



CentOS

Manual de configuración servidores en CentOS 5.6 Final.

CentOS es un sistema Operativo Libre de tipo server bajo una plataforma Unix/Linux de 64 bit que nos permite el ensayo y configuración de servicios tales como Samba, Apache, Mysql, DHCP, Squit, DNS, Openssh que serán parte de los servicios abordados en este documento.

Ileana Patricia Loaisiga Hernández.
Carlos Eduardo Barahona.
23/11/2011





Attribution-Share Alike 3.0 Unported

You are free:



to Share — to copy, distribute and transmit the work



to Remix — to adapt the work



Under the following conditions:



Attribution — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).



Share Alike — If you alter, transform, or build upon this work, you may distribute the resulting work only under the same, similar or a compatible license.



Contenido

Introducción	4
Servidor de compartimiento de Archivos 'Samba'	5
Repositorio configuración.	6
Configuración Servicio Samba	7
Servidor de protocolo comunicación y conexión 'Openssh'	13
Configuración servicio Openssh.....	14
Servidor HTTP o APACHE	21
Configuración del servicio HTTP o APACHE	21
Servidor Mysql y CMS joomla	25
Configuración de Mysql y CMS joomla.....	25
Configuración del CMS vía web.....	30
Servidor DHCP.....	36
Configuración del servidor DHCP	37
Servidor DNS.....	44
Configuración del servicio DNS	45
Servicio Proxy o Squid	52
Configuración servicio Squid.....	53
Conclusiones.....	61
Bibliografía.....	62



Introducción

El presente manual fue elaborado en transcurso del curso de Facultativa II materia que se imparte para dar un conocimiento general de forma introductoria en a la administración de servidores basados en software libre que hoy en día representa mas del 85% de usabilidad por el entorno de seguridad que estos ofrecen al momento de sus configuraciones. Los servidores como bien su nombre lo indica es una plataforma que se crea para dar servicios diversos de acuerdo a la necesidades de las empresas, instituciones, ONG, etc.

Se trata de introducir a usuarios medios avanzados en la administración de dichos servidores, los cuales cumplen tareas diversas como ya se explicaran uno a uno en este documento. El manual alcanza las configuraciones básicas necesarias para una pequeña empresa que va desde la asignación de direcciones IP, accesos a bases de datos y navegación internet así como algunas restricciones típicas que se le asignan a los ordenadores clientes de pequeñas redes o Grandes si es el caso.

Servidor de compartimiento de Archivos 'Samba'

Samba es una suite de aplicaciones UNIX, que habilita el protocolo SMB (Server Message Block). Fue creado por IBM, en 1985.

Samba fue desarrollado originalmente para Unix por Andrew Tridgell utilizando un sniffer o capturador de tráfico para entender el protocolo a través de la ingeniería inversa. Actualmente es mantenido por un grupo de personas de todo el mundo, (The SAMBA Team), bajo la licencia pública general (GLP) que permite a los sistemas basados en UNIX como GNU/LINUX, el intercambio de archivos en una red de datos con sistemas operativos Windows.

Samba se puede ejecutar en las distribuciones GNU/LINUX, Solaris y las diferentes variantes BSD entre las que podemos encontrar el Mac OS X Server de Apple.

Beneficios de utilizar el servicio samba

- ✓ Compartir uno o más sistemas de archivo.
- ✓ Compartir impresoras instaladas tanto en el servidor como en los clientes.
- ✓ Permite compartir recursos entre máquinas Windows y Linux (siendo un recurso, una carpeta o la impresora).
- ✓ Autenticar clientes logiando contra un dominio Windows.

Instalación de samba y demás servidores

Para configurar el servidor de archivos samba y demás servicios primero deberemos de configurar algunos parámetros de forma grafica para poder tener acceso a las instalaciones de cada uno de ellos.

El proceso es necesario dado que los archivos a configurar se encuentran en el DVD o disco que utilizamos para la instalación del mismo sistema operativo server CentOS 5.6 Final ah este proceso se le denomina Repositorio que nos permite no confundir o dañar por así decir el sistema central que inicialmente instalaron en su ordenador.

Repositorio configuración.

1. Ubicarse en el menú **Applications**.
2. Buscar y seleccionar la opción **Add/remove software** por lo general se encuentra al final del menú **Applications**.
3. Se desplegara una ventana con muchas opciones de las cuales no ocuparemos ni una de estas pero si el menú **Edit** que se encuentra en esta ventana seleccionamos la opción **Repositories**.
4. Se despliega la ventana **Repository Manager**.
 - 4.1 Desmarcar todos y cada uno de los **Repositories** que aparecen en la ventana como marcados.
 - 4.2 Hacer clic en el botón **Add** se desplegara una nueva ventana similar ah esta donde configuraremos nuestro nuevo Repositorio con lo siguientes parámetros:

Name:	CentOS
Description:	Repositorio CentOS
<input checked="" type="radio"/> Location:	file:///media/CentOS_5_6_Final
<input type="radio"/> Mirror List:	
<input type="checkbox"/> Advanced:	
<input checked="" type="checkbox"/> GPG Key:	file:///media/CentOS_5_6_Final/RPM-GPG-KEY-CentOS-5

- 4.5 Clic en Ok y listo ya hemos creado nuestro repositorio solo debemos cerciorarnos que solo este repositorio este seleccionado en la lista **Repositories**.
5. Cerrar todas las ventanas y solo basta recordar que siempre que instalemos algún archivo del disco o DVD tenerlo ingresado para que haya problemas.



6. Instalamos nuestros primeros paquetes pertenecientes a nuestro nuevo repositorio a través de comandos de consola o terminal en el siguiente orden:

6.1 **yum -y install iptraf**

6.2 **yum -y install mc**

6.3 **yum repolist**

7. Listo ahora si ya tenemos completo nuestro repositorio.

Configuración Servicio Samba

Este servicio permite compartir archivos en una red local y asignar permisos adecuados a cada uno de ellos, según sea la necesidad requerida por el administrador de red.

1. Para Instalar Samba primero ejecutaremos el comando.

```
yum install samba
```

2. Se creará una copia de respaldo del archivo de configuración **smb.conf** ubicado en el directorio **/etc/samba/** a la cual le pondremos el nombre de **smb.conf.org**

```
cd /etc/samba/  
cp smb.conf smb.conf.org
```

3. Verificamos que el archivo de respaldo se haya creado con el comando **ls** si se creó el archivo de respaldo exitosamente procederemos a editar el archivo origen **smb.conf**

```
vim smb.conf
```

3.1 Primeros parámetros ah editar en el archivo de configuración:

```
workgroup = grupo8
server string = Servidor de archivos %v
netbios name = Soportado por CentOS
interfaces = lo eth0 10.1.25.0/24
```

Ver figura smb1

```
File Edit View Terminal Tabs Help
#----- Global Settings -----
[global]
# ----- Network Related Options -----
#
# workgroup = NT-Domain-Name or Workgroup-Name, eg: MIDEARTH
#
# server string is the equivalent of the NT Description field
#
# netbios name can be used to specify a server name not tied to the hostname
#
# Interfaces lets you configure Samba to use multiple interfaces
# If you have multiple network interfaces then you can list the ones
# you want to listen on (never omit localhost)
#
# Hosts Allow/Hosts Deny lets you restrict who can connect, and you can
# specify it as a per share option as well
#
workgroup = grupo8
server string = Servidor de Archivo %v
netbios name = Soportado por CentOS
interfaces = lo eth0 10.1.25.0/24
; hosts allow = 127. 192.168.12. 192.168.13.
```

Figura smb1

3.2 segundos parámetros ah editar o ah verificar:

```
# Tipo de seguridad, compartida.
security = share
```

Ver figura smb2

```
# use either tdbsam or ldapsam. smbpasswd is available for backwards
# compatibility. tdbsam requires no further configuration.
security = share
passdb backend = tdbsam
```

Figura smb2

3.3 Guardamos el archivo de configuración y lo cerramos temporalmente.

4. Se creara un directorio de Nombre **archivo** en el usuario **dmiller**

```
mkdir /home/dmiller/archivo
```

5. Procederemos ah editar nuevamente el archivo de configuración **smb.conf**

```
vim /etc/samba/smb.conf
```

6. Editaremos los parámetros de opciones de impresión:

```
load printers = yes  
cups options = raw
```

7. Agregaremos algunos parámetros que nos permitirá visualizar los archivos colocados en el directorio **archivo** que creamos en el usuario dmiller

```
[archivo]  
comment = Archivo compartidos  
path = /home/dmiller/archive  
guest ok = yes  
create mask = 0755  
directory mask = 0755  
read only = yes  
writable = no
```

7.1 Estas líneas se agregan de ultimo si no quieres un archivo con mucho comentario puedes limpiarlo para que te quede como en la figurara smb3, después de la ultima línea no puede haber espacio para que no produzca un error a la hora de iniciar el servicio.

```
root@localhost:~  
[global]  
# ----- Network Related Options -----  
#  
workgroup = grupo8  
server string = Servidor de Archivo %v  
netbios name = Soportado por CentOS  
  
interfaces = lo eth0 10.1.25.0/24  
; hosts allow = 127. 192.168.12. 192.168.13.  
  
# ----- Standalone Server Options -----  
#  
security = share  
passdb backend = tdbsam  
  
# ----- Printing Options -----  
  
load printers = yes  
cups options = raw  
[archivo]  
comment = Archivo compartidos  
path = /home/dmiller/archivo  
  
guest ok = yes  
  
create mask = 0755  
directory mask = 0755  
  
read only = yes  
writable = no
```

Figura smb3

8. Para finalizar e iniciar el servicio ejecutamos la siguiente línea de comando.

```
setsebool -P samba_enable_home_dirs 1
```

9. Procedemos a levantar o iniciar el servicio samba para ello ejecutaremos los siguientes comandos:

```
service smb start  
chkconfig smb on
```

En la figura smb4 se muestra el correcto funcionamiento del levantado del servicio.

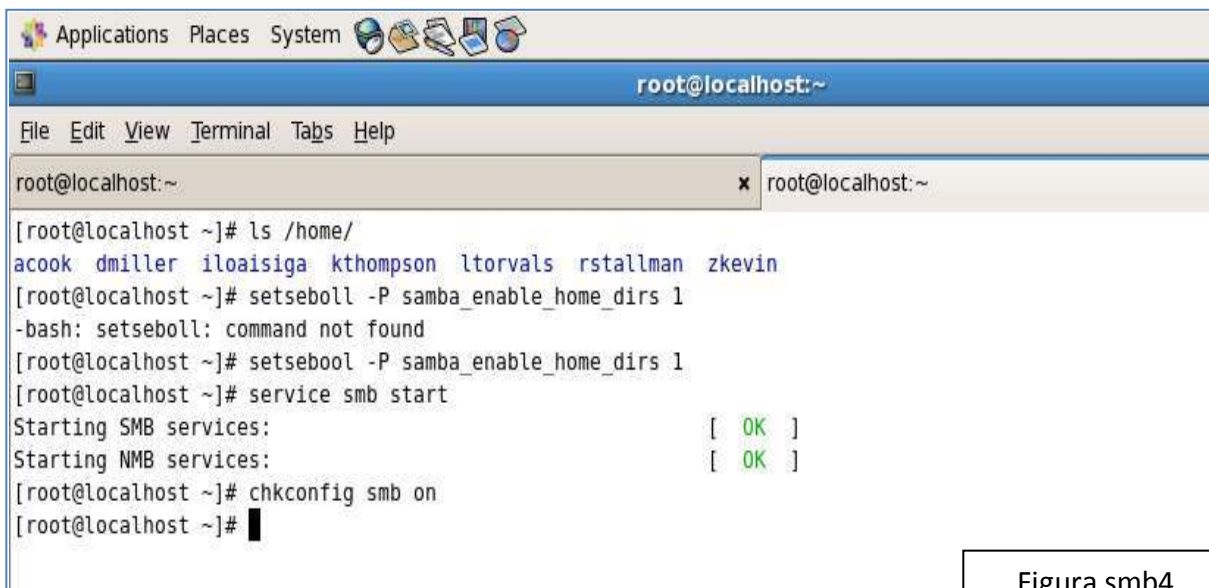


Figura smb4

Recordemos de forma grafica abrir el puerto del servicio samba en el menú **System >> Administration >> Firewall and security** como se muestra en la figura smb5.



Figura smb5

Para comprobar que el servidor Samba está activo. Podemos acceder a través de un cliente Windows, Haciendo clic en el botón inicio y seleccionando la opción ejecutar, digite la dirección IP del servidor y haga clic en aceptar en nuestro caso estamos dentro de una red privada así que nuestra dirección IP es privada y solo los que pertenecen a ella tienen acceso a este servidor de archivos.

\\10.1.25.126

Aparecerá una pantalla mostrada en la figura smb6 en donde podemos visualizar los directorios que contiene el servidor Samba. A los cuales el cliente podrá tener acceso si así lo deseamos.

