CAPITULO II

2. ELEMENTOS NECESARIOS PARA LA CONFIGURACIÓN Y FUNCIONAMIENTO DEL SERVIDOR PROXY Y FIREWALL

2.1. Parámetros que se toman en cuenta para configurar un Servidor Proxy y Firewall con Linux Red Hat Enterprise 4

El término en ingles **«Proxy»** tiene un significado muy general y al mismo tiempo ambiguo, aunque invariablemente se considera un sinónimo del concepto de **«Intermediario»**. Se suele traducir, en el sentido estricto, como **delegado** o **apoderado** (el que tiene el que poder sobre otro).

Un **Servidor Intermediario** (Proxy) se define como una computadora o dispositivo que ofrece un servicio de red que consiste en permitir a los clientes realizar conexiones de red indirectas hacia otros servicios de red. Durante el proceso ocurre lo siguiente:

- Cliente se conecta hacia un Servidor Intermediario (Proxy).
- Cliente solicita una conexión, fichero u otro recurso disponible en un servidor distinto.
- **Servidor Intermediario** (Proxy) proporciona el recurso ya sea conectándose hacia el servidor especificado o sirviendo éste desde un caché.

- En algunos casos el **Servidor Intermediario** (Proxy) puede alterar la solicitud del cliente o bien la respuesta del servidor para diversos propósitos.

Los **Servidores Intermediarios** (Proxies) generalmente se hacen trabajar simultáneamente como muro cortafuegos operando en el **Nivel de Red**, actuando como filtro de paquetes, como en el caso de **iptables**, o bien operando en el **Nivel de Aplicación**, controlando diversos servicios, como es el caso de **TCP Wrapper**. Dependiendo del contexto, el muro cortafuegos también se conoce como **BPD** o **B**order **P**rotection **D**evice o simplemente **filtro de paquetes**.

Una aplicación común de los **Servidores Intermediarios** (Proxies) es funcionar como caché de contenido de Red (principalmente HTTP), proporcionando en la proximidad de los clientes un caché de páginas y ficheros disponibles a través de la Red en servidores HTTP remotos, permitiendo a los clientes de la red local acceder hacia éstos de forma más rápida y confiable.

Cuando se recibe una petición para un recurso de Red especificado en un **URL** (**U**niform **R**esource **L**ocator) el **Servidor Intermediario** busca el resultado del **URL** dentro del caché. Si éste es encontrado, el **Servidor Intermediario** responde al cliente proporcionado inmediatamente el contenido solicitado. Si el contenido solicitado no estuviera disponible en el caché, el **Servidor Intermediario** lo traerá desde servidor remoto, entregándolo al cliente que lo solicitó y guardando una copia en el caché. El contenido en el caché es eliminado luego a través de un algoritmo de expiración de acuerdo a la

antigüedad, tamaño e historial de **respuestas a solicitudes** (hits) (ejemplos: **LRU**, **LFUDA** y **GDSF**).

Los **Servidores Intermediarios** para contenido de Red (Web Proxies) también pueden actuar como filtros del contenido servido, aplicando políticas de censura de acuerdo a criterios arbitrarios.

Acerca de Squid.

Squid es un **Servidor Intermediario** (Proxy) de alto desempeño que se ha venido desarrollando desde hace varios años y es hoy en día un muy popular y ampliamente utilizado entre los sistemas operativos como GNU/Linux y derivados de Unix®. Es muy confiable, robusto y versátil y se distribuye bajo los términos de la Licencia Pública General GNU (**GNU/GPL**). Siendo sustento lógico **libre**, está disponible el código fuente para quien así lo requiera.

Entre otras cosas, **Squid** puede funcionar como **Servidor Intermediario** (Proxy) y **caché de contenido de Red** para los protocolos **HTTP**, **FTP**, **GOPHER** y **WAIS**, Proxy de **SSL**, caché transparente, **WWCP**, aceleración **HTTP**, caché de consultas DNS y otras muchas más como filtración de contenido y control de acceso por IP y por usuario.

