Seguridad en red con software libre

TFC-UOC 2013

Ingeniería Técnica de Informática de Sistemas

Autor: Juan Diego Toval Barjacoba

Tutor: Joaquin Lopez Sanchez

19/05/2013

Dedicatoria y agradecimientos:

Quisiera dedicar este trabajo de fin de carrera a mi novia Beatriz González por el apoyo y comprensión que me ha dado durante estos años.

También quiero agradecer a la familia y amigos por la ayuda que me han prestado para que yo pudiese centrarme en los estudios y sobre todo por la paciencia que han tenido conmigo ya que no he podido dedicarles todo el tiempo que quisiera.

Por otro lado agradecer en especial a mi tutor del proyecto Joaquín López Sánchez Montañés que me ha estado orientando con cada entrega y a los profesores que han hecho posible que llegase este momento, gracias por vuestra paciencia.

Índice de contenido

Índice de ilustraciones

Seguridad en red con software libre (LINUX)

1. Introducción:

En este proyecto se aborda el tema de cómo poder dar seguridad a una red mediante el software libre. Ver como siguiendo unos pasos y utilizando las herramientas oportunas que están a nuestro alcance se puede proteger de posibles ataques externos malintencionados que quieran acceder a información privada o hacer algún tipo de daño como borrado de datos, obtener cuentas de usuarios, etc.

2. Objetivos:

Dar seguridad a la red y a sus equipos mediante software libre (Linux) como podría ser una pequeña PYME o una pequeña Red domestica, con ello conseguiremos abaratar costes que puedan acarrear tanto por compras de licenciamientos como por su mantenimiento, ya que con software libre como veremos más adelante conseguiremos una red segura y actualizable con un coste cero, salvo por aquellos equipos hardware necesarios los cuales serian necesarios en cualquiera de las plataformas.

Analizar las alternativas que más nos interese para que pueda cubrir las necesidades que se nos puedan surgir o aparecer, tanto por que crezca la red o por necesitar nuevos servicio, ya que en el mundo de las tecnologías está en constante movimiento.

Además la seguridad de la red es un tema fundamental a la hora de proteger nuestra información tanto sea privada como de acceso restringido, de todas aquellas personas ajenas que intenten acceder a dicha información, o protegernos de posibles ataques (Virus, troyanos, gusanos,....)

3. Memoria:

En el apartado de Planificación: Se numeran los plazos y la evolución de los trabajos que ello implica para la finalización del proyecto.

En el apartado de Topología de red: Se indican los diferentes configuraciones que podemos dar a nuestra red tanto de forma física como lógica. Hemos añadido la topológica propia para nuestra red de la PYME con una estructura DMZ, Explico las las diferentes redes que podemos montar pero para nuestro diseño montaremos distintas Lan, con privilegios y restricción de acceso a servidores o conexiones a exterior más la

correspondiente zona DMZ , también indico el material necesario para poder desarrollar este proyecto (Software y Hardware).

En el apartado de Seguridad: Indico los pasos y medidas que hay que tomar para protegernos de cualquier posible amenaza o pérdida de información y cuáles son los sistemas operativos recomendables.

También especifico como realizar una conexión segura desde la instalación del servidor SSH como su configuración y he añadido alguna prueba que he realizado de forma aleatoria.

En el apartado de Ataques externos: Hago una breve introducción a las posibles amenazas externas que nos rodean, de cómo actúan y que procedimientos o herramientas podemos utilizar para poder llegar a detectar o prevenir.

Explico la utilización de la herramienta Snort desde su instalación hasta su posterior uso, ya que es una herramienta fundamental para la prevención de ataques. Snort se suele instalar el firewall pero gracias a su bajo coste de recursos lo podremos instalar en todas las maquinas de nuestra red.

En el apartado de Herramientas preventivas de seguridad: Se definen los sistemas de cortafuegos (firewall), las tablas de enrutamiento, las zonas desmilitarizadas (DMZ) y la utilización de los servidores tanto Proxy como LDAP.

En el apartado de Configuración de nuestra red: Quedan definidas las tablas cargadas en los dos firewall, he decidido sepáralo en un apartado único por su volumen para facilitar su comprensión.

4. Planificación:

En este apartado se muestra en una tabla a modo resumen con la planificación esperada a lo largo del semestre, esta planificación puede sufrir variaciones por mor de algún retraso de algún punto (posibles complicaciones) o algún desvió del objetivo.

Planificación TFC			
Tareas	Fecha	Trabajo	Observaciones
Tarea 1	27/02/2013-12/03/2013	Borrador del plan de trabajo	Lectura de la documentación: "Presentación de documentos y elaboración de presentaciones" Recopilación de Información
Tarea 2	28/02/2013-19/03/2013	Plan de trabajo (PEC1)	Lectura de documentación: "Redacción de textos científico-técnicos" Crear escenario red para la practica Recopilación y redactar los primeros pasos de configuración y creación de un guión (Índice)
Tarea 3	20/03/2013-31/03/2013	PEC2	Instalación y configuración de MV para el escenario de red,
Tarea 4	01/04/2013-21/04/2013	PEC3	Estudio de requisitos necesarios para el escenario. Configurar aplicaciones y pruebas
Tarea 5	22/04/2013-	PEC4	Ajustes de las aplicaciones y pruebas

	19/05/2013		
Tarea 6	20/05/2013-14/06/2013	Memoria	
Tarea 7	15/06/2013-21/06/2013	Video Presentación	
Tarea 8	25/06/2013-28/06/2013	Tribunales evaluación	

Adjunto un diagrama de Gantt con la planificación anterior para poder apreciar la evolución de los diferentes trabajos de una forma más visual:

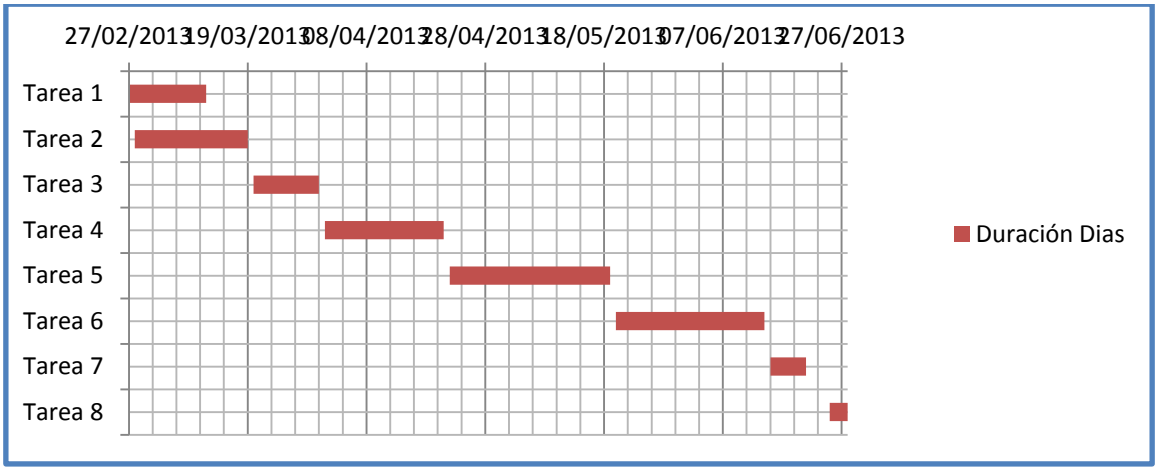

Ilustración 1 Diagrama de Gantt

5. Topología de la Red

5.1.LAN (Local Area Network)

LAN sus siglas significa "*Local Area Network"* es decir Red de área local. Una LAN está formada por un grupo de equipos (Ordenadores, impresoras, fax) que se conectan en un área relativamente pequeña (oficina, hogar, edificio) a través de una red que normalmente tienen la misma tecnología (wifi, cable). Lo que facilita que los usuarios puedan comunicarse entre ellos y que puedan compartir los recursos o dispositivo como una impresora.

Se pueden formar diferentes redes Lan que se unan en una red más grande abarcando un área mayor son las denominadas MAN "Metropolitan Area Netwok" , estas como la palabra indica suele abarcar áreas metropolitanas, estas a su vez se pueden unir formando otras más grandes denominadas WAN "Wide Area Network".

Sólo cabe comentar que estas redes se pueden montar haciendo distintos tipos de topologías ya conocidas como son: En forma de bus, anillo, estrella, malla, árbol. y se escogerá la que más convenga en cada momento aunque algunas veces no podamos poner la situación que más se nos ajuste.

Para ver más gráficamente las diferentes topologías de red que estamos comentando adjunto una tabla con ellas.

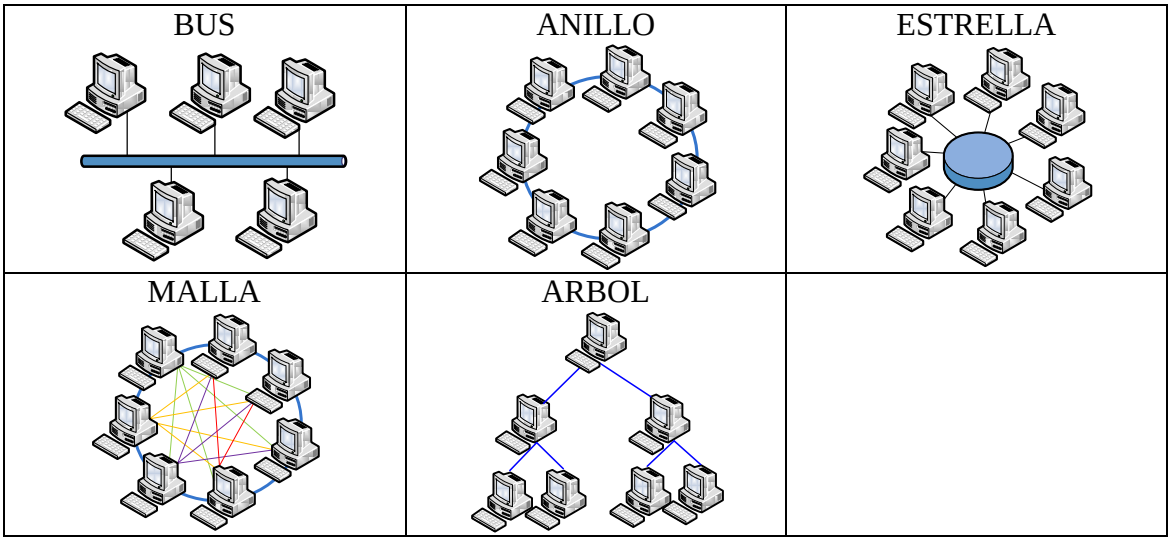

Ilustración 2 Topología de red

Y diferenciar los tres tipos de emisiones que se pueden propagar sobre las distintas redes que acabamos de ver:

- Unicast: La emisión de los paquetes es enviado desde un único equipo a un único destino de la red.
- Multicast: La emisión de los paquetes es enviado desde un único equipo a barios destinatarios de la red.
- Broadcast: La emisión de los paquetes se envían por toda la red recibiéndolos cualquier equipo

5.2.Diseño topología PYME

Nuestra PYME se trata de una empresa que se dedica al sector del desarrollo de aplicaciones interactivas, se trata de crear una estructura que vaya acorde con las especificaciones y posibles crecimientos que pueda tener este tipo de empresas.

Por un lado hablamos de que tendremos que dar soporte a unos 20 equipos entre los que estarán los directivos de la empresa, los empleados donde se incluyen los administradores de la red, los administrativos, los distintos servidores y por último los técnicos que trabajan en desarrollo de aplicaciones.

Para ello se montará un servidor web donde podrán alojar su página web con la presentación de la empresa con sus datos de contacto y formulario con las consultas que puedan tener los clientes, estas consultas se enviarán por correo electrónico.

Se contratará un servidor de correo externo (cada vez más habitual) con dominio de la empresa, al cual tendrán acceso todos los empleados e incluso el servidor web para recibir las consultas de los clientes.

Se instalará un servidor SSH para realizar las conexiones seguras de los trabajadores que puedan estar desplazados fuera de la empresa.

Se montará un servidor intermedio para subir algunos documentos de poca relevancia para que puedan estar accesibles desde el exterior y desde el interior de la empresa.

Se montarán distintos servidores internos para el almacenamiento de información:

- Servidor de documentos: En este servidor es donde los empleados irán almacenado su información de trabajo de todos los días para que todos o casi todos según privilegios tengan acceso a la información.

- Servidores de backups: Es donde se irán almacenando los distintos backup de los equipos de empleados y directivos, más el servidor de documentos.

Además, hay que tener en cuenta la siguiente tabla con protocolos y puertos:

Equipos	Dirección IP	Protocolo	Puerto
Servidor de Correo Externo IMAP	80.19.45.123	TCP	143
Servidor SSH	10.20.0.32	TCP	22
Servidor intermedio	10.20.0.33	TCP	10000
Servidor Backups	10.40.1.11-19	FTP	20
Servidor Documentos	10.40.1.21	TCP (MySQL)	3306
PC Directivos	10.40.1.100-101	HTTPS/HTTP	443/80
PC Admin de Red	10.40.1.200-201	HTTPS/HTTP	443/80
PC Empleados	10.40.1.300	HTTPS/HTTP	443/80

Ilustración 3 Configuración de equipos

En cuanto a los privilegios de acceso a los equipos:

- Todos los empleados tendrán acceso al correo web,
- Todos los empleados tendrán acceso al servidor intermedio.
- Todos los empleados tendrán acceso al servidor de documentos.
- Todos los directivos tendrán acceso al correo web,
- Todos los directivos tendrán acceso al servidor intermedio.
- Todos los directivos tendrán acceso al servidor de documentos.

- Todos los directivos tendrán acceso al servidor SSH por usuario (desde exterior).
- Todos los Admi de Red tendrán acceso al correo web,
- Todos los Admi de Red tendrán acceso al servidor intermedio.
- Todos los Admi de Red tendrán acceso al servidor de documentos.
- Todos los Admi de Red tendrán acceso al servidor SSH por usuario (desde interior).

Las conexiones a la red se regularán por medio de dos ordenadores con Ubuntu y por medio de las IPTables, además estos dos equipos contarán con dos interfaces de Red, más dos switches que admitan enrutar por la capa3, conocidos como switch L3.

Esto nos permitirá crear una topología de red del tipo "estrella" con el inconveniente de que tenemos un punto de fallo importante en la red, ya que si se cae el switch se nos caería toda la red.

A continuación adjunto el esquema general de cómo sería nuestra red y con la configuración de IPs correspondiente, más los firewall oportunos para crear una zona segura llamada DMZ.

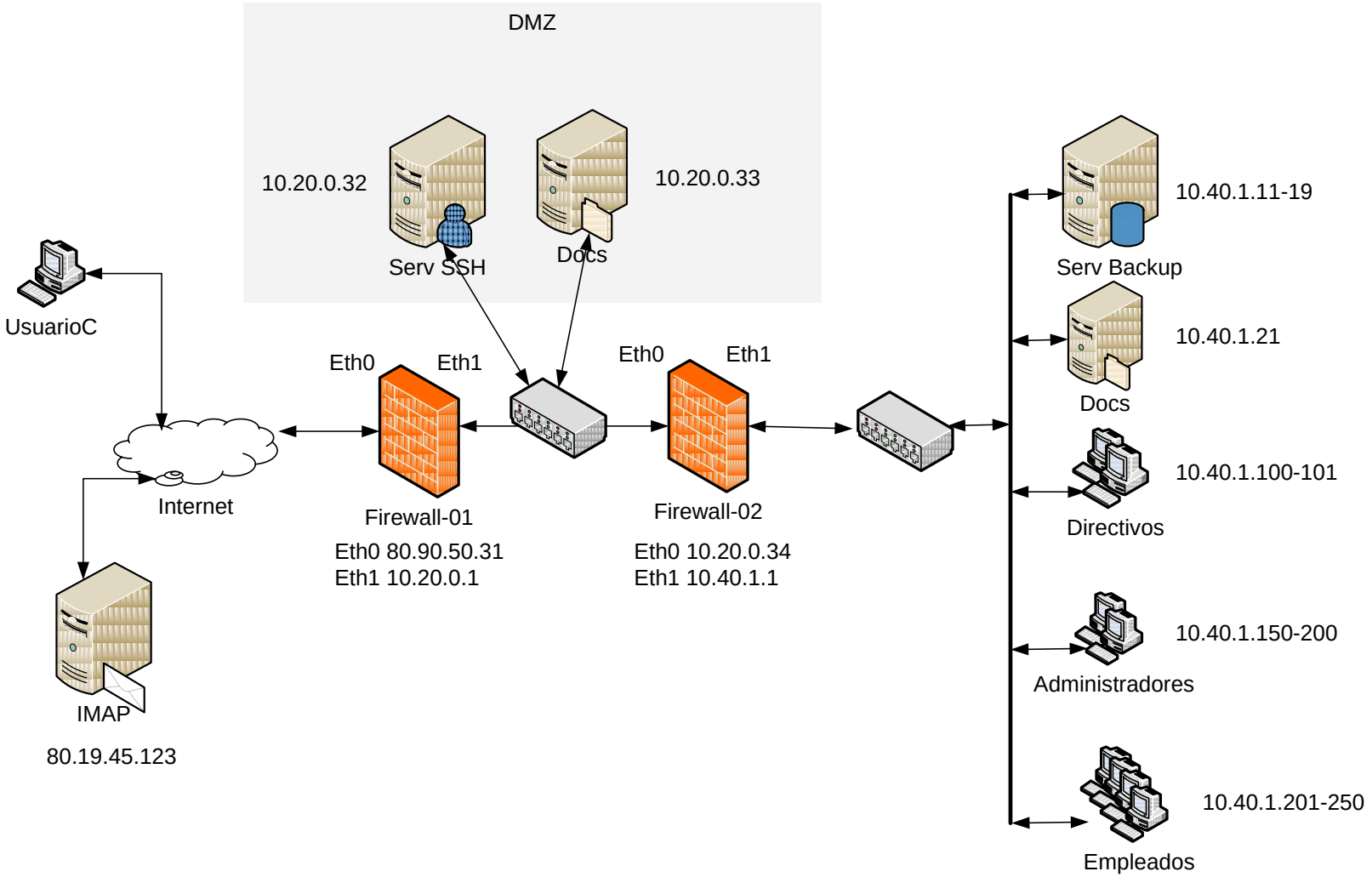

Ilustración 4 Esquema general de nuestra Red

5.2.1. Hardware utilizado:

Para la realización de este proyecto se usa un PC Intel Core2 Duo CPU 2.2Ghz y 4 GB de Ram

5.2.2. Software utilizado:

Para la realización de este proyecto se utiliza MV Virtual Box, que corre sobre el entorno de Windows.

Estoy utilizando las distribuciones de Ubuntu para la simulación de equipos en la red y también se utilizaré la distribución de Bactrack, para realizar comprobaciones de seguridad en la red.

6. Seguridad e instalación de las estaciones de trabajo

6.1.Seguridad en estación de trabajo:

Uno de los puntos importantes para dar seguridad en las estaciones de trabajo es cambiar la configuración que trae por defecto cuando se instala ya que puede traer cuentas por defecto sin password, servicios abiertos, carpetas o ficheros compartidos, etc., por ello es importante seleccionar el S.O que más se ajuste a las necesidades del usuario y configurarlo acorde a los accesos y usos.

Otro de los puntos para mantener la seguridad en las estaciones de trabajo es lo que se conoce como "buenas prácticas", que corresponden a acciones, medidas y aspectos que hay que tener en cuenta para garantizar la seguridad y evitar que los usuarios se conviertan en víctimas de amenazas externas (virus, troyanos, malwares, gusanos..)

- Por ello es importante descargar y acceder a contenido que sean oficiales para evitar riesgos de infección.
- Tener cuidado con los correos recibidos, si no son de confianza o solicita datos de una cierta relevancia como datos bancarios (Phishing) .
- Links que nos redireccionan a otra pagina o descarga.

Tener siempre en la medida de lo posible actualizados los S.O y aplicaciones de seguridad (parches de seguridad, Firewall, etc).

Gran parte de las amenazas externas están pensadas para otras plataformas como pueden ser Windows o Mac ya que tienen un número más elevado de usuarios que Linux y además en Linux no se les permite escribir de forma promiscua en una dirección de memoria, algo que Windows en las últimas versiones ha empezado a implementar, las pocas amenazas que puedan afectar a Linux suelen venir por aplicaciones comunes a los

distintos Sistemas operativos como pueden ser los navegadores con las cookis, ataques Phishing, etc.)