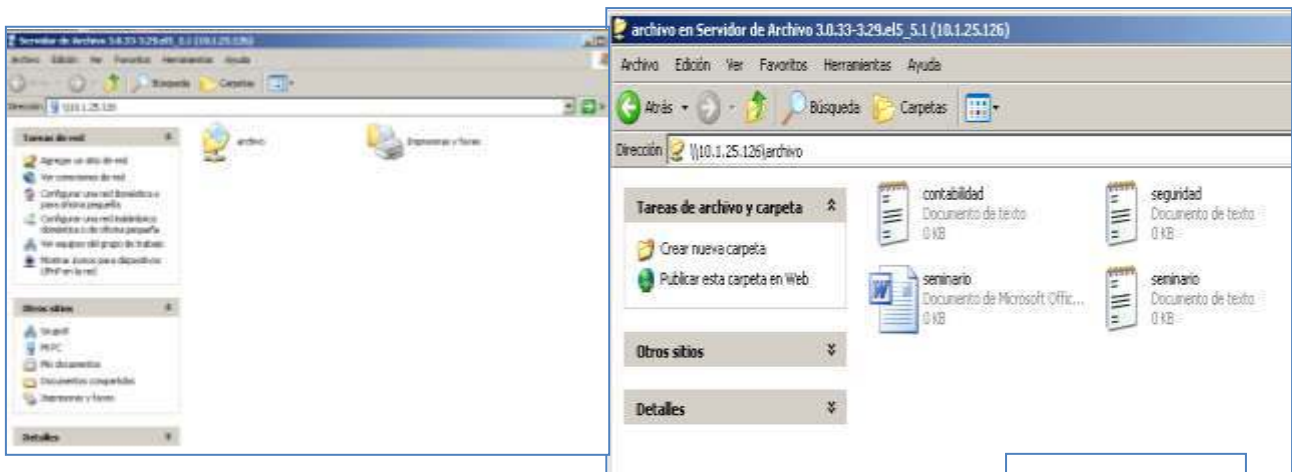


Figura smb6

Nota: Si los archivos no se muestran o no tienes acceso de la forma deseada revisa parámetro por parámetro el archivo de configuración, así como cerciorarse que los archivos tengan los mismos permisos que los otorgados en el archivo de configuración.



Servidor de protocolo comunicación y conexión 'Openssh'

Openssh (Open Secure Shell) es una alternativa de código abierto, con licencia BSD. Se considera más segura que su contraparte propietaria debido a la constante auditoría que se realiza sobre el código fuente por parte de una gran comunidad de desarrolladores, una ventaja que brinda al tratarse de un proyecto de fuente abierta. Openssh incluye servicio y clientes para los protocolos SSH, SFTP y SCP.

Beneficios de usar Openssh:

- ✓ Cifra todo el tráfico (incluyendo contraseñas) para eliminar efectivamente las escuchas, secuestro de conexiones y otros ataques.
- ✓ Ofrece la creación de túneles seguro
- ✓ Varios métodos de autenticación y Soporta todas las versiones del protocolo SSH.

Si bien cuando instalamos el Sistema Operativo Server CentOS también instalamos por defecto el protocolo de conexión **ssh** este es muy inseguro dado que las conexiones remotas no son con claves o contraseñas encriptadas como se nos mostrara mas adelante Openssh no hace más que configurar el nivel de seguridad de ssh.

Configuración servicio Openssh

1. Se instala a través de comando de consola el paquete Openssh o más bien el archivo de configuración del servicio Openssh dado que el paquete ya se encuentra instalado como veremos en la figura openssh1.

```
yum -y install openssh-server
```

```
root@localhost:~  
File Edit View Terminal Tabs Help  
[root@localhost ~]# yum -y install openssh-server  
Loaded plugins: fastestmirror  
Loading mirror speeds from cached hostfile  
Setting up Install Process  
No package openssh-server available.  
Nothing to do  
[root@localhost ~]# █
```

Figura openssh1

2. Crearemos una copia de respaldo del archivo de configuración **sshd_config** y lo guardaremos con el nombre de **sshd_config.org** como se muestra en la figura openssh2.

```
cp /etc/ssh/sshd_config /etc/ssh/sshd_config.org
```

```
root@localhost:~  
File Edit View Terminal Tabs Help  
[root@localhost ~]# cp /etc/ssh/sshd_config /etc/ssh/sshd_config.org █
```

Figura openssh1

3. Abrimos el archivo de configuración **sshd_config** para poder editar los parámetros que nos permitirán seguridad en el protocolo de comunicación **openssh** con el siguiente comando.

```
vim /etc/ssh/sshd_config
```

4. Editaremos los parámetros en protocolo 2 como se nos pide a continuación solo esos parámetros serán los que se editar sin cambiar algún otro como aparece en la figura openssh3:

```
Protocol 2  
ServerKeyBits 1024  
LoginGraceTime 2m  
PermitRootLogin no  
StrictModes yes  
MaxAuthTries 4  
RSAAuthentication no  
PubKeyAuthentication yes  
AuthorizedKeysFile .ssh/authorized_keys
```

```
Applications Places System 
root@localhost:~
File Edit View Terminal Tabs Help
# possible, but leave them commented. Uncommented options change a
# default value.
#Port 22
#Protocol 2,1
Protocol 2
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
# HostKey for protocol version 1
#HostKey /etc/ssh/ssh_host_key
# HostKeys for protocol version 2
#HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
# Lifetime and size of ephemeral version 1 server key
#KeyRegenerationInterval 1h
ServerKeyBits 1024
# Logging
# obsoletes QuietMode and FascistLogging
#SyslogFacility AUTH
SyslogFacility AUTHPRIV
#LogLevel INFO
# Authentication:
LoginGraceTime 2m
PermitRootLogin no
StrictModes yes
MaxAuthTries 4
RSAAuthentication no
PubkeyAuthentication yes
AuthorizedKeysFile .ssh/authorized_keys
# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
```


5. Otros parámetros a editar son los siguientes tener mucho cuidado como siempre al momento de editarlo:

```
PasswordAuthentication no  
PermitEmptyPasswords no  
PasswordAuthentication no  
AllowUsers usuario1 usuario2 usuario N
```

En este parámetro se agregan los usuarios que uno desee tengan acceso al servidor claro debidamente creados en el servidor en nuestro caso el usuario será iloaisiga como veremos en la figura openssh4.

```
# To disable tunneled clear text passwords, change to no here!  
PasswordAuthentication no  
PermitEmptyPasswords no  
PasswordAuthentication no  
AllowUsers iloaisiga  
# Change to no to disable s/key passwords  
#ChallengeResponseAuthentication yes  
ChallengeResponseAuthentication no
```

Figura openssh4

6. Especifica si se permite hacer uso del X11 Forwarding, esto para ejecutar aplicaciones gráficas en el servidor como se muestra en la figura openssh5.

```
# Accept locale-related environment variables  
AcceptEnv LANG LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY LC_M  
AcceptEnv LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT  
AcceptEnv LC_IDENTIFICATION LC_ALL  
#AllowTcpForwarding yes  
#GatewayPorts no  
#X11Forwarding no  
X11Forwarding no  
#X11DisplayOffset 10  
#X11UseLocalhost yes  
-- INSERT --
```

Figura openssh5

7. Cerramos y guardamos el archivo de configuración y seguiremos los siguientes pasos:

- 7.1 nos cambiamos de usuario root ah usuario simple en este caso al usuario que le brindarás los permisos de conexión remota como se muestra en la figura openssh6.

```
[root@localhost ~]# vim /etc/ssh/sshd_config  
[root@localhost ~]# su - iloaisiga  
[iloaisiga@localhost ~]$
```

Ileana

Carlos Eduardo Barahona.

Figura openssh6

7.2 Luego de cambiar de usuario o cuenta procederemos a crear las claves por las cuales tendremos acceso remotamente al servidor que luego se nos mostrara en la figura openssh7 y se explicara algunas cosas que se hacen.

```
ssh-keygen -t dsa -b 1024
```

```
[iloaisiga@localhost ~]$ ssh-keygen -t dsa -b 1024
Generating public/private dsa key pair.
Enter file in which to save the key (/home/iloaisiga/.ssh/id_dsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/iloaisiga/.ssh/id_dsa.
Your public key has been saved in /home/iloaisiga/.ssh/id_dsa.pub.
The key fingerprint is:
33:2b:9e:aa:2d:2e:a4:6b:e9:3e:a4:87:a8:27:33:8a iloaisiga@localhost.localdomain
[iloaisiga@localhost ~]$
```

Enter después del comando se nos muestran unas líneas que nos dicen que está creando un directorio oculto en el usuario iloaisiga que es donde se guardarán las llaves, damos un enter.

En este momento nos pide una frase que será la que generará la clave privada y pública del usuario y que se alojarán en el directorio `/home/iloaisiga/.ssh/` con los nombres `id_dsa` y `id_dsa.pub` respectivamente hasta ahí se crean nuestras archivos encriptados por así decirlo.

Figura openssh7

7.3 Se procede a ubicarse en el directorio oculto `.ssh/` donde crearemos con otro nombre y utilizando los archivos creados nuestro único archivo que posea los privilegios de verificación de claves de acceso como se muestra en la figura openssh8.

```
cd .ssh/
ls
cat id_dsa.pub >> authorized_keys
ls
```

```
[iloaisiga@localhost ~]$ ssh-keygen -t dsa -b 1024
Generating public/private dsa key pair.
Enter file in which to save the key (/home/iloaisiga/.ssh/id_dsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/iloaisiga/.ssh/id_dsa.
Your public key has been saved in /home/iloaisiga/.ssh/id_dsa.pub.
The key fingerprint is:
33:2b:9e:aa:2d:2e:a4:6b:e9:3e:a4:87:a8:27:33:8a iloaisiga@localhost.localdomain
[iloaisiga@localhost ~]$ cd .ssh/
[iloaisiga@localhost .ssh]$ ls
id_dsa id_dsa.pub
[iloaisiga@localhost .ssh]$ cat id_dsa.pub >> authorized_keys
[iloaisiga@localhost .ssh]$ ls
authorized_keys id_dsa id_dsa.pub
[iloaisiga@localhost .ssh]$ █
```

Figura openssh8

7.4 Luego de que creamos nuestro archivo **authorized_keys** procederemos a mover los otros dos archivos tanto el **id_dsa** como el **id_dsa.pub** para que solo nos quede el archivo de autorización ¿Por qué hacer esto? Simple en el archivo que configuramos el servicio openssh se editaron algunos parámetros donde se especifica la ruta de búsqueda del archivo de autorización del usuario al encontrar otros archivos en la ruta se toma como vulnerable o posible incompatibilidad con lo que busca al momento de la conexión.

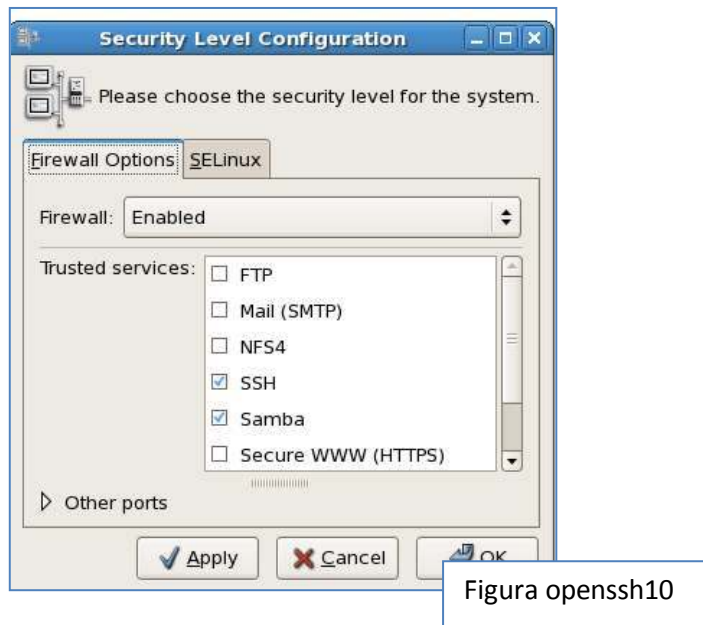
```
mv id_dsa /home/iloaisiga/
mv id_dsa.pub /home/iloaisiga/
```

7.5 Cuando este solo el archivo **authorized_keys** cambiar los permisos a 755 como se mostrara en la Figura openssh9.

```
[iloaisiga@localhost .ssh]$ ls
id_dsa id_dsa.pub
[iloaisiga@localhost .ssh]$ cat id_dsa.pub >> authorized_keys
[iloaisiga@localhost .ssh]$ ls
authorized_keys id_dsa id_dsa.pub
[iloaisiga@localhost .ssh]$ mv id_dsa /home/iloaisiga/
[iloaisiga@localhost .ssh]$ ls
authorized_keys id_dsa.pub
[iloaisiga@localhost .ssh]$ mv id_dsa.pub /home/iloaisiga/
[iloaisiga@localhost .ssh]$ ls
authorized_keys
[iloaisiga@localhost .ssh]$ chmod 755 authorized_keys
[iloaisiga@localhost .ssh]$ █
```

Figura openssh9

7.6 Para ir finalizando nos cercioramos que los puertos de ssh o 22 estén abiertos en el menú **System >> Administration >> Firewall and security** como se muestra en la figura openssh10.



7.7 Ya con eso solo nos queda levantar o levantar los servicios del **openssh**.

```
service sshd restart  
chkconfig sshd on
```

8. Probaremos si funciona desde otro ordenador a la red que pertenecemos en nuestro caso lo haremos con una distribución Linux CentOS puede ser cualquier distribución Linux que hacer sigamos los siguientes pasos.

