
Gráfico Anexo 1. 5: Elección del teclado en español



Fuente: Centos

El programa de instalación seguirá realizando preguntas para poder configurar correctamente el equipo. En este punto preguntará por el nombre de la computadora.

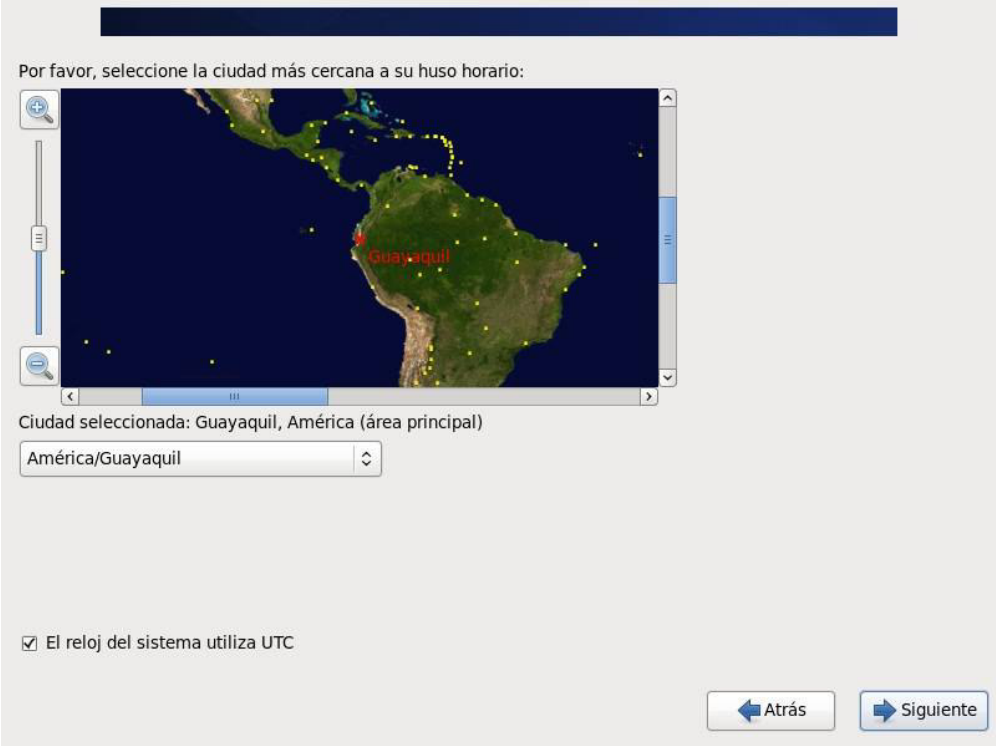
Gráfico Anexo 1. 6: Configuración del nombre del equipo



Fuente: Centos

A continuación se mostrará un mapa del mundo para elegir la zona horaria para nuestro equipo. Esta zona horaria la podremos seleccionar también mediante el menú desplegable que aparece debajo del mapa. En nuestro caso elegiremos Guayaquil Ecuador.

