

Universidad Luterana Salvadoreña



Facultad de Ciencias del Hombre y la Naturaleza

Proyecto:

Servidor DNS.

Catedrático:

Ing. Manuel Flores.

Cátedra:

Redes I

Estudiantes: N^o de Carnet

Ismael Díaz Martínez DMO1121472

Armando Antonio Pineda López PLO1121393

Fecha:

Sábado 25 de mayo de 2013

Índice

Tabla de contenido

INTRODUCCION	3
Objetivos del Proyecto	4
Objetivo General:	4
Objetivo Específico:	4
Generalidades del proyecto	5
Base teórica	6
Funcionamiento del servicio de DNS en la web	7
¿Qué es BIND9?	7
¿Qué es APACHE?	8
¿Que es Proftpd?	8
Características del software	9
Descripción del proyecto.....	9
Diagrama de red	11
Configuración del Servidor DNS Y APACHE	11
Configuración del servidor DNS y FTP	20
Diagrama de Gantt	27
Factibilidad del proyecto.....	28
Conclusiones:	29
Bibliografía:	30

INTRODUCCION

Los servidores DNS realizan un papel muy importante dentro de Internet y redes locales pues son quienes guían y los permiten buscar un dominio por medio de un nombre fácil de recordar ejemplo uls.sytes.net en ves estar recordando o digitando en los navegadores IP numéricas como por ejemplo 255.333.322.322.

Basándose del escrito anterior se pretenderá ejecutar un servidor máster que es quien resolverá las peticiones hechas por clientes, pero en ocasiones suelen suceder fallos en los servidores y es por eso que se debe esclavizar una maquina como servidor secundario que se ejecutara automáticamente cuando el máster no responda y no se caiga el servicio.

Componentes

Para la operación práctica del sistema DNS se utilizan tres componentes principales:

Los Clientes fase 1: Un programa cliente DNS que se ejecuta en la computadora del usuario y que genera peticiones DNS de resolución de nombres a un servidor DNS (Por ejemplo: ¿Qué dirección IP corresponde a nombre de dominio?);

Los Servidores DNS: Que contestan las peticiones de los clientes. Los servidores recursivos tienen la capacidad de reenviar la petición a otro servidor si no disponen de la dirección solicitada.

Y las Zonas de autoridad, porciones del espacio de nombres raros de dominio que almacenan los datos. Cada zona de autoridad abarca al menos un dominio y posiblemente sus subdominios, si estos últimos no son delegados a otras zonas de autoridad

Objetivos del Proyecto

Objetivo General:

- Implementar un Servidor DNS maestro y esclavo que traduzca IP a nombre de dominio fácil de recordar, para un servidor web y ftp en una Intranet.

Objetivo Específico:

- Realizar investigaciones de campo para conocer y definir las operaciones que deberá realizar el servidor.
- Diseñar una Mini-red .donde se pueda probar y configurar la interfaz de dichos servidores para su correcto funcionamiento.

Generalidades del proyecto

Título del proyecto

Nombre
Implementación de un servidor DNS

Equipo operacional de proyecto

Apellidos	Nombres	Profesión / Especialidad	Función técnica	% de dedicación
Pineda López	Armando Antonio	Lic. Computación	Diseñador - implementador	100%
Díaz Martínez	Ismael	Lic. Computación	Diseñador - implementador	100%

Base teórica

El servicio DNS “Domain Name System”, se utiliza para traducir un nombre de dominio en direcciones IP. DNS permite ya una vez configurado que tu sitio WEB y Correo sean localizados desde cualquier parte de la WWW. Los DNS se utilizan para distintos propósitos:

- Resolución de nombres.
- Resolución inversa de direcciones.
- Resolución de servidores de correo.

Existen varios tipos de servidores de DNS como Bind, PowerDNS, djbdns y todos trabajan sobre el puerto 53 protocolo TCP/UDP.

Tipos de DNS

Existen 4 tipos de servidores DNS:

- Maestro: Nuestro servidor se comportara como un auténtico servidor DNS, ya que atenderá las peticiones de resolución de nombres. Así mismo responde a consultas de otros servidores DNS.
- Esclavo: Este tipo de servidor solamente sirve como espejo de un servidor DNS Maestro, cuando el servidor DNS Maestro tiene alguna modificación, se vera reflejado en servidor DNS esclavo ya que están sincronizados.
- Cache: Este tipo de servidor se utilizan dentro de una red local, cuando hace una consulta a un servidor DNS Cache y no contiene la resolución envía una petición a un DNS Maestro y la resolución quedara guardada en la cache del DNS local hasta que expire el tiempo de vida.
- Reenvío: Reenvía las peticiones a una lista especifica de servidores DNS para la resolución de nombres.