8.1 Primero ingresar una memoria USB en el servidor en donde copiaremos el archivo **id_dsa** que en pasos anteriores lo reubicamos en el directorio **/home/iloaisiga/**, en nuestro caso conectaremos una memoria que lleva el nombre de **ILEANA** que por defecto se monta en el directorio **/media/**

```
cp /home/iloaisiga/id_dsa /media/ILEANA/id_dsa
```

8.2 Luego de revisar y copiar el archivo en la memoria USB esta se desmonta y la llevamos al otro ordenador con sistema operativo Linux no importa la distribución porque en todos contamos con un Shell o terminal donde podemos utilizar comandos en nuestro caso lo aremos con un sistema CentOS como se había mencionado en pasos anteriores para ello ejecutaremos las siguientes sintaxis como se muestra en la figura openssh11:

```
cd /media/ILEANA
ls -al id_dsa
ssh -i id_dsa iloaisiga@10.1.25.126
```

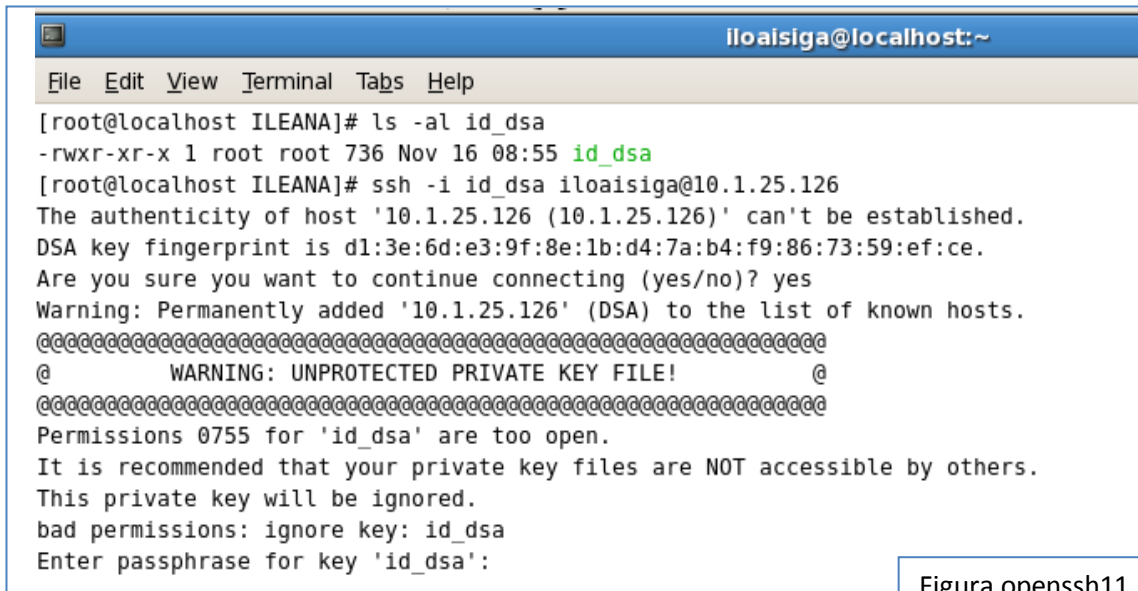


Figura openssh11

Al intentar hacer la conexión nos muestra un mensaje de advertencia con respecto al archivo **id_dsa** en la cual nos explica que no cuenta con los permisos apropiados para continuar con la conexión que hacer para resolverlo, como es problema es permisos del archivo **id_dsa** se los cambiaremos ah **700** como se muestra en la figura openssh12 y listo tenemos configurado con éxito nuestro servidor Openssh.

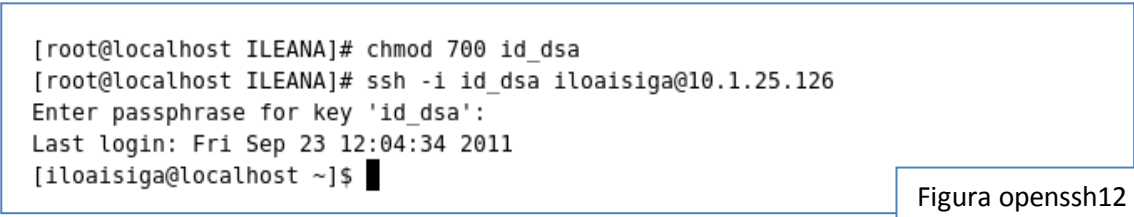


Figura openssh12

Servidor HTTP o APACHE

Apache es un servidor HTTP (HyperText Transfer Protocol, es el protocolo usado en cada transacción de la Web), es de código abierto y licenciamiento libre, funciona en Linux, sistemas operativos derivados de Unix, Windows y otras plataformas. Es el servidor HTTP más utilizado; es desarrollado y mantenido por una comunidad de desarrolladores auspiciada por Apache Software Foundation.

Configuración del servicio HTTP o APACHE

1. Para dar inicio a la configuración primero instalamos los paquetes o el paquete necesario que nos permitirá su configuración, como siempre se recuerda ingresar el DVD del sistema CentOS antes de ejecutar cualquier comando que necesite del mismo para ahorrarnos algún posible problema ejemplo en la figura apache1.

```
yum install httpd
```

```
Applications Places System [Icons] [Network] [Power]
File Edit View Terminal Tabs Help
[root@localhost ~]# yum install httpd
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
Setting up Install Process
Package httpd-2.2.3-45.el5.centos.i386 already installed and latest version
Nothing to do
[root@localhost ~]#
```

Figura apache1

2. El archivo de configuración del **http** se encuentra en el directorio **/etc/httpd/conf/** de donde aremos una copia de seguridad del archivo **httpd.conf** y la guardaremos con el nombre **httpd.conf.org** para ello utilizaremos la siguiente línea de comando y también puede observa la figura apache2 al mismo tiempo nos aseguramos que la copia del archivo se haya creado en el directorio deseado.

```
cp /etc/httpd/conf/httpd.conf /etc/httpd/conf/httpd.conf.org  
ls /etc/httpd/conf/
```

```
Applications Places System  
[root@localhost ~]# cp /etc/httpd/conf/httpd.conf /etc/httpd/conf/httpd.conf.org  
[root@localhost ~]# ls /etc/httpd/conf/  
httpd.conf httpd.conf.org magic  
[root@localhost ~]#
```

Figura apache2

3. Realizada y verificada la copia del archivo de configuración **httpd.conf** procederemos a editar los parámetros necesarios de el servidor luego de editar el archivo guardamos los cambios.

```
vim /etc/httpd/conf/httpd.conf
```

```
ServerTokens Prod  
KeepAlive Off  
UseCanonicalName Off  
AllowOverride None  
UserDir disabled  
UserDir disabled root  
DirectoryIndex index.html index.htm index.jsp index.php index.html.var  
ServerSignature Off
```

4. Una vez guardado el archivo de configuración procederemos ah iniciarlos no obstante seguiremos instalando otros paquetes que necesitamos como se muestra en la figura apache3.

```
service httpd start
chkconfig httpd on
```

```
root@localhost:~
File Edit View Terminal Tabs Help
[root@localhost ~]# service httpd start
Starting httpd: [ OK ]
[root@localhost ~]# chkconfig httpd on
```

Figura apache3

5. Los otros archivos o paquetes que necesitamos son los que nos permiten que los formatos **php** y **Mysql** sean admitidos o soportados por el **http** atraves de su navegación los cuales serán **php**, **php-mysql** y **php-gd** que se los mostraremos en las figuras apache4, 5 y 6 respectivamente.

```
yum install php
yum install php-mysql
yum install php-gd
```

```
root@localhost:~
File Edit View Terminal Tabs Help
[root@localhost ~]# service httpd start
Starting httpd: [ OK ]
[root@localhost ~]# chkconfig httpd on
[root@localhost ~]# yum install php
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
Setting up Install Process
Package php-5.1.6-27.el5_5.3.i386 already installed and latest version
Nothing to do
[root@localhost ~]#
```

Figura apache4


```
root@localhost:~# yum install php-mysql
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package php-mysql.i386 0:5.1.6-27.el5_5.3 set to be updated
--> Processing Dependency: php-pdo for package: php-mysql
--> Running transaction check
--> Package php-pdo.i386 0:5.1.6-27.el5_5.3 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

Package Arch Version Repository Size
Installing:
php-mysql i386 5.1.6-27.el5_5.3 CentOS 86 k
Installing for dependencies:
php-pdo i386 5.1.6-27.el5_5.3 CentOS 65 k

Transaction Summary
Install 2 Package(s)
Upgrade 0 Package(s)

Total download size: 151 k
Is this ok [y/N]: y
Downloading Packages:
Total 45 MB/s | 151 kB 00:00
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
Installing : php-pdo 1/2
Installing : php-mysql 2/2

Installed:
php-mysql.i386 0:5.1.6-27.el5_5.3
Dependency Installed:
php-pdo.i386 0:5.1.6-27.el5_5.3
```

Figura apache5

```
root@localhost ~# yum install php-gd
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package php-gd.i386 0:5.1.6-27.el5_5.3 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

Package Arch Version Repository Size
Installing:
php-gd i386 5.1.6-27.el5_5.3 CentOS 117 k

Transaction Summary
Install 1 Package(s)
Upgrade 0 Package(s)

Total download size: 117 k
Is this ok [y/N]: y
Downloading Packages:
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
Installing : php-gd 1/1

Installed:
php-gd.i386 0:5.1.6-27.el5_5.3

Complete!
root@localhost ~#
```

Figura apache6

6. Con la instalación de estos paquetes hemos finalizado la instalación del servicio **http** o **Apache**.

Servidor Mysql y CMS joomla

MYSQL, es actualmente el servidor de base de datos más popular para los desarrollos web. Es muy rápido y sólido, son muchos los administradores que lo instalan, y sin embargo no tantos los que lo configuran correctamente, o que tal vez no saben que hay que configurarlo.

Esta parte del manual es solo una **referencia rápida** para el procedimiento de instalación y configuración de un servidor MySQL™. La generación de una base de datos necesaria para el SMS joomla y configuración del mismo, esto se puede hacer a través de mandatos SQL en el Shell de MySQL™ lo cual lo demostraremos aquí, utilizando un fichero .SQL (como en el caso de PHP NUKE y otras aplicaciones web) o bien utilizando clientes MySQL™, como son [MySQLGUI](#), [GtkSQL](#) o [Gmysql](#).

Configuración de Mysql y CMS joomla

1. Primera mente hay que descargar el paquete de instalación de de **SMS joomla** ([Joomla_1.5.24-Spanish-pack_completo.tar.gz](http://www.joomlaspanish.org/Joomla_1.5.24-Spanish-pack_completo.tar.gz)) de donde podemos descargar o obtener en todo caso el paquete **SMS joomla** se obtiene de la pagina oficial del sitio <http://www.joomlaspanish.org/>
2. Una vez descargado el paquete procederemos a descomprimirlo en el directorio `/var/www/html`, es de importancia saber en donde tenemos el paquete porque primero haremos una copia de este en nuestro caso lo ubicamos en el escritorio o Desktop.

```
cp Desktop/Joomla_1.5.24-Spanish-pack_completo.tar.gz /var/www/html/
```

```
tar -zxvf /var/www/html/Joomla_1.5.24-Spanish-pack_completo.tar.gz
```

3. Una vez des-compreso visualizamos los archivos como lo muestran las figuras joomla1 y 2 respectivamente.

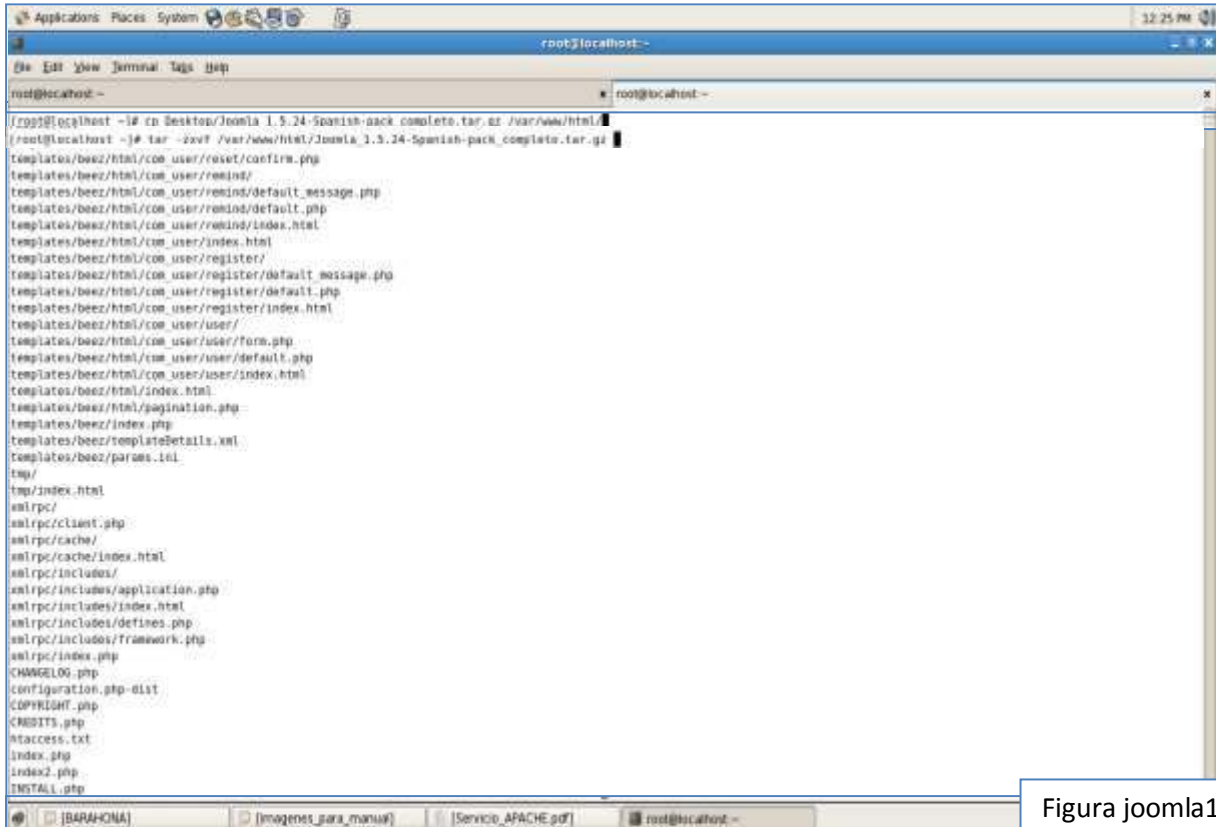


Figura joomla1



Figura joomla2

4. Listo una vez des-compresso el **SMS joomla** procederemos ah configurar un poco nuestro servicio **Mysql** para ello primero tenemos que instalar algunos paquetes desde el DVD de CentOS como son **mysql** y **mysql-server** como se muestran en las figuras mysql1 y 2 respectivamente a través de los siguientes comandos:

```
yum install mysql
yum install mysql-server
```

