

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA INFORMÁTICA

## **TERCER TALLER DE REDES DE COMPUTADORES**

**Sebastián Reyes Fuenzalida  
Gonzalo Zeballos Alvarez**

INFORME PARA REDES DE COMPUTADORES

DICIEMBRE, 2015

# Índice

<b>1. Introducción</b>	<b>1</b>
<b>2. Desarrollo</b>	<b>2</b>
2.1. Instalación de servicios necesarios. . . . .	2
2.1.1. Servidor dns (Bind). . . . .	2
2.1.2. Servidor web (Apache). . . . .	4
2.2. Configuración de servicios. . . . .	5
2.2.1. Configuración de servidor dns. . . . .	5
2.2.2. Configuración de apache. . . . .	9
2.3. Resultados parte 1. . . . .	11
2.4. Instalar y configurar un servidor proxy SQUID . . . . .	12
<b>3. Conclusión</b>	<b>17</b>

## Lista de Figuras

1.	Contenido de Index usando nano. . . . .	10
2.	Resultados obtenidos en casa.miquipo.cl. . . . .	11
3.	Resultados obtenidos en home.miquipo.cl. . . . .	11
4.	Instalando Servidor SQUID . . . . .	12
5.	Nombrar el PROXY . . . . .	13
6.	determinar que el puerto a escuchar es 1234 . . . . .	14
7.	Creando grupo llamado arealocal . . . . .	15
8.	Otorgando acceso a arealocal . . . . .	16

## Lista de Comandos

1.	Instalación Del Servicio Bind . . . . .	2
2.	resultado de Instalación Del Servicio Bind . . . . .	4
3.	Instalación Del Servicio HTTPD . . . . .	4
4.	resultado de Instalación Del Servicio HTTPD . . . . .	5
5.	Acceder a la configuración de named.conf . . . . .	5
6.	resultado de los cambios realizados en el archivo name.conf . . . . .	6
7.	Acceder a la configuración de named.rfc1912.zones . . . . .	6
8.	Codigo agregado en named.rfc1912.zones . . . . .	7
9.	Acceder a la configuración de named.db.miequipo.cl . . . . .	7
10.	Codigo para configuracion de named.db.miequipo.cl . . . . .	8
11.	Acceder a la configuración de resolv.conf . . . . .	8
12.	Resultado de la configuracion de resolv.conf . . . . .	8
13.	Reinicio del servicio named . . . . .	8
14.	Realizando las pruebas con el comando name-checkconf . . . . .	8
15.	Realizando pruebas con el comando name-checkzone . . . . .	8
16.	Ruta y acceso a la configuración de httpd.conf . . . . .	9
17.	Resultado de los cambios en httpd.conf . . . . .	9
18.	Acceso y creación de directorios . . . . .	10
19.	creación de los archivos Index . . . . .	10
20.	Reinicio del servicio HTTPD . . . . .	10

## 1. Introducción

EL servidor DNS nació de la necesidad de recordar fácilmente los nombres de todos los servidores conectados a Internet. En un inicio, SRI (ahora SRI International) alojaba un archivo llamado *HOSTS* que contenía todos los nombres de dominio conocidos. El crecimiento explosivo de la red causó que el sistema de nombres centralizado en el archivo *Hosts* no resultara práctico y en 1983, Paul V. Mockapetris publicó los RFC 882 y RFC 883 definiendo lo que hoy en día ha evolucionado hacia el DNS moderno.

En este taller el servidor DNS dará el soporte para el uso de sub-dominios para un dominio maestro el cual podrá ser accedido a través del servidor web que dará en soporte para la paginas a mostrar. Según el dominio ingresado se mostrara una u otra web pero siempre con la misma dirección IP. Dentro del desarrollo se encontraran la mayor parte de la configuración de los archivos necesarios.

La forma abordar esta practica sera necesario aplicar conocimientos de direccionamiento IP, la instalación de servicios, configuración de zonas DNS y configuración de APACHE. Los materiales a utiliza serán un computador como servidor y computadores como clientes o varias maquinas virtuales. Con las cuales se pretende emular lo que hace un *Hosting* real, donde una empresa con 1 servidor puede alojar una gran cantidad de sitios web distinto.