Squid consiste de un programa principal como servidor, un programa para búsqueda en servidores **DNS**, programas opcionales para reescribir solicitudes y realizar autenticación y algunas herramientas para administración y herramientas para clientes. Al iniciar **Squid** da origen a un número configurable (5, de modo predefinido a través del parámetro **dns_children**) de procesos de búsqueda en

servidores **DNS**, cada uno de los cuales realiza una búsqueda única en servidores **DNS**, reduciendo la cantidad de tiempo de espera para las búsquedas en servidores **DNS**.

NOTA ESPECIAL: Squid no debe ser utilizado como Servidor Intermediario (Proxy) para protocolos como **SMTP**, **POP3**, **TELNET**, **SSH**, **IRC**, etc. Si se requiere intermediar para **cualquier protocolo distinto** a **HTTP**, **HTTPS**, **FTP**, **GOPHER** y **WAIS** se requerirá implementar obligatoriamente un enmascaramiento de IP o **NAT** (**N**etwork **A**ddress **T**ranslation) o bien hacer uso de un servidor **SOCKS** como **Dante.**

Algoritmos de caché utilizados por Squid.

A través de un parámetro (**cache_replacement_policy**) **Squid** incluye soporte para los siguientes algoritmos para el caché:

- **LRU** Acrónimo de **L**east **R**ecently **U**sed, que traduce como **Menos Recientemente Utilizado**. En este algoritmo los objetos que no han sido accedidos en mucho tiempo son eliminados primero, manteniendo siempre en el caché a los objetos más recientemente solicitados. **Ésta política es la utilizada por Squid de modo predefinido**.

- **LFUDA** Acrónimo de **L**east **F**requently **U**sed with **D**ynamic **A**ging, que se traduce como **Menos Frecuentemente Utilizado con Envejecimiento Dinámico**. En este algoritmo los objetos más solicitados permanecen en el caché sin importar su tamaño

optimizando la **eficiencia** (hit rate) por **octetos** (Bytes) a expensas de la eficiencia misma, de modo que un objeto grande que se solicite con mayor frecuencia impedirá que se pueda hacer caché de objetos pequeños que se soliciten con menor frecuencia.

- **GDSF** Acrónimo de **G**reedy**D**ual **S**ize **F**requency, que se traduce como **Frecuencia de tamaño *GreedyDual*** (*codicioso dual*), que es el algoritmo sobre el cual se basa **GDSF**. Optimiza la **eficiencia** (hit rate) por objeto manteniendo en el caché los objetos pequeños más frecuentemente solicitados de modo que hay mejores posibilidades de lograr **respuesta a una solicitud** (hit). Tiene una eficiencia por **octetos** (Bytes) menor que el algoritmo **LFUDA** debido a que descarta del caché objetos grandes que sean solicitado con frecuencia.

Sustento Lógico Necesario para la Configuración del SQUID:

Para poder llevar al cabo los procedimientos descritos en este manual y documentos relacionados, usted necesitará tener instalado al menos lo siguiente:

- Al menos squid-2.5.STABLE6
- Httpd-2.0.x (Apache), como auxiliar de caché con aceleración

- Todos los parches de seguridad disponibles para la versión del sistema operativo que esté utilizando. No es conveniente utilizar un sistema con posibles vulnerabilidades como Servidor Intermediario.

Debe tomarse en consideración que, de ser posible, se debe utilizar **siempre** las versiones estables más recientes de todo sustento lógico que vaya a ser instalado para realizar los procedimientos descritos en este manual, a fin de contar con los parches de seguridad necesarios. Según un sondeo de opinión realizado por el grupo investigador ninguna versión de **Squid** anterior a la **2.5.STABLE6 se** considera como apropiada debido a fallas de seguridad de gran importancia.

Squid no se instala de manera predeterminada a menos que especifique lo contrario durante la instalación del sistema operativo, sin embargo viene incluido en casi todas las distribuciones actuales. El procedimiento de instalación es exactamente el mismo que con cualquier otro sustento lógico.

Instalación a través de yum.

Si cuenta con un sistema con CentOS o White Box Enterprise Linux 3 o versiones posteriores, utilice lo siguiente y se instalará todo lo necesario junto con sus dependencias:

```
yum -y install squid httpd
```

Instalación a través de up2date.