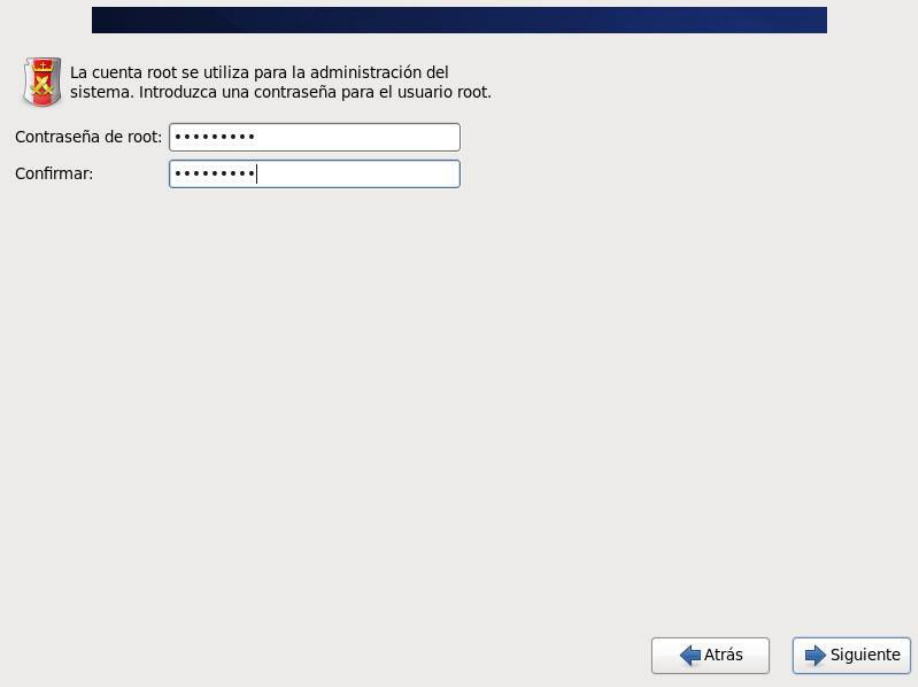
Gráfico Anexo 1. 7: Elección de la zona horaria



Fuente: Centos

El siguiente paso que se plantea es la elección de la contraseña para el super usuario root) del sistema. En el menú que se presenta escribir la contraseña elegida y la confirmar, para evitar equivocaciones.