## 2. Desarrollo

Usar un Servidor Linux con el servicio de Servidor de Nombres BIND, el cual asigne diferentes nombres a una misma dirección IP correspondiente a otro servidor que corre APACHE 2 (Puede ser Windows o Linux, recomendado Linux). La idea es que el mismo servidor Apache muestre diferentes páginas (sitios) según sea llamado con un nombre u otro (ej: Si es invocado como `http://miequipo.cl` aparezca una página, y si se invoca como `http://miotroequipo.cl` muestre otra, aunque el DNS retorne la misma IP para ambas direcciones). Además, instale y configure un servidor proxy SQUID, con el fin de que usuario de una red internan naveguen a través de él (en primer lugar, configurando los browser para usar proxy, y luego configure para que sea en forma transparente).

### 2.1. Instalación de servicios necesarios.

Aquí se encontrara una definición y los comandos para realiza la instalación de los servicios que son los siguientes:

#### 2.1.1. Servidor dns (Bind).

El sistema de nombres de dominio (dns) es un sistema para asignar nombres a equipos y servicios de red que se organiza en una jerarquía de dominios. Las redes como Internet, usan dns para buscar equipos y servicios mediante nombres descriptivos.

Los sistemas de nombres como dns proporcionan un método para asignar el nombre descriptivo de un equipo o servicio a otros datos asociados a dicho nombre, como una dirección IP. Un nombre descriptivo es más fácil de aprender y recordar que las direcciones numéricas que los equipos usan para comunicarse a través de una red. La mayoría de la gente prefiere usar un nombre descriptivo (por ejemplo, `sales.fabrikam.com`) para buscar un servidor de correo electrónico o servidor web en una red en lugar de una dirección IP, como `157.60.0.1`. Cuando un usuario escribe un nombre dns descriptivo en una aplicación, los servicios dns convierten el nombre en su dirección numérica.

Para empezar la instalación del servicio utilizaremos el comando yum en la terminal de CentOS.

```
|| [root@laboratorio alumno]# yum install bind
```

Comando 1: Instalación Del Servicio Bind

Como se puede observar no fue necesario el instalarlo, pues ya lo estaba.

```
Complementos cargados:fastestmirror, refresh-packagekit,
security
Configurando el proceso de instalacion
Loading mirror speeds from cached hostfile
* base: centos.zeotec.cl
* epel: mirror.cedia.org.ec
* extras: centos.zeotec.cl
* updates: centos.zeotec.cl
El paquete 32:bind-9.8.2-0.30.rc1.el6_6.3.x86_64 ya se
encuentra instalado con su version mas reciente
Nada para hacer
```

Comando 2: resultado de Instalación Del Servicio Bind

### 2.1.2. Servidor web (Apache).

Un servidor web o servidor HTTP es un programa que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente y generando una respuesta en cualquier lenguaje o Aplicación del lado del cliente. El código recibido por el cliente suele ser compilado y ejecutado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se usa el protocolo HTTP para estas comunicaciones, perteneciente a la capa de aplicación del modelo OSI.

Instalación del servicio httpd utilizaremos el comando yum en la terminal de CentOS.

```
|| [root@laboratorio alumno]# yum install httpd
```

Comando 3: Instalación Del Servicio HTTPD

Como se puede observar no fue necesario el instalarlo.



```
Complementos cargados:fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: centos.zeotec.cl
* extras: centos.zeotec.cl
* updates: centos.zeotec.cl
El paquete httpd 2.4.6-31.e17.centos.x86_64 ya se encuentra
instalado con su version mas reciente
Nada para hacer
```

Comando 4: resultado de Instalación Del Servicio HTTPD

## 2.2. Configuración de servicios.

Aquí se encontrara la configuración y los comandos para realízalo según sea necesario y que son los siguientes:

### 2.2.1. Configuración de servidor dns.

la primera acción a realizar es la de configuración de la maquina y red en el archivo named.conf ubicado en la carpeta etc en el servidor de LINUX CentOS.

```
|| [root@laboratorio ~]# vi /etc/named.conf
```

Comando 5: Acceder a la configuración de named.conf

en el archivo named.conf se debe agregar la configuración de la IP que posee el servidor y la red en la que esta trabajado o usando, en nuestro caso son los siguientes.

- IP: 158.251.93.141
- RED: 158.251.0.0

la IP de la maquina debe ser agregada en la linea de código

*"listen-on port 53 { 127.0.0.1; 158.251.93.141; };"*

y la de RED en

*"allow-query { localhost; 158.251.0.0; };"*.