6.2.Seguridad Física:

La seguridad física es un punto importante para proteger nuestra red e información, ya que muchas veces nos olvidamos de que los ataques pueden venir de nuestro propio entorno, supongamos que una persona no tiene acceso a un servidor de almacenamiento vía red, pero si puede colarse en la sala donde está físicamente el equipo, esta persona podría copiar o llevarse el disco duro con toda la información, por ello aunque tomemos todas las medidas oportunas en cuanto a restricción lógicas de acceso vía red si descuidamos las vulnerabilidades físicas estaremos dejando un agujero en nuestra seguridad.

Otro de los puntos importantes para salvaguardar nuestros equipos es el que atañe a la energía, es importante instalar algunos SAIs (Sistema de alimentación Ininterrumpida) que nos permitirá en caso de caída de corriente mantener los equipos encendidos durante un periodo de tiempo corto para guardar los trabajos que se estén utilizando en ese momento, además la mayoría de los SAIs disponen de un sistema para proteger a los equipos de un pico o subida de tensión.

6.3.Copias de Seguridad (Backup):

Ante la pérdida de información, borrados de ficheros de forma errónea o como seguridad, es importante realizar backups de seguridad.
Es importante que las copias de seguridad no se guarden en la misma ubicación donde se encuentras los equipos, ya que en caso de algún accidente; inundación, incendio, toda la información se perdería, por ello se aconseja tener más de una copia y en distintas ubicaciones.
Otro detalle a tener en cuenta en las copias de seguridad es la nomenclatura de dichas copias, ya que es importante que tengan un código identificativo pero no que se sepa a simple vista a que copia pertenece a que usuario o PC, de esta forma solo las personas autorizadas podrán saber que copia es la que necesitan y a quien corresponde.

Para este proyecto se ha utilizado la herramienta Deja Dup, (en cualquier versión posterior al 11.10 no es necesario instalarla), para su instalación ejecutamos:

```
$ sudo apt-get install deja-dup
[sudo] password for usuarioa:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  duplicity gvfs-fuse libexempi3 librsync1 nautilus nautilus-data
  python-boto
Paquetes sugeridos:
  ncftp eog gnome-app-install
Se instalarán los siguientes paquetes NUEVOS:
  deja-dup duplicity gvfs-fuse libexempi3 librsync1 nautilus nautilus-data
  python-boto
0 actualizados, 8 se instalarán, 0 para eliminar y 144 no actualizados.
Necesito descargar 2936kB de archivos.
Se utilizarán 25,8MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]?
```

Ilustración 5 Deja Dup - Instalación

Una vez instalado nos vamos a apliaciones/accesorios y desde ahí nos dejara dos opciones:
Restaurar: Si ya disponemos de una copia anterior, procedemos a su volcado.
Respaldar: Nos permite crear una copia de seguridad.

Ilustración 6 Deja Dup - Opciones

Nos aparece una pestaña de opciones, que por un lado destacamos dos cosas la posibilidad de cifrar los archivos (importante cifrarlo para su seguridad) y posibilidad de hacer el volcado sobre un servidor de Backups.

Ilustración 7 Deja Dup - Cifrado

Opciones de servidor: Importante como ya veremos en otro apartado el tipo de conexión que se realiza. FTP, SSH, HTTPs.

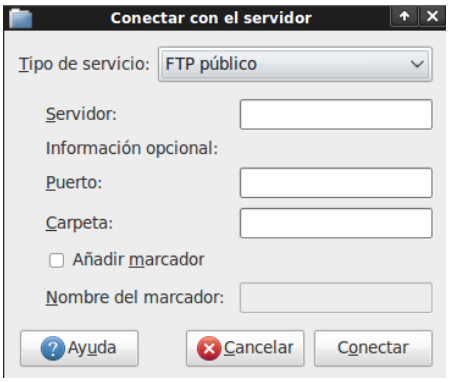

Ilustración 8 Deja Dup - Protocolo conexión

Otro de los factores a tener en cuenta a la hora de realizar un backup, es la frecuencia con la que se va a realizar ya que si lo hacemos una vez al año es posible que a los seis meses de la copia si se quiere recuperar la información ya no este disponible porque no se ha copiado la información, pero si lo configuramos para que nos haga la copia cada hora, estaremos saturando el maquina del usuario, lo recomendable es hacer copias una vez al final de día o de dos días, para evitar en caso de que sea necesario un volcado perder la mínima información posible.

Esta aplicación nos permite realizar diferentes tipos de copias de seguridad en función de la cantidad de archivos que se quieren guardar, estas son:

- Copia de seguridad Total: Es una copia de seguridad normal, es decir copia todos los archivos y directorios.

- Copia incremental: Es este caso se copian sólo los archivos que se han cambiado desde la última copia de seguridad realizada. Es decir, se realiza la copia solo de los ficheros que han cambiado o que son nuevos desde la última copia incremental
  En caso de querer hacer un volcado de información (recuperar la información) sería necesario disponer de la copia total más las incrementales.

- Copia diferencial: Se realiza una copia de todos los archivos que han cambiado desd la última copia de seguridad total que se haya realizado, Una de las diferencias de la anterior es que una copia diferencial anula a la copia diferencial anterior lo que nos permite ocupar menos espacio en disco

6.4.Sistemas Operativos:

En este apartado veremos la descripción de los sistemas operativos que más utilizan los usuarios para este tipo de redes.

- **Ubuntu**: Es un sistema de fácil acceso y pensado para usuarios que no tengan un nivel alto de conocimientos sobre informática. Este sistema dispone de la herramienta SUDO (Switch User Do) , con lo que se evita crear un superusuario administrador.

- **BackTrack**: Es un sistema dedicado para testear y probar la seguridad en una red y evitar posibles intrusiones, dispone de herramientas ya instaladas para los escaneos de puertos, búsquedas de vulnerabilidades, sniffers, herramientas de seguridad de Wireless, etc.

- **Fedora**: Es un sistema estable que se mantiene con el apoyo de diferentes personalidades como la comunidad internacional de ingenieros, diseñadores gráficos y por los propios usuarios. Cabe destacar de este sistema es que las actualizaciones se realizan sobre las fuentes originales en lugar de aplicar parches, así se aseguran que estas modificaciones estén disponibles para todas las variantes de Linux.

Incluyo dos tablas una con las distintas distribuciones donde se pueden ver los recursos hardware mínimos necesarios y otra con las aplicaciones de seguridad que aplican a cada uno de ellos.

REQUISITOS HARDWARE					
Distribución	**Tamaño (MB)**	**RAM (MB)**	**X86**	**X86-64**	**Ia64**
Debian/Ubuntu	700	256	Sí	Sí	Sí
Fedora	683	256 a 1024	Sí	Sí	No
Gentoo	100	128	Sí	Sí	Sí
Mandriva	700	128 a 768	Desconocido	Desconocido	Desconocido
SUSE Linux	Desconocido	Desconocido	Hardware NX	No	No
Tor-ramdisk	5	256	Sí	Sí	No
BackTrack	1.910	Desconocido	Sí	Sí	No

Ilustración 9 Requisitos Hardware

Cada una de las distribuciones implementa distintas aplicaciones para la seguridad de los equipos, parches de kernel para la protección y seguridad ante posibles bug o ataques que se puedan realizar a estos.

FUNCIONES DE SEGURIDAD					
Distribución	**Comprobación del buffer en tiempo de compilado**	**Control de acceso obligatorio**	**Software ejecutable para protección**	**grsecurity**	**RSBAC**
Debian/Ubuntu	Sí	AppArmor	PaX	Opcional	Opcional
Fedora	Sí	SELinux	Exec Shield	No	No
Gentoo	Opcional	SELinux	PaX	Opcional	Opcional
Mandriva	Desconocido	AppArmor	Desconocido	Desconocido	Sí
SUSE Linux	Sí	AppArmor	Hardware NX	No	No
Tor-ramdisk	Desconocido	Desconocido	PaX	Sí	No

Ilustración 10 Funciones de Seguridad

6.5.Conexiones seguras SSH

Secure Shell, sería interprete de comandos seguros, esto es un protocolo de comunicación que nos permite crear un túnel de seguridad a través de una CLI (Command Line Interface), es decir la finalizad de este túnel es encriptar todas las conexiones que se realizan indiferente que sean del tipo http o https y conectarnos desde cualquier ordenador que esté fuera de nuestra red (internet) a uno de nuestros equipos que se encuentre dentro de nuestra red (Lan) y que esta conexión se realice de forma segura.

Este protocolo es parecido a Telnet pero con la diferencia de que en las conexiones SSH la información que se transmite se realiza codificada con lo que es mucho más segura.

6.5.1. Instalación:

Para comenzar la instalación hay que lanzar el siguiente comando: apt-get install ssh en el equipo server

```
root@vm-server:/etc/apache2/sites-available# apt-get install ssh
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  openssh-server
Paquetes sugeridos:
  ssh-askpass rssh molly-guard openssh-blacklist openssh-blacklist-extra
Se instalarán los siguientes paquetes NUEVOS:
  openssh-server ssh
0 actualizados, 2 se instalarán, 0 para eliminar y 137 no actualizados.
Necesito descargar 286kB de archivos.
Se utilizarán 827kB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? S
AVISO: ¡No se han podido autenticar los siguientes paquetes!
  openssh-server ssh
¿Instalar estos paquetes sin verificación [s/N]? s
```

Ilustración 11 SSH - Instalación

Reinicio el servidor con el siguiente comando
/etc/init.d/ssh restart

Comprobamos la configuración del equipo con el cual queremos acceder:

```
root@vm-server:/home/uocseg# ifconfig
eth0      Link encap:Ethernet  direcciónHW 08:00:27:0f:e1:b1
          Direc. inet:10.20.0.32  Difus.:10.20.0.255  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:fe0f:e1b1/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
          Paquetes RX:610 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:754 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:78437 (78.4 KB)  TX bytes:79331 (79.3 KB)

lo        Link encap:Bucle local
          Direc. inet:127.0.0.1  Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Anfitrión
          ACTIVO BUCLE FUNCIONANDO  MTU:16436  Métrica:1
          Paquetes RX:17 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:17 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:0
          Bytes RX:1559 (1.5 KB)  TX bytes:1559 (1.5 KB)

root@vm-server:/home/uocseg#
```

Ilustración 12 SSH - Configuración equipo

6.5.2. Tipos de conexiones SSH:

- Conexión con password:

Compruebo que me permite por ejemplo al usuarioC conectarse desde el equipo exterior a nuestro servidor SSH que está en la zona DMZ. Para ello es necesario que previamente el administrador haya creado a dicho usuario en el servidor.

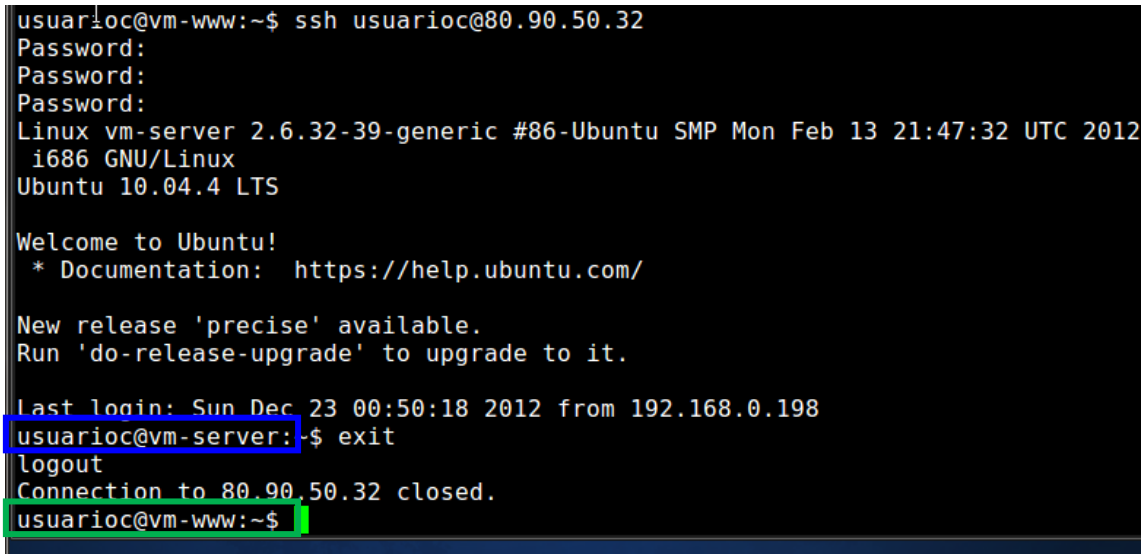

Ilustración 13 SSH - Acceso usuarios

Como podemos ver en esta imagen nos hemos conectado al servidor donde nos ha solicitado la contraseña (he errado dos intentos), una vez dentro del server, podríamos acceder a las diferentes carpetas que tanga permiso el usuarioc o a aplicaciones que estén instaladas.
Para salir es tan sencillo como realizar un exit, de esta manera cerramos la conexión, es importante cerrar siempre las conexiones para poder evitar posibles ataques.

- Conexión con clave pública y privada RSA:

Para dar más seguridad a nuestras conexiones y evitar que si hay algún usuario que este esnifando la red y capture nuestra transmisión, vamos a crear un certificado RSA que será único para nuestro usuarioc, con lo que haremos que nuestros datos sean casi imposible de descifrar.

Para ello primero tengo que crear un certificado que lo creamos con el siguiente comando:ssh-keygen –t rsa –b 4096

Una vez lanzado el comando se verá por pantalla la generación del mismo y las carpetas donde se ha guardado nuestras claves.

```
usuarioc@vm-www:/home/uocseg$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/usuarioc/.ssh/id_rsa):
Created directory '/home/usuarioc/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/usuarioc/.ssh/id_rsa.
Your public key has been saved in /home/usuarioc/.ssh/id_rsa.pub.
The key fingerprint is:
77:39:18:b9:11:7b:b9:0d:76:d0:6d:81:23:a7:62:46 usuarioc@vm-www
The key's randomart image is:
+--[ RSA 4096]----+
|          . ...o.|
|         E = *. o|
|        . = 0 o. |
|         + 0 *   |
|        S = = .  |
|         . . .   |
|                 |
|                 |
|                 |
+-----------------+
usuarioc@vm-www:/home/uocseg$
```

Ilustración 14 SSH - Generar claves

Una vez realizado el certificado comprobamos el contenido de nuestra clave pública lo que se conoce como fingerprint donde vemos que corresponde con el usuarioc@vm-www, para ello lanzamos el siguiente comando: more /home/usuarioc/.ssh/id_rsa.pub

```
usuarioc@vm-www:/etc/ssh$ more /home/usuarioc/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAgEAn8PkYDL+0tNlqL/iEZUmLcxyRag/kYMK55rtmRcpb
NTE9l1gndVptUh88NsBcMcMKeUHnZx1XIwMeygVFX+hiGIwGNFyKgHL7qkhMYTjcsals7nQpAV/as
/xngcppGhGI088zA+ZjflxJi4shOMiOfGy33b07vDbn3aySaqkSSTCrL/oHhTWKvOy8p5Znm/ztJd
ItTu6RQFB8Ff0SQrp+Fy9eUiLgFzPy5ERPAhnK1Np3Kn3dzuRiR+FM3GbfCoJCyk7dHxtW38rcAZT
suZ+eBltnCU6zeoDPapjkXH/STSLvFd2Q30ge3wRIRx04mBAM5JFb1/FlWQwoSlqfsz21iHXfZ5/l
DXI5J4chvEZWWUghMioHnlOiwbYnX2fvoUP3F/6E/K9A1aaRTQJeiPjOUPbRs7iqbh1mN9EqPOjhn
0BTV15jUm8idBA3Z71BrIRd81HW3Penz/YrsVc23SpFaxgFb6a2gwh5w/jg/ucNRi/e7jUfnNhhM/
4bReLMqCw3P8a5gkbpQC+yDo3/JU/NIEFaNOjTtS9wGTlnWIJfCAtquhuw8wW78VAZPcs9n3TYiqb
KoTqdCnVY3ll/z3fSZeBTGI7g5EmkG9WjgZJ9tLEUZc45opTvwzNYthlbXWKbay5/dyVSuE4L8rFR
SuoCofc/Kgu8dDUGzO9NxwMbPE= usuarioc@vm-www
usuarioc@vm-www:/etc/ssh$
```

Ilustración 15 SSH - Comprobar claves

Una vez comprobado que la calve pública corresponde creamos las carpetas para el usuarioc (/home/usuarioc) donde guardaremos la key publicas

usuarioc@vm-Server $> mkdir .ssh

usuarioc@vm-Server $> cd .ssh

usuarioc@vm-Server $> mkdir .ssh/authorized_keys

Ahora que ya tenemos acceso al servidor SSH copiamos desde el equipo externo la clave pública al servidor en la carpeta de usuarios autorizados "authorized_Keys", otra manera sería que el administrador dispusiera ya de la clave pública, por ejemplo: enviada por correo electrónico y fuese él quien copiase la clave en las carpetas correspondientes.

De no ser así el usuario con un sencillo comando podría copiar su clave pública en la carpeta de claves autorizadas.

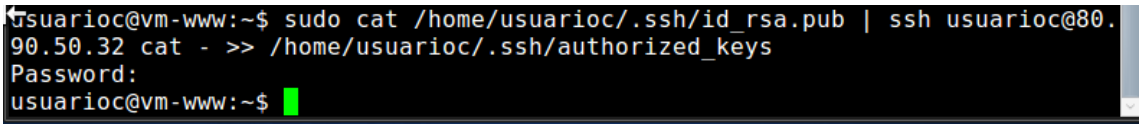

Ilustración 16 SSH - Copiar Claves

Una vez copiadas las claves publicas, podemos dar más seguridad al servidor SSH permitiendo solo las conexiones con autentificaion con certificado RSA, es importante que el usuarioc no borre ni modifique su clave pribada ya que será entre estas dos (RSA_pub y RSA_pri) las que identifiquen al usuarioc.

6.5.3. Configuración conexiones SSH:

Es muy importante la configuración de las conexiones SSH para evitar permitir acceso a usuarios no autorizados. Para ello lo primero que tenemos que hacer en modificar el fichero sshd_config con el siguiente comando: sudo nano /etc/ssh/sshd_config

Una vez dentro modificamos los parámetro que indico a continuación según nos interese, el ejemplo que he realizado seria para una autentificación por RSA sin que nos pida cada vez que intentemos conectarnos al servidor SSH la passwoord:

PermitRootLogin no ➔No permitimos el acceso al usuario root

Para permitir solo el acceso con key hay que deshabilitar la password

PubkeyAuthentication yes

AuthorizedKeysFile .ssh/authorized_keys

PasswordAuthentication no ➔Deshabilitar la password

Añadir la línea siguiente para permitir solo a los usuarios que pertenezca al grupo "sysadmin":

AllowGroups sysadmin, ➔En este punto podemos añadir los diferentes grupos de usuarios que tengamos con sus correspondiente privilegios.

Como podemos ver en la siguiente imagen.

```
PermitRootLogin no
StrictModes yes
AllowGroups sysadmin

RSAAuthentication yes
PubkeyAuthentication yes
#AuthorizedKeysFile     %h/.ssh/authorized_keys

# Don't read the user's ~/.rhosts and ~/.shosts files
IgnoreRhosts yes
# For this to work you will also need host keys in /etc/ssh_known_hosts
RhostsRSAAuthentication no
# similar for protocol version 2
HostbasedAuthentication no
# Uncomment if you don't trust ~/.ssh/known_hosts for RhostsRSAAuthentication
#IgnoreUserKnownHosts yes

# To enable empty passwords, change to yes (NOT RECOMMENDED)
PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no

# Change to no to disable tunnelled clear text passwords
PasswordAuthentication no
```

Ilustración 17 SSH - Fichero de configuración

Una vez realizados los pasos anteriones no que da por último reiniciar el servidor SSH con el siguiente comando: sudo /etc/init.d/ssh restart

Y con ello ya conseguimos disponer del acceso a nuestros equipos con una conexión segura gracias al servidor SSH.

```
usuarioc@vm-www:~$ ssh usuarioc@80.90.50.32
The authenticity of host '80.90.50.32 (80.90.50.32)' can't be established.
RSA key fingerprint is 2d:7e:26:ea:7d:80:a2:2c:bb:63:63:64:99:c5:7f:bd.
Are you sure you want to continue connecting (yes/no)?
```

Ilustración 18 SSH - Conexión Autentificada

A continuación describimos las opciones de usuarios y grupos más las diferentes configuraciones de acceso al servidor SSH, con estas dos tablas podríamos modifcar según las modificaciones de la red que nos puedan ir surgiendo (nuevos usuarios, nuevos grupos, número de intentos de acceso a nuestro servidor SSH, etc).