Un servidor DNS puede ser de varios tipos configurados en el mismo servidor DNS.

En resumen 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 TCP/IP, como Internet, usan DNS para buscar equipos y servicios mediante nombres descriptivos.

Para que el uso de los recursos de red sea más fácil, 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.

Funcionamiento del servicio de DNS en la web

El funcionamiento es el siguiente, cuando ponemos por ejemplo, turismotour.com en la barra de direcciones del navegador, este realiza la consulta en Internet de cómo está configurado este dominio. El servidor DNS le indica a nuestro navegador que tiene que ir a buscar la información de la página web a la IP del servidor web.

El navegador envía la petición de la página web al servidor web, indicándole el nombre del dominio que desea. El servidor web sirve la página web y el navegador la muestra. Todo esto pasa en cuestión de milésimas de segundo.

Servidor DNS que utilizaremos para nuestro proyecto “BIND”

¿Qué es BIND9?

Bind (Berkeley Internet Name Domain), es el servidor de DNS más utilizado en el internet, fue escrito en los años 80 bajo el patrocinio de la DARPA. Actualmente se encuentra en la versión 9 del BIND, la cual fue escrita desde cero para poder superar las dificultades de las diferentes plataformas.

Cuando nos damos a la tarea de configurar un Servidor DNS para una Red, debemos escoger el paquete o programa que mejor nos funcionara para lo que queremos. En esta caso, seleccionaremos el Bind9, que es un Servidor DNS, disponible en muchas versiones y que además brinda gran flexibilidad con este servicio.

ISC BIND

Berkeley Internet Name Domain es la implementación del DNS más importante y usada en toda la Internet.

- Open Source
- Robusto
- Corazón de toda la Internet: root servers, TLDs
- Versiones de BIND: (actualizado hasta el 06/08/2001)

- Bind 4: 4.9.8 (obsoleto)

- Bind 8: 8.2.4
- Bind 9: 9.1.3

Cabe mencionar que para que nuestros clientes puedan ver en pantalla nuestro sitios web, es necesario también configurar un servidor web ya que es quien tiene alojados los dominios (Sitios web). Utilizaremos como servidor web a APACHE

¿Qué es APACHE?

El es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/HTTPS . Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que Behelendorf quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU, y en esos momentos la preocupación de su grupo era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de internet.

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP más usado. Alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años. (Estadísticas históricas y de uso diario proporcionadas por Netcraft³).

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los usuarios locales malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache.

¿Que es Proftpd?

ProFTPD es uno de los mejores servidores ftp que existen para Linux (también se lo puede utilizar en otros sistemas operativos). Es seguro, sumamente flexible, modular y fácil de configurar. Entre todas las características que tiene, las más rescatables son: permite autenticar usuarios con casi cualquier cosa, se pueden utilizar servidores virtuales de ftp, se pueden tener múltiples servidores brindando servicio de ftp anónimo, entre otras. Además no está basado en ningún otro servidor, su código fuente fue escrito totalmente desde cero.

ProFTPD surgió del deseo de tener un servidor FTP configurable y seguro. Cuando el proyecto comenzó, el servidor más utilizado era wu-ftp. Mientras que wu-ftp proporciona un rendimiento excelente y es generalmente un buen producto, carece de muchas características que se encuentran en los nuevos servidores FTP, además presenta un historial de seguridad pobre. Muchas personas, incluyendo a los desarrolladores que trabajan en ProFTPD, habían pasado mucho tiempo corrigiendo errores debido a que los ataques a servidores wu-ftp eran muy frecuentes. Lamentablemente, pronto quedó claro que un rediseño completo era necesario para aplicar la configuración y las características deseadas.