Ahora observaremos el resultado de aquello en el archivo.

```
//
// named.conf
//
// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS
// server as a caching only nameserver (as a localhost DNS resolver only).
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//

options {
    listen-on port 53 { 127.0.0.1; 158.251.93.141; };
    listen-on-v6 port 53 { ::1; };
    directory      "/var/named";
    dump-file       "/var/named/data/cache_dump.db";
    statistics-file "/var/named/data/named_stats.txt";
    memstatistics-file "/var/named/data/named_mem_stats.txt";
    allow-query     { localhost; 158.251.0.0; };
    recursion yes;

    dnssec-enable yes;
    dnssec-validation yes;
    dnssec-lookaside auto;

    /* Path to ISC DLV key */
    bindkeys-file   "/etc/named.iscdlv.key";

    managed-keys-directory "/var/named/dynamic";
};

logging {
    channel default_debug {
        file "data/named.run";
        severity dynamic;
    };
};

zone "." IN {
    type hint;
    file "named.ca";
};

include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
```

Comando 6: resultado de los cambios realizados en el archivo name.conf

Segundo paso sera configurar el archivo de zonas en nuestro caso ocupamos el “*named.rfc1912.zones*”, para acceder al archivo usaremos el comando

```
|| [root@laboratorio etc]# vi named.rfc1912.zones
```

Comando 7: Acceder a la configuración de named.rfc1912.zones

Ahora solo basta que se adicione el siguiente fragmento de código con la configuración del nombre dominio que utilizaremos y la ruta del archivo para el servidor dns donde se configuraran los alias.

- Dominio: miequipo.cl
- File: named.db.miequipo.cl

```
// named.rfc1912.zones:
//
// Provided by Red Hat caching-nameserver package
//
// ISC BIND named zone configuration for zones recommended by
// RFC 1912 section 4.1 : localhost TLDs and address zones
// and http://www.ietf.org/internet-drafts/draft-ietf-dnsop-default-local-zones-02.txt
// (c)2007 R W Franks
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//

zone "localhost.localdomain" IN {
    type master;
    file "named.localhost";
    allow-update { none; };
};

zone "localhost" IN {
    type master;
    file "named.localhost";
    allow-update { none; };
};

zone "1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.arpa" IN {
    type master;
    file "named.loopback";
    allow-update { none; };
};

zone "1.0.0.127.in-addr.arpa" IN {
    type master;
    file "named.loopback";
    allow-update { none; };
};

zone "0.in-addr.arpa" IN {
    type master;
    file "named.empty";
    allow-update { none; };
};

zone "miequipo.cl" IN {
    type master;
    file "/etc/named.db.miequipo.cl";
};
```

---

Comando 8: Codigo agregado en named.rfc1912.zones

El tercer paso es crear el archivo de configuración directa en donde ira la definición de dominio maestro, la maquina en que se encuentra y los alias, en nuestro caso es el archivo “named.db.miequipo.cl”

```
[root@laboratorio etc]# vi named.db.miequipo.cl
```

Comando 9: Acceder a la configuración de `named.db.miequipo.cl`

como sub-dominios hemos escogido “home” y “casa”.

```

$TTL 604800
@      IN      SOA      dns.miequipo.cl.  admin.miequipo.cl. (
                        2015100617
                        604800
                        86400
                        2419200
                        60800)

;
@      IN      NS       dns.miequipo.cl.
dns    IN      A        158.251.93.141
home   IN      CNAME    dns
casa   IN      CNAME    dns
www    IN      CNAME    dns

```

Comando 10: Código para configuración de named.db.miequipo.cl

El último paso es ver la configuración del archivo “resolv.conf” en el cual se pondrá la dirección IP del servidor.

```

|| [root@laboratorio etc]# vi resolv.conf

```

Comando 11: Acceder a la configuración de resolv.conf

```

# Generated by NetworkManager
search localdomain miequipo.cl
nameserver 158.251.93.141
nameserver 192.168.88.2

```

Comando 12: Resultado de la configuración de resolv.conf

Ahora solo queda reiniciar el servicio “named” y realizar las pruebas.

```

|| [root@laboratorio etc]# service named restart
Deteniendo named: [ OK ]
Iniciando named: [ OK ]