Si cuenta con un sistema con Red Hat™ Enterprise Linux 3 o versiones posteriores, utilice lo siguiente y se instalará todo lo necesario junto con sus dependencias:

```
up2date -i squid httpd
```

SEGURIDADES EN LA RED, FIREWALL E IPTABLES

El presente trabajo de investigación pretende ayudar al administrador a instalar y configurar un firewall dentro de una maquina con Linux, así mismo, como varias maquinas pueden salir a otras redes por medio de una sola dirección IP (Enmascaramiento).

Esta primera versión cubriría aspectos básicos de instalación y configuración, esperamos ir mejorando con el paso del tiempo este documento y pueda ser una valiosa guía.

Está investigación a sido probado en una maquina con un procesador a 2.0 Ghz y 1024 Mbytes en RAM, dos tarjetas de red una de Fast Ethernet 3COM y una de red inalámbrica, así como también un disco duro de 120Gbytes. El equipo tiene instalado Linux Enterprise Server 4 de Red Hat™) y la aplicación de Firewall fue instalado del mismo disco de las Maquinas Virtuales, donde se localizan la mayor parte de los RPM's. La maquina en la que ha sido probada funciona como gateway, Proxy y firewall entre dos redes (una red corporativa: que comprende 3 maquinas virtuales), el cual protege a una de la otra de accesos indebidos a equipos de misión crítica.

Por otra parte, si tu deseas salir a Internet desde una línea telefónica y también los equipos de la red interna primero debes de leer el manual de configuración PPP (Point to Point) para que tu Servidor Linux pueda tener acceso a Internet; Una vez configurado tu salida a Internet puedes proseguir con este documento.

Finalmente, toda colaboración, corrección y demás es bien recibido para poder mejorar este documento.

Breve historia sobre Firewall y Enmascaramiento de direcciones

No hace mucho tiempo una red de computadores era algo que no muchas empresas o escuelas tenían en su centro de cómputo. La gran mayoría de estas computadoras se encontraban aisladas una de la otra aunque mantener en el mismo cuarto o laboratorio. Únicamente grandes empresas, Instituciones Gubernamentales o Universidades tenían este recurso. Eso a cambiado con el paso del tiempo y hoy en día es común encontrar computadoras conectadas a una red y obtener información de ella o brindar servicios (tal es el caso de Internet o una intranet en una oficina u hogar), y es común el uso de ellas, como por ejemplo enviar y recibir correo electrónico, entre muchos servicios más.

El gran desarrollo de estas redes no ha sido del todo positivo en varios aspectos. Uno de ellos es la disponibilidad de Direcciones IP, que esta limitado a 4.300 millones de direcciones IP validas aproximadamente. Esta cantidad de direcciones puede ser a primera vista muchísimas direcciones, pero direcciones validas libres en Internet son actualmente muy pocas, por lo que cada vez es más difícil poder obtener una dirección valida en Internet. Con la llegada de la

versión 6 del protocolo IP se espera poder extender este rango de direcciones en un par de millones más. Pero como esta nueva versión aun no se encuentra disponible debemos de trabajar con la actual (IPv4) y por ende debemos administrar mejor el uso de este tipo de direcciones. Una forma de administrar mejor esto, es escondiendo computadoras con direcciones no validas dentro de una red, detrás de una dirección IP valida. A esta técnica se le conoce como enmascaramiento de direcciones.

Existe otro problema que no es técnico sino social. Cada día existen más computadoras y personas que accedan a Internet. La necesidad de proteger los sistemas conectados a una red de usuarios no deseados es cada vez más común y se vuelve más importante día a día.

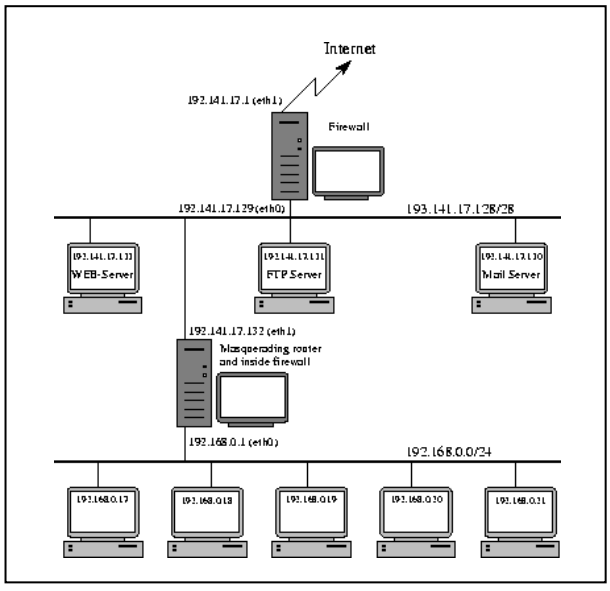

Figura 2.1: Ilustración de un Firewall con Linux
Fuente: Grupo Investigador. (www.linuxparatodos.com)

