UNIVERSIDAD LUTERANA SALVADOREÑA

FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA

Tema: Perfil de proyecto Final.

Cátedra: Redes I

Catedrático: Ing. Manuel Flores Villatoro.

Alumno: Jonathan Daniel Mejía Martínez.

Carnet: mm02110894

Fecha de entrega: San Salvador 22 de Febrero del 2013.

Servidor DNS, Maestro y Esclavo en Gnu/linux Ubuntu 12.10

Febrero del 2013.

INDICE GENERAL

INTRODUCCIÓN

En el siguiente documento se pretende crear una guía básica correspondiente al proyecto de fin de ciclo de la materia de Redes I de la Universidad Luterana Salvadoreña, que sirva como referencia para el uso de herramientas y técnicas de software libre, necesarias para para la instalación y configuración de un servidor DNS-MAESTRO-Y-ESCLAVO, utilizando plataformas de Sistemas Operativos GNU/LINUX.

Para el desarrollo de este proyecto de ciclo final se utilizaran dos computadoras de Escritorio con Sistemas Operativos GNU/LINUX Ubuntu 12.10, donde se configuraran los DNS maestro y esclavo, que en caso que el DNS maestro no pueda resolverse, pase a ocupar su lugar el DNS esclavo. Ambos servicios estarán asociados a un dominio el cual deberá de resolverse desde internet o una red local, ofreciendo un servicio web básico que se configurar con un CMS Drupal 7 (Sistema de gestión de contenidos), que a su veces estará asociado un servidor ssh (Secure SHell, en español: intérprete de órdenes segura) por cada uno de los DNS implementados.

Un servidor DNS (Domain Name System) es un sistema que nos permite usar nombres de dominio en lugar de direcciones IP. Su principal ventaja es que para nosotros es mucho más fácil recordar un nombre que una dirección IP.

OBJETIVOS DEL PERFIL

Objetivo general

- Configurar un servidor DNS maestro, esclavo y mostrar su funcionamiento utilizando bind9.

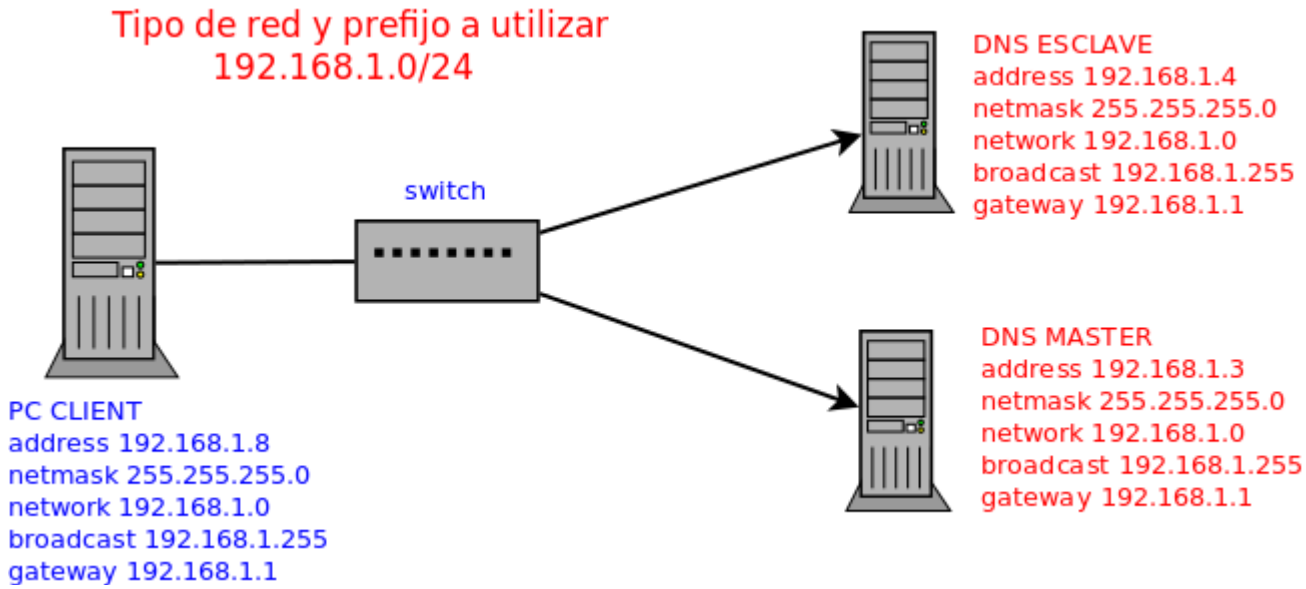
Objetivo especifico:

- Identificar y reconocer los diferentes comandos para la adecuada instalación del servidor DNS.

- Aprender a configurar un servicio ssh asociado a la instalación del servidor DNS.

- Aprender a utilizar herramientas libres que nos permitan crear, configurar, y administrar un servidor DNS y servicios web.

DIAGRAMA DE RED

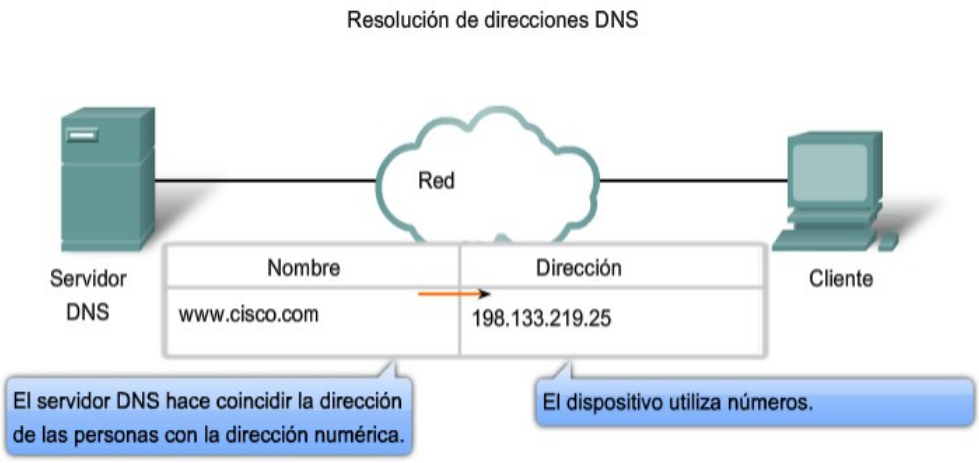


Para la demostración de la funcionalidad del dns-maestro-y-esclavo utilizaremos un switch ya que nos encontramos en la misma red de trabajo 192.168.1.0

MARCO TEORICO

Cada vez que los usuarios de internet necesitamos localizar una paginas web lo hacemos por su nombre de dominio (ej. www.google.com),esto permite al usuario acceder a las de paginas web de la Internet sin necesidad de recordar todas y cada una de las direcciones IP asociadas al nombre de la pagina que desea visitar.

Una forma de solucionar este problema es mediante la complementación de un mecanismo que al momento que un usuario pregunte por el nombre de una pagina web este servidor conozca que dirección IP le corresponde al sitio web por el cual pregunta el usuario. El mecanismo del cual hablamos es un servidor de nombres mayormente conocido como servidor DNS (Domain Name Server).



SERVIDOR DNS

El servicio DNS “Domain Name System”, se utiliza para traducir un nombre de dominio en direcciones IP. Se utiliza cuando un nodo (o host) en Internet contacta a otro mediante el nombre de domino de la máquina y no por su dirección 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.