Características del software

- Un único archivo de configuración principal, con las directivas y los grupos de directiva que son intuitivos para cualquier administrador que haya utilizado el servidor web Apache.
- Permite la configuración de varios servidores FTP virtuales.
- Diseñado para funcionar tanto como un servidor independiente o desde inetd / xinetd, dependiendo de la carga del sistema.
- Permite mantener un directorio raíz anónimo, no requiere ninguna estructura de directorios específicos, los binarios del sistema o de otros archivos del sistema.
- El código fuente está disponible (y siempre deben estar disponibles) para los administradores y desarrolladores.
- Oculta los directorios y archivos, con base en los permisos al estilo Unix o usuario / propietario del grupo.
- Se ejecuta como un usuario sin privilegios configurables en modo stand-alone con el fin de disminuir las posibilidades de ataques.
- Su diseño es modular, permitiendo ampliar su funcionalidad. Algunos de los principales módulos son; para bases de datos SQL, servidores LDAP, cifrado SSL / TLS, soporte RADIUS, etc.
- Es compatible con IPv6.

Descripción del proyecto

El proyecto consiste en implementar un servidor de DNS que pueda funcionar desde una Intranet el propósito será convertir peticiones IP a nombres de dominios fácil de recordar

ejemplo www.proyectodns.com (para servidor web)y ftp.proyectodns.com(para servidor ftp)

La implementación debe poseer un servidor máster y un esclavo que estará sincronizado con el servidor maestro para impulsar mas el servicio y sea eficiente evitando así sobre saturaciones, caídas etc . También se configurara apache para que pueda alojar el sitio web www.proyectodns.com , además de hacer una interfaz entre el servidor web Apache Y BIND para que podamos cumplir nuestro objetivo. Además de configurar un servidor proftpd (ftp) para mostrar cuyo contenido que tendrá el dominio ftp.proyectodns.com.

Se usara una maquina como cliente dentro de la red para realizar pruebas del servidor.

Las maquinas estarán bajo administración del Sistema Operativo Linux (Debian 7)..

Para mejor comprensión se creo el siguiente Diagrama de funcionamiento con un sitio y una ip hipotética del servidor DNS y el Servidor web Apache. :

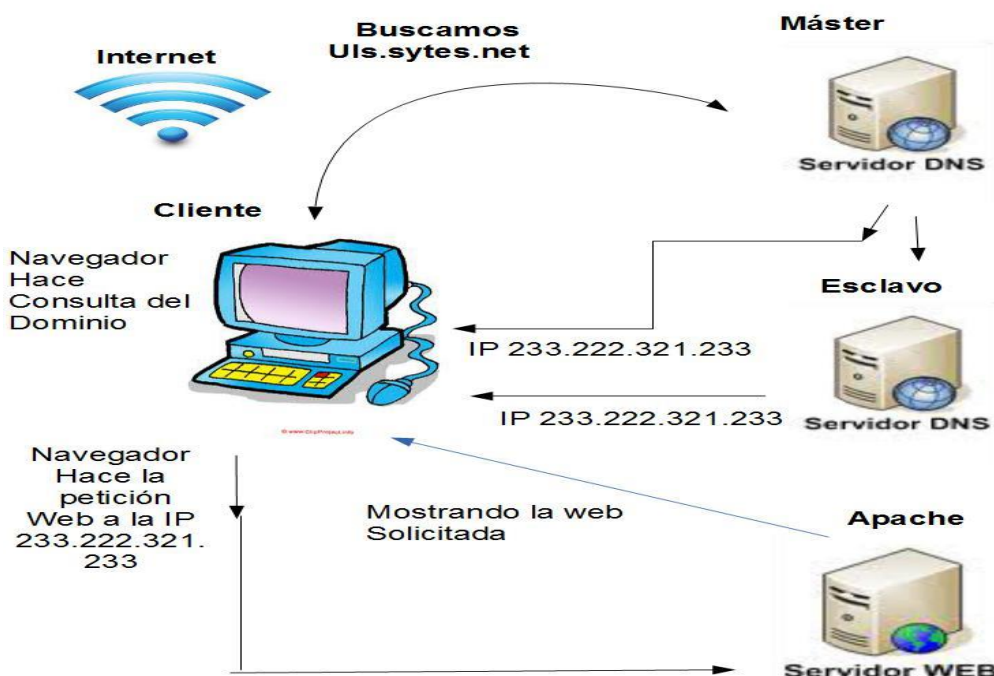
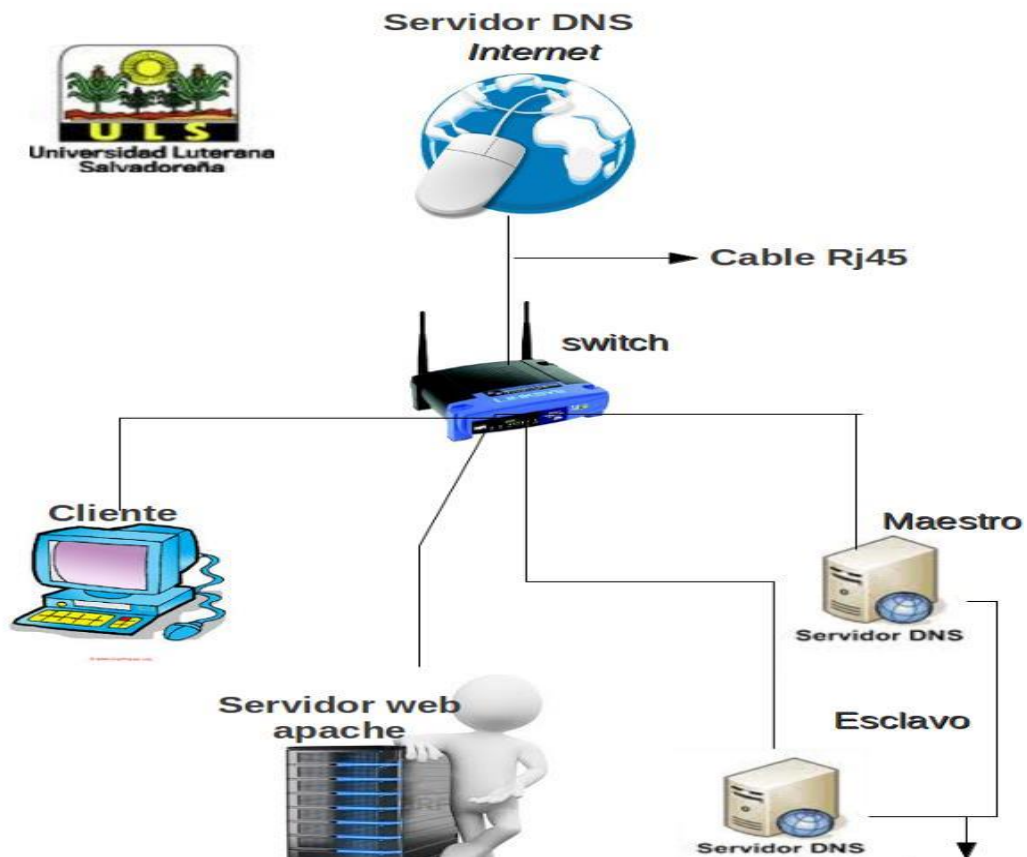


Diagrama de red

El diagrama describe como sera implementado el proyecto y los dispositivos que se involucran en el mismo.



Configuración del Servidor DNS Y APACHE

Se resolverá el dominio www.proyectodns.com, con la ip 192.168.1.4 y viceversa (esta ip pueda variar de acuerdo a la movilidad del proyecto)

Para instalar el servidor BIND se hace con el siguiente comando:

```
# apt-get install bind9 o aptitude install bind9
```

El servidor bind es instalado en sistemas de archivo, en la carpeta etc/ localizado como carpeta bind.

En el bind se configura la zonas y se le asigna una ip al servidor, esta ip se obtiene do correr el comando ifconfig desde la consola , en este caso utilizaremos la ip de la conexión eth0.

Para configurar las zonas es necesario configurar zonas de búsquedas directas e inversas y decirle a nuestro servidor, que también existirá un servidor DNS esclavo, para realizar esta operación es necesario ingresar al archivo named.conf.local, lo hacemos con el siguiente comando

```
nano /etc/bind/named.conf.local
```

NOTA: todo lo que lleve “//” se tomara como un comentario en este caso utilizaremos esto para dar una breve explicación de la configuraron, en la configuración física de los archivos no lo pondremos para fines organizativos aunque no afectase nada.

```
//
```

```
// Do any local configuration here
```

```
//
```

```
// Consider adding the 1918 zones here, if they are not used in your
```

```
// organization
```

```
//include "/etc/bind/zones.rfc1918";
```

```
// Archivo para búsquedas directas
```

```
zone "proyectodns.com" { // es el dominio que utilizaremos para nuestro proyecto.
```

```

type master; // especificamos que es nuestro servidor maestro.

file "/etc/bind/db.directa"; // búsquedas directas

};

allow-query { any; };

allow-transfer { 192.168.1.3; }; // le indicamos que existe un servidor esclavo servidor
esclavo con la ip 192.168.1.3.

// Archivo para búsquedas inversas

zone "4.1.168.192.in-addr.arpa" { //ip de nuestro servidor 192.168.1.4

type master;

file "/etc/bind/db.inversa"; // busqueda inversa

};

```

posteriormente utilizamos el archivo db.local como plantilla (este ya viene por default en la instalación de bind) lo copiaremos y crearemos el archivo db.directa de este mismo.

Lo podemos hacer desde interfaz gráfica o también desde la consola con el comando: cp /etc/bind/db.local /etc/bind/db.directa , se modifica el archivo que quedaría de la siguiente forma.

Archivo db.directa contenido:

```

;

; BIND data file for local loopback interface

;

$TTL 604800

@ IN SOA ns1.proyectodns.com. hostmaster.proyectodns.com . ( // registro "SOA" por
que brinda la información de nuestro DNS primario.

```

2 ; Serial

604800 ; Refresh

86400 ; Retry

2419200 ; Expire

604800) ; Negative Cache TTL

;

@ IN NS ns1.proyectodns.com. // el NS los permitirá tener la asociación entre los dominios que estamos traduciendo y los servidores que mostraran cuyo contenido.

@ IN NS ns2.proyectodns.com. // por que otro archivo NS por que es el de nuestro servidor esclavo.

ns1 IN A 192.168.1.4 // ip del servidor maestro y el registro tipo "A" traduce nombres de dominio a direcciones ip. En este caso la de nuestro servidor primario.

ns2 IN A 192.168.1.3 // ip del servidor esclavo y registro tipo "A" traduce nombres de dominio a direcciones ip. En este caso la de nuestro servidor secundario.

www IN A 192.168.1.5 // aquí le asignamos a un servidor web una ip de un determinado dominio en el caso de nuestro proyecto el dominio www.proyectodns.com además de hacer la búsqueda de este host o dominio promedio de la dirección ip.

FTP IN A 192.168.1.4 // aquí le asignamos a un servidor FTP una ip de un determinado dominio en el caso de nuestro proyecto el dominio ftp.proyectodns.com además de hacer la búsqueda de este host o dominio promedio de la dirección ip.

Ahora crearemos el archivo db.inversa para eso utilizaremos el archivo db.127 como plantilla y lo copiaremos a un nuevo archivo llamado db.inversa con el comando cp /etc/bind/db.127 /etc/bind/db.inversa

;

; BIND reverse data file for local loopback interface

;

\$TTL 604800

@ IN SOA ns1.proyectodns.com. hostmaster.proyectodns.com. (// registro "SOA" por que brinda la información de nuestro DNS primario.

1 ; Serial

604800 ; Refresh

86400 ; Retry

2419200 ; Expire

604800) ; Negative Cache TTL

;

@ IN NS ns1.proyectodns.com. // se explico anteriormente

@ IN NS ns2.proyectodns.com. // se explico anteriormente

48 IN PTR ns1.proyectodns.com. // registro "PTR" funciona a la inversa del registro "A" traduciendo nombres de ip a nombres de dominio, es decir que los permite hacer la búsqueda en el navegador con el nombre del dominio en este caso www.proyectodns.com

13 IN PTR ns2.proyectodns.com. // registro "PTR" funciona a la inversa del registro "A" traduciendo nombres de ip a nombres de dominio, es decir que los permite hacer la búsqueda en el navegador con el nombre del dominio en este caso www.proyectodns.com, funcionando para servidor esclavo.

41 IN PTR www.proyectodns.com. // registro "PTR" funciona a la inversa del registro "A" traduciendo nombres de ip a nombres de dominio, es decir que los permite hacer la búsqueda en el navegador con el nombre del dominio en este caso www.proyectodns.com, cuyo contenido lo mostrara el servidor web.

48 IN PTR ftp.proyectodns.com. registro "PTR" funciona a la inversa del registro "A" traduciendo nombres de ip a nombres de dominio, es decir que los permite hacer la búsqueda en el navegador con el nombre del dominio en este caso ftp.proyectodns.com.

Ahora se configuran los reenviadores en este caso utilizaremos el reenviador de google 8.8.8.8; y el de nuestro servidor 192.168.1.4 nano /etc/bind/named.conf.options

```
options {
```

```
directory "/var/cache/bind";
```

```
// If there is a firewall between you and nameservers you want
// to talk to, you may need to fix the firewall to allow multiple
// ports to talk. See http://www.kb.cert.org/vuls/id/800113

// If your ISP provided one or more IP addresses for stable
// nameservers, you probably want to use them as forwarders.

// Uncomment the following block, and insert the addresses replacing
// the all-0's placeholder.
```

```
forwarders {
8.8.8.8;
};
```

```
//=====
=====
```

```
// If BIND logs error messages about the root key being expired,
// you will need to update your keys. See https://www.isc.org/bind-keys
[ 26 líneas escritas ]
```

Configuraron del Servidor DNS esclavo se instala el bind en la maquina que utilizaremos como esclava, entramos en la carpeta /etc/bind/named.config.local y agregamos el siguiente contenido:

```
//

// Do any local configuration here

//

// Consider adding the 1918 zones here, if they are not used in your
```



```
// organization

//include "/etc/bind/zones.rfc1918";

zone "proyectodns.com" {

type slave; //indicamos que es el servidor esclavo

file "/etc/bind/db.directa";

allow-query { any; };

masters { 192.168.1.4; }; // esta bajo manejo del servidor maestro

};
```

Reiniciamos el servidor para que coja las configuraciones `/etc/init.d/bind9 restart`

Comprobamos que los servicios estén bien configurados con `nslookup ns1` , `nslookup www` ,
`nslookup 192.168.1.4` ,`nslookup` , `nslookup ftp`,

Hay que recordar que el servidor DNS es incapaz de mostrar un sitio web por si solo, es por eso que es necesario configurar también el servidor web, en este caso utilizaremos apache.

Configuración de apache junto al servidor DNS Bind9

Instalación de apache `apt-get install apache2` o `aptitude install apache2`

Apache también se instala en la carpeta `/etc/`

Abrimos la terminal como "Root"

corremos el nautilus para obtener permisos de modificación , en sistema de archivos seguimos la siguiente ruta `/etc/apache2/sites-available/`

Cuando ya estemos en `sites-available` podemos usar como plantilla el archivo `default` y creamos a partir de este el host www.proyectodns.com

Que cuya configuración seria la siguiente

```
<VirtualHost 192.168.1.5:80> // la ip es la que le asigna el bind a nuestro dominio

ServerAdmin webmaster@proyectodns.com // correo del administrador

ServerName www.proyectodns.com // nombre del servidor

ServerAlias www.proyectodns.com

DocumentRoot /var/www/host/ //al seguir esta ruta es donde tenemos el contenido de
nuestro host

<Directory />

Options FollowSymLinks

AllowOverride None

</Directory>

<Directory /var/www/>

Options Indexes FollowSymLinks MultiViews

AllowOverride None

Order allow,deny

allow from all

</Directory>

ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/

<Directory "/usr/lib/cgi-bin">

AllowOverride None

Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch

Order allow,deny

Allow from all

Redes I
```

```
</Directory>
```

```
ErrorLog ${APACHE_LOG_DIR}/error.log
```

```
# Possible values include: debug, info, notice, warn, error, crit,
```

```
# alert, emerg.
```

```
LogLevel warn
```

```
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

```
</VirtualHost>
```

Luego se activa el host con el comando `a2ensite www.proyectodns.com` desde la consola

posteriormente configuramos el archivo `nano /etc/resolv.conf` de nuestro servidor y también de nuestros clientes para indicarle la ip del servidor.

```
nano /etc/resolv.conf
```

```
domain proyectodns.com
```

```
search proyectodns.com
```

```
nameserver 192.168.1.4
```

Después de haber creado el host, en apache es necesario crear el contenido del dominio para eso es necesario seguir la siguiente ruta sistemas de archivos `/var/www/` dentro de la carpeta `www` creamos una carpeta llamada `host` y creamos nuestro sitio web.

```
reiniciamos el bind9 /etc/init.d/bind9 restart
```

```
reiniciamos apache /etc/init.d/apache2 restart
```

Listo abrimos el navegador web y escribimos en la URL www.proyectodns.com o 192.168.1.5 y observamos nuestra web.

Configuración del servidor DNS y FTP

Como anteriormente ya se había configurado la ip y el nombre del dominio en el servidor dns para el servidor ftp, solo explicaremos como hacer la interfaz entre el servidor BIND Y el servidor FTP.

Como servidor ftp utilizaremos Proftpd por ser muy fácil de configurar y obedecer la asignación de ip del servidor BIND.

Instalación del servidor ftp

```
# apt-get install proftpd
```

localizado en sistemas de archivo /etc/proftpd

De acuerdo a nuestra conveniencia se hizo la configuración de nuestro ftp en el archivo proftpd.conf, que quedaría de la siguiente manera:

```
# Includes DSO modules
```

```
Include /etc/proftpd/modules.conf
```

```
# Set off to disable IPv6 support which is annoying on IPv4 only boxes.
```

```
UseIPv6 off // ponemos en modo apagado por que utilizamos IPv4
```

If set on you can experience a longer connection delay in many cases.

IdentLookups off

ServerName "ftp.proyectodns.com" // nombre del servidor

ServerType standalone

DeferWelcome off

MultilineRFC2228 on

DefaultServer on

ShowSymlinks on

TimeoutNoTransfer 600

TimeoutStalled 600

TimeoutIdle 1200

DisplayLogin welcome.msg

DisplayChdir .message true

ListOptions "-l"

DenyFilter *.*/*

Use this to jail all users in their homes

DefaultRoot ~

Redes I

Port 21 is the standard FTP port.

Port 21

refresh any configured MasqueradeAddress directives every 8 hours

<IfModule mod_dynmasq.c>

DynMasqRefresh 28800

</IfModule>

MaxInstances 30

Set the user and group that the server normally runs at.

User ftp

Group ftp

Umask 022 is a good standard umask to prevent new files and dirs

(second parm) from being group and world writable.

Umask 022 022

Normally, we want files to be overwriteable.

AllowOverwrite on

TransferLog /var/log/proftpd/xferlog

SystemLog /var/log/proftpd/proftpd.log

```
<IfModule mod_quotatab.c>
```

```
QuotaEngine off
```

```
</IfModule>
```

```
<IfModule mod_ratio.c>
```

```
Ratios off
```

```
</IfModule>
```

```
# Delay engine reduces impact of the so-called Timing Attack described in
```

```
# http://www.securityfocus.com/bid/11430/discuss
```

```
# It is on by default.
```

```
<IfModule mod_delay.c>
```

```
DelayEngine on
```

```
</IfModule>
```

```
<IfModule mod_ctrls.c>
```

```
ControlsEngine off
```

```
ControlsMaxClients 2
```

```
ControlsLog /var/log/proftpd/controls.log
```

```
ControlsInterval 5
```

```
ControlsSocket /var/run/proftpd/proftpd.sock
```

```
</IfModule>
```

```
<IfModule mod_ctrls_admin.c>
```

```
AdminControlsEngine off
```

```
</IfModule>
```

```
#
```

```
# Alternative authentication frameworks
```

```
#
```

```
#Include /etc/proftpd/ldap.conf
```

```
#Include /etc/proftpd/sql.conf
```

```
#
```

```
# This is used for FTPS connections
```

```
#
```

```
#Include /etc/proftpd/tls.conf
```

```
#
```

```
# Useful to keep VirtualHost/VirtualRoot directives separated
```

```
#
```

```
#Include /etc/proftpd/virtuals.conf
```

```
# A basic anonymous configuration, no upload directories.
```



```
<Anonymous ~ftp>
User ftp
Group nogroup
# # We want clients to be able to login with "anonymous" as well as "ftp"
# UserAlias anonymous ftp
# # Cosmetic changes, all files belongs to ftp user
# DirFakeUser on ftp
# DirFakeGroup on ftp
#
# RequireValidShell off
#
# # Limit the maximum number of anonymous logins
# MaxClients 10
#
# # We want 'welcome.msg' displayed at login, and '.message' displayed
# # in each newly chdired directory.
# DisplayLogin welcome.msg
# DisplayChdir .message
#
# # Limit WRITE everywhere in the anonymous chroot
<Directory /root/home/ftp> //ruta a seguir para nuestro usuario FTP
<Limit WRITE>
DenyAll
```

```
</Limit>
</Directory>
#
# # Uncomment this if you're brave.
# # <Directory incoming>
# # # Umask 022 is a good standard umask to prevent new files and dirs
# # # (second parm) from being group and world writable.
# # Umask 022 022
# # <Limit READ WRITE>
# # DenyAll
# # </Limit>
# # <Limit STOR>
# # AllowAll
# # </Limit>
# # </Directory>
#
</Anonymous>
```

Include other custom configuration files

```
Include /etc/proftpd/conf.d/
```

Luego vamos a agregarle una contraseña al usuario ftp con el comando # passwd ftp y elegimos la que deseemos.

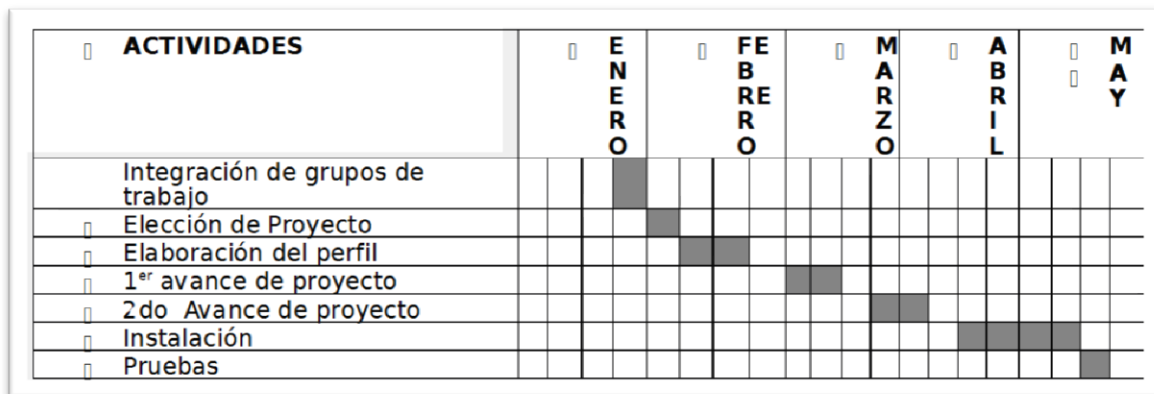
En la configuración de proftpd.conf estamos diciendo la ruta del documento que estará visible para los usuarios se crea en home un carpeta llamada ftp y dentro de esta ponemos lo que queramos que los usuarios vean o descarguen /home/ftp

Reiniciamos el servidor ftp con el comando

`/etc/init.d/proftpd restart` , listo ahora escribimos en el navegador <ftp://192.168.1.4> o <ftp.proyectodns.com> y observaremos que bind y proftpd ya están operando juntos...

También podemos hacer pruebas con el programa cliente ftp el filezilla y no habrá ningún problema..

Diagrama de Gantt



Factibilidad del proyecto.

Factibilidad Técnica

Se harán investigaciones de campo y experimentales con el objetivo de conocer cómo funciona y como está estructurado el servidor de DNS, para modernos capacitar debidamente sobre el tema.

Factibilidad Operativa

El equipo operacional del proyecto se enfocara al control y tratamiento del proyecto así, como pruebas previas a su resultado final.

Factibilidad Legal

Se trabajara bajo software libre y si se necesitase una licencia la haremos con creative commons.

Factibilidad Económica

Si alguna entidad quisiera retomar nuestro esquema y modelo de proyecto su costo seria de:

Inversión	Costo
Sistema operativo (Debian 7)	\$ 0.00
Servidor DNS (BIND)	\$ 0.00
Servidor Web (APACHE)	\$ 0.00
Servidor Ftp (Proftpd)	\$ 0.00
1 Computadora como servidor DNS	\$ 450
1 Computadora como servidor Esclavo	\$ 450
1 Computadora como servidor Apache	\$ 450
1 Computadora como servidor Proftpd	\$ 450
Instalación	\$ 200
“X” computadoras clientes	\$X.xx
Valor	\$ 2,000

Conclusiones:

De acuerdo a nuestros estudios trazados el servidor DNS juega un papel muy importante dentro de internet ya que esta llamado a resolver nombres de dominios fácil de recordar y evitarlos la tediosa tarea de andar escribiendo en la URL del navegador la ip del dominio. Lo interesante del DNS que es capaz de hacer esto aunque internet trabaje en base a IPs.

El Servidor DNS fue implementado con éxito resolviendo los dominios www.proyectodns.com , [ftp.proyectodns.com](ftp://ftp.proyectodns.com) para que nuestros posibles clientes puedan ver dicho contenido del host.

Además de apreciar al Bind como servidor DNS muy versátil a la hora de querer implementar por su robustez, y una excelentísima interfaz con el servidor web apache y Proftpd para realizar servicios de esta índole.

Bibliografía:

<http://www.youtube.com/watch?v=R18SpFzw4aQ>

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

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