```

Comando 13: Reinicio del servicio named

```

|| [root@laboratorio etc]# named-checkconf

```

Comando 14: Realizando las pruebas con el comando name-checkconf

```

|| [root@laboratorio etc]# named-checkzone miequipo.cl named.db
.miequipo.cl
zone miequipo.cl/IN: loaded serial 2015100617
OK

```

Comando 15: Realizando pruebas con el comando name-checkzone

### 2.2.2. Configuración de apache.

Primer paso para configurar el servicio de http es entrar en el archivo de configuración que se encuentra en “etc/httpd/conf” y luego editarlo.

```
|| [root@laboratorio ~] # cd /etc/httpd/conf/  
|| [root@laboratorio conf] # vi httpd.conf
```

Comando 16: Ruta y acceso a la configuración de httpd.conf

Dentro de httpd.conf buscar la parte de los host virtuales y agregar la configuración. en nuestro caso es la siguiente.

- Des-comentar “NameVirtualHost \*:80”.
- Agregar los 2 servidores virtuales.
  - home.
  - casa.
- Agregar la ruta de los archivo a mostrar.
- Definir en ServerName para estos.
- Agregar el comando “DirectoryIndex index.html” en cada uno de los tag.

```
# Use name-based virtual hosting.  
#  
NameVirtualHost *:80  
#  
# NOTE: NameVirtualHost cannot be used without a port specifier  
# (e.g. :80) if mod_ssl is being used, due to the nature of the  
# SSL protocol.  
#  
#  
# VirtualHost example:  
# Almost any Apache directive may go into a VirtualHost container.  
# The first VirtualHost section is used for requests without a known  
# server name.  
  
<VirtualHost *:80>  
    DocumentRoot /var/www/html/home  
    ServerName home.miequipo.cl  
    DirectoryIndex index.html  
</VirtualHost>  
  
<VirtualHost *:80>  
    DocumentRoot /var/www/html/casa  
    ServerName casa.miequipo.cl  
    DirectoryIndex index.html  
</VirtualHost>
```

Comando 17: Resultado de los cambios en httpd.conf

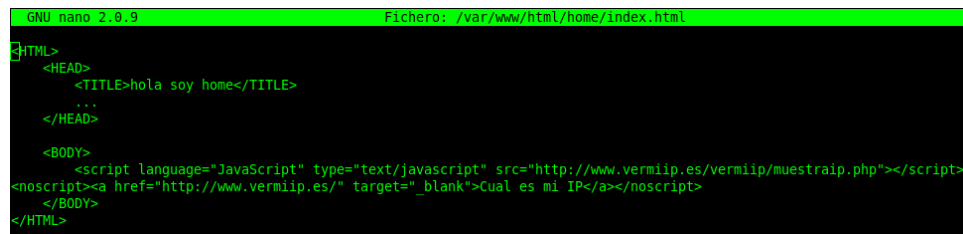
Ahora solo queda crear las carpetas, los archivos Index y reiniciar el servicio HTTPD.

```
[root@laboratorio conf]# cd /var/www/html/  
[root@laboratorio html]# mkdir home  
[root@laboratorio html]# mkdir casa
```

Comando 18: Acceso y creación de directorios

```
[root@laboratorio home]# nano /var/www/html/home/index.html  
[root@laboratorio home]# nano /var/www/html/casa/index.html
```

Comando 19: creación de los archivos Index



```
GNU nano 2.0.9 Fichero: /var/www/html/home/index.html  
HTML>  
<HEAD>  
  <TITLE>hola soy home</TITLE>  
  ...  
</HEAD>  
<BODY>  
  <script language="JavaScript" type="text/javascript" src="http://www.vermiip.es/vermiip/muestraip.php"></script>  
<noscript><a href="http://www.vermiip.es/" target="_blank">Cual es mi IP</a></noscript>  
</BODY>  
</HTML>
```

Figura 1: Contenido de Index usando nano.

```
[root@laboratorio home]# service httpd restart  
Parando httpd: [ OK ]  
Iniciando httpd: httpd: apr_sockaddr_info_get() failed for  
laboratorio  
httpd: Could not reliably determine the server's fully  
qualified domain name, using 127.0.0.1 for ServerName  
[ OK ]
```

Comando 20: Reinicio del servicio HTTPD

Como ultima consideración queda el de habilitar los puertos 53 tanto para tcp como utp. Agregar la maquina, en el archivo Hosts ubicado en la carpeta /etc, agrega en nuestro caso “158.251.93.141 miequipo.cl Server”. Cambiar la configuración del FireWalls.