Opciones de usuarios y grupos		
Opción	**Descripción**	**Valor por defecto**
AllowGroups	Lista de nombres de grupos, hay que separarlos por espacios cuyos miembros sean como grupo primario o grupo secundario, tienen acceso permitido al sistema mediante SSH, permite el uso de caracteres comodín * e ?	Todos los grupos
AllowUsers	Lista los nombres de grupos, separados por espacios,	Todos los usuarios

	cuyo acceso al sistema SSH esta permitido. Puede escribirse como usuarioC@192.168.0.197 comprobando entonces tanto el nombre del usuario como la dirección del PC desde el que se conecta, permite el uso de comodines * e ?	
DenyGroups	Lista de nombres de grupos, separados por espacios cuyos miembros sean tando del grupo primario como del secundario, no tiene acceso al sistema median SSH, permite el uso de los comodines * e ?	Ningún grupo
DenyUser	Lista de nombres de usuarios separados por espación cuyo acceso al sistema no está permitido por SSH. Puede tomar la forma de usuarioC@192.168.0.197 comprobando entonces tanto el nombre del usuario como la dirección del PC desde el que intenta conectarse , permite el uso de comodines * e ?	Ningún usuario

Ilustración 19 SSH - Opciones de usuarios y grupos

Opciones de configuración de acceso		
Opción	**Descripción**	**Valor por defecto**
AuthorizedKeysFile	Fichero con las claves públicas usadas para autentificación.	/.ssh/authorized_keys
ChallengeResponseAuthentication	indica si el intercambio de respuestas de autenticación es permitido	YES
Cipher	Indica los cifrados permitidos por el protolo	Todos
GSSAPIAuthentication	Especifica si la autenticación está basada en GSSAPI es permitida	NO
GSSAPICleanupCredentials	Especifica si lsa credenciales son automáticamente destruidas cuando termina la sesión	YES
HostbasedAuthentication	Autoriza el acceso mediante clave publica de usuarios de los ordenadores indicados en rhosts o en /etc/hosts.equiv	NO
HostKey	Especifica el fichero que contiene la clave privada del servidor	
IgnoreRhosts	Deniega el uso de los ficheros .rhosts y .shosts en el acceso remoto	YES
IgnoreUserKnowHost	Deniega el uso del fichero /.shh/know:hosts para encontrar los ordenadores conocidos	NO
LoginGraceTime	Tiempo en segundos antes de que se cierre la sesión de autenticación	120 segundos
LogLevel	Información que se escribirá en los accesos, sus valores son: Quiet, Fatal, Error, Info, Debug	Info
MaxAuthTries	Número máximo de intentos de autentificación por conexión	6
MaxStartups	Número máximo de conexiones	10

	simultaneas en comunicación	
PasswordAuthentication	Permite la autenticación mediante contraseñas	YES
PermitEmptyPasswords	Permite el acceso a usuarios sin contraseña	NO
PermitRootLogin	Permite el acceso de root mediante SSH	YES
PubKeyAuthentication	Permite la authenticacion mediante clave pública	YES
RSAAuthentication	Permite mediante RSA	YES
UseLogin	Indica si se utiliza login para comprobar el acceso de los usuarios	NO
UsePAM	Indica si se utiliza PAM para comprobar el acceso de los usuarios	NO

Ilustración 20 SSH - Opciones de configuración de Acceso

7. Ataques externos al sistema y detección

7.1. Fases de intrusión por parte de un atacante:

Los atacantes siguen unas pautas para poder llevar a cabo su ataque hacia otros equipos, estas son:

Fase de vigilancia: En esta fase el atacante sniffa los paquetes de las conexiones e intenta lanzar comandos de exploración de puertos e ips, para ir comprobando las posibles vulnerabilidades que pueda tener la red (equipos que se utilizan en la red, puertos abiertos), a la que se quiere atacar.

Fase de exploración de servicios: Una vez comprobado los equipos de la red, intentará acceder a ellos para obtener privilegios y ponerse como administrador para poder controlar los equipos y así impedir que lo puedan detectar y logear de los equipos fácilmente.

Fase de ocultación de huellas: En esta fase consiste en que una vez que ya has tomado posesión de los equipos se tratará en la mayor medida posible eliminar todo tipo de huella que se haya dejado, desde eliminar los registros (log) a ocultar la entrada al sistema de Red.

Fase de extracción de la información: En este último punto el atacante ya puede tener acceso a información privada con un mínimo riego de ser detectado.

7.2. Detección de intrusiones (ID):

La detección de intrusiones consiste como la propia palabra indica en detectar e identificar las posibles actividades ilícitas contra los recursos de nuestra red. Los primeros sistemas se crearon con la finalidad de hacer auditorias sobre el tiempo que dedicaban los operadores a usar los sistemas, fue la empresa norteameriaca Bell Telephone System quien creó un grupo de desarrolladores para analizar los ordenadores de telefónia. A partir de los años 70 se empieza a invertir en la investigación sobre políticas de seguridad y es aquí cuando surge un nuevo termino denominado Sistemas

de confianza, estos consisten en catalogar que sistemas emplean suficientes recursos (software y hardware) para poder procesar de forma simultánea información confidencial y privada.

Para conseguir que los sistemas de detección de intrusiones funcionen hay que dividirlos en partes:

Por un lado están los componentes básicos de un sistema de detección de intrusiones que son:

- **Recolectores de información**: Conocidos como sensores, es el responsable de la recogida de la información de los equipos monitorizados, se dividen en: basados en equipos, basados en Red, basados en aplicación.

- **Procesadores de eventos o analizadores**: Conforman el núcleo central del sistema de detección, se encargan de tratar la información recogida por los sensores para poder inferir posibles intrusiones. Se dividen en: detección de usos indebidos, detección de anomalías.

- **Unidades de respuesta**: Se encargan de iniciar acciones de respuesta en el momento en que se detecta un ataque o intrusión, se dividen en: automáticas (respuesta activa) o requerir interacción humana (respuesta pasiva).

- **Elementos de almacenamiento**: Permiten el almacenamiento de la información recogida por los distintos sensores del sistema de detección, este almacenamiento se puede enviar a dispositivos secundarios dependiendo de si su análisis se realizará en un período a corto o largo plazo.

Y por otro lado cabe destacar que hay diferentes tipos de sensores que se utilizan en un sistema de detección de intrusiones:

- **Sensores basados en equipo**: Se encarga de analizar y recoger información de eventos a nivel de sistema operativo, (ejemplo, intentos de conexión o llamadas al sistema). un sw podría ser CyberCop, que se utilizaría como apoyo a otros antivirus y además les permite observar al usuario y su actividad de proceso por si intenta acceder a sitios restringidos.

- **Sensores basados en red:** Se encargan de recoger información de eventos sucedidos a nivel de tráfico de red, (analiza las cabeceras IP de los datagramas que pasan por la interfaz de red). un sw podría ser CMDS. Estos se usan es sistemas cambiantes donde cada día se actualizan con nuevos posibles ataques.

- **Sensores basados en aplicación**: Se encarga de recibir la información de las aplicaciones que se están ejecutando, y podrían ser considerados como un caso especial de sensores basados en equipos, un sw podría ser OKENA con su Storm watch.

7.3.Ventajas y desventajas del sistema ID

Ventajas:

- Ayuda a descubrir donde se producen las amenazas en nuestra red.
- Nos dan información sobre el tráfico malicioso que pueda haber en nuestra red.
- Permite detectar intrusiones desconocidas y abordarlas en tiempo real.
- Permiten ser instaladas en cualquier maquina sin que esto suponga un alto coste para ella (consumo bajo de recursos).
- Almacenamiento de alertas y avisos en ficheros log, para su posterior revisión.

Desventajas:

- Hay una alta posibilidad de que se produzcan falsas alarmas.
- Un alto mantenimiento de gestión y control para revisar los logs que produzcan las diferentes máquinas.
- Según su configuración, durante el proceso de aprendizaje o de adaptación al los recursos consumidos por un usuario en su PC pueden hacer que generen falsas alarmas o por el contrario, no notar un comportamiento anómalo.

7.4. Analizador de tráfico SNORT

Snort es una aplicación multiplataforma pensada para analizar el tráfico que pasa a través de la red, lo que se conoce como sniff, este tráfico lo compara con unos patrones que previamente se han definido y permite generar las alarmas correspondientes, es decir.

Esta aplicación permite defenderse de ataques externos pero primero hay que identificarlos, estos ataques suelen ser provenientes de una red IP, lo que nos permite de forma forzosa que sean detectables desde cualquier ordenador que esté a la entrada de la red que queremos proteger.

Los programas de detección de intrusiones (NIDS) como SNORT permite examinar todos y cada uno de los paquetes que recibimos en nuestros interfaces de red. Una vez que reciben el paquete lo analizan para ver sí este cumple con alguna de las reglas que previamente se han establecido por el administrador de la red. Si corresponde con alguna de estas reglas snort se encarga de realizar la acción correspondiente como guardar un log del tráfico, copiar alguno de los paquetes recibidos o mostrar una alarma según corresponda.

7.4.1. Funciones de SNORT

- Es capaz de escuchar de forma promiscua en un interfaz de red que se le indique, lo que permite escuchar tanto a los paquetes que van dirigidos a dicho interfaz como a los que van a otros interfaces y mostrar o guardar de forma local en un disco dicho tráfico. En este modo se asemeja al programa tcpdump ya que comparte el sistema de registro en el disco de tráfico.

- Escuchar en modo promiscuo en un interfaz de red como en el apartado anterior pero atendiendo a una serie de reglas más avanzadas que nos permitirá analizar y depurar tráfico y además protocolos de red.

- También se puede usar como herramienta para el análisis avanzado de tráfico en la red, obviamente en modo promiscuo y usando una base de reglas sofisticada para detectar y avisar sobre el tráfico no deseado que esté circulando por nuestra red.

SNORT es capaz de analizar los protocolos básicos como: IP, TCP, UDP, ICMP y otros como el Ethernet, arp, fddi, rarp, lat, decnet, moprc y mopdl. para poder analizar todos estos protocolos snort lleva unos módulos que le permiten:

- Recomponer tráfico que se haya desfragmentado.
- Recomponer cadenas de caracteres enviadas a través de servicios como por ejemplo un telnet, ftp, smtp, http, pop3 e imap, por lo que le permite analizar el tráfico a nivel de aplicación y no solo quedarse en el nivel de red.
- Permite entender el tráfico http con su extensión de caracteres ejemplo: %20.
- Permite detectar escaneos remotos en busca de puertos abiertos.

7.4.2. Especificación de SNORT

Snort es una aplicación bajo la licencia de GNU, por lo que su utilización es gratuita, además está escrita en C, lo que le permite funcionar en múltiples plataformas como Linux, Windows, MacOS entre otros. otra de sus características fundamentales es que no requiere de gran capacidad de recursos tanto por el hardware como por consumo de recursos (Ram, CPU) para su funcionamiento a diferencia de otras aplicaciones que hay en el mercado

7.4.3. Instalación snort

Snort es una aplicación que viene ya instalada en casi todas las versiones de Unix, en caso de que no estuviese instalada con el siguiente comando se podría instalar facilmente: apt-get install snort

```
root@vm-fw:/home/uocseg# apt-get install snort
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  libmysqlclient16 libprelude2 mysql-common oinkmaster snort-common
  snort-common-libraries snort-rules-default
Paquetes sugeridos:
  snort-doc
Se instalarán los siguientes paquetes NUEVOS:
  libmysqlclient16 libprelude2 mysql-common oinkmaster snort snort-common
  snort-common-libraries snort-rules-default
0 actualizados, 8 se instalarán, 0 para eliminar y 137 no actualizados.
Necesito descargar 4488kB de archivos.
Se utilizarán 14,7MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]?
```

Ilustración 21 SNORT - Instalación

Configuramos los dos puertos por los que va a escuchar snort datos procedentes de la red externa eth0 y de nuestra red interna eth1

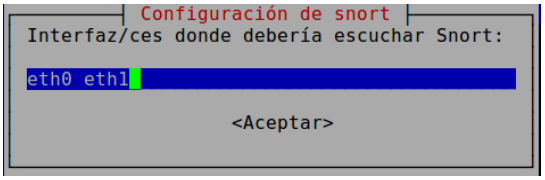

Ilustración 22 SNORT - Interfaz

7.4.4. Reglas de SNORT y su configuración

Como ya hemos mencionado antes las reglas se pueden crear de forma manual, según nos interesen o ya existen unas por defecto (lo cual no impide añadir a estas últimas las nuestras) para la detección de intrusiones, estas las podemos descargas de la página oficial WWW.snort.org o de otras páginas no oficiales que se van actualizando con información de hackeos y sobre los distintos patrones de tráfico utilizados en herramientas NIDS.

En cualquier de los casos las reglas que nos descarguemos por defectos suelen incluir entre otras muchas estas:

- Escaneos de puertos abiertos: Tanto si en el escaneo se llega a establecer la conexión o solo responde con la información de si el puerto está abierto o no.

- Permite detectar ataques del tipo GCi de páginas web con pequeños script desde el propio navegador.

- Detecta análisis de escaneos para identificar el tipo de sistema operativo que se está utilizando en una de nuestras máquinas. Este punto es fundamental para un hacker ya que saber a qué sistema operativo se enfrenta le permitirá buscar u orientar su ataque de una manera u otra para identificar las vulnerabilidades de dicho sistema operativo.

Una vez descargado e instalado nos podemos crear un fichero con nuestras reglas locales a mayores de las que hemos descargado.

Este fichero que llamaremos "local.rules" lo guardamos en la siguiente ruta: etc/snort/rules/local.rules

Dentro de este fichero se podrán crear las reglas que más nos interesen, siguiendo la normativa que explicamos a continuación con un ejemplo básico, el cual se podrá moldear según nuestras necesidades.

alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"Escaneo TCP connect";sid:1000001;rev1;)

Estructura de las reglas: Las reglas se dividen principalmente en la cabecera y opciones, la cabecera será donde se defina el tipo de aviso queremos que nos notifique más los direccionamientos y protocolos que queremos que cumpla, para las opciones serán los mensajes de texto que queremos que nos aparezca y número de revisiones de la regla o definir algunos de los parámetros de las tramas.

CABECERA

- ***Acción de la regla:***
  - **alert** genera una alerta usando el método de alerta seleccionado y almaceno el log.
  - **log** archiva el log del paquete
  - **pass** ignora el paquete
  - **activate** activa la alerta y llama a una regla dinámica
  - **dynamic** cuando es llamada por una regla **activate** se pone en funcionamiento
- ***Protocolo:*** se incluyen los protocolos que queremos analizar para esa regla: tcp, udp, etc.
- ***Direccion IP origen:*** Puede ser una ip numérica 192.168.0.2, o por variables predefinidas del tipo $EXTERNAL_NET *(toda la red), HOME_NET (toda nuestra red), any (para indicar cualquier dirección ip)*
- ***Puerto IP origen:*** any *(cualquiera),* 22 (ssh), 80 (http), 443 (https), etc.

- ***Dirección de la operación:*** *puede ser ->, <-, según la direccionalidad que queramos darle. entrada salida.*
- ***Direccion IP destino:*** corresponde al mismo caso que dirección IP origen.
- ***Puerto IP destino:*** corresponde al mismo caso que puerto IP origen

OPCIONES

- ***Mensaje:*** msg es el mensaje que queremos que nos muestre cuando salte una de las acciones que hemos definido
- ***Opciones:*** tiene una gran variedad de opciones que iremos añadiendo las más comunes son:
  - ***flags:*** Establecen los tipos de flags como de respuesta, ejemplo ACK.
  - ***content:*** Busca una palabra o frase dentro de trama.
  - ***sid:*** permite identificar las reglas de forma única, para las locales se utilizan a partir de 1000000.
  - ***rev:*** Es la versión de revisión que se ha realizado a esa regla.

Una vez creadas las reglas necesarias hay que añadirlas al fichero de configuración para ello nos situamos en la ruta: /etc/snort/snort.conf y dentro de este buscamos o creamos el local.rules, normalmente ya existe solo habría que borrar la almohadilla (#) para descomentarlo.

Finalmente si queremos que nos muestre por pantalla los resultados solo los quedaría activar snort con un comando como este:

snort –q –A console –i lo –c /etc/snort/snort.conf

Probamos una de las reglas que indico a continuación, que es hacer un ping a uno de nuestros equipos, y este es el resultado del la alerta de SNORT

alert icmp any any -> any any (msg:”lanzado un ping”;icode:0;itype:8;sid:1000004;rev:4;)

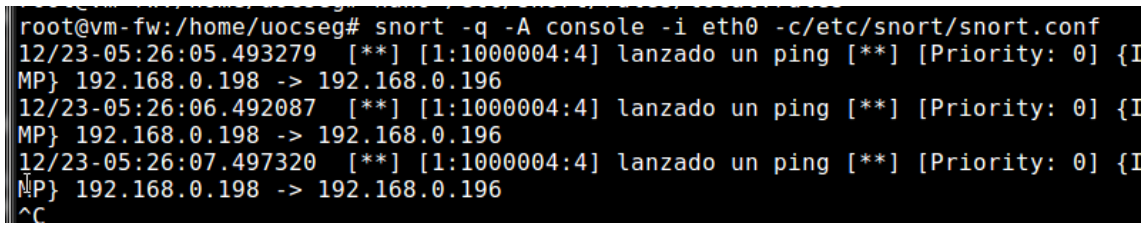

Ilustración 23 SNORT - Alerta por pantalla

8. Herramientas preventivas de seguridad

8.1. Sistema cortafuegos (firewall)

Un cortafuego o Firewall (en ingles) es un sistema diseñado para prevenir acceso no autorizado hacia o desde una red privada. Provee un punto de defensa entre dos redes o protege una red de la otra. Normalmente un firewall se usa para proteger una red privada como puede ser la de una pyme y la red pública, también permite crear o controlar de forma separada una red privada poniendo en puntos estratégicos de la red estos firewall.

Los firewall pueden estar implementados tanto en software como en hardware o combinando las dos.

Existen diferentes técnicas de implementar el firewall:

- **Primera generación – cortafuegos de red: filtrado de paquetes**

En esta primera generación de firewalls se encarga de analizar el tráfico de la red, Cada uno de los paquetes que entra o sale de la red es inspeccionado y dependiendo de las reglas que tenga definidas permitirá o denegará dicho tráfico, es decir, cada paquete se filtrará únicamente por la información contenida en el paquete (ip origen, ip destino, protocolo, tipo tráfico –TCP, UDP, puerto al que se conecta-). Lo que se conoce como listas de control de acceso (ACL).

El filtrado de paquetes actúa sobre las tres primeras capas de modelo OSI (Nivel de Red –Direccionamiento lógico-, Nivel de enlace de datos – Direccionamiento Físico, Nivel Físico –Señal y transmisión binaria-)

- **Segunda generación – cortafuegos de estado: Nivel de circuito**

Esta segunda generación de firewall corresponden a solicitudes de conexión o a conexiones entre dos máquinas, es decir solo aplica mecanismo de seguridad cuando una conexión es establecida tanto sea por TCP como UDP, una vez que la conexión esta

realizada los paquetes pueden navegar por la red sin tener que comprobar de cada vez si son válidos o no, esto permite no sobrecargar los firewall. Este cortafuegos combina las capacidades de filtrado de paquetes y filtrado de contenido.

- **Tercera generación - cortafuegos de aplicación: Nivel de aplicación**

La tercera generación de firewall, permite examinar la información de los paquetes manteniendo la secuencia de la información, trabaja sobre la capa de aplicación del modelo OSI, lo que nos permite validar claves de acceso y algunos tipos de solicitudes de servicios ya que nos permite filtrar protocolos superiores como FTP, Telnet, DNS, DHCP, HTTP, TCP, UDP. Son conocidos como servidores Proxy

Este tipo de firewall es más seguro que los firewall de filtrado de paquetes, ya que atañe a las siete capas del modelo OSI. El mejor firewall de aplicación es ISA (Internet Security and Acceleration)

Los firewall son un punto de referencia y esencial en la seguridad de una red, ya que es importante su ubicación en una Lan, para poder segmentar la red según nos interese el tipo de conexión que queramos permitir. También nos permite enmascarar el tráfico de la red hacia el exterior lo que se conoce como NAT.

- **Politica por defecto Aceptar**: permite el paso de todo tipo de tráfico sin ninguna restricción y solo se negará los que se diga explícitamente.
- **Politica por defecto Denegar**: no permite ningún tipo de trafíco y solo se permitirá el que se especifique explícitamente, es más restrictiva que la anterior, pero hay que ser muy conscientes del tráfico que se vaya a cursar e implica un mayor mantenimiento.