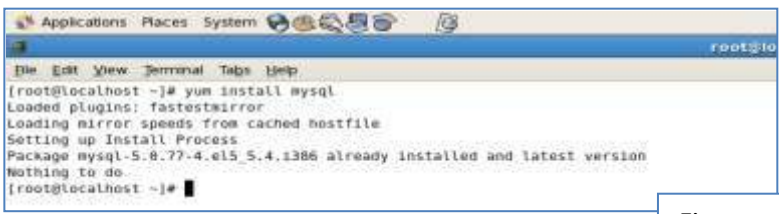


Figura mysql1

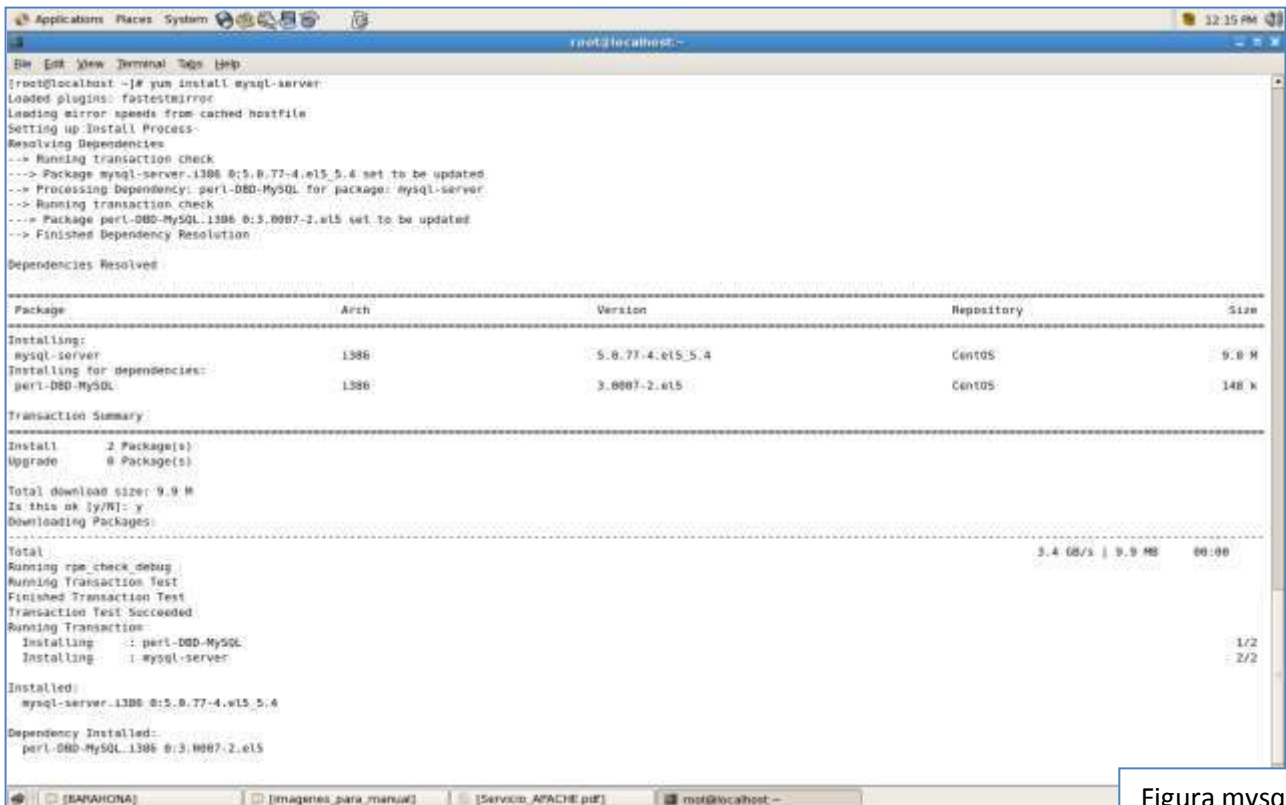


Figura mysql2

5. Una vez instalados los paquetes anteriores procederemos a levantar el servicio **mysql**.

```
service mysqld start  
chkconfig mysqld on
```

6. Listo una vez instalado e iniciados los servicios procederemos a la configuración de Mysql donde lo que aremos es crear una contraseña para poder tener acceso al Shell del servidor con el siguiente comando para esto hay que tener cuidado al momento de poner la contraseña o password dado que se enmarcan dentro de comías simples ('facultativa') como se muestra en la figura mysql3.

```
mysqladmin -u root password 'facultativa'
```



Figura mysql3

7. Ya configurada la parte de contraseña o password podremos obtener acceso a la Shell de Mysql con lo cual se logra con la siguiente línea de comando mostrada adelante y el funcionamiento en la figura mysql4.

```
mysql -u root -p
```

8. Después de digitalizar la línea de comando y teclear la tecla entrar el sistema nos solicitara la contraseña o password la cual deberemos de digitalizar la cual en nuestro caso es facultativa y listo ya tenemos acceso al Shell como se muestra en la figura mysql4.

```
[root@localhost ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.0.77 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> █
```

Figura mysql4

9. Nuestros siguientes pasos se mostraran en la siguiente línea de comandos de Mysql, lo que aremos es crear una bases de datos llamada joomla, dentro de dentro de esta crear un usuario de nombre (juser) y luego se otorgaran los privilegios todo esto se les mostrara en las siguiente líneas de pasos detallados de la mejor forma posible:

- 9.1 crear la bases de datos de nombre joomla figura mysql5.

```
create databases joomla;
```

```
mysql> create database joomla;
Query OK, 1 row affected (0.01 sec)
```

Figura mysql5

9.2 Crearemos un usuario dentro de la Mysql de nombre (juser) figura mysql6.

```
create user juser;
```

```
mysql> create user juser;  
Query OK, 0 rows affected (0.00 sec)
```

Figura mysql6

9.3 Asignaremos los privilegios a la bases de datos de nombre joomla con las siguiente línea de comando por factor de espacio se mostrara en varias líneas pero perfectamente se puede hacer en una sola línea como se muestra en la figura mysql7.

```
GRANT ALL ON joomla.* TO juser@localhost IDENTIFIED  
-> BY 'facultativa';
```

```
mysql> GRANT ALL ON joomla.* TO juser@localhost IDENTIFIED BY 'facultativa';  
Query OK, 0 rows affected (0.00 sec)
```

Figura mysql7

9.4 Realizados estos pasos procedemos ah salir del Shell Mysql lo que se logra con un comando exit demostrado en la figura mysql8.

```
mysql> exit  
Bye  
[root@localhost ~]# █
```

Figura mysql8

9.5 Para ir finalizando la instalación es necesario aplicar reglas de **setsebool** para los servicios apache y Mysql con las siguientes líneas de códigos mostradas esto con el fin de brindar los privilegios al usuario de la bases joomla.

```
setsebool httpd_enable_homedir on  
setsebool -P mysqld_disable_trans 1
```

10. Una vez finalizado hemos creado y otorgado permiso a la bases de datos joomla y procederemos ah su configuración vía web sin olvidar reiniciar los servicios **Mysql** y **Apache** respectivamente.

```
service httpd restart  
service mysqld restart
```

Configuración del CMS vía web

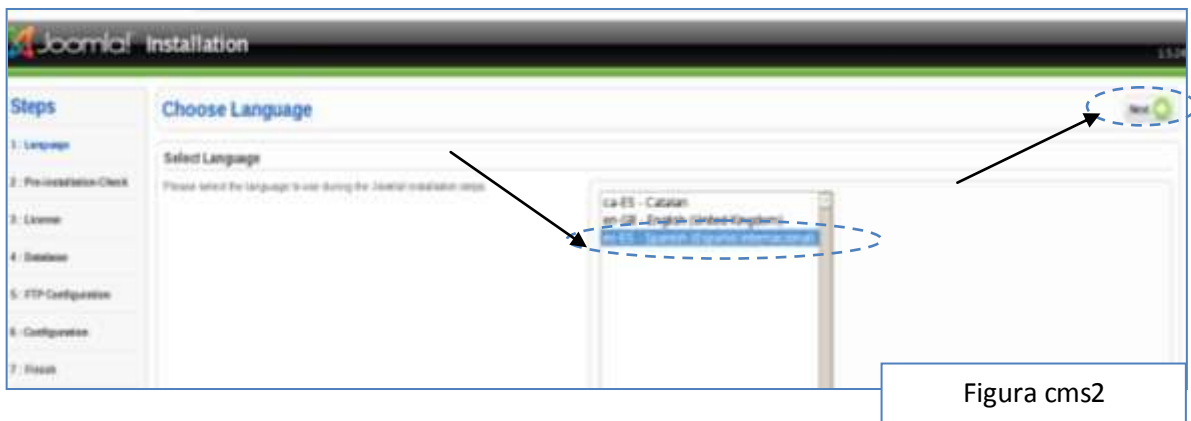
La configuración del **CMS** vía web, se realiza a través del navegador web de nuestro sistema server, el cual nos permitirá la configuración de joomla y al mismo tiempo para comprobar que los servicios apache y Mysql estén correctamente configurados para realizar la configuración realizamos los siguientes pasos ah describirse.

1. Iniciamos nuestro navegador web, una vez iniciado procedemos ah escribir en la barra de navegación la frase **localhost** que no es mas que hacer referencia al servicio montado propio del sistema de muestra en la figura cms1.



Figura cms1

- Una vez puesta la frase esta se auto completa y nos refleja el icono Joomla!, así como la auto completación de la frase quedándonos la línea mostrada en la figura cms1, ya cargada la pagina web continuaremos con la configuración y la primera opción es escoger el lenguaje en que deseamos realizar la configuración luego clic en el botón **next** situado en la parte derecha superior como se nos muestra en la figura cms2.

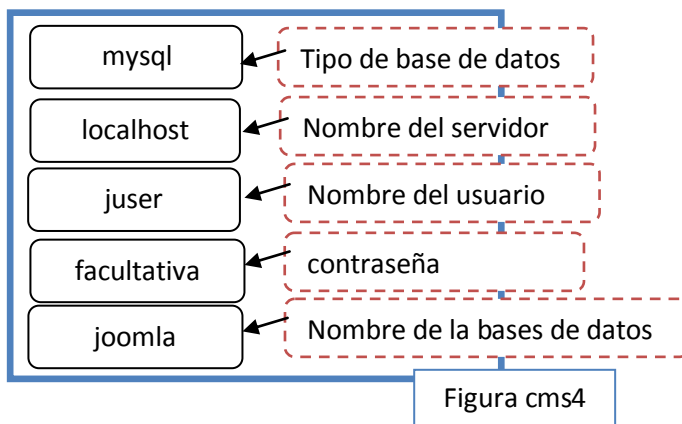


- El siguiente paso es donde nos damos cuenta si uno de nuestros servicios esta mal configurado dado que lo enmarca en rojo, sin embargo ah uno que por lo general se marca de ese color dado que al final de la configuración nos darán el código para su corrección lo que debemos fijarnos es que los servicios antecesores de php estén marcados de forma correcta entre ellos se encuentra en Mysql que es el que realmente nos importa si todo esta bien clic en botón siguiente como se nos muestra en la figura cms3.

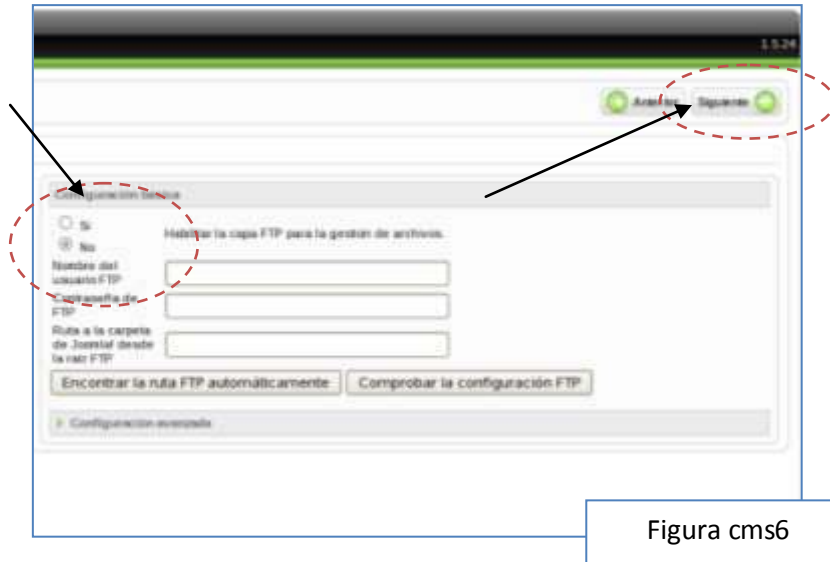


Ileana Patricia Loaisiga Hernandez.
Carlos Eduardo Barahona.