### 2.3. Resultados parte 1.

Los resultado obtenido son que al ingresar con los sub-dominio nos redirecciona a distinta pagina apartir de un mismo dominio y una sola IP.

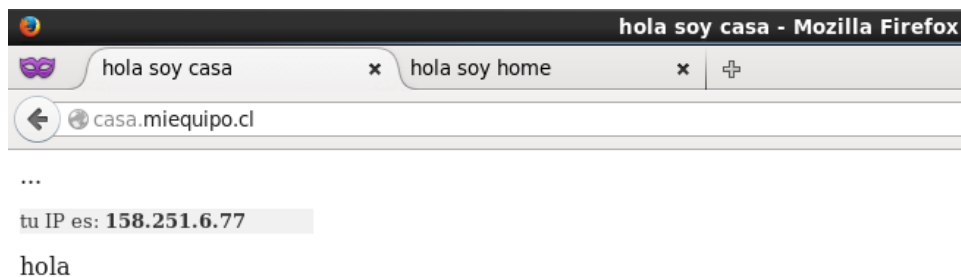


Figura 2: Resultados obtenidos en casa.miequipo.cl.

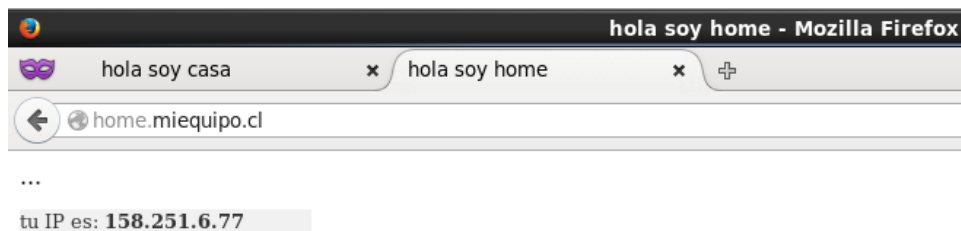
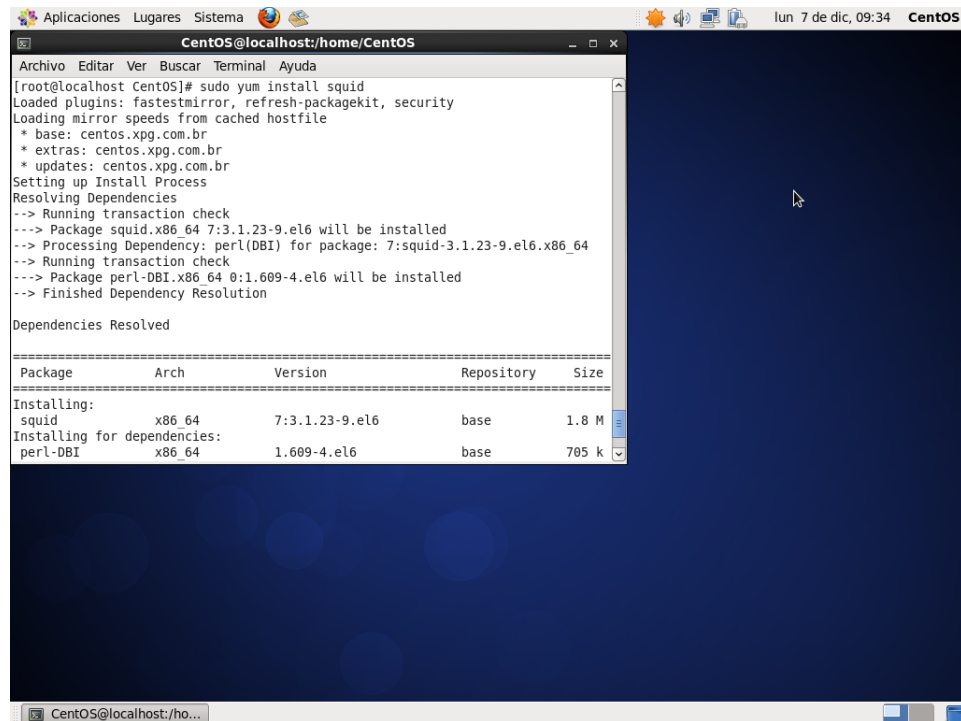


Figura 3: Resultados obtenidos en home.miequipo.cl.