Gráfico Anexo 1. 8: Elección de la contraseña de root

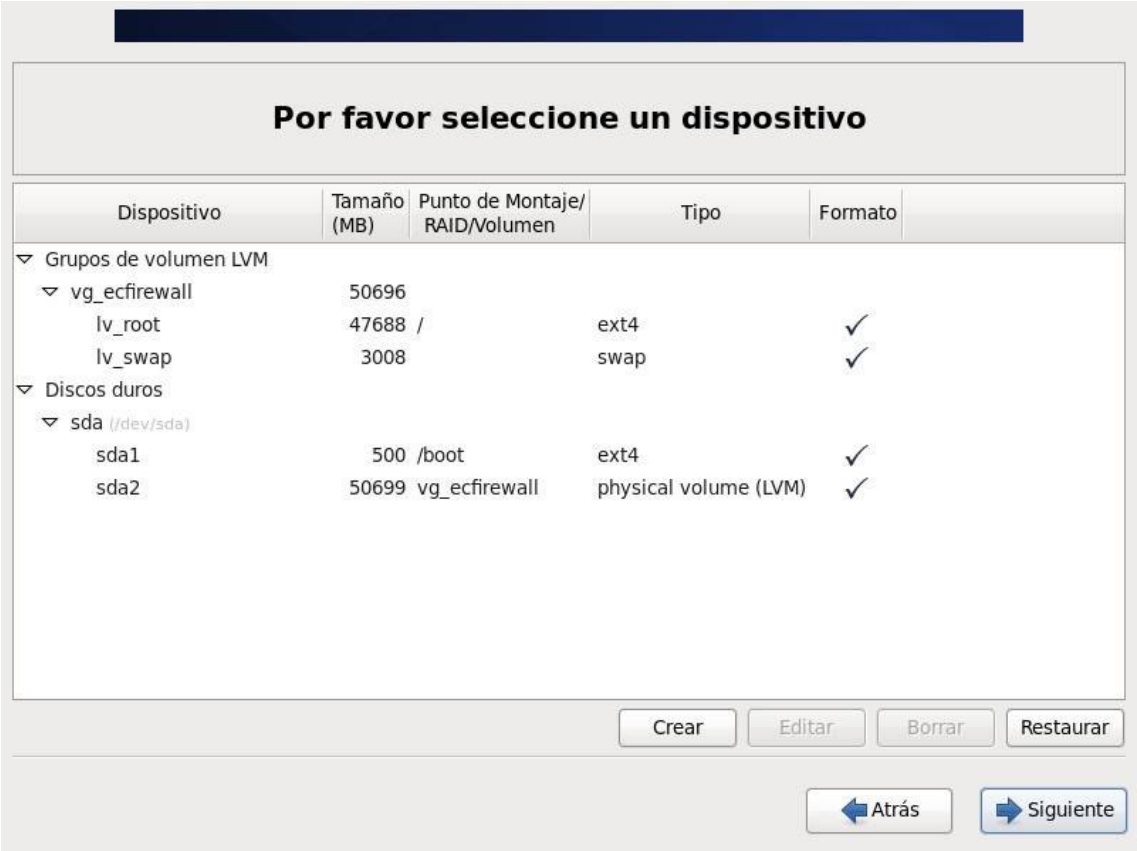


Fuente: Centos

Por supuesto en cuanto a contraseñas se refiere, y más aún si estamos hablando de la contraseña de root, debemos tener en cuenta que al menos debemos elegir una contraseña con un mínimo de ocho caracteres, utilizando para las mismas minúsculas, mayúsculas, números y caracteres especiales. La elección de una contraseña débil redundará en un posible acceso no autorizado en nuestro sistema, con todas las complicaciones que esto conlleva. El menú a continuación se refiere al particionado de nuestro disco duro, con el fin de disponer de las particiones necesarias para la instalación de nuestro sistema operativo Centos 6.5. En rigor sólo necesitaremos crear una partición, que se llamará partición raíz y de la cual colgarán todos los directorios/archivos de nuestro sistema Linux. No obstante cabe decir que una distribución Linux debe pensarse con varias particiones. Cada una de estas particiones privilegiará un directorio. Con esto queremos decir que todo lo contenido en estos directorios será almacenado en la partición generada a tal efecto, de ahí que hayamos dicho que lo privilegiamos.

Cabe decir que anaconda, el programa de instalación que se está utilizando para instalar Centos 6.5, es capaz de crear particiones sin interacción por parte del usuario, pero por supuesto estas particiones serán como están programadas.