Instalar un firewall es en buena medida una buena solución para protegerse de ataques a una red interna o de usuarios no deseados. Actualmente, el Kernel de

Linux (Por ejemplo LinuxPPP 6.2, Red Hat™ 4) soporta filtrado de paquetes, que pueden se utilizados para implementar un sencillo firewall.

Dentro de este documento se hablara del mecanismo para el filtrado de paquetes, como puede ser utilizado para el enmascaramiento de paquetes y construcción de un Firewall. El RPM que se encuentra dentro del disco de Linux contiene el software necesario para que el firewall y el enmascaramiento funcionen correctamente.

El filtrado de paquetes

Actualmente Linux soporta el filtrado de paquetes. La versión de Kernel 2.2 de Linux contiene cambios significativos en su estructura para brindar este servicio.

Existen cadenas o reglas que los paquetes IP deben de igualar para que este pueda ser aceptado. Si llega algún paquete al equipo una regla decide que hacer con el. Este paquete puede ser aceptado, negado, rechazado, enmascarado o enviado a otra regla.

Con este mecanismo es fácil construir reglas sencillas para el filtrado de paquetes con un firewall. Esto quiere decir que todos lo paquetes que lleguen a la maquina, independientemente si son TCP o UDP, serán primero filtrados antes de ser enviados a si destino.

Linux soporta un gran número de características para las reglas de un firewall y el enmascaramiento de paquetes.

Cadenas IP

Las cadenas de un firewall no son mas que reglas que se utilizan para que el paquete cumpla con alguna de ellas y en un cierto orden. Esto quiere decir que el paquete debe de cumplir con alguna regla. La regla determina que es lo que va a suceder con el paquete que ha sido recibido. Si el paquete no coincide la próxima regla determinara que hacer con el. Si llega al final de esta regla se utilizara la política que se encuentra por omisión.

2.2. Estándares de calidad de servicio y rendimiento a seguir para la instalación y configuraciones de las Maquinas virtuales y de los servidores Proxy y Firewall en Linux.

Una máquina virtual es un software especializado que permite tener instalado más de un sistema operativo de manera simultánea sobre la misma máquina. Pues sí, sin más ni más. A través de las máquinas virtuales podemos realizar el sueño de tener un sistema base como lo puede ser Windows (a lo que se le conoce como sistema anfitrión o Host) y de manera simultánea ejecutar Linux (convirtiéndose en el sistema invitado o Guest), o si lo prefieren otra vez GNU/Linux, Novell Netware, Sun Solaris y otros más que ya se darán cuenta al momento de instalarla.

Ejecutar una máquina virtual es como abrir un programa cualquiera, sólo que en lugar de centrarnos en el uso de las herramientas del programa como tal (aunque igual tiene muchas opciones) lo que hacemos es 'arrancar' otro PC dentro de

nuestro sistema Host, entonces podremos ver cómo se expone la pantalla de booteo en una ventana. Desde este punto ya tratamos a esta aplicación como si en realidad fuera otro PC físicamente, así que podemos realizar tareas como introducir un disquete o CD booteable y veremos cómo efectivamente el PC virtual se reinicia con el mismo, y de hay en adelante es más bien “carpintería”.

Como lo que estamos haciendo es montar un PC virtual sobre uno físico, tendremos de todos modos que asignarle recursos físicos que no salen de otro lado que del mismo PC.

Disco duro y memoria son los más críticos, al fin y al cabo la tarjeta de red o las unidades de CD se pueden compartir sin ningún problema.

En teoría podremos montar todas las máquinas virtuales que necesitemos, siempre y cuando nuestros recursos físicos puedan soportar la demanda.