NECESIDAD DEL DNS

En los orígenes de Internet, cuando sólo había unos cientos decomputadores conectados, la tabla con los nombres de dominio y direccionesIP se encontraba almacenada en un único computador con el nombre deHOSTS.TXT. El resto de computadores debían consultarle a éste cada vezque tenían que resolver un nombre. Este archivo contenía una estructuraplana de nombres, tal como hemos visto en el ejemplo anterior y funcionababien ya que la lista sólo se actualizaba una o dos veces por semana.Sin

embargo, a medida que se fueron conectando más computadores a la redcomenzaron los problemas: el archivo HOSTS.TXT comenzó a ser demasiadoextenso, el mantenimiento se hizo difícil ya que requería más de unaactualización diaria y el tráfico de la red hacia este computador llegó asaturarla.Es por ello que fue necesario diseñar un nuevo sistema de resolución denombres que distribuyese el trabajo entre distintos servidores. Se ideó unsistema jerárquico de resolución conocido como DNS (Domain Name System ,sistema de resolución de nombres).

TIPOS DE DNS

Existen 4 tipos de servidores DNS:

•**Maestro:** Nuestro servidor se comportara como un autentico servidor DNS, ya que atenderá las peticiones de resolución de nombres. Así mismo responde a consultas de otro 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.

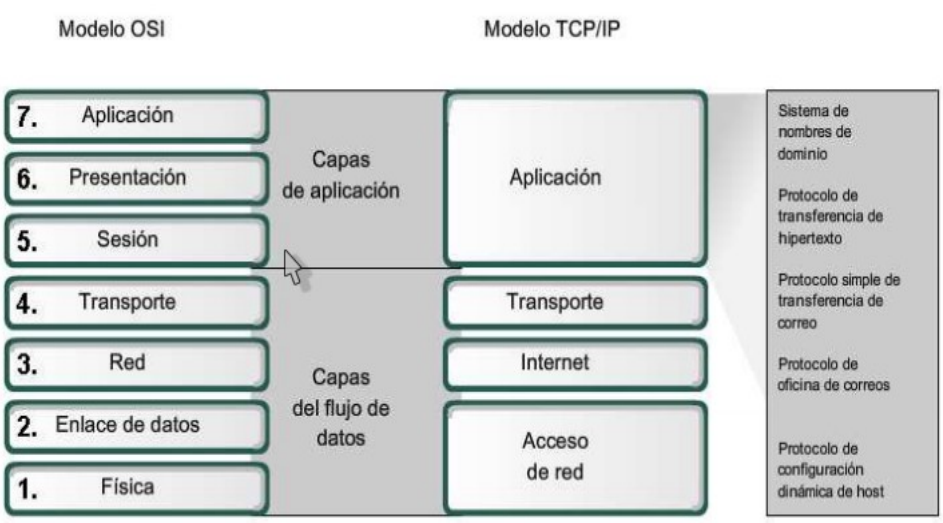
•**Reenvío: Reenvía las peticiones a una lista especifica de servidores DNS para la resolución de nombres.**

Protocolo y funcionalidad del DNS en la capa de aplicación del modelo OSI y TCP/IP

Existen varios tipos de servidores de DNS como Bind, PowerDNS, djbdns y todos trabajan sobre el puerto 53 protocolo TCP/UDP para responder a las consultas. Casi todas las consultas consisten de una sola solicitud UDP desde un Cliente DNS, seguida por una sola respuesta UDP del servidor. Se realiza una conexión TCPcuando el tamaño de los datos de la respuesta exceden los 512 bytes, tal como ocurre con tareas como transferencia de zonas.

La capa de Aplicación, Capa siete, es la capa superior de los modelos OSI y TCP/IP. Es la capa que proporciona la interfaz entre las aplicaciones que utilizamos para comunicarnos y la red subyacente en la cual se transmiten los mensajes. Los protocolos de capa de aplicación se utilizan para intercambiar los datos entre los programas que se ejecutan en los hosts de origen y destino.

Funcionamiento del dns es la de conectar un host a un sitio web por ejemplo atravez de un nombre que identifica una ip.



RFCS DE DNS

RFC's principales RFC 920: Domain Requirements
RFC 1101: DNS Enconding of Network Names and Other Types
RFC 1033 : Domain Adminstrators Operations Guide
RFC 1034: Domain Names – Concepts and Facilities
RFC 1035: Domain Names – Implementation and Specifica
tion RFC 1591: Domain Name System Structure and Delegation
RFC 1183: New RR Types.

También se está trabajando en DNS y seguridad para evitar el ataque conocido como DNS Spoofing o suplantación. RFC 2535. DNS Spoofing. Un intruso se hace pasar por un DNS. El intruso puede entregar o bien información modificada al host, o bien engañar al DNS local para que registre información en su cache. P.ej, puede hacer resolver www.mibanco.es a una IP que será la del atacante, de forma que cuando un usuario de MiBANCO se conecta, lo hará realmente con el atacante.

ACRONIMOS

LAS NIC (NETWORK INFORMATION CENTER).

NIC (acrónimo de Network Information Center o Centro de Información sobre la Red) es una institución encargada de asignar los nombres de dominio en Internet ya sean nombres de dominio genéricos o por países, permitiendo personas o empresas, montar sitios de Internet a través de un ISP, mediante un DNS. Técnicamente existe un NIC por cada país en el mundo y cada uno de éstos es responsable por todos los dominios con la terminación correspondiente a su país. Por ejemplo: **svnet** es la entidad encargada de gestionar todos los dominios con terminación .sv, la cual es la terminación correspondiente asignada a los dominios de El Salvador.

FQDN (Fully Qualified Domain Name).

FQDN (acrónimo de Fully Qualified Domain Name o Nombre de Dominio Plenamente Calificado) es un Nombre de Dominio ambiguo que especifica la posición absoluta del nodo en el árbol jerárquico del DNS. Se distingue de un nombre regular porque lleva un punto al final.
Como ejemplo: suponiendo que se tiene un dispositivo cuyo nombre de anfitrión es «maquina1» y un dominio llamado «dominio.com», el FQDN sería «maquina1.dominio.com.», así es que se define de forma única al dispositivo mientras que pudieran existir muchos anfitriones llamados «maquina1», solamente puede haber uno llamado «maquina1.dominio.com.». La ausencia del punto al final definiría que se pudiera tratar solamente de un prefijo, es decir «maquina1.dominio.com» pudiera ser un dominio de otro más largo como «maquina1.dominio.com.mx».

La longitud máxima de un FQDN es de 255 bytes, con una restricción adicional de 63 bytes para cada etiqueta dentro del nombre del dominio. Solamente se permiten los caracteres A-Z de ASCII, dígitos y el carácter «-» (guión medio). Sin distinción de mayúsculas y minúsculas.

Desde 2004, a solicitud de varios países de Europa, existe el estándar IDN (acrónimo de Internationalized Domain Name) que permite caracteres no-ASCII, codificando caracteres Unicode dentro de cadenas de bytes dentro del conjunto normal de caracteres de FQDN. Como resultado, los limites de longitud de los nombres de dominio IDN dependen directamente del contenido mismo del nombre.

COMPONENTES DE UN DNS

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

•**Los Clientes DNS:** 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.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.

•**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.