8.2.Ventajas y desventajas del sistema cortafuegos (firewall)

Ventajas:

- **Da protección ante los intrusos:** Permite proteger de ataques o accesos que se puedan realizar desde otro equipo.
- **Servicio de protección de información:** Permite clasificar los accesos de la información o contenido dependiendo de cada grupo o usuario que le corresponda.
- **Optimización de Acceso:** Identifica los equipos de la Red lo que permite una optimización de saltos o comunicación entre ellos.

Desventajas:

- Solo gestiona el tráfico que pasa por él, es decir, el tráfico interno por ejemplo en un switch que redirecciona a otro equipo sin pasar por el firewall podría contener algún software malicioso sin que este se entere.

- El firewall no puede proteger contra los ataques que se puedan realizar contra puertos, direccionamientos o protocolos que estén permitidos en el firewall.
- Una posible desventaja seria su alto mantenimiento si no se configura con una cierta previsión de crecimiento o con restricciones demasiado altas, ya que, si no cada vez que haya un usuario nuevo o permitir un nuevo acceso a una maquina, habría que incluirla.

8.3.IPTABLES

IPTABLES es un sistema de firewall que viene instalado en todas las versiones de Ubuntu ya que viene siendo parte del sistema operativo, lo que se conoce como kernel. para que las iptables puedan funcionar habría que ir cargando, borrando o modificando reglas según sea necesario.

Tipos de filtros:

- **Tipo Mangle**: Permite añadir reglas que cambian algunos campos de los paquetes, como TTL o TOS.

- **Tipo NAT (network address translation):** permite añadir reglas que cambian las direcciones IP de los paquetes, pueden ser del tipo:

SNAT, para cambiar la dirección fuente.

DNAT para cambiar la dirección destino.

Además permite el enrutamiento lo que se conoce como:

PREROUTING: Donde se modifican los paquetes entrantes antes de enrutarlos hacia el destino.

POSTROUTING: Se modifican los paquetes una vez enrutados en local.

- **Tipo Filter:** Se encarga del filtrado de los paquetes se divide en:

FORWARD: Son paquetes con origen y destino remoto donde nuestro firewall tiene que funcionar como Gateway, es decir reenviando los paquetes que recibe de una dirección origen a otra de destino.

INPUT: Son paquetes con origen remoto pero con destino local, serian los paquetes que las maquinas externas nos envían a nuestro equipo.

OUTPUT**:** Son los paquetes con origen local pero con destino remoto, serían los paquetes que nuestra máquina envía a las maquinas externas.

Incluyo un esquema del filtrado de paquetes, poder entender de una forma más visual el proceso que explico a continuación sobre los pasos que se siguen.

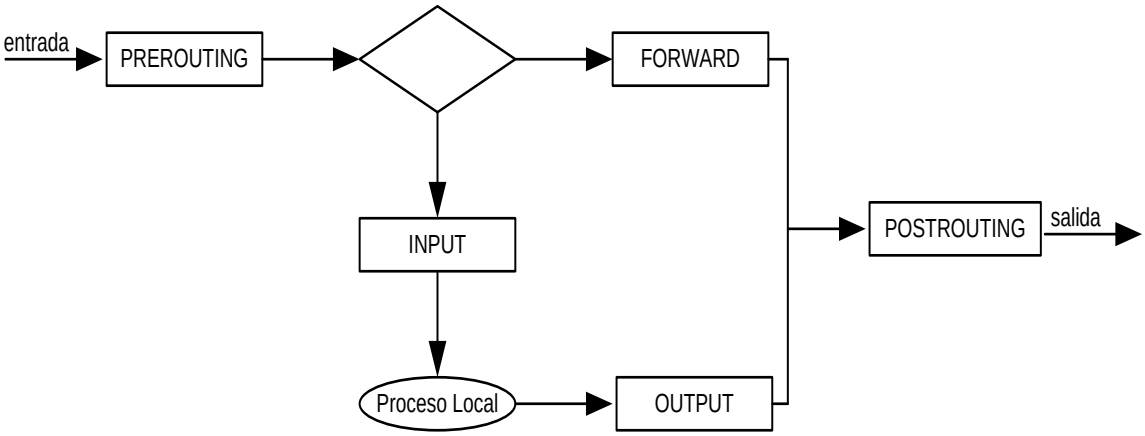

Ilustración 24 IPTABLES - Esquema de filtrado

En este esquema se detalla el proceso desde que se recibe un paquete hasta que sale del firewall.

Cuando recibimos un paquete el primer filtro que pasa es el PREROUTING donde podemos modificar los datos de destino tanto su direccionamiento ip como su puerto de destino (DNAT), una vez pasado este filtro se comprueba si el paquete va dirigido a la propia maquina en cuyo caso se irá por INPUT o si va a otra máquina reenviando el paquete por FORWARD.

Si el paquete va dirigido a la propia maquina y consigue pasar el filtro de INPUT los paquetes llegarán a la maquina que lo solicito, si esta máquina tiene que enviar un paquete este tiene que pasar por el filtro OUTPUT.

Finalmente si el paquete ha conseguido pasar tanto sea por FORWARD o por OUTPUT tendrán que pasar antes de salir del firewall por el POSTROUTING donde se le aplicará si fuese necesario SNAT

8.3.1. Interfaces gráficas para el manejo de las IPTables

Existen muchas maneras de gestionar las iptables y que no se nos hagan tan tediosas seguir que regla era para que cosa y que otra regla era para otra cosa, para ello hay diferentes herramientas con un interfaz gráfico mucho más intuitivo y fácil de seguir.

Hay dos aplicaciones que destacan por ser de código abierto GNU Public License y que nos permite una gestión mucho más cómoda, son : FWbuilder y guFW, esta ultima ya viene embebida en el kernel, y es la que vamos a comentar.

Ufw significa Uncomplicated Firewall que como la propia palabra indica firewall sin complicaciones, dispone de una interfaz gráfica super sencilla de usar.

8.3.2. Instalación de uFW

Primero comprobamos el estado de la aplicación que viene embebida en el kernel con el siguiente comando: sudo ufw status

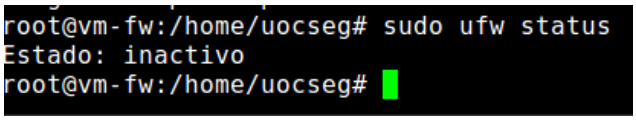

Ilustración 25 uFW - Estado del servicio

Como podemos ven en la imagen anterior la aplicación está instalada pero desactivada. En caso de no estar instalada se puede instalar con el siguiente comando: sudo apt-get install ufw

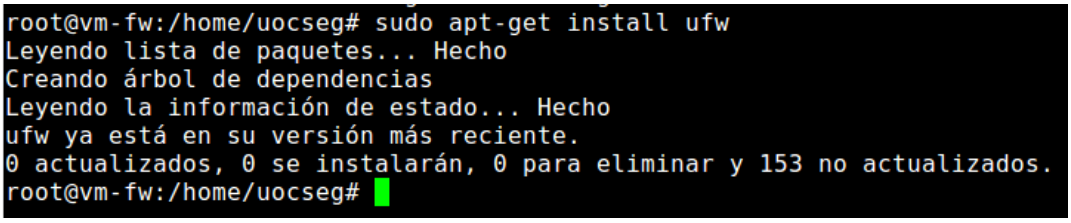

Ilustración 26 uFW – Instalación aplicación

Una vez instalada habrá que cargarle el entorno gráfico con el siguiente comando: sudo apt-get install gufw

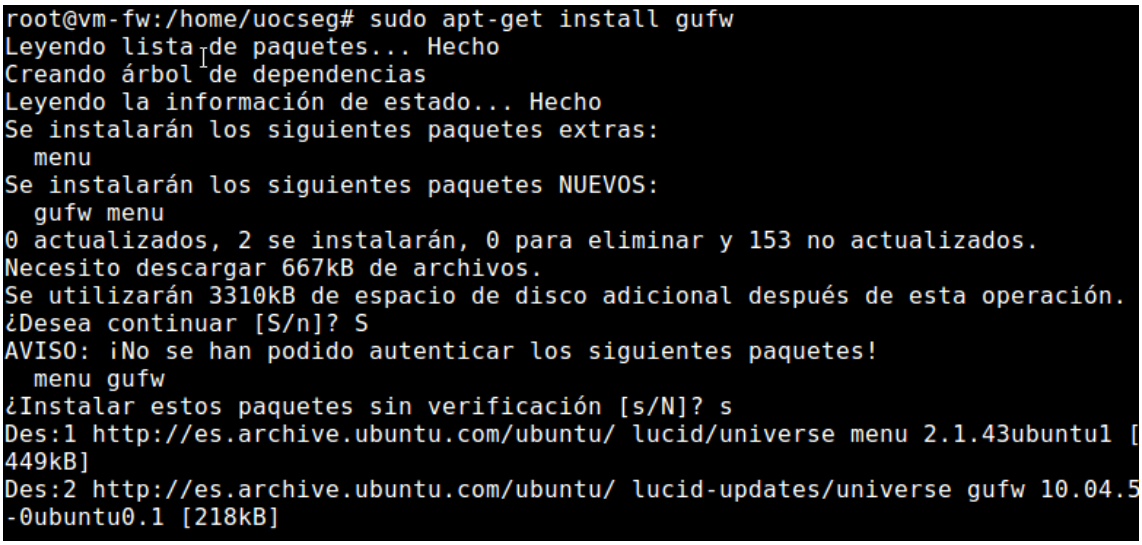

Ilustración 27 uFW – Instalación Interfaz gráfica

Una vez instalado ya podemos acceder a el por medio de sistemas→Configuración de Cortafuegos

8.3.3. Configuración y reglas ufw

Como podemos ver en la siguiente imagen chequeamos "Activado" y ya nos permite crear las reglas por un lado la política por defecto que tiene Entrante: Denegado y Saliente: Permitido, y luego vamos añadiendo las reglas según nos vayan interesando. De forma más intuitiva tenemos las pestaña "Preconfigurado" que nos permite crear reglas de forma genérica sin direccionamientos IPs y otra pestaña "Avanzado" que nos permite añadir rangos de direccionamientos IPs, de una manera fácil y sencilla.

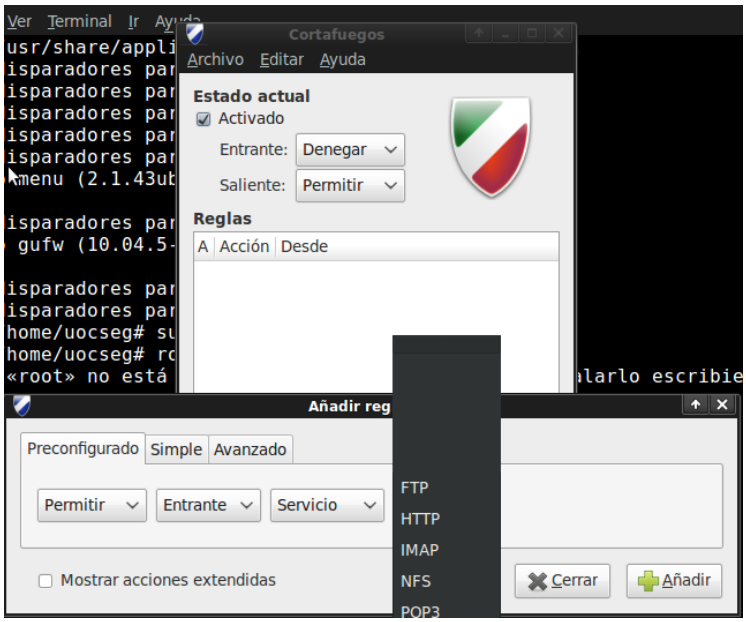

Ilustración 28 uFW – Preconfigurado

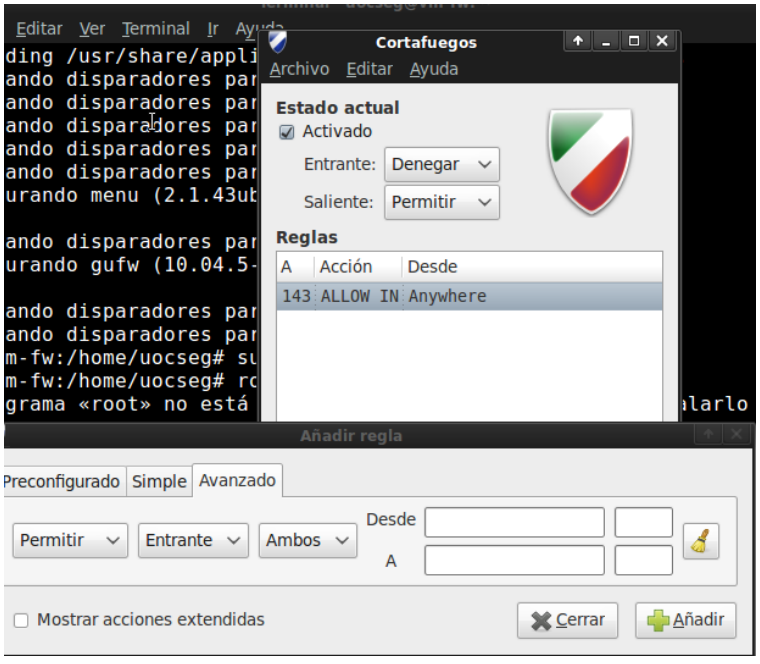

Ilustración 29 uFW – Avanzado

8.4.Arquitectura firewall dual homed.

La arquitectura dual homed se realiza a través de un equipo con la capacidad de encaminamieto desactivada conocidos como equipos de bastión (en ingles bastion host). Este equipo en lo que consiste es que se ha protegido de forma segura para que pueda soportar ataques desde el exterior y nos permita dar una cierta seguridad a nuestra red interna, es decir, nuestros equipos de la red interna se podrá comunicar con el equipo dual homed y la red externa (internet) también se podrá comunicar con el equipo dual homed, pero los equipos de la red externa e interna no se pueden comunicar directamente sino que tendrán que tendrán que pasar por el firewall para permitir o denegar esa comunicación.

Este sistema tiene una gran desventaja ya que si un atacante consiguiese superar nuestro firewall, nada le impediría llegar a nuestra red interna pudiendo llegar a obtener contenido privado o hackear nuestros PCs e instalar desde malwares a gusanos, etc.

A continuación muestro la arquitectura dual homed con un esquema que se podría instaurar en una empresa, dispone de tres patas, por un lado tenemos los equipos de los distintos usuarios de la empresa (directivos, empleados, administrativos, etc), por otro lado tenemos la red donde están los distintos servidores: correo, DNS, web. La comunicación de estas distintas partes de la red se realiza por medio de un router y firewall, permitiendo o denegando el acceso de usuarios a contenidos que queramos proteger o de extraños que puedan acceder a nuestra red interna.

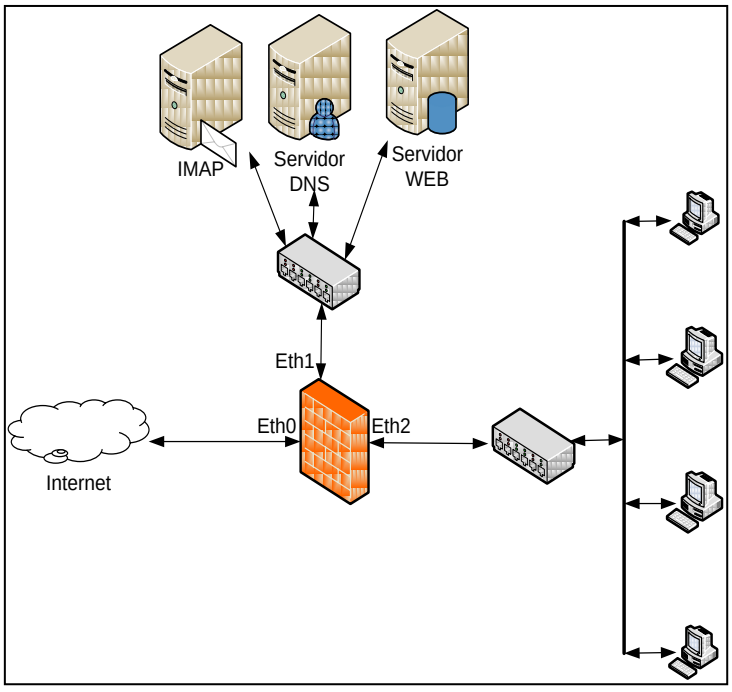

Ilustración 30 Dual homed - Esquema

Para evitar las vulnerabilidades antes mencionadas de un solo firewall se pueden montar dos firewall lo que se conoce como zona desmilitarizada (DMZ).

8.5. Zonas Desmilitarizadas DMZ.

Es una parte fundamental a la hora de diseñar y proteger una red, una zona desmilitarizada conocida como DMZ (en ingles demilitarized zone), permite la separación de equipos como servidores de correo electrónico, servidores web, servidores de DNS o de equipos que queramos que tengan acceso desde el exterior, de otros equipos que queramos que estén aislados del exterior, es decir el objetivo de una red DMZ es que partiendo de que disponemos dos firewalls, uno de ellos (firewall 01) por una de sus patas tenga acceso directo desde el exterior y otra de sus patas a nuestra red semi-interna, los que nos permite separar totalmente las conexiones desde la red externa al DMZ y este a su vez los equipos que se encuentren en esta zona con la conexiones a nuestro segundo firewall (firewall 02) que este a su vez se conectará con la red interna (LAN).

Las políticas de seguridad que se suelen implantar en estos sistemas DMZ, es:

- Permitir tráfico desde la red externa (internet) hacia la DMZ que hemos creado.
- Permite tráfico de la red interna hacia la zona segura DMZ.
- Permite tráfico de la red interna hacia la red externa que nosotros permitamos, sólo si es preciso y con fuertes restricciones.
- No permite el tráfico de la red externa hacia la red interna, nuestra LAN.
- No permite el tráfico de la DMZ hacia la red interna.

En este esquema se puede entender más claramente donde se ubica un DMZ y su comportamiento en la red.

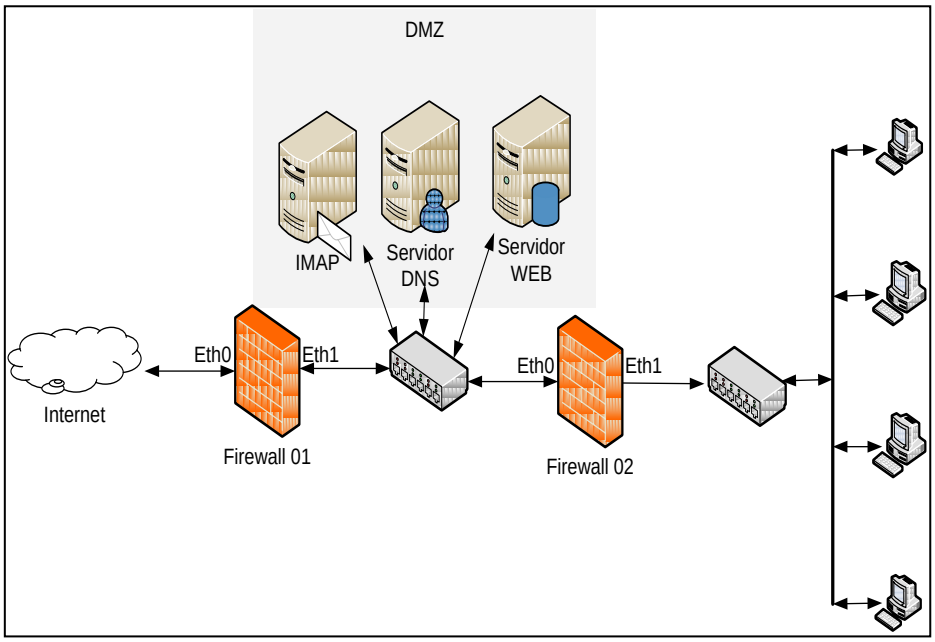

Ilustración 31 DMZ - Esquema

Esta arquitectura DMZ es la más segura y será el que implantemos en nuestra PYME, para dar la máxima seguridad a nuestra red y evitar en la mayor medida posible las intrusiones.

8.6. Squid (Proxy)

Squid es un programa de software libre que realiza la función de un servidor proxy y un dominio para las páginas web, está publicado bajo la licencia GPL.

Un servidor Squid tiene una gran variedad de utilidades que nos permitiría desde acelerar un servidor web guardando las peticiones más comunes hasta dar protección filtrando el tráfico. Este servidor se considera muy robusto y completo pero está orientado hacia el tráfico de HTTP y FTP sobretodo, aunque es compatible con otros protocolos como Gopher y distintos tipos de cifrado como SSL, TLS y HTTPS.