Como lo dije al principio, vamos a instalar VMware Workstation sobre Windows Xp Service Pack II, siendo este el SO anfitrión o Host. Igualmente se puede lo contrario, estando en Linux, instalamos VMware y montamos una máquina virtual para instalar GNU/Linux.

Figura 2.2: Ilustración del VMWare Workstation 6
Fuente: Grupo Investigador.

Al igual que todos los programas compatibles con Windows constan de un asistente el mismo que guiara durante la instalación, hay que tener en cuenta la ubicación de las maquinas virtuales, es preferible encaminar de manera diferente a la dirección que nos da por defecto que para este caso es:

C:\Documents and Settings\Administrador\Mis documentos\Mis maquinas virtuales

Una vez instalada las maquinas virtuales tenemos que configurar de acuerdo al sistema operativo que se desee instalar así las ventajas que tiene está maquina virtual es que se puede obtener los recursos que brinda el sistemas operativo Windows de Microsoft , el GNU Linux con sus distintas presentaciones y versiones, así como también el Solaris de la empresa SUN Microsystems.

La pantalla principal del VMWare consta de las siguientes partes

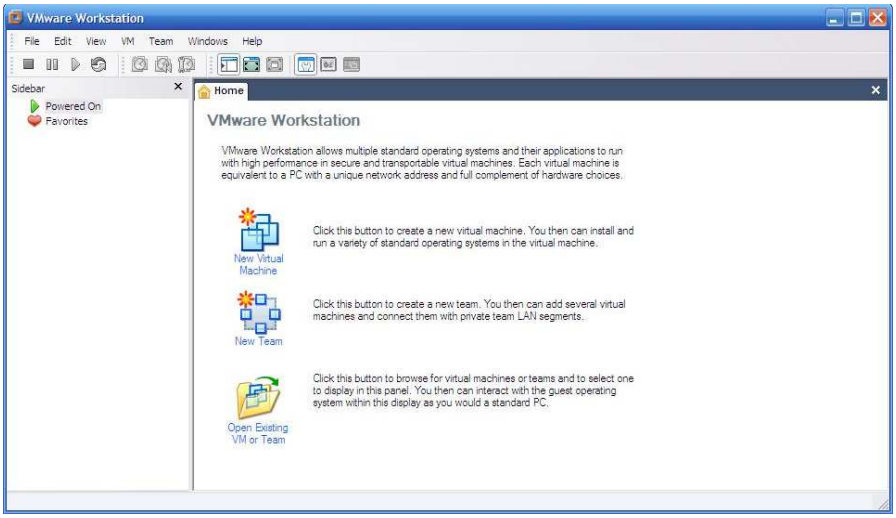

Figura 2.3: Ilustración del VMWare Workstation
Fuente: Grupo Investigador.

La pantalla de principal del VMWare consta de 3 segmentos claramente identificados los mismos que están divididos en menú principal, pantalla de administración de la Maquina Virtual VMWare y el sitio donde están alojadas las maquinas virtuales que van a ser configuradas.

Ahora vamos a proceder a instalar una maquina virtual, para el sistema operativo que vamos a utilizar para las configuraciones que constan en nuestro objetivo del proyecto planteado.

Figura 2.4: Creación de una nueva maquina virtual
Fuente: Grupo Investigador

Luego de empezar el asistente para crear un nuevo asistente de creación de un equipo virtual debemos configurar todas las propiedades que pueden tener el hardware del equipo host que va albergar las maquinas virtuales.

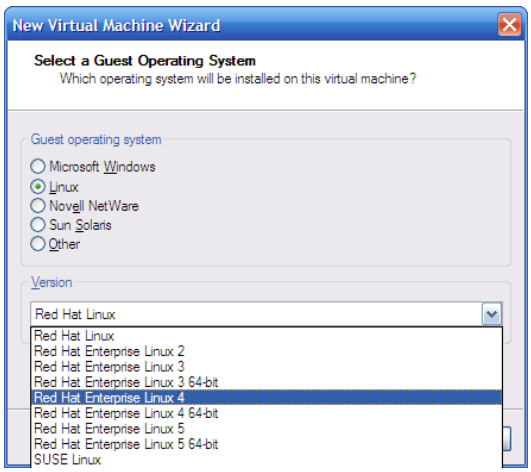

Figura 2.5: Creación de una nueva maquina virtual
Fuente: Grupo Investigador

Escogemos el sistema operativo que vamos a configurar el mismo que nos va a servir de acuerdo al equipo que tenemos para el desarrollo de este trabajo investigativo que es el Linux Red Hat Enterprise 4