Gráfico Anexo 1. 9: Partición completa del Disco Duro

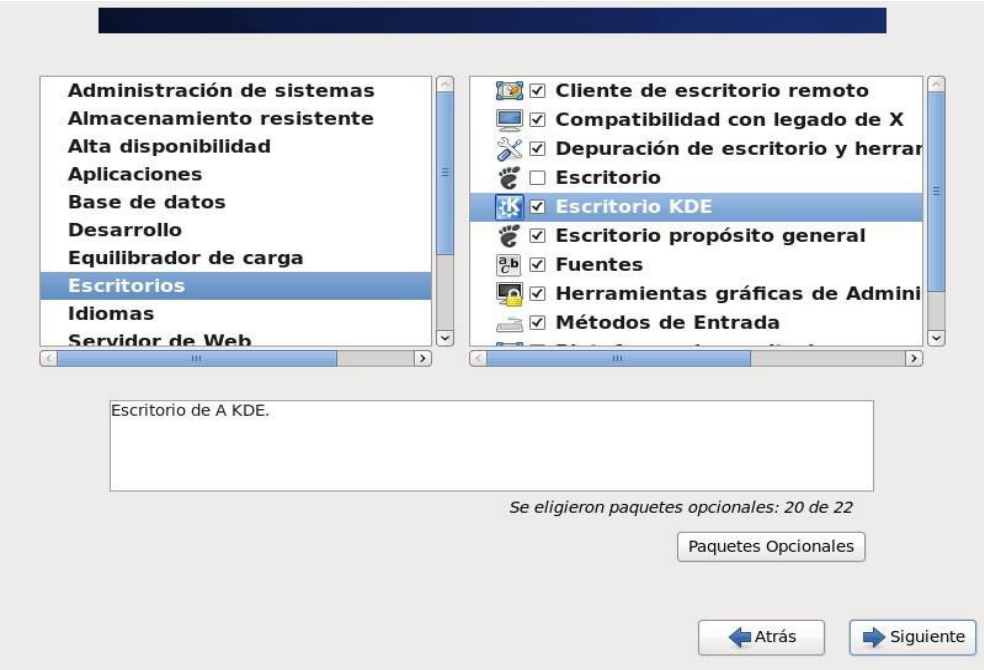


Fuente: Centos

Si tuviéramos más de un disco duro donde se pudiera crear esta partición, podríamos seleccionarlo en el apartado Unidades admisibles, al que no podremos acceder si sólo tenemos un disco duro en nuestro sistema. Se debe indicar a continuación el tamaño de la partición. Este tamaño dependerá de las particiones que creemos y del uso que demos al equipo. En este momento aparecerá un aviso indicando que en cuanto se acepte (pulsar en Guardar los cambios al disco), el disco quedará afectado y no se podrá volver a la configuración previa. Si se está seguro de que se ha particionado correctamente el disco duro, aceptar el mensaje para que los cambios en el disco tomen efecto.

A continuación se indicará al programa de instalación de Centos 6.5 cómo se quiere elegir el software que se instalará en el equipo. Se muestra la posibilidad de instalar en el equipo un conjunto de aplicaciones pre establecido por parte de los desarrolladores de Centos 6.5, que hagan que la máquina se comporte adecuadamente si lo que necesitamos es software de ofimática, o si se quiere desarrollar software y lo que se necesita son compiladores y herramientas que ayuden en este sentido o si lo que se quiere es instalar un servidor web.

Gráfico Anexo 1. 10: Elección de paquetes a instalar



Fuente: Centos

De una manera o de otra siempre se puede, una vez se tenga al equipo operativo, instalar y desinstalar el software que deseemos. Si se ha elegido personalizar el software que se quiere instalar, se ofrecerán diferentes familias de software. Estas familias de software permiten ordenar de una forma sencilla todo el software que ofrece Centos 6.5, de tal forma que si se quiere buscar un programa en concreto, no habrá una lista interminable donde buscar, sino que se puede ir acotando debido a este orden que propone Centos 6.5.

Gráfico Anexo 1. 11: Instalación de los programas seleccionados

Fuente: Centos

Una vez se hayan transferido al disco duro los programas elegidos, se nos mostrará una pantalla indicando el fin de la instalación. El arranque del sistema comenzará con la elección de Centos 6.5 en el menú grub que aparecerá en el arranque. Esta elección no la tiene que hacer nunca si sólo se tiene Centos 6.5 instalado en el equipo, pero si existe más de un sistema operativo será necesario elegir qué sistema operativo se quiere utilizar.

Sólo la primera vez que arranque Centos 6.5, aparecerá un menú de Bienvenida en el que se darán detalles del sistema operativo y se harán ciertas preguntas para acabar la configuración del equipo.