## 2.4. Instalar y configurar un servidor proxy SQUID

Primero deberemos instalar el servidor SQUID colocando el código:

```
#sudo yum install squid
```



```
CentOS@localhost/home/CentOS
Archivo Editar Ver Buscar Terminal Ayuda
[root@localhost CentOS]# sudo yum install squid
Loaded plugins: fastestmirror, refresh-packagekit, security
Loading mirror speeds from cached hostfile
* base: centos.xpg.com.br
* extras: centos.xpg.com.br
* updates: centos.xpg.com.br
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package squid.x86_64 7:3.1.23-9.el6 will be installed
--> Processing Dependency: perl(DBI) for package: 7:squid-3.1.23-9.el6.x86_64
--> Running transaction check
--> Package perl-DBI.x86_64 0:1.609-4.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package             Arch      Version      Repository    Size
=====
Installing:
squid                x86_64    7:3.1.23-9.el6      base          1.8 M
Installing for dependencies:
perl-DBI              x86_64    1.609-4.el6         base           785 k
=====
```

Figura 4: Instalando Servidor SQUID

ya instalado procederemos a nombrar el PROXY, donde primero tendremos que ingresar a la configuración del SQUID ingresando el código: `/etc/squid.conf` ya adentro procederemos a configurar puerto el cual corra, el directorio de la pagina, Tamaño máximo de los archivos a almacenar y la mascara de la red