El servidor proxy nos da una seguridad extra a la hora de que los usuarios puedan acceder a internet ya que este equipo se encuentra en una posición intermedia entre los usuarios que quieren acceder a internet e internet, como se puede ver en la siguiente imagen.

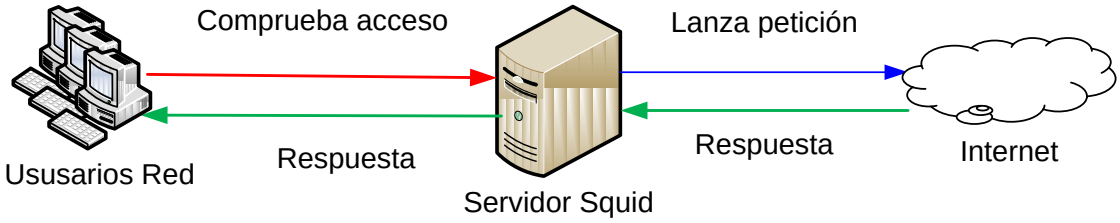

Ilustración 32 Squid - Esquema

8.6.1. Características del servidor:

- Servicio cache de URL y Dominios:

Este servidor permite almacenar en cache las páginas que se han visitado de forma reciente por los distintos usuarios, lo que permite que en accesos venideros a esas páginas la consulta se realizará más rápida y si la URL no se ha modificado no será necesario utilizar ancho de banda para volver a solicitarla. Es importante tener en cuenta que el servidor proxy soporta peticiones HTTP, HTTPS Y FTP.

También tiene la posibilidad de que si tenemos barios proxys anidados, las peticiones de un navegador primero se comprueba si la URL está disponible en el primer Proxy y de no ser así será este servidor Proxy el que pregunte a los otros servidores Proxy si disponen de la URL en cuention.

- Servicios SSL:

El servidor permite acceso por medio de SSL (Secure Socket Layer) lo que permite acelerar el tráfico cifrado.

- Cache Transparente:

El servidor permite la configuración para ser usado de manera transparente, lo que hace que las conexiones se enrutan dentro del proxy sin que el cliente tenga que configurar su equipo lo que hace que el cliente suela desconocer la existencia de este. De modo predefinido Squid utiliza el puerto 3128.

- Listas de control de acceso:

Dispone de listas de control de acceso lo que permite establecer reglas y politicas de control de acceso de forma centralizada.

8.6.2. Instalación:

Lanzamos el comando: sudo apt-get install squid

```
root@vm-fw:/home/uocseg# sudo apt-get install squid
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  squid-common squid-langpack
Paquetes sugeridos:
  squidclient squid-cgi logcheck-database resolvconf winbind
Se instalarán los siguientes paquetes NUEVOS:
  squid squid-common squid-langpack
0 actualizados, 3 se instalarán, 0 para eliminar y 148 no actualizados.
Necesito descargar 1353kB de archivos.
Se utilizarán 8479kB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]?
```

Ilustración 33 Squid - Instalación

Una vez instalado el servidor proxy podemos inicializarlo o pararlo con los siguientes comandos:

Arrancar el servidor squid.
sudo /etc/init.d/squid restart

Parar el servidor squid
sudo /etc/init.d/squid stop

Recargar la configuración del servidor squid
sudo /etc/init.d/squid reload

Para configurar el servidor proxy el fichero que hay que manejar es el squid.conf, que se encuentra en la siguiente ruta: /etc/squid/squid.conf

Para utilizar la opción de Access Control, primero tenemos que tener claro que usuarios van a poder navegar y quiénes no. Para ello creamos lo que se conoce como listas de acceso ACL (Access Control List) y luego sobre estas listas damos permisos a los usuarios según correspondan.

En nuestro esquema y teniendo en cuenta que el servidor squid va leyendo los permisos por orden primero denegaremos el acceso a internet a nuestro equipo de service Backup y nuestro repositorio de Docs, con las siguientes instrucciones:

Estas instrucciones son sencillas de crear teniendo en cuenta las siguientes reglas:

src: Añadimos la Ip o la url de origen.

dst: Añadimos la Ip o la url de destino.

port: para configurar los puertos. (http_proxy 3128)

http_access: palabra clave seguido del permiso allow o deny.

Un ejemplo sobre nuestra red sería este:

Lista de denegado acceso a internet:

acl servidoresBackup src 10.40.1.11/29

acl usuarios src 10.40.1.100/24

Ahora damos permisos o denegamos a las listas con el siguiente comando:

http_access deny servidoresBackup

http_access allow usuarios

Para trabajar de forma transparente al usuario:

La manera de no tener que configurar cada uno de los PCs de los usuarios consistiría en poner como puerta de enlace la IP del servidor Proxy. De esta manera los equipos ya pasarían automáticamente por el Proxy sin que ellos tengan conocimiento de ello, pero aun nos queda un último paso que sería configurar nuestro cortafuegos para que redireccione las peticiones que val al puerto 80 que pasen al puerto 3128.

Para ello primero configuramos nuestro Firewall con la siguiente instrucción:

sudo iptables –t nat –A PREROUTING –i eth1 –p tcp –dport 80 –j REDIRECT –to-port 3128

Y por ultimo añadimos la siguiente instrucción en el Proxy:

http_proxy 3128 transparent

8.6.3. Ventajas y desventajas del Servidor Proxy (squid)

Ventajas:

- **Manejo del acceso a la red:** Los usuarios se comunican hacia el exterior por medio del servidor Proxy, lo que nos permite un mayor control (permitir o denengar) en el acceso web.
- **Optimizacion de recursos:** Permite almacenar en cache las URL mas utilizadas por los usuarios agilizando así los recursos y el ancho de banda.
- **Log de servicios:** Identifica y almacena las URLs que visitan nuestros usuarios de la red.

Desventajas:

- **Punto de fallo**: El servidor sería nuestra única puerta de salida a la red si falla, perderíamos el acceso a la red.
- **Alto mantenimiento:** Hay que cambiar la configuración del servidor por cada una de las paginas o acceso que queramos permitir.

8.7. Servidor LDAP:

Un servidor LDAP en inglés Lightweight Directory Access Protocol (en español Protocolo ligero de Acceso a Directorios), esto significa que hace referencia a un protocolo de nivel de aplicación que nos permite el acceso a la información almacenada en nuestra red. LDAP también puede funcionar como una base de datos dependiendo del formato y configuración que proporcionemos.

Normalmente la funcionalidad de LDAP es la de un servidor que nos permite autentificar distintos servicios y usuarios, es decir: identificar el PC que pretende acceder a información almacenada, o un usuario que quiere acceder en una aplicación web que tengamos alojada en el servidor de nuestra empresa, usuarios que quieran acceder a nuestro servidor utilizando el protocolo FTP, etc.

La funcionalidad seria como se indica que el esquema, donde los usuarios de nuestra red tendrían creadas unas cuentas o pertenecientes a un grupo el cual les permitiría acceder o no a ciertos espacios de almacenamiento o aplicaciones. Lo ideal para facilitar la administración de permisos es que las cuentas de usuarios estén centralizadas en un único repositorio y además habrá que configurar los PCs de los usuarios para que utilicen el servidor LDAP para autentificarse.

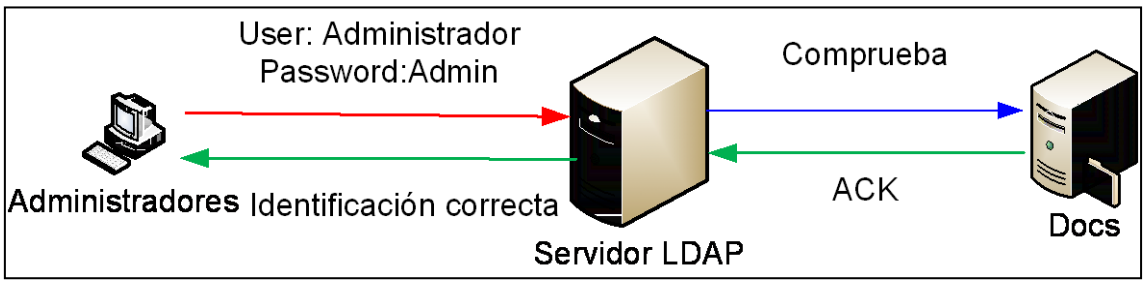

Ilustración 34 LDAP - Equema gráfico

8.7.1. Instalación de OpenLDAP

OpenLDAP es una implementación libre del protocolo que permite conectarse a cualquier LDAP. Tiene su propia licencia, OpenLDAP Public License lo que permite que algunas de las distribuciones de GNU/LINUX y BSD la incluyan en su kernel.

Es necesario instalar el paquete slapd y para ello utilizamos el comando:

sudo apt-get install slapd ldap-utils

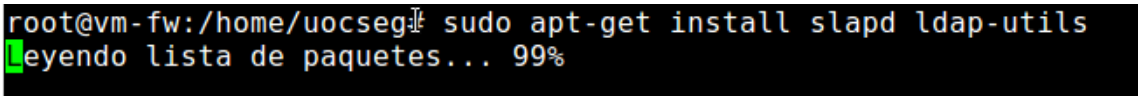

Ilustración 35 OpenLDAP - Instalación

Una vez instalado pasamos a configurarlo, para ello hay que tener en cuenta que los ficheros de configuración se almacenan en la siguiente ruta: /etc/ldap/ pero en lugar de configurarlo de forma manual es preferible lanzar el wizard para la configuración del servidor LDAP con el siguiente comando: sudo dpkg-reconfigure slapd.

Al lanzar el comando nos aparecerá la siguiente pantalla, donde nos preguntará si queremos salir de la configuración de LDAP. Seleccionamos la opción NO.

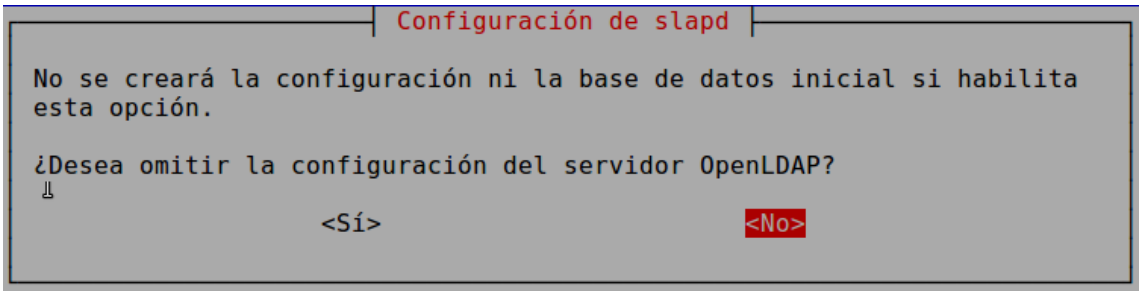

Ilustración 36 Slapd - Configuración 01

A continuación nos pregunta si borra las bases de datos que podamos tener anteriores, en nuestro caso aunque sea la primera que tenemos seleccionamos que si, para eliminar cualquier posible contenido.

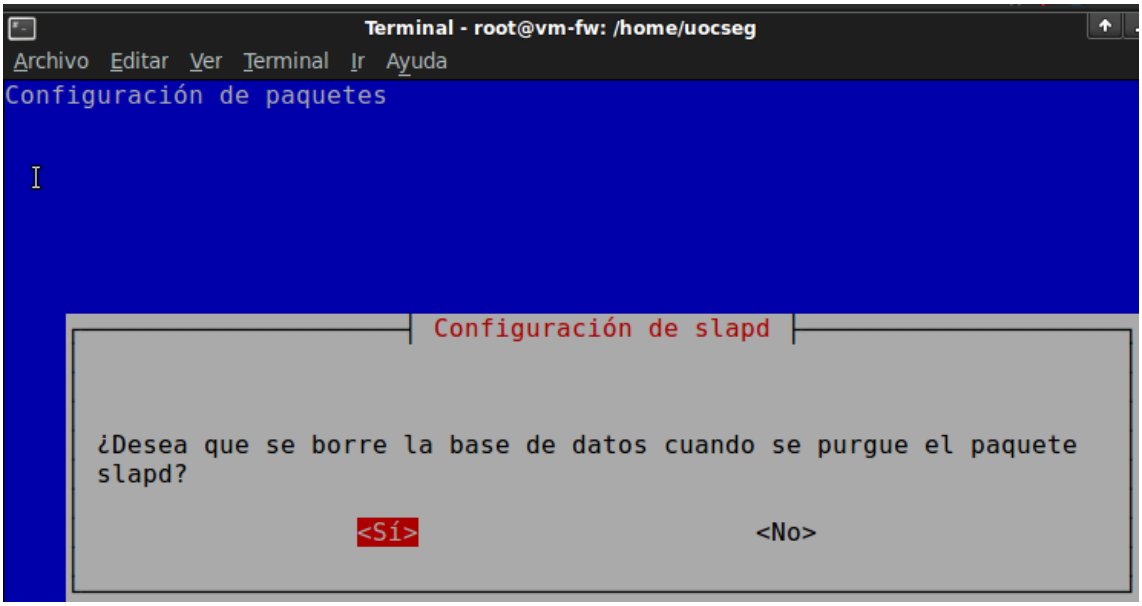

Ilustración 37 Slapd - Configuración 02

En caso de que tengamos otra base de datos que no queramos eliminar habrá que seleccionar que NO, y nos preguntará si queremos mover la base de datos, le diremos que sí y entonces nos pedirá el usuario y contraseña para poder moverla.

La siguiente ventana que nos muestra es la de si queremos utilizar LDAPv2 seleccionamos que NO, ya que esta versión actualmente casi no se utiliza.

Una vez instalada se podrá para el servidor o arrancar con los siguientes comandos:

- Arrancar/reiniciar el servidor LDAP:

sudo /etc/init.d/slapd restart

Para el servidor LDAP

sudo /etc/init.d/slapd stop

8.7.2. Interfaces gráficas para el manejo del servidor LDAP:

8.7.2.1. Aplicación GQ

Instalamos la aplicación con el siguiente comando: apt-get install gq

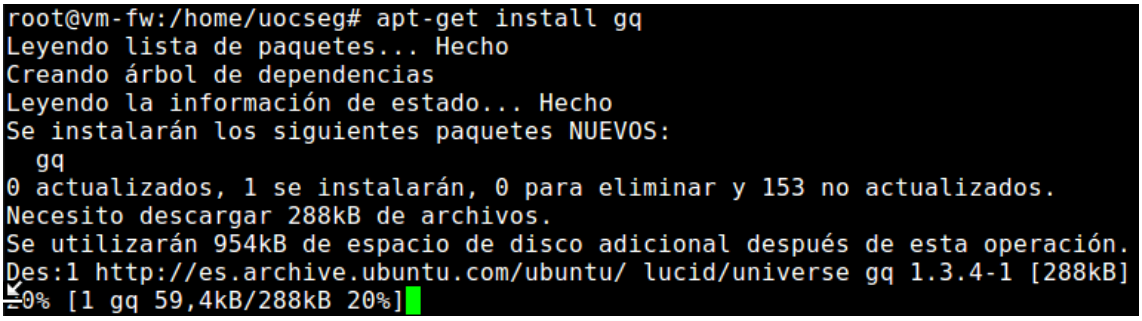

Ilustración 38 GQ - Instalación

Una vez instalado es muy sencillo arrancar el servicio, para ello lanzamos el ejecutador con alt+f2 y escribimos gq.

Una vez arrancado podemos crear y gestionar servicio o cambiar la configuración que trae, como podemos ver en la siguiente imagen

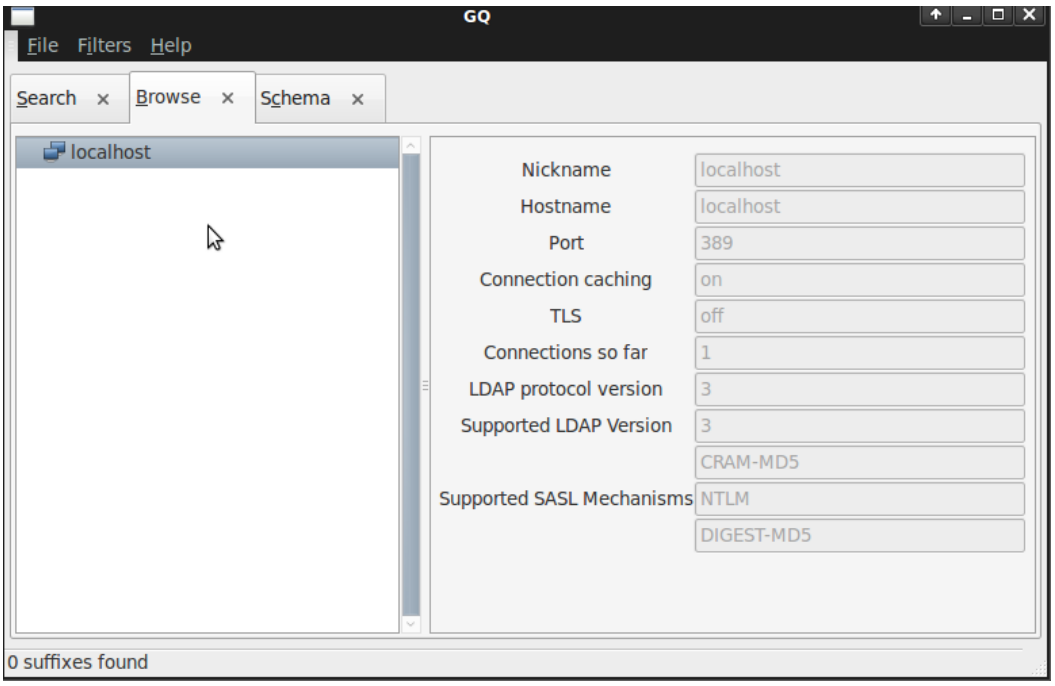

Ilustración 39 GQ - Gestión

8.7.2.2. Aplicación JXplorer:

Para la instalación de esta aplicación requiere que con anterioridad se haya instalado java y activado el repositorio partner, pero esta aplicación dispone de una interfaz mucho más intuitiva que la anterior.

Para ello primero tenemos que modificar el repositorio partner, este se encuentra en la siguiente ruta: /etc/apt/sources.list dentro de este fichero hay que quitar el comentario de las líneas:

deb http://archive.canonical.com/ubuntu lucid partner
deb-src http://archive.canonical.com/ubuntu lucid partner

Una vez activados los partner, instalamos java6, para ello lanzamos el siguiente comando:

apt-get install openjdk-6-jre icedtea-6-jre-cacao default-jre default-jre-headless openjdk-6-jre-lib openjdk-6-jre-headless

```
root@vm-fw:/home/uocseg# sudo apt-get install sun-java6-bin sun-java6-jre sun-
va6-sun/jre/bin/
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
E: No se pudo encontrar el paquete sun-java6-bin
root@vm-fw:/home/uocseg#
```

Ilustración 40 JXplorer - Instalación Java

Una vez terminado el comando de instalación de java, tenemos que modificar el fichero /root/.bashrc añadiendo al Shell las variables que permitan encontrar los binarios de JRE.

Parámetros a añadir en /root/.bashrc
CLASSPATH=/usr/lib/jvm/java-6-sun/jre/bin/
JAVA_HOME=/usr/lib/jvm/java-6-sun/jre/bin/
PATH=/usr/lib/jvm/java-6-sun/jre/bin:/sbin:/bin:/usr/sbin:/usr/bin:/usr/bin/X11:/usr/local/sbin:/usr/local/bin

Una vez que tenemos instalado java y modificado el Shell podemos comprobar el correcto funcionamiento con el comando set.

Una vez instalado el java y establecidas las variables CLASSPATH, JAVA_HOME y PATH en el archivo /root/.bashrc, debes cerrar el terminal y volver a abrirlo, para que cargue nuevamente las variables de entorno. Si ejecutas el comando set en el terminal, podrás comprobar que ha cargado las variables de entorno y podrás instalar JXplorer. JXplorer no está disponible en los repositorios de paquetes de debian, pero se puede descargar haciendo clic aquí. Debemos copiar el archivo en la carpeta /tmp de nuestro sistema y ejecutar:
Por ultimo nos queda descargarnos la aplicación de JXplorer, por ejemplo de esta ubicación: http://enebro.pntic.mec.es/arug0000/servicio/jxplorer3.2_linux.bin

Lo descargamos en la carpeta temp y lanzamos la instalación (ojo es importante instalarlo como usuario no como root)

para la instalación lanzamos el siguiente comando: sh /tmp/jxplorer3.2_linux.bin y automáticamente se abrirá el asistente de instalación.