Gráfico Anexo 1. 12: Tras la instalación se reinicia el equipo

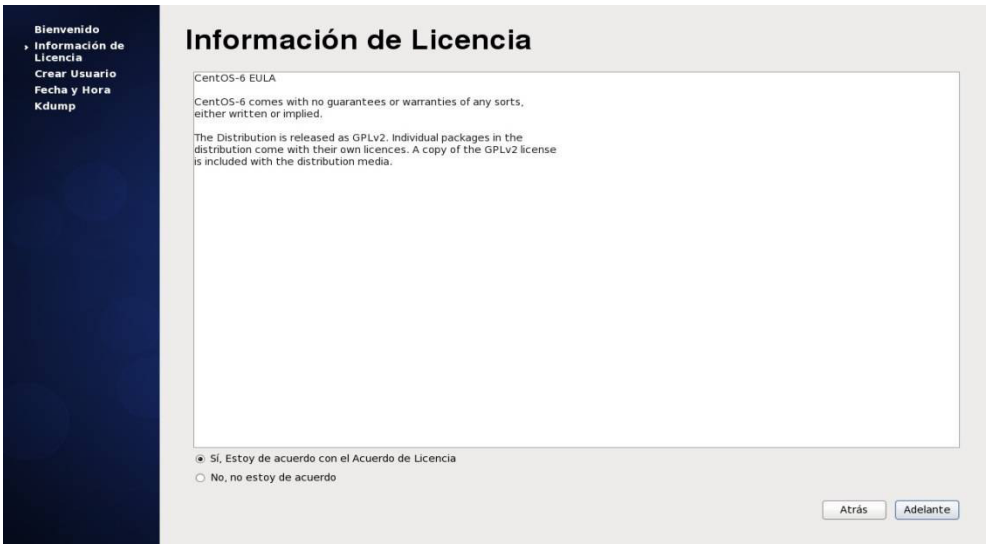
Fuente: Centos

Gráfico Anexo 1. 13: Mensaje de bienvenida – Primer arranque de Centos 6.5

Fuente: Centos

Lo primero sobre lo que se informa es sobre la licencia, que se puede leer y pulsar Siguiente.

Gráfico Anexo 1. 14: Información de licencia



Fuente: Centos

A continuación se despliega un menú para la creación de un usuario para la utilización del sistema operativo.

Gráfico Anexo 1. 15: Creación de usuario

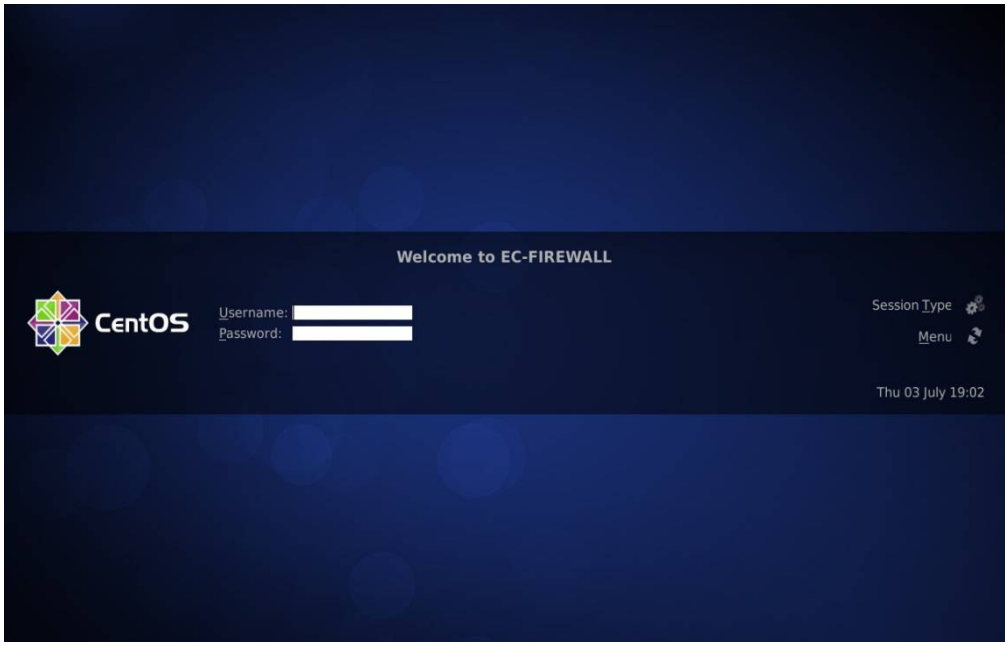
Fuente: Centos

Los datos que piden son sencillos de cumplimentar, ya que se debe indicar el login que se quiere para este usuario, así como su Nombre y Apellidos (aunque estos datos no son obligatorios) y la contraseña elegida.

Gráfico Anexo 1. 16: Configuración de fecha y hora

Fuente: Centos

Gráfico Anexo 1. 17: Pantalla de login – Finalizada instalación de Centos 6.5



Fuente: Centos