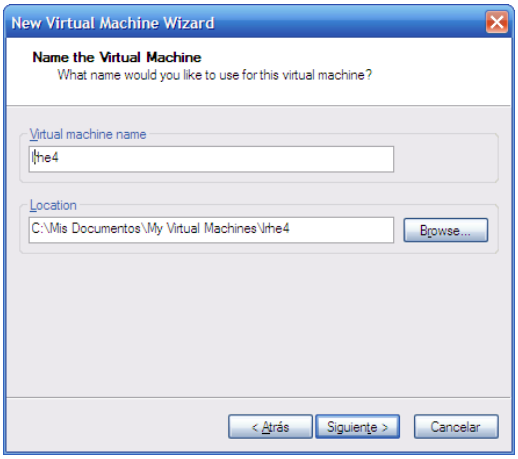

Figura 2.6: Creación de una nueva maquina virtual
Fuente: Grupo Investigador

Cuando escogemos un nombre para las maquinas virtuales debemos poner un nombre que sea fácil de recordar que en nuestro caso es lrhe4 que son las siglas de Linux red Hat Enterprise 4, ya que este nombre va a ser el nombre del archivo así como de la carpeta que contiene todas las configuraciones.

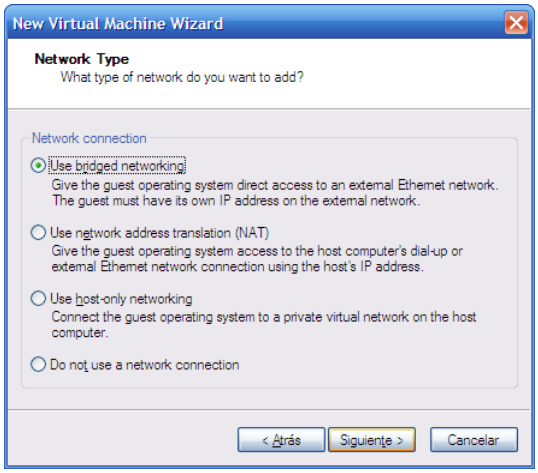

Figura 2.7: Creación de una nueva maquina virtual
Fuente: Grupo Investigador

Luego debemos escoger las configuraciones de las redes si es que estas van a salir a través de hardware, o si se quiere conectar dentro del PC como simulación de una red de área local interna.

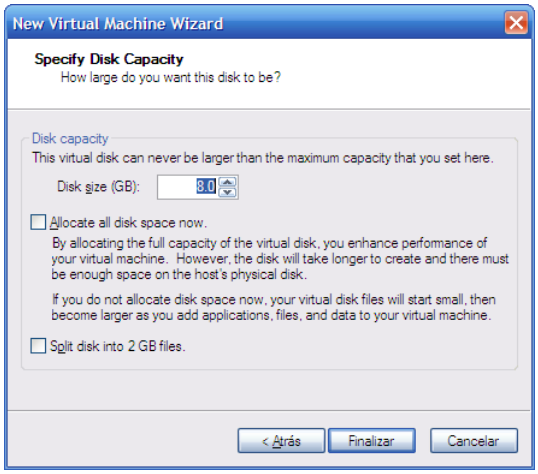

Figura 2.8: Creación de una nueva maquina virtual
Fuente: Grupo Investigador

El espacio requerido para instalar el sistema operativo en la maquina virtual es el ultimo paso en las configuraciones, debemos estar seguros de lo que va a realizar el S.O. ya que según esto vamos a dar la capacidad necesaria para la maquina virtual sin que afecte al rendimiento de los equipos.

Figura 2.9: Creación de una nueva maquina virtual
Fuente: Grupo Investigador

Finalmente se concluye con el proceso de creación de las maquinas virtuales y estaría listas para la instalación del sistema operativo.