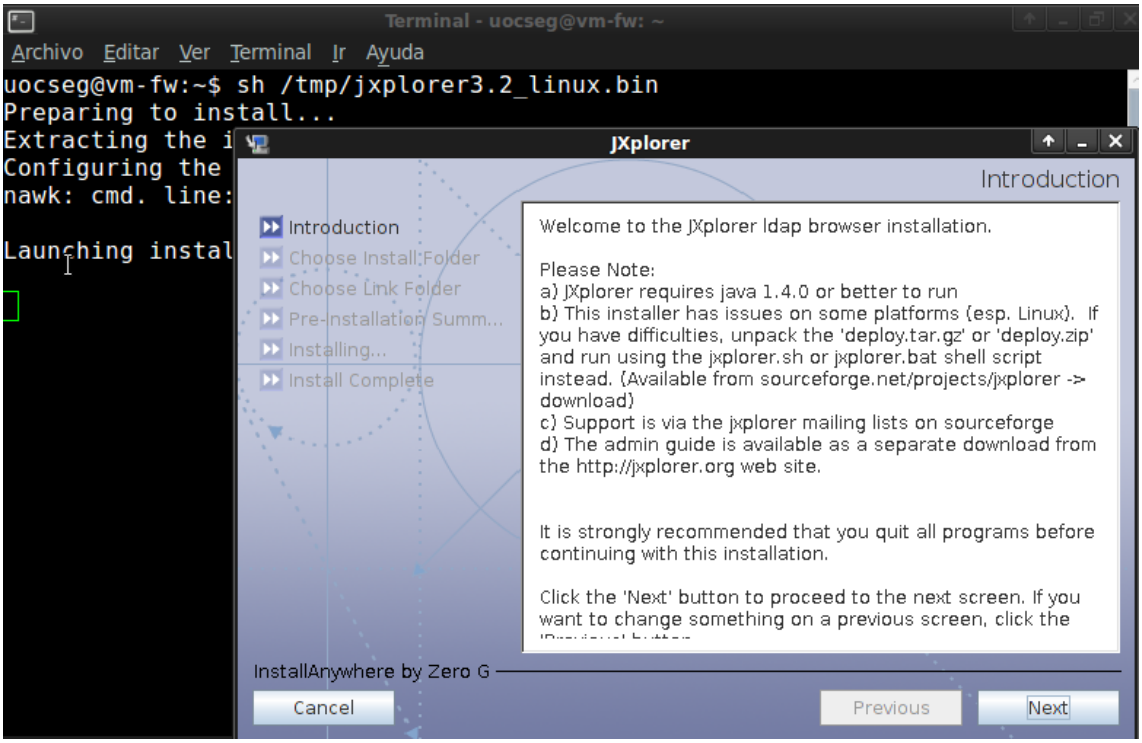

Ilustración 41 JXplorer - Instalación 01

En este primer paso de la instalación nos indica los requisitos necesarios y que podemos hacer si nos encontramos con dificultades en la instalación, seleccionamos Next.

En el segundo paso nos indica en que carpeta queremos realizar la instalación, por eso es importante ejecutarla con el usuario que sea el administrador,

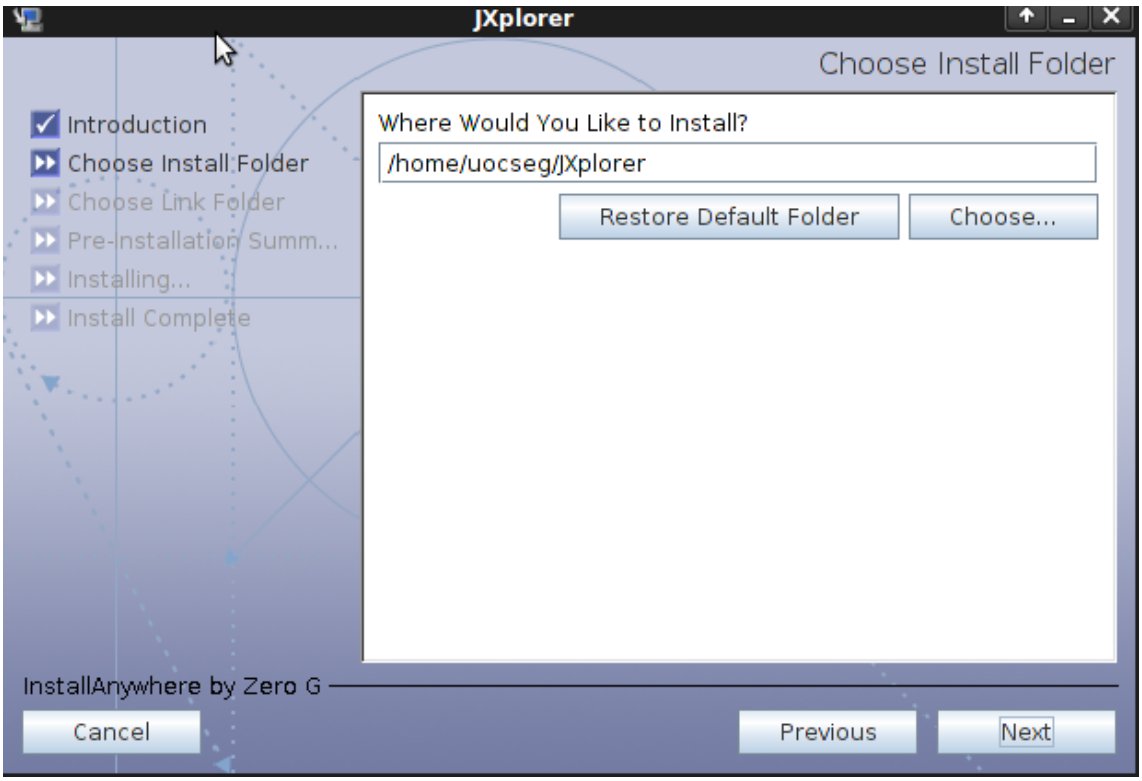

Ilustración 42 JXplorer - Instalación 02

En el tercer paso nos crea un acceso directo en la carpeta que le indiquemos, por defecto nos la crea en home.

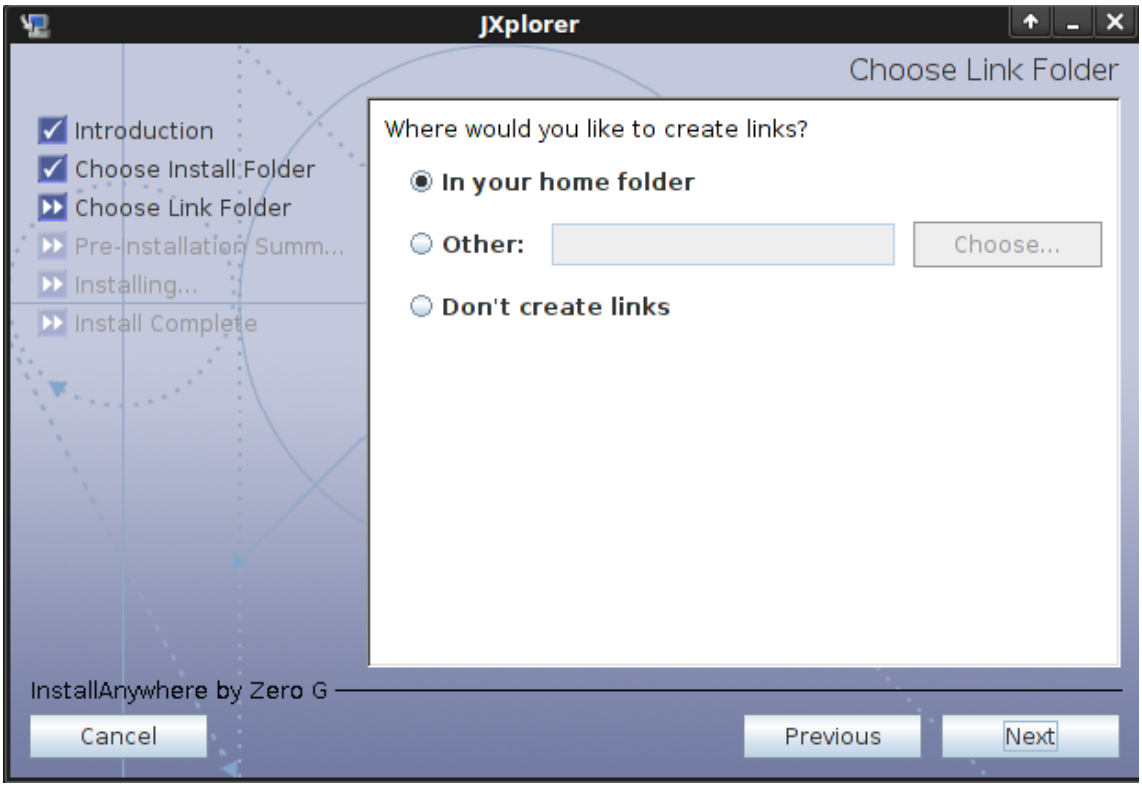

Ilustración 43 JXplorer - Instalación 03

Cuarto paso, nos muestra un resumen de la instalación que se va a realizar.

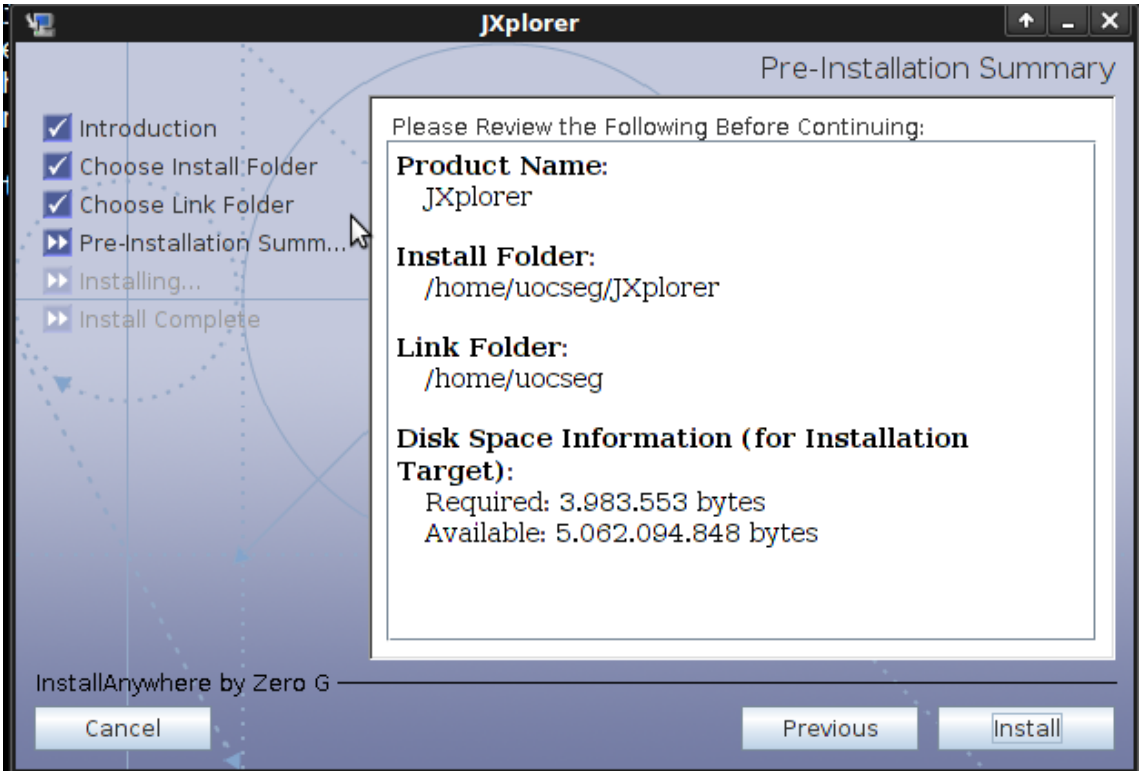

Ilustración 44 JXplorer - Instalación 04

Seleccionamos Install y automáticamente nos aparece la siguiente imagen con una barra de progreso de la instalación.

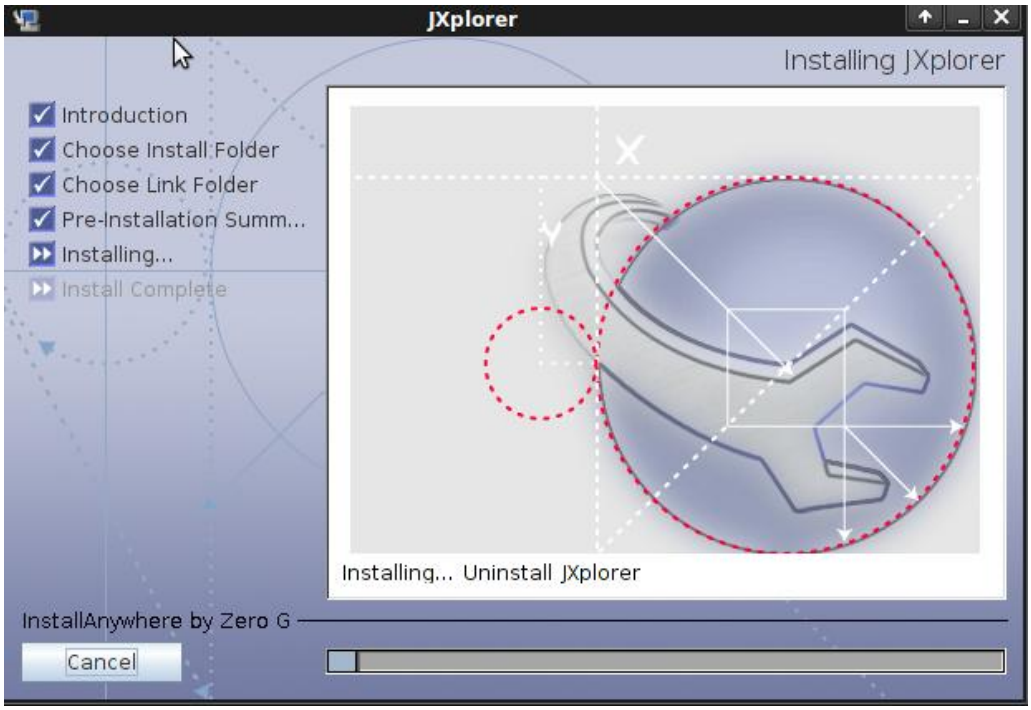

Ilustración 45 JXplorer - Instalación 05

Una vez finalizada la instalación nos aparece la siguiente imagen indicando que se ha fianlizado de forma satisfactoria, en caso contrario nos mostrará un error.

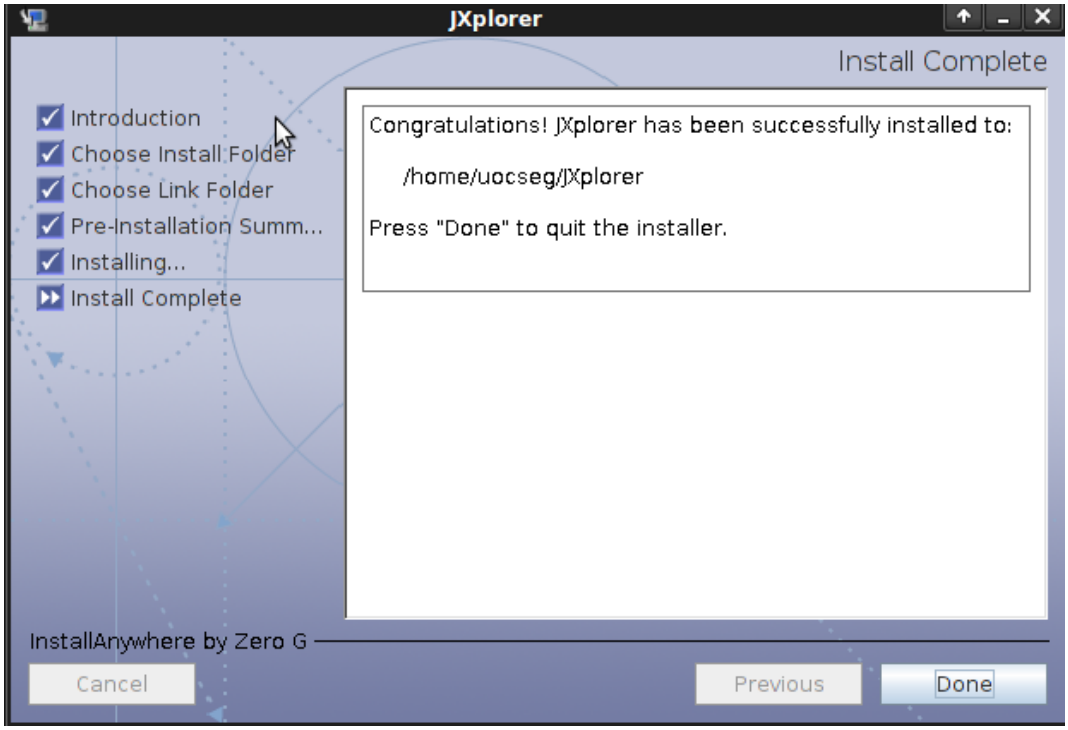

Ilustración 46 JXplorer - Instalación 06

Ahora una vez finalizada la instalación y arrancar la aplicación nos situamos en la carpeta que nos indica la imagen y encontramos el fichero jxplorer.sh.

Ejecutamos el fichero y lo primero que vemos es la pantalla principal de jxplorer que a simple vista se nota que esta interfaz es mucho más amigable que gq, lo que nos permitirá un mejor manejo de permisos para los usuarios.

Sobre la pantalla principal de JXplorer hacemos click en la conexión:

Ilustración 47 JXplorer - Conexión 01

En la ventana que aparece es donde tenemos que configurar la ip del host donde está instalad la base de datos, el puerto por el que escucha el DN raíz que identifica la estructura y el tipo de seguridad de acceso que queremos permitir.

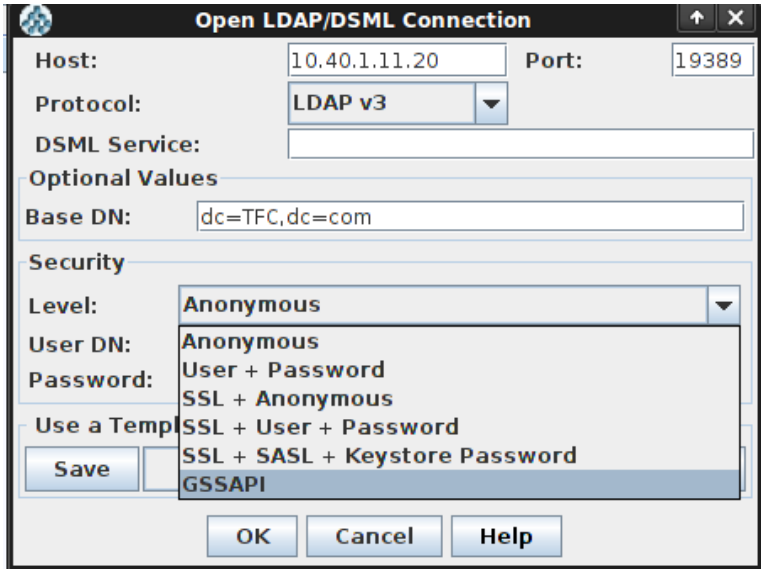

Ilustración 48 JXplorer - Conexión 02

En esta última ventana es la configuración que tendría en nuestra red,

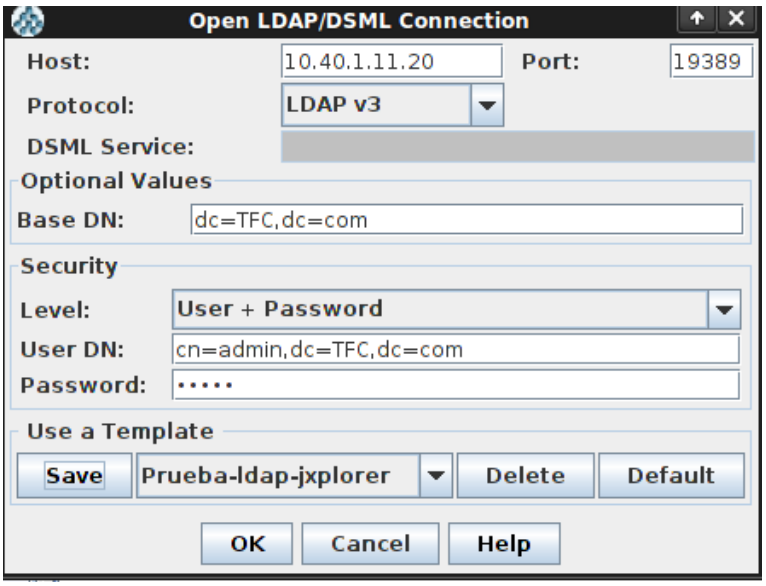

Ilustración 49 JXplorer - Conexión 03

A partir de aquí ya se podrían gestionar los grupos o usuarios de forma independiente añadiéndolos o quitándoles permiso según fuese necesario, de una forma centralizada, lo que nos facilita la gestión.