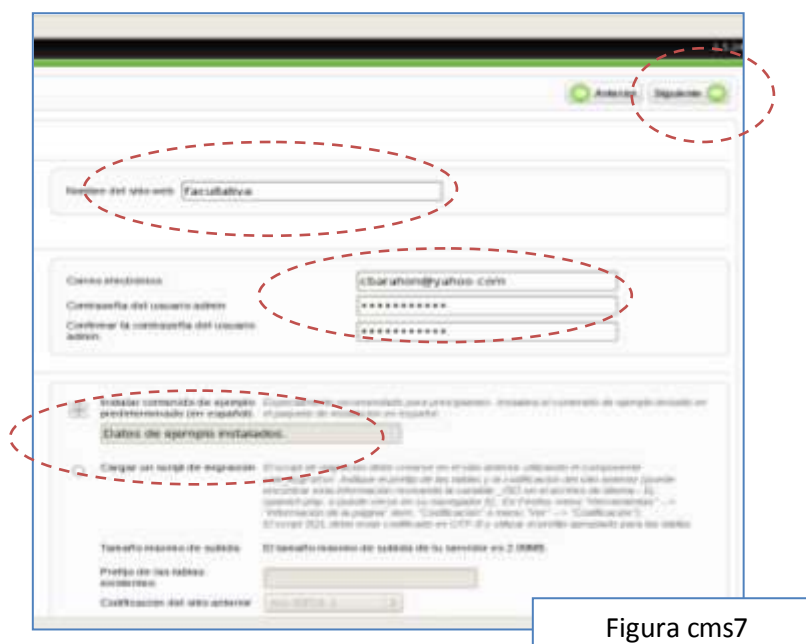
4. Una vez que la comprobación de los servicios nos da el acceso a la siguiente ventana en donde digitalizaremos algunos parámetros de configuración de acceso, que son los parámetros que configuramos en el servicio Mysql con los permisos de usuario, bases de datos y contraseña además que añadiremos el nombre del servidor en donde se encuentra alojada en este caso se pone el nombre de **localhost** haciendo referencia a nuestro sistema server como proveedor del servicio a continuación se muestra la figura cms4 y 5 respectivamente con los parámetros de la configuración.



5. Se nos muestra la configuración FTP aquí lo único es que nos cerciorarse que el radio button con la frase **no** este seleccionada como se muestra en la figura cms6.



6. A continuación se nos pedirán unos datos como nombre del sitio web que lo pondremos **facultativa** un correo electrónico valido que pondremos **cbarahon@yahoo.com** y una contraseña administrador que será **facultativa** y por ultimo clic en el botón **datos de ejemplos** y clic en siguiente para continuar como se nos muestra en la figura cms7.



- Para finalizar nos iremos a la terminal del sistema operativo y crearemos un archivo vacio de nombre **configuracion.php** en el directorio **/var/www/html** y luego abrirlo o visualizarlo.

```
touch /var/www/html/configuracion.php
vim /var/www/html/configuracion.php
```

- Regresamos al navegador donde estamos configurando el cms joolam y copiamos el código que se nos brinda luego lo pegamos en el archivo que tenemos abierto en la terminal como se nos muestra en las figuras cms8 y 9 respectivamente.

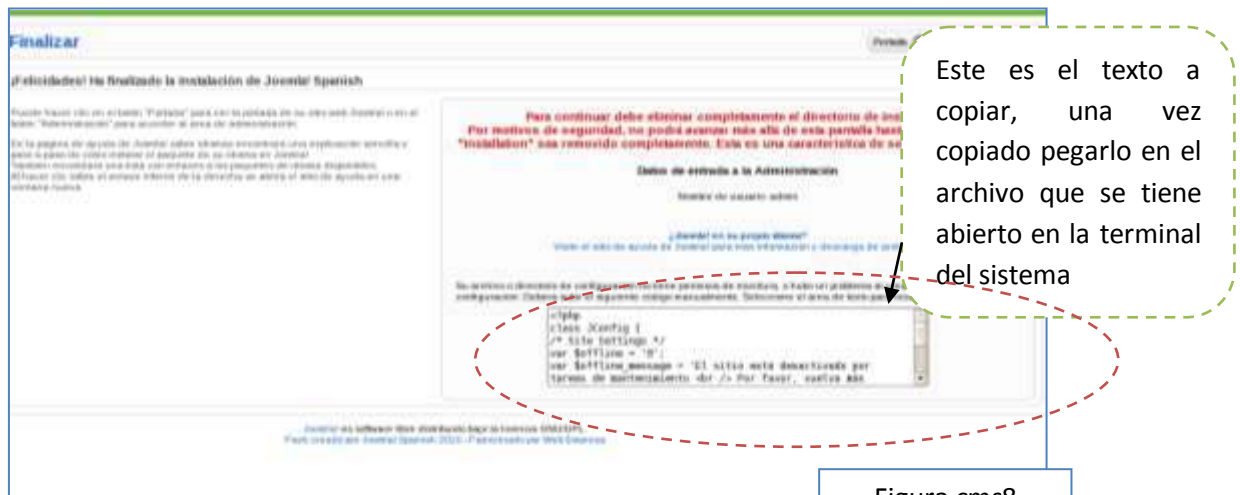


Figura cms8

```
var $ftp_user = '';
var $ftp_pass = '';
var $ftp_root = '';
var $ftp_path = '';
var $ftp_dir = '';
/* Local Settings */
var $offset = '0';
var $offset_user = '0';
/* Mail Settings */
var $mailer = 'mail';
var $mailfrom = 'www@joomla.com';
var $fromname = 'Joomla!';
var $sendmail = '/usr/sbin/sendmail';
var $smtpauth = '0';
var $smtpsecure = 'none';
var $smtpport = '25';
var $smtpuser = '';
var $smtppass = '';
var $smtphost = 'localhost';
/* Cache Settings */
var $caching = '0';
var $cache_ttl = '15';
var $cache_handler = 'file';
/* Meta Settings */
var $metaDesc = 'Joomla! - el motor de portales dinámicos y sistema de administración de contenidos';
var $metaKeywords = 'Joomla, Joomla!';
var $metaTitle = '';
var $metaAuthor = '';
/* SEO Settings */
var $sef = '0';
var $sef_rewrite = '0';
var $sef_suffix = '0';
/* Feed Settings */
var $feed_limit = 10;
var $feed_email = 'author';
var $img_path = '/var/www/html/images';
var $tmp_path = '/var/www/html/tmp';
/* Session Setting */
var $session = '15';
var $session_handler = 'database';
};
```

Figura cms9

10. Una vez pegado el ejemplo guardamos los cambios en el archivo.
11. Como ultimo punto necesitamos eliminar el archivo de **installation** ubicado en el directorio **/var/www/html/** como se nos muestra en la figura cms10.

```
cd /var/www/html/  
rm -Rf installation
```

```
[root@localhost html]# rm -Rf installation
```

Figura cms10

12. Con esto se finaliza la configuración e instalación del **CMS Joomla** como se muestra en las figura cms11.



Servidor DHCP

DHCP, es un protocolo (Dynamic Host Configuration Protocol) tiene como función proporcionar configuraciones de forma centralizada desde un servidor de la red, evitando así el tener que hacerlo de forma descentralizada desde cada estación de trabajo.

Un cliente que haya sido configurado con DHCP no posee direcciones estáticas sino que se configura totalmente de manera automática según las especificaciones del servidor DHCP.

En este último caso, el servidor DHCP procurará asignar a un cliente siempre la misma dirección para cada consulta (aunque estén espaciadas en el tiempo) – claro que esto no funcionará si en la red hay más ordenadores que direcciones.

Por lo tanto, el administrador del sistema puede beneficiarse de DHCP de dos formas.

Por una parte es posible realizar de forma centralizada, cómoda y automática grandes Modificaciones (de configuración y/o de direcciones de red) en el archivo de configuración del servidor DHCP y todo ello sin tener que configurar los clientes uno a uno.

Por otra parte y sobre todo, es posible integrar fácilmente nuevos ordenadores a la red Asignándoles un número IP del conjunto de direcciones.

Configuración del servidor DHCP

1. Para configurar el servicio dhcp es necesario obtener direcciones IP con su mascara de red correspondiente para ello nos otorgaron la siguiente dirección de red **193.168.0.0/24**, de la cual se nos pide un subneting para 12 sub redes y su nueva mascara de sub red.

cantidad	Sub red	Inicio red	Fin red	broadcast
1	192.168.0.0	192.168.0.1	192.168.0.14	192.168.0.15
2	192.168.0.16	192.168.0.17	192.168.0.30	192.168.0.31
3	192.168.0.32	192.168.0.33	192.168.0.46	192.168.0.47
4	192.168.0.48	192.168.0.49	192.168.0.62	192.168.0.63
5	192.168.0.64	192.168.0.65	192.168.0.78	192.168.0.79
6	192.168.0.80	192.168.0.81	192.168.0.94	192.168.0.95
7	192.168.0.96	192.168.0.97	192.168.0.110	192.168.0.111
8	192.168.0.112	192.168.0.113	192.168.0.126	192.168.0.127
9	192.168.0.128	192.168.0.129	192.168.0.142	192.168.0.143

- 1.1 Una vez obtenida la sub red que en nuestro caso nos toco la subred número **8** con dirección sub red **192.168.0.112** y una nueva **sub Mask 255.255.255.240**, se nos indica lo siguiente.
- 1.2 De las 14 direcciones IP utilizables las primeras 5 se apartan para servidores de la cual la primera se la asignaremos a nuestro servidor vía tarjeta de red en forma grafica a la cual se le asignara la IP **192.168.0.113** y su sub Mask **255.255.255.240** respectivamente.

1.3 De forma grafica nos ubicamos en el menú **systems >> administration >> Network** lo cual nos mostrara las tarjetas de red que tenemos disponibles para asignar dirección IP a nuestro servidor figura dhcp1.

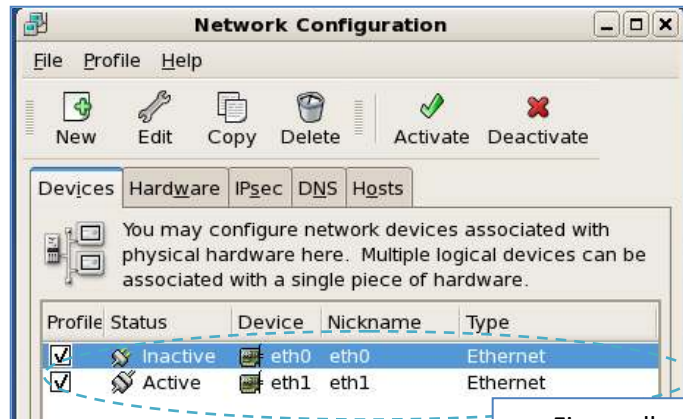


Figura dhcp1

1.4 Clic en la herramienta Edit esta nos permitirá editar los parámetros de nuestra tarjeta de red que en nuestro caso es la eth1 como se muestra en la figura dhcp2.

1.4.1 clic en la opción **Statically set IP addresses** colocamos respetivamente los valores a las líneas address y subnet mask como se lo mostramos en la figura dhcp2.

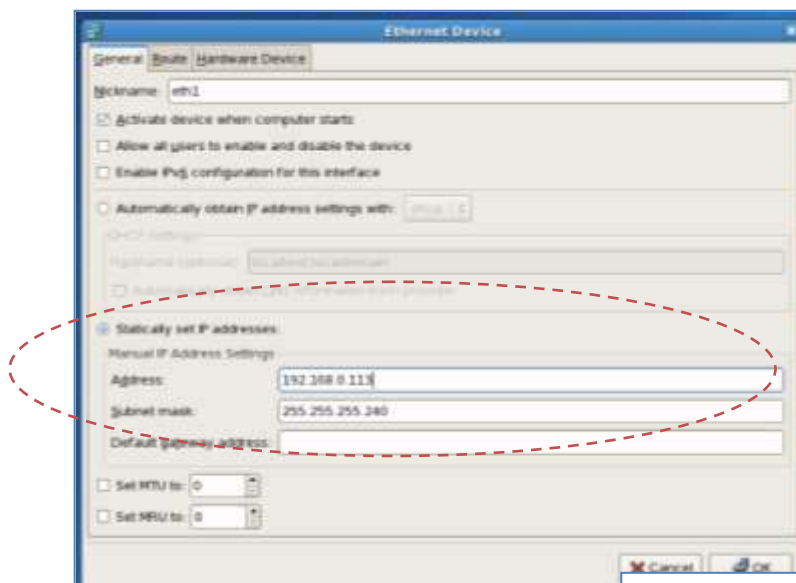


Figura dhcp2

1.5 Solo nos queda activar nuestra tarjeta como se muestra en la figura dhcp3 y listo podemos proseguir con la configuración del servidor dhcp.