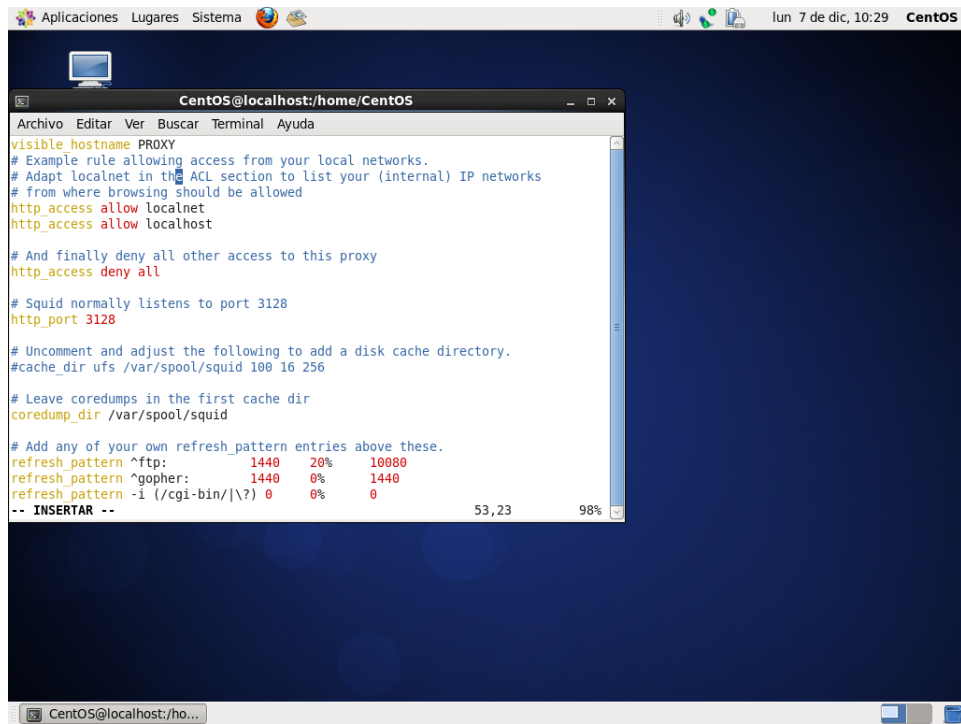


Figura 5: Nombrar el PROXY

Configuraremos que el puerto de entrada o a .<sup>escuchar</sup>será el 1234

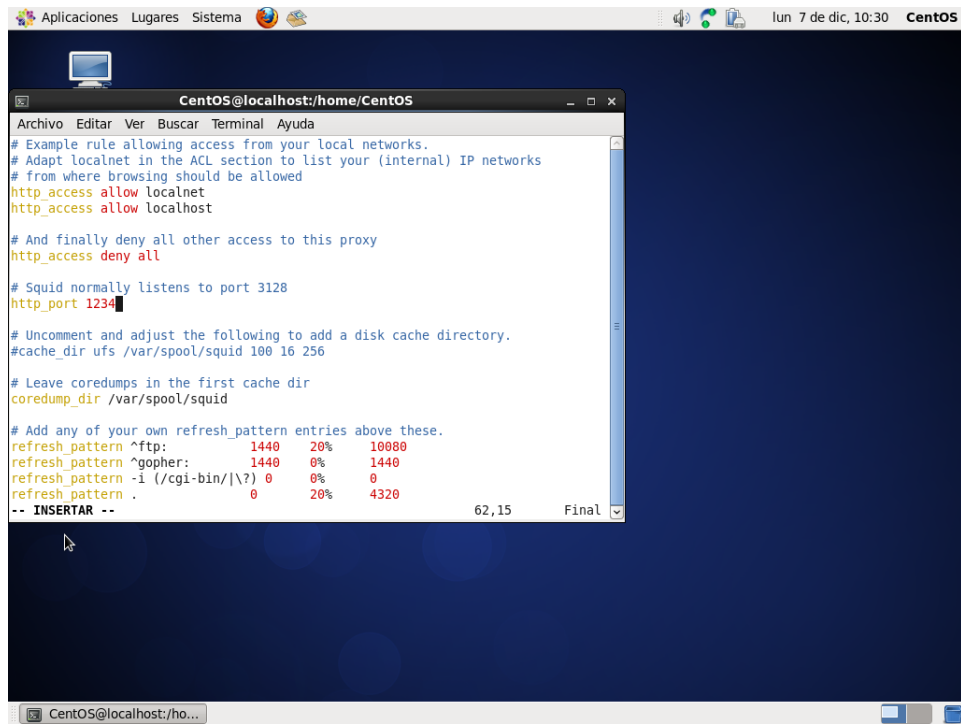


Figura 6: determinar que el puerto a escuchar es 1234

Definimos un nuevo grupo llamado `arealocal`; que abarca la red local para definir autorizaciones en la red.