8.7.3. Estructura de la información en LDAP

Una vez que tenemos instalado y creado el servidor de LDAP, es el momento de añadir la información en la base de datos (LDBM -que es la utilizad por LDAP), pero con una estructura definida que explico a continuación:

Todo nodo está formado por un DN (Distinguished Name), en el cual se definen los atributos de cada uno de ellos como pueden ser: las contraseñas, Nombre, Apellidos, DNI, etc y se va crando una estructura.

- Distinguished Name Raiz (suffix)

Partiendo de la base anterior, si queremos crear un fichero para la base de datos, primero tenemos que crear un Distinguished Name Raíz que represente a nuestra empresa u organización, por ejemplo dc=empresaTIC, dc=com. Para este ejemplo hemos usado dc (Domain Component), pero podemos usar otros muchos como c de (Country), o de (Object), p de (país).

La composición de cada **distinguished name** puede variar, en este caso se utilizaron los vocablos dc de "Domain Component", c de "Country", o de "Object", sin embargo también hubiera sido posible utilizar p de "País",cd de "Componente Dominio". Los vocablos son *solo descriptivos* y su única restricción (si existiese) es llevada acabo en la *definición de Schemas* .

- Distinguished Name Administrativo (rootdn)

Aparte del DN anterior también hay un DN de acceso global a la base de datos, este suele usar por defecto la configuración de cn=Admin, dc=empresaTIC, dc=com. pero además para poder acceder se usa la contraseña que está definida en slapd.conf por medio del parámetro rootpw.

- Archivos LDIF

Los archivos .LDIF son ficheros donde se definen las estructuras DN que acabo de comentar. También existen plantillas que podemos modificar para crear o añadir usuarios o grupos de una manera más sencilla, como por ejemplo estas, que las lanzo con los siguientes comandos:

sudo ldapadd –Y EXTERNAL –H ldapi:/// -f /etc/ldap/schema/nis.ldif

sudo ldapadd –Y EXTERNAL –H ldapi:/// -f /etc/ldap/schema/inetorgperson.ldif

sudo ldapadd –Y EXTERNAL –H ldapi:/// -f /etc/ldap/schema/cosine.ldif

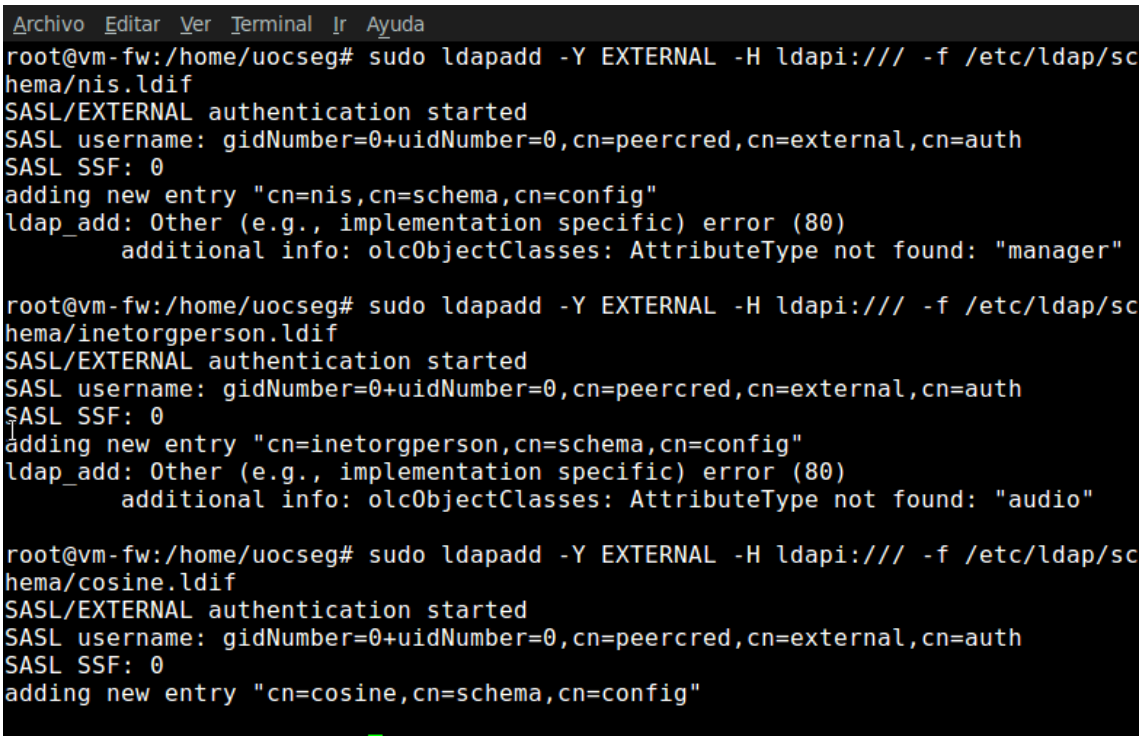

Ilustración 50 LDAP – Plantillas .ldif

Tomando como ejemplos estas plantillas podemos añadir un esquema de configuración con los siguientes parámetros:

- Base del directorio: se configura con el parámetro *olcSuffix*.
- Nombre administrador: se configura con el parámetro *olcRootDN*.
- Contraseña: se configura con el parámetro olcRootPW.

- Permiso de acceso a las contraseñas: se configura con el parámetro olcAccess: to attrs=userPassword (con ello conseguiremos que cada usuario pueda tener permisos para configurar su propia contraseña).
- Permiso de acceso como Root: se configura con el parámetro olcAccess: to *. (El administrador tendrá permisos de escritura y los usuarios de lectura)

Un ejemplo sería:

dn: cn=module,cn=config
objectClass: olcModuleList
cn: module
olcModulepath: /usr/lib/ldap
olcModuleload: back_hdb

Database settings
dn: olcDatabase=hdb,cn=config
objectClass: olcDatabaseConfig
objectClass: olcHdbConfig
olcDatabase: {1}hdb
olcSuffix: dc=TFC,dc=com
olcDbDirectory: /var/lib/ldap
olcRootDN: cn=admin,dc=TFC,dc=com
olcRootPW: admin
olcDbConfig: set_cachesize 0 2097152 0
olcDbConfig: set_lk_max_objects 1500
olcDbConfig: set_lk_max_locks 1500
olcDbConfig: set_lk_max_lockers 1500
olcDbIndex: objectClass eq
olcLastMod: TRUE
olcDbCheckpoint: 512 30
olcAccess: to attrs=userPassword by dn="cn=admin,dc=TFC,dc=com" write by anonymous auth by self write by * none
olcAccess: to attrs=shadowLastChange by self write by * read
olcAccess: to dn.base="" by * read
olcAccess: to * by dn="cn=admin,dc=TFC,dc=com" write by * read

Donde cada parte de la estructura empieza con un distinguished name (dn) y después la definición del objeto con objectClass y cada uno de sus atributos que mencionamos anteriormente.

Añadir los DNs definifos en el fichero LDIF

Una vez definidos los DNs Raiz y los administrativos en el achivo .ldif hay que añadirlos a la base de datos con el siguiente comando:

sudo ldapadd –Y EXTERNAL –H ldapi:/// -f /tmp/tfc-ldap.ldif

```
uocseg@vm-fw:~$ sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /tmp/TFC-ldap.ldif
[sudo] password for uocseg:
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=module,cn=config"

adding new entry "olcDatabase=hdb,cn=config"
ldap_add: Other (e.g., implementation specific) error (80)
        additional info: <olcAccess> handler exited with 1

uocseg@vm-fw:~$
```

Ilustración 51 LDAP – Crear fichero .ldif

Una vez añadidos los ficheros de Raiz y administrativos hay que añadir los que pasarían a formar parte de la jerarquía, los denominados como DN generales.

- DN's Generales

Para la definición de los DNs Generales se crean en otro fichero .ldif distinto.

dn: dc=TFC,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
dc: TFC
o: TFC

Aquí creamos las tres estructuras de los distintos objetos (administrador, usuarios, grupos)
dn: cn=admin,dc= TFC,dc=com
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword:: e2NyeXB0fXdSVDNLMEpKSlQydmM=

dn: ou=users,dc= TFC,dc=com
objectClass: organizationalUnit
objectClass: top
ou: users

dn: ou=groups,dc= TFC,dc=com
objectClass: organizationalUnit
objectClass: top
ou: groups

```
# Aquí añadimos los atributos a los objetos (administrador, usuarios, grupos )
dn: cn=Jefe,ou=users,dc=TFC,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: posixAccount
objectClass: top
cn: Jefe
gidNumber: 1001
homeDirectory: /home/javier
loginShell: /bin/bash
sn: Jefe
uid: Jefe
uidNumber: 1001

dn: cn=User01,ou=users,dc= TFC,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: posixAccount
objectClass: top
cn: User01
gidNumber: 1001
homeDirectory: /home/joaquin
loginShell: /bin/bash
sn:: R8OzbWV6
uid: User01
uidNumber: 1002

dn: cn=Administrador01,ou=users,dc= TFC,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: posixAccount
objectClass: top
cn: Administrador01
gidNumber: 1002
homeDirectory: /home/miguel
loginShell: /bin/bash
sn: Administrador01
uid: Administrador01
uidNumber: 1003


# Aquí asociamos los objetos con los atributos los grupos (Usuarios, Administradores)
dn: cn=Usuarios,ou=groups,dc= TFC,dc=com
objectClass: posixGroup
objectClass: top
```

cn: profesores
gidNumber: 1001
memberUid: Jefe
memberUid: User01

dn: cn=alumnos,ou=groups,dc= TFC,dc=com
objectClass: posixGroup
objectClass: top
cn: alumnos
gidNumber: 1002
memberUid: Administrador01

Para agregar esta estructura a la base de datos(LDIF) hay la lanzar el siguiente comando:

Sudo ldapadd –c –x –D cn=admin,dc=TFC,dc=com –W –f /tmp/ tfc-grupos.ldif

Con esto ya tendríamos nuestra estructura global para la gestión de permisos de la empresa.

8.7.4. Autentificación con OpenSSl en nuestra base de datos

La autentificación por medio de OpenSSL nos permite dar un punto más de seguridad a nuestra red, ya que si para acceder al servidor lo hacemos por medio de un Login, este podría ser detectado por cualquiera que este esnifando nuestra red.

Con la autentificación por medio de OpenSSL haría que aunque haya un usuario que estuviese esnifando la red y tenga paquetes para analizar sería casi imposible que los descifrase ya que OpenSSl utiliza un sistema de cifrado asimétrico.

Hay que tener en cuenta que el servidor LDAP utiliza el puerto 389 y si queremos cifrar el contenido con SSl habría que cambiar el puerto por el 636 que corresponde con el protocolo seguro de LDAP, también no hay que olvidad que hay que seguir unos pasos para crear y disponer de un certificado autofirmado o por el contrario por una entidad certificadora (CA).

Como se indican en los siguientes pasos:

- Crear una nueva entidad certificadora
- Crear una petición de firma de certificado del servidor
- Firmar el certificado con la CA
- Copiar los certificados a la carpeta deseada, renombrar y proteger

- Configurar slapd para que utilice los certificados
- Modificar script de inicio de slapd para que utilice protocolo seguro ldaps
- Reiniciar slapd

9. Configuración de nuestra Red para la PYME

9.1. Configuración de nuestro Firewall-01 y Firewall-02

Como veremos más adelante la configuración de nuestros firewalls, la hemos realizado por la rama de la denegación de servicios, es más costosa de mantener pero más segura, empezamos mostrando la configuración de los puertos de red, para ver esta configuración usamos el comando ifconfig.

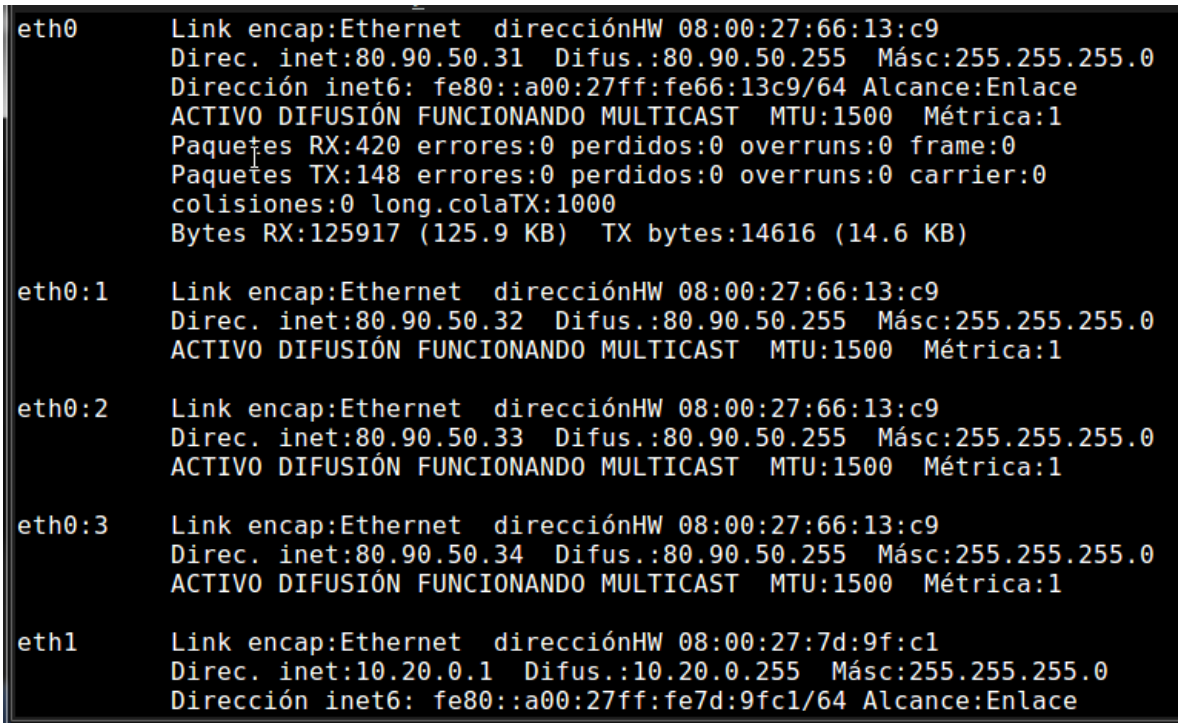

Ilustración 52 Firewall 01 - ifconfig

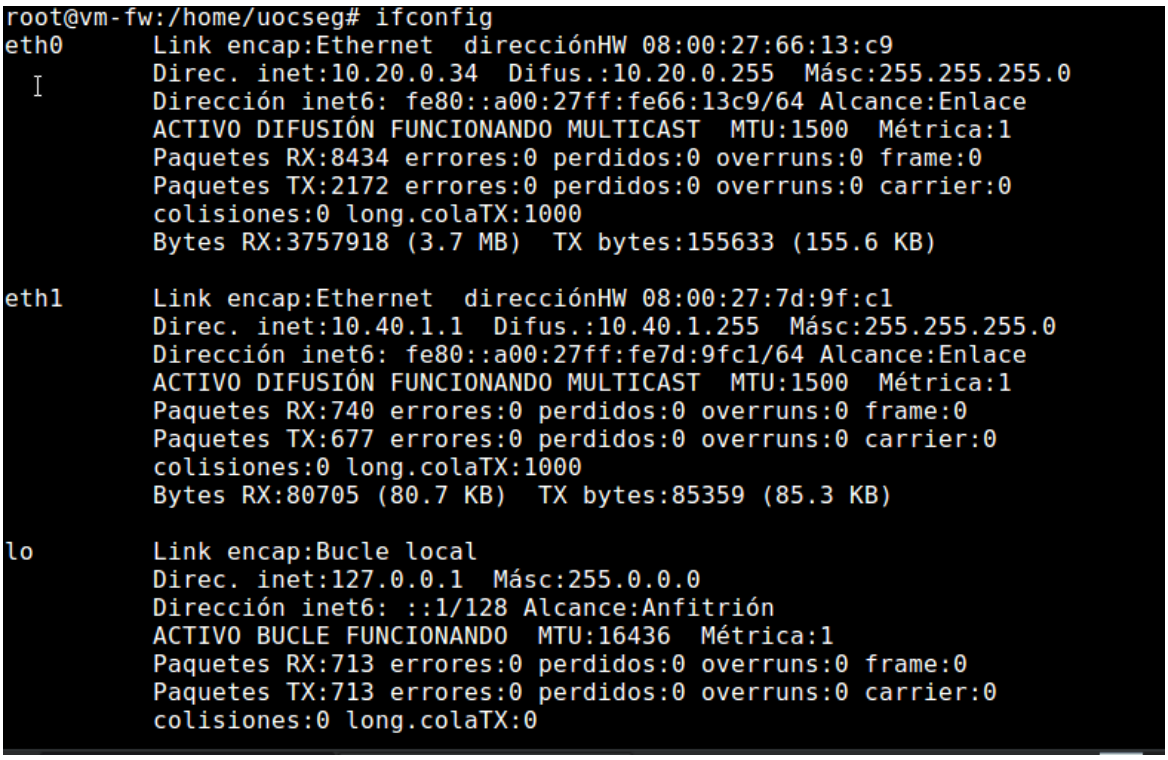

Ilustración 53 Firewall 02 - ifconfig

Para que todo funcione correctamente debemos tener en cuenta la puerta de enlace (Gateway) ya que muchas veces aunque tengamos todos los direccionamientos correctos, si no le indicamos cual sería la puerta de "salida" no habrá comunicación posible, los equipos recibirán paquetes de información en la entrada pero estos no serían enrutados correctamente.

Una vez configurados los puertos correctamente, se hace necesario realizar la carga de las reglas IPTables, atendiendo al criterio que definirá nuestra red que permitirá el enrutamiento correcto de los diferentes tipos de protocolos, puertos, etc. En nuestro caso como ya hemos indicado, utilizamos una política de denegación de servicios por lo tanto en INPUT, FORWARD, OUTPUT están definidas con la política (DROP)

9.2.Comandos básicos de consultas

Los comandos básicos de consulta son los siguientes:

iptables -L -n –v

-L: Lista de reglas.

-V: Muestra información detallada. Esta opción hace que el comando de la lista muestra el nombre de la interfaz, las opciones de la regla, y las máscaras TOS.

-N: la dirección IP y el puerto de visualización en formato numérico. No utilizar DNS para resolver nombres. Esto acelerará la lista.

Para controlar los procesos:

#service iptables stop
service iptables start
service iptables restart

Como podemos ver en las dos siguientes imágenes donde hemos realizado unas consultas a los firewall 01 y firewall 02 con la configuración que se utiliza para nuestra PYME. Este tipo de consulta es un poco más difícil de seguir a simple vista y quizás sería necesario entrar en cada una de las reglas para poderlas ver con más detalle que es lo que implica cada una de nuestras restricciones.

Por ello es conveniente utilizar un interfaz gráfico que nos permita hacer un seguimiento mucho más rápido y claro como es el uFW que ya hemos visto anteriormente.