Figura dhcp3

2. Para iniciar la instalación siempre se recuerda la inserción del DVD de nuestro sistema server en la unidad de lectura correspondiente, para instalar primero ejecutamos el nuestra terminal o Shell el siguiente comando como se nos muestra en la figura dhcp4.

```
yum install dhcp
```

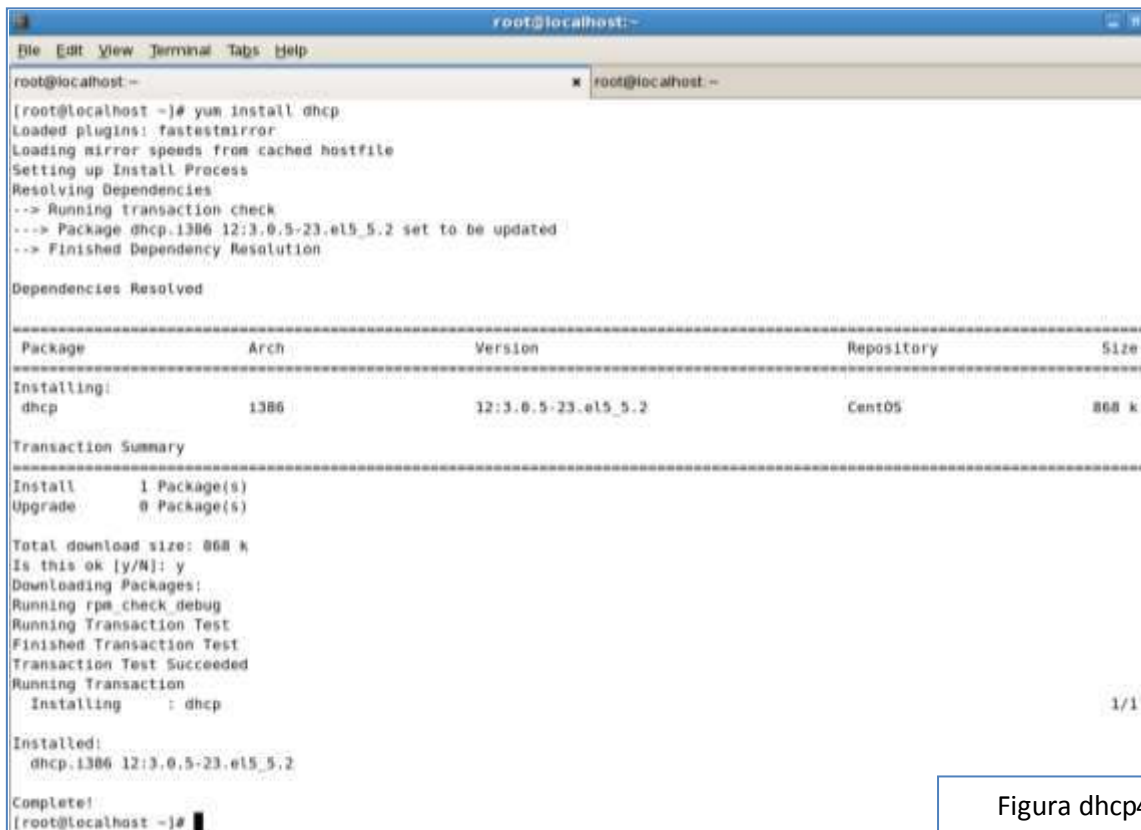


Figura dhcp4

3. El comando anterior solo nos configura por así decir o llamar la parte existencial en los directorios de configuración del servicio nota subrayar que este servicio no posee un archivo de configuración como los anteriores pero si posee un ejemplo de configuración el cual tomaremos para configurar nuestro servicio como se nos muestra en la figura dhcp5.

```
cp /usr/share/doc/dhcp-3.0.5/dhcpd.conf.sample /etc/dhcpd.conf
```

```
root@localhost:~  
File Edit View Terminal Tabs Help  
root@localhost:~ x root@localhost:~  
[root@localhost ~]# cp /usr/share/doc/dhcp-3.0.5/dhcpd.conf.sample /etc/dhcpd.conf
```

Figura dhcp5

4. Una vez copiado el archivo procedemos a borrar el archivo de ejemplo como se nos muestra en la figura dhcp6.

```
rm -R /usr/share/doc/dhcp-3.0.5/dhcpd.conf.sample
```

```
[root@localhost ~]# rm -R /usr/share/doc/dhcp-3.0.5/dhcpd.conf.sample  
rm: remove regular file `/usr/share/doc/dhcp-3.0.5/dhcpd.conf.sample'? y  
[root@localhost ~]#
```

Figura dhcp6

5. Como siguiente paso creamos una copia de respaldo del archivo **dhcpd.conf** con el nombre **dhcpd.conf.org**, luego de procedemos a editar el archivo de configuración de nuestro dhcp como se muestra en la figura dhcp7.

```
cp /etc/dhcpd.conf /dhcpd.conf.org  
vim /etc/dhcpd.conf
```

```
[root@localhost ~]# vim /etc/dhcpd.conf  
[root@localhost ~]#
```

Figura dhcp7

6. Una vez abierto nuestro archivo de configuración editaremos los parámetros como se muestran en las figuras dhcp8,9 y 10 respectivamente.

```
ddns-update-style interim;
ignore client-updates;
subnet 192.168.0.112 netmask 255.255.255.240 {
```

En esta línea ingresaremos nuestra sub red y nuestra mascara de sub red.

Figura dhcp8

```
# --- default gateway
option routers 192.168.0.113;
option subnet-mask 255.255.255.240;

option nis-domain "grupo8noct.edu";
option domain-name "grupo8noct.edu";
option domain-name-servers 192.168.0.113;

option time-offset -18000; # Eastern Standard Time
```

En esta parte ponemos nuestra direccion ip de servidor asi como la mascara de red que posee que hace referencia la proveedor del servicio

Figura dhcp9

```
# range dynamic-bootp 192.168.0.128 192.168.0.254;
default-lease-time 21600;
max-lease-time 43200;

# we want the nameserver to appear at a fixed address
host compl {
    hardware ethernet 00:08:74:F3:37:86;
    fixed-address 192.168.0.118;
}
```

La primera linea de range se comentara dado que es para dhcp dinamico, en las ultimas dos lines asignamos la direccion a nuestro computador cliente atravez de su mac.

Figura dhcp10

7. Una vez editado nuestro archivo de configuracion guardamos los cambios y procedemos a algo muy importante que es editar el archivo donde le indicaremos al servicio escuche las peticiones del cliente el cual lo encontramos **/etc/sysconfig/dhcpd** y procedemos ah editar el archivo como se nos muestran en las figuras dhcp11 y 12.

```
vim /etc/sysconfig/dhcpd
```

```
[root@localhost ~]# vim /etc/sysconfig/dhcpd
```

Figura dhcp11

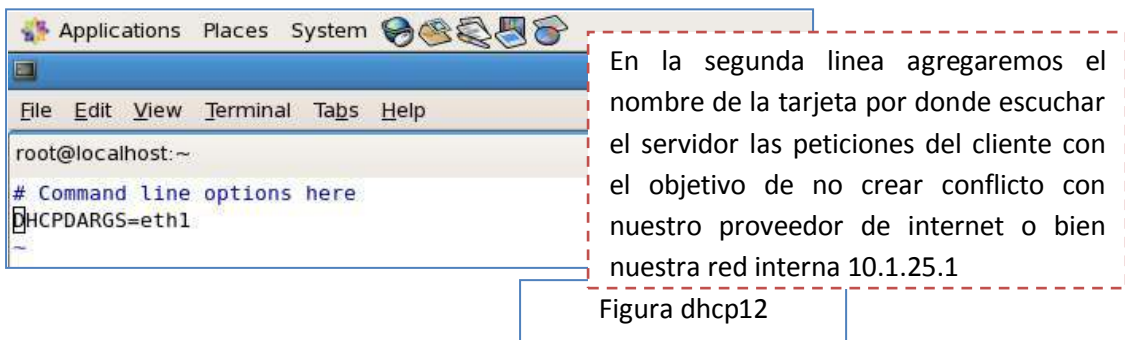


Figura dhcp12

8. Una vez configurado guardmos el archivo y procedemos a abrir los puestos 67 y 68 en udp desde **system>>administration>>securitylevelandfirewall** como se nos muestra en la figura dhcp13.

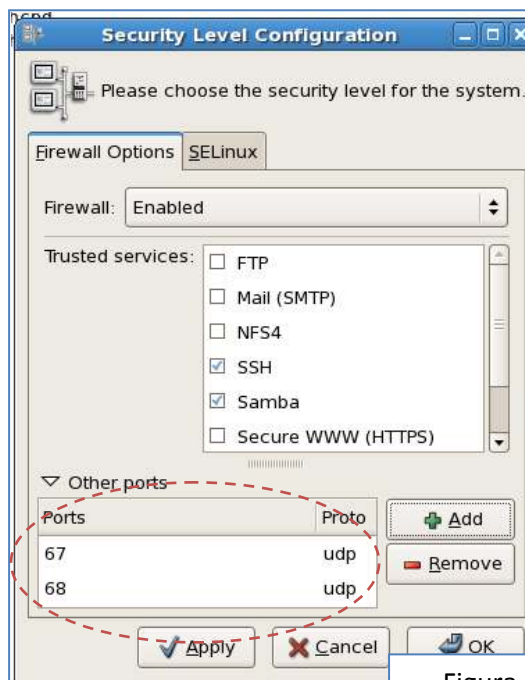


Figura dhcp13

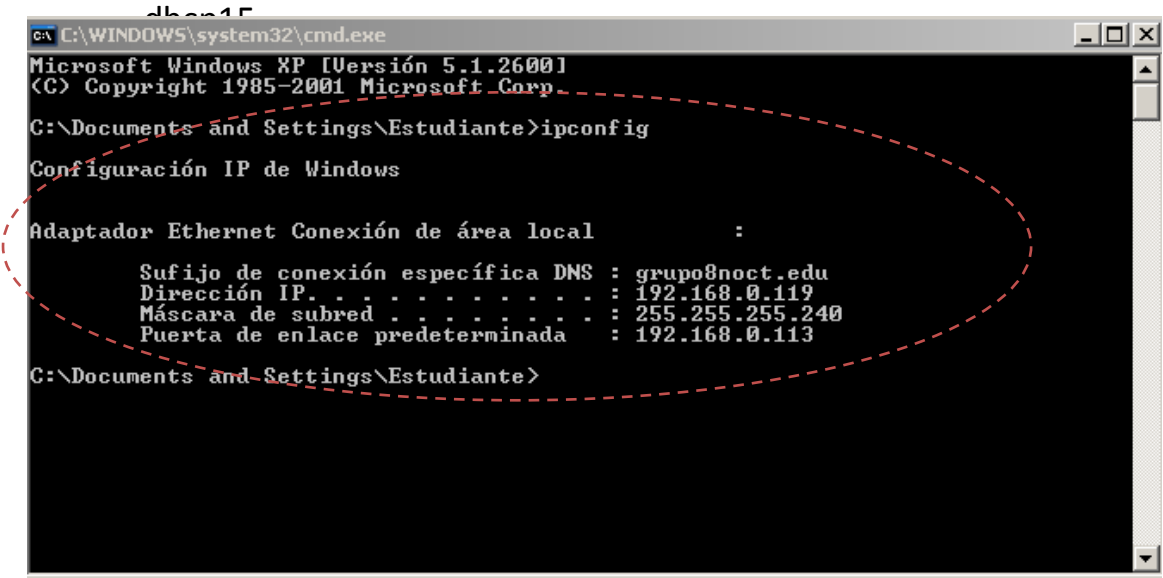
9. Guardar cambios y procederemos ah reiniciar el servicio de nuestras network, luego iniciamos los rervicios dhcpd y los encendemos como se nos muestra en la figura dhcp14.

```
service restart network  
chkconfig dhcpd on  
service dhcps start
```

```
[root@localhost ~]# chkconfig dhcpd on
[root@localhost ~]# service dhcpd start
Starting dhcpd: [ OK ]
```

Figura dhcp 14

10. Una vez iniciado nuestro servicio nos dirigimos ah nuestro computador cliente e iniciamos el **cmd** donde escribiremos el comando **ipconfig** para verificar que nuestro servicio dhcp esta funcionando de forma correcta como se nos muestra en la figura



11. Listo una vez confirmada que nuestro servidor esta funcionando de forma correcta se podria decir que hemos terminado la configuracion.

Servidor DNS

Un servidor DNS (Domain Name System) se utiliza para proveer a las computadoras de los usuarios (clientes) un nombre equivalente a las direcciones IP. Es un sistema de nomenclatura jerárquica para computadoras, servicios o cualquier recurso conectado al internet o a una red privada. Este sistema asocia información variada con nombres de dominios asignados a cada uno de los participantes. Su función más importante, es traducir (resolver) nombres inteligibles para los humanos en identificadores binarios asociados con los equipos conectados a la red, esto con el propósito de poder localizar y direccionar estos equipos mundialmente.

Un DNS se compone de tres componentes básicos, los cuales son:

- ✓ Cliente DNS: Es el host o usuario que hace la petición; el cual genera la petición al DNS preguntando por el nombre de algún dominio existente en internet.
- ✓ Servidor DNS: Existen 3 tipos de servidores básicos de un DNS los cuales son:
 - ✚ Servidor Maestro: maestro almacena los registros de las zonas originales y de autoridad, también es el encargado de responder a las peticiones hechas por otros servidores DNS.
 - ✚ Servidor Esclavo: tiene la capacidad de responder a las peticiones hechas por un Cliente DNS así como otro servidor de DNS, la diferencia radica en que los servidores esclavos obtienen la información acerca de los nombres de dominio desde los servidores maestros.
 - ✚ Servidor de Cache: Este ofrece servicios de resolución de nombres como respuesta a las peticiones hechas por los clientes de dns, Este tipo de servidores no tiene ninguna autoridad sobre las zonas de autoridad.

Los servidores DNS son los encargados de hacer las consultas producto de las peticiones solicitadas por los clientes DNS. Para ello el servidor DNS hace uso de 2 tipos de consultas

- ✓ Consultas Iterativas: toda la carga es nuestro cliente DNS (nuestra maquina).
- ✓ Consultas Recursivas: asume toda la carga es el servidor DNS pues es el encargado de proporcionar una respuesta completa a la petición hecha por el Cliente dns se puede concluir que las consultas recursivas son mejores que las consultas iterativas, debido a que las consultas recursivas liberan a nuestro cliente DNS (nuestra maquina) de la tarea de responder las peticiones solicitadas por el mismo, haciendo que toda la carga la asuma el servidor DNS.