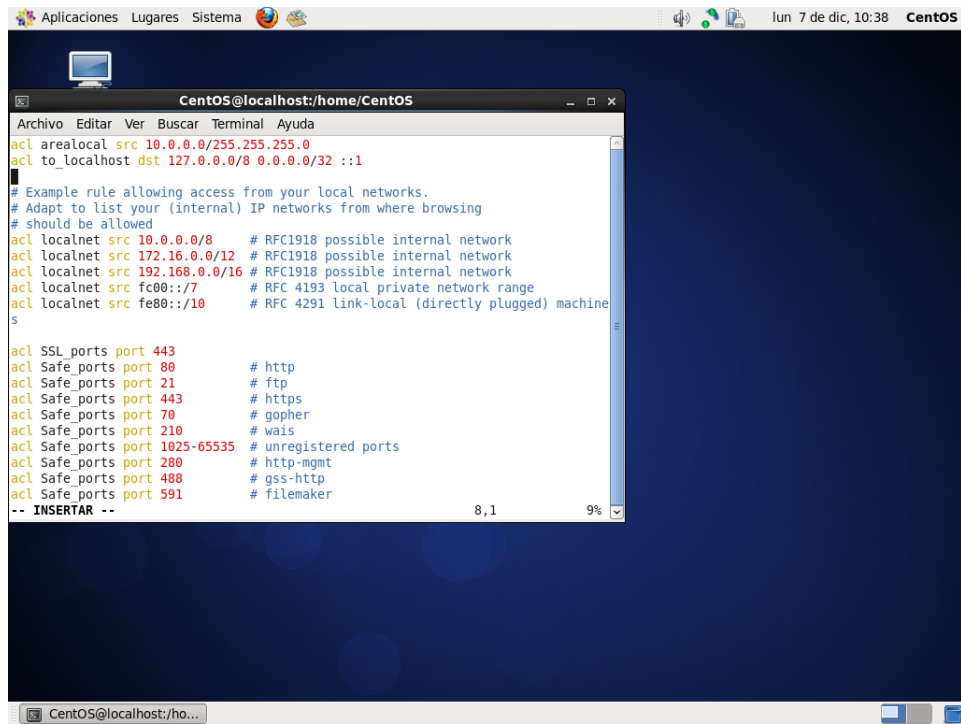
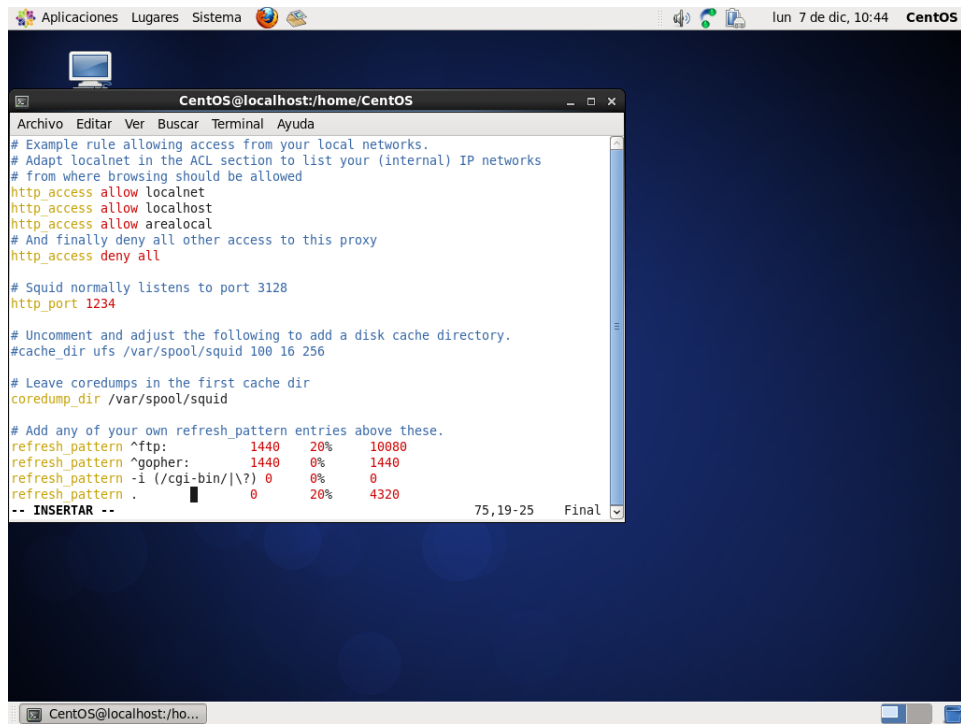


Figura 7: Creando grupo llamado arealocal

Permitimos al grupo arealocal acceso mediante el comando: **http\_access allow**



```
CentOS@localhost:/home/CentOS
Archivo Editar Ver Buscar Terminal Ayuda
# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed
http_access allow localnet
http_access allow localhost
http_access allow arealocal
# And finally deny all other access to this proxy
http_access deny all

# Squid normally listens to port 3128
http_port 1234

# Uncomment and adjust the following to add a disk cache directory.
#cache_dir ufs /var/spool/squid 100 16 256

# Leave coredumps in the first cache dir
coredump_dir /var/spool/squid

# Add any of your own refresh_pattern entries above these.
refresh_pattern ^ftp:      1440      20%      10080
refresh_pattern ^gopher:   1440      0%       1440
refresh_pattern -i (/cgi-bin/|\?) 0      0%       0
refresh_pattern .          0         20%     4320
-- INSERTAR --
```

Figura 8: Otorgando acceso a arealocal

### **3. Conclusión**

Aplicando lo aprendido en los talleres anteriores, utilizando IP Tables y servidores APACHE logramos realizar los objetivos que se plantearon para este taller, el cual consistía de acceder a dos páginas web mediante a una misma IP. Ahora seremos capaces de poder configurar un servidor para una página web, como también poder configurar un servidor a nivel intermedio.

Por otro lado, aprendimos la complejidad que poseen estos servidores, puesto a que si uno no tiene cuidado y realiza el más mínimo error como ingresar mal un dígito numérico u olvidar reiniciar los servidores, el servidor podría presentar problemas al punto de llegar a colapsar. Dependiendo del nivel al que ocurrió el error, se podría llegar hasta al punto de tener que volver a configurar los servidores desde 0.