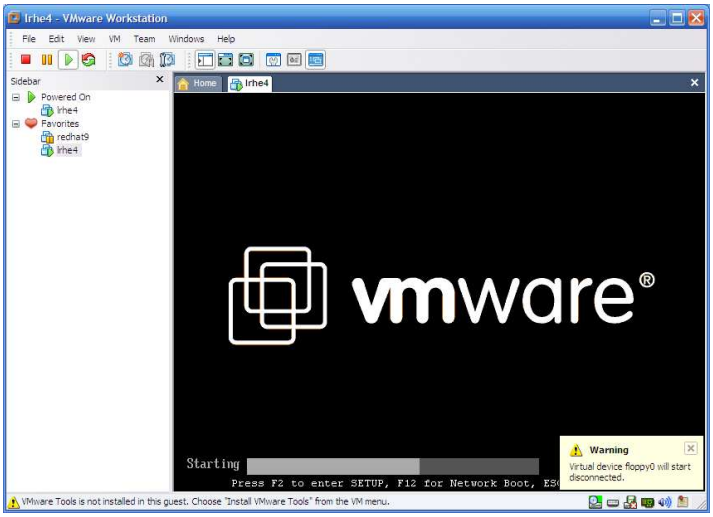

Figura 2.10: Creación de una nueva maquina virtual
Fuente: Grupo Investigador

Una vez instalado la maquina virtual procedemos a iniciar el VMWare para la instalación del sistema operativo para lo cual reinicia como si fuera una maquina real, lo que existe de igual manera es el setup ya que se considera que es una PC real ya que hace el conteo normalmente de la memoria RAM.

2.3. Logros e Insuficiencias encontradas en la manera de implementación en equipos físicos

Para poder encontrar los logros e insuficiencias que se pueden observar con la implementación de equipos físicos (host), hemos realizado una serie de entrevistas con administradores de servidores los mismos que nos ha dado las siguientes observaciones:

- Sobre la utilización de servidores físicos nos manifiestan que tener un servidor para cada servicio resulta importante ya que garantizan las seguridades con la configuración de un servidor firewall, así como también el buen servicio que deben tener los usuarios de las redes al momento de recibir un recurso tan importante como lo es el Internet con la implementación de un servidor Proxy.

- El tener un servidor firewall en un solo equipo es ventajoso siempre y cuando este pueda estar ocupando todos los recursos que debe tener un hardware avanzado y que este dedicado a precautelar la información que tenga la red de datos, cabe mencionar que en algunas empresas disponen de 2 y hasta de 3 servidores de firewall los mismos que están dedicados a proteger tanto de usuarios externos como internos, y en muchas de las empresas precautelan los DMZ.
- Un servidor Proxy como en la mayoría de instituciones o empresas que utilizan el Linux Red Hat como Sistema Operativo requiere de muchos recursos los mismos que deben constar físicamente en el servidor y no puede ser compartido con otras actividades ya que podría ocasionar un daño y cortaría el servicio de Internet a toda la institución, por lo que nos manifestaban, que siempre el Proxy era un servidor que está solamente dedicado a ser de Proxy o a compartir el recurso del Internet.

- Todos estos servicios trabajando por separado garantizan que la institución trabaje sin problema alguno, pero limitado cuando no se cuenta con un buen presupuesto económico el mismo que garantizaría el normal desenvolvimiento de las funciones diarias de trabajo de las empresas.

2.4. Análisis de los resultados obtenidos de las fuentes consultadas, a nivel de administración, docentes y estudiantes de la especialidad de Sistemas

- De lo explicado por algunos de los administradores de redes, es necesario trabajar en forma independiente, tanto los servidores de Firewall, como el Proxy ya que el primero garantiza la seguridad de la información, mientras que el segundo comparte el recurso más preciado en este tiempo como lo es el Internet.
- Cabe mencionar que para las empresas cuenten con estos servicios requieren de una fuerte inversión, si tomamos en cuenta que precios son demasiados altos en lo que son servidores físicos y más aun si tomamos en cuenta que para equipar un departamento de sistemas de manera optima requiere cuando menos de 6 a 7 servidores para cada una de las actividades que normalmente se desarrollan en estos departamentos.
- Nos comentan que la implementación y posterior administración de los recursos de redes e muy delicado y que en la mayoría de casos hay que equipar a los servidores de fuente de alimentación permanente como son los casos de los UPS y los reguladores de voltaje, y esto por supuesto acarrea otra inversión fuerte.