Configuración del servicio DNS

1. Para empezar con la instalación del servidor **DNS** es necesario la edición de tres archivos distintos que se encuentran en directorios totalmente distintos, una vez aclarado este punto procederemos a la instalación de 4 paquetes necesarios para poder dar inicio como siempre se recuerda la inserción previa del DVD de nuestro sistema operativo server.

```
yum install bind-utils  
yum install bind-chroot  
yum install bind-libs  
yum install caching-nameserver
```

2. Una vez instalados hacemos una copia de nuestro primer archivo que utilizaremos para configurar el **DNS**, este archivo así como los otros dos que utilizaremos más adelante fueron facilitados por el docente por lo cual se encuentran en la ruta **/root/Desktop/FACII/DNS/** desde ese punto copiaremos nuestros tres archivos de configuración como se nos muestra en la figura dns1.

```
cp /root/Desktop/FACII/DNS/named.conf /var/named/chroot/etc/
```

```
[root@localhost ~]# cp /root/Desktop/FACII/DNS/named.conf /var/named/chroot/etc/
```

Figura dns1

- Una vez copiado nuestro primer archivo procederemos a cambiar los permisos así como el dueño del mismo ah named como se nos muestra en la figura dns2.

```
cd /var/named/chroot/etc/  
chown named:named named.conf  
chmod 644 named.conf
```

```
[root@localhost ~]# cd /var/named/chroot/etc/  
[root@localhost etc]# chown named:named named.conf  
[root@localhost etc]# chmod 644 named.conf
```

Figura dns2

- Una vez realizada la acción anterior procedemos a la copia de los dos archivos restantes que se ubicaran en el directorio **/var/named/chroot/var/named/** de igual forma otorgar permisos y cambiar dueño como se nos muestra en la figura dns3 y 4 respectivamente.

```
cp /root/Desktop/FACII/DNS/grupo8.com.zone /var/named/chroot/var/named/  
cp /root/Desktop/FACII/DNS/0.168.192.in-addr.arpa.zone  
/var/named/chroot/var/named/  
cd /var/named/chroot/var/named/  
chown named:named grupo8.com.zone  
chown named:named 0.168.192.in-addr.arpa.zone  
chmod 644 grupo8.com.zone  
chmod 644 0.168.192.in-addr.arpa.zone
```

```
[root@localhost etc]# cp /root/Desktop/FACII/DNS/grupo8.com.zone /var/named/chroot/var/named/  
[root@localhost etc]# cp /root/Desktop/FACII/DNS/0.168.192.in-addr.arpa.zone /var/named/chroot/var/named/
```

Figura dns3

```
[root@localhost named]# chown named:named grupo8.com.zone  
[root@localhost named]# chown named:named 0.168.192.in-addr.arpa.zone  
[root@localhost named]# chmod 644 grupo8.com.zone  
[root@localhost named]# chmod 644 0.168.192.in-addr.arpa.zone
```

Figura dns4

6. Una vez realizado el paso anterior procedemos a la edición del archivo `named.conf` como se nos muestra en la figura dns5.

```
cd /var/named/chroot/etc/  
vim named.conf
```

```
[root@localhost named]# cd /var/named/chroot/etc  
[root@localhost etc]# vim named.conf
```

Figura dns5

- 6.1 abierto el archivo procedemos a editar las líneas 6 y 7 donde especificaremos nuestra sub red padre y la sub red que nuestro servidor está utilizando para brindar el servicio como se nos muestra en la figura dns6.

```
Linea6 10.1.25.0/24;  
Linea7 192.168.0.112;
```

```
root@localhost:/var/named/chroot/etc x  
// Red Hat BIND Configuration Tool  
//  
// Default initial "Caching Only" name server configuration  
//  
acl "confiable" {  
10.1.25.0/24;  
192.168.0.112;  
localhost;  
};
```

Figura dns6

- 6.2 Luego procedemos a editar a partir de la línea 60 donde aremos referencia a las zonas creadas como se muestra en la figura dns7.

```
zone "grupo8noct.edu" {  
    type master;  
    file "grupo8noct.edu.zone";  
    allow-update { none; };  
};  
  
zone "0.168.192.in-addr.arpa" {  
    type master;  
    file "0.168.192.in-addr.arpa.zone";  
};  
#include "/etc/rndc.key";  
~
```

60.1

Figura dns7

7. Una vez guardado los cambios procederemos ah editar los archivos de zona pero al archivo de **grupo8.com.zone** le cambiaremos el nombre ah **grupo8noct.edu.zone** para que no cree conflicto con la configuración del **DNS** como se nos muestra en la figura dns8.

```
cd /var/named/chroot/var/named/  
mv grupo8.com.zone grupo8noct.edu.zone
```

```
[root@localhost etc]# cd /var/named/chroot/var/named/  
[root@localhost named]# mv grupo8.com.zone grupo8noct.edu.zone
```

Figura dns8

8. Una vez realizados los cambios procederemos ah editar los archivos de configuración de las zonas.

- 8.1 abrimos el archivo de configuración **grupo8noct.edu.zone** como se nos muestra en la figura dns9.

```
root@dns1:~  
File Edit View Terminal Tabs Help  
$TTL 86400  
@ IN SOA dns1.grupo8noct.edu. ileanaloesiga@hotmail.com. (  
    2011111800; ; serial  
    3H ; tiempo de refresco  
    15M ; tiempos entre reintentos de consulta  
    1W ; tiempo tras el cual expira la zona  
    1D ) ; tiempo total de vida  
@ IN NS grupo8noct.edu.  
www IN CNAME 192.168.0.113  
dns1.grupo8noct.edu. IN A 192.168.0.113  
grupo8noct.edu. IN A 192.168.0.113  
comp1 IN A 192.168.0.119
```

Figura dns9

- 8.1.1 en este archivo editaremos inicialmente la segunda y tercer línea respectivamente, la primer línea hace referencia al archivo zone junto con un correo valido del administrador la segunda línea hace referencia a la fecha que el servicio será iniciado o fecha actual en formato AA/MM/DD+00 que se refiere que es su primera edición a realizarse como se muestra en la figura dns10.

```
$TTL 86400  
@ IN SOA dns1.grupo8noct.edu. ileanaloesiga@hotmail.com. (  
    2011111800; ; serial
```

Figura dns10

8.1.2 Las siguientes líneas a editar empiezan desde la línea 8 en donde mostraremos al cliente quien es su proveedor de servicio a través de las direcciones de nuestro servidor así como la agregación de los clientes como se muestra en la figura dns11.

```
@           IN      NS      grupo8noct.edu.
www        IN      CNAME   192.168.0.113
dns1.grupo8noct.edu. IN      A       192.168.0.113
grupo8noct.edu.   IN      A       192.168.0.113
comp1      IN      A       192.168.0.119
~
```

Figura dns11

8.1.3 Una vez editado y guardado nuestro archivo procedemos ah editar nuestro siguiente archivo de zona **0.168.192.in-addr.arpa.zone** como se nos muestra en la figura dns12.

```
Vim 0.168.192.in-addr.arpa.zone
```

```
[root@localhost named]# vim 0.168.192.in-addr.arpa.zone
```

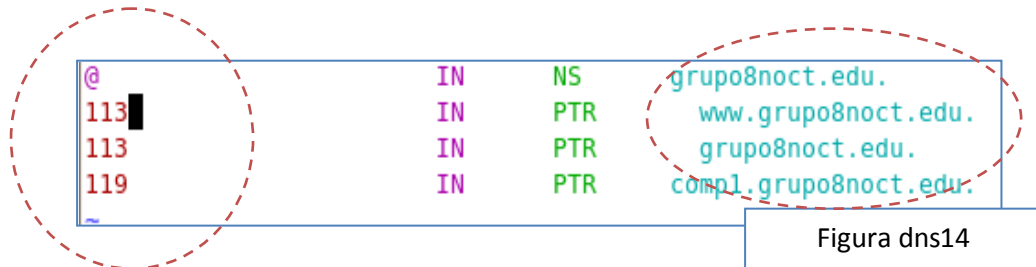
Figura dns12

8.1.4 Una vez abierto nuestro archivo procedemos ah editar la segunda línea y cuarta respectivamente será igual a nuestro archivo de zona editado anteriormente como se muestra en la figura dns13.

```
root@localhost:/var/named/chroot/var/named (on l...
File Edit View Terminal Tabs Help
root@localhost:/var/named/chroot/var/named x root@localho
$TTL 86400
@ IN SOA dns1.grupo8noct.edu. ileanaloesiga@hotmail.com. (
201111800; ; serial
```

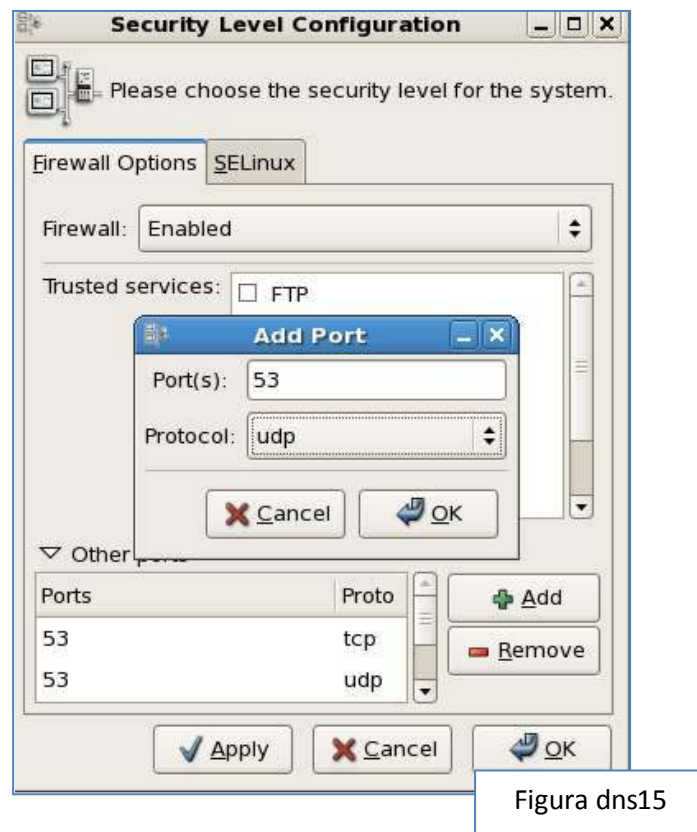
Figura dns13

8.1.5 Como siguiente editaremos a partir de la novena línea donde en la parte izquierda del texto se le hace referencia la ultimo octeto de nuestras direcciones IP tanto de servidor como de los ordenadores clientes, en la parte derecha los dominios de referencia asignados como se muestran en la figura dns14.



8.1.6 Una vez editado los archivos y guardados estaremos casi listos para iniciar nuestro servicio de dominio.

9. Antes de iniciar el servicio debemos de abrir los puertos 53 en tcp y udp de forma grafica **System>> administration>> securitylevelandfirewall** como se nos muestra en la figura dns15.



10. Luego de abrir los puertos procedemos a asignarle de forma gráfica a darle la verificación a nuestra tarjeta de red desde **System>>administration>>network** luego clic en la pestaña DNS y editamos los parámetros tal como se nos muestra en la figura dns16.



Figura dns16

11. Guardamos los cambios y reiniciamos nuestra red como se nos muestra en la figura dns17.

```
[root@localhost ~]# service network restart
Shutting down interface eth0:           [ OK ]
Shutting down interface eth1:         [ OK ]
Shutting down loopback interface:     [ OK ]
Bringing up loopback interface:       [ OK ]
Bringing up interface eth0:
Determining IP information for eth0... done.
Bringing up interface eth1:           [ OK ]
                                         [ OK ]
```

Figura dns17

12. Una vez reiniciada nuestra red procedemos a iniciar nuestro servicio **DNS** como ya es acostumbrado como se muestra continuación.

```
service named start
```

13. Luego de iniciar nuestro servicio, entramos como usuario a la consola de Windows y ejecute el comando **nslookup** para verificar que nuestro servidor este funcionando correctamente.

Servicio Proxy o Squid

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

- ✓ Cliente se conecta hacia un Servidor Intermediario (Proxy).
- ✓ Cliente solicita una conexión, fichero u otro recurso disponible en un servidor distinto.
- ✓ Servidor Intermediario (Proxy) proporciona el recurso ya sea conectándose hacia el servidor especificado o sirviendo éste desde un caché.
- ✓ En algunos casos el Servidor Intermediario (Proxy) puede alterar la solicitud del cliente o bien la respuesta del servidor para diversos propósitos.

Los Servidores Intermediarios generalmente trabajan simultáneamente como muro cortafuegos operando en el Nivel de Red, actúan como filtro de paquetes, como en el caso de iptables, o bien operando en el Nivel de Aplicación, controlando diversos servicios.

Cuando se recibe una petición para un recurso de Red especificado en un URL el Servidor Intermediario busca el resultado del URL dentro del caché. Si éste es encontrado, el Servidor Intermediario responde al cliente proporcionado inmediatamente el contenido solicitado. Si el contenido solicitado no estuviera disponible en el caché, el Servidor Intermediario lo traerá desde servidor remoto, entregándolo al cliente que lo solicitó y guardando una copia en el caché. El contenido en el caché es eliminado luego a través de un algoritmo de expiración de acuerdo a la antigüedad, tamaño e historial de respuestas a solicitudes.