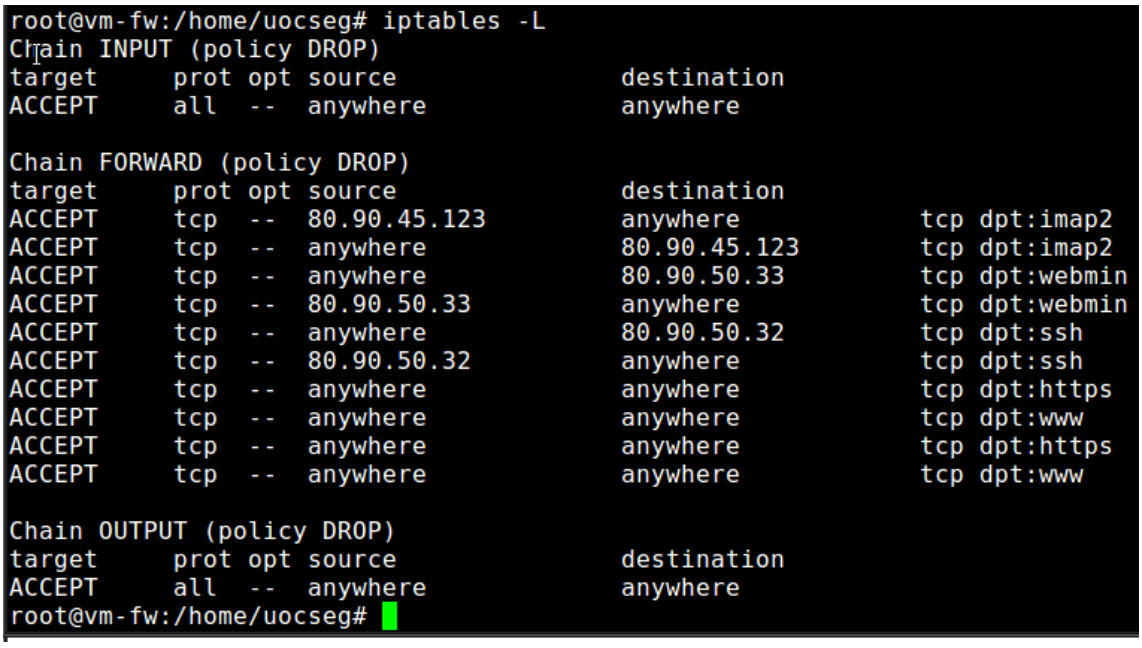

Ilustración 54 Firewall 01 - Consulta

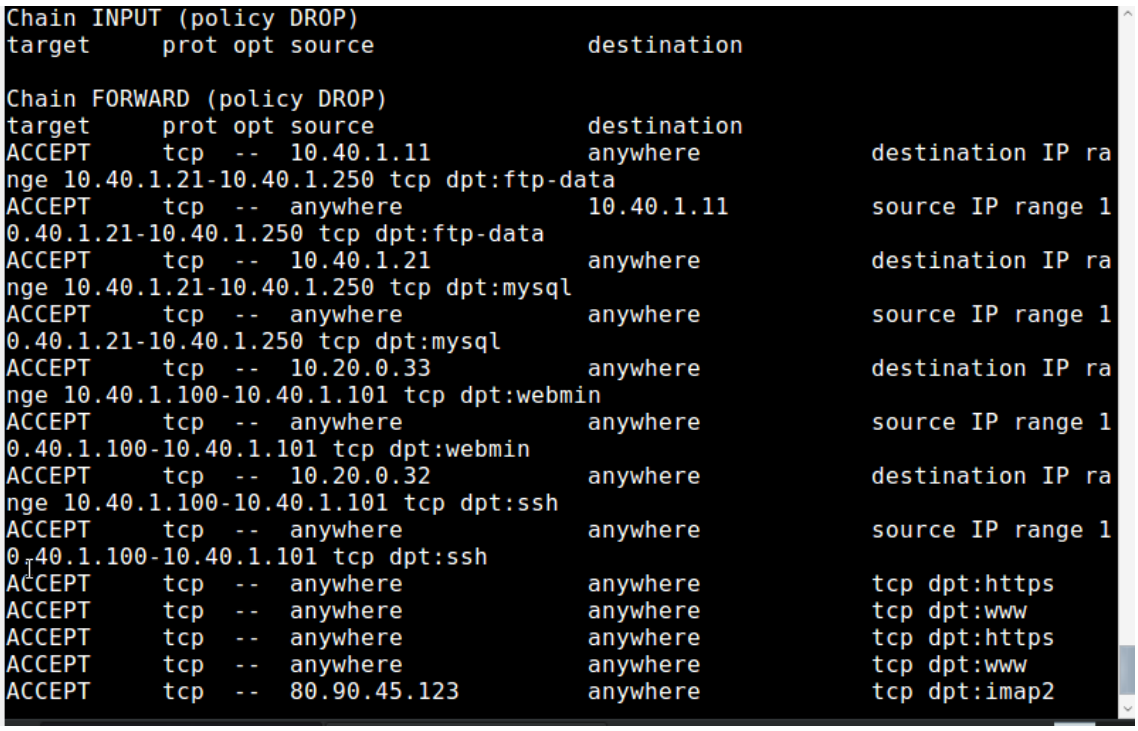

Ilustración 55 Firewall 02 - Consulta

A continuación muestro la configuración implementada en los firewall 01 y firewall 02 de forma separada, en cada uno de ellos se añade una aclaración del propósito de cada uno de los comandos a ejecutar, con el fin de implementar las reglas.

Reglas de configuración para los distintos firewall.

```
# Configuramos los direccionamientos del FW01
sudo ifconfig eth0 80.90.50.31 netmask 255.255.255.0 up
sudo ifconfig eth0:1 80.90.50.32 netmask 255.255.255.0 up
sudo ifconfig eth0:2 80.90.50.33 netmask 255.255.255.0 up
sudo ifconfig eth0:3 80.90.50.34 netmask 255.255.255.0 up
sudo ifconfig eth1 10.20.0.1 netmask 255.255.255.0 up

sudo route add default gw 80.90.50.1 eth0

# Eliminamos las posibles reglas existentes.
iptables -F
iptables -X
iptables -Z
iptables -t nat –F

# Permitir hacer de router
echo 1 > /proc/sys/net/ipv4/ip_forward

# Traducir direccion hacia servidor
iptables –t nat –A PREROUTING –i eth0 –d 80.90.50.32 –j DNAT --to-destination 10.20.0.32
iptables –t nat –A PREROUTING –i eth0 –d 80.90.50.33 –j DNAT --to-destination 10.20.0.33
iptables –t nat –A PREROUTING –i eth0 –d 80.90.50.34 –j DNAT --to-destination 10.20.0.34

#Enmascarar el direccionamiento para que no salga con el propio del equipo
iptables –t nat –A POSTROUTING –o eth0 –j MASQUERADE


#Establecer politica de denegacion: DROP
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
```

```
#permitir acceso desde la red local al IM por la red local
iptables  -A INPUT -i lo -j ACCEPT
iptables  -A OUTPUT -o lo -j ACCEPT


=======================================
# permitir acceso del servidor iMAP a nuestra red y de la red al IMAP
iptables -A FORWARD -s 80.90.45.123 -p tcp --dport 143  -j ACCEPT
iptables -A FORWARD -i eth1 -d 80.90.45.123 -p tcp --dport 143  -j ACCEPT

=======================================
# permitir a la nube acceder a nuestro servidor intermedio
iptables -A FORWARD -i eth0 -d 80.90.50.33 -p tcp --dport 10000  -j ACCEPT
iptables -A FORWARD -s 80.90.50.33 -p tcp --dport 10000  -j ACCEPT

=======================================
# permitir a la nube acceder a nuestro servidor SSH
iptables -A FORWARD -i eth0 -d 80.90.50.32 -p TCP --dport 22  -j ACCEPT
iptables -A FORWARD -s 80.90.50.32 -p TCP --dport 22  -j ACCEPT

=======================================
# permitir navegar desde la red interna
iptables -A FORWARD -i eth1 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -i eth1 -p tcp --dport 443 -j ACCEPT
iptables -A FORWARD -o eth0 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -o eth0 -p tcp --dport 443 -j ACCEPT

=======================================
```

```
# Configuración del FW02 con IP(10.20.0.34)

sudo ifconfig eth0 10.20.0.34 netmask 255.255.255.0 up
sudo ifconfig eth1 10.40.1.1 netmask 255.255.255.0 up


sudo route add default gw 10.20.0.1 eth0

# Eliminamos las posibles reglas existentes.
iptables -F
iptables -X
iptables -Z
iptables -t nat –F

# Permitir hacer de router
echo 1 > /proc/sys/net/ipv4/ip_forward


#Establecer politica de denegación: DROP
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP

#permitir acceso desde la red local al IM por la red local
iptables  -A INPUT -i lo -j ACCEPT
iptables  -A OUTPUT -o lo -j ACCEPT



=======================================
# permitir a Servidor de Backup acceder a nuestros equipos de red
iptables -A FORWARD -s 10.40.1.11 -m iprange --dst-range 10.40.1.21-10.40.1.250  -p tcp --dport 20  -j ACCEPT
iptables -A FORWARD -m iprange --src-range 10.40.1.21-10.40.1.250 -p tcp --dport 20  -j ACCEPT
```

```
=======================================
# permitir a Servidor de Docs acceder a nuestros equipos de red
iptables -A FORWARD -s 10.40.1.21 -m iprange --dst-range 10.40.1.21-10.40.1.250  -p tcp --dport 3306  -j ACCEPT
iptables -A FORWARD -m iprange --src-range 10.40.1.21-10.40.1.250 -p tcp --dport 3306  -j ACCEPT

=======================================

# permitir al Servidor intermedio de Docs acceder a nuestro Directores
iptables -A FORWARD -s 10.20.0.33 -m iprange --dst-range 10.40.1.100-10.40.1.101 -p TCP --dport 10000  -j ACCEPT
iptables -A FORWARD -m iprange --src-range 10.40.1.100-10.40.1.101 -p TCP --dport 10000  -j ACCEPT
=======================================

# permitir al Servidor SSH acceder a nuestro Directores
iptables -A FORWARD -s 10.20.0.32 -m iprange --dst-range 10.40.1.100-10.40.1.101 -p TCP --dport 22  -j ACCEPT
iptables -A FORWARD -m iprange --src-range 10.40.1.100-10.40.1.101 -p TCP --dport 22  -j ACCEPT
=======================================

# permitir navegar desde la red interna
iptables -A FORWARD -i eth1 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -i eth1 -p tcp --dport 443 -j ACCEPT
iptables -A FORWARD -o eth0 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -o eth0 -p tcp --dport 443 -j ACCEPT
=======================================

# permitir acceso del servidor iMAP a nuestra red y de la red al IMAP
iptables -A FORWARD -s 80.90.45.123 -p tcp --dport 143  -j ACCEPT
iptables -A FORWARD -i eth1 -d 80.90.45.123 -p tcp --dport 143  -j ACCEPT
=======================================
```

10. Conclusiones

La realización de este trabajo demuestra la importancia de la seguridad en la red, tanto como si se realiza en una pequeña empresa o incluso en una red local como en el hogar, ya que de otro modo estaríamos a merced de posibles atacantes malintencionados.

El software libre es una gran herramienta con gran potencial que nos permite ahorrarnos muchos costes de licencias, mantenimientos y tenerlo siempre actualizado, el único coste que presentamos es el de los componentes hardware, que como hemos visto utilizando estos sistemas libres no necesitamos tantos recursos como los de pago.

También vemos la importancia de segmentar correctamente las redes, centralizar su gestión y controlar los permisos o restricciones de los usuarios que nos permitirá facilitar enormemente su gestión.

11. Glosario

ACL :: Acrónimo de Lista de Control de Acceso (del inglés, Access Control List).
ARM :: Se trata de una arquitectura RISC (Reduced Instruction Set Computer, Computación de Juegos de Instrucciones Reducidas) de 32 bits desarrollada por ARM Holdings.
ARP :: Acrónimo de Protocolo de Resolución de Direcciones (del inglés, Address Resolution Protocolo).
Arquitectura dual-homed:: Equipo que tiene, al menos, dos interfaces de red.
Asociación de seguridad (SA):: Relación entre un nodo origen y un nodo destino que utilizan uno de los protocolos IPsec (AH o ESP) para enviar datagramas IP protegidos.
Ataque:: Acción realizada por una tercera parte, distinta del emisor y del receptor de la información protegida, para intentar contrarrestar esta protección.
Autoridad de certificación (CA):: Entidad que emite certificados de clave pública que sirven para que los usuarios que confíen en esta autoridad se convenzan de la autenticidad de les claves públicas.
Autoridad de certificación (CA) raíz:: CA que no tiene ninguna otra superior que certifique la autenticidad de su clave pública y que por tanto tiene un certificado firmado por ella misma.
Certificado de clave pública:: También conocido como certificado digital, es una estructura de datos que contiene un nombre de usuario y su clave pública y que está firmado digitalmente por una autoridad de certificación dando fe de esta asociación usuario-clave pública.
Certificado digital:: Certificado de clave pública.
Cifrado:: Transformación de un texto en claro, mediante un algoritmo que tiene como parámetro una clave, en un texto cifrado ininteligible para quien no conozca la clave de descifrado.
Clave privada:: Clave que permite realizar la transformación criptográfica inversa a la que se obtiene con una clave pública y que es computacionalmente inviable obtener a partir de esta última.
Clave pública:: Clave que permite realizar la transformación criptográfica inversa a la que se obtiene con una clave privada.
Clave simétrica:: Clave que permite realizar tanto una transformación criptográfica como la transformación inversa, es decir, cifrado y descifrado.
Confidencialidad:: Protección de la información contra lectura por parte de terceros no autorizados.
Contraseña:: Palabra (“password’) o cadena de caracteres secreta, de longitud relativamente corta, usada por una entidad para autenticarse.
Cookie :: Fichero con información relativa a la combinación computador-navegador-usuario que se almacena de forma local.
Cortafuegos:: Elemento de prevención que realizará un control de acceso con el objetivo de separar nuestra red de los equipos del exterior (potencialmente hostiles). En inglés, firewall.
Descifrado:: Transformación inversa al cifrado para obtener el texto en claro a partir del texto cifrado y la clave de descifrado.

DHCP :: Protocolo que permite a los clientes de una red IP obtener sus parámetros

de configuración. Del inglés (Dynamic Host Configuration Protocol).
Dirección MAC :: Es un identificador de 48 bits (6 bloques hexadecimales) que corresponde de una forma única a una tarjeta o dispositivo de red.
DNS :: Sistema de nomenclatura jerárquica para computadoras, servicios o cualquier recurso conectado a Internet o una red privada, del inglés Domain Name System.
Encaminador con filtrado de paquetes:: Dispositivo de red que encamina tráfico TCP/IP sobre la base de una serie de reglas de filtrado que deciden qué paquetes se encaminan a través suyo y cuáles son descartados.
Equipo bastión:: quipo que ha sido fuertemente protegido para soportar los supuestos ataques desde un lugar hostil y que actúa como punto de contacto entre el interior y el exterior de una red.
Extranet :: Red privada que utiliza protocolos de Internet, protocolos de comunicación y probablemente infraestructura pública de comunicación para compartir de forma segura información propia de una organización.
Firma digital:: Valor calculado a partir de un texto con una clave privada y que puede ser comprobado con la correspondiente clave pública, lo cual permite confirmar que solamente lo puede haber generado el poseedor de la clave privada.
Framework :: Conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar.
Hacker :: Gente apasionada por la seguridad informática. Esto concierne principalmente a entradas remotas no autorizadas por medio de redes de comunicación como Internet ("Black hats"). Pero también incluye a aquellos que depuran y arreglan errores en los sistemas ("White hats") y a los de moral ambigua como son los "Grey hats".
Host :: Se refiere a cada una de las computadoras conectadas a una red que proveen o utilizan servicios de ella.
HTTP :: Es el protocolo utilizado en cada transacción de la World Wide Web, del inglés Hyper Text Transfer Protocol.
HTTPS :: Protocolo seguro de transferencia de hipertexto, del inglés Hyper Text Transfer Protocolo Secure.
ICMP :: Es el subprotocolo de control y notificación de errores del Protocolo de Internet (IP), del inglés Internet Control Message Protocol).
Intranet :: Red de ordenadores privados que utiliza la tecnología Internet para compartir dentro de una organización parte de sus sistemas de información y sistemas operacionales.
IP :: Se trata de un protocolo no orientado a conexión, usado por el origen y el destino para comunicación de datos a través de una red de paquetes conmutados, del inglés Internet Protocol.
IPSec :: Es un conjunto de protocolos cuya función es asegurar las comunicaciones sobre el protocolo IP autenticando y/o cifrando cada paquete IP.
Kernel :: Se refiere al núcleo de un sistema operativo.
Linux :: Núcleo libre de sistema operativo basado en Unix.
Log :: Registro de eventos que se producen durante un rango de tiempo en particular.
Malware :: Es un tipo de software que tiene como objetivo infiltrase o dañar una computadora sin el consentimiento del propietario.
Memoria RAM :: Memoria utilizada como memoria de trabajo para el software instalado en un ordenador, del inglés Random-Access Memory.
OSI :: Es un modelo de red descriptivo creado por la Organización Internacional

para la Estandarización.
OSX :: Sistema operativo desarrollado y comercializado por Apple Inc.
Pasarela a nivel de aplicación:: Dispositivo de red que actúa como retransmisor a nivel de aplicación.
Pasarela a nivel de circuito:: Similar a una pasarela a nivel de aplicación en cuanto a la conexión, pero operando de manera similar a un filtro de paquetes a nivel de red (una vez que la conexión ha sido inicializada).
Política de seguridad:: Resultado de documentar las expectativas de seguridad de una red, tratando de plasmar en el mundo real los conceptos abstractos de seguridad.
PPC :: Nombre original de la arquitectura de computadoras de tipo RISC que fue desarrollada por IBM, Motorola y Apple.
PYME :: Acrónimo de Pequeña y Mediana Empresa
Router :: Dispositivo usado para la interconexión de redes informáticas que permite asegurar el direccionamiento de paquetes de datos entre ellas o determinar la mejor ruta que deben tomar.
Script :: Archivo de órdenes o archivo de procesamiento por lotes, es un programa usualmente simple que por lo regular se almacena en un archivo de texto plano.
Seguridad perimetral:: Seguridad basada únicamente en la integración en la red de sistemas cortafuegos y otros mecanismos de control de acceso.
Servidor intermediario:: Servidor software que se encarga de realizar las conexiones solicitadas con el exterior y retransmitirlas hacia el equipo que inició la conexión. En inglés, proxy.
SFTP :: Versión segura del protocolo de transferencia de ficheros, del inglés Secure File Transfer Protocol.
Shell :: Programas que proveen una interfaz de usuario para acceder a los servicios del sistema operativo.
Solaris :: Sistema operativo de tipo Unix desarrollado inicialmente por Sun Microsystems y actualmente por Oracle Corporation.
TCP :: Protocolo de Control de Transmisión, del inglés Transmission Control Protocol.
UDP :: Protocolo del nivel de transporte basado en el intercambio de datagramas, del inglés User Datagram Protocol.
Unix :: Sistema operativo portable, multitarea y multiusuario.
Virus :: Tipo de malware que tiene como objetivo el alterar el normal funcionamiento de una computadora.
VPN :: Red privada virtual, del inglés Virtual Private Network.
Wi-fi :: Mecanismo de conexión de dispositivos electrónicos de forma inalámbrica.
x86 :: Arquitectura de procesadores utilizada habitualmente en ordenadores domésticos.
x86-64:: Es una arquitectura basada en la extensión del conjunto de instrucciones x86 para manejar direcciones de 64 bits.
XML :: Se trata de un lenguaje de marcas desarrollado por el World Wide Web Consortium, del inglés eXtensible Markup Language.
Zona desmilitarizada:: Dentro de una red protegida por un cortafuegos, zona separada de los servidores públicos por un segundo cortafuegos.

12. Bibliografía

http://es.wikipedia.org/wiki/Red_de_%C3%A1rea_local

https://launchpad.net/deja-dup

http://es.wikipedia.org/wiki/Ubuntu

http://es.wikipedia.org/wiki/Fedora_(distribuci%C3%B3n_Linux)

http://es.wikipedia.org/wiki/BackTrack

http://en.wikipedia.org/wiki/Comparison_of_Linux_distributions

http://www.tress.com.mx/boletin/julio2003/firewall.htm

http://es.wikipedia.org/wiki/Cortafuegos_(inform%C3%A1tica)

http://www.guia-ubuntu.com/index.php?title=Cortafuegos

http://dns.bdat.net/documentos/cortafuegos/t1.html

http://personals.ac.upc.edu/elara/documentacion/LINUX%20-%20UD12%20-%20Configuracion%20de%20iptables%20en%20Linux.pdf

www.snort.org

http://www.snort.org/docs/writing_rules/

http://www.snort.org/docs/lisapaper.txt

http://www.snort.org/docs/idspaper/

http://www.guia-ubuntu.com/?title=Servidor_ssh

http://www.imh.es/es/comunicacion/dokumentazio-irekia/manuales/instalar-un-servidor-ssh-en-linux/referencemanual-all-pages

http://www.howtoforge.com/intrusion_detection_snort_base_postgresql_ubuntu6.06

http://www.howtoforge.com/intrusion-detection-with-snort-mysql-apache2-on-ubuntu-7.10

http://www.linux-party.com/index.php/57-seguridad/7980-20-ejemplos-iptables-para-administradores-de-sistemas-linux

http://usemoslinux.blogspot.com/2011/03/como-configurar-el-firewall-en-ubuntu.html

http://es.wikipedia.org/wiki/LDAP

http://es.blogxpopuli.org/wiki/LDAP

http://es.wikipedia.org/wiki/Squid_%28programa%29

http://www.squid-cache.org/

http://www2.idesoft.es/squid/

http://www2.idesoft.es/squid/manual.php

http://www.gulix.cl/wiki/Proxy_squid