ENTENDIENDO LAS PARTES DE UN NOMBRE DE DOMINIO

Un nombre de dominio usualmente consiste en dos o más partes (técnicamente etiquetas), separadas por puntos cuando se las escribe en forma de texto. Por
ejemplo, `www.example.com` o `www.wikipedia.es`
•A la etiqueta ubicada más a la derecha se le llama dominio de nivel superior (en inglés top level domain). Como `org` en`www.ejemplo.org` o `es` en `www.wikipedia.es`

•Cada etiqueta a la izquierda especifica una subdivisión o subdominio. Nótese que "subdominio"

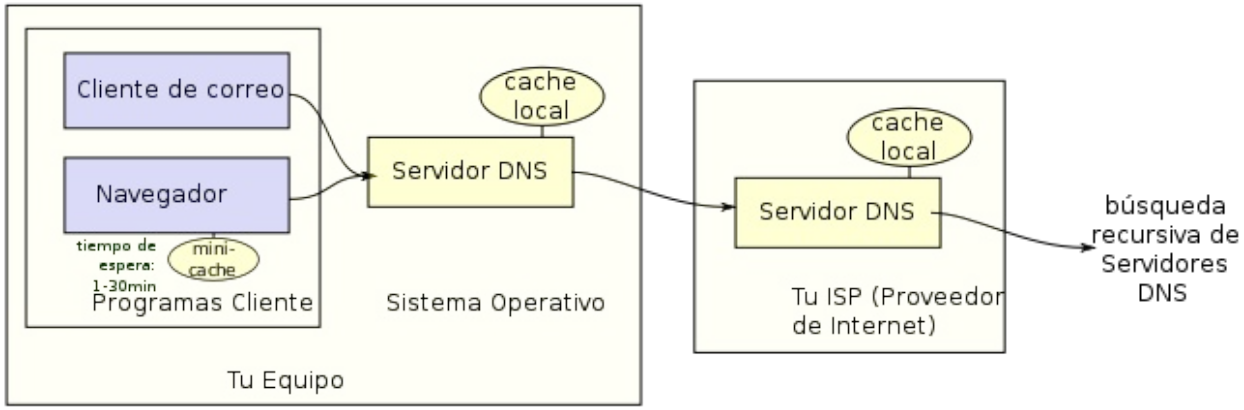
expresa dependencia relativa, no dependencia absoluta. En teoría, esta subdivisión puede tener hasta 127 niveles, y cada etiqueta puede contener hasta 63 caracteres, pero restringidos a que la longitud total del nombre del dominio no exceda los 255 caracteres, aunque en la práctica los dominios son casi siempre mucho más cortos.

•Finalmente, la parte más a la izquierda del dominio suele expresar el nombre de la máquina (en inglés hostname). El resto del nombre de dominio simplemente especifica la manera de crear una ruta lógica a la información requerida. Por ejemplo, el dominio`es.wikipedia.org` tendría el nombre de la máquina "es", aunque en este caso no se refiere a una máquina física en particular.

El DNS consiste en un conjunto jerárquico de servidores DNS. Cada dominio o subdominio tiene una o más zonas de autoridad que publican la información acerca del dominio y los nombres de servicios de cualquier dominio incluido. La jerarquía de las zonas de autoridad coincide con la jerarquía de los dominios. Al inicio de esa jerarquía se encuentra los servidores raíz: los servidores que responden cuando se busca resolver un dominio de primer y segundo nivel.

DNS EN EL MUNDO REAL

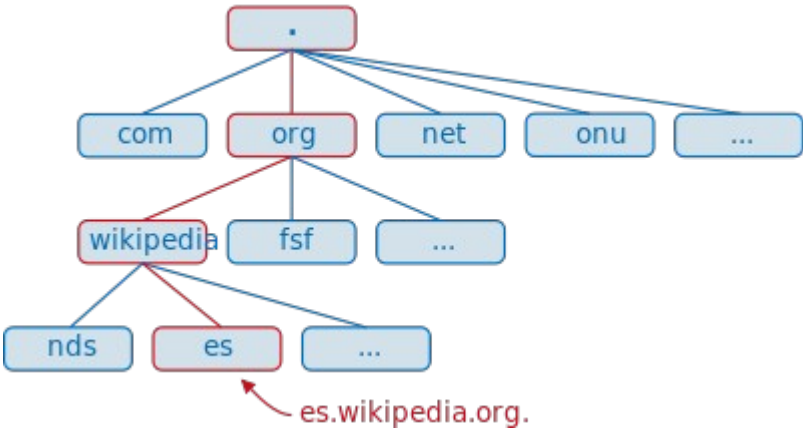
Los usuarios generalmente no se comunican directamente con el servidor DNS: la resolución de nombres se hace de forma transparente por las aplicaciones del cliente (por ejemplo, navegadores, clientes de correo y otras aplicaciones que usan Internet). Al realizar una petición que requiere una búsqueda de DNS, la petición se envía al servidor DNS local del sistema operativo. El sistema operativo, antes de establecer alguna comunicación, comprueba si la respuesta se encuentra en la memoria caché. En el caso de que no se encuentre, la petición se enviará a uno o más servidores DNS. La mayoría de usuarios domésticos utilizan como servidor DNS el proporcionado por el proveedor de servicios de Internet. La dirección de estos servidores puede ser configurada de forma manual o automática mediante DHCP. En otros casos, los administradores de red tienen configurados sus propios servidores DNS.



DNS PÚBLICOS

Uno de los servicios fundamentales para la navegación por Internet son los servidores de DNS, es decir, la traducción de un nombre de dominio o un servidor de correo a su dirección IP. Tradicionalmente, estos servicios siempre habían estado vinculados a los ISP y, cuando fallaban, el usuario tenía que andar cambiando a los de otro ISP o desesperarse hasta que se solventase la avería. Afortunadamente, existen servicios de DNS públicos y gratuitos que podemos utilizar para ganar velocidad de respuesta o una mayor disponibilidad y, entre ellos, el de Google es uno de los más utilizados. Google puso a disposición de los usuarios sus servidores de DNS en diciembre de 2009 y, poco más de dos años más tarde, están resolviendo unos 70.000 millones de peticiones cada día.

JERARQUÍA DNS



El espacio de nombres de dominio tiene una estructura arborescente. Las hojas y los nodos del árbol se utilizan como etiquetas de los medios. Un nombre de dominio completo de un objeto consiste en la concatenación de todas las etiquetas de un camino. Las etiquetas son cadenas alfanuméricas (con '-' como único símbolo permitido), deben contar con al menos un carácter y un máximo de 63 caracteres de longitud, y deberá comenzar con una letra (y no con '-') (ver la RFC 1035, sección "2.3.1. Preferencia nombre de la sintaxis "). Las etiquetas individuales están separadas por puntos. Un nombre de dominio termina con un punto (aunque este último punto generalmente se omite, ya que es puramente formal). Un FQDN correcto (también llamado Fully Qualified Domain Name), es por ejemplo este: www.example.com. (Incluyendo el punto al final)
Un nombre de dominio debe incluir todos los puntos y tiene una longitud máxima de 255 caracteres.
Un nombre de dominio se escribe siempre de derecha a izquierda. El punto en el extremo derecho de un nombre de dominio separa la etiqueta de la raíz de la jerarquía (en inglés, root). Este primer nivel es también conocido como dominio de nivel superior (TLD - Top Level Domain).

TIPOS DE RESOLUCIÓN DE NOMBRES DE DOMINIO

Existen dos tipos de consultas que un cliente puede hacer a un servidor DNS, la iterativa y la recursiva.

Resolución Iterativa

Las resoluciones iterativas consisten en la respuesta completa que el servidor de nombres pueda dar. El servidor de nombres consulta sus datos locales (incluyendo su caché) buscando los datos solicitados. El servidor encargado de hacer la resolución realiza iterativamente preguntas a los diferentes DNS de la jerarquía asociada al nombre que se desea resolver, hasta descender en ella hasta la máquina que contiene la zona autoritativa para el nombre que se desea resolver.

Resolución Recursiva

En las resoluciones recursivas, el servidor no tiene la información en sus datos locales, por lo que busca y se pone en contacto con un servidor DNS raíz, y en caso de ser necesario repite el mismo proceso básico (consultar a un servidor remoto y seguir a la siguiente referencia) hasta que obtiene la mejor respuesta a la pregunta.

El proceso de resolución normal se da de la siguiente manera:

1.El servidor A recibe una consulta recursiva desde el cliente DNS.
2.El servidor A envía una consulta recursiva a B.
3.El servidor B refiere a A otro servidor de nombres, incluyendo a C.
4.El servidor A envía una consulta recursiva a C.
5.El servidor C refiere a A otro servidor de nombres, incluyendo a D.
6.El servidor A envía una consulta recursiva a D.
7.El servidor D responde.
8.El servidor A regresa la respuesta al resolver.
9.El resolver entrega la resolución al programa que solicitó la información.

FORMATO DEL MENSAJE DNS

DNS utiliza el mismo formato de mensaje para:

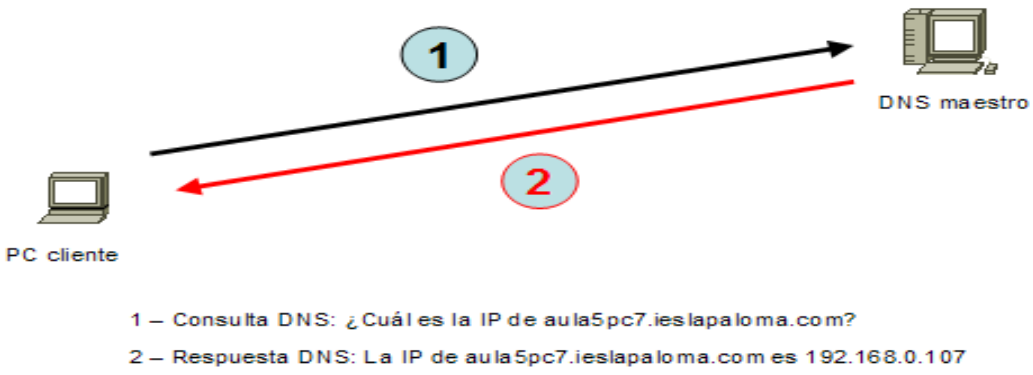
- todos los tipos de consultas de clientes y respuestas de servidor
- mensajes de error
- la transferencia de información de registros de recursos entre servidores

Encabezado	
Pregunta	La pregunta para el servidor de nombres
Respuesta	Registros de recursos que responden la pregunta
Autoridad	Registros de recursos que apuntan a una autoridad
Adicional	Registros de recursos que poseen información adicional

DNS PRIMARIO / DNS SECUNDARIO

Servidor DNS maestro

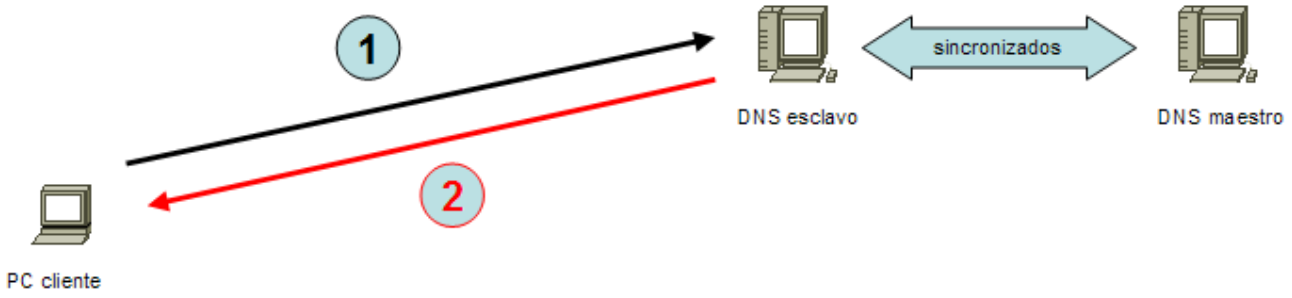
En este modo de funcionamiento, nuestro servidor se comporta como un auténtico servidor DNS para nuestra red local. Atenderá directamente a las peticiones de resolución de direcciones pertenecientes a la red local y reenviará a servidores DNS externos las peticiones del resto de direcciones de Internet.



Consulta a un DNS maestro

Servidor DNS esclavo

Un servidor esclavo actuará como un servidor espejo de un servidor DNS maestro. Permanecerá sincronizado con el maestro. Se utilizan para repartir las peticiones entre varios servidores aunque las modificaciones solo se realicen en el maestro. En redes locales salvo por razones de disponibilidad, es raro que exista la necesidad de tener dos servidores DNS ya que con uno será suficiente.



Consulta a un DNS esclavo

HERRAMIENTAS DE SOFTWARE A UTILIZAR EN EL PROYECTO

BIND es el servidor DNS mas comúnmente implementado en Sistemas Operativos Linux, y actualmente el mas usando en Internet. Originalmente BIND nació a principios de los años 80 bajo el patrocinio de DARPA (Agencia de Investigación de Proyectos Avanzados de Defensa) agencia del Departamento de Defensa de los Estados Unidos, el cual fue desarrollado en la Universidad de California, Berkeley por cuatro estudiantes. A mediados de los años 80 su desarrollo paso a manos de los empleados de DEC (Digital Equipment Corporation, compañía que mas tarde seria adquirida por Compaq y esta a su vez comprada por HP) Paul Vixie,empleado de DEC continuó trabajando en BIND luego de desvincularse de DEC. Más adelante ayudaría a fundar la ISC (Internet Systems Consortium), la cual se convirtió en la responsable del mantenimiento de BIND.

El desarrollo de BIND 9 fue realizado con el auspicio conjunto del área comercial y militar. La mayoría de las funcionalidades de BIND 9 fueron impulsadas por proveedores de UNIX quienes querían asegurar que BIND se mantuviera competente con la oferta de Microsoft en el sector de soluciones DNS.
La versión mas actual de BIND, en particular la versión 9 fue reescrita desde cero, esto con el fin de reparar algunas de sus funcionalidades arquitectónicas de la misma (problemas en la programación de Bajo Nivel) que agrega características importantes como: TSIG, notificación DNS, nsupdate, IPv6, rndc flush, vistas, procesamiento en paralelo, y una arquitectura mejorada en cuanto a portabilidad.

SERVIDOR HTTP APACHE

El servidor HTTP Apache 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/1.12 y la noción de sitio virtual. 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. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. En inglés, a patchy server (un servidor "parcheado") suena igual que Apache Server.

CONFIGURACION DEL SERVIDOR DNS CON BIND9

lo primero que debemos hacer es establecer nuestra ip como estatica, el linux ubuntu en casi todas las versiones los comandos o pasos a seguir son similares, para ellos debemos cambiar el contenido del archivo:

```
# /etc/network/interfaces.
```

PASO UNO: CONFIGURACION DE LA IP ESTATICA:

sudo nano /etc/network/interfaces:

```
iface eth0 inet static
address 192.168.1.4
netmask 255.255.255.0
gateway 192.168.1.1
auto eth0
```

luego que editar el archivo de configuración interfaces procedemos a guardas los cambios, y por ultimo ejecutamos el siguiente comando:

```
sudo service networking restart
```

cuando editemos este archivo tenemos que tener en cuenta algunos valores como:

Address: la dirección IP que tendrá nuestra máquina.
Netmask: la máscara de red. Si usamos una dirección IP de clase C, habitual en redes pequeñas, del tipo 192.168.1.x la máscara de red será 255.255.255.0
Network: la dirección de la red. Será nuestra dirección IP pero con el último valor acabado en 0.
Broadcast: la dirección de broadcast. Será nuestra dirección IP pero con el último valor acabado en 255.
Gateway: la puerta de enlace, es decir, la IP del router de salida.

ahora verificamos que nuestra ip estatica ah sido asignada correctamente con el comando ifconfig

Si hubiera algún problema, la solución típica es tirar y levantar el interfaz de red. Esto se hace de la siguiente manera:

```
sudo ifdown eth0 y sudo ifup eth0
```

esta configuración es necesaria para la dirección de nuestro servidor se mantenga siempre la misma.

INSTALACION DE BIND9

Para instalar bind9 desde la terminal de linux ubuntu utilizaremos el siguiente comando:

```
# sudo aptitude update
#sudo aptitude install bind9 bind9-doc dnsutils
```

De esta forma instalaríamos los programas necesarios para disponer de un servidor DNS. Tan solo será necesario configurarlo y ponerlo en marcha.

INSTALACIÓN DE APACHE2 CON EL COMANDO:

```
# sudo aptitude update
# sudo aptitude install apache2 apache2-mpm-prefork libapache2-mod-php5
```

CONFIGURACION DEL DNS

Archivos de configuración del DNS
El archivo de configuración del DNS es el archivo /etc/bind/named.conf, pero este hace referencia a otros cuantos archivos como por ejemplo:

Archivo	Descripción
named.conf	Archivo principal de configuración
named.conf.options	Opciones genéricas
named.conf.local	Especificación particular de este servidor DNS
db.127	Especificación dirección de retorno
db.root	DNSs de nivel superior
otros	db.0, db.255, db.empty, db.local, rndc.conf, rndc.key, zones.rfc1918

ARRANQUE Y PARADA MANUAL DEL SERVIDOR DNS

El servidor DNS, al igual que todos los servicios en Debian y Ubuntu, dispone de un script de arranque y parada en la carpeta /etc/init.d.

// Arranque del servidor DNS
/etc/init.d/bind9 start

// Parada del servidor DNS
/etc/init.d/bind9 stop

// Reinicio del servidor DNS
/etc/init.d/bind9 restart

Configuración de la zona de autoridad (SOA).

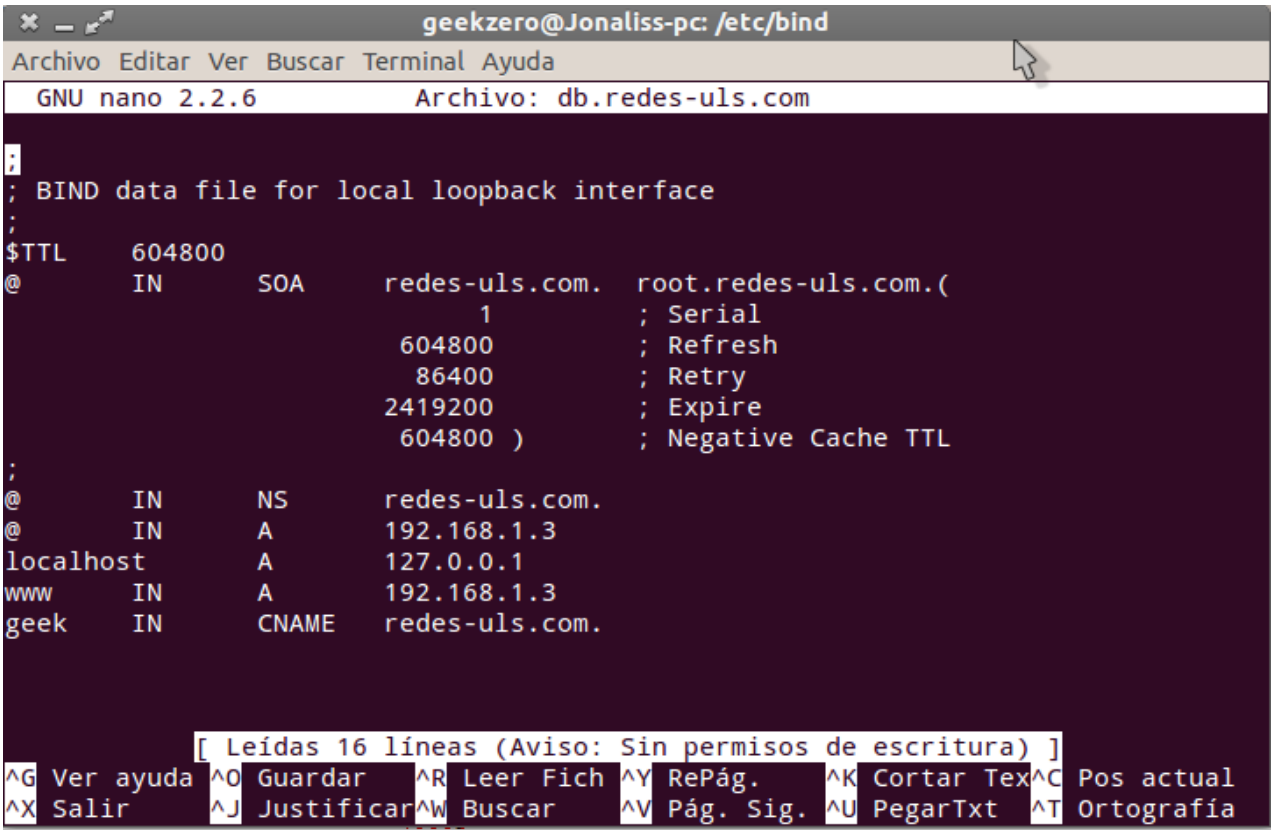
Paso 1:
Ingresar a la carpeta de configuración del bind9

```
# cd /etc/bind/
```

paso 2:
copiar las configuraciones de ejemplo a los que realmente utilizaremos:

```
root@Jonaliss-pc:/etc/bind# cp db.empty db.redes-uls.com
root@Jonaliss-pc:/etc/bind# cp db.127 db.1.168.192.in-addr.arpa
```

paso 3:
configuración de la zona de autoridad “redes-uls.com” con el editor de texto nano:



Formato del SOA

En este archivo configuramos los siguientes parámetros:
Todo lo que diga “root” por “redes-uls.com”
Agregar al final del archivo el siguiente registro:

```
@       IN      A       192.168.1.3 # Esta ip es la del servidor maestro.
```

Guardamos el archivo. Control Ctrol+O.

DESCRIPCIÓN DE LA ZANA DE AUTORIDAD (SOA).

SOA ó “inicio de autoridad” Start Of Authority es donde se define la zona primaria.

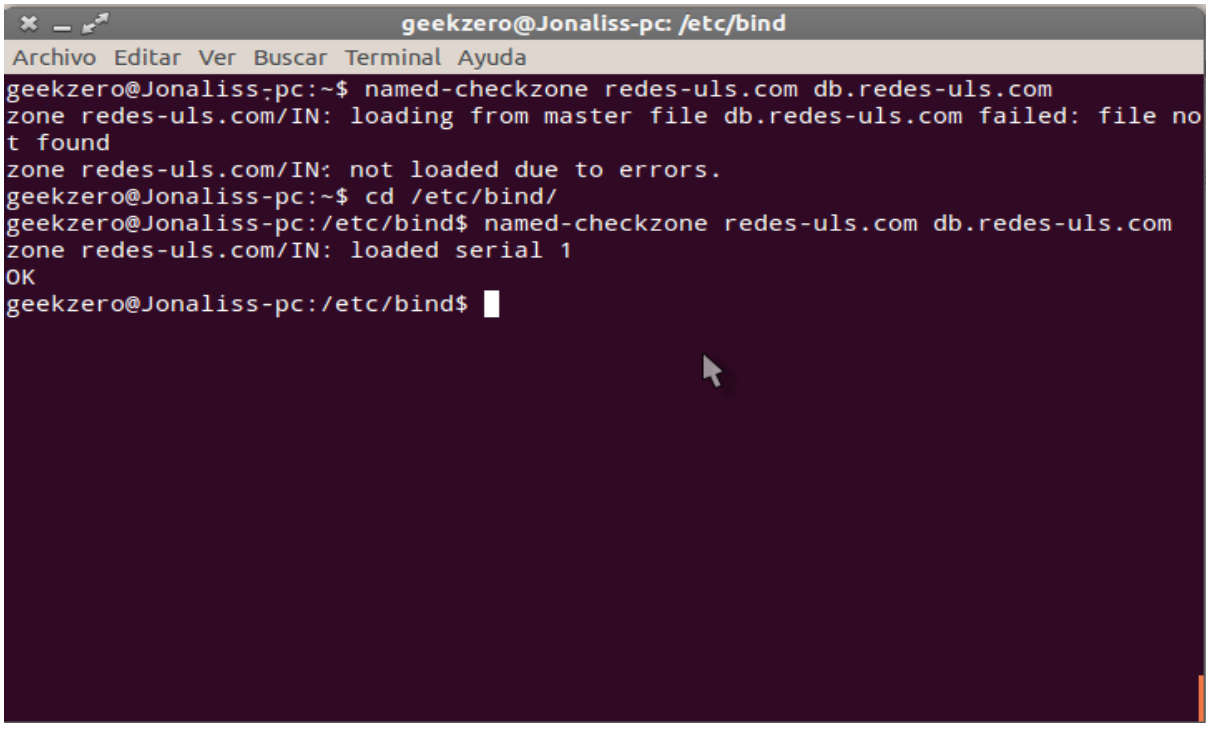
Tipo de Registro.	**Descripción.**
A (Address)	Registro de dirección que resuelve un nombre de un anfitrión hacia una dirección IPv4 de 32 bits.
AAAA	Registro de dirección que resuelve un nombre de un anfitrión hacia una dirección IPv6 de 128 bits.
CNAME (Canonical Name)	Registro de nombre canónico que hace que un nombre sea alias de otro. Los dominios con alias obtienen los sub-dominios y registros DNS del dominio original.
MX (Mail Exchanger)	Registro de servidor de correo que sirve para definir una lista de servidores de correo para un dominio, así como la prioridad entre éstos.
PTR (Pointer)	Registro de apuntador que resuelve direcciones IPv4 hacia los nombres anfitriones. Es decir, hace lo contrario al registro A. Se utiliza en zonas deResolución Inversa.
NS (Name Server)	Registro de servidor de nombres, que sirve para definir una lista de servidores de nombres con autoridad para un dominio.
SOA (Start of Authority)	Registro de inicio de autoridad, encargado de especificar el Servidor DNSMaestro (o Primario) que proporcionará la información con autoridad acerca de un dominio de Internet, dirección de correo electrónico del administrador, número de serie del dominio y parámetros de tiempo para la zona.
SRV (Service)	Registros de servicios, encargados de especificar información acerca de servicios disponibles a través del dominio. Protocolos como SIP (SessionInitiation Protocol) y XMPP (Extensible Messaging and Presence Protocol) suelen requerir registros SRV en la zona para proporcionar información a los clientes.
TXT (Text)	Registros de texto, encargados de permitir al administrador insertar

Tipo de Registro.	Descripción.
	texto arbitrariamente en un registro DNS. Este tipo de registro es muy utilizado por los servidores de listas negras DNSBL (DNS-based Blackhole List) para la filtración de Spam. Otro ejemplo de uso sería el caso de las VPN, donde suele requerirse un registro TXT, para definir una firma digital que será utilizada por los clientes.
Sn = Serial number:	es un identificador del archivo, puede tener un valor arbitrario pero se recomienda que tenga la fecha con una estructura AAAA-MM-DD y un consecutivo.
Refresh •	número de segundos que un servidor de nombres secundario debe esperar para comprobar de nuevo los valores de un registro.
Retry:	número de segundos que un servidor de nombres secundario debe esperar después de un intento fallido de recuperación de datos del servidor primario.
Expiry • •	Expiración: número de segundos máximo que los servidores de nombre secundarios retendrán los valores antes de expirarlos.
TTL	mínimo: Significa Time To Live y es el número de segundos que los registros se mantienen activos en los servidores NS caché antes de volver a preguntar su valor real.

Paso 4:
verificamos la configuración de la zona con:

```
# /etc/bind$ named-checkzone redes-uls.com db.redes-uls.com
```

luego de asegurarnos que todo esta correcto procedemos con la configuración de la zona inversa.



Paso 1:
En la carpeta “/etc/bind/” editar el archivo db.1.168.192.in-addr.arpa con:

```
# nano db.1.168.192.in-addr.arpa
```

paso 2:
Sustituir dentro del archivo todo lo que diga “root” por “redes-uls.com”

paso 3:
agregar al final del archivo el siguiente registro:
3 IN PTR redes-uls.com
donde el tres el corresponde al ultimo octeto de la ip del DNS.
Guardamos el archivo con la nueva configuración. (Ctrol+O).

paso 4:
verificamos la configuración de la zona con:

```
# /etc/bind# named-checkzone redes-uls.com db.redes-uls.com
```

el resultado serial asi:

```
zone redes-uls.com/IN: loaded serial 1
OK
```

Nota: En el nombre de la zona inversa coresponde a la red inversa donde esta conectada el DNS, en este caso es la red 192.168.1.0 y el inverso es 0.1.168.192.
el punto al final del dominio “carnet.com” no se debe omitir. El significado que tiene es que no le agregara al final de ese nombre la zona de autoridad que para el caso es “redes-uls.com” y si lo dejamos sin punto resolveriamos “redes-uls.com. Redes-uls.com” cosa que no se desea.

AGREGANDO LAS ZONAS A LA CONFIGURACION PRINCIPAL

Paso 1:
editamos el archivo “named.conf.local” dentro de la carpeta de bind.

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

zone  "redes-uls.com" {
     type master;
     file "/etc/bind/db.redes-uls.com";
    };

zone "1.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.1.168.192.in-addr.arpa";
};
```

paso 2:
Guardamos los nuevos cambios y detenemos el servicio de bind con:

```
# /etc/init.d/bind9 stop
```

paso 3:
Iniciar el servicio de bind con

```
# /etc/init.d/bind9 start
```

paso 4:
verificando que el dns responda correctamente al dominio redes-uls.com con el comando dig
resultado:

```
root@Jonaliss-pc:/etc/bind# dig @localhost redes-uls.com

; <<>> DiG 9.8.1-P1 <<>> @localhost redes-uls.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 53490
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;redes-uls.com.                 IN      A

;; ANSWER SECTION:
redes-uls.com.          604800  IN      A       192.168.1.3

;; AUTHORITY SECTION:
redes-uls.com.          604800  IN      NS      redes-uls.com.

;; Query time: 1 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Tue May 14 21:57:24 2013
;; MSG SIZE  rcvd: 61
```

también podemos verificar los logs con:

```
# /etc/bind# tail -n 100 /var/log/syslog
```

AGREGAR SUB DOMINIOS AL DNS

Paso 1:
Editar la zona de autoridad “redes-uls.com” con:

```
# nano db.redes-uls.com
```

paso 2:
Agregar al final del archivo las siguientes lineas:

```
www                     IN          A       192.168.1.3
uls                     IN          A       192.168.1.3
```

Se agregan al final de los www y uls las ips que estaran asociadas a cada uno de los dominios.

Paso 3:
guardamos los cambios.

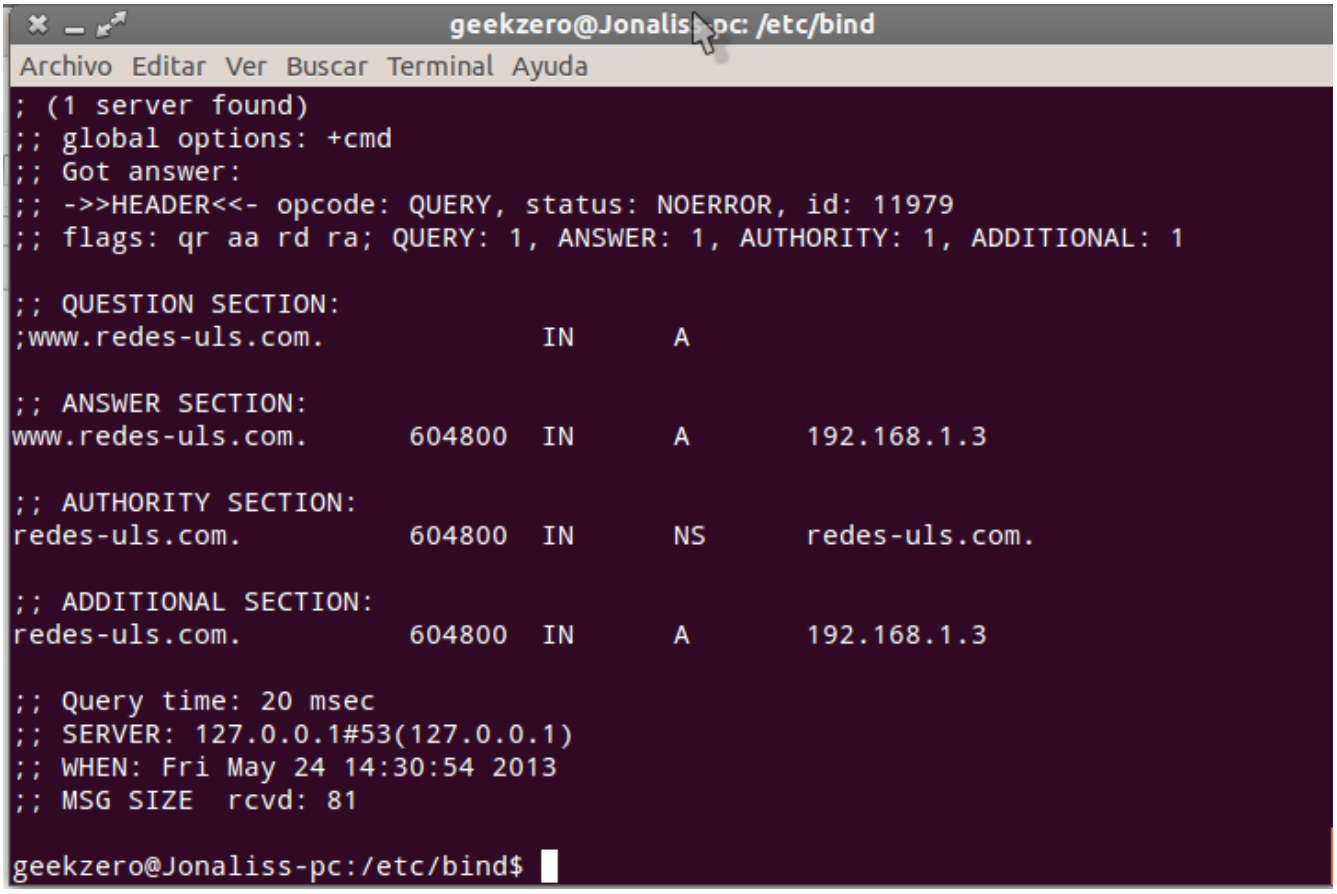
detener el servicio de bind9 con

```
# /etc/init.d/bind9 stop
```

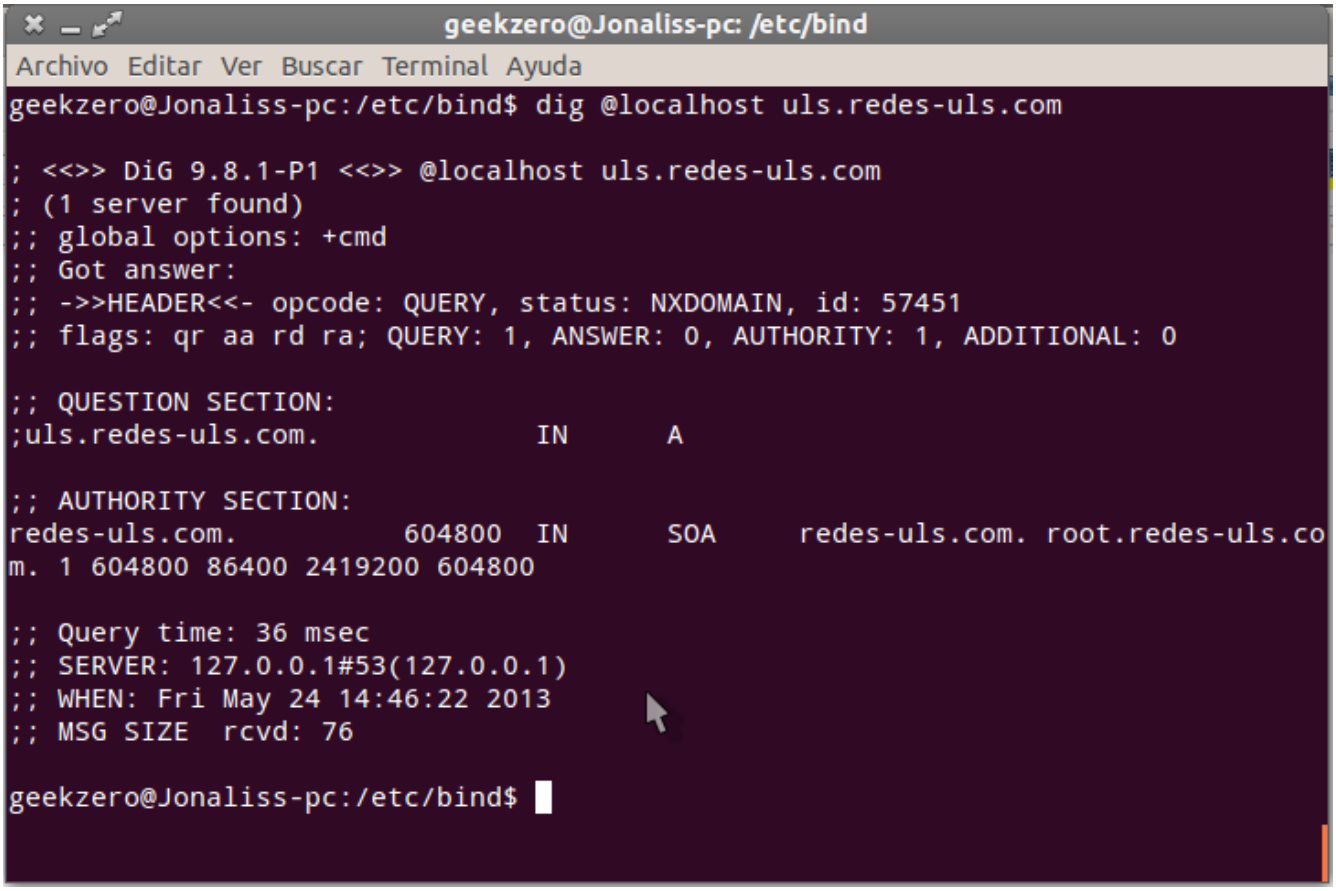
iniciar nuevamente el servicio de bind9 con

```
# /etc/init.d/bind9 start
```

verficar que el dns responda correctamente al dominio www.redes-uls.com y uls.redes-uls.com
con el comando dig:
dig @localhost www.redes-uls.com



dig @localhost uls.redes-uls.com



CONFIGURACIÓN DEL DNS SECUNDARIO

Paso 1:
En el DNS primario editar el archivo “named.conf.local” editar la zona “redes-uls.com” y dejarla de la siguiente manera:

```
zone  "redes-uls.com" {
    type master;
    file "/etc/bind/db.redes-uls.com";
    allow-query { any; };
    allow-transfer { 192.168.1.4; };
   };
```

Donde la ip debe pertenecer al DNS secundario.

Paso 2:
detener e iniciar el servicio de bind9 del dns primario con

```
#/ etc/init.d/bind9 stop
```

```
#  /etc/init.d/bind9 start
```

paso 3:
En el DNS secundario en el archivo “named.conf.local” agregar la zona “
“<redes-uls-dnsprimario>.com” y dejarlo de la siguiente manera:

```
zone  "<redes-uls-dnsprimario>.com" {
     type slave;
     file "/etc/bind/slave.<redes-uls-dnsprimario>.com";
     allow-query { any; };
     masters { 192.168.1.3; };
    };
```

Donde <redes-uls-dnsprimario> es la zona del DNS primario que se quiere configurar su esclavo y la ip correspnde al DNS primario.

Paso 4:
detener e iniciar el servicio de bind9 del DNS primario con:

```
#/ etc/init.d/bind9 stop
```

```
#  /etc/init.d/bind9 start
```

paso 5:
verificar que el DNS secundario responda correctamente al dominio “redes-uls.com con:

```
# dig @ 192.168.1.4  <redes-uls>.com
```

CONFIGURACION DE APACHE2

CONFIGURACION DEL VIRTUALHOST 1

Paso 1:
ir a la carpeta de configuración de los virtualhosts del apache en “etc/apache2/sittes-aviable/” con:

```
#cd /etc/apache2/sites-aviable/
```

paso 2:

copiar la configuración por defecto y renombrar para el virtualhosts1 “www.redes-uls.com” con:

```
# cp default www.redes-uls.com
```

paso 3:
editar el archivo “www.redes-uls.com” y cambiar la ruta “/var/www” por :

“/var/www/cms/” adicionando la siguiete linea:

ServerName www.redes-uls.com
antes del document root.

Paso 4:
habilitar el virtualhosts1 con:

```
# a2ensite www.redes-uls.com
```

paso 5:
creamos la carpeta donde estara alojado nuestro sitio con:

```
# mkdir /var/www/cms/
```

paso 6:
agregar archivos a la carpeta “cms” que hemos creado para nuestro sitio web, de igual manera podemos configurar un cms (sistema de gestor de contenidos) que sirve como nuestro sitio web.

Paso 7:
detenemos e iniciamos el servicio de apache2 con:

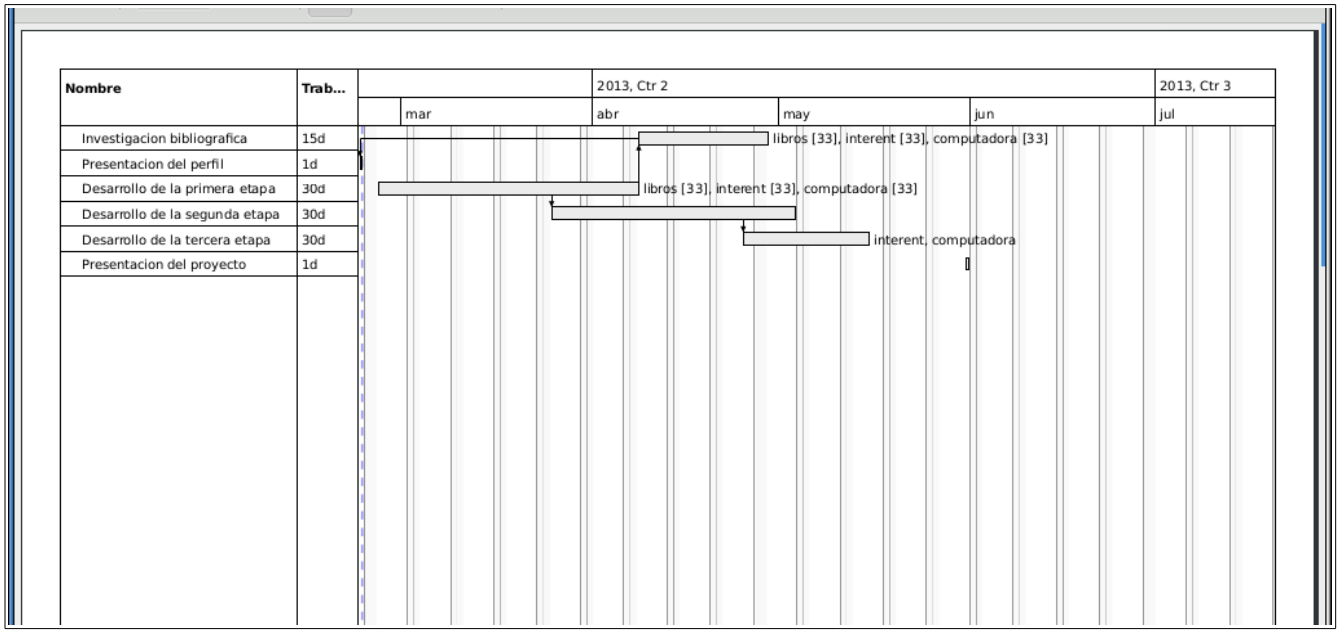
```
# /etc/init.d/apache2 stop
```
y
```
# /etc/init.d/apache2 start
```

paso 8:
Verificamos que nuestro sitio este funcionando.

LISTA DE ACTIVIDADES.

Las actividades a realizar estarán divididas en etapas.
Primero se realizara una investigación bibliográfica consultando libros e internet.
Seguidamente de la presentación del perfil del proyecto.
Luego las etapas 1,2,3 que tendrán una duración máxima de 30 días para realizarse.
Finalmente la realización o conclusión del proyecto.

DIAGRAMA DE GANTT



VIABILIDAD DEL SERVIDOR DNS PRIMARIO Y SECUNDARIO.

Un servidor casero DNS no se comparara al de un Hosting con equipo adecuado. Ellos tienen infraestructura redundante, en muchos casos con varias lineas de conexión a diferentes ISPs; el equipo que utilizan están diseñado para darle mantenimiento en el menor tiempo posible. El mayor inconveniente posible seria la refrigeración de los equipos, el lugar y espacio adecuado y posiblemente con nuestro protocolo ISP al no dar salida a internet.

Por otra parte un servicio de servidor DNS maestro, esclavo que resuelva nombres utilizando software libre seria poco costoso económicamente bastaría con tener dos computadoras con linux, una conexión a internet al menos de un 1 mega, un switch.

BIBLIOGRAFIA

http://www.taringa.net/posts/linux/16000741/Servidor-DNS-Primario-y-Secundario-con-BIND9.html

viernes 16 de febrero del 2013.

http://www.taringa.net/posts/linux/9738017/Montar-Servidor-DNS-en-Linux-CentOS.html

Domingo 18 de febrero del 2013.

http://www.slideshare.net/jessidi/dns-maestro-y-esclavo1

Lunes 19 de febrero del 2013.

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

Martes 20 de febrero del 2013.

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

http://rodeadosdetecnologia.blogspot.com/2012/11/configurar-ip-estatica-en-ubuntu-server.html

ANEXOS

Requisitos mínimos de hardware para los servidores DNS-MAESTRO-Y-ESCLAVO.

Se recomienda como mínimo un Pentium 4, a 1 GHz para un sistema de escritorio Gnu/linux ubuntu Desktop.

Tipo de instalación	RAM (mínimo)	RAM (recomendado)	Disco duro
Sin escritorio	64 Megabytes	256 Megabytes	1 Gigabyte
Con escritorio	128 Megabytes	512 Megabytes	5 Gigabytes

Tabla 1 requisitos de sistema