Configuración servicio Squid

1. Para empezar la instalación del servicio squid cabe recordar tener insertado el DVD del sistema operativo dentro de la unidad, procedemos instalar el servicio como se nos muestra a continuación en la figura squid1.

```
yum install squid
```

```
root@dns1:~# yum install squid
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
CentOS                    | 1.1 kB    00:00
CentOS/primary            | 954 kB    00:00
CentOS                    |          2683/2683
Setting up Install Process
Package 7:squid-2.6.STABLE21-6.el5.i386 already installed and latest version
Nothing to do
[root@dns1 ~]#
```

Figura squid1

2. Luego ingresamos en el directorio donde se ubica el archivo de configuración squid como se muestra en la figura squid2.

```
[root@dns1 ~]# cd /etc/squid/
```

Figura squid2

3. Se crea una copia de respaldo del archivo de configuración **squid.conf** con el nombre **squid.conf.org**

```
cp squid.conf squid.conf.org
```

4. Realizada la copia procedemos a abrir el archivo de configuración **squid.conf** como se muestra en la figura squid3.

```
[root@dns1 squid]# vim squid.conf
```

Figura squid3

5. Se inicia la configuración de los parámetros como se nos muestra en las figuras squid4, 5, 6 y 7 respectivamente a continuación.

```
root@dns1:/etc/squid
#
# WELCOME TO SQUID 2.6.STABLE6
#
http_port 192.168.0.113:8080
hierarchy_stoplist cgi-bin ?
acl QUERY urlpath_regex cgi-bin \?
cache deny QUERY
acl apache rep_header Server ^Apache
broken_vary_encoding allow apache
#Puerto del ICP - Activar en FAC II
icp_port 3130
cache_mem 256 MB
visible_hostname grupo8noct
cache_swap_low 90
cache_swap_high 95
#Algoritmo de implementacion
cache_replacement_policy heap LRU
memory_replacement_policy heap LRU

# memory_replacement_policy lru
cache_dir ufs /var/spool/squid 33000 16 256
access_log /var/log/squid/access.log squid
cache_log /var/log/squid/cache.log
# tiempo maximo para cacheados
minimum_object_size 0 KB
maximum_object_size 4096 KB
```

Figura squid4

```
#Sugerencias predeterminadas:
refresh_pattern ^ftp:          1440    20%    10080
refresh_pattern ^gopher:      1440    0%     1440
refresh_pattern .              0       20%    4320
```

Figura squid5


```
#Squid Padre
cache_peer 10.1.25.1 parent 3128 3130 proxy-only

acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl to_localhost dst 127.0.0.0/8
acl labs src 192.168.0.112/255.255.255.240
acl password proxy_auth REQUIRED
acl spermitidos url_regex "/etc/squid/sitios-inocentes"
acl sitios url_regex "/etc/squid/sitios"
acl listaextensiones urlpath_regex "/etc/squid/listaextensiones"
acl rangos src "/etc/squid/informatica"

acl SSL_ports port 443
acl Safe_ports port 80          # http
acl Safe_ports port 21         # ftp
acl Safe_ports port 443       # https
acl Safe_ports port 70        # gopher
acl Safe_ports port 210       # wais
acl Safe_ports port 1025-65535 # unregistered ports
acl Safe_ports port 280       # http-mgmt
acl Safe_ports port 488       # gss-http
acl Safe_ports port 591       # filemaker
acl Safe_ports port 777       # multiling http
acl Safe_ports port 5050      # Gaim-yahoo
acl Safe_ports port 1863      # Gaim-msn
acl CONNECT method CONNECT

#Recommended minimum configuration:
# Only allow cachemgr access from localhost
http_access allow manager localhost
http_access allow labs password !sitios !rangos !listaextensiones
http_access deny manager

# Deny requests to unknown ports
-- INSERT --
```

Figura squid6

```
# Deny requests to unknown ports
http_access deny !Safe_ports
# Deny CONNECT to other than SSL ports
http_access deny CONNECT !SSL_ports
# And finally deny all other access to this proxy
http_access allow localhost
http_access allow all spermitidos
http_access deny all
# and finally allow by default
http_reply_access allow all
#Allow ICP queries from everyone
icp_access allow all
cache_mgr cbarahon@yahoo.com
# Leave coredumps in the first cache dir
coredump_dir /var/spool/squid
log_icp_queries off
buffered_logs on
-- INSERT --
```

Figura squid7

- Realizada la configuración del archivo **squid.conf** procedemos a crear los archivos sitios, sitios-inocentes, listaextensiones e informática como se nos muestran en las figuras squid8.

```
root@dns1:/etc/squid x root@d
[root@dns1 squid]# touch sitios
[root@dns1 squid]# touch sitios-inocentes
[root@dns1 squid]# touch listaextensiones
[root@dns1 squid]# touch listaextensiones
```

Figura squid8

- Una vez creado los archivos procedemos a cambiarles de dueño a cada uno de ellos como se muestra en la figura squid9.

```
File Edit View Terminal Tabs Help
root@dns1:/etc/squid x root@dns1:~
[root@dns1 squid]# chown squid:squid sitios
[root@dns1 squid]# chown squid:squid sitios-inocentes
[root@dns1 squid]# chown squid:squid listaextensiones
[root@dns1 squid]# chown squid:squid informatica
```

Figura squid9

- Cambiado el dueño de los archivos procederemos a la edición de cada uno de ellos y le agregamos las palabras, frases, direcciones, extensiones y direcciones IP como se nos muestran en las figuras ejemplos squid10, 11, 12 y 13 respectivamente.

8.1 sitios.

```
Applications Places System
root@dns1:/etc/squid
File Edit View Terminal Tabs Help
root@dns1:/etc/squid x root@dns1:~
#Expresiones
pornografia
#Paginas
astalavista.com
# Virus
143.106.5.75/cartao.scr
150.187.4.94/cb/cartao2006.exe
200.142.99.6/.tim2/tim/spc/cpf_debito.exe
200.146.119.203/charges/charge37394.scr
200.148.212.234/.ebt/consulta/consulta.exe
200.157.137.82/usage/msnmessegner8.scr
200.207.9.220/.../orkut.scr
200.222.86.61/cartao.scr
200.241.212.23/adm/floresonline/surpresa.scr
201.2.129.186/.tim/d_aberto.exe
201.45.142.130/consultas/embratel/spc/cpf.exe
211.32.65.138/bbs/data/netemp.exe
213.16.42.226/en/mambots/cartao.scr
213.186.33.16/~cvr/ads/kasino.scr
54699.com/wm/ma/2.txt
61.100.9.220/jabong/board/.vervivofototorpedo/vivofototorpedo.scr
61.96.198.20/naboard318/sapo/vercartao.php=84309
66.226.22.41/a/uolcartao.scr
```

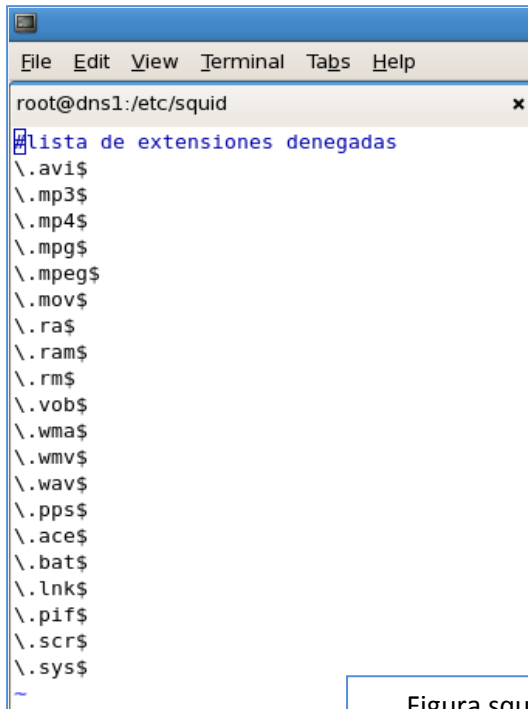
Figura squid10

8.2 sitio-inocentes.

```
Applications Places System
root@dns1:/etc/squid
File Edit View Terminal Tabs Help
root@dns1:/etc/squid x root@dns1:~
sexualidad
computacion
aspx
analyser
analizador
analizadora
analisis
articulo
articulos
adulto
adultos
encachimbados
www.sexualidadjoven.cl
support.dell.com/support/topics/global.aspx/support/downloads/en/download_flyout?
support.dell.com
show
choose
users
usuarios
www.claro.com.ni
www.ideasclaro.com.ni
www.telecom.com.sv
ns.enitel.net.ni
ns2.enitel.net.ni
en.wikipedia.org/wiki/production_code
users.pandora.be/worldstandards/electricity.htm
https://*.contacts.msn.com
http://www.travian.ws/analyser.pl?s=ar1&uid=81366
http://www.elnuevodiario.com.ni
```

Figura squid11

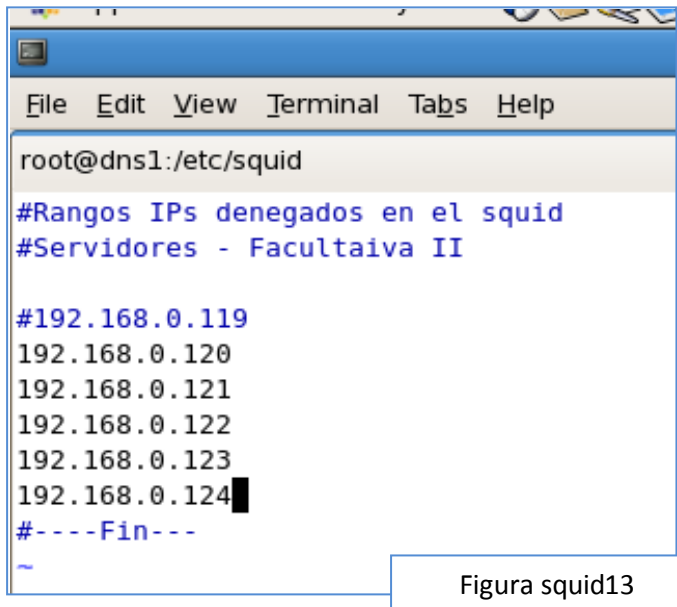
8.3 Lista de extensiones.



```
root@dns1:/etc/squid
#Lista de extensiones denegadas
\.avi$
\.mp3$
\.mp4$
\.mpg$
\.mpeg$
\.mov$
\.ra$
\.ram$
\.rm$
\.vob$
\.wma$
\.wmv$
\.wav$
\.pps$
\.ace$
\.bat$
\.lnk$
\.pif$
\.scr$
\.sys$
```

Figura squid12

8.4 Informática.



```
root@dns1:/etc/squid
#Rangos IPs denegados en el squid
#Servidores - Facultaiva II

#192.168.0.119
192.168.0.120
192.168.0.121
192.168.0.122
192.168.0.123
192.168.0.124█
#----Fin---
```

Figura squid13

9. Realizado los archivos ejecutamos la regla de setsebool para que nuestro sistema permita la ejecución correcta del servicio como se muestra en la figura squid14.

```
setsebool -P squid_connect_any = 1
```

```
root@dns1:~  
File Edit View Terminal Tabs Help  
root@dns1:/etc/squid x root@dns1:~  
[root@dns1 squid]# setsebool -P squid_connect_any = 1
```

Figura squid14

10. Como recordatorio para ir finalizando debemos abrir el puerto 8080 en tcp y udp respectivamente de forma grafica en **system >> administation >> securitylevelandfirewall**.

11. Procedemos a encender el servicio squid y dejarlo permanente como se muestra en la figura squid15.

```
chkconfig squid on  
service squid restart
```

```
[root@dns1 squid]# chkconfig squid on  
[root@dns1 squid]# service squid restart  
Stopping squid: ..... [ OK ]  
Starting squid: . [ OK ]  
[root@dns1 squid]#
```

Figura squid15

12. Procedemos a verificar las pruebas con el ordenador cliente como se muestra en las figuras squid16 y 17 respectivamente con la uno de nuestras listas de extensiones.



Figura squid16

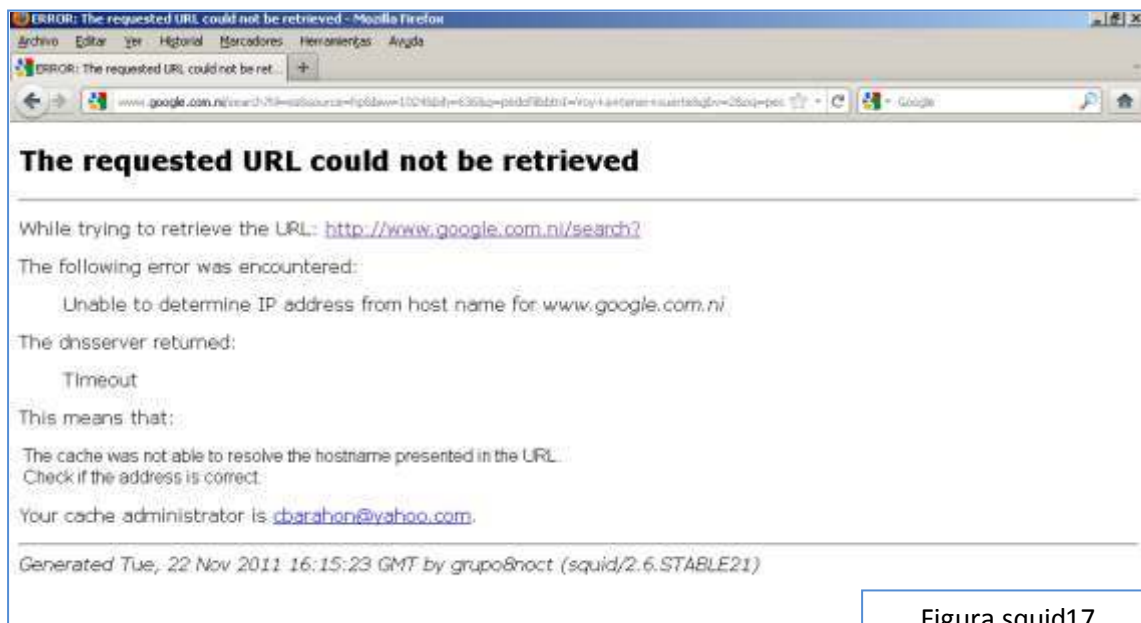


Figura squid17



Conclusiones

El mundo de configuración de servicios es muy extenso se espera que el manual de iniciación de administración de servidores sea de gran utilidad para aquellas personas que desean inclinarse a este tipo de tareas. El manual fue basado en software libre como bien lo dice la palabra es de distribución libre y gratuita. La distribución utilizada de software fue CentOS 5.6 final así como su versión superior CentOS 6 Final, que cuenta con las características principales del Sistema Operativo RedHat de Licencia no libre que pertenece a la misma familia Unix-Linux. Lo esperado es que sea de provecho para aquellos usuarios de este tipo de sistema.



Bibliografía

Ileana Patricia Loaisiga Hernández.... (Estudiante Informática Educativa UNAN-Managua).

Carlos Eduardo Barahona.... (Estudiante Informática Educativa UNAN-Managua).

Lenin Silva Colome.... (Ms. Software Libre UNAN-Managua, Administrador de Servidores Unix/Linux).

CentOS Nicaragua... (www.